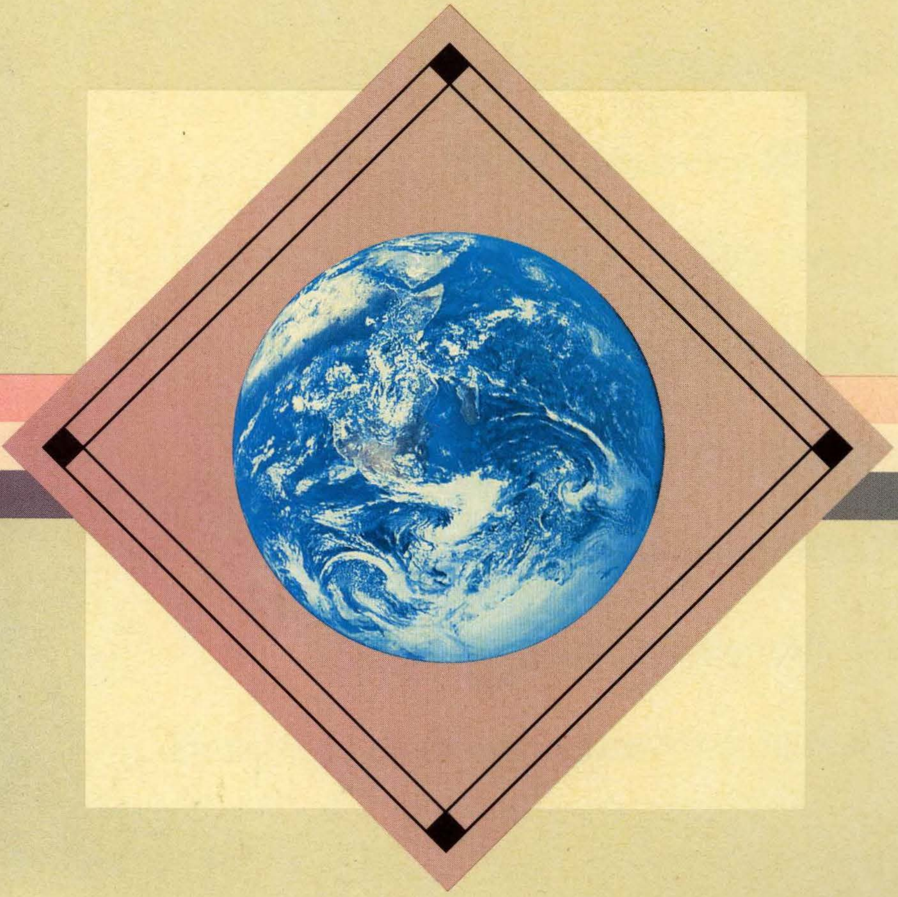
MC68605UM/AD

MC68605

X.25 PROTOCOL CONTROLLER

USER'S MANUAL



MOTOROLA

MC68605 USER'S MANUAL MOTOROLA

MC68605

X.25 PROTOCOL CONTROLLER (XPC) USER'S MANUAL

Motorola reserves the right to make changes without further notice to any products herein to improve reliability, function or design. Motorola does not assume any liability arising out of the application or use of any product or circuit described herein; neither does it convey any license under its patent rights nor the rights of others. Motorola products are not authorized for use as components in life support devices or systems intended for surgical implant into the body or intended to support or sustain life. Buyer agrees to notify Motorola of any such intended end use whereupon Motorola shall determine availability and suitability of its product or products for the use intended. Motorola and Ⓜ are registered trademarks of Motorola, Inc. Motorola, Inc. is an Equal Employment Opportunity/Affirmative Action Employer.

TABLE OF CONTENTS

Paragraph Number | Title | Page Number

Section 1
Introduction

Section 2
XPC Implementation of LAPB Procedure

Section 3
XPC Transparent Mode of Operation

TABLE OF CONTENTS (Continued)

Section 4
Internal Registers

TABLE OF CONTENTS (Continued)

Paragraph Number | Title | Page Number

Section 5
Shared Memory Structures

TABLE OF CONTENTS (Continued)

Section 6
Command Set

TABLE OF CONTENTS (Continued)

Paragraph Number — Title — Page Number

Section 7
Signal Description

TABLE OF CONTENTS (Continued)

Section 8
Bus Operation

TABLE OF CONTENTS (Concluded)

Paragraph Number | Title | Page Number

Section 9
XPC/Host Processor Interface

Section 10
Electrical Specifications

Section 11
Ordering Information and Mechanical Data

LIST OF ILLUSTRATIONS

Figure Number **Title** **Page Number**

LIST OF ILLUSTRATIONS (Continued)

LIST OF TABLES

Table Number	Title	Page Number

SECTION 1
INTRODUCTION

The MC68605 X.25 Protocol Controller (XPC) is an intelligent HCMOS communications protocol controller that implements the 1984 International Telegraph and Telephone Consultative Committee (CCITT) X.25 Recommendation, data link access procedure (LAPB). It supports full-duplex point-to-point serial communication at up to 10 megabits per second (MBPS) and relieves the host processor of managing the communications link by providing sequencing using HDLC framing, error control, retransmission based upon a cyclic redundancy check (CRC), and flow control using the receive not ready supervisory frame. The XPC directly supports the physical level interfaces (Recommendation X.21 *bis* and V-series) and also provides an efficient interface to the packet level for information and control exchange.

1.1 KEY FEATURES

Key features of the XPC include:

- Fully Implements X.25 Recommendation LAPB Procedure by Independently Generating Link Level Commands and Responses
- Option to Implement X.75 Recommendation
- Optional Transparent Operation (Monitor Mode) where XPC Provides HDLC/SDLC Framing Functions for User Generated Frames
- Performs DMA Transfer of Information Frames to and from Memory Using Two On-Chip 22-Byte FIFOs
- Primary Communication through Shared Memory Structures with a Powerful Command Set to Off-Load Data Link Management
- Flexible Rx/Tx Linked Memory Structures Minimize Host Intervention and Simplify Memory Management
- Basic (Modulo 8) and Extended (Modulo 128) Operation
- Automatic Comparison of the Programmable Local and Remote Addresses
- Detection of Programmable Timeout and Retries Limit Conditions

- 16- or 32-Bit CRC Generation and Checking
- Standard Modem Interface
- NRZ or NRZI Encoding/Decoding
- Vectored Interrupts and Status Reporting
- Built-In Diagnostics Provide Local Loopback and External Loopback Testing
- Up to 10-Mbps Synchronous Serial Data Rate
- 12.5- and 10-MHz System Clock Versions
- 8- and 16-Bit Data Bus Support
- 32-Bit Address Bus with Virtual Address Capability
- M68000 Family Asynchronous Bus Structure
- Programmable Byte Ordering of Data for Alternate Memory Organization Schemes

1.2 GENERAL DESCRIPTION

The XPC supports high-speed X.25 communications between host computers, between host computers and remote units, and between remote units. The XPC also supports a transparent operation mode which does not apply the LAPB procedure. Data is passed between the XPC and the host processor through shared memory structures. This permits a minimum command set for host processor/XPC communication. Additionally, the XPC is a full MC68000 bus master, providing on-chip DMA capability for management of memory tables and frame buffers. Since the XPC data bus interface is configurable, the XPC can handle both 8-bit and 16-bit data transfers.

When the X.25 mode is selected by the user, the XPC is configured as a combined station for full-duplex point-to-point communication. The XPC supports a nonoperational mode and two operational modes as defined by the LAPB procedure. The nonoperational mode is asynchronous disconnect mode (ADM). In this balanced data link mode, the combined station is logically disconnected from the data link and is not permitted to transmit or accept information. Operational modes include asynchronous balanced mode (ABM) and asynchronous balanced mode extended (ABME). A balanced data link allows a combined station to send a command or initiate a response frame transmission without receiving explicit permission from the other station. In ABM/ABME the XPC performs the following operations:

1. Transmission of a chain of information (I) frames when instructed by the host,
2. Transmission of supervisory (S) frames as defined by the X.25 LAPB Recommendation,
3. Transmission of unnumbered (U) commands as required or when instructed by host, and
4. Transmission of unnumbered (U) responses as defined by the X.25 LAPB Recommendation.

When the transparent mode is selected, the XPC can be configured as a primary, a secondary, or a combined station for full-duplex operation. The XPC can support any HDLC/SDLC-defined operational mode. All frames are user-generated and are transmitted only when instructed by the host.

A typical system configuration using the XPC with a M68000 Family MPU is shown in Figure 1-1. A block diagram of the MC68605 is shown in Figure 1-2.

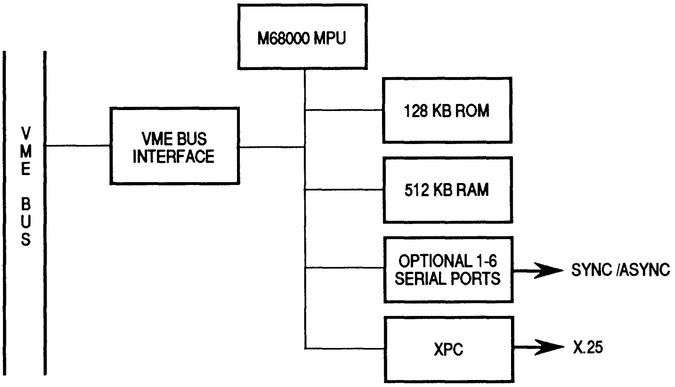


Figure 1-1. XPC System Configuration

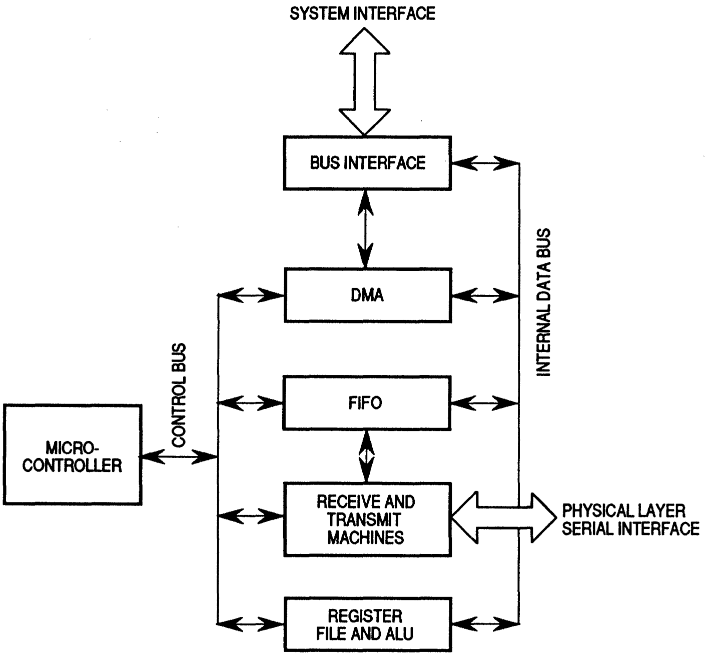


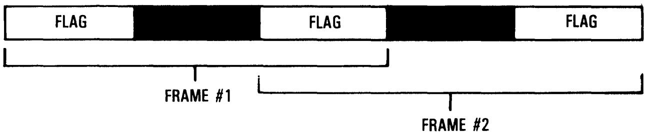
Figure 1-2. MC68605 Block Diagram

1.3 X.25 FRAME DESCRIPTION

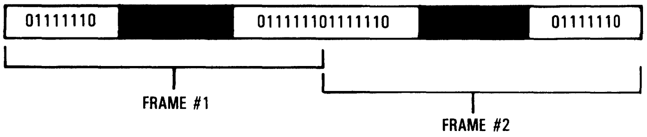
Before looking in detail at the modes of operation, internal registers of the XPC, commands, and the shared memory structures, a quick review of the X.25 frame structure will be covered. There are three types of X.25 frame formats. These are the information (I) frame, supervisory (S) frame, and unnumbered (U) frame. Information frames transfer data between the network nodes. Supervisory frames pass data link supervisory and control information, such as information frame reception acknowledgement, retransmission of information frame requests, and temporary information frame transmission suspension requests. Unnumbered frames provide additional data link control functions. Although the unnumbered frame format contains no sequence numbers, the format does contain a poll/final bit to request a response frame.

The beginning and end of an X.25 frame is identified by a flag sequence of 01111110. The XPC transmits a minimum of two flags between frames.

The XPC permits sharing of the flag sequence for back-to-back frames.



The XPC also permits a shared zero within the two flag sequences for back-to-back frames.



The address field identifies a frame as either a command or a response. An XPC-generated command frame contains the remote address. An XPC-generated response frame contains the local station address.

FRAME TYPE	**ADDRESS FIELD**
Rx Command	Local Address
Rx Response	Remote Address
Tx Command	Remote Address
Tx Response	Local Address

X.25 defines an extended and basic frame format. These formats are shown in Figure 1-3. The difference between the extended and basic control format is the length of the control field. The basic control field contains eight bits and the extended control field contains 16 bits. Unnumbered frames always have 8-bit control fields. The extended control field format allows a maximum of 127 outstanding frames (Modulo 128) on the line; whereas, the basic control field format provides a limit of seven outstanding frames (Modulo 8). Figure 1-4 shows the basic control field format for X.25 I, S, and U frames, and Figure 1-5 shows the extended control field format for I, S, and U frames.

Address, commands, responses, and sequence numbers are transmitted beginning with the low-order bit. For example, the sequence number bit with the weight 2° is transmitted first. Within the information field, the XPC transmits the low-order bit of the low-order data byte first, regardless of the organization of data in memory. The FCS is transmitted beginning with the coefficient of the highest order exponential term.

The various commands and responses defined for the three X.25 frame types are shown in Figure 1-6. Figure 1-7 illustrates which X.25 frame types are host generated.

FLAG	ADDRESS	CONTROL	INFORMATION	FRAME CHECK SEQUENCE	FLAG
01111110	8 BITS	8 BITS	N BITS[1]	16 BITS[2]	01111110

(a) X.25 Basic Frame Format

FLAG	ADDRESS	CONTROL	INFORMATION	FRAME CHECK SEQUENCE	FLAG
01111110	8 BITS	16 BITS[3]	N BITS[1]	16 BITS[2]	01111110

(b) X.25 Extended Frame Format

NOTES:
1. XPC information field limited to 64K bytes.
2. XPC also provides optional 32-bit FCS.
3. Unnumbered frames have an 8-bit control field in extended mode.

Figure 1-3. X.25 Frame Formats

	BIT ORDER							
FRAME TYPE	1	2	3	4	5	6	7	8
I FRAME	0	N(S)			P	N(R)		
S FRAME	1	0	S	S	P/F	N(R)		
U FRAME	1	1	M	M	P/F	M	M	M

N(S) — Transmitting Station Send Sequence Number
N(R) — Transmitting Station Receive Sequence Number
S — Supervisory Function Bits
M — Modifier Function Bits
P/F — Poll/Final: Poll Bit for Command Frames
Final Bit for Response Frames

Figure 1-4. Basic Control Format

	BIT ORDER															
FRAME TYPE	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
I FRAME	0	N(S)							P	N(R)						
S FRAME	1	0	S	S	X	X	X	X	P/F	N(R)						
U FRAME	1	1	M	M	P/F	M	M	M								

N(S) — Transmitting Station Send Sequence Number
N(R) — Transmitting Station Receive Sequence Number
S — Supervisory Function Bits
M — Modifier Function Bits
P/F — Poll/Final: Poll Bit for Command Frames
Final Bit for Response Frames
X — Reserved and Set to Zero

Figure 1-5. Extended Control Format

COMMANDS		RESPONSES	
I	INFORMATION		—

(a) Sequenced Information

COMMANDS		RESPONSES	
RR	RECEIVE READY	RR	RECEIVE READY
RNR	RECEIVE NOT READY	RNR	RECEIVE NOT READY
REJ	REJECT	REJ	REJECT

(b) Supervisory Format

COMMANDS		RESPONSES	
SABM	SET ASYNCHRONOUS BALANCED MODE		—
SABME	SET ASYNCHRONOUS BALANCED MODE EXTENDED		—
DISC	DISCONNECT		—
	—	UA	UNNUMBERED ACKNOWLEDGEMENT
	—	DM	DISCONNECT MODE
	—	FRMR	FRAME REJECT

(c) Unnumbered Format

Figure 1-6. Commands and Responses

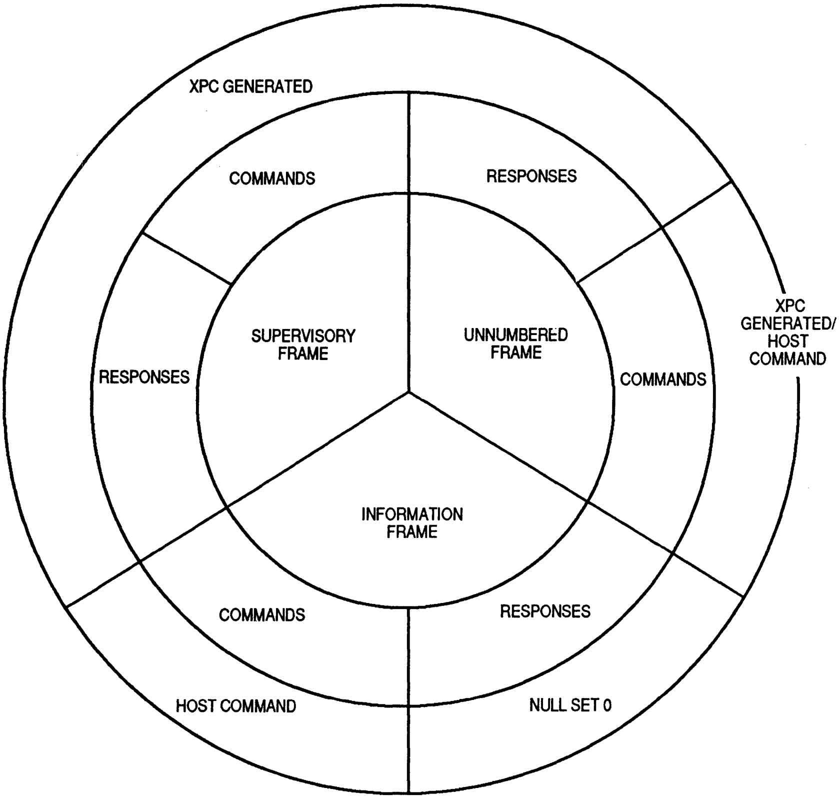


Figure 1-7. X.25 Frame Originate

SECTION 2
XPC IMPLEMENTATION OF LAPB PROCEDURE

This section describes the procedures used by the XPC in the X.25 mode to implement the LAPB procedure.

2.1 INITIALIZATION PROCEDURE

The XPC enters the initialization procedure as the result of a hardware or software reset. During this initialization, the station table address and function code (FC), system configuration information, and the XPC interrupt vector should be loaded by the XPC under the direction of the host, as shown in the sample program below. The internal XPC registers directly accessed during the initialization procedure are the command register (CR), data register (DR), interrupt vector register (IV), and semaphore register (SR).

```
RESET
   Repeat: Read Semaphore Register Until it is "FF"
 Write CR: Set Hardware Configuration
   Repeat: Read Semaphore Register Until it is "FF"
 Write CR: Set Station Configuration
   Repeat: Read Semaphore Register Until it is "FF"
 Write DR: 4-Bit Function Code Value for Station Table Access
 Write CR: Load Function Code
   Repeat: Read Semaphore Register Until it is "FF"
 Write DR: 32-Bit Address of Station Table
 Write CR: Load Station Table Pointer (STP)
   Repeat: Read Semaphore Register Until it is "FF"
 Write IV: Interrupt Vector
 Write CR: Load Station Table Parameters
   Repeat: Read Semaphore Register Until it is "FF"
```

Note that the XPC will not come out of hardware or software reset without the system clock and the transmit clock. The transmit clock is used to initialize the serial section of the chip.

When the XPC receives a SABM/SABME command frame from the remote station and the XPC can enter the information transfer phase, the XPC transmits an unnumbered acknowledge (UA) response frame, clears its internal V(R) and V(S) state variables in the V(R) and V(S) frame sequence registers respectively, and (re)enters operational mode. If a SABM/SABME command frame is received and the XPC cannot enter the information transfer phase, the XPC transmits a DM response and considers that the link is not set up.

When the link setup procedure is initiated as a result of a host processor command or link exception, the XPC begins by transmitting a SABM or SABME command frame, based on the value of the extended control (ECNT) bit in the station configuration register. The XPC also starts timer T1 to determine when the programmed time permitted for a reply to be received has elapsed and zeros the retries count register.

2.2 LINK SETUP PROCEDURE

Upon completion of the initialization routine, the XPC may be connected to the link. The XPC enters connect mode by executing a start link command from the host processor or by receiving a set asynchronous balanced mode/set asynchronous balanced mode extended (SABM/SABME) command frame from the remote station. The XPC never transmits an unsolicited disconnect mode (DM) response frame to request the remote station to initiate link setup.

When the XPC receives a SABM/SABME command frame from the remote station and the XPC can enter the information transfer phase, the XPC transmits an unnumbered acknowledge (UA) response frame, clears its internal V(R) and V(S) state variables in the V(R) and V(S) frame sequence registers respectively, and (re)enters operational mode. If a SABM/SABME command frame is received and the XPC cannot enter the information transfer phase, the XPC transmits a DM response and considers that the link is not set up.

When the link setup procedure is initiated as a result of a host processor command or link exception, the XPC begins by transmitting a SABM or SABME command frame, based on the value of the extended control (ECNT) bit in the station configuration register. The XPC also starts timer T1 to determine when the programmed time permitted for a reply to be received has elapsed and zeros the retries count register.

Upon reception of a UA response frame, the XPC resets its frame sequence registers V(R) and V(S), stops timer T1, and enters operational mode. Alternately, upon reception of a DM response, the XPC will stop timer T1 and consider that the link is not set up. Frames other than UA, DM, SABM/SABME, and disconnect (DISC) will be ignored. The reception of SABM/SABME or DISC is a collision on unnumbered command frames as discussed in **2.3 COLLISION OF UNNUMBERED COMMANDS**.

If timer T1 expires before reception of an UA response frame from the remote station, the XPC retransmits the SABM/SABME command frame, restarts timer T1, and increments the retries count register. If the retries count register becomes equal to the retries limit register, the XPC stops the link setup procedure and reports the status to the host processor.

2.3 COLLISION OF UNNUMBERED COMMANDS

If the sent and received unnumbered command frames are the same, the XPC and the remote station send a UA response at the earliest opportunity. The XPC then enters the indicated operational mode, after receiving the UA responses from the remote.

If the sent and received unnumbered command frames are different, the XPC and the remote station enter the asynchronous disconnect mode. A DM response frame is transmitted by both stations at the earliest opportunity.

2.4 INFORMATION FRAME TRANSMISSION

After the XPC enters asynchronous balanced mode (ABM) or asynchronous balanced mode extended (ABME), the host processor can instruct the XPC to transmit a chain of information frames by issuing the load transmit table pointer command. In response, the XPC loads the transmit table pointer and the transmit table function code from the station table into its internal registers. Next, the XPC loads the first transmit buffer pointer, transmit buffer function code, and transmit buffer count from the transmit frame specification table into the corresponding XPC registers. Now the XPC is ready to build the first frame.

The remote address is copied from the remote address register into the XPC transmit FIFO. Next, the control field is generated internally and placed in the FIFO. The information field pointed to by the transmit buffer pointer register is then read from the memory buffer into the transmit FIFO until the transmit buffer count is satisfied. A frame check sequence is attached to complete the frame. Zero insertion is performed throughout the transmission. After frame transmission, V(S) is incremented and timer T1 is started (if it is not already running).

This transmission sequence repeats for each frame until the end of the transmit chain is reached or until the outstanding frames limit is reached. The XPC continues to transmit any available information frames even when the XPC receiver is in the busy condition. The XPC prematurely terminates frame transmission if a stop link command interrupts the information frame transmission or an error condition arises.

Transmission begins when six bytes are present in the transmit FIFO. Transmission can begin when less than six bytes are present in the FIFO, if the entire frame is less than six bytes in length. Between frames, the XPC transmits the user-selected number of pad flags. Additional pad flags are transmitted if the required number of bytes is not present in the transmit FIFO for transmission to begin. While transmitting an information frame, the XPC requests the bus when there are at least six empty bytes in the transmit FIFO.

2.4.1 Receiving Acknowledgement

When the XPC correctly receives an I frame or an S frame, the sequence number N(R) contained in the control field of this frame will acknowledge all I frames previously transmitted with sequence numbers up to and including the received N(R) − 1. The XPC stops timer T1 when it correctly receives an N(R) higher than the last received N(R), which acknowledges some I frames, or when the XPC receives a REJ frame with the N(R) equal to the last received N(R). If timer T1 is stopped by the reception of an I, RR, or RNR frame and there are outstanding I frames still unacknowledged, the XPC will restart timer T1.

2.4.2 Waiting Acknowledgement

Timer T1 is started after a frame has been transmitted to check that an acknowledgement for the frame is received before the programmed timeout value is reached. If timer T1 expires while waiting for acknowledgement from the remote station, the XPC will enter the timer recovery condition. The XPC will send a receive ready (RR) with the poll (P) bit set to one if receive buffers are available or receive not ready (RNR) with the P bit set to one if no receive buffers are available. The XPC then increments the retries count register and restarts timer T1.

If timer T1 expires while in the timer recovery condition, the XPC will increment the retries count register, restart timer T1, and transmit the appropriate supervisory command frame with the P bit set to one. If the retries count register becomes equal to the retries limit register, the XPC will initiate a link resetting procedure as described in **2.2 LINK SETUP PROCEDURE**.

If, while in the timer recovery condition, the XPC correctly receives a S frame with the P/F bit set to zero and with a valid N(R), the timer recovery condition is not cleared. The XPC will use the incoming N(R) to acknowledge previously transmitted I frames. If, while in the timer recovery condition, the XPC correctly receives a S frame with the P/F bit set to zero or an I or S frame with the P bit set to one and with the N(R) equal to the value of V(S) (acknowledging all transmitted I frames), the XPC will not clear the timer recovery condition nor stop timer T1.

Furthermore, if the received S frame is a REJ frame with the P/F bit set to zero and with a valid N(R), the XPC will ignore the retransmission request and wait for a supervisory frame with the F bit set to one. If the received S frame is a REJ frame with the P bit set to one and with a valid N(R), the XPC will immediately transmit an appropriate supervisory response frame with the F bit set to one. The request for retransmission is ignored until the XPC receives a S frame with the F bit set to one.

When the XPC receives a supervisory frame with the F bit set to one and with the N(R) within the range from the value of the last received N(R) to the current value of the send state variable V(S) inclusive, then the XPC stops timer T1, sets its send state variable to the value of the received N(R), updates the transmit table pointer register to point to the appropriate transmit table block, and exits timer recovery mode. The XPC will resume I frame transmission or retransmission, as appropriate.

2.4.3 Receiving an REJ Frame

When a REJ frame is received, the XPC compares its send state variable V(S) to the N(R) in the control field of the received REJ frame. If the N(R) in the received REJ frame is equal to V(S), the XPC transmits the corresponding I frame when it is available. If the received N(R) is less than V(S), the XPC will begin sequential retransmission of the specified frame(s), according to the X.25 Recommendation. If the REJ frame received from the remote station is a command frame with the P bit set to one, the XPC will transmit a RR, RNR, or REJ response with the F bit set to one before (re)transmitting the requested I frame(s).

2.4.4 Receiving an RNR Frame

When a RNR frame is received, the XPC waits for timer T1 to expire and then transmits a supervisory command frame with the P bit set to one. Timer T1 is then started to determine if there is any change in the receive status of the remote station. If the remote station responds with a RNR supervisory response frame with the F bit set to one, indicating the continuance of the busy condition, the XPC repeats the above sequence the user-specified number of times (retries limit) and then initiates link reset. If the remote station responds with a RR or REJ frame, indicating the clearance of the busy condition, the XPC stops timer T1 and begins (re)transmission as appropriate.

2.5 INFORMATION FRAME RECEPTION

The host processor enables information reception by instructing the XPC to load receive table pointer. The XPC will load the receive table pointer and function code into its internal registers. Next, the receive buffer pointer and function code and the receive buffer count are loaded into the corresponding XPC registers. The XPC is now ready to receive information frames.

The address field of an incoming I frame is compared to the local address register and the remote address register. If the address does not match the local or remote address, the frame is ignored. If the address field matches the remote address, a frame reject (FRMR) is transmitted and the W (invalid or unimplemented control field) bit of the frame reject descriptor register (FRD) is set. If the address field matches the local address, the frame is accepted by the XPC, and the received N(R) acknowledges previously transmitted I frames.

Next, the send sequence number N(S) of the incoming frame is compared to the XPC internal receive state variable V(R). If the frame is in sequence, then the information field is transferred through the receive FIFO to the receive memory buffer. Out-of-sequence frames are rejected.

Lastly, the XPC performs a CRC check on the incoming information frame. If an error-free frame is received, the XPC acknowledges the frame reception with a supervisory frame (RR or RNR) or with an updated N(R) in the next information frame.

Zero deletion is performed throughout the reception process. The XPC requests the bus when there are six bytes in the receive FIFO. Only a single frame can reside in the receive FIFO. Frames are received in sequence as long as memory buffers are available.

2.5.1 Invalid Frame Condition

Invalid frames are discarded. An invalid frame is defined as a frame that:

1. Is not properly bounded by two flags,
2. Contains fewer than 32/48 bits between flags in modulo 8 operation (basic) with 16-/32-bit CRC, contains fewer than 40/56 bits between flags for frames with sequence numbers in modulo 128 (extended) operation with 16-/32-bit CRC, or contains fewer than 32/48 bits between flags for frames without sequence numbers in modulo 128 operation with 16-/32-bit CRC,
3. Contains a FCS error, or
4. Contains an address other than the local or remote station address.

The XPC does provide a means to accept frames with CRC error test/diagnostic purposes. If the CRC bypass option (CRCNOA) bit in the option bits register is set to one, then a frame containing a CRC error will be accepted. If the CRCNOA bit is set to zero and a frame is received with a CRC error, the XPC ignores the frame, and receive buffer will be reused for the next frame.

2.5.2 Receiving an Out-of-Sequence I Frame

When the XPC receives a valid I frame whose send sequence number N(S) is not equal to the internal receive state variable V(R), the information field will be discarded. The XPC transmits an REJ frame with the N(R) field set equal to V(R) to initiate retransmission, beginning with the next expected frame. Only one REJ condition is established at a time. A REJ condition is cleared when the requested I frame is received.

2.5.3 Frame Reject Mode

The XPC enters the frame reject mode (FRMR) on reception of an error-free frame and the frame contains:

1. Unimplemented or illegal control field/unrecognized frame/nonzero "X" bits in the control field (extended control, S frame),
2. Supervisory frame or unnumbered frame with an information field which is not permitted,

3. Invalid N(R), or
4. Information field too long to fit the available memory buffer.

If the X.75 option is selected (bit 8 in the option bits register is set), then the XPC will also enter frame reject mode on reception of the following:

1. A supervisory frame with the F bit set to one, except during a timer recovery condition or except as a reply to a command sent with the P bit set to one,
2. An unexpected UA or DM response, or
3. An information frame with an invalid N(S).

Once the XPC sends the FRMR response frame, no additional information or supervisory frames are accepted until the FRMR condition is cleared. The XPC stops all information frame transmission activities and updates status information held in the station table, which includes the reason for the FRMR response.

The XPC monitors the poll/final (P/F) bit of all incoming frames. When a frame is received with the P bit set to zero or when T1 expires, the XPC sends a FRMR response again with the F bit set to zero. If the received frame has a P bit set to one, the XPC responds with the FRMR frame with the F bit set to one. If the retries count register becomes equal to the retries limit register before the FRMR condition is cleared, the XPC will initiate the link resetting procedure described in **2.2 LINE SETUP PROCEDURE**. The XPC also initiates link reset when a FRMR response frame is received.

The FRMR condition is cleared when the XPC receives or transmits a SABM/SABME or DISC command or a DM response. Upon receiving a SABM/SABME or DISC command frame, the XPC sends an UA response and enters the indicated operation mode. The XPC also clears both internal state variables and disables the transmit table pointer register, so that the host processor must reload this register to resume transmission. If a DM or FRMR response frame is received, the XPC initiates link setup by transmitting SABM/SABME. The XPC also writes the received FRMR information field to the station table and updates status. The XPC transmits a DISC command frame to clear the FRMR condition only as the result of a stop link command from the host.

2.5.4 Idle Channel Condition

When the XPC detects a channel idle condition of at least 15 consecutive ones, it reports the channel idle status to the host processor. The XPC does not implement timer T3 as described in 1984 CCITT X.25 Recommendation, which would allow the detection of excessive idle periods. This timer must be implemented by higher layer software, if desired.

2.5.5 Busy Condition

The XPC enters the busy condition when it is unable to continue to receive I frames due to internal constraints, such as receive buffer limitations. A RNR frame is transmitted at the earliest opportunity. I frames pending transmission by the XPC are transmitted prior to and following the RNR frame. The XPC clears the busy condition following a load RTP command from the host.

2.6 DISCONNECT MODE

The XPC enters asynchronous disconnect mode (ADM) when:

1. Initialization of the XPC is complete,
2. The XPC transmits an UA response after receiving a DISC command from the remote station,
3. The XPC receives a DM with F set to one or a UA response after transmitting a DISC command,
4. The XPC receives a DM with F set to one response after transmitting a SABM/SABME command,
5. The retries limit is reached during the link setup or link disconnect procedures, or
6. DISC and SABM/SABME command frames collide.

In disconnect mode, the XPC initiates link setup after receiving a start link command from the host processor. If the XPC receives a DISC command frame, it transmits a DM response and remains in disconnect mode. If the XPC receives a SABM/SABME command frame from the remote station, an UA response is transmitted. When any other command frame is received with the P bit set to a one, the XPC transmits a DM response with the F bit set to one. Other frames received while in disconnect mode are ignored.

2.7 LINK DISCONNECT PROCEDURE

The XPC initiates a link disconnect by transmitting a DISC command frame and starting timer T1. All received frames, except SABM/SABME or DISC commands and UA or DM responses, are ignored. Upon reception of a UA response frame, the XPC enters disconnect mode and reports its status to the host processor. Upon reception of a DM response with F set to one from the remote station, indicating that the remote station is already in the disconnect mode, the XPC stops timer T1 and enters the disconnect mode. The reception of an SABM/SABME or DISC command frame is considered a collision situation that is resolved as described in **2.3 COLLISION OF UNNUMBERED COMMANDS**.

If timer T1 expires before reception of a UA or DM with F set to one response frame from the remote station, the XPC retransmits the DISC command frame, restarts timer T1, and adds one to the retries count register. If the retries count register becomes equal to the retries limit register, the XPC stops the link disconnect procedure, enters disconnect mode, and reports its status to the host processor.

2.8 STATE TABLE

The state table (see Figure 2-1 found on a foldout page at the back of this document) defines the various XPC states based on command frames received (no errors), response frames received (no errors), and miscellaneous inputs received. For example, referring to Figure 2-1, if the command frame received was a RR with the P bit set to one while in the remote station busy condition (state 9), then the XPC responds with RR with F set to one and changes to information transfer (state 5).

SECTION 3
XPC TRANSPARENT MODE OF OPERATION

The XPC transparent mode of operation can be used to implement a variety of bit oriented protocols. This section describes the XPC transparent mode of operation.

3.1 INITIALIZATION PROCEDURE

The XPC enters the initialization procedure as the result of a hardware or software reset. During initialization, the station table address and function code, system configuration information, and the XPC interrupt vector should be loaded by the XPC under the direction of the host, as shown in the following sample program:

```
RESET
  Repeat: Read Semaphore Register Until it is "FF"
Write CR: Set Hardware Configuration
  Repeat: Read Semaphore Register Until it is "FF"
Write CR: Set Station Configuration
  Repeat: Read Semaphore Register Until it is "FF"
Write DR: 4-bit Function Code Value for Station Table Access
Write CR: Load Function Code
  Repeat: Read Semaphore Register Until it is "FF"
Write DR: 32-bit Address of Station Table
Write CR: Load Station Table Pointer (STP)
  Repeat: Read Semaphore Register Until it is "FF"
 Write IV: Interrupt Vector
Write CR: Load Preset Values
  Repeat: Read Semaphore Register Until it is "FF"
```

Note that the XPC will not come out of hardware or software reset without the system clock and the transmit clock. The transmit clock is used to initialize the serial section of the chip.

When function codes are not implemented in the system, the host is not required to load the station table function code into the data register and then issue a load FC command. Function codes are usually found in virtual memory systems. Function codes divide memory into user data, user program, supervisor data, and supervisor program spaces. The XPC provides the user-specified function codes during XPC DMA cycles. However, the XPC performs no checking on the validity of the function code values.

During transparent operation, the following registers are not used and do not need to be initialized in the station table.

Option Bits Register	Word 0
Time Out Preset	Word 1
Time Scale Divider	Word 2 — high byte
Outstanding Frames Limit	Word 3 — high byte
Retries Count	Word 3 — low byte
Local Address	Word 8 — low byte
Remote Address	Word 9 — low byte

Once the initialization process is complete, the XPC is disconnected. The host can write commands to the XPC. However, it is always necessary to check the semaphore register for "FF" to ensure that the XPC is ready to accept the next command. The load addresses command should not be issued to the XPC, since, after the load addresses command, the XPC monitors the RxD line and could respond to a received SABM and enter X.25 operation.

3.2 ENTERING TRANSPARENT OPERATION

Transparent operation is entered when the host issues the monitor command. After the monitor command, the XPC asserts $\overline{\text{RTS}}$, transmits flags, and monitors RxD. Since handshaking between nodes is not possible before the monitor command is executed, the host processor at each node must issue the monitor command in order for the two nodes to communicate.

The V(S) and V(R) registers are zeroed upon entering transparent operation. During transmission and reception of frames, the V(S) and V(R) registers are incremented modulo 8 or modulo 128 depending on the ECNT (extended control) bit in the station configuration register. V(S) and V(R) are incremented regardless of frame type. User-generated control fields may be any length since the XPC does not analyze the address and control fields of transmitted and received frames.

The following eleven commands are valid during transparent operation:

- Load RTP
- Load TTP
- Continue Transmit
- Clear Tx/Link Status
- Clear Rx/Host Status
- Clear Status
- Update Status
- Dump Parameters
- Dump Registers
- End Monitor
- Reset

After executing the dump parameters command or the dump registers command in transparent operation, the following locations in the station table and dump area are not valid:

Frame Reject Descriptor	Word 21 — low byte	station table
First Unacknowledged Pointer	Words 24 and 25	station table
Received FRMR Information Field	Words 38, 39, and 40	station table
Local Address	Word 14 — low byte	dump area
Remote Address	Word 15 — low byte	dump area
Frame Reject Descriptor	Word 16 — low byte	dump area
Time Out Preset	Word 20	dump area
Outstanding Frames Limit	Word 21 — high byte	dump area
Retries Limit	Word 21 — low byte	dump area
Last Received N(R)	Word 22 — high byte	dump area
Retries Count	Word 22 — low byte	dump area

3.3 FRAME TRANSMISSION

After the monitor command is issued, the XPC begins transmission of frames only after receiving a load transmit table pointer command from the host. All frames are user-generated and may contain user-provided address, control, and data fields or may contain only a data field. After the host issues the load transmit table pointer command, the XPC loads the transmit table pointer and the transmit table function code from the station table into its internal registers. Next, the XPC loads the first transmit buffer pointer, transmit buffer function code, and transmit buffer count from the transmit frame specification table into the corresponding XPC registers. Now the XPC is ready to transmit the first frame.

The frame pointed to by the transmit buffer pointer register is read from the memory buffer into the transmit FIFO until the transmit buffer count is satisfied. An XPC-generated frame check sequence is then attached to complete the frame. After each frame transmission, the internal V(S) register is incremented without regard to the frame type. This transmission sequence repeats for each frame until the end of the transmit chain is reached. Zero insertion is performed throughout the transmission process.

In transparent operation, the XPC transmits frames until the end of the transmit specification table is reached. After the last byte of the last frame is loaded into the Tx FIFO, the XPC sets the IFAK (information frames acknowledged) bit in the Tx/link status register to indicate the end of the transmit table. The XPC also clears the TBEN (transmit buffer enable) and then sets the NEWMD (new mode) bit in the Tx/link status register. The XPC does not analyze any incoming frames for acknowledgments or link control information during transparent operation. The only errors reported in the Tx/link status register are address error, bus error, clear-to-send lost, and underrun.

Note that frames can be added to the transmit queue dynamically as the XPC is transmitting. After adding entries to the end of the transmit queue, the host must set the EOT bit in the last added entry and clear the EOT bit at the previous end of table. If the XPC has already read the EOT bit, then the added entries will not transmitted. In this case, the continue transmit command should be issued to XPC.

Transmission begins when six bytes are present in the transmit FIFO. Transmission can begin when less than six bytes are present in the FIFO if the entire frame is less than six bytes in length. Between frames, the XPC transmits the user-selected number of pad flags. Additional pad flags are transmitted if the required number of bytes are not present in the transmit FIFO for transmission to begin. While transmitting a frame, the XPC requests the bus when there are at least six empty bytes in the transmit FIFO.

3.4 FRAME RECEPTION

The host processor enables frame reception by instructing the XPC to load receive table pointer. The XPC then loads the receive table pointer and function code into its internal registers. Next, the receive buffer pointer and function code and the receive buffer count are loaded into the corresponding XPC registers. The XPC is now ready to receive frames.

The XPC does not analyze the address and control fields of incoming frames, but does perform a CRC check on incoming frames. After the flags are stripped off, the entire frame, including CRC, is written into the current receive buffer, and the RXI (received information frame) bit is set in the Rx/host status register. If a frame is received that is nonoctet aligned or that has a CRC error, the XPC sets the E bit in that frame's receive specification block. The shortest frame that can be received is two bits. Note that CRC checking cannot be performed on frames shorter than 16 bits, and no CRC error indication will be given.

To set the E bit, the XPC writes the receive status and receive buffer function code bytes. The L(link) bit is written as zero, regardless of its previous state. The EOT bit and FC bits are unchanged. Even though the link bit is cleared during the write cycle, the XPC has previously read its value, and the XPC will handle the linking operation normally. However, the user should not use the link bit value for his own routines. If no CRC error exists, the XPC does not write the receive status and receive buffer FC bytes. Therefore, the user must initialize the CRC error bit to zero. After a frame is received, the XPC increments V(R) without regard to frame type. Zero deletion is performed throughout the reception process.

In transparent operation, the XPC continues to receive frames until the end of the receive specification table is reached. The XPC then sets the RTE (receive table ended) bit in the Rx/host status register. The only errors reported in the Rx/host status register during transparent operation are buffer too short, no receive memory buffer available, undefined host command, illegal host command, address error, bus error, abort sequence, overrun, and receiver idle.

The XPC requests the bus when there are six bytes in the receive FIFO. Only a single frame can reside in the receive FIFO. Frames are received in sequence as long as memory buffers are available.

3.5 EXITING TRANSPARENT MODE

The transparent mode of operation is exited by issuing the end monitor command to the XPC. This will disable frame transmission and reception. There is no verification that the link is disconnected.

3

SECTION 4
INTERNAL REGISTERS

The XPC has four functional blocks: serial, DMA, microcode controller, and register-file/ALU. Each section contains user-visible and nonvisible registers that define and control the operation of the XPC.

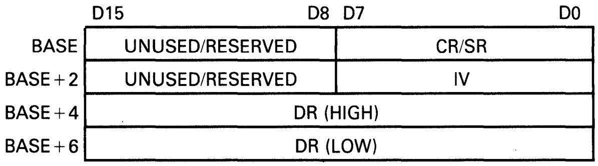
Because the XPC communicates with the host primarily through shared memory, a minimum number of host processor accessible registers are required. Registers in the XPC fall into two groups. One group is directly accessible by the user, and the other group is indirectly accessed through the station table. The directly accessible registers include the command register, semaphore register, interrupt vector register, and data register. The complete register set is shown in Table 4-1. A register map is shown in Figure 4-1.

Table 4-1. XPC Register Set (Sheet 1 of 2)

Register	Mnemonic	Mode	Read By	Written By
Directly Accessible				
Command	CR	Write	Not Applicable	Host Processor
Semaphore	SR	Read	Host Processor	Not Applicable
Interrupt Vector	IV	Read/Write	Host Processor (IACK)	Host Processor
Data	DR	Write	Not Applicable	Host Processor
Indirectly Accessible				
Station Table Pointer	STP	Write	Not Applicable	Load Station Table Pointer
Station Table Function Code	STFC	Write	Not Applicable	Load Function Code
Transmit Table Pointer	TTP	Read/Write	Dump Registers Dump Parameters	Load Transmit Table Pointer
Transmit Table Function Code	TTFC	Read/Write	Dump Registers Dump Parameters	Load Transmit Table Pointer
Transmit Buffer Pointer	TBP	Read/Write	Dump Registers	Load Transmit Table Pointer
Transmit Buffer Function Code	TBFC	Read/Write	Dump Registers	Load Transmit Table Pointer
Transmit Buffer Count	TBC	Read/Write	Dump Registers	Load Transmit Table Pointer
Receive Table Pointer	RTP	Read/Write	Dump Registers Dump Parameters	Load Receive Table Pointer
Receive Table Function Code	RTFC	Read/Write	Dump Registers Dump Parameters	Load Receive Table Pointer
Receive Buffer Pointer	RBP	Read/Write	Dump Registers	Load Receive Table Pointer
Receive Buffer Function Code	RBFC	Read/Write	Dump Registers	Load Receive Table Pointer
Receive Buffer Count	RBC	Read/Write	Dump Registers	Load Receive Table Pointer

Table 4-1. XPC Register Set (Sheet 2 of 2)

Register	Mnemonic	Mode	Read By	Written By
Directly Accessible				
Local Address	LA	Read/Write	Dump Registers	Load Addresses Load Station Parameters
Remote Address	RA	Read/Write	Dump Registers	Load Addresses Load Station Parameters
Hardware Configuration	HC	Write	Not Applicable	Set Hardware Configuration
Station Configuration	SC	Write	Not Applicable	Set Station Configuration
Option Bits	OB	Write	Not Applicable	Load Station Parameters Load Option Bits
Mode Descriptor	MD	Read	Dump Parameters Dump Registers	Not Applicable
Frame Reject Descriptor	FRD	Read	Dump Parameters Dump Registers	Not Applicable
Rx/Host Status	RHS	Read/Write	Dump Parameters Dump Registers Update Status	Clear Rx/Host Status Clear Status
Tx/Link Status	TLS	Read/Write	Dump Parameters Dump Registers Update Status	Clear Tx/Link Status Clear Status
V(S)	V(S)	Read	Dump Parameters Dump Registers	Not Applicable
V(R)	V(R)	Read	Dump Parameters Dump Registers	Not Applicable
Time-Out Preset	TOP	Read/Write	Dump Registers	Load Preset Values Load Station Parameters
Retries Limit	RL	Read/Write	Dump Registers	Load Preset Values Load Station Parameters
Outstanding Frames Limit	OFL	Read/Write	Dump Registers	Load Preset Values Load Station Parameters
Time Scale Divider	TSD	Write	Not Applicable	Load Preset Values Load Station Parameters
Pad Time Select	PTS	Write	Not Applicable	Load Preset Values Load Station Parameters
Retries Count	RC	Read	Dump Registers	Not Applicable
Last Received N(R)	LRN	Read	Dump Registers	Not Applicable



	D15 — D8	D7 — D0
BASE	UNUSED/RESERVED	CR/SR
BASE+2	UNUSED/RESERVED	IV
BASE+4	DR (HIGH)	
BASE+6	DR (LOW)	

Figure 4-1. XPC Register Map

4.1 COMMAND REGISTER

The control interface between the XPC and the host processor is the command register (CR). This 8-bit register is written by the host processor to issue commands to the XPC. The 24 valid XPC commands are shown in Table 4-2. Each command is described in detail in **SECTION 6 COMMAND SET**.

Before the host processor writes to the XPC, the semaphore register must be checked to ensure that the XPC is ready to accept the next command. Passing the XPC an undefined command will set the undefined instruction bit (UDEF) in the Rx/host status register.

Table 4-2. Valid Commands

Instruction	Bit								Hex Value
	IR7	IR6	IR5	IR4	IR3	IR2	IR1	IR0	
Load FC	1	0	0	0	0	0	0	0	80
Load STP	1	0	0	0	0	0	0	1	81
Load Option Bits	1	0	0	0	0	0	1	0	82
Load Preset Values	1	0	0	0	0	0	1	1	83
Load Addresses	1	0	0	0	0	1	0	0	84
Load Tx Table Pointer	1	0	0	0	0	1	0	1	85
Load Rx Table Pointer	1	0	0	0	0	1	1	0	86
Load Station Table Parameters	1	0	0	0	0	1	1	1	87
Update Status	1	0	0	0	1	0	0	0	88
Clear Tx/Link Status	1	0	0	0	1	0	0	1	89
Clear Rx/Host Status	1	0	0	0	1	0	1	0	8A
Clear Status	1	0	0	0	1	0	1	1	8B
Dump Parameters	1	0	0	0	1	1	0	0	8C
Start Link	1	0	0	0	1	1	0	1	8D
Stop Link	1	0	0	0	1	1	1	0	8E
Dump Registers	1	0	0	0	1	1	1	1	8F
DMA Transfer	1	0	0	1	0	0	0	0	90
Serial Loopback	1	0	0	1	0	0	0	1	91
Monitor	1	0	0	1	0	0	1	0	92
End Monitor	1	0	0	1	0	0	1	1	93
Continue Tx	1	0	0	1	0	1	0	1	95
Reset	1	1	1	1	1	1	1	1	FF
Set Station Configuation	1	0	1	X	M	X	M	X	
Set Hardware Configuration	1	1	0	M	M	X	M	M	

X = Don't Care
M = Modifier bits in the configuration commands:
Valid commands for set station configuration are hex A0-BF.
Valid commands for set hardware configuration are hex C0-DF.
All other command encodings are undefined/reserved.

4.2 SEMAPHORE REGISTER

The 8-bit semaphore register (SR) is the mechanism by which the XPC indicates that it has completed the current processor command. The semaphore register is written by the XPC and read by the host processor. When a command is written to the XPC command register, the XPC indicates that it has accepted the command by setting the semaphore register to 'FE'. After the completion of the command, the XPC sets this register to 'FF'. The host processor must read the semaphore register to ensure that it is 'FF' before writing a new command to the XPC. The semaphore register also allows the XPC and the host processor to share external memory without conflict.

4.3 INTERRUPT VECTOR REGISTER

The interrupt vector register (IV) contains the 8-bit interrupt vector number which is presented to the system during an interrupt acknowledge cycle. The upper six bits of the vector are user programmable. The least significant two bits are generated internally by the XPC to provide a unique interrupt vector number corresponding to the specific interrupt source. These encodings are shown in the following table.

IV1	IV0	Interrupt Source
0	0	Transmitter and Receiver
0	1	Receiver
1	0	Transmitter
1	1	BERR or AERR During a Station Table Access or Default Vector Number ('0F' Hex)

During reset the interrupt vector register is initialized to '0F' Hex by the XPC. If the interrupt vector register is not written by the host processor to specify a different interrupt vector prior to an interrupting condition, the XPC will pass a '0F' regardless of the interrupt source.

Three sources of internal interrupts are recognized by the XPC. These sources are the transmitter, the receiver, or both the transmitter and receiver. A fourth interrupt condition indicates that the XPC had a bus error (BERR) or address error (AERR) while accessing the station table.

Except when a BERR or AERR has occurred during a station table access, the XPC interrupt indicates to the host processor that the appropriate status word in the station table contains additional information about the specific interrupt condition. As part of the interrupt service routine to handle the XPC interrupt request, the host should set the appropriate bit(s) in the Rx/host and/or Tx/

link status clear locations in the station table. The host should then issue a clear Rx/host, clear Tx/link, or clear status command to cause the XPC to clear the corresponding internal status register bit(s) and the station table status bit(s).

A bus or address error during a station table access is a fatal condition, and the XPC will continue to generate an interrupt until a hardware or software reset is received.

4.4 DATA REGISTER

The 32-bit data register (DR) is accessed during initialization to load the station table pointer register and the station table function code register using a load STP or load FC command, respectively.

4.5 INDIRECTLY ACCESSIBLE REGISTERS

The remaining XPC registers are visible to the user but are not directly accessible. Commands issued to the XPC cause the XPC to load these registers from the station table or write these registers to the station table. Internal registers can also be written to a dump area in memory using the dump registers command.

4.5.1 Station Table Pointer Register

The 32-bit station table address is stored in the station table pointer register (STP). The STP is initialized as the result of a load STP command during initialization. A hardware or software reset clears this register.

4.5.2 Station Table Function Code Register

The station table function code register (STFC) holds the 4-bit function code value which may be required by the system to access the station table. This register is initialized after reset by the load FC command. A hardware or software reset clears this register.

4.5.3 Transmit Table Pointer Register

The 32-bit transmit frame specification table address is stored in the transmit table pointer register (TTP) (refer to Figure 4-2). Execution of the load TTP command loads the transmit table pointer, words 14 and 15 of the station table, into the transmit table pointer register. The TTP register is incremented as frames are transmitted to point to the current transmit frame specification block. The transmit table pointer location in the station table is not updated. After the entire information chain has been transmitted, the TTP register points to the first memory location following the end of the transmit frame specification table.

To allow the host to determine which frame the XPC is currently transmitting and which frames have been acknowledged, the host can issue a dump parameters command. The transmit pointer and the first unacknowledged pointer are written to the station table. The transmit pointer contains the current value of the transmit table pointer register, and the first unacknowledged pointer is based on the current value of the transmit table pointer register and the value of the last received N(R). The transmit table pointer register is written to the dump area by the dump registers command. The TTP is cleared by a hardware or software reset.

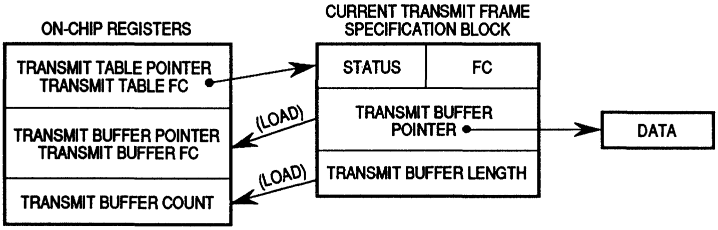


Figure 4-2. Transmit Registers

4.5.4 Transmit Table Function Code Register

The transmit table function code register (TTFC) contains the 4-bit function code value that may be required by the system to access the transmit table pointed to by the transmit table pointer register. The TTFC is loaded with the transmit table function code, word 13 in the station table, using the load TTP command. The TTFC register is written to the dump area by executing the dump registers command. A hardware or software reset clears this register.

4.5.5 Transmit Buffer Pointer Register

The transmit buffer pointer register (TBP) contains the 32-bit address of the current transmit buffer. The transmit buffer pointer register is autoincremented as the data in the buffer is transmitted until the count in the buffer length location of the current transmit frame specification block is reached. The TBP register is first loaded from the buffer address location of the transmit frame specification block using the load TTP command. The TBP register is then loaded automatically by the XPC to access each frame buffer until the end of the transmit frame specification table is reached. This register is written to the dump area by executing the dump registers command. The TBP is not cleared by a hardware or software reset.

4.5.6 Transmit Buffer Function Code Register

The transmit buffer function code register (TBFC) contains the 4-bit function code value that may be required to access the transmit buffer pointed to by the transmit buffer pointer register. The TBFC is loaded from the buffer function code location of the transmit frame specification block pointed to by the transmit table pointer register. This register is first loaded by the load TTP command. The TBFC is then loaded automatically by the XPC to access each frame buffer until the end of the transmit frame specification table is reached. The TBFC is written to the dump area by the dump registers command. The TBFC is not cleared by a hardware or software reset.

4.5.7 Transmit Buffer Count Register

The transmit buffer count register (TBC) is a 16-bit register that contains the number of bytes to be transmitted from the transmit buffer pointed to by the transmit buffer pointer register. As the information is transmitted, the XPC autodecrements the TBC until the count is zero. The TBC is loaded from the buffer length location in the transmit frame specification block pointed to by the transmit table pointer register. The TBC is first loaded by executing the load TTP command. This register is then loaded automatically by the XPC for each frame buffer until the end of the transmit frame specification table is reached. The dump registers command causes the TBC to be written to the dump area. The TBC is not cleared by a hardware or software reset.

4.5.8 Receive Table Pointer Register

The receive table pointer register (RTP) contains the 32-bit address of the receive frame specification table (refer to Figure 4-3). The value in the receive table pointer location, words 11 and 12 in the station table, is loaded into the RTP by executing the load RTP command. The receive table pointer register is incremented as frames are received to point to the current receive frame specification block. The receive table pointer location in the station table is not updated as frames are received. However, the host can determine the next receive pointer by issuing a dump parameters command. The XPC writes the current value of the receive table pointer register to the next receive table pointer location in the station table to allow the host to determine which receive buffers contain information. After the end of the receive frame specification table has been reached, the RTP points to the first memory location following the end of the receive frame specification table.

The receive pointer is written to the station table by a dump parameters command. The dump registers command causes the RTP to be written to the dump area. A hardware or software reset clears the RTP register.

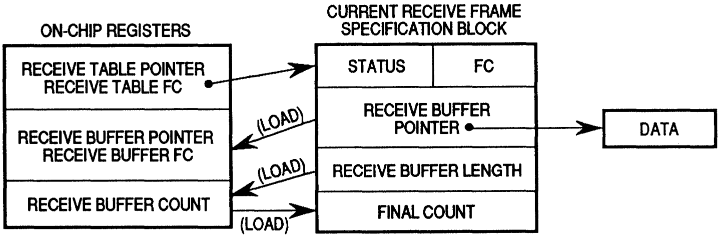


Figure 4-3. Receive Registers

4.5.9 Receive Table Function Code Register

The receive table function code register (RTFC) contains the 4-bit function code that may be required by the system to access the receive table pointed to by the receive table pointer register. This register is loaded by execution of the load RTP command from the receive table function code entry in word 10 of the station table. This register is written to the dump area by the dump registers command. A hardware or software reset clears the receive table function code register.

4.5.10 Receive Buffer Pointer Register

The current receive buffer address is stored in the 32-bit receive buffer pointer register (RBP). As the data is received, the RBP is autoincremented until the count in the buffer length location of the current receive frame specification block is reached or until the frame is completed. The RBP is first loaded from the buffer address location of the receive frame specification block by the execution of the load RTP command. This register is then loaded automatically by the XPC to access each receive frame buffer until the end of the receive frame specification table is reached. The RBP is written to the dump area by execution of the dump registers command. A hardware or software reset does not clear this register.

4.5.11 Receive Buffer Function Code Register

The receive buffer function code register (RBFC) contains the 4-bit function code value that may be required to access the receive buffer pointed to by the receive buffer pointer register. This register is first loaded from the receive buffer function code location in the current receive frame specification block by the execution of the load RTP command. This register is then loaded automatically by the XPC to access each receive frame buffer until the end of the receive frame specification table is reached. This register is written to the dump area by the dump registers command. A hardware or software reset does not clear the RBFC register.

4.5.12 Receive Buffer Count Register

The receive buffer count register (RBC) is a 16-bit register that contains the number of bytes available in the current receive buffer that is pointed to by the RBP register. This register is first loaded from the receive buffer length location in the receive frame specification block by the load RTP command. Thereafter, the RBC is loaded automatically by the XPC as each receive frame buffer is accessed.

The XPC autodecrements the RBC as information is received into the corresponding receive buffer until the count is exhausted or until the entire frame is received. After the frame is received, this register contains the number of bytes remaining in the current receive buffer. This value is written to the final count location of the receive frame specification block.

In transparent operation, if the length of the incoming frame exceeds the buffer length available, the XPC sets the BTS (buffer too short) bit in the Rx/host status register.

In X.25 operation, if the length of the incoming frame exceeds the buffer length available, the XPC enters frame reject mode and reports its status to the host.

The RBC is written to the dump area by executing the dump registers command. A hardware or software reset does not clear the RBC.

4.5.13 Local Address Register

The local address register (LA) is an 8-bit register that stores the station address of the XPC. Execution of the load addresses or load station parameters command loads the LA from the local address location, the low-order byte of word 8 in the station table. This register is written to the dump area by executing the dump registers command. A software or hardware reset clears this register. This register is not used in transparent operation.

4.5.14 Remote Address Register

The remote address register (RA) is an 8-bit register that stores the address of the remote station that communicates with the XPC. Execution of the load addresses or load station parameters command loads the RA from the remote address location, the low-order byte of word 9 in the station table. This register is written to the dump area by executing the dump registers command. A hardware or software reset clears this register. This register is not used in transparent operation.

4.5.15 Hardware Configuration Register

The hardware configuration register (HC) is an 8-bit register that contains system operating parameters. The operating parameters define the data encoding and decoding scheme, DMA burst control, data organization in memory, and data bus width. Execution of the set hardware configuration command during initialization loads this register. The valid operation parameters are shown below. This register is cleared by a hardware or software reset.

7	6	5	4	3	2	1	0
0	0	0	NRZI	BRSC	0	DORGM	BUSW

NRZI -Nonreturn to Zero Invert
0 = NRZ Decoding and Encoding
1 = NRZI Decoding and Encoding

BRSC — Burst Control
0 = XPC DMA Burst Is Unlimited
1 = XPC DMA Is Limited to Eight Successive Memory Cycles

DORGM — Data Organization in Memory (16-Bit Bus Only)
0 = Data in Memory Buffers Organized with High-Order Byte in Lower Address (Motorola, IBM Convention)
1 = Data in Memory Buffers Organized with High-Order Byte in Higher Address (Digital, Intel Convention)

BUSW — Bus Width
0 = 8-Bit Data Bus
1 = 16-Bit Data Bus

4.5.16 Station Configuration Register

The station configuration register (SC) is an 8-bit register that contains the station operating parameters. Execution of the set station configuration command during initialization loads this register. The SC register format is shown below. A hardware or software reset clears this register.

7	6	5	4	3	2	1	0
1	0	1	0	ECRC	0	ECNT	0

ECRC — Extended CRC
0 = 16-Bit CRC (CRC CCITT)
1 = 32-Bit CRC

ECNT — Extended Control
0 = Basic Control Field Format (Modulo 8)
1 = Extended Control Field Format (Modulo 128)

4.5.17 Option Bits Register

The option bits register (OB) is a 16-bit register that contains operation options. Execution of the load option bits or load station parameters command loads the OB register from the option bits location in word 0 of the station table. The register format is shown below. This register is cleared by a hardware of software reset. This register is not used in transparent operation.

F	E	D	C	B	A	9	8
0	0	0	0	0	0	0	X.75

7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	CRCNOA

X.75 — X.75 Option
0 = X.25 Operation
1 = X.75 Operation

CRCNOA — CRC Bypass Option
0 = Nonoctet Aligned Frames or Frames with a CRC Error Are Not Accepted
1 = Nonoctet Aligned Frames or Frames with a CRC Error Are Accepted

All other bits are reserved for future use.

If the X.75 option is selected, then, in addition to the frame reject conditions for X.25, the XPC will enter frame reject mode on reception of the following:

1. A supervisory frame with the F bit set to one, except during a timer recovery condition or except as a reply to a command sent with the P bit set to one,
2. An unexpected UA or DM response, or
3. An information frame with an invalid N(S).

4.5.18 Mode Descriptor Register

The 8-bit mode descriptor register (MD) contains the current operation mode information. The MD register is updated when the XPC operation mode changes. If the new mode bit in the Tx/link status register is not set when an operation mode change occurs, the XPC sets this bit and writes the MD register to the mode descriptor location in the high-order byte of word 21 in the station table (except for the TBEN bit and WACK bit discussed later). If the new mode bit is set when a mode change occurs, the XPC updates the

MD register but does not write the register to the station table location. As part of a new mode status condition service routine, the host should read the MD station table location and then issue a dump parameters command to get the latest new mode information that is stored in the MD register. Execution of the dump registers command causes the MD to be written to the dump area. The dump parameters command causes this register to be written to the station table. A hardware or software reset clears the mode descriptor register.

The TBEN (transmit buffer enable) bit is set internally after the load transmit table pointer command has been executed by the XPC. In order to prevent unnecessary interruptions to the host, the XPC will not dump the MD register into the station table or set the new mode bit in the Tx/link status register. The completion of the load transmit table pointer command is an indication that TBEN has been set internally. Similarly, the MD register is not written to the station table when TBEN is cleared due to the acknowledgement of all transmitted information frames, since the IFAK status bit in the Tx/link status register will be set and an interrupt generated to the host.

The WACK bit is set after the Start Link command is executed by the XPC or after a SABM or SABME is sent by the XPC to initiate link reset. When the WACK bit is set and the new mode bit is not currently set internally, the XPC will write the MD to the station table and generate a new mode interrupt. When the WACK bit is cleared, the XPC will not generate a separate new mode interrupt. Instead, the clearing of the WACK bit is indicated by either the CONN bit set (generating a new mode interrupt) or by the RTRYL bit set in the Tx/Link status register.

7	6	5	4	3	2	1	0
0	TBEN	ECNT	CONN	WACK	FRMR	RCVR	RSBY

TBEN — Transmit Buffer Enable
X.25
0 = Transmission of I Frames is Disabled
1 = Station Is Connected and Transmission of I Frames Is Enabled
Transparent
0 = Transmission of Frames Is Disabled
1 = Transmission of Frames Is Enabled

ECNT — Extended Control
0 = Basic Control Field Format (Modulo 8)
1 = Extended Control Field Format (Modulo 128)

CONN — Connected
- X.25
  - 0 = Station Is Disconnected
  - 1 = Station Is Connected
- Transparent — Not Used

WACK — Waiting for Acknowledgement
- X.25
  - 0 = Station Is Not Waiting for an UA Response
  - 1 = Station Is Waiting for an UA Response
- Transparent — Not Used

FRMR — Frame Reject Mode
- X.25
  - 0 = Station Not in FRMR Mode
  - 1 = Station in FRMR Mode
- Transparent — Not Used

4

RCVR — Recovery Mode
- X.25
  - 0 = Station Not in Timer Recovery Mode
  - 1 = Station in Timer Recovery Mode
- Transparent — Not Used

RBSY — Remote Station Busy
- X.25
  - 0 = Remote Station Is Not in Busy Condition
  - 1 = Remote Station Is in Busy Condition
- Transparent — Not Used

Table 4-3 shows the relationship between the TBEN, CONN, and IFAK (information frames acknowledged bit in the Tx/link status register).

Table 4-3. Relationship Between TBEN, CONN, and IFAK

Event	TBEN	CONN	IFAK
Chip Reset (Hardware or Software)	0	0	0
Enter Connect State (S5) due to SABM/UA Exchange[1]	0	1	0
Load Transmit Table Pointer Command	1	1	0
All Information Frames have been Transmitted and Acknowledged	0	1	1
Receive Response with F=1 after Transmitting Frame with P=1 (Enter S5 from S7), All Information Frames Acknowledged	0	1	1
Receive Response with F=1 after Transmitted Frame with P=1 (Enter S5 from S7), Some Information Frames Unacknowledged	1	1	0
Receive DISC (Enter S1 from S5) Information Frame Transmission was Not Enabled	0	0	0
Receive DISC (Enter S1 from S5) Some Information Frames Unacknowledged[2]	1	0	0
Receive DISC (Enter S1 from S5) All Information Frames Acknowledged	0	0	1

NOTES:

1. The XPC clears TBEN upon entering S5 due to a SABM/UA exchange, this is done in order to let upper layer software distinguish between normal return to S5 (from S6 and above, i.e., clear busy condition) and a link reset situation (from S6 and above where SABM/UA are exchanged). The software may instruct the XPC to dump parameters, analyze the transmit and receive pointers and recover without any information loss.
2. TBEN = 1 at DISC does not mean that the XPC is going to transmit I frames. However, it may be used as an indication that not all information frames were transmitted and acknowledged before entering the DISC mode. The XPC does not update TBEN and IFAK upon entering DISC from the connect states (S5 and above).

4.5.19 Frame Reject Descriptor Register

The 8-bit frame reject descriptor register (FRD) contains information about the cause of the current frame reject condition when the FRMR bit in the mode descriptor register is set. The FRD register is updated when the frame reject mode is entered and cleared when the frame reject mode is exited. If the new mode bit in the Tx/link status register is not set when the FRD is updated or cleared, the XPC will also write this register to the corresponding station table location. If the new mode bit is set, the XPC only updates the internal FRD register. This register is written to the dump area by executing the dump registers command. The dump parameters command causes this register to be written to the station table. A hardware or software reset clears this register. This register is not used in transparent operation.

7	6	5	4	3	2	1	0
C/R	0	0	0	Z	Y	X	W

C/R — Command/Response
0 = The Cause of FRMR Mode was a Command Frame
1 = The Cause of FRMR Mode was a Response Frame

Z — Invalid N(R)
Set if received I or S frame contained an invalid N(R).

Y — Receive Memory Buffer Too Short
Set if received information field is longer than the receive memory buffer, while in normal operation. Not used during monitor test.

X — Illegal I Field
Set if received frame has nonzero length information field and the control field does not permit an I field. The W bit must also be set.

W — Invalid or Not Implemented Control Field
Set if a correct frame with invalid or unimplemented control field is received.

4.5.20 Rx/Host Status Register

The Rx/host status register (RHS) is a 16-bit register that contains XPC status information relating to the condition of the receiver and host interface. These status bits have the same meaning for both X.25 and transparent operation unless otherwise specified. If a status bit is already set internally when the corresponding event occurs, the XPC takes no action. If the internal status bit is clear, the XPC sets the bit and checks its internal interrupt pending flag, which is a copy of the $\overline{IRQ}$ signal. If no interrupt is pending, the RHS is written to word 19 in the station table. This event can then cause a interrupt to occur if the corresponding bit in the Rx/host interrupt mask location is set. If an interrupt request is already pending when the event occurs, the RHS is not written to the station table. When the current interrupt request is cleared, the internal RHS is immediately written to the station table location. Any set status bits can then cause an interrupt.

Individual bits in the Rx/host status register are cleared by executing the clear Rx/host status or a clear status command. A hardware or software reset clears the Rx/host status register. The dump registers command causes this register to be written to the dump area. The update status or dump parameters command causes this register to be written to the station table.

F	E	D	C	B	A	9	8
BTS	NBA	CRCERR	0	RREJ	ETST	UDEF	ILL

7	6	5	4	3	2	1	0
AERR	BERR	ABORT	OVRN	RXIDLE	RXI	RFRMR	RTE

BTS — Receive Buffer is Too Short

The current buffer is used for the next frame received.

X.25

Not used. Instead, the FRMR mode is entered and the Y bit in the FRD register is set.

Transparent

Set when the received frame is longer than the receive buffer

NBA — No Receive Memory Buffer Available

The current buffer will be used for the next frame received.

X.25

Set when an I frame is received and no memory buffer is available.

Transparent

Set when a frame is received and no memory buffer is available.

CRCERR — CRC Error or Nonoctet Aligned Frame Received

A CRC check can not be performed on a frame which is less than 16 bits long; thus, this bit will not be set for those frames.

X.25

Set when a frame with local or remote address is nonoctet aligned or received with a CRC error.

Transparent — Not Used

RREJ — Received REJ Frame

X.25

Set when a REJ frame is received.

Transparent — Not Used

ETST — End of Test Command

Set upon the successful completion of DMA transfer or serial loopback commands.

UDEF — Undefined Host Command

Set when an unimplemented command from one of the following ranges is received from the host:

'20' — '7F' hex
'94' hex
'96' — '9F' hex
'E0' — 'FE' hex

If the XPC receives an unimplemented command in the range '00' — '1F' hex, the XPC will set the semaphore register to 'FE' and ignore the command. The semaphore register will remain at 'FE' until the next command is issued to the XPC.

ILL — Illegal Host Command

Set when a command is received from the host which conflicts with the current state of the XPC.

AERR — Address Error

Set when an address presented to the bus by the XPC to access the receive frame specification table or a receive buffer produces an XPC chip select or an XPC interrupt acknowledge input signal.

BERR — Bus Error

Set when the BERR encoding is asserted on the BEC input pins during a XPC access to the receive frame specification table or a receive buffer.

ABORT — Abort Sequence

Set when a minimum of seven consecutive ones are received (abort) during frame reception or when the carrier detect input signal is negated for 1 Tx clock cycle during frame reception. The current buffer will be used for the next frame received.

OVRN — Overrun

Set when a receive FIFO overrun occurs during frame reception.

RXIDLE — Receiver Idle

Set when more than 15 consecutive ones are received between frames.

RXI — Received Information Frame

X.25

Set when an information frame is received.

Transparent

Set when a frame is received.

RFRMR — Received FRMR Frame

X.25

Set when a FRMR frame is received. The FRMR I field is written to the station table.

Transparent — Not Used.

RTE — Receive Frame Specification Table Ended

X.25

Set after filling the last buffer specified by the receive frame specification table — RNR frame or RR frame is sent to acknowledge the reception of the previous frame(s) depending on whether the link bit in the receive status location of the last receive specification block is set.

Transparent

Set after filling the last buffer specified by the receive frame specification table.

4

4.5.21 Tx/Link Status Register

The Tx/link status register (TLS) is a 16-bit register containing the status information relating to the transmitter and communication link. These status bits have the same meaning for both X.25 and transparent operation unless otherwise specified. If a status bit is already set internally when the corresponding event occurs, the XPC takes no action. If the internal status bit is clear, the XPC sets the bit and checks its internal interrupt pending flag, which is a copy of the $\overline{\text{IRQ}}$ signal. If no interrupt is pending, the TLS is written to word 20 in the station table. This event can then cause an interrupt to occur if the corresponding bit in the Tx/link interrupt mask location is set. If an interrupt request is already pending when the event occurs, the TLS is not written to the station table. When the current interrupt request is cleared, the internal TLS is immediately written to the station table location. Any set status bits can then cause an interrupt.

Individual bits in the Tx/link status register are cleared by executing the clear Tx/link status or a clear status command. A hardware or software reset clears the Tx/link status register. The dump registers command causes this register to be written to the dump area. The update status or dump parameters command causes this register to be written to the station table.

F	E	D	C	B	A	9	8
0	0	0	0	0	0	NEWMD	RTRYL

7	6	5	4	3	2	1	0
AERR	BERR	CTSL	URUN	0	0	0	IFAK

NEWMD — New Mode

Set when the XPC operation mode changes. The mode descriptor register and the FRMR descriptor register give information about the mode change.

RTRYL — Retry Limit

X.25

Set when the retransmission limit specified in the retries limit register is exceeded.

Transparent — Not Used

AERR — Address Error

Set when an address presented to the bus by the XPC to access the transmit frame specification table or a transmit buffer produces an XPC chip select or an XPC interrupt acknowledge input signal.

BERR — Bus Error

Set when the BERR encoding is asserted on the BEC input pins during an XPC access to the receive frame specification table or a receive buffer.

CTSL — Clear to Send Lost

Set when CTS is not asserted within 68 Tx clock cycles after RTS was asserted or when CTS is negated for 1 Tx clock cycle during frame transmission.

URUN — Underrun

Set when a transmit FIFO underrun occurs during frame transmission. The XPC transmits an abort sequence and then retransmits the frame.

IFAK — Information Frames Acknowledged

X.25

Set when the current transmit frame specification table has ended and all information frames have been acknowledged.

Transparent

Set when the last byte of the last frame is loaded into the Tx FIFO.

4.5.22 V(R) Register

During transparent operation, the 8-bit V(R) register counts the number of frames received regardless of frame type. After a frame is received, V(R) is incremented Modulo 8 or Modulo 128 depending on the ECNT bit in the station configuration register.

During X.25 operation, the 8-bit V(R) register contains the sequence number N(S) of the last received I frame incremented by one. Therefore, V(R) is equal to the sequence number of the next expected I frame.

This register is stored in the V(R) location, the low-order byte of word 22, in the station table as the result of a dump parameters command. Execution of the dump registers command causes the V(R) register to be written to the dump area. A hardware or software reset clears this register.

4.5.23 V(S) Register

During transparent operation, the 8-bit V(S) register counts the number of frames transmitted regardless of frame type. After a frame is transmitted, V(S) is incremented Modulo 8 or Modulo 128 depending on the ECNT bit in the station configuration register.

During X.25 operation, the 8-bit V(S) register contains the send sequence N(S) of the next information frame to be transmitted by the XPC.

This register is stored in the V(S) location, the high-order byte of word 22, in the station table as the result of a dump parameters command. Execution of the dump registers command causes the V(R) register to be written to the dump area. A hardware or software reset clears this register.

4.5.24 Time-Out Preset Register

The time-out preset register (TOP) is a 16-bit register that contains the user specified time-out counter preset. The time-out period range is from 0 to $2^{16}-1$ clock cycles. The clock frequency is determined by the time scale divider register. The TOP is loaded from the time-out preset location, word 1 in the station table, using the load preset values or load station parameters command. The dump registers command dumps the TOP to the dump area. A hardware or software reset clears this register. This register is not used in transparent operation.

4.5.25 Retries Limit Register

The retries limit register (RL) is an 8-bit register that contains the user-specified number of retransmissions permitted. The retries limit register contains the value of the LAPB N2 parameter (maximum number of attempts to complete transmission) minus 1. The retries limit is loaded from the retries limit station table location, the lower byte of word 3, by a load preset values or load station parameters command. The RL is written to the dump area by the dump registers command. A hardware or software reset clears this register. This register is not used in transparent operation.

7	6	5	4	3	2	1	0
RL7	RL6	RL5	RL4	RL3	RL2	RL1	RL0

RL7–RL0

Contain the Binary Number of Retries (0 to 255)

4.5.26 Time Scale Divider Register

The time scale divider register (TSD) is an 8-bit register which contains the user specified system clock divider for generating the lower frequency clock used by the time-out counter. Valid divisors are 512, 1024, 2048, 4096, and 8192. This register is loaded from the time scale location, the high-order byte of word 2, in the station table by executing the load preset values or load station parameters command. This register is cleared by a hardware or software reset. This register is not used in transparent operation.

7	6	5	4	3	2	1	0
0	0	0	0	TS3	TS2	TS1	TS0

TS3–TS0

Determine the System Clock Divisor

TS3	**TS2**	**TS1**	**TS0**	**Divisor**
0	0	0	0	512
0	0	0	1	1,024
0	0	1	0	2,048
0	1	0	0	4,096
1	0	0	0	8,192

Encodings other than those specified are undefined.

4.5.27 Outstanding Frames Limit Register

The outstanding frames limit register (OFL) is an 8-bit register that contains the user-specified number of outstanding frames permitted. The number of outstanding frames may be zero to seven in the basic control field mode and zero to 127 in the extended control field mode of operation. This register is loaded from the outstanding frames limit location, the high-order byte of word 3, in the station table by executing the load preset values or load station parameters command. This register is cleared by a hardware or software reset. The OFL register is written to the dump area by executing the dump registers command. This register is not used in transparent operation.

7	6	5	4	3	2	1	0
0	OFL6	OFL5	OFL4	OFL3	OFL2	OFL1	OFL0

OLF6–OLF0

These bits are the binary number of outstanding frames allowed in extended control field operation.

OFL2–OFL0

These bits are the binary number of outstanding frames allowed in basic control field operation.

4.5.28 Pad Time Select Register

The pad time select register (PTS) is an 8-bit register that contains the user-specified minimum number of flags to be transmitted between frames. This register is loaded from the low-order byte of word 2 in the station table by executing the load preset values or load station parameters command. The PTS register is initialized to zero during a hardware or software reset.

4.5.29 Retries Count Register

The retries count register (RC) is an 8-bit register that contains the current number of retransmission attempts. This register is initialized to zero when a frame is transmitted. The RC register is incremented for each retransmission of the frame, until the value in the retries limit register (RL) is reached. No further transmission of the frame is attempted, and the XPC reports its status to the host. The RC register is written to the dump area by executing the dump registers command. The RC register is cleared when the XPC receives a UA or RNR frame or when the XPC receives information or supervisory frames with the N(R) higher than the last received N(R). The RC register is also cleared when the XPC enters a new mode (i.e., ABM, ABME, or ADM) and after transmitting a FRMR command or UA response. This register is also cleared on reset. This register is not used during transparent operation.

4.5.30 Last Received N(R)

The last received N(R) register (LRN) is an 8-bit register that contains the receive sequence number N(R) contained in the last received frame. This register is written to the dump area by executing the dump registers command. The LRN register is cleared by hardware or software reset and when the XPC (re)enters connect mode. This register is not used in transparent operation.

4.6 HARDWARE OR SOFTWARE RESET

Table 4-4 summarizes the effect of a hardware or software reset on the internal XPC registers.

Table 4-4. Reset Impact on Registers

Register	Cleared by Hardware or Software Reset
Station Table Pointer	Yes
Station Table Function Code	Yes
Transmit Table Pointer	Yes
Transmit Table Function Code	Yes
Transmit Buffer Pointer	No
Transmit Buffer Function Code	No
Transmit Buffer Count	No
Receive Table Pointer	Yes
Receive Table Function Code	Yes
Receive Buffer Pointer	No
Receive Buffer Function Code	No
Receive Buffer Count	No
Local Address	Yes
Remote Address	Yes
Hardware Configuration	Yes
Station Configuration	Yes
Option Bits	Yes
Mode Descriptor	Yes
Frame Reject Descriptor	Yes
Rx/Host Status	Yes
Tx/Link Status	Yes
V(S)	Yes
V(R)	Yes
Time-Out Preset	Yes
Retries Limit	Yes
Outstanding Frames Limit	Yes
Time Scale Divider	Yes
Pad Time Select	Yes
Retries Count	Yes
Last Received N (R)	Yes

SECTION 5
SHARED MEMORY STRUCTURES

The host processor communicates with the XPC using three tables located in shared memory (see Figure 5-1). The station table allows the host processor to initialize and update the XPC operating parameters and table pointers and to receive status and error information. The transmit frame specification table queues frames to be transmitted by the XPC, and the receive frame specification table queues available receive buffers for the XPC to store received information frames. The XPC is given a pointer to the station table during initialization. The transmit frame specification table and receive frame specification table pointers are contained in the station table.

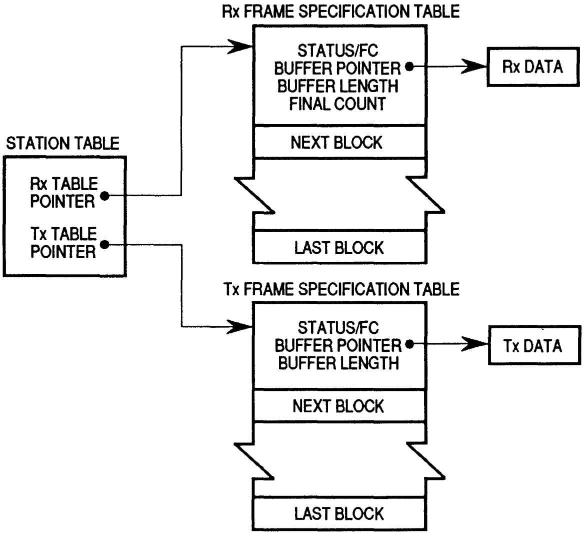


Figure 5-1. Shared Memory Tables

5.1 STATION TABLE

The station table format is shown in Table 5-1. The first 19 words of the station table are written by the host processor and are read by the XPC. This portion of the table contains the XPC operating information. The XPC accesses this table area as the result of a host processor command. The next 22 words of the table are written by the XPC and read by the host processor. Some of these entries are written by the XPC as the result of a command; other entries are updated by the XPC when a change occurs. When the XPC accesses the table as the result of a host processor command, it sets the semaphore register to hex 'FF' upon completion of the access. While the XPC is processing a command, the semaphore register is hex 'FE'.

5.1.1 Option Bits

Word zero of the station table contains the option bits. This word is loaded into the option bits register by executing a load option bits or load station parameters command. For more details see **4.5.17 Option Bits Register**.

F	E	D	C	B	A	9	8
0	0	0	0	0	0	0	X.75

7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	CRCNOA

5.1.2 Time-Out Preset

Word 1 of the station table contains the user-specified time-out counter preset. This value is loaded into the time-out preset register as a result of a load preset values or load station parameters command. The value can range from 0 to $2^{16}-1$ clock cycles. For more details see **4.5.24 Time-Out Preset Register**. The location is not used in transparent operation.

Table 5-1. Station Table Structure

WORD	15 12	11 8	7 4	3 0	
0	OPTION BITS				
1	TIME-OUT PRESET				
2	TIME SCALE DIVIDER		PAD TIME SELECT		
3	OUTSTANDING FRAMES LIMIT		RETRIES LIMIT		
4	Rx/HOST MASK BITS				
5	Tx/LINK MASK BITS				
6	Rx/HOST STATUS CLEAR BITS				
7	Tx/LINK STATUS CLEAR BITS				
8	0 0 0 0	0 0 0 0	LOCAL ADDRESS		
9	0 0 0 0	0 0 0 0	REMOTE ADDRESS		HOST PROCESSOR AREA READ BY THE XPC WRITTEN BY HOST PROCESSOR
10	0 0 0 0	0 0 0 0	0 0 0 0	RTFC	
11	RECEIVE TABLE POINTER — HIGH WORD				
12	RECEIVE TABLE POINTER — LOW WORD				
13	0 0 0 0	0 0 0 0	0 0 0 0	TTFC	
14	TRANSMIT TABLE POINTER — HIGH WORD				
15	TRANSMIT TABLE POINTER — LOW WORD				
16	0 0 0 0	0 0 0 0	0 0 0 0	DAFC	
17	DUMP AREA POINTER — HIGH WORD				
18	DUMP AREA POINTER — LOW WORD				

WORD	15 12	11 8	7 4	3 0	
19	Rx/HOST STATUS				
20	Tx/LINK STATUS				
21	MODE DESCRIPTOR		FRAME REJECT DESCRIPTOR		
22	V(S)		V(R)		
23	0 0 0 0	0 0 0 0	0 0 0 0	FUFC	
24	FIRST UNACKNOWLEDGED POINTER — HIGH WORD				
25	FIRST UNACKNOWLEDGED POINTER — LOW WORD				
26	0 0 0 0	0 0 0 0	0 0 0 0	TFC	
27	TRANSMIT POINTER — HIGH WORD				
28	TRANSMIT POINTER — LOW WORD				
29	0 0 0 0	0 0 0 0	0 0 0 0	RTFC	XPC AREA READ BY THE HOST PROCESSOR WRITTEN BY THE XPC
30	RECEIVE POINTER — HIGH WORD				
31	RECEIVE POINTER — LOW WORD				
32	0 0 0 0	0 0 0 0	0 0 0 0	REFC	
33	RECEIVE BUS/ADDRESS ERROR POINTER — HIGH WORD				
34	RECEIVE BUS/ADDRESS ERROR POINTER — LOW WORD				
35	0 0 0 0	0 0 0 0	0 0 0 0	TEFC	
36	TRANSMIT BUS/ADDRESS ERROR POINTER — HIGH WORD				
37	TRANSMIT BUS/ADDRESS ERROR POINTER — LOW WORD				
38	RECEIVED FRMR INFORMATION FIELD — WORD 1				
39	RECEIVED FRMR INFORMATION FIELD — WORD 2				
40	RECEIVED FRMR INFORMATION FIELD — WORD 3				

5.1.3 Time Scale Divider

The high-order byte of station table word 2 contains the time scale divider. This is a user-defined parameter for generating the lower frequency clock used by the time-out counter. The valid divider values are 512, 1024, 2048, 4096, and 8192. The time scale divider is loaded into the time scale divider register as the result of a load preset values or load station parameters command. For more details see **4.5.26 Time Scale Divider Register**. This location is not used in transparent operation.

7	6	5	4	3	2	1	0
*	*	*	*	TS3	TS2	TS1	TS0

*Not Used

5.1.4 Pad Time Select

The lower order byte of station table word 2 contains the user-specified minimum number of pad flags to be transmitted between frames. This parameter is loaded into the pad time select register by executing the load preset values or load station parameters command. This value can range from 0 to 255. When the pad time select value is zero, the XPC transmits a closing flag immediately followed by the opening flag of the next frame. Then, if six bytes are present in the transmit FIFO, the XPC will begin to transmit the frame address, etc. If the required number of bytes are not present in the FIFO, the XPC will transmit additional flags. This requirement for a minimum number of bytes to be available in the FIFO decreases the occurrence of a transmit underrun condition. Of course, transmission can begin when less than six bytes are present if the entire frame is less than six bytes.

5.1.5 Outstanding Frames Limit

The upper byte of station table word 3 contains the outstanding frames limit. This user-defined parameter is the number of frames which can be sent before an acknowledgement is required. The format of this table entry is shown below. The outstanding frames limit is loaded into the outstanding frames limit register by a load preset values or load station parameters command. The maximum number of outstanding frames is 127 for extended control mode and 7 for basic control mode. For more details see **4.5.27 Outstanding Frames Limit Register**. This location is not used in transparent operation.

7	6	5	4	3	2	1	0
*	OFL6	OFL5	OFL4	OFL3	OFL2	OFL1	OFL0

*Not Used

5.1.6 Retries Limit

The lower byte of station table word 3 holds the retries limit. This 8-bit user-specified parameter is the permitted number of attempts to successfully retransmit a frame. The maximum number of retransmission attempts is 255. The retries limit is loaded into the retries limit register by the load preset values or load station parameters command. This location is not used in transparent operation.

7	6	5	4	3	2	1	0
RL7	RL6	RL5	RL4	RL3	RL2	RL1	RL0

5.1.7 Rx/Host Mask

Word 4 of the station table contains the Rx/host mask. This mask allows the user to selectively disable interrupts caused by receiver or host status conditions. Interrupt conditions are individually masked by clearing the appropriate bit in the mask entry. Even though an interrupting condition is masked, the corresponding status bit in the Rx/host status register and the Rx/host status entry in the station table will be updated. If a masked interrupt condition is later enabled and the corresponding bit in the Rx/host status register is set, an interrupt request will be issued. There is no internal copy of the Rx/host mask. Each time the Rx/host status changes, the XPC reads the current mask residing in the station table to determine whether to issue an interrupt. The format is identical to the Rx/host status entry in the station table and to the Rx/host status register. For more details see **4.5.20 Rx/Host Status Register**.

F	E	D	C	B	A	9	8
BTS	NBA	CRCERR	*	RREJ	ETST	UDEF	ILL

7	6	5	4	3	2	1	0
AERR	BERR	ABORT	OVRN	IDLE	RXI	RFRMR	RTE

*Not Used

5.1.8 Tx/Link Mask

Word 5 of the station table contains the Tx/link mask. This mask allows the user to selectively disable interrupts caused by transmitter or link status conditions. Interrupt conditions are individually masked by clearing the appropriate bit in the mask. Even though an interrupting condition is masked, the corresponding status bit in the Tx/link status register and the Tx/link status entry in the station table will be updated. If a masked interrupt condition is later enabled and the corresponding bit in the Tx/link status register is set, an interrupt request will be issued. There is no internal copy of the Tx/link mask. Each time the Tx/link status changes, the XPC reads the current mask residing in the station table to determine whether to issue an interrupt. The format of the mask is identical to the Tx/link status entry in the station table and to the Tx/link status register. For more details see **4.5.21 Tx/Link Status Register**.

F	E	D	C	B	A	9	8
*	*	*	*	*	*	NEWMD	RTRYL

7	6	5	4	3	2	1	0
AERR	BERR	CTSL	URUN	*	*	*	IFAK

*Not Used

5.1.9 Rx/Host Status Clear

The Rx/host status clear bits in word 6 of the station table allow the host to selectively clear status bits relating to receiver or host interface conditions. These bits are individually cleared by setting the appropriate bit in the Rx/host status clear entry in the station table and then instructing the XPC to clear Rx/host status or to clear status. The XPC clears the corresponding bit in the Rx/host status register and writes the register to the Rx/host status entry in the station table. The format is identical to the Rx/host status register. For more details see **4.5.20 Rx/Host Status Register**.

F	E	D	C	B	A	9	8
BTS	NBA	CRCERR	*	RREJ	ETST	UDEF	ILL

7	6	5	4	3	2	1	0
AERR	BERR	ABORT	OVRN	IDLE	RXI	RFRMR	RTE

*Not Used

5.1.10 Tx/Link Status Clear

Tx/link status clear bits in word 7 of the station table allow the host to selectively clear status bits relating to transmitter or link conditions. These bits are individually cleared by setting the appropriate bit in the Tx/link status clear entry in the station table and then instructing the XPC to clear Tx/link status or to clear status. The XPC clears the corresponding bit in the Tx/link status register and writes the register to the Tx/link status entry in the station table. The format is identical to the Tx/link status register. For more details see **4.5.21 Tx/Link Status Register**.

F	E	D	C	B	A	9	8
*	*	*	*	*	*	NEWMD	RTRYL

7	6	5	4	3	2	1	0
AERR	BERR	CTSL	URUN	*	*	*	IFAK

*Not Used

5.1.11 Local Address

The lower order byte of station table word 8 contains the XPC's station address. The 8-bit address is loaded into the local address register by the load addresses or load station parameters command. This location is not used in transparent operation.

5.1.12 Remote Address

The lower order byte of station table word 9 contains the address of the remote station that communicates with the XPC. The 8-bit remote address is loaded into the remote address register by the load addresses or load station parameters command. This location is not used in transparent operation.

5.1.13 Receive Table Function Code

Bits 0–3 of station table word 10 contains the function code value that may be required by the system to access the receive frame specification table, which starts at the address contained in the receive table pointer location of the station table. The function code value is stored in the receive table function code register by executing a load RTP command. The function codes are defined by the user, and the XPC does not perform any checking on these values.

5.1.14 Receive Table Pointer

The 32-bit receive table pointer is stored in words 11 and 12 of the station table. This pointer is the address of the receive frame specification table which specifies free memory buffers for storage of received information frames. The XPC presents this address and the receive table function code when first accessing the receive frame specification table. This pointer is stored in the receive table pointer register by executing a load receive table pointer command. For more details see **5.3 RECEIVE FRAME SPECIFICATION TABLE**.

NOTE

The XPC does not check for an "odd pointer" value and does not generate an address error for the odd word boundary access. In an 8-bit data bus system, an "odd pointer" is proper. In a 16-bit data bus system, the XPC zeros the least significant address bit of the pointer and therefore presents an even word address to the bus. The system designer must ensure that the XPC is not given an odd pointer in a 16-bit data bus configuration.

5.1.15 Transmit Table Function Code

Bits 0–3 of station table word 13 contain the function code value that may be required by the system to access the transmit frame specification table, which starts at the address contained in the transmit table pointer location of the station table. The function code value is stored in the transmit table function code register by executing a load TTP command. The function codes are defined by the user, and the XPC does not perform any checking on these values.

5.1.16 Transmit Table Pointer

Station table words 14 and 15 contain the 32-bit transmit table pointer. This pointer is the address of the transmit frame specification table. The XPC presents this pointer and the transmit table function code to the system when first accessing the transmit frame specification table. This pointer is stored in the transmit table pointer register by executing a load TTP command. For more details see **5.2 TRANSMIT FRAME SPECIFICATION TABLE**.

5.1.17 Dump Area Function Code

Bits 0–3 in station table word 16 contain the function code value that may be required by the system to access the dump area, which starts at the address contained in the dump area pointer entry in the station table. The dump area function code is loaded by the XPC during the execution of the dump registers command.

5.1.18 Dump Area Pointer

Words 17 and 18 in the station table contain the address of the dump area in memory. This 32-bit address is loaded into the XPC during the execution of the dump registers command. This command writes internal registers into this dump area in memory for diagnostic purposes.

5.1.19 Rx/Host Status

The Rx/host status entry in station table word 19 contains XPC status information which relates to the condition of the receiver and host interface. Rx/host status bits are individually cleared by the clear Rx/host status or clear status commands. For more details see **4.5.20 Rx/Host Status Register**.

F	E	D	C	B	A	9	8
BTS	NBA	CRCERR	*	RREJ	ETST	UDEF	ILL

7	6	5	4	3	2	1	0
AERR	BERR	ABORT	OVRN	IDLE	RXI	RFRMR	RTE

*Not Used

5.1.20 Tx/Link Status

Word 20 of the station table, Tx/link status, contains the XPC status information relating to the condition of the transmitter and communications link. Tx/link status bits are individually cleared by the clear Tx/link status or clear status commands. For more details see **4.5.21 Tx/Link Status Register**.

F	E	D	C	B	A	9	8
*	*	*	*	*	*	NEWMD	RTRYL

7	6	5	4	3	2	1	0
AERR	BERR	CTSL	URUN	*	*	*	IFAK

*Not Used

5.1.21 Mode Descriptor

The mode descriptor, in the high-order byte of station table word 21, contains information about the XPC operation mode. The mode descriptor register and the mode descriptor in the station table are updated to reflect the current XPC operation mode as changes occur. The mode descriptor format is shown below. For more details see **4.5.18 Mode Descriptor Register**.

7	6	5	4	3	2	1	0
*	TBEN	ECNT	CONN	WACK	FRMR	RCVR	RBSY

*Not Used

5.1.22 Frame Reject Descriptor

The frame reject descriptor, in the low-order byte of station table word 21, contains information about the cause of the current FRMR mode. The frame reject descriptor is a copy of the frame reject descriptor register. Both the register and the station table entry are updated whenever the XPC enters or exits the FRMR mode. The format of the frame reject descriptor is shown below. For more details refer to **4.5.19 Frame Reject Descriptor Register**. In transparent operation, zero is written to this location.

7	6	5	4	3	2	1	0
C/R	*	*	*	Z	Y	X	W

*Not Used

5.1.23 V(R)

The V(R) register is stored in the low-order byte of station table word 22 as the result of a dump parameters command. The maximum value of the V(R) register is seven in basic operation mode and 127 in extended operation mode.

5.1.24 V(S)

The V(S) register is stored in the high-order byte of station table word 22 as the result of a dump parameters command. The maximum value of the V(S) register is seven in basic operation mode and 127 in extended operation.

5.1.25 First Unacknowledged Function Code

Bits 0–3 of word 23 in the station table contain the function code value that may be required by the system to access the contents of the address stored in the first unacknowledged pointer location in the station table. A dump parameters command updates this station table entry. This function code is the same as the transmit table function code. This location is undefined in transparent operation.

5.1.26 First Unacknowledged Pointer

Words 24 and 25 in the station table contain the first unacknowledged pointer. This pointer is the 32-bit address of the first transmit frame specification block whose corresponding transmit buffer has not been acknowledged. All transmit frame specification blocks up to this block identify transmit buffers that have been transmitted and acknowledged. The host can use this information to manage the transmit queue and frame buffers. This entry is updated by the XPC when a dump parameters command is executed. If all transmitted frames have been acknowledged, then this station table entry points to the first memory location following the end of the transmit frame specification table. This location is undefined in transparent operation.

5.1.27 Transmit Function Code

Bits 0–3 of station table word 26 contain the function code value that may be required by the system to access the contents of the address stored in the transmit pointer location in the station table. Execution of the dump parameters command updates this station table entry. The transmit function code is the same as the transmit table function code.

5.1.28 Transmit Table Pointer

Words 27 and 28 in the station table contain the transmit table pointer. This pointer is the 32-bit address of the transmit frame specification block which points to the next transmit buffer to be transmitted or to the transmit buffer that is currently being transmitted. All transmit frame specification blocks up to this block identify transmit buffers that have been transmitted by the XPC. Execution of the dump parameters command updates the transmit pointer entry. When the XPC executes a dump parameters command after all frames have been transmitted, this entry is updated to point to the first memory location following the end of the transmit frame specification table.

5.1.29 Receive Function Code

Bits 0–3 of station table word 29 contain the function code value that may be required by the system to access the contents of the address stored in the receive pointer location in the station table. The dump parameters command updates this entry. The receive pointer function code is the same as the receive table function code.

5.1.30 Receive Table Pointer

Words 30 and 31 of the station table contain the receive table pointer. This pointer is the 32-bit address of the receive frame specification block, which points to the first empty receive buffer or to the receive buffer that is currently being filled. The host may use this information for receive queue and frame buffer management. The receive pointer is updated when a dump parameters command is executed. When the XPC executes a dump parameters command and the receive frame specification table has ended, this entry points to the first memory location following the end of the table.

5.1.31 Receive Bus/Address Error Function Code

Bits 0–3 of station table word 32 contain the receive bus/address error function code. This function code was presented to the system by the XPC when a bus or address error occurred while accessing the receive frame specification table or a receive memory buffer.

5.1.32 Receive Bus/Address Error Pointer

Words 33 and 34 of the station table contain the receive bus/address error pointer. This address was presented to the system bus by the XPC when a bus/address error occurred while accessing the receive frame specification table or a receive memory buffer. An address error is generated by an address presented to the bus by the XPC which produces a XPC $\overline{\text{CS}}$ or $\overline{\text{IACK}}$. A bus error is generated when the BERR encoding is asserted on the $\overline{\text{BEC}}$ input pins during a XPC bus master cycle. Following an error, the XPC reaccesses the bus to write the receive bus/address error pointer entry in the station table.

5.1.33 Transmit Bus/Address Error Function Code

Bits 0–3 of station table word 35 contain the transmit bus/address error function code. This function code was presented to the system by the XPC when a bus/address error occurred while accessing the transmit frame specification table or a transmit memory buffer.

5.1.34 Transmit Bus/Address Error Pointer

Words 36 and 37 of the station table contain the transmit bus/address error pointer. This 32-bit address was presented to the system by the XPC when a bus/address error occurred while accessing the transmit frame specification table or a transmit memory buffer. An address error is generated by an address presented to the bus by the XPC which produces a XPC $\overline{\text{CS}}$ or $\overline{\text{IACK}}$. A bus error is generated when the BERR encoding is asserted on the $\overline{\text{BEC}}$ pins during a XPC bus master cycle. When a transmit bus/address error occurs, the XPC reaccesses the system bus to write the transmit bus/address error pointer entry in the station table.

5.1.35 Received FRMR Information Field

Words 38, 39, and 40 of the station table contain the information field of the received frame reject. When operating in basic mode (modulo 8), the XPC dumps the 24-bit information field into words 38 and 39. The low byte of word 39 is the same as the high byte of word 39. When operating in extended mode (modulo 128), the 40-bit information field is dumped into words 38, 39, and 40. The low byte of word 40 is the same as the high byte of word 40. This location is not updated in transparent operation.

5.2 TRANSMIT FRAME SPECIFICATION TABLE

The transmit frame specification table queues transmit frames for the XPC. These frames are stored in memory buffers located throughout memory. The transmit frame specification table contains a sequential list of transmit frame specification blocks. The transmit frame specification blocks describe the location of transmit buffers and provide information about the transmit queue. The transmit table pointer location in the station table points to the first transmit frame specification block. Each transmit frame specification block has the format shown below.

WORD	15 — 8	7 — 0
0	TRANSMIT TABLE STATUS	TRANSMIT BUFFER FC
1	TRANSMIT BUFFER ADDRESS — HIGH WORD	
2	TRANSMIT BUFFER ADDRESS — LOW WORD	
3	TRANSMIT BUFFER LENGTH	

5.2.1 Transmit Table Status

The high-order byte in word zero of a transmit frame specification block contains the queue status. The format of the status byte is shown below.

F	E	D	C	B	A	9	8
EOT	*	*	*	*	*	*	*

*Not Used

EOT — End of Table
0=Another Transmit Frame Specification Block Follows
1=This Block Is the Last Block in the Table

5.2.2 Transmit Buffer Function Code

The low-order byte of word zero is the transmit buffer function code. This function code value may be required by the system to access the transmit buffer identified by the transmit buffer address. The transmit buffer function code register is loaded from this location by executing a load TTP command to begin a new transmit frame specification table. This register is then automatically loaded by the XPC from each block's transmit buffer function code entry to access each frame buffer until the end of the transmit frame specification table is reached.

7	6	5	4	3	2	1	0
0	0	0	0	FC3	FC2	FC1	FC0

5.2.3 Transmit Buffer Address

Words 1 and 2 of the transmit frame specification block contain the 32-bit address of the transmit buffer. The transmit buffer pointer register is loaded with this 32-bit address by executing the load TTP command to begin a new transmit frame specification table. This register is then loaded automatically by the XPC from each block's buffer address entry to access each transmit buffer until the end of the transmit frame specification table is reached.

Note that the XPC does not check for an "odd pointer" value and does not generate an address error for the odd word boundary access. In an 8-bit data bus system, an "odd pointer" is proper. In a 16-bit data bus system, the XPC zeros the least significant address bit of the pointer and therefore produces an even word address to the bus. The system designer must ensure that the XPC is not given an "odd pointer" in 16-bit data bus configuration.

5.2.4 Transmit Buffer Length

Word 3 of the transmit frame specification block contains the number of bytes to be transmitted from the transmit buffer pointed to by the transmit buffer address. The transmit buffer count register is loaded with this count by executing a load TTP command to begin a new transmit frame specification table. This register is then loaded automatically by the XPC from each block's transmit buffer length entry before transmitting each buffer until the end of the transmit frame specification table is reached.

5.2.5 Transmit Bus/Address Error

If a bus error or address error occurs during a transmit frame specification table access or transmit buffer access, the XPC aborts any frame being transmitted and disables the transmit frame specification table as if the end of the table had been reached. Next, the XPC writes the function code and the address that caused the error into the transmit bus/address error entry and updates the Tx/link status entry in the station table. Before I frame transmission may resume, the host processor must instruct the XPC to load TTP.

5.3 RECEIVE FRAME SPECIFICATION TABLE

The receive frame specification table queues receive buffers for the XPC. These buffers are stored throughout memory. The receive frame specification table contains a sequential list of receive frame specification blocks. The receive frame specification blocks describe the location of the receive buffers and provide information about the queue. The receive table pointer in the station table points to the first receive frame specification block. Each receive frame specification block has the format shown below.

WORD	15	8	7	6	0
0	RECEIVE STATUS		L	RECEIVE BUFFER FC	
1	RECEIVE BUFFER ADDRESS — HIGH WORD				
2	RECEIVE BUFFER ADDRESS — LOW WORD				
3	RECEIVE BUFFER LENGTH				
4	RECEIVE FINAL COUNT				

5.3.1 Receive Status

The high-order byte in word zero of the receive frame specification block contains status information. The format is shown below.

F	E	D	C	B	A	9	8
EOT	*	*	*	*	*	*	E

*Not Used

EOT — End of Table

0 = Another Receive Frame Specification Block Follows

1 = This Block Is the Last Block in the Table

E — CRC Error or Nonoctet Aligned Frame Received

Set during transparent operation when the XPC receives a frame which is nonoctet aligned or receives a frame which contains a CRC error. The entire frame, including the received CRC, is written in the corresponding receive buffer. Since a CRC check cannot be performed on a frame which is less than 16 bits long, this bit will not be set for frames shorter than 16 bits.

5.3.2 Receive Buffer Function Code

The low-order byte of word zero contains the receive buffer function code and the link bit. This function code may be required by the system to access the receive buffer identified by the receive buffer address. The link bit provides a means to link receive tables.

7	6	5	4	3	2	1	0
L	0	0	0	FC3	FC2	FC1	FC0

L — Link

0 = This receive frame specification table is not linked to another receive table.

1 = This receive frame specification table is linked to another receive table. When this bit is set in the last receive frame specification block in the receive table, a valid receive table pointer and FC are stored in the corresponding station table locations.

FC3–FC0

These bits are the function code value associated with the receive memory buffer.

5.3.3 Receive Buffer Address

Words 1 and 2 of the receive frame specification block contain the 32-bit receive buffer address. The receive buffer pointer register is loaded with this 32-bit address by executing the load RTP command to begin a new receive frame specification table. The RBP is then loaded automatically by the XPC from each block's receive buffer address entry as information frames are received until the end of the receive frame specification table is reached.

Note that the XPC does not check for an "odd pointer" value and does not generate an address error for the odd word boundary access. In an 8-bit data bus system, an "odd pointer" is proper. In a 16-bit data bus system, the XPC zeros the least significant address bit of the pointer and therefore presents an even word address to the bus. The system designer must ensure that the XPC is not given an "odd pointer" in a 16-bit data bus configuration.

5.3.4 Receive Buffer Length

Word 3 of the receive frame specification table block contains the number of bytes available in the receive buffer pointed to by the receive buffer address. The receive buffer count register is loaded with this count by a load RTP command to begin a new receive frame specification table. This register is then loaded automatically by the XPC from each block's receive buffer length entry as information frames are received until the end of the receive frame specification table is reached.

5.3.5 Final Count

Word 4 of the receive frame specification block contains the final count. This count is the number of unused bytes in the memory buffer pointed to by the receive buffer address after the XPC has stored a received frame in the buffer. The receive buffer count register is loaded with the number of available bytes in the memory buffer from the receive buffer length entry. As a frame is received, the RBC is decremented. When the entire frame is stored in the receive buffer, the count in the receive buffer count register is written in the final count location in the current receive frame specification block.

5.3.6 Receive Bus/Address Error

If a bus error or address error occurs during a receive frame specification table access or a receive buffer access, XPC disables the receive frame specification table and stops receiving information frames. Next, the XPC writes the function code and the address that caused the error into the receive bus/address error entry and updates the Rx/host status entry in the station table. The host processor must instruct the XPC to load RTP to resume reception of information frames.

SECTION 6
COMMAND SET

The host processor issues commands to the XPC to perform various functions by writing to the XPC command register. There are 24 commands that fall in the following four categories:

1. Initialization
2. Table Handling
3. Link Handling
4. Test/Diagnostics

When the XPC receives a command from the host processor, the XPC sets the semaphore register to 'FE' hex. The XPC sets the semaphore register to 'FF' after completing the command to indicate to the host processor that it is ready for the next command. The host processor must read the semaphore register before writing the next command to ensure that the last command has been completed by the XPC.

6.1 INITIALIZATION COMMANDS

Initialization commands configure the XPC for operation after a hardware or software reset. The four initialization commands specify various system attributes, communication protocol options, and the location of the station table in memory. These commands should only be issued as part of an initialization procedure following reset.

6.1.1 Reset

The reset command (hex 'FF') and hardware reset cause the following actions:

Reset the Receive Channel and Isolate RxD
Reset the Transmit Channel, Negate $\overline{\text{RTS}}$, and Transmit Ones
Immediately Relinquish the System Bus
Set the Interrupt Vector Register to 'OF' Hex
Disable Transmit and Receive Memory Buffers
Clear All Rx/Host and Tx/Link Status Bits
Clear All Hardware and Station Configuration Bits

Clear All Option Bits
Clear All Mode Descriptor and Frame Reject Descriptor Bits
Zero Station Table Pointer and Station Table FC Registers
Zero Transmit Table Pointer and Transmit Table FC Registers
Zero Receive Table Pointer and Receive Table FC Registers
Zero Remote Address and Local Address Registers
Zero V(R), V(S), and Last Received N(R) Registers
Zero Preset Values and Retries Count Register

The transmit clock is required to execute the reset command.

6.1.2 Set Station Configuration

The set station configuration command specifies protocol parameters. The command has the following format.

7	6	5	4	3	2	1	0
1	0	1	X	ECRC	X	ECNT	X

ECRC — Extended CRC
0 = 16-Bit CRC
CRC CCITT $(X^{16}+X^{12}+X^{5}+1)$
1 = 32-Bit CRC
$(X^{32}+X^{26}+X^{23}+X^{22}+X^{16}+X^{12}+X^{11}+X^{10}+X^{8}+X^{7}+X^{5}+X^{4}+X^{2}+X^{1}+1)$

ECNT — Extended Control
0 = Basic Control Field Format (Modulo 8)
1 = Extended Control Field Format (Modulo 128)

X — Don't Care
Valid commands are A0–BF.

6.1.3 Set Hardware Configuration

The set hardware configuration command defines the data decoding/encoding scheme, DMA burst control, data organization in memory, and data bus size. The format of the command is shown below.

7	6	5	4	3	2	1	0
1	1	0	NRZI	BRSC	X	DORGM	BUSW

NRZI — Nonreturned to Zero Invert
0 = NRZ Decoding/Encoding
1 = NRZI Decoding/Encoding

BRSC — Burst Control
0 = DMA Burst is Unlimited
1 = DMA Burst is Limited to Eight Successive Memory Cycles

DORGM — Data Organization in Memory for a 16-Bit Data Bus System
0 = Data in Memory is Organized with High-Order Byte in Lower Memory Address (Motorola and IBM Convention)
1 = Data in Memory is Organized with Low-Order Byte in Lower Memory Address (DEC and Intel Convention)
(This capability is available only for I frame buffers and not for parameters or tables.)

BUSW — Bus Width
0 = 8-Bit Data Bus
1 = 16-Bit Data Bus

X — Don't Care
Valid commands include C0–DF.

6.1.4 Load Function Code

The load FC command (hex '80') writes the function code value in the data register into the station table FC register. This command is issued after the host processor has written the function code to the data register.

6.1.5 Load Station Table Pointer

The load station table pointer command (hex '8l') writes the initial station table address from the data register into the station table pointer register. This command is issued after the host processor has written the station table pointer to the data register.

6.2 TABLE HANDLING COMMANDS

The 12 table handling commands cause the XPC to access the station table, transmit table, or receive table.

6.2.1 Load Option Bits

The load option bits command (hex '82') loads the option set from the station table into the option bits register. This command is issued as part of the initialization routine or whenever options need to be changed.

6.2.2 Load Preset Values

The load preset values command (hex '83') loads the time-out preset value, time scale divider, pad time select, outstanding frames limit, and retries limit from the station table into the respective XPC internal registers. This command should only be issued following reset or when the XPC is in asynchronous disconnect mode (ADM).

6.2.3 Load Addresses

The load addresses command (hex '84') loads the local and remote addresses from the station table into the internal XPC registers. After these registers are loaded, the XPC is ready to establish the link. The XPC monitors the receive line, asserts $\overline{\text{RTS}}$, and transmits continuous flags. This command should only be issued following reset or when the XPC is in asynchronous disconnect mode (ADM).

6.2.4 Load Transmit Table Pointer

The load transmit table pointer (hex '85') loads the transmit table pointer and the transmit table FC from the station table into the corresponding XPC registers and enables the transmission of a chain of information frames. If the load TTP command is received when the XPC has transmit buffers not yet transmitted, not yet acknowledged in X.25 information transfer, or not yet transmitted in transparent operation, the illegal (ILL) bit in the Rx/host status register is set.

The load TTP command is accepted by the XPC after:

1. The last transmit buffer specified by the current transmit frame specification is transmitted in transparent operation or is transmitted and acknowledged in X.25 operation,
2. An address or bus error occurs while accessing the transmit frame specification table or a transmit memory buffer,

3. Executing the monitor or end monitor command, or
4. The XPC (re)enters the X.25 information transfer mode (ABM/ABME).

When the host processor instructs the XPC to load TTP, the XPC loads the transmit table pointer and transmit table function code registers from the corresponding station table entries. The transmit table pointer register then has the address of the first transmit frame specification block. Before the transmission of each frame, the XPC accesses the current transmit frame specification block to load the transmit buffer function code, transmit buffer address, and transmit buffer length into the corresponding internal registers. The XPC presents the transmit buffer address and function code to the system to load the information contained in the transmit buffer.

During transparent operation, the XPC accesses the next transmit frame specification block and transmits the corresponding frame buffer until the end of the transmit frame specification table is reached. The XPC updates its internal V(S) register after the transmission of each frame. When all frames have been transmitted, the XPC sets the IFAK bit in the Tx/link status register.

During X.25 operation, the XPC accesses the next transmit frame specification block and transmits the corresponding frame buffer according to the X.25 Recommendation until either the outstanding frames limit or the end of the transmit frame specification table is reached. The XPC updates its internal V(S) register after the transmission of an information frame. The XPC monitors the N(R) of incoming frames for reception acknowledgement. When all information frames queued in the transmit frame specification table have been transmitted, the XPC continues to monitor the N(R) of incoming frames until all transmitted frames have been acknowledged. After all frames have been acknowledged, the XPC sets the IFAK bit in the Tx/link status register.

Note that frames can be added to the transmit queue dynamically as the XPC is transmitting. After adding entries to the end of the transmit queue, the host must set the EOT bit in the last added entry and clear the EOT bit at the previous end of table. If the XPC has already read the EOT bit, then the added entries will not be transmitted. In this case, the continue transmit command (described in **6.2.5 Continue Transmission**) should be issued to the XPC.

The following events cause the XPC to abort the transmission of a frame:

1. Transmit or severe BERR/AERR
2. Transmit FIFO underrun
3. $\overline{\text{CTS}}$ is negated for 1 Tx clock cycle.
4. During X.25 operation, if a start link or stop link command is received from the host.
5. During X.25 operation, if the XPC receives a frame which causes the XPC to enter FRMR mode. The XPC transmits an ABORT and then transmits the FRMR.
6. During X.25 operation, if the XPC receives a REJ, SABM, DISC, DM, UA, FRMR, or an unexpected final bit.

6.2.5 Continue Transmit

The continue transmit command (hex '95') is used to extend the transmit queue after adding entries to the transmit frame specification table. The user should set the EOT bit in the transmit status location of the last added entry and then clear the EOT bit at the previous end of table. Finally, the user should instruct the XPC to "Continue Transmit". This command is useful in the case where the XPC has already detected the previous EOT and will not read a new table entry. Instead, it is waiting for all transmitted frames to be acknowledged, and, during this period, it will not accept a new load transmit table pointer command.

The XPC action after receiving the continue transmit command is:

1. In the case where the XPC has not yet reached the previous EOT, no action is taken. The XPC will continue transmission normally with the extended transmit specification table.
2. In the case during X.25 operation where the XPC has transmitted the whole table and is now waiting for acknowledgement, the XPC will continue transmission with the extended table when the number of outstanding frames permits it.
3. In the case where the whole table has been transmitted and the IFAK status bit has been set, the XPC will continue transmission similar to number 2 above. Note that, in this case, the load transmit table pointer command could be given instead of the continue transmit command.

6.2.6 Load Receive Table Pointer

The load receive table pointer command (hex '86') loads the receive table pointer and the receive table FC from the station table into the corresponding XPC registers and enables the reception of frames into the Rx FIFO. If the load RTP command is received when the XPC has receive buffers available, the illegal (ILL) bit in the Rx/host status register is set.

This command is accepted by the XPC after:

1. Hardware or software reset,
2. The last free receive memory buffer specified by the current receive frame specification table is filled,
3. An address or bus error occurs while accessing the receive frame specification table or a receive memory buffer, or
4. Executing a monitor or end monitor command.

When the host processor instructs the XPC to load RTP, the XPC loads the receive table function code and receive table pointer registers from the corresponding station table entries. The receive table pointer register then contains the address of the first receive frame specification block. The XPC accesses the receive frame specification block to load the receive buffer function code, receive buffer address, and the receive buffer length into its internal registers. The XPC then presents the receive buffer address and function code to the system to store the received information field in the memory buffer. After reception of a frame, the XPC writes the number of unused bytes in the final count entry in the current receive frame specification block and updates its internal V(R) register. Next, the XPC sets the received RXI bit in the Rx/host status register. The XPC accesses the next receive frame specification block to store incoming frames until the end of the receive frame specification table is reached.

To decrease the possibility of a receiver not ready condition due to a lack of available receive buffers, a method is provided for linking receive frame specification tables. When the EOT (end of table) bit is set in a receive frame specification block, the XPC inspects the link bit value. If the link bit is set, then the XPC loads the receive table pointer and FC registers from the corresponding station table locations. The XPC then sets the RTE (receive table ended) bit in the Rx/host status register and issues an interrupt if enabled. The link operation can be used to implement a cyclical queue by using the original RTP and FC values in the station table. However, the user must read filled receive buffers expediently to ensure that the XPC does not overwrite the buffers with incoming frames.

During transparent operation, if the link bit is not set in the last receive frame specificaton block, the XPC sets the RTE bit in the Rx/host status register. If a frame is received before another receive table is assigned via a load RTP command, the XPC sets the NBA (no receive buffer available) bit in the Rx/host status register.

During X.25 operation, if the link bit is set in the last receive frame specification block, the XPC transmits an RR frame, and then the XPC is ready to receive additional I frames into memory buffers using the new receive frame specification table. If the link bit is not set in the last receive frame specification block, the XPC sets the RTE bit in the Rx/host status register. If an I frame is received before another receive table is assigned via a load RTP command, the XPC sets the NBA (no receive buffer available) bit in the Rx/host status register and transmits an RNR frame.

Note that the XPC reads the next receive frame specification block immediately after completing the reception of a frame. The XPC does not reread the block when a new frame is received. So there is a period when the RTE bit is not yet set, but the last block has been read. During this time a modification of the link bit by the user is not detected by the XPC. Following an RTE status indication, the user may wish to verify that the linking operation was performed by the XPC or the user can wait for an NBA status condition which indicates that the link operation was not successful.

6.2.7 Load Station Parameters

The load station parameters command (hex '87') loads the station parameters from the station table into the XPC registers. This command should only be issued following reset or when the XPC is in asynchronous disconnect mode (ADM). This command combines the load option bits, load preset values, and load addresses commands. The XPC will behave as if each of these commands was issued sequentially to the XPC. The XPC will release the bus and rearbitrate for the bus between each command execution.

6.2.8 Update Status

The update status command (hex '88') allows the host to request current XPC status information. When the XPC receives this command, the XPC will negate the $\overline{\text{IRQ}}$ line (if it was asserted) and write both the Rx/host and Tx/link status registers to the station table. If the Rx/host or Tx/link status register does not equal zero, the XPC reads the station table mask entries and asserts

$\overline{\text{IRQ}}$ if necessary. Note that the same function is achieved by a clear status command when both station table status clear entries contain all zeros.

6.2.9 Clear Tx/Link Status

The clear Tx/link status command (hex '89') clears the status bits in the Tx/link status register as specified by the Tx/link status clear bits in the station table. When the XPC receives this command, it negates the $\overline{\text{IRQ}}$ line (if it is asserted), reads the Tx/link status clear bits, clears the corresponding status register bits and writes both Tx/link and Rx/host status registers into the station table. If the Tx/link status register does not equal zero, the XPC reads the station table mask entries and asserts $\overline{\text{IRQ}}$ if necessary.

6.2.10 Clear Rx/Host Status

The clear Rx/host status command (hex '8A') clears the Rx/host status bits in the Rx/host status register as specified by the Rx/host status clear bits in the station table. When the XPC receives this command, it negates the $\overline{\text{IRQ}}$ line (if it is asserted), reads the Rx/host status clear bits, clears the corresponding status register bits, and writes both Tx/link and Rx/host status registers into the station table. If the Rx/host status does not equal zero, the XPC reads the station table mask entries and asserts $\overline{\text{IRQ}}$ if necessary.

6.2.11 Clear Status

The clear status command (hex '8B') clears both the Tx/link and Rx/host status bits in the respective XPC registers as specified by the Tx/link status clear bits and the Rx/host status clear bits in the station table. This command combines the clear Tx/link status and clear Rx/host status commands. The XPC will behave as if each of these commands was issued sequentially to the XPC. The XPC will release the bus and rearbitrate for the bus between each command execution.

6.2.12 Dump Parameters

The dump parameters command (hex '8C') writes the following XPC parameters into the corresponding station table locations in the order given: Rx/host status, Tx/link status, mode descriptor, frame reject descriptor, V(R), V(S), first unacknowledged FC and pointer, transmit FC and pointer, and receive FC and pointer.

6.3 LINK HANDLING COMMANDS

The two link handling commands cause the XPC to set the link to a new mode and to automatically handle communication on both channels according to the predefined configuration and option bits.

6.3.1 Start Link

The start link command (hex '8D') initiates the link setup procedure as described in **2.2 LINK SETUP PROCEDURE**. If this command is issued while the waiting acknowledgement (WACK) bit in the mode descriptor register is set, the illegal command (ILL) bit in the Rx/host status register is set.

6.3.2 Stop Link

The stop link command (hex '8E') initiates the link disconnect procedure as described in **2.7 LINK DISCONNECT PROCEDURE**. If this command is issued while the waiting acknowledgement (WACK) bit in the mode descriptor register is set, the illegal command (ILL) bit in the Rx/host status register is set.

6.4 TEST/DIAGNOSTIC COMMANDS

The five commands in the test/diagnostic category test the XPC circuit and run diagnostics on the link. Upon completion of the serial loopback, DMA transfer, or end monitor, the XPC requires a load addresses command. Until this command is received, the XPC is isolated from the serial link.

6.4.1 Dump Registers

The dump registers command (hex '8F') writes the XPC registers listed below to a user specified dump area in external memory. The XPC loads the dump area pointer and FC from the station table. This command should not be executed during X.25 operation while the XPC is in connect mode. When the dump registers command is executed, the T1 timer is reset.

<table>
<tr><th>WORD</th><th>15 12</th><th>11 8</th><th>7 4</th><th>3 0</th></tr>
<tr><td>0</td><td>0 0 0 0</td><td>0 0 0 0</td><td>0 0 0 0</td><td>TTFC</td></tr>
<tr><td>1</td><td colspan="4">TRANSMIT TABLE POINTER — HIGH WORD</td></tr>
<tr><td>2</td><td colspan="4">TRANSMIT TABLE POINTER — LOW WORD</td></tr>
<tr><td>3</td><td>0 0 0 0</td><td>0 0 0 0</td><td>0 0 0 0</td><td>RTFC</td></tr>
<tr><td>4</td><td colspan="4">RECEIVE TABLE POINTER — HIGH WORD</td></tr>
<tr><td>5</td><td colspan="4">RECEIVE TABLE POINTER — LOW WORD</td></tr>
<tr><td>6</td><td>0 0 0 0</td><td>0 0 0 0</td><td>0 0 0 0</td><td>TBFC</td></tr>
<tr><td>7</td><td colspan="4">TRANSMIT BUFFER POINTER — HIGH WORD</td></tr>
<tr><td>8</td><td colspan="4">TRANSMIT BUFFER POINTER — LOW WORD</td></tr>
<tr><td>9</td><td>0 0 0 0</td><td>0 0 0 0</td><td>0 0 0 0</td><td>RBFC</td></tr>
<tr><td>10</td><td colspan="4">RECEIVE BUFFER POINTER — HIGH WORD</td></tr>
<tr><td>11</td><td colspan="4">RECEIVE BUFFER POINTER — LOW WORD</td></tr>
<tr><td>12</td><td colspan="4">TRANSMIT BUFFER COUNT</td></tr>
<tr><td>13</td><td colspan="4">RECEIVE BUFFER COUNT</td></tr>
<tr><td>14</td><td>0 0 0 0</td><td>0 0 0 0</td><td colspan="2">LOCAL ADDRESS</td></tr>
<tr><td>15</td><td>0 0 0 0</td><td>0 0 0 0</td><td colspan="2">REMOTE ADDRESS</td></tr>
<tr><td>16</td><td colspan="2">MODE DESCRIPTOR</td><td colspan="2">FRAME REJECT DESCRIPTOR</td></tr>
<tr><td>17</td><td colspan="4">Rx/HOST STATUS</td></tr>
<tr><td>18</td><td colspan="4">Tx/LINK STATUS</td></tr>
<tr><td>19</td><td colspan="2">V(S)</td><td colspan="2">V(R)</td></tr>
<tr><td>20</td><td colspan="4">TIME-OUT PRESET</td></tr>
<tr><td>21</td><td colspan="2">OUTSTANDING FRAMES LIMIT</td><td colspan="2">RETRIES LIMIT</td></tr>
<tr><td>22</td><td colspan="2">LAST RECEIVED N(R)</td><td colspan="2">RETRIES COUNT</td></tr>
</table>

6.4.2 DMA Transfer

The DMA transfer command (hex '90') tests the handling of parallel data in the logical configuration shown in Figure 6-1. The XPC reads the data from a transmit memory buffer and writes it to a receive memory buffer. Only one memory buffer is read and written each time this command is executed. The XPC transfers data from the transmit buffer to the receive buffer via the data register without using the internal transmit or received FIFOs. The serial link is not affected by this operation.

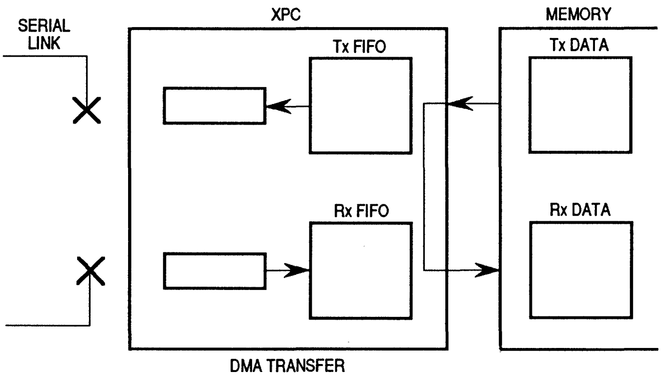


Figure 6-1. DMA Configuration

The XPC first loads the transmit table function code and pointer and the receive table function code and pointer from the station table. Next, the XPC loads the transmit buffer function code and pointer and the transmit buffer length from the transmit frame specification table. Finally, the XPC loads the receive buffer function code and pointer from the receive frame specification table and begins to transfer the data. The number of bytes to be transferred is specified by the transmit buffer length. The receive buffer final count is not updated.

When the DMA transfer test is completed, the XPC sets the ETST (end of test) bit in the Rx/host status register. The XPC may terminate the DMA transfer test prematurely due to an error condition (BERR, AERR, etc.). In this case, the appropriate status bit is set and the ETST bit is not set. The XPC only accepts this command while in disconnect mode. In connect mode, the XPC sets the ILL (illegal command) bit in the Rx/host status register.

6.4.3 Serial Loopback

The serial loopback command (hex '91') tests the handling of parallel and serial data in the logical configuration shown in Figure 6-2. The XPC reads data from the transmit memory buffer into the transmit FIFO. The data is then serialized and shifted internally into the receive FIFO and onto the TxD line. Finally, the data is stored in the receive memory buffer. Only one memory buffer is read and written each time this command is executed.

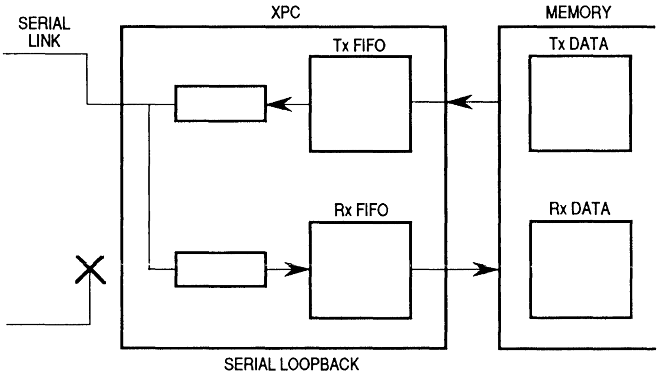


Figure 6-2. Serial Loopback Configuration

The XPC first loads the transmit table function code and pointer and the receive table function code and pointer from the station table. Next, the XPC loads the receive buffer function code and pointer from the receive frame specification table. Finally, the XPC loads the transmit buffer function code and pointer and transmit buffer length from the transmit frame specification table and begins to transfer the data. The number of bytes of data to be transmitted is specified by the transmit buffer length. The receive clock (RCLK) is used to synchronize both the receive and transmit channels. $\overline{\text{RTS}}$ is not active during serial loopback.

Following the transmit data, the XPC calculates and writes the CRC to the receive buffer. The CRC is 16 or 32 bits, depending on the CRC option selected. Then following the CRC, the XPC writes one word to indicate whether the XPC receiver determined a CRC error. If no CRC error occurred, all zeros are written in this word. If a CRC error has occurred, all ones are written in this word. Two or three extra words should be reserved in the receive memory buffer for CRC and CRC error indication. The receive buffer final count is not updated.

When the serial loopback test is completed, the XPC sets the ETST (end of test) bit in the Rx/host status register. The XPC may terminate the serial loopback test prematurely due to an error condition (BERR, AERR, etc.). In this case, the appropriate status bit is set and the ETST bit is not set. The XPC only accepts this command while in disconnect mode. In connect mode, the XPC sets the ILL (illegal command) bit in the Rx/host status register.

6.4.4 Monitor

The monitor command (hex '92') allows the XPC to check the communication channel by reading/writing the entire frame from/to memory. The monitor command may be used to perform an external loopback test of the system or to implement any HDLC/SDLC operation mode where all frames are user generated. The XPC transmits and/or receives multiple information frames until an end monitor command is received. The monitor command is only accepted by the XPC in disconnect mode. In connect mode, the XPC sets the ILL (illegal command) bit in the Rx/host status register.

After receiving the monitor command, the XPC monitors the receive line, asserts $\overline{\text{RTS}}$, and transmits continuous flags. The XPC then requires a load receive table pointer and/or load transmit table pointer command(s). Thereafter, the XPC transmits and/or receives frames using the transmit and/or receive frame specification table(s) pointed to by the TTP and/or RTP respectively.

In each transmit buffer, the user places the address, control, and data (if any) fields. For each frame buffer, the XPC transmits the user-specified number of pad flags, the opening flag, the user-specified address, control and data fields, the XPC-generated CRC, and the closing flag. Zero insertion is performed throughout.

On the receive side, the XPC strips off pad flags and the opening flag for each frame. The address, control, data (if any), and CRC fields are written into the current receive buffer. Zero deletion is performed throughout.

If a receive CRC error is detected by the XPC, the E (CRC error) bit of the receive status byte is set in the corresponding receive frame specification block. To set this bit, the XPC writes the receive status and receive buffer function code bytes. The L (link) bit is written as zero, regardless of its previous state. The EOT (end of table) bit and FC bits are unchanged. Even though the link bit is cleared during the write cycle, the XPC has previously read its value and will handle the linking operation normally. However, the user should not use the link bit value for its own routines.

If no CRC error exists, the XPC does not write the receive status and receive buffer FC bytes. Therefore, the user must initialize the CRC error bit to zero.

The V(S) and V(R) registers are zeroed upon entering the monitor mode. V(S) is incremented with each frame transmission, and V(R) is incremented upon receiving a frame.

The following 11 commands are valid during monitor mode:
Load RTP
Load TTP
Continue Transmit
Clear Tx/Link Status
Clear Rx/Host Status
Clear Status
Update Status
Dump Parameters
Dump Registers
End Monitor Test
Reset

6.4.5 End Monitor

The end monitor command (hex '93') terminates the monitor or transparent operation. After receiving the end monitor command, the XPC quits monitoring the receive line, negates $\overline{\text{RTS}}$, and quits transmitting flags. This command is only accepted by the XPC when the CONN bit in the mode descriptor register is not set. If the CONN bit is set, the XPC sets the ILL bit in the Rx/host status register.

SECTION 7
SIGNAL DESCRIPTION

This section contains a brief description of the input and output signals of the XPC. Reference is given (if applicable) to other paragraphs that contain more information about the function being performed.

NOTE

> The terms **assertion** and **negation** will be used extensively. This is done to avoid confusion when dealing with a mixture of "active low" and "active high" signals. The terms assert and assertion are used to indicate that a signal is active or true, independent of whether that level is represented by a high or low voltage. The terms negate and negation are used to indicate that a signal is inactive or false.

The XPC has two system bus operation modes: master mode and slave mode. The XPC enters slave mode whenever $\overline{\text{CS}}$ or $\overline{\text{IACK}}$ is asserted. In this mode, the XPC accepts data from or places data on the data bus according to the level on the R/$\overline{\text{W}}$ pin. When in master mode, the XPC is the bus master and is performing memory reads and writes.

The input and output signals can be functionally organized into the groups shown in Figure 7-1. Each of these groups is discussed in the following paragraphs.

7.1 ADDRESS BUS (A1–A31)

This is a 32-bit (when combined with the $\overline{\text{UDS}}$/A0 signal), unidirectional (with the exception of A1 and A2), three-state bus capable of addressing up to 4 gigabytes of memory. A1 and A2 are bidirectional three-state lines that address internal XPC registers in the slave mode and that provide the lower two address outputs in the master mode.

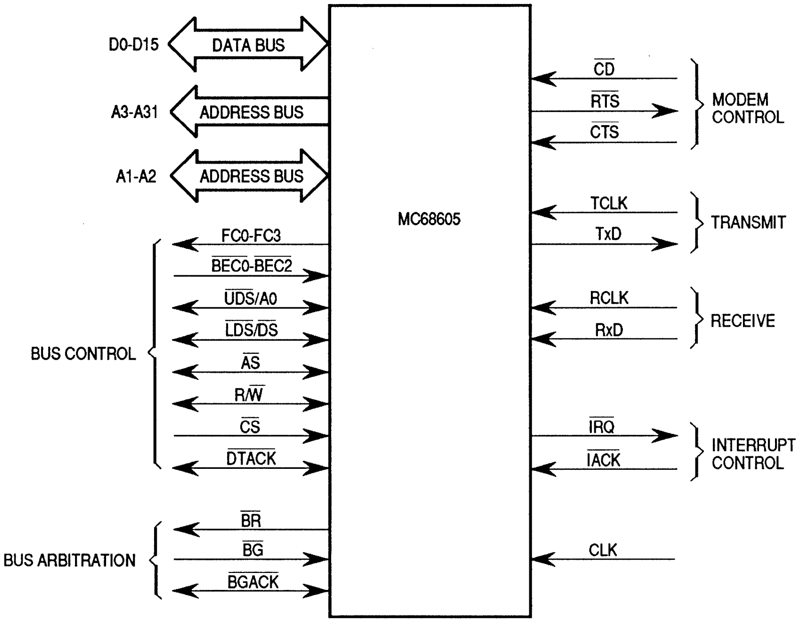


Figure 7-1. Input and Output Signals

7.2 DATA BUS (D0–D15)

The XPC has a 16-bit, bidirectional, three-state bus for the general-purpose data transfer. The XPC can transmit and receive data using an 8-bit or 16-bit data bus. The data bus is used for data input during a host processor write or XPC read cycle and for data output during a host processor read or XPC write cycle.

7.3 FUNCTION CODES (FC0–FC3)

These three-state output pins can be used in the master mode to further qualify the value on the address bus by providing 16 separate address spaces that may be defined by the user. The value placed on these lines by the XPC is taken from one of the internal function code registers, depending on the source register for the address used during the DMA cycle. The XPC performs no checking on the function code value.

7.4 BUS CONTROL

The following paragraphs describe the bus control signals.

7.4.1 Chip Select ($\overline{\text{CS}}$)

This input pin selects the XPC for a host processor bus cycle. When $\overline{\text{CS}}$ is asserted, the address on A1, A2, and the data strobes select the internal XPC register that will be involved in the transfer. $\overline{\text{CS}}$ should be generated by qualifying an address decode signal with address strobe.

7.4.2 Address Strobe ($\overline{\text{AS}}$)

This bidirectional three-state signal is an output in the master mode which indicates that a valid address is present on the address bus. In the slave mode, $\overline{\text{AS}}$ is an input that is monitored to determine when the XPC can take control of the bus (after the XPC has requested and been granted use of the bus).

7.4.3 Read/Write (R/$\overline{\text{W}}$)

This bidirectional three-state signal indicates the direction of the data transfer during a bus cycle. The R/$\overline{\text{W}}$ pin is an input in the slave mode. A high level indicates that the transfer is from the XPC onto the data bus, and a low level indicates that the transfer is from the data bus into the XPC. The R/$\overline{\text{W}}$ pin is an output in the master mode. A high level indicates that the transfer is from the data bus into the XPC, and a low level indicates that the transfer is from the XPC onto the data bus.

7.4.4 Upper Data Strobe ($\overline{\text{UDS}}$/A0) and Lower Data Strobe ($\overline{\text{LDS}}$/$\overline{\text{DS}}$)

These bidirectional three-state signals control the flow of data on the data bus. When using a 16-bit data bus, these pins function as $\overline{\text{UDS}}$ and $\overline{\text{LDS}}$. During any bus cycle, $\overline{\text{UDS}}$ is asserted if data is to be transferred over data lines D8–D15 and $\overline{\text{LDS}}$ is asserted if data is to be transferred over data lines D0–D7. $\overline{\text{UDS}}$ and $\overline{\text{LDS}}$ are controlled by the XPC when operating in the master mode and by the host when operating in slave mode (see Table 7-1).

When using an 8-bit data bus, these pins function as A0 and $\overline{\text{DS}}$. A0 is an extension to the lower address lines to provide the address of a byte in the address map and is valid when A1–A31 are valid. $\overline{\text{DS}}$ is a data strobe that enables external data buffers and indicates that valid data is on the bus during a write cycle (see Table 7-1).

Table 7-1. Data Strobe Control of Data Bus in Master Mode

$\overline{\text{UDS}}$/A0	$\overline{\text{LDS}}/\overline{\text{DS}}$	R/$\overline{\text{W}}$	D8–D15	D0–D7
16-Bit Transfer				
High	High	X	No Valid Data	No Valid Data
Low	Low	X	Valid Data	Valid Data
High	Low	Low	No Valid Data	Valid Data
High	Low	High	X	Valid Data
Low	High	Low	Valid Data	No Valid Data
Low	High	High	Valid Data	X
8-Bit Transfer				
X	Low	Low	No Valid Data	Valid Data
X	Low	High	X	Valid Data
X	High	X	No Valid Data	No Valid Data

X — Don't Care Condition

7.4.5 Data Transfer Acknowledge ($\overline{\text{DTACK}}$)

This bidirectional three-state line signals that the asynchronous bus cycle may be terminated. In the slave processor mode, this output indicates that the XPC has accepted data from the host or placed data on the bus for the host. In the master mode, this input is monitored by the XPC to determine when to terminate a bus cycle. As long as $\overline{\text{DTACK}}$ remains negated, the XPC will insert wait cycles into the bus cycle. When $\overline{\text{DTACK}}$ is asserted, the bus cycle will be terminated.

7.5 BUS ARBITRATION

The three signals discussed in the following paragraphs form a bus arbitration circuit that determines which device in a system will be the current bus master.

7.5.1 Bus Request ($\overline{\text{BR}}$)

This open-drain output pin is asserted by the XPC to request control of the bus. $\overline{\text{BR}}$ is wire-ORed with all other devices that may be bus masters.

7.5.2 Bus Grant ($\overline{\text{BG}}$)

This input is asserted by an external bus arbiter to inform the XPC that it may assume bus mastership as soon as the current bus cycle is completed. The XPC will not take control of the bus until $\overline{\text{AS}}$ and $\overline{\text{BGACK}}$ are negated and the $\overline{\text{BEC}}$ lines are encoded as normal mode.

7.5.3 Bus Grant Acknowledge ($\overline{\text{BGACK}}$)

This bidirectional three-state signal is asserted by the XPC to indicate that it is the current bus master. $\overline{\text{BGACK}}$ is monitored as an input to determine when the XPC can become bus master. $\overline{\text{BGACK}}$ is not asserted as an output until the following conditions are met:

1. $\overline{\text{BR}}$ is asserted,
2. $\overline{\text{BG}}$ is asserted,
3. $\overline{\text{AS}}$ is inactive, indicating that the current bus cycle has ended,
4. $\overline{\text{BGACK}}$ is inactive, indicating that no other device is claiming bus mastership, and
5. $\overline{\text{BEC}}$ lines are encoded as normal mode.

7.6 INTERRUPT CONTROL

The two signals discussed in the following paragraphs form an interrupt request/acknowledge handshake circuit with a host processor.

7.6.1 Interrupt Request ($\overline{\mathrm{IRQ}}$)

This open-drain output is asserted by the XPC to request service from the host.

7.6.2 Interrupt Acknowledge ($\overline{\mathrm{IACK}}$)

This input is asserted by the host to acknowledge that it has received an interrupt from the XPC. In response to the assertion of $\overline{\mathrm{IACK}}$, the XPC will place a vector on D0–D7 that is used by the host to fetch the address of the proper XPC interrupt handler routine.

7.7 BUS EXCEPTION CONDITIONS ($\overline{\mathrm{BEC0}}$–$\overline{\mathrm{BEC2}}$)

These input lines provide an encoded signal that indicates an abnormal bus condition such as a bus error or reset. For more detailed information refer to **8.3 BUS EXCEPTION CONTROL**.

7.8 CLOCK (CLK)

This input signal is the XPC parallel clock. This signal can range from 4 to 16 MHz. The XPC can operate with a CLK input, which is synchronous or asynchronous with respect to the host clock, as long as the bus requirements are satisfied.

7

7.9 MODEM CONTROL

The following paragraphs describe the modem control signals.

7.9.1 Carrier Detect ($\overline{\mathrm{CD}}$)

The XPC considers the data on the RxD pin to be valid only if the $\overline{\mathrm{CD}}$ input pin is asserted. If the $\overline{\mathrm{CD}}$ pin is negated for one Tx clock cycle during frame reception, the XPC sets the abort bit in the Rx/host status register.

7.9.2 Request-to-Send ($\overline{\text{RTS}}$)

The XPC asserts the $\overline{\text{RTS}}$ output pin upon executing the load addresses or monitor test commands. At this point, the XPC starts transmitting continuous flags. The XPC negates the $\overline{\text{RTS}}$ pin upon hardware or software reset or upon executing the end monitor or serial loopback command.

7.9.3 Clear-to-Send ($\overline{\text{CTS}}$)

If the $\overline{\text{CTS}}$ input pin is not asserted within 68 cycles of the Tx clock following the assertion of $\overline{\text{RTS}}$ or if the $\overline{\text{CTS}}$ pin is negated for more than one cycle while transmitting interframe flags, the XPC sets the clear to send lost (CTSL) bit in the Tx/link status register. If the $\overline{\text{CTS}}$ pin is negated for more than one cycle during a frame transmission, the XPC sets the CTSL bit in the Tx/link status register and aborts the current frame.

7.10 TRANSMIT

The following paragraphs describe the transmit signals.

7.10.1 Transmit Clock (TCLK)

This input signal can range from dc to 10 MHz. The XPC synchronizes the transmit data to this clock. Ten transmit clock cycles are required during reset to initialize the serial logic. The TCLK must be a clean signal which meets the specified electrical characteristics. Any noise in the TCLK signal will cause the XPC to behave erratically.

7.10.2 Transmit Data (TxD)

This output pin is used to send the serial bit stream.

7.11 RECEIVE

The following paragraphs describe the receive signals.

7.11.1 Receive Clock (RCLK)

This input signal can range from dc to 10 MHz. The receive data is synchronized to this clock. The RCLK must be a clean signal which meets the specified electrical characteristics. Any noise in the RCLK signal will cause the XPC to behave erratically.

7.11.2 Receive Data (RxD)

This input line receives the serial bit stream from the communications link synchronized to the receive clock.

7.12 SIGNAL SUMMARY

Table 7-2 is a summary of all the signals discussed in the previous paragraphs.

Table 7-2. Signal Summary

Signal Name	Mnemonic	Input/Output	Active State	Driver Type
Address Bus	A1–A2	Input/Output		Three-State
Address Bus	A3–A31	Output		Three-State
Data Bus	D0–D15	Input/Output		Three-State
Function Codes	FC0–FC3	Output		Three-State
Bus Exception Codes	$\overline{\text{BEC0}}$–$\overline{\text{BEC2}}$	Input	Low	
Upper Data Strobe	$\overline{\text{UDS}}$/A0	Input/Output	Low	Three-State[1]
Lower Data Strobe	$\overline{\text{LDS}}$/$\overline{\text{DS}}$	Input/Output	Low	Three-State[1]
Address Strobe	$\overline{\text{AS}}$	Input/Output	Low	Three-State[1]
Read/Write	R/$\overline{\text{W}}$	Input/Output	High/Low	Three-State[1]
Chip Select	$\overline{\text{CS}}$	Input	Low	
Data Transfer Acknowledge	$\overline{\text{DTACK}}$	Input/Output	Low	Three-State[1]
Bus Request	$\overline{\text{BR}}$	Output	Low	Open-Drain[2]
Bus Grant	$\overline{\text{BG}}$	Input	Low	
Bus Grant Acknowledge	$\overline{\text{BGACK}}$	Input/Output	Low	Three-State
Carrier Detect	$\overline{\text{CD}}$	Input	Low	
Request-to-Send	$\overline{\text{RTS}}$	Output	Low	Normal
Clear-to-Send	$\overline{\text{CTS}}$	Input	Low	
Transmit Clock	TCLK	Input		
Transmit Data	TxD	Output		Normal
Receive Clock	RCLK	Input		
Receive Data	RxD	Input		
Interrupt Request	$\overline{\text{IRQ}}$	Output	Low	Open-Drain[2]
Interrupt Acknowledge	$\overline{\text{IACK}}$	Input	Low	
Clock	CLK	Input		

[1] These signals require a pullup resistor to maintain a high voltage when in the high-impedance or negated state. However, when these signals go to the high-impedance state, they will first drive the pin high momentarily to reduce the signal rise time.

[2] These signals are wire-ORed and require a pullup resistor to maintain a high voltage when not driven.

7

SECTION 8
BUS OPERATION

The following section describes the bus signal operation of the XPC during bus arbitration, bus exception conditions, slave operations, master operations, and the reset operation. Functional timing diagrams are included to assist in the definition of signal timing; however, these diagrams are not intended as parametric timing definitions. For detailed relationships, refer to **SECTION 10 ELECTRICAL SPECIFICATIONS**.

8.1 SLAVE OPERATION MODE

In the slave operation mode, the XPC is a peripheral slave to the bus master. The XPC enters the slave operation mode when chip select or interrupt acknowledge is asserted. During slave mode operations, the XPC accepts data from, or places data on the data bus according to the level on the R/$\overline{W}$ pin. The data transferred will either be loaded into or read from the internal register that is selected by the encoding of A1 and A2 and the data strobes (refer to Figure 4-1). This mode of operation is used during XPC initialization to load system configuration information and the initial station table pointer and function code into the XPC. After initialization, the slave mode of operation is used by the host processor to place commands into the XPC command register.

In slave mode, the XPC can operate with a CLK input, which is synchronous or asynchronous with respect to the host clock, as long as the bus requirements are satisfied. In the functional diagrams showing host operations, the bus master is assumed to be an M68000 Family processor with a clock signal identical to the XPC CLK signal. The state numbers (S0, S1, etc.) refer to the numbering convention for those processors.

8.1.1 Host Processor Read Cycles

During host processor read cycles, the XPC places data on the data bus and asserts $\overline{DTACK}$ to indicate to the bus master that the data is valid. Figure 8-1 shows the functional timing for a word read cycle on a 16-bit data bus. The timing for even- and odd-byte host reads on a 16-bit data bus or any host read on an 8-bit bus are identical, with the encoding of UDS/A0 and LDS/DS selecting the proper byte. The 8-bit semaphore register is always selected during a host processor read cycle, regardless of the A1, A2 encoding, as this is the only XPC register that is directly readable by the host processor. When the upper data bus is selected during a host read cycle, the XPC drives D8–D15 to 'FF'.

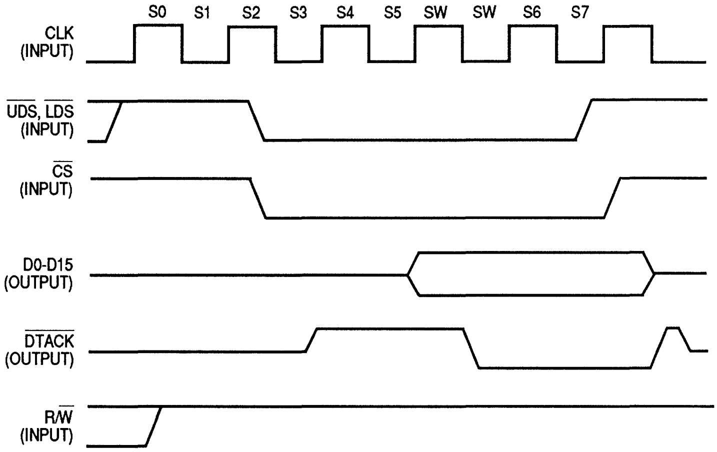


Figure 8-1. Host Processor Read Cycle

The XPC begins a host read cycle when CS is asserted and the R/W line is high. The XPC responds to CS by decoding UDS/A0 and LDS/DS, driving the appropriate data lines, and asserting DTACK. The XPC then waits until both UDS/A0 and LDS/DS or CS is negated, three-states the data lines, negates and three-states DTACK. The content of the semaphore register is always hex 'FE' or hex 'FF'.

8.1.2 Host Processor Write Cycles

During host processor write cycles, the XPC accepts data from the data bus and asserts DTACK to indicate to the bus master that the data has been loaded into the selected register. The only XPC registers that are directly writable by the host processor are the command register (CR), interrupt vector register (IV), and the data register (DR). The timing is identical for even- and odd-byte host processor writes to a 16-bit data bus or any host processor write to an 8-bit data bus. The proper byte is selected by the encodings of UDS/A0 and LDS/DS signals.

A host processor write cycle begins when CS is asserted and R/W is low. The XPC responds by decoding A1, A2, UDS/A0, and LDS/DS signals. When a valid register (CR, IV, or DR) is selected, the XPC accepts the data from the data bus, places the data into the selected register, and asserts DTACK. Next, the XPC waits for both LDS/DS and UDS/A0 or CS to be negated and then negates and three-states DTACK. The timing for this operation is shown in Figure 8-2.

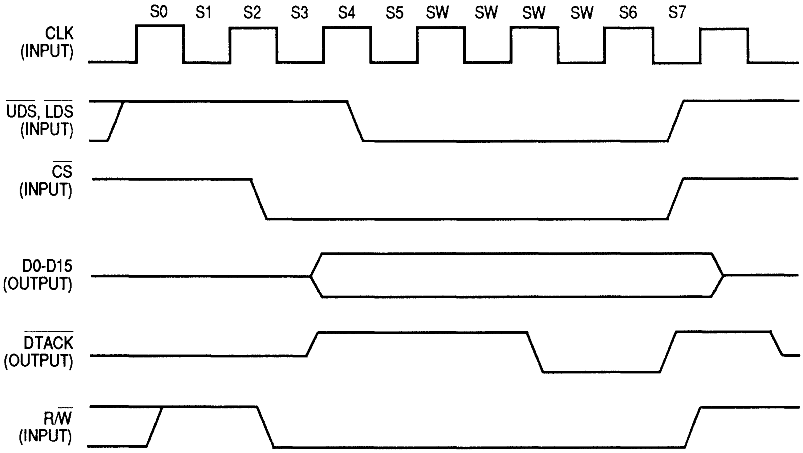


Figure 8-2. Host Processor Write Cycle

8

8.1.3 Interrupt Acknowledge Cycles

During interrupt acknowledge cycles, the host processor is responding to an interrupt request from the XPC. The timing of an interrupt acknowledge cycle is identical to an odd-byte read cycle, except that it is started by the assertion of an IACK signal rather than CS. CS and IACK are mutually exclusive signals and should not be asserted at the same time. If IACK is asserted when the XPC is bus master, an address error is generated.

The interrupt acknowledge operation is started by the XPC when IACK is asserted and LDS/DS is asserted. The XPC responds to IACK by placing a vector number on D0–D7 and asserting DTACK. The vector number remains valid on the data bus until IACK or LDS is negated by the host processor, at which time the XPC will three-state the data lines, negate and three-state DTACK. The timing for this operation is shown in Figure 8-3.

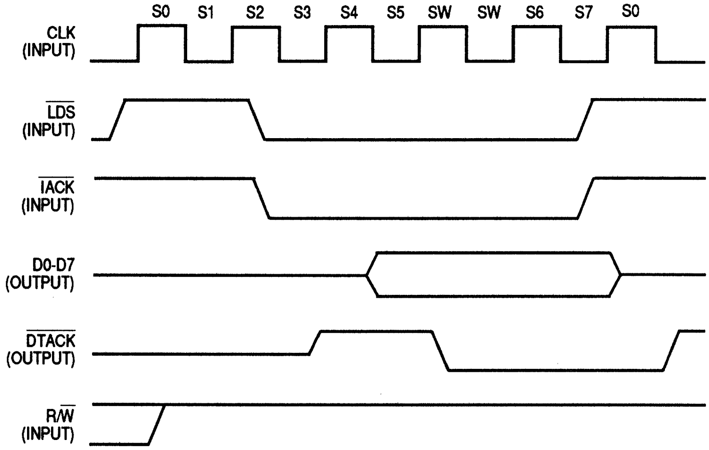


Figure 8-3. Interrupt Acknowledge Cycle

8.2 MASTER OPERATION MODE

In the master operation mode, the XPC is the bus master and performs memory read and write operations. The XPC can operate in either an 8-bit or a 16-bit bus configuration.

8.2.1 XPC Read Cycles

During a DMA read operation, the XPC controls the transfer of data from memory into the XPC. The functional timing for a DMA read operation is shown in Figure 8-4. The timing for an even- or odd-byte read on a 16-bit data bus or any read on an 8-bit data bus is identical, with the encoding of UDS/A0 and LDS/DS selecting the proper byte.

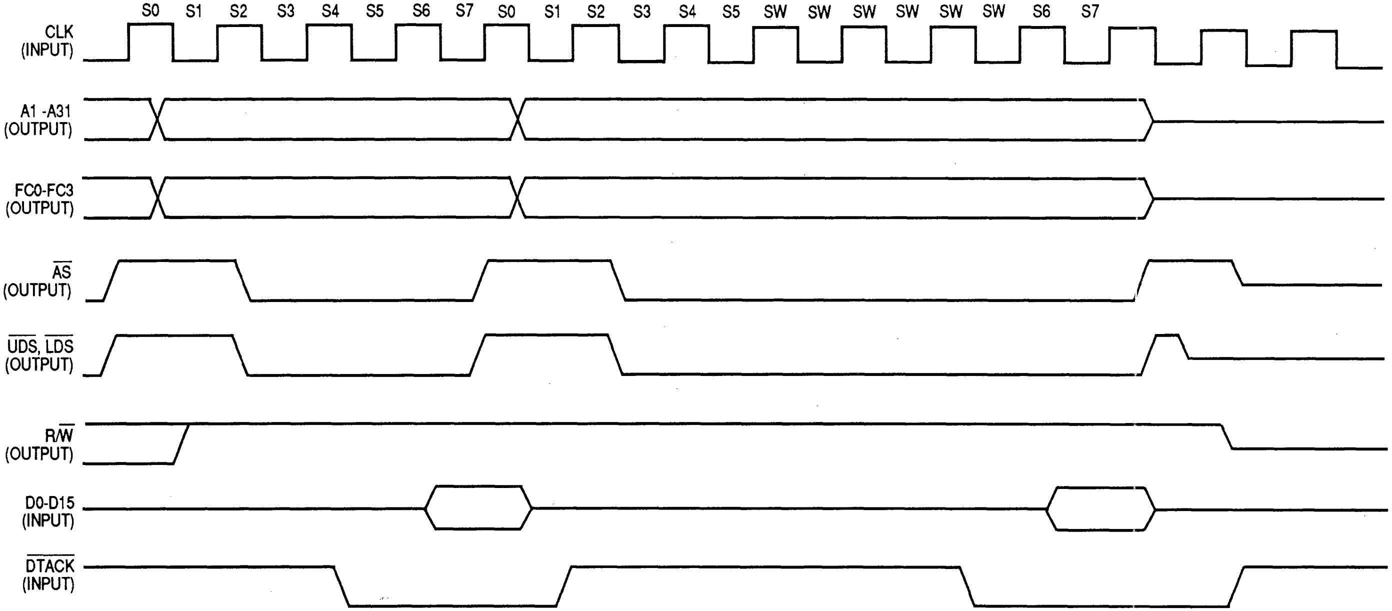


Figure 8-4. Read Cycle and Slow Read Cycle

The XPC drives FC0–FC3 and A1–A31 pins with the address of the memory location that the XPC wants to read. Then R/$\overline{\text{W}}$ is driven high, $\overline{\text{AS}}$, $\overline{\text{UDS}}$/A0, and/or $\overline{\text{LDS}}$/$\overline{\text{DS}}$ are asserted. $\overline{\text{DTACK}}$ is asserted by memory when valid data from memory is on lines D0–D15. If using an 8-bit bus, only the data on lines D0–D7 is assumed to be valid. When $\overline{\text{DTACK}}$ is asserted, the data is latched by the XPC from the data lines, and the bus cycle is terminated.

8.2.2 XPC Write Cycles

During a DMA write operation, the XPC controls the transfer of data to memory from the XPC. The functional timing for this operation is shown in Figure 8-5. The timing for an even- or odd-byte write on a 16-bit data bus or any write to an 8-bit data bus is identical, with encoding of $\overline{\text{UDS}}$/A0 and $\overline{\text{LDS}}$/$\overline{\text{DS}}$ selecting the proper byte.

The XPC drives FC0–FC3 and A1–A31 pins with the address of the memory location to be written. Then R/$\overline{\text{W}}$ is driven low, $\overline{\text{AS}}$ is asserted, and, depending on data size, $\overline{\text{UDS}}$/A0 and/or $\overline{\text{LDS}}$/$\overline{\text{DS}}$ are asserted. Data to be written to memory is placed on the bus and, when $\overline{\text{DTACK}}$ is asserted, the cycle is terminated. On a slow write, as shown in Figure 8-5, the bus cycle is extended because $\overline{\text{DTACK}}$ is not asserted by the end of S4.

8.2.3 XPC DMA Priority Scheme

The XPC has four DMA channels which are used to access the receive specification table, the receive memory buffers, the transmit specification table, and the transmit memory buffers. During one DMA burst, the XPC can access one or more of the following: the station table, the dump area, the receive specification table, the receive memory buffers, the transmit specification table, and/or the transmit memory buffers. Commands from the host that are received between DMA bursts are handled by the XPC microcode and have a higher priority than the receiver and transmitter. For each memory access, an internal arbiter determines which section of the XPC requires priority: the microcode controller, the receiver, or the transmitter. The following priority scheme is used for memory accesses:

1. microcode controller, if execution of microcode requires a memory access
2. receiver, if Rx FIFO ⩾7 full words
3. transmitter, if Tx FIFO ⩾7 empty words
4. receiver, if Rx FIFO ⩾3 full words

5. transmitter, if Tx FIFO ⩾3 empty words
6. receiver, if Rx FIFO < 3 full words and end of receive frame
7. transmitter, if Tx FIFO < 3 empty words and end of transmit frame

where Rx FIFO counts the number of full words and Tx FIFO counts the number of empty words.

For example, the Rx FIFO has 6 full words, the Tx FIFO has 8 empty words, and no commands have been received from the host. The sequential memory cycles are: Tx, Tx, Rx, Rx, Rx, Rx, Tx, Tx, Tx, Tx, Rx, Rx, Tx, and Tx. As the state of the Rx FIFO and Tx FIFO changes, the priority of the required memory accesses for each changes.

8.2.4 XPC Memory Bandwidth Requirements

The serial bit rate and system clock frequency determine the percent of memory bandwidth required by the XPC. The serial bit rate determines how often the XPC has to request the bus, and the system clock frequency and memory access time affect how long the XPC will control the bus during each DMA burst. If the memory is slow and/or there are other bus masters in the system, then Tx FIFO underruns and Rx FIFO overruns are more likely to occur.

The following equation can be used to estimate the average percent of memory bandwidth used by the XPC during transmit and receive operations:

$$\%\text{ of memory bandwidth} = \overset{\textbf{Term 1}}{\left[\frac{(4+W)\bullet T_{sys}}{BUSW\bullet T_{ser}}\right]}\bullet\overset{\textbf{Term 2}}{\left[1+\left(\frac{8+10}{2\bullet F}\right)\right]}\bullet\overset{\textbf{Term 3}}{\left[\frac{F}{F+H+N}\right]}\bullet\quad Q$$

where

T_{sys} = period of the system clock

T_{ser} = period of the serial clock

W = number of memory wait states

4 = number of system clocks per memory cycle

BUSW = the data bus width, either 8 or 16 bits

$(4+W) \bullet T_{sys}$ = the memory cycle time

$\mathrm{BUSW} \bullet T_{ser}$ = the time to receive or transmit the data which can be accessed in one memory cycle

Term 1 = the ratio of the time to access the receive or transmit data in memory to the time it takes to receive or transmit the data on the serial line

8 = the number of bytes accessed in the transmit frame descriptor in order to transmit a frame (frame overhead)

10 = the number of bytes accessed in the receive frame descriptor in order to receive a frame (frame overhead)

F = the average number of bytes of data in the information field of transmit and receive frames

2 = required because the XPC is full duplex

Term 2 = factor necessary to account for frame overhead on receive and transmit

H = 4 for basic mode (1 byte address, 1 byte control, and 2 bytes CRC)
5 for extended mode (1 byte address, 2 bytes control, and 2 bytes CRC)
Note: if extended CRC is used, 2 more bytes must be added

N = the number of pad flags between frames

Term 3 = percent of the serial line used for transmission and reception of data

Q = overhead for bus arbitrations
1.33 for $F_{ser} < 4$ MHz, 16-bit bus
1.2 for $F_{ser} > 4$ MHz, 16-bit bus
1.16 for $F_{ser} < 4$ MHz, 8-bit bus
1.1 for $F_{ser} > 4$ MHz, 8-bit bus

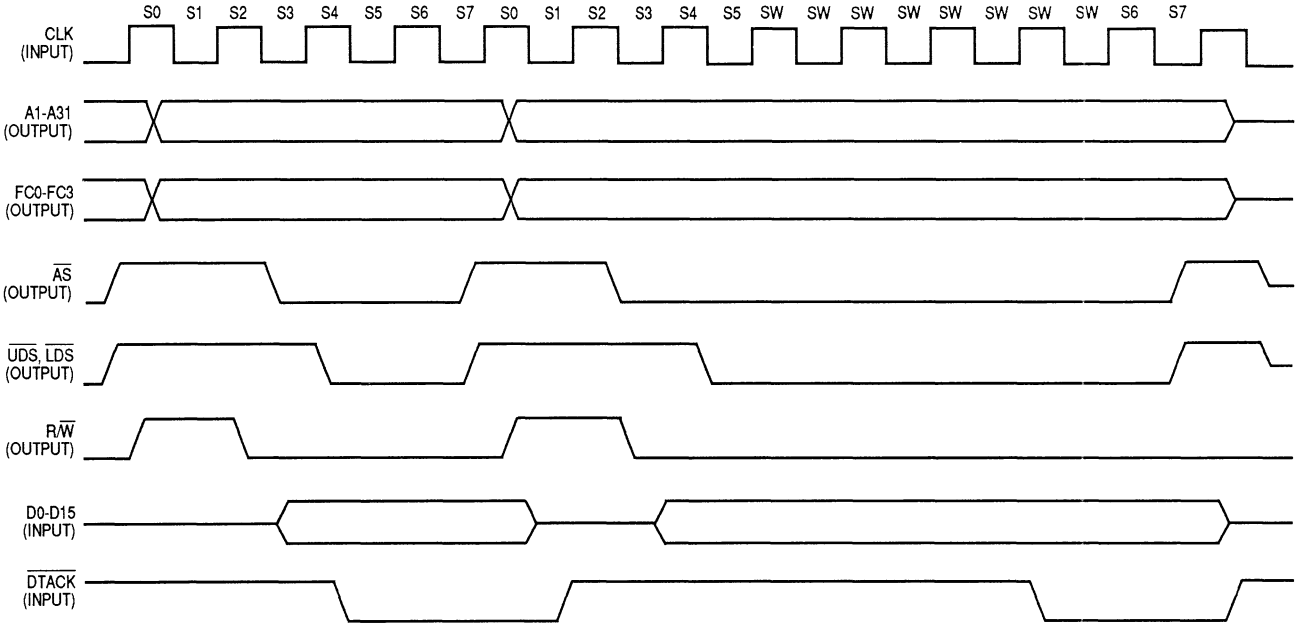


Figure 8-5. Write Cycle and Slow Write Cycle

8.3 BUS EXCEPTION CONTROL

To fully support the M68000 bus architecture, the XPC has three encoded inputs, BEC0–BEC2, that indicate abnormal bus cycle termination conditions. These three lines function in a similar manner to the RESET, HALT, and BERR signals on an M68000 processor, but have different definitions than the processor counterparts. Except for RESET, bus exception encodings are ignored if the XPC is not the bus master. Table 8-1 shows the definition of each encoding of the BEC0–BEC2 pins. Figure 8-6 illustrates how the three inputs may be generated from signals normally present in an M68000 system.

Table 8-1. BEC Encoding Definitions

BEC2	BEC1	BEC0	Definition	XPC Action
High	High	High	No Exception	No Affect
High	High	Low	Halt (Release Bus)	Halt after DTACK and Release the Bus
High	Low	High	Bus Error	Terminate the Current Cycle and Release the Bus
High	Low	Low	Retry	Terminate the Current Cycle and Rerun the Same Cycle Again After the Exception Disappears
Low	High	High	Relinquish and Retry	Terminate the Current Cycle, Release the Bus — Rerun Last Cycle after Rearbitration
Low	High	Low	Undefined, Reserved	Current Cycle Terminated with DTACK, but No Bus Cycles Run until BEC = No Exception
Low	Low	High	Undefined, Reserved	Current Cycle Terminated with DTACK, but No Bus Cycles Run until BEC = No Exception
Low	Low	Low	Reset	Reset XPC Registers and Logic

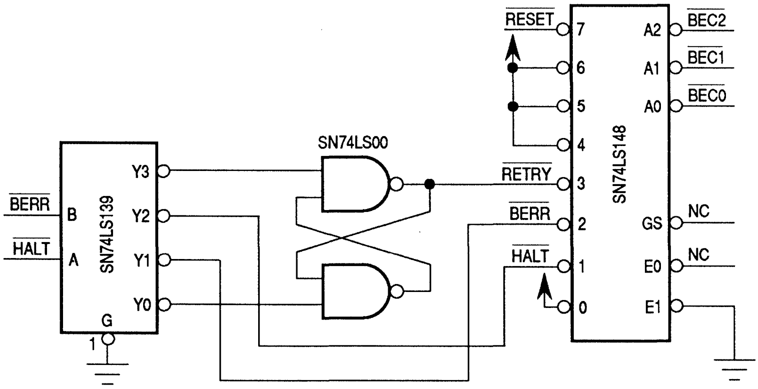


Figure 8-6. Example BEC Signal Generation Circuit

8.3.1 $\overline{\text{BEC0}}$–$\overline{\text{BEC2}}$ Synchronization

The bus exception signals are synchronized as are all asynchronous inputs. The particular clock cycle in which the $\overline{\text{BEC}}$ lines become synchronized internally affects the length of the current bus cycle. In the first case, a very early bus exception occurs in which the bus exception signal is asserted more than one clock cycle before $\overline{\text{DTACK}}$ is asserted. This bus exception causes no delay in the termination of the bus cycle. In case two, which is the typical case, the bus exception occurs in the same clock cycles as $\overline{\text{DTACK}}$. This bus exception causes one clock cycle delay in the termination of the current bus cycle. In the third case, the bus exception occurs in the clock cycle after $\overline{\text{DTACK}}$ was asserted. This bus exception causes one clock delay in the termination of the current bus cycle.

A late bus exception which does not meet electrical specification (58) may cause improper behavior of the XPC, including bus lockup.

8.3.2 Bus Exception Functions

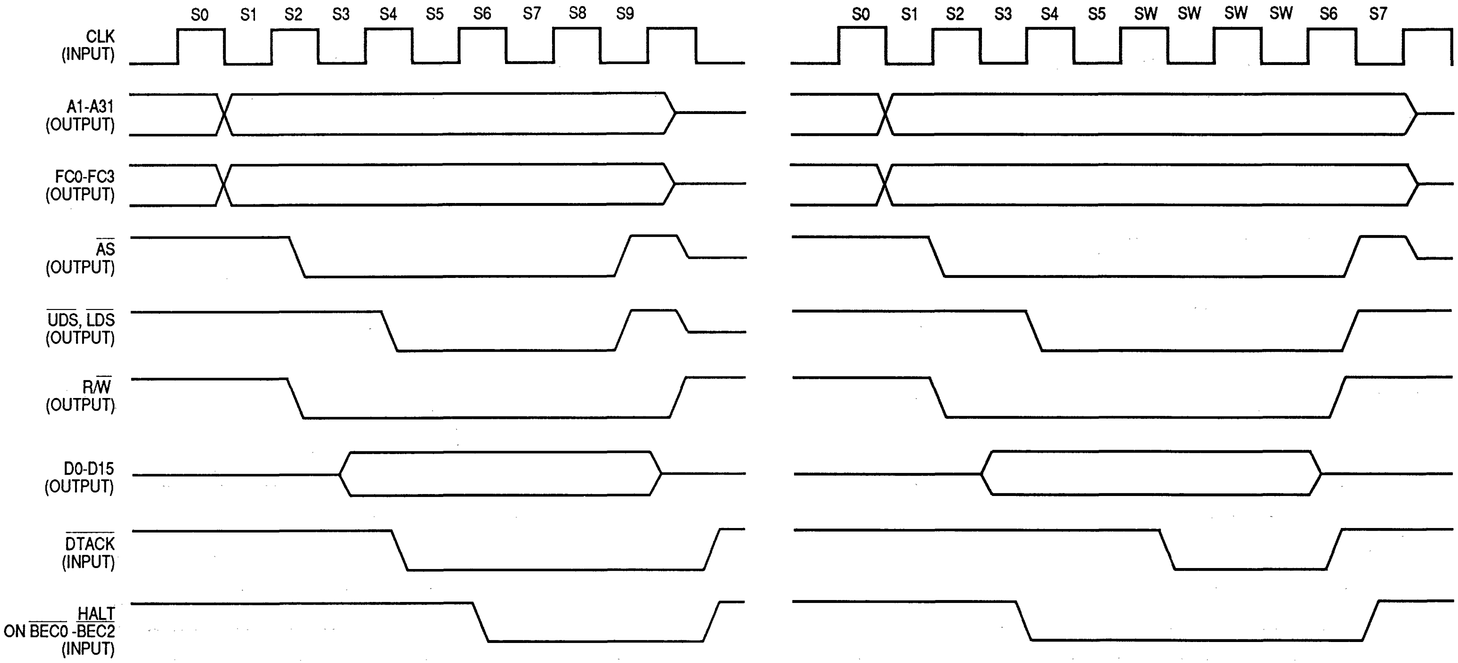
The 3-bit bus exception control code allows eight different bus termination conditions. $\overline{\text{BEC0}}$ is the least significant bit and $\overline{\text{BEC2}}$ is the most significant bit.

8.3.2.1 NORMAL TERMINATION. When HHH is encoded on the $\overline{\text{BEC}}$ pins, the XPC operates in normal mode and $\overline{\text{DTACK}}$ is used to terminate bus cycles.

8

8.3.2.2 HALT. HHL encoded on the $\overline{\text{BEC}}$ pins halts the XPC after the current bus cycle is terminated by the assertion of $\overline{\text{DTACK}}$. The XPC releases ownership of the bus and enters the idle state until the $\overline{\text{BEC}}$ pins return to normal mode (all zeros). At this time, the XPC will rearbitrate for the bus and continue DMA operations if necessary. The halt timing diagram is shown in Figure 8-7.

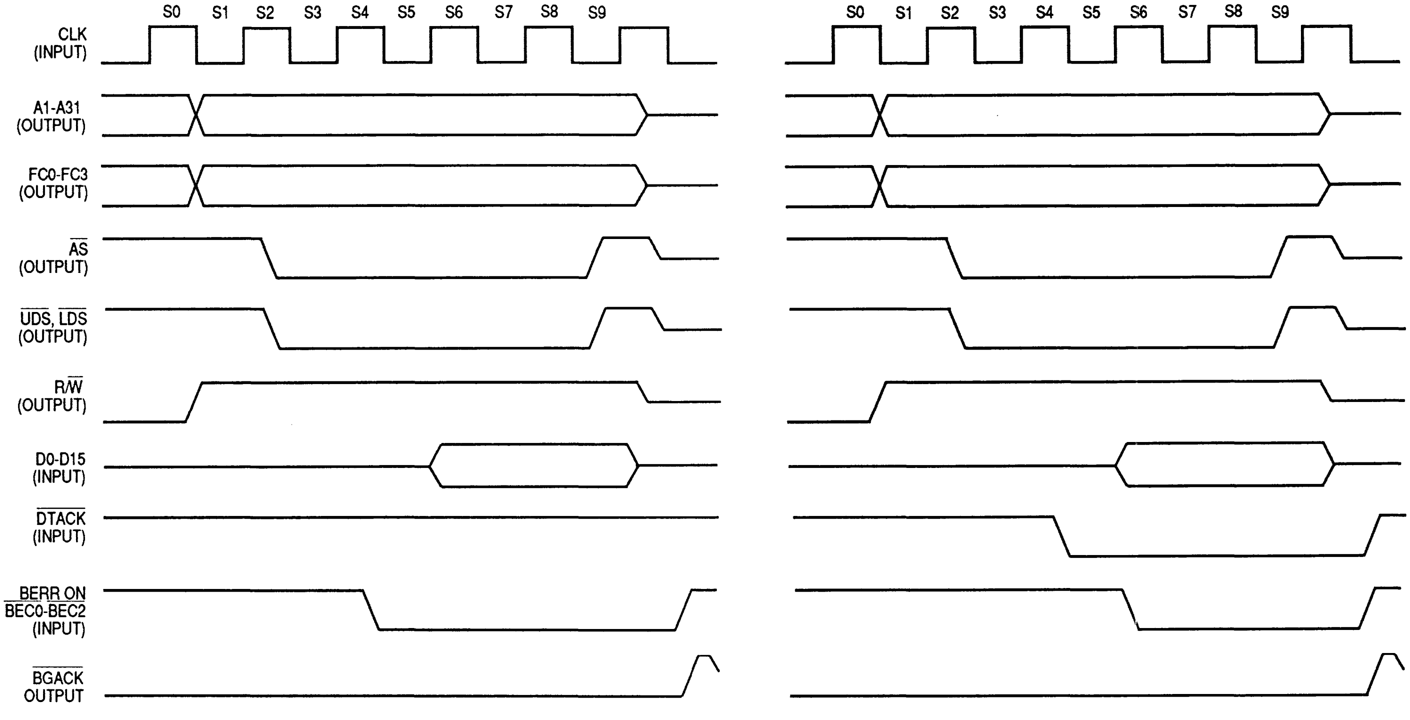
8.3.2.3 BUS ERROR. When HLH is encoded on the $\overline{\text{BEC}}$ pins, the XPC aborts the current bus cycle and releases bus mastership. After the $\overline{\text{BEC}}$ lines return to normal, the XPC reports the bus error to the host processor by writing the address and function code that caused the error into the station table. The XPC also sets the bus/address error bit in the appropriate status register and interrupts the host. The channel in which the bus error occurred is disabled


S0
S1
S2
S3
S4
S5
S6
S7
S8
S9
S0
S1
S2
S3
S4
S5
SW
SW
SW
SW
S6
S7
CLK (INPUT)
A1-A31 (OUTPUT)
FC0-FC3 (OUTPUT)
$\overline{AS}$ (OUTPUT)
$\overline{UDS}$, $\overline{LDS}$ (OUTPUT)
R/$\overline{W}$ (OUTPUT)
D0-D15 (OUTPUT)
$\overline{DTACK}$ (INPUT)
$\overline{HALT}$ ON $\overline{BEC0-BEC2}$ (INPUT)

$\overline{\text{DTACK}}$ ACTIVE BEFORE HALT

$\overline{\text{DTACK}}$ ACTIVE AFTER HALT

Figure 8-7. Write Cycle with Halt



EARLY ASYNCHRONOUS EXCEPTION

LATE ASYNCHRONOUS EXCEPTION

Figure 8-8. Read Cycle with Bus Error

until the host clears the status bit. After the $\overline{\text{BEC}}$ lines return to normal, the XPC may rearbitrate for the bus and continue DMA operations if necessary. A bus error condition is shown in the timing diagram in Figure 8-8.

8.3.2.4 RETRY. When HLL is encoded on the $\overline{\text{BEC}}$ pins, the XPC terminates the current bus cycle and enters a waiting mode. $\overline{\text{BGACK}}$ remains asserted, so the XPC retains bus mastership. When the $\overline{\text{BEC}}$ pins return to normal (all zeros), the XPC reruns the same bus cycle, using the same address and function code. Figures 8-9 and 8-10 show the timing diagram for a retry operation.

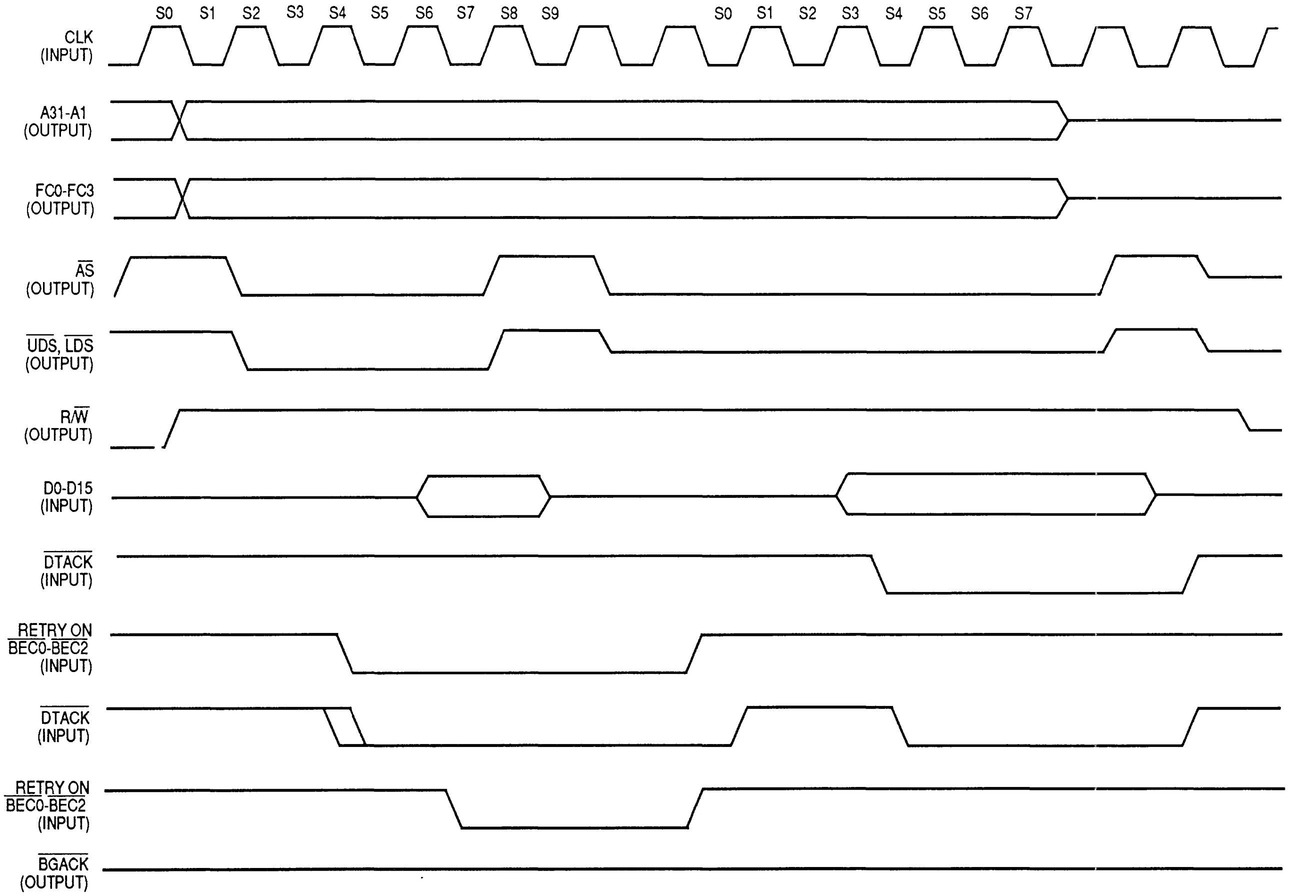
8.3.2.5 RELINQUISH AND RETRY. When the $\overline{\text{BEC}}$ pins are encoded with LHH, the XPC terminates the current bus cycle and relinquishes bus mastership. One debounce delay after the $\overline{\text{BEC}}$ pins have returned to normal, the XPC will rearbitrate for the bus and rerun the same bus cycle as shown in Figure 8-11.

8.3.2.6 RESET. When LLL is encoded on the $\overline{\text{BEC}}$ pins, the XPC executes an internal reset sequence. After completion of the reset, the XPC enters the idle mode.

8.3.2.7 UNDEFINED $\overline{\text{BEC}}$ CODE. When LHL or LLH is encoded on the $\overline{\text{BEC}}$ pins, the XPC takes no action as shown in Figure 8-12. However, the cycle may be extended one more clock cycle due to the debouncing of the $\overline{\text{BEC}}$ lines. The XPC terminates the current cycle after $\overline{\text{DTACK}}$ is asserted. No further bus cycles run until $\overline{\text{BEC}}$ = No Exception.

8.4 BUS ARBITRATION

Once the host has initialized the XPC and the XPC enters connect mode, the XPC uses the M68000 bus arbitration protocol to request bus mastership before entering the master mode of operation. Bus arbitration is a technique used by bus master devices to request, be granted, and acknowledge bus mastership. The bus arbitration timing diagram is shown in Figure 8-13.



NOTE: Two alternatives of DTACK and retry are drawn in the first one. The second cycle is postponed for one additional cycle due to BEC synchronization.

Figure 8-9. Write Cycle with Retry

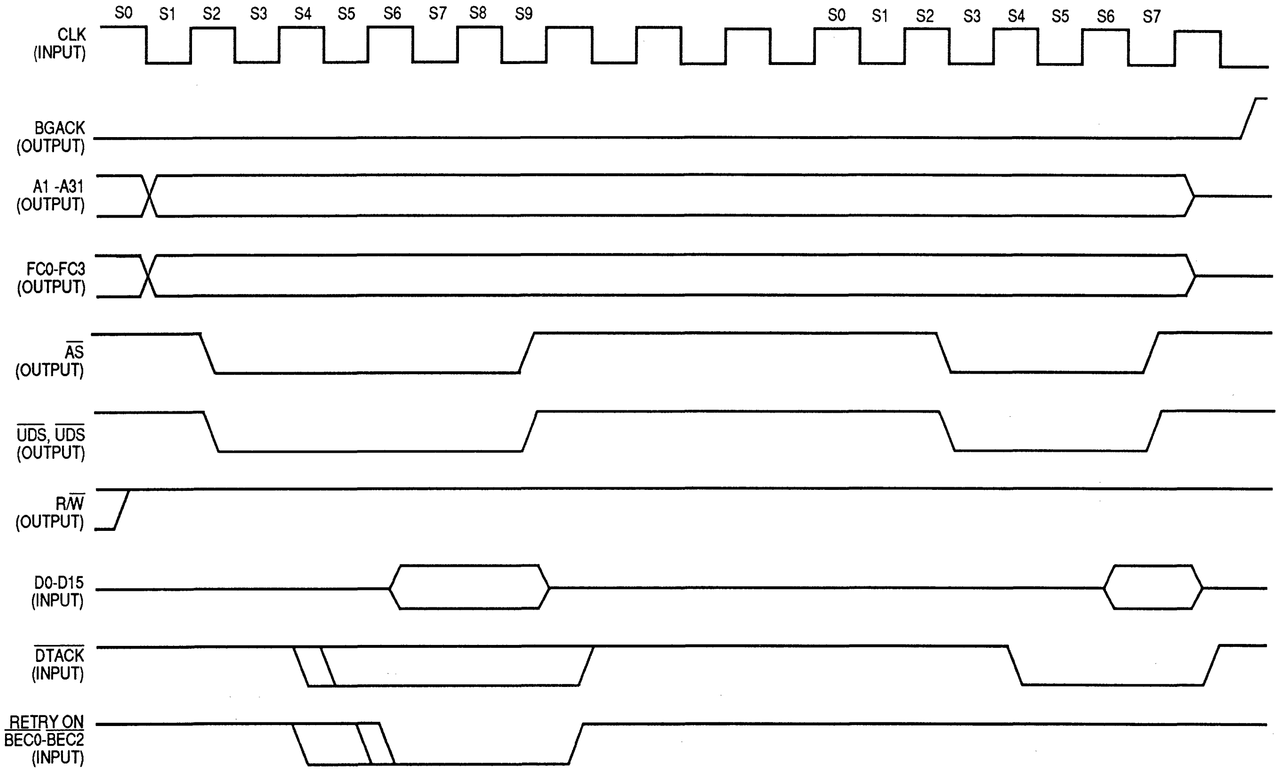
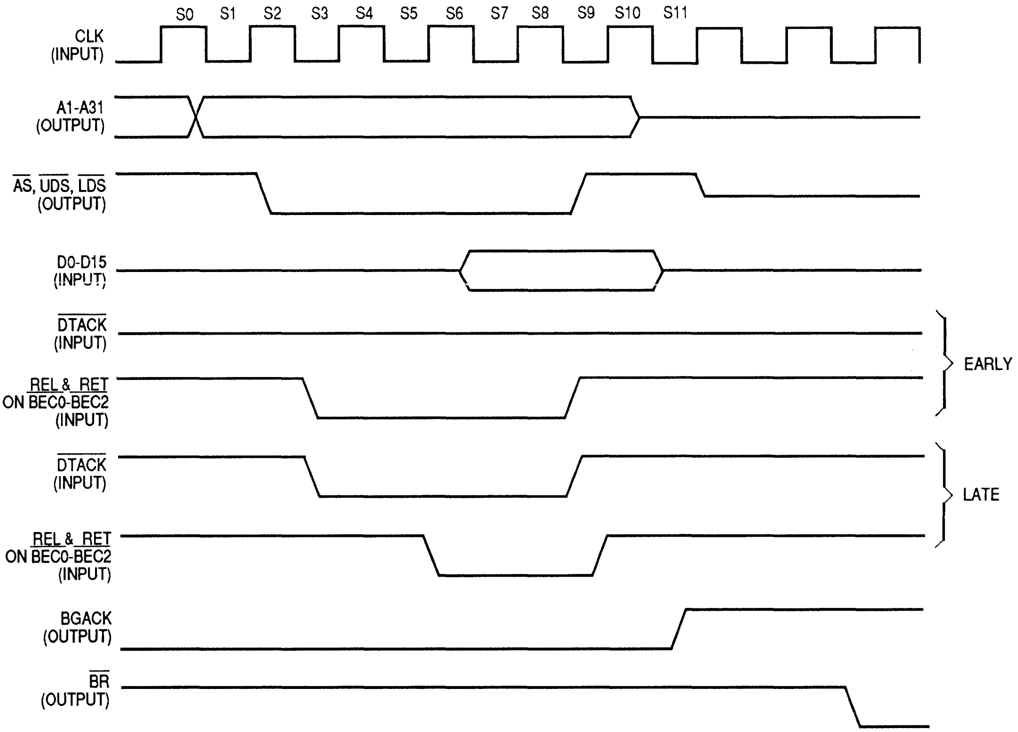


Figure 8-10. Read Cycle with Retry



TWO ALTERNATIVES OF $\overline{\text{DTACK}}$ AND RELINQUISH AND RETRY

Figure 8-11. Read Cycle with Relinquish and Retry

8.4.1 Requesting the Bus

External devices capable of becoming bus masters request the bus by asserting the $\overline{\text{BR}}$ signal. This is a wire-ORed signal that indicates to the external bus arbiter that some external device requires control of the external bus. The XPC requests the bus when a command is issued that requires a DMA operation or when data needs to be moved to or from the internal FIFOs.

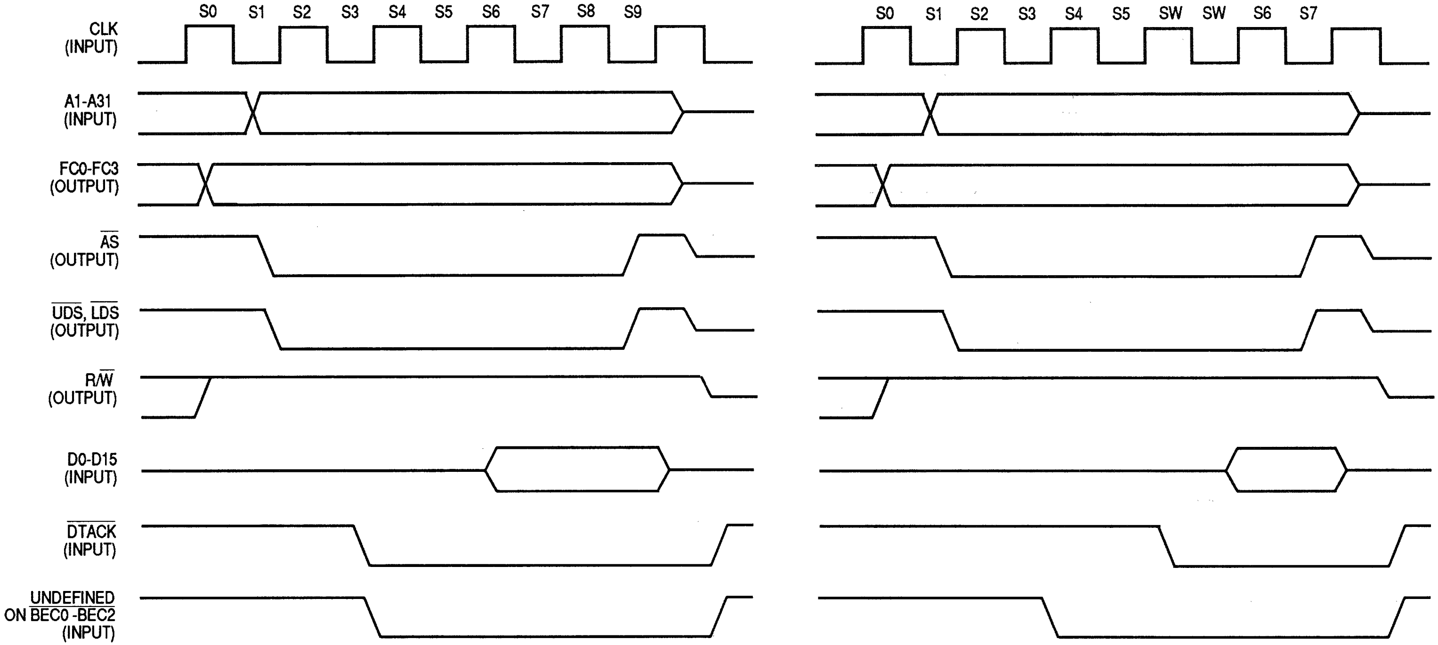

S0
S1
S2
S3
S4
S5
S6
S7
S8
S9
S0
S1
S2
S3
S4
S5
SW
SW
S6
S7
CLK (INPUT)
A1-A31 (INPUT)
FC0-FC3 (OUTPUT)
$\overline{AS}$ (OUTPUT)
$\overline{UDS}$, $\overline{LDS}$ (OUTPUT)
R/$\overline{W}$ (OUTPUT)
D0-D15 (INPUT)
$\overline{DTACK}$ (INPUT)
UNDEFINED ON $\overline{BEC0}$-$\overline{BEC2}$ (INPUT)

Figure 8-12. Read Cycle with Undefined $\overline{BEC0}$–$\overline{BEC2}$

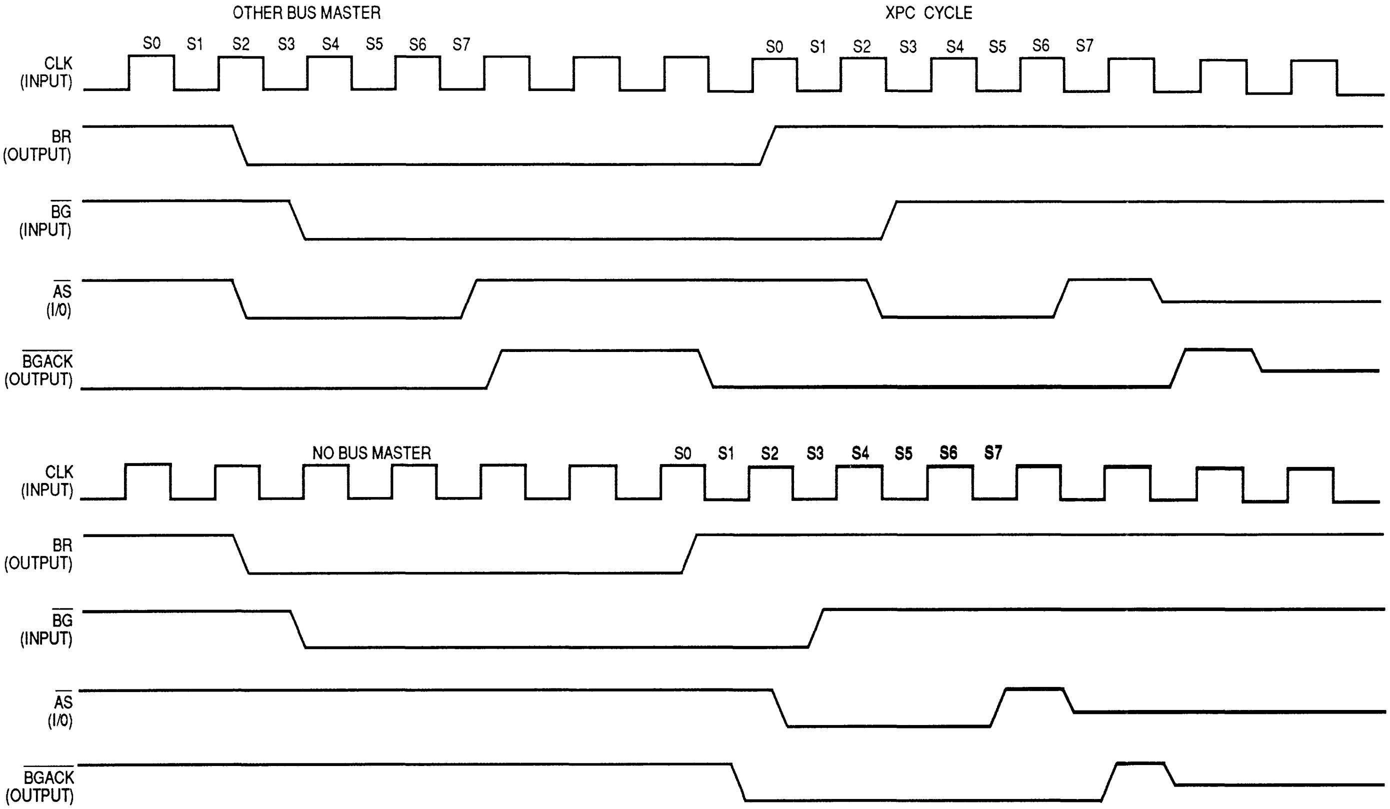


Figure 8-13. Bus Arbitration

8.4.2 Receiving the Bus Grant

The external bus arbiter, which may be a separate unit such as the MC68452 Bus Arbitration Module or the arbiter built into an M68000 processor, will then assert $\overline{\text{BG}}$ to indicate that bus mastership will belong to the XPC as soon as the current bus master has released the bus. $\overline{\text{BG}}$ may be routed through a daisy-chained network or through a specific priority-encoded network. The bus arbiter is not affected by the external method of arbitration as long as the protocol is obeyed.

8.4.3 Acknowledgement of Mastership

Upon receiving a bus grant, the XPC waits until $\overline{\text{AS}}$ and $\overline{\text{BGACK}}$ are negated before asserting $\overline{\text{BGACK}}$. The negation of $\overline{\text{AS}}$ indicates that the previous master has completed its cycle; negation of $\overline{\text{BGACK}}$ indicates that the previous master has released control of the bus. When these conditions are met, the XPC asserts $\overline{\text{BGACK}}$. After $\overline{\text{BGACK}}$ is asserted, $\overline{\text{BR}}$ is negated to allow the external arbiter to begin arbitration for the next bus master. The XPC maintains control of the bus for up to eight bus cycles or until all data transfers have been serviced, based upon the value in the burst control (BRSC) bit of the hardware configuration register. $\overline{\text{BGACK}}$ is negated after the bus cycle(s) is (are) completed. Bus mastership is terminated at the negation of $\overline{\text{BGACK}}$.

8.4.4 Bus Arbitration State Machine

The bus arbitration control unit in the XPC is implemented with a finite state machine. The state diagram is shown in Figure 8-14. All asynchronous signals to the XPC are synchronized before they are used internally. This synchronization is accomplished in a maximum of one cycle of the system clock, assuming that the asynchronous input setup time has been met. The input signal is sampled on the falling edge of the clock and is valid internally after the next rising edge.

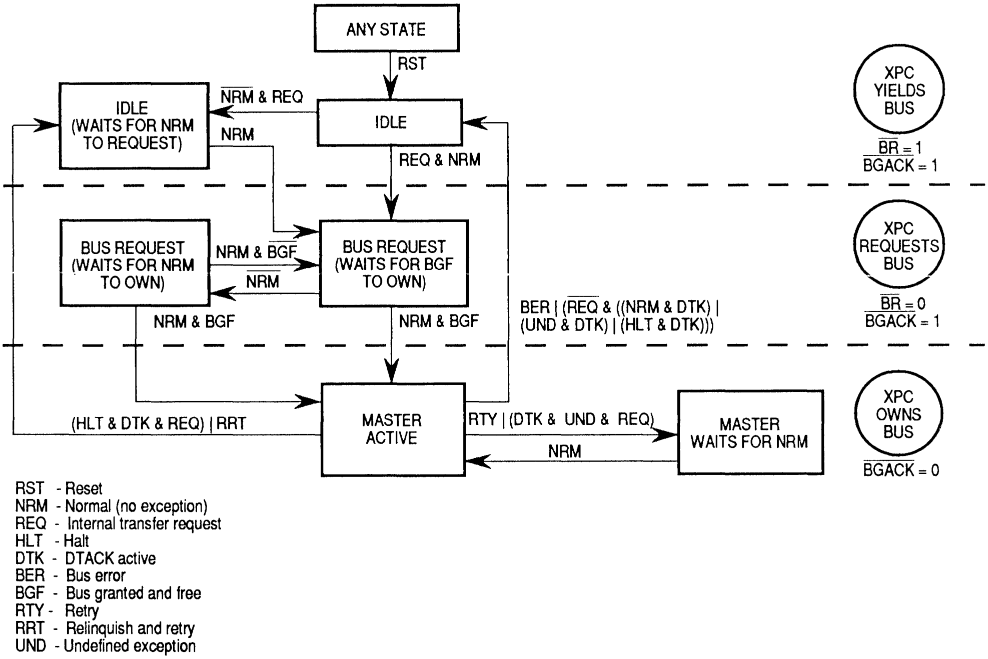


Figure 8-14. MC68605 Bus Arbitration Unit State Diagram

8.5 RESET OPERATION

The XPC is reset by a host processor command (hex 'FF') or by a hardware reset encoded on the $\overline{BEC0}$, $\overline{BEC2}$ lines. A reset encoding on the $\overline{BEC}$ lines should be asserted for at least 10 clock cycles. Normally the XPC requires 32 system clock cycles and ten TCLK cycles before it is ready to accept a command after reset.

8.6 BUS OVERHEAD TIME

In asynchronous bus systems, such as those defined for the M68000 Family, a certain amount of time is used to synchronize incoming signals and is thus "wasted time" since no data transfer activity can take place during those periods. In many applications, the synchronization overhead time required to switch bus masters must be known to predict system behavior. For the

XPC there are two types of overhead: overhead for the XPC to take control of the bus and overhead that occurs when the XPC releases control of the bus. The timing diagram for the front-end and back-end overhead for the XPC is shown in Figure 8-15.

8.6.1 Front-End Overhead

This overhead is the delay that occurs from the time that the host processor terminates a bus cycle by negating $\overline{\text{AS}}$ to when the XPC starts the bus cycle by placing the function codes and address on the bus. It is assumed that $\overline{\text{BG}}$ is asserted and $\overline{\text{BGACK}}$ is negated prior to negation of $\overline{\text{AS}}$ by the host processor so that no additional synchronization delays are introduced by those signals. After one synchronization delay plus 1 1/2 clock cycles, the XPC asserts $\overline{\text{BGACK}}$ to assume control of the bus and begins the DMA cycle. The front-end overhead is between 2 1/2 and 3 1/2 clock cycles.

8.6.2 Back-End Overhead

This overhead is the delay between when the XPC has completed all operations and negated $\overline{\text{AS}}$ and the start of the next bus cycle, which is controlled by another bus master. The XPC negates the $\overline{\text{BGACK}}$ signal one cycle after the last bus cycle. One synchronization delay plus one-half clock cycle later, the new bus master begins the next bus cycle.

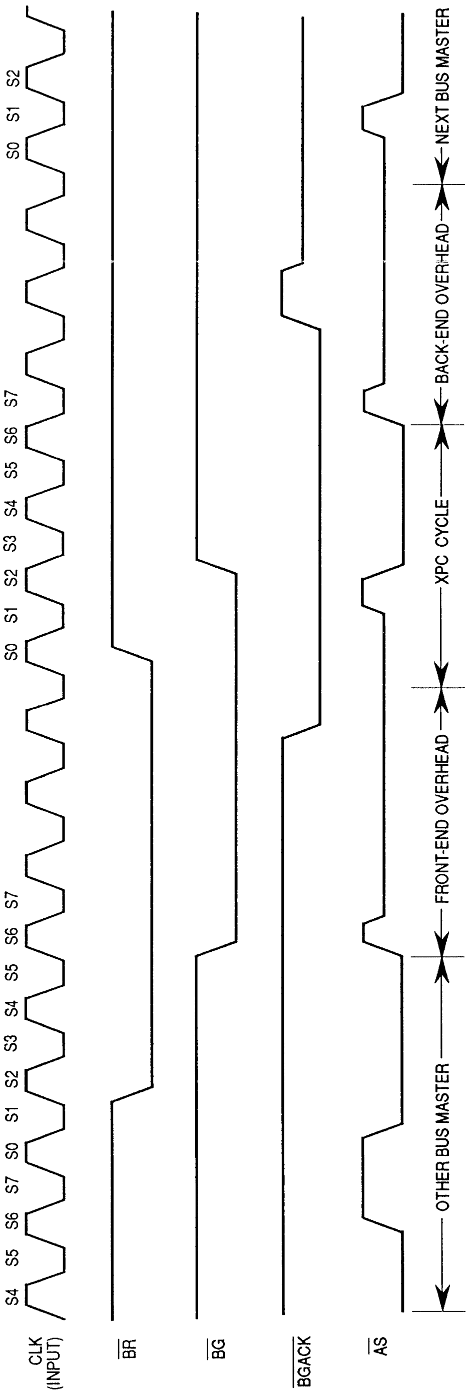


Figure 8-15. Bus Timing Diagram

8

SECTION 9
XPC/HOST PROCESSOR INTERFACE

Figure 9-1 shows the connections between the XPC and a host processor with the XPC having local memory. The configuration for the XPC and the MC68020 microprocessor is shown in Figure 9-2. A typical modem interface is shown in Figure 9-3.

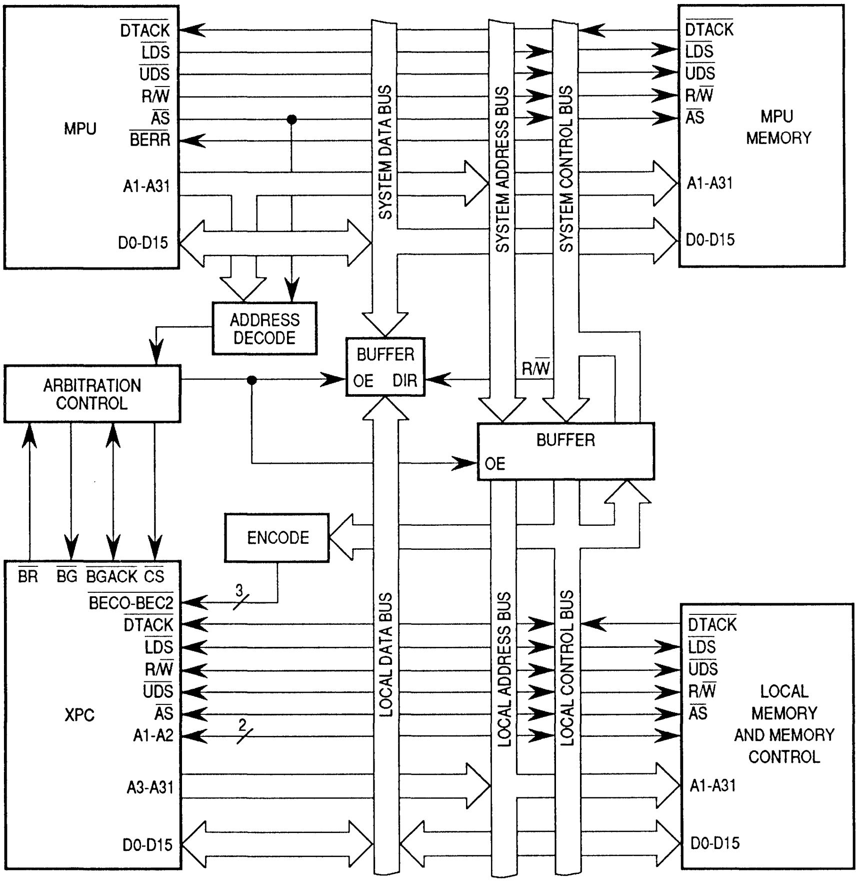


Figure 9-1. XPC-to-Host-Processor Interface

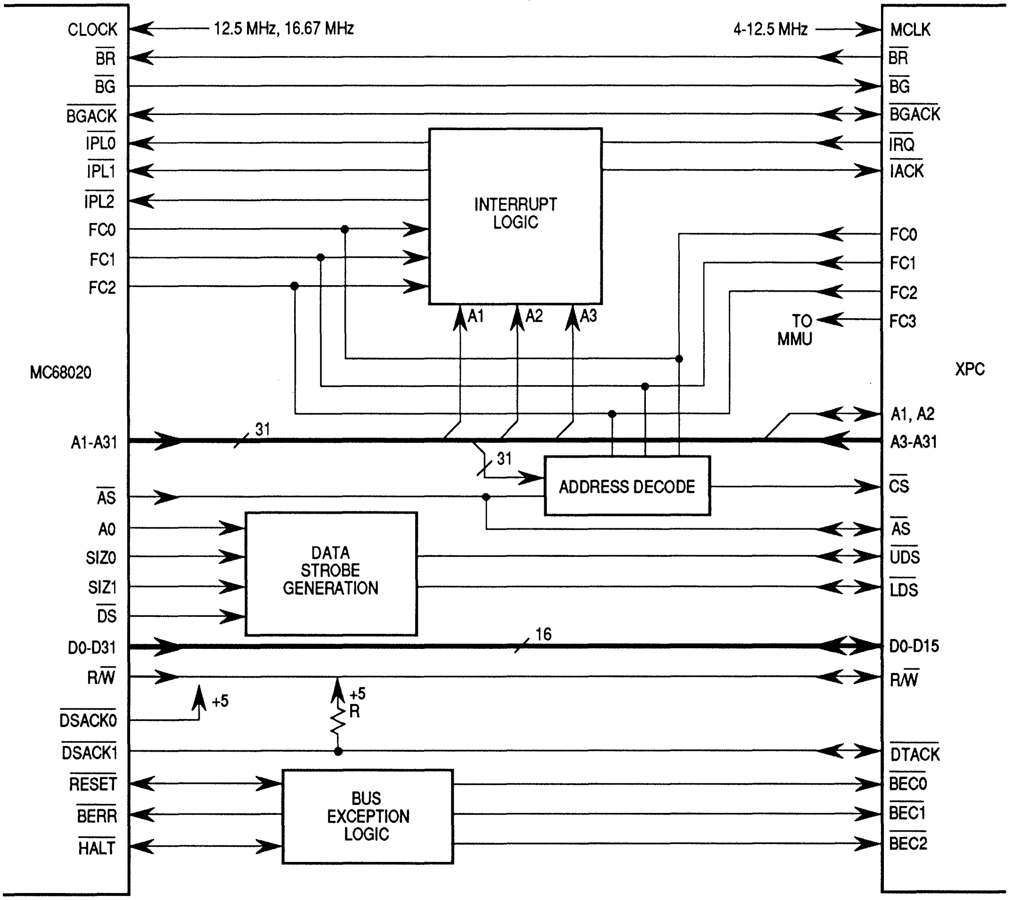


Figure 9-2. XPC-to-MC68020 Interface

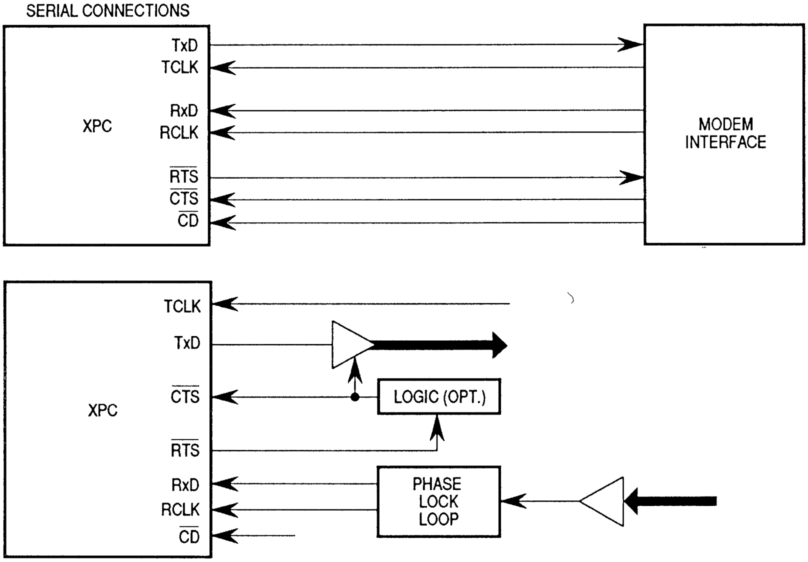


Figure 9-3. Serial Interface Examples

SECTION 10
ELECTRICAL SPECIFICATIONS

This section contains the electrical specifications and associated timing information for the MC68605. Refer to the timing diagrams (Figures 10-1–10-12) found on foldout pages at the back of this document.

10.1 MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Supply Voltage	V_{DD}	−0.3 to +7.0	V
Input Voltage	V_{in}	−0.3 to +7.0	V
Operating Temperature	T_A	0 to 70	°C
Storage Temperature	T_{stg}	−55 to 150	°C

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum-rated voltages to this high-impedance circuit. Reliability of operation is enhanced if unused inputs are tied to an appropriate logic voltage level (e.g., either GND or V_{DD}).

10.2 THERMAL CHARACTERISTICS

Characteristic	Symbol	Value	Unit
Thermal Resistance PGA	θ_{JA}	33	°C/W

$T_J = T_A + (P_D \bullet \theta_{JA})$

$P_D = (V_{DD} \bullet I_{DD}) + P_{I/O}$

where:

$P_{I/O}$ is the lower dissipation on pins (user determined) which can be neglected in most cases.

For $T_A = 70°C$ and $P_D = 0.55$ W @ 12.5 MHz

$T_J = 88°C$

10.3 POWER CONSIDERATIONS

The average chip-junction temperature, T_J, in °C can be obtained from:

$$T_J = T_A + (P_D \cdot \theta_{JA}) \qquad (1)$$

where:

T_A = Ambient Temperature, °C

θ_{JA} = Package Thermal Resistance, Junction-to-Ambient, °C/W

P_D = $P_{INT} + P_{I/O}$

P_{INT} = $I_{DD} \times V_{DD}$, Watts — Chip Internal Power

$P_{I/O}$ = Power Dissipation on Input and Output Pins, Watts — User Determined

For most applications $P_{I/O} < P_{INT}$ and can be neglected.

The following is an approximate relationship between P_D and T_J (if $P_{I/O}$ is neglected):

$$P_D = K \div (T_J + 273°C) \qquad (2)$$

Solving equations (1) and (2) for K gives:

$$K = P_D \cdot (T_A + 273°C) + \theta_{JA} \cdot P_D^2 \qquad (3)$$

where K is a constant pertaining to the particular part. K can be determined from equation (3) by measuring P_D (at equilibrium) for a known T_A. Using this value of K, the values of P_D and T_J can be obtained by solving equations (1) and (2) iteratively for any value of T_A.

10.4 DC ELECTRICAL CHARACTERISTICS

All specifications are valid under the following conditions: V_{DD} = 4.75 V to 5.25 V, V_{SS} = 0 V, T_A = T_L to T_H and 130 pF total capacitance on output pins.

Characteristic	Symbol	Min	Max	Unit
Input High Voltage (Except System Clock)	V_{IH}	2.0	V_{DD}	V
Input Low Voltge (Except System Clock)	V_{IL}	$V_{SS}-0.3$	0.8	V
Input High Voltage (System Clock)	V_{CIH}	2.4	V_{DD}	V
Input Low Voltage (System Clock)	V_{CIL}	$V_{SS}-0.3$	0.5	V
Input Leakage Current	I_{in}	—	20	μA
Input Capacitance	C_{in}	—	13	pF
Three-State Leakage Current (2.4/0.5 V)	I_{TSI}	—	20	μA
Open-Drain Leakage Current (2.4 V)	I_{OD}	—	20	μA
Output High Voltage (I_{OH} = 400 μA)	V_{OH}	2.4	—	V
Output Low Voltage (I_{OL} = 3.2 mA) A1–A31, FC0–FC3, $\overline{RTS}$, $\overline{TXD}$, $\overline{UDS}$/A0 as A0 (I_{OL} = 5.3 mA) D0–D15, $\overline{AS}$, $\overline{LDS}$, $\overline{UDS}$/A0 as $\overline{UDS}$, $\overline{DTACK}$, $\overline{BGACK}$, R/$\overline{W}$ (I_{OL} = 8.9 mA) $\overline{BR}$, $\overline{IRQ}$	V_{OL}	— — —	0.5 0.5 0.5	V
Power Dissipation @ 10 MHz, 0°C @ 12.5 MHz, 0°C @ 16.67 MHz, 0°C	P_D	— — —	0.50 0.55 0.65	W

10.5 AC ELECTRICAL CHARACTERISTICS

High and low outputs are measured at 2.0 V minimum and 0.8 V maximum, respectively. High and low inputs are driven to 2.4 V and 0.5 V, respectively, for AC test purposes. However, input specifications are still measured from 2.0 V to 0.8 V. All specifications are valid under the following conditions: V_{DD} = 4.75 V to 5.25 V, V_{SS} = 0 V, T_A = T_L to T_H, output load = 130 pF, and output current as specified in **DC ELECTRICAL CHARACTERISTICS**. See Figures 10-1–10-12.

Num.	Characteristic	10 MHz		12.5 MHz		16.67		Unit
		Min	Max	Min	Max	Min	Max	
1	Asynchronous Input Setup Time	20	—	20	—	10	—	ns
2	$\overline{UDS}$, $\overline{LDS}$ Inactive to $\overline{CS}$, $\overline{IACK}$ Inactive	—	100	—	80	—	60	ns
3	CLK Low (On Which $\overline{UDS}$ or $\overline{LDS}$ and $\overline{CS}$ or $\overline{IACK}$ are Recognized) to Data-Out Valid (see Note 5)	— —	1/2 +150	— —	1/2 +120	— —	1/2 +90	Clk. Per. ns
4	$\overline{CS}$ or $\overline{IACK}$ High to Data-Out High Impedance	—	60	—	50	—	35	ns
5	$\overline{LDS}$/$\overline{DS}$ High to Data-Out Hold Time (see Note 6)	0	—	0	—	0	—	ns
6	$\overline{IACK}$ or $\overline{CS}$ Low to $\overline{DTACK}$ High (Driving Three-State $\overline{DTACK}$ High)	—	80	—	70	—	60	ns
7	CLK Low (On Which $\overline{UDS}$ or $\overline{LDS}$ and $\overline{CS}$ or $\overline{IACK}$ are Recognized) to $\overline{DTACK}$ Low (see Note 5)	— —	2 +90	— —	2 +80	— —	2 +50	Clk. Per. ns
8	CLK Low to $\overline{DTACK}$ Low	—	90	—	80	—	50	ns
9	Data-Out Valid to $\overline{DTACK}$ Low	20	—	20	—	20	—	ns
10	$\overline{DTACK}$ Low to $\overline{UDS}$, $\overline{LDS}$, $\overline{CS}$, $\overline{IACK}$ High (Earliest)	100	—	80	—	60	—	ns

10.5 AC ELECTRICAL CHARACTERISTICS (Continued)

Num.	Characteristic	10 MHz		12.5 MHz		16.67 MHz		Unit
		Min	Max	Min	Max	Min	Max	
11	CS or IACK or Data Strobes (The Earliest) High to DTACK High (see Note 7)	—	60	—	50	—	40	ns
12	DTACK High to DTACK High Impedance (At End of Bus Cycle)	—	50	—	50	—	40	ns
13	UDS, LDS Inactive Time	1	—	1	—	1	—	Clk. Per.
14	CS, IACK Inactive Time	0	—	0	—	0	—	ns
15	A1–A2 Valid to UDS, LDS, CS (The Latest One) Low (Write)	30	—	20	—	20	—	ns
16	DTACK Low to Data and A1–A2 Hold Time	100	—	80	—	60	—	ns
17	UDS or LDS, CS or IACK (The Latest One) Low to Data-In Valid	—	80	—	70	—	60	ns
18	R/W Valid to UDS or LDS, CS or IACK (The Latest One) Low	20	—	20	—	10	—	ns
19	UDS, LDS High to R/W High	0	—	0	—	0	—	ns
20	CLK High to IRQ Low	—	100	—	80	—	60	ns
21	Reserved							
22	Reserved							
23	CLK High to BR Low	—	60	—	55	—	40	ns
24	CLK High to BR High Impedance	—	55	—	50	—	40	ns
25	BGACK Low to BR High Impedance	20	—	20	—	10	—	ns
26	BG Active/Inactive to CLK Low Setup Time	20	—	20	—	10	—	ns
27	CLK Low to BGACK Low	—	60	—	55	—	40	ns
28	CLK High to BGACK High Impedance	—	45	—	40	—	40	ns
29	AS and BGACK High (The Latest One) to BGACK Low (When BG is Previously Asserted)	2 +20	3 +80	2 +20	3 +70	2 +10	3 +50	Clk. Per. ns
30	BG Low to BGACK Low (No Other Bus Master)	2 +20	3 +80	2 +20	3 +70	2 +10	3 +50	Clk. Per. ns
31	BR High Impedance to BG High	0	—	0	—	0	—	ns
32	Clock on which BGACK Low to Clock on which AS Low	1.5	1.5	1.5	1.5	1.5	1.5	Clk. Per.
33	Clock Low to BGACK High	—	55	—	50	—	40	ns
34	CLK on which BR Low to CLK on which BGACK Low (Assuming that BG is Active and BGACK and AS are Inactive for at Least 2 CLK Periods)	1.5	1.5	1.5	1.5	1.5	1.5	Clk. Per.
35	CLK on which AS is High to CLK on which BGACK is High	—	1	—	1	—	1	Clk. Per.
36	CLK High to Address Valid	—	100	—	80	—	60	ns
37	CLK High to Address/FC High Impedance	—	70	—	60	—	50	ns
38	CLK High to FC Valid	—	60	—	55	—	50	ns
39	Address Valid to AS Valid	20	—	15	—	10	—	ns
40	CLK High to AS, UDS, LDS Low	—	50	—	40	—	40	ns
41	CLK to AS, UDS, LDS High	—	55	—	50	—	40	ns

10.5 AC ELECTRICAL CHARACTERISTICS (Continued)

Num.	Characteristic	10 MHz		12.5 MHz		16.67 MHz		Unit
		Min	Max	Min	Max	Min	Max	
42	$\overline{\text{AS}}$ High to Address/FC Invalid	20	—	10	—	0	—	ns
43	CLK High to $\overline{\text{AS}}$, $\overline{\text{UDS}}$, $\overline{\text{LDS}}$ High Impedance	—	70	—	60	—	45	ns
44	CLK to R/$\overline{\text{W}}$ High (see Note 4)	—	55	—	50	—	45	ns
45	CLK Low to R/$\overline{\text{W}}$ High Impedance	—	70	—	60	—	45	ns
46	$\overline{\text{UDS}}$, $\overline{\text{LDS}}$ High to Data-In Invalid	0	—	0	—	0	—	ns
47	$\overline{\text{AS}}$, $\overline{\text{UDS}}$, $\overline{\text{LDS}}$ High to $\overline{\text{DTACK}}$ High (Earliest of $\overline{\text{AS}}$, $\overline{\text{UDS}}$, or $\overline{\text{LDS}}$)	0	100	0	90	0	60	ns
48	Data-In to CLK Low Setup Time Required when $\overline{\text{DTACK}}$ Satisfies (1) (see Note 1)	10	—	10	—	5	—	ns
49	$\overline{\text{DTACK}}$ Low to Data-In Valid Required when $\overline{\text{DTACK}}$ does not Satisfy (1) (see Note 2)	—	65	—	50	—	40	ns
50	CLK High to R/$\overline{\text{W}}$ Low	—	60	—	55	—	40	ns
51	$\overline{\text{AS}}$ Low to Data-Out Valid (Write)	—	90	—	80	—	60	ns
52	CLK Low to Data-Out Valid	—	55	—	55	—	40	ns
53	Data-Out Valid to $\overline{\text{UDS}}$, $\overline{\text{LDS}}$ Low	20	—	15	—	10	—	ns
54	$\overline{\text{UDS}}$, $\overline{\text{LDS}}$ High to Data-Out Invalid	20	—	15	—	0	—	ns
55	CLK High to Data-Out Hold Time	0	100	0	100	0	60	ns
56	No Exception to $\overline{\text{BR}}$ ($\overline{\text{DTACK}}$ Active)	1.5 +20	2.5 +80	1.5 +20	2.5 +70	1.5 +10	2.5 +50	Clk. Per. ns
57	$\overline{\text{DTACK}}$ Low to Asynchronous Exception Active Required When $\overline{\text{DTACK}}$ Does Not Satisfy (1) (see Note 2)	—	55	—	35	—	30	ns
58	Exception Active to CLK Low Setup Time Synchronous Input ("Late Exception") Required when $\overline{\text{DTACK}}$ Satisfies (1) (see Note 1)	45	—	45	—	20	—	ns
59	Exception Active to CLK Low Setup Time Asynchronous Input (Required when $\overline{\text{DTACK}}$ is Absent) (see Note 3)	20	—	20	—	10	—	ns
60	$\overline{\text{AS}}$, $\overline{\text{UDS}}$, $\overline{\text{LDS}}$ High to Exception Inactive	0	—	0	—	0	—	ns
61	Exception Inactive to CLK Low Setup Time (for Identification of No Exception)	20	—	20	—	10	—	ns
62	No Exception to $\overline{\text{BR}}$ ($\overline{\text{DTACK}}$ Inactive)	2.5 +20	3.5 +80	2.5 +20	3.5 +70	2.5 +10	3.5 +50	Clk. Per. ns
63	$\overline{\text{RESET}}$ (on $\overline{\text{BEC0}}$–$\overline{\text{BEC2}}$) Width	10	—	10	—	10	—	Clk. Per.
64	CLK Frequency	4	10	4	12.5	4	16.67	MHz
65	CLK Period	100	250	80	250	60	250	ns
66	CLK Width High (see Note 8)	45	125	35	125	25	125	ns
67	CLK Rise/Fall Time (see Note 8)	—	10	—	5	—	5	ns
68	CLK Width Low (see Note 8)	45	125	35	125	25	125	ns
69	RCLK, TCLK Frequency	0	10	0	12.5	0	16.67	MHz
70	RxD to RCLK High Setup Time	35	—	35	—	25	—	ns
71	RCLK High to RxD Hold Time	5	—	5	—	5	—	ns

10.5 AC ELECTRICAL CHARACTERISTICS (Concluded)

Num.	Characteristic	10 MHz		12.5 MHz		16.67 MHz		Unit
		Min	Max	Min	Max	Min	Max	
72	RCLK, TCLK Rise/Fall Time	—	10	—	10	—	5	ns
73	RCLK, TCLK Width Low	45	—	35	—	25	—	ns
74	RCLK, TCLK Width High	45	—	35	—	25	—	ns
75	RCLK, TCLK Period	100	—	80	—	60	—	ns
76	TCLK Low to TxD Valid	10	80	10	60	10	45	ns
77	Reserved							
78	CD Low to RCLK Low Setup Time	25	—	25	—	25	—	ns
79	CTS Low to TCLK High Setup Time	25	—	25	—	25	—	ns

NOTES:

1. If $\overline{\text{DTACK}}$ satisfies the asynchronous setup time (1), then (48) is required for the data-in setup time and (58) for the synchronous exception setup time. Erroneous behavior may occur if (58) is not satisfied.
2. If $\overline{\text{DTACK}}$ does not satisfy (1), then (49) is required for data-in and (57) for the exception. Erroneous behavior may occur if (57) is not satisfied.
3. Active exception when $\overline{\text{DTACK}}$ is absent must satisfy the asynchronous setup time (59). Erroneous behavior may occur if (59) is not satisfied.
4. R/$\overline{\text{W}}$ rises on the end of a write cycle (i.e., on the phase following S7). If the XPC relinquishes the bus, then R/$\overline{\text{W}}$ is three-stated one phase later. When the XPC takes the bus, R/$\overline{\text{W}}$ is three-stated until S2 and changes on that phase.
5. Data (3) and $\overline{\text{DTACK}}$ (7) will be timed from the latest of $\overline{\text{CS}}$ and either data strobe during an MPU cycle. Data (3) and $\overline{\text{DTACK}}$ will be timed from the latest of $\overline{\text{IACK}}$ and either data strobe during an $\overline{\text{IACK}}$ cycle.
6. If $\overline{\text{CS}}$ or $\overline{\text{IACK}}$ is negated before $\overline{\text{UDS}}$/$\overline{\text{LDS}}$, the data bus will be three-stated (4), possibly before $\overline{\text{UDS}}$/$\overline{\text{LDS}}$ negation.
7. If an 8-bit bus is used, only $\overline{\text{LDS}}$ need be considered. If a 16-bit bus is used, both $\overline{\text{UDS}}$ and $\overline{\text{LDS}}$ must negate to apply to this specification.
8. The clock signal during test has 5 ns of rise time and 5 ns of fall time. For system implementations that have less clock rise and fall time, the clock pulse minimum should be commensurately wider such that:
   1. System $(\text{TCL}+(\text{TCR}+\text{TCF})\div 2)\geq(\text{minimum } t_{cyc})\div 2$
   2. System $(\text{TCH}+(\text{TCR}+\text{TCF})\div 2)\geq(\text{minimum } t_{cyc})\div 2$

   where:
   TCL is CLK width low (see electrical specification #68)
   TCH is CLK width high (see electrical specification #66)
   TCR is CLK rise time
   TCF is CLK fall time
   t_{cyc} is CLK period (see electrical specification #65)

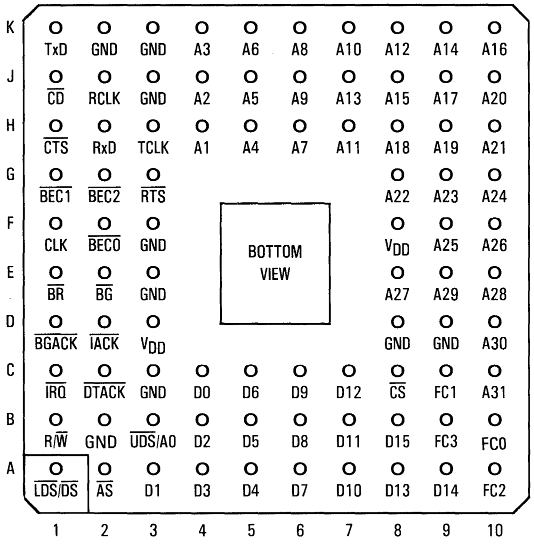
SECTION 11
ORDERING INFORMATION AND MECHANICAL DATA

This section contains the pin assignments and package dimensions for the PGA (pin grid array) for the MC68605RC. In addition, detailed information is provided to be used as a guide when ordering.

11.1 PACKAGE TYPES

Suffix	Package Type	Comments
R	Pin Grid Array (PGA) Ceramic	Depopulated Center Pins Solder Lead Finish No Standoffs
RC	Pin Grid Array (PGA) Ceramic	Depopulated Center Pins Gold Lead Finish No Standoffs
FN	Plastic Leaded Chip Carrier	Suitable for Socketing or Surface Mounting

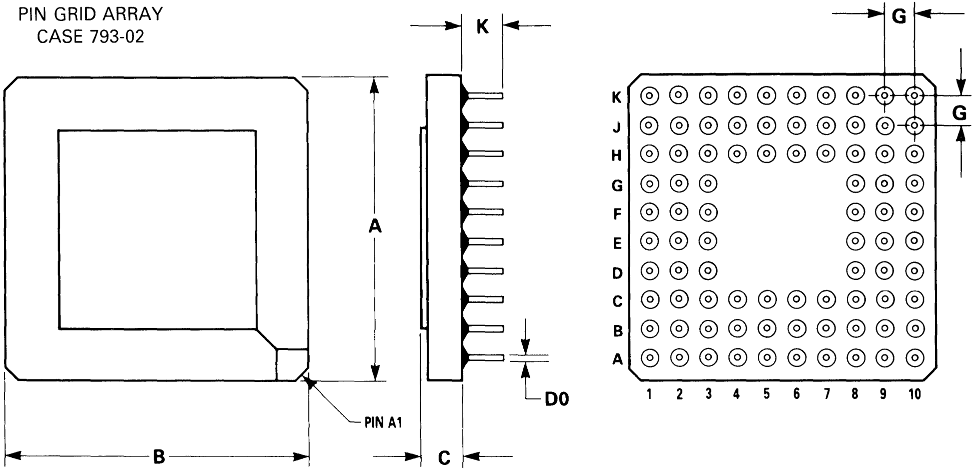
11.2 PIN ASSIGNMENTS



	1	2	3	4	5	6	7	8	9	10
K	TxD	GND	GND	A3	A6	A8	A10	A12	A14	A16
J	$\overline{CD}$	RCLK	GND	A2	A5	A9	A13	A15	A17	A20
H	$\overline{CTS}$	RxD	TCLK	A1	A4	A7	A11	A18	A19	A21
G	$\overline{BEC1}$	$\overline{BEC2}$	$\overline{RTS}$					A22	A23	A24
F	CLK	$\overline{BEC0}$	GND					V_{DD}	A25	A26
E	$\overline{BR}$	$\overline{BG}$	GND					A27	A29	A28
D	$\overline{BGACK}$	$\overline{IACK}$	V_{DD}					GND	GND	A30
C	$\overline{IRQ}$	$\overline{DTACK}$	GND	D0	D6	D9	D12	$\overline{CS}$	FC1	A31
B	R/$\overline{W}$	GND	$\overline{UDS}$/A0	D2	D5	D8	D11	D15	FC3	FC0
A	$\overline{LDS}$/$\overline{DS}$	$\overline{AS}$	D1	D3	D4	D7	D10	D13	D14	FC2

11.3 PACKAGE DIMENSIONS

RC SUFFIX (84 PIN)
PIN GRID ARRAY
CASE 793-02



NOTES:

1. DIMENSIONS A AND B ARE DATUMS AND T IS A DATUM SURFACE.
2. POSITIONAL TOLERANCE FOR LEADS: (84 PL)
   | ⌖ | ϕ 0.13 (0.005) Ⓛ | T | A Ⓢ | B Ⓢ |
3. DIMENSIONING AND TOLERANCING PER Y14.5M, 1982.
4. CONTROLLING DIMENSION: INCH.

	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	—	27.43	—	1.080
B	—	27.43	—	1.080
C	2.03	2.67	0.080	0.105
D	0.43	0.61	0.017	0.024
G	2.54 BSC		0.100 BSC	
K	3.56	4.95	0.140	0.195

STATE	I FRAME WITH POLL	I FRAME W/O POLL	RR WITH POLL	RR W/O POLL	REJ WITH POLL	REJ W/O POLL	RNR WITH POLL	RNR W/O POLL	SABM WITH OR W/O POLL	DISC. WITH OR W/O POLL	RR WITH FINAL	RR W/O FINAL	REJ WITH FINAL	REJ W/O FINAL	RNR WITH FINAL	RNR W/O FINAL	UA WITH OR W/O FINAL	DM WITH FINAL	DM W/O FINAL	FRMR WITH OR W/O FINAL	LOCAL START COMMAND	LOCAL STOP COMMAND	STATION BECOMES BUSY	BUSY CONDITION CLEAR	T1 EXPIRES	N2 × T1 IS EXCEEDED	NS SEQUENCE ERROR	INVALID NR RECEIVED	UNRECOGNIZED FRAME RECEIVED
S1 DISCONNECTED	DM, F = 1	—	DM, F = 1	—	DM, F = 1	—	DM, F = 1		UA, F = P TO S5	DM, F = P	—	—	—	—	—	—	—	SABM TO S2	SABM TO S2	—	SABM TO S2	DISC TO S4	X	—	X	X	*DM	*DM	*DM
S2 LINK SETUP	—	—	—	—	—	—	—		UA, F = P	DM, F = P TO S1	—	—	—	—	—	—	TO S5	TO S1	—	—	X	X	X	—	SABM	TO S1	—	—	—
S3 FRAME REJECT	FRMR, F = 1	FRMR, F = 0	FRMR, F = 1	FRMR, F = 0	FRMR, F = 1	FRMR, F = 0	FRMR, F = 1	FRMR, F 0	UA, F = P TO S5	UA, F = P TO S1	—	—	—	—	—	—	—	SABM TO S2	SABM TO S2	SABM TO S2	SABM TO S2	DISC TO S4	X	—	FRMR F = 0	SABM TO S2	*FR	*FR	*FR
S4 DISCONNECT REQUEST	—	—	—	—	—	—	—		DM, F = P TO S1	UA, F = P	—	—	—	—	—	—	TO S1	TO S1	—	—	X	X	X	—	DISC	TO S1	—	—	—
S5 INFORMATION TRANSFER	RR, F 1	**	RR, F = 1	**	RR, F = 1	**	RR, F = 1 TO S9	RR, F 0 TO S9	UA, F = P	UA, F = P TO S1	(UNXF) SABM TO S2	***	(UNXF) SABM TO S2	***	(UNXF) SABM TO S2	TO S9	SABM TO S2	SABM TO S2	SABM TO S2	SABM TO S2	SABM TO S2	DISC TO S4	RNR, F = P TO S8	X	RR, P = 1 TO S7	SABM TO S2	*J TO S6	FRMR(Z) TO S3	FRMR(W) TO S3
S6 REJ FRAME SENT	RR, F = 1 TO S5	** TO S5	RR, F = 1	**	RR, F = 1	**	RR, F = 1 TO S14	RR, F 0 TO S14	UA, F = P TO S5	UA, F P TO S1	(UNXF) SABM TO S2	***	(UNXF) SABM TO S2	***	(UNXF) SABM TO S2	TO S14	SABM TO S2	SABM TO S2	SABM TO S2	SABM TO S2	SABM TO S2	DISC TO S4	RNR, F = P TO S8	X	RR, P = 1 TO S7	SABM TO S2	IF P = 1 Tx RR, F = 1	FRMR(Z) TO S3	FRMR(W) TO S3
S7 WAITING I FRAME ACKNOWLEDGE- MENT	RR, F = 1	RR, F = 0	RR, F = 1	RR, F = 0	RR, F = 1	RR, F = 0	RR, F = 1 TO S12	RR, F 0 TO S12	UA, F = P TO S5	UA, F = P TO S1	*** TO S5	—	*** TO S5	—	TO S9	TO S12	SABM TO S2	SABM TO S2	SABM TO S2	SABM TO S2	SABM TO S2	DISC TO S4	RNR, F = P TO S11	X	RR, P = 1	SABM TO S2	*J	FRMR(Z) TO S3	FRMR(W) TO S3
S8 STATION BUSY	RNR, F = 1	RNR, F = 0	RNR, F = 1	*N	RNR, F = 1	*N	RNR, F = 1 TO S10	RNR, F 0 TO S10	UA, F = P	UA, F = P TO S1	(UNXF) SABM TO S2	***	(UNXF) SABM TO S2	***	(UNXF) SABM TO S2	TO S10	SABM TO S2	SABM TO S2	SABM TO S2	SABM TO S2	SABM TO S2	DISC TO S4	X	**J TO S5	RNR, P = 1 TO S11	SABM TO S2	RNR, F = P	FRMR(Z) TO S3	FRMR(W) TO S3
S9 REMOTE STATION BUSY	RR, F = 1	RR, F = 0	RR, F = 1 TO S5	** TO S5	RR, F = 1 TO S5	** TO S5	RR, F = 1	RR, F 0	UA, F = P TO S5	UA, F = P TO S1	(UNXF) SABM TO S2	*** TO S5	(UNXF) SABM TO S2	*** TO S5	(UNXF) SABM TO S2	—	SABM TO S2	SABM TO S2	SABM TO S2	SABM TO S2	SABM TO S2	DISC TO S4	RNR, F = P TO S10	X	RR, P = 1 TO S12	SABM TO S2	*J TO S14	FRMR(Z) TO S3	FRMR(W) TO S3
S10 BOTH STATIONS BUSY	RNR, F = 1	RNR, F = 0	RNR, F = 1 TO S8	*N TO S8	RNR, F = 1 TO S8	*N TO S8	RNR, F = 1	RNR, F 0	UA, F = P TO S8	UA, F = P TO S1	(UNXF) SABM TO S2	*** TO S8	(UNXF) SABM TO S2	*** TO S8	(UNXF) SABM TO S2	—	SABM TO S2	SABM TO S2	SABM TO S2	SABM TO S2	SABM TO S2	DISC TO S4	X	**J TO S9	RNR, P = 1 TO S13	SABM TO S2	RNR, F = P	FRMR(Z) TO S3	FRMR(W) TO S3
S11 WAITING I FRAME ACKNOWLEDGE- MENT AND STA- TION BUSY	RNR, F = 1	RNR, F = 0	RNR, F = 1	RNR, F = 0	RNR, F = 1	RNR, F = 0	RNR, F = 1 TO S13	RNR, F 0 TO S13	UA, F = P TO S8	UA, F = P TO S1	*** TO S8	—	*** TO S8	—	TO S10	TO S13	SABM TO S2	SABM TO S2	SABM TO S2	SABM TO S2	SABM TO S2	DISC TO S4	X	**J TO S7	RNR, P = 1	SABM TO S2	RNR, F = P	FRMR(Z) TO S3	FRMR(W) TO S3
S12 WAITING I FRAME ACKNOWLEDGE- MENT AND RE- MOTE STATION BUSY	RR, F = 1	RR, F = 0	RR, F = 1 TO S7	RR, F = 0 TO S7	RR, F = 1 TO S7	RR, F = 0 TO S7	RR, F = 1	RR, F 0	UA, F = P TO S5	UA, F = P TO S1	*** TO S5	— TO S7	*** TO S5	— TO S7	— TO S9	—	SABM TO S2	SABM TO S2	SABM TO S2	SABM TO S2	SABM TO S2	DISC TO S4	RNR, F = P TO S13	X	RR, P = 1	SABM TO S2	*J	FRMR(Z) TO S3	FRMR(W) TO S3
S13 WAITING I FRAME ACKNOWLEDGE- MENT AND BOTH STATIONS BUSY	RNR, F = 1	RNR, F = 0	RNR, F = 1 TO S11	RNR, F = 0 TO S11	RNR, F = 1 TO S11	RNR, F = 0 TO S11	RNR, F = 1	RNR, F 0	UA, F = P TO S8	UA, F = P TO S1	*** TO S8	— TO S11	*** TO S8	— TO S11	— TO S10	—	SABM TO S2	SABM TO S2	SABM TO S2	SABM TO S2	SABM TO S2	DISC TO S4	X	**J TO S12	RNR, P = 1	SABM TO S2	RNR, F = P	FRMR(Z) TO S3	FRMR(W) TO S3
S14 REJ FRAME SENT AND REMOTE STATION BUSY	RR, F = 1 TO S9	RR, F = 0 TO S9	RR, F = 1 TO S6	** TO S6	RR, F = 1 TO S6	** TO S6	RR, F = 1	RR, F 0	UA, F = P TO S5	UA, F = P TO S1	(UNXF) SABM TO S2	*** TO S6	(UNXF) SABM TO S2	*** TO S6	(UNXF) SABM TO S2	—	SABM TO S2	SABM TO S2	SABM TO S2	SABM TO S2	SABM TO S2	DISC TO S4	RNR, F = P TO S10	X	RR, P = 1 TO S12	SABM TO S2	IF P = 1 Tx RR, F = 1	FRMR(Z) TO S3	FRMR(W) TO S3

** If I available then Tx I frame else Tx RR, F = 0
*** If I available then Tx I frame else do nothing
*N If I available then Tx I frame else Tx RNR, F = 0
**N If P = 1 then Tx RNR, F = 1 else if I available then Tx I frame else Tx RNR, F = 0
*FR If P = 1 then Tx FRMR, F = 1 else if P = 0 then Tx FRMR, F = 0 else do nothing
*DM If P = 1 then Tx DM, F = 1 else do nothing
*J If no REJ FRAME is outstanding then transmit REJ, F = P else if P = 1 then Tx RR, F 1 else do nothing
**J If the I field of a correctly received frame has been discarded (due to the busy condition) then Tx REJ, F = 0 else Tx RR, F = 0
— Do nothing
X This event never occurs in this state
UNXF Unexpected final bit

Figure 2-1. XPC State Diagram

Figure 2-1. XPC State Diagram

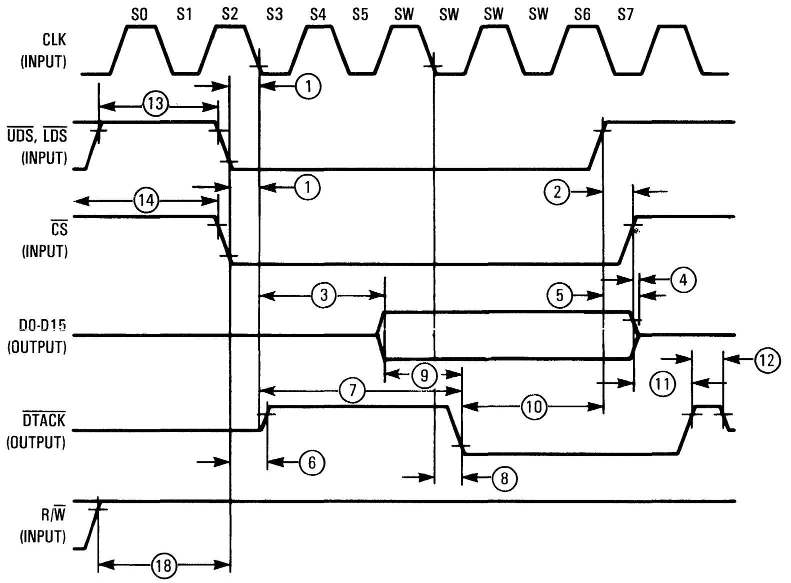


Figure 10-1. Host Processor Read Cycle Timing Diagram

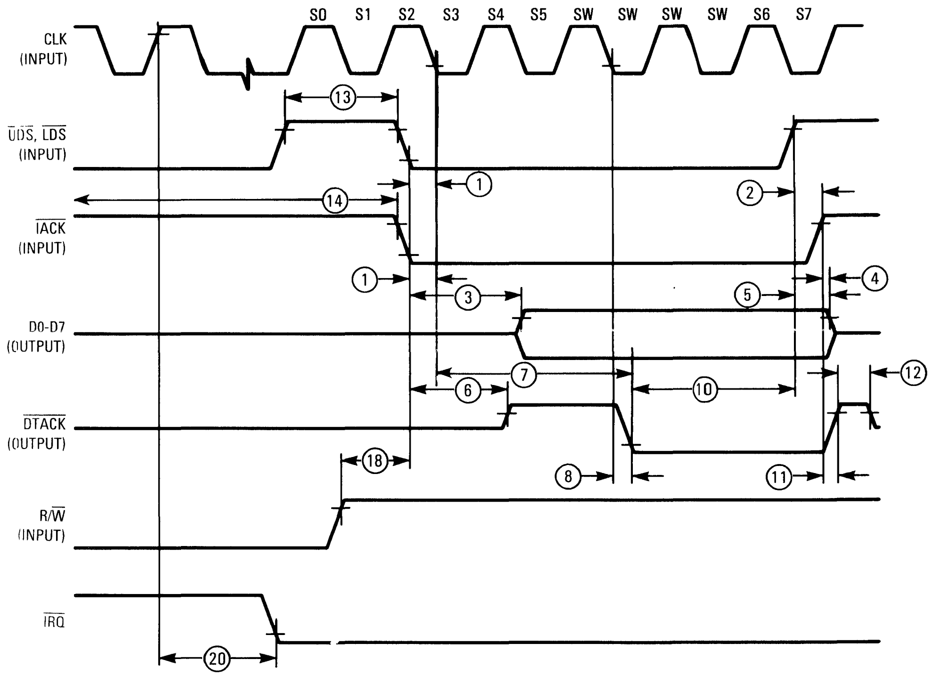


Figure 10-3. Interrupt Acknowledge Cycle Timing Diagram

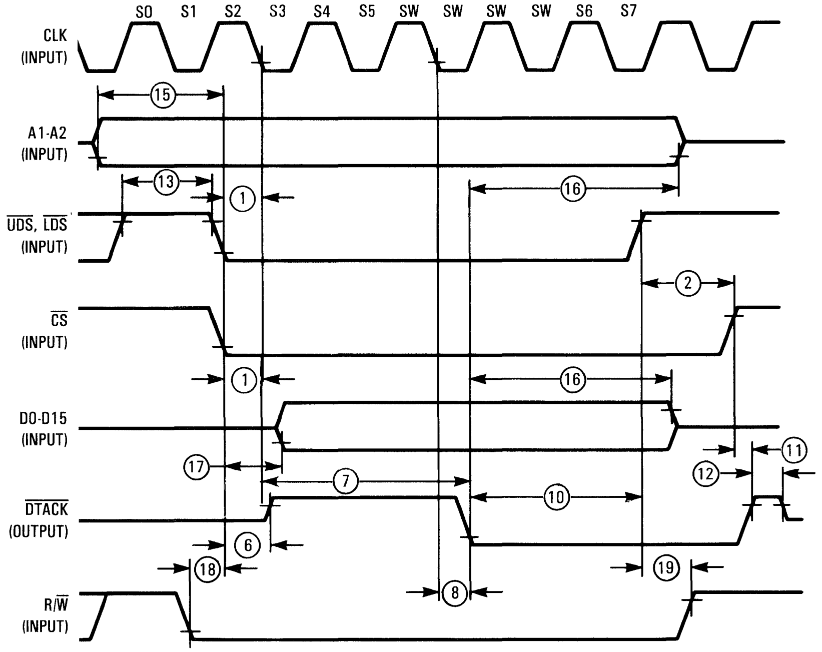


Figure 10-2. Host Processor Write Cycle Timing Diagram

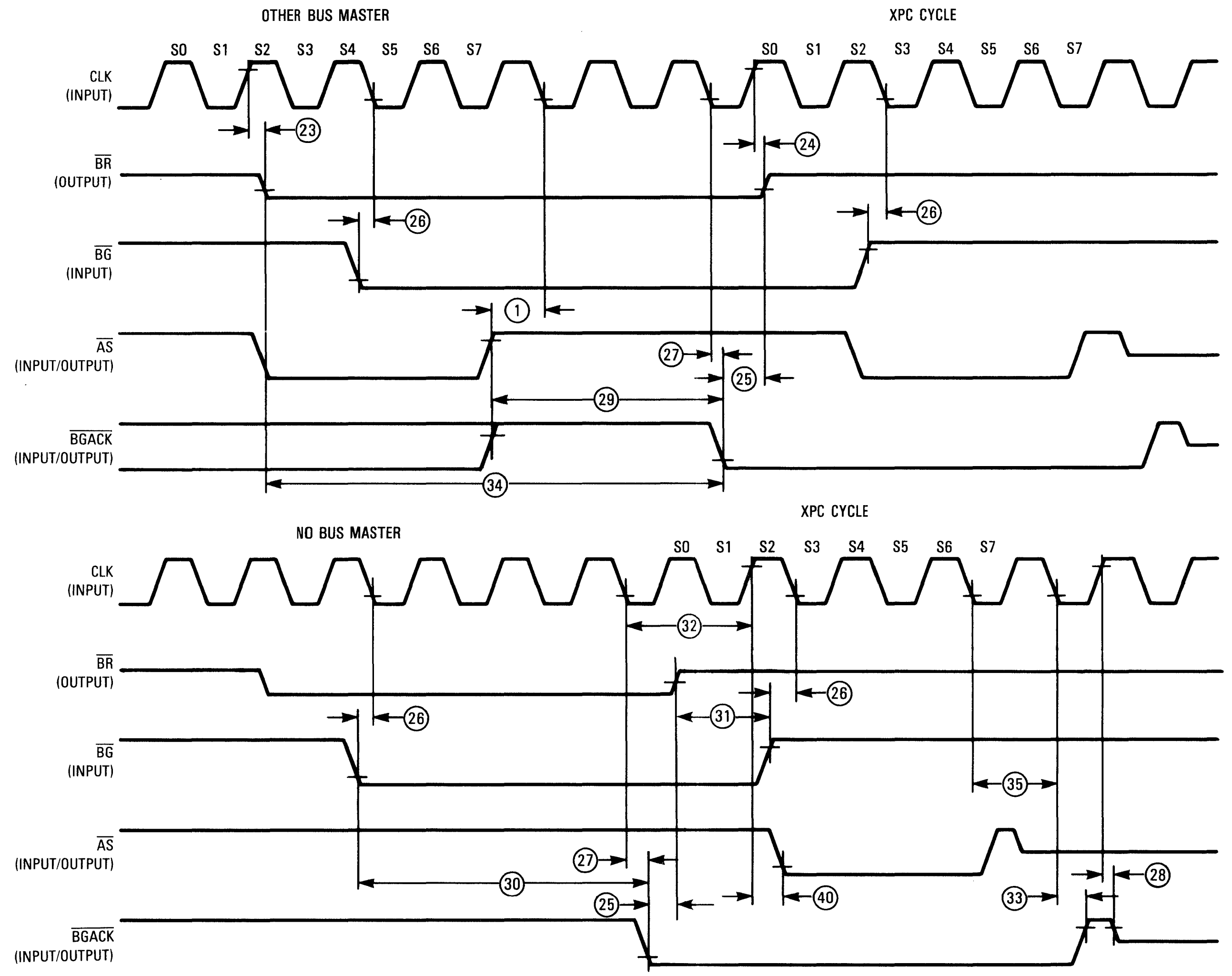
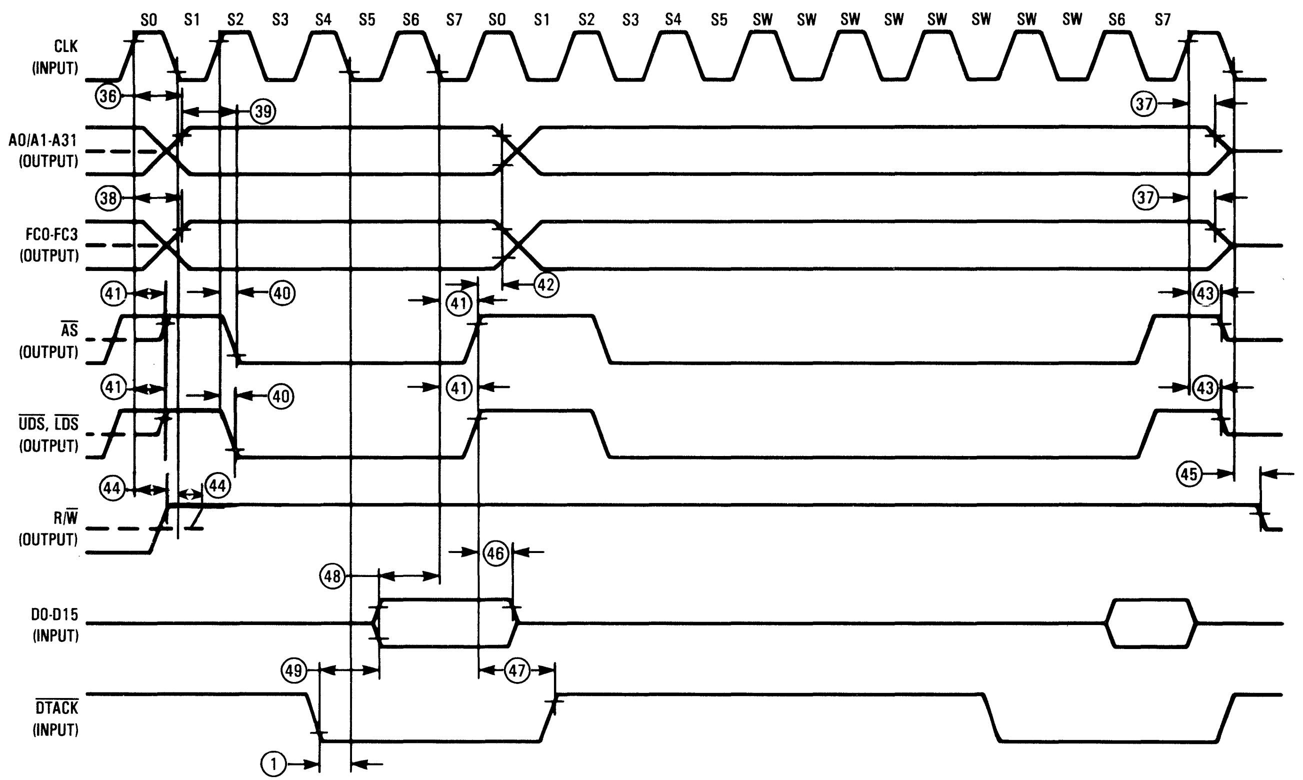

OTHER BUS MASTER
XPC CYCLE
S0 S1 S2 S3 S4 S5 S6 S7
S0 S1 S2 S3 S4 S5 S6 S7
CLK (INPUT)
$\overline{\text{BR}}$ (OUTPUT)
$\overline{\text{BG}}$ (INPUT)
$\overline{\text{AS}}$ (INPUT/OUTPUT)
$\overline{\text{BGACK}}$ (INPUT/OUTPUT)
23
24
26
26
1
27
25
29
34
NO BUS MASTER
XPC CYCLE
S0 S1 S2 S3 S4 S5 S6 S7
CLK (INPUT)
$\overline{\text{BR}}$ (OUTPUT)
$\overline{\text{BG}}$ (INPUT)
$\overline{\text{AS}}$ (INPUT/OUTPUT)
$\overline{\text{BGACK}}$ (INPUT/OUTPUT)
32
31
26
26
35
30
27
25
40
33
28

Figure 10-4. Bus Arbitration Timing Diagram


S0 S1 S2 S3 S4 S5 S6 S7 S0 S1 S2 S3 S4 S5 SW SW SW SW SW SW SW SW S6 S7
CLK (INPUT)
A0/A1-A31 (OUTPUT)
FC0-FC3 (OUTPUT)
$\overline{\text{AS}}$ (OUTPUT)
$\overline{\text{UDS}}$, $\overline{\text{LDS}}$ (OUTPUT)
R/$\overline{\text{W}}$ (OUTPUT)
D0-D15 (INPUT)
$\overline{\text{DTACK}}$ (INPUT)
36
39
37
38
37
41
40
41
42
43
41
40
41
43
44
44
45
48
46
49
47
1

NOTE:
The solid lines assume that the communication controller was bus master on the last cycle. The dotted lines assume that there was a different bus master.

Figure 10-5. Read Cycle and Slow Read Cycle Timing Diagram

Figure 10-4. Bus Arbitration Timing Diagram

Figure 10-5. Read Cycle and Slow Read Cycle Timing Diagram

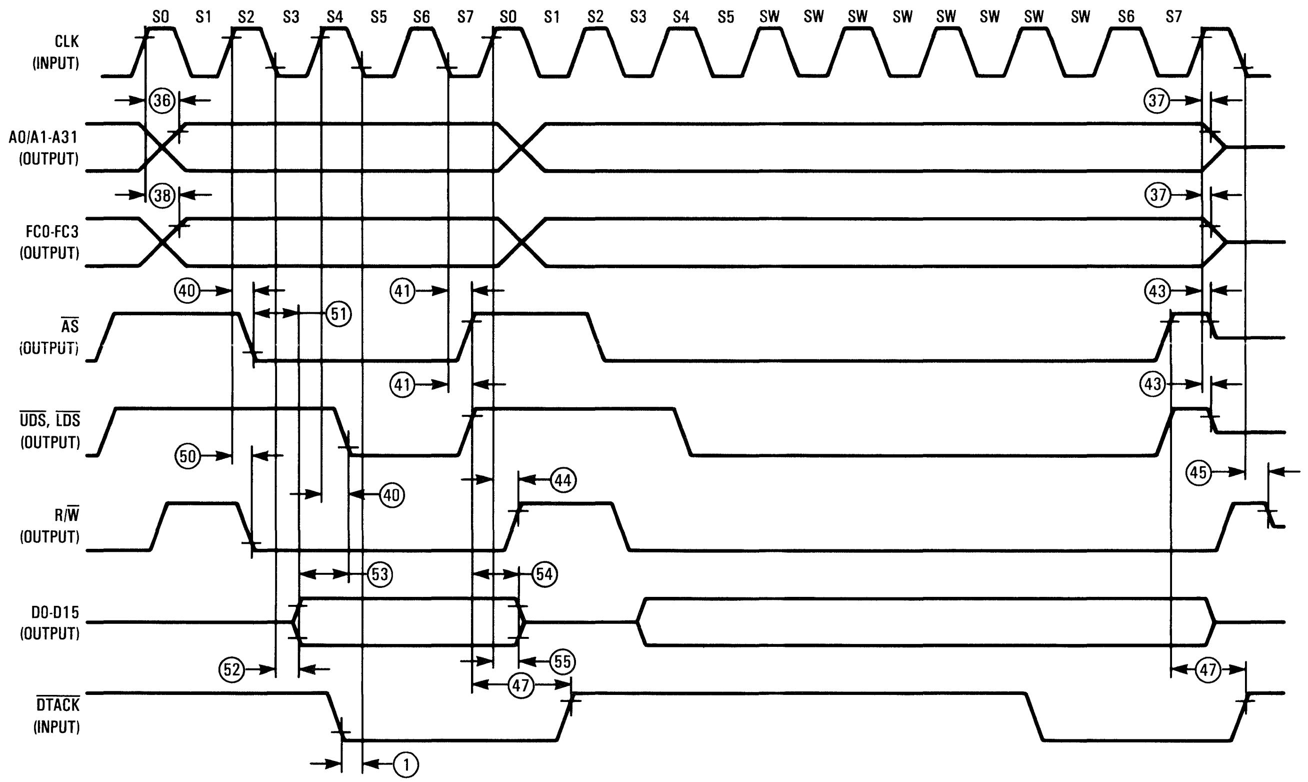
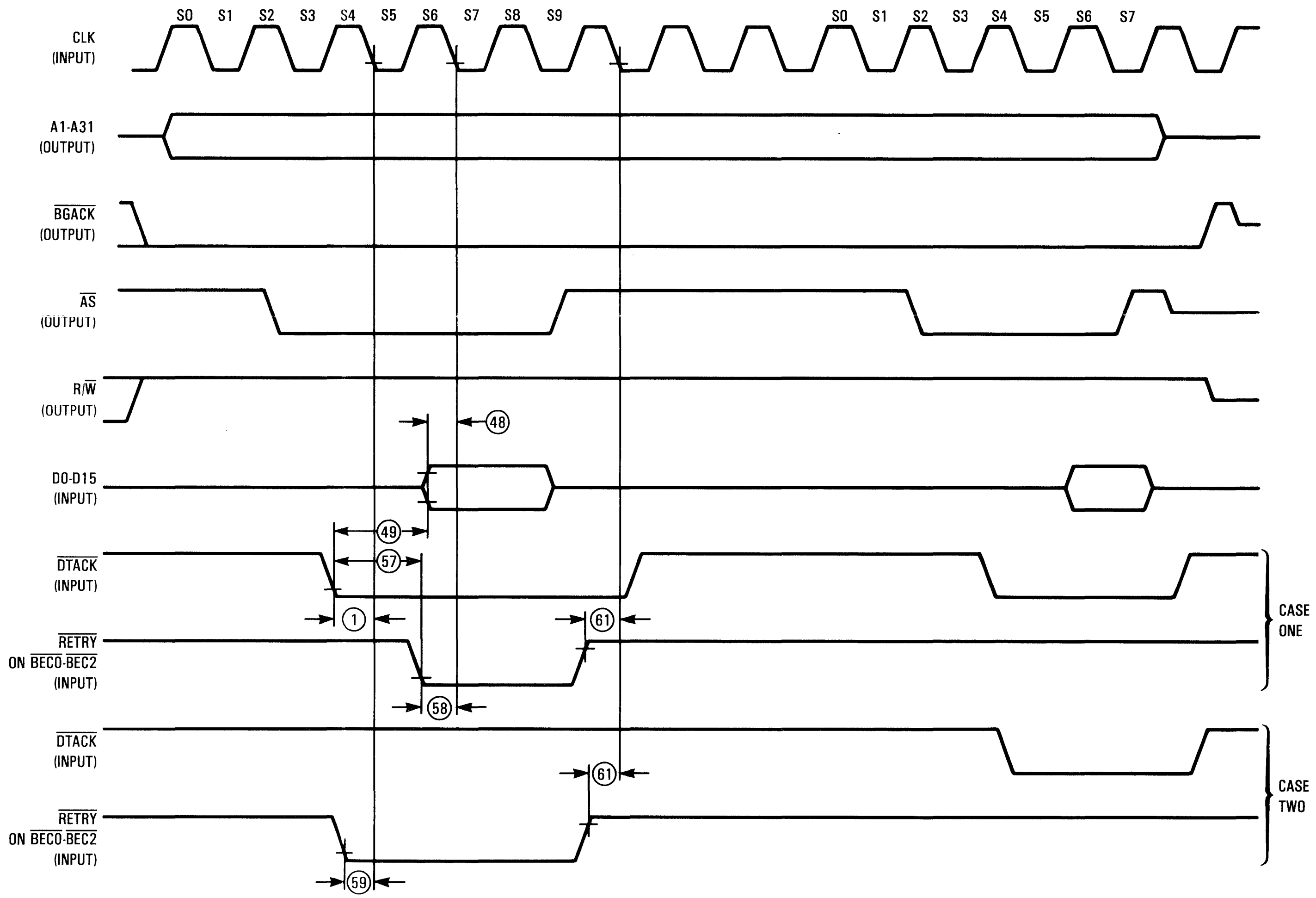


Figure 10-6. Write Cycle Timing Diagram

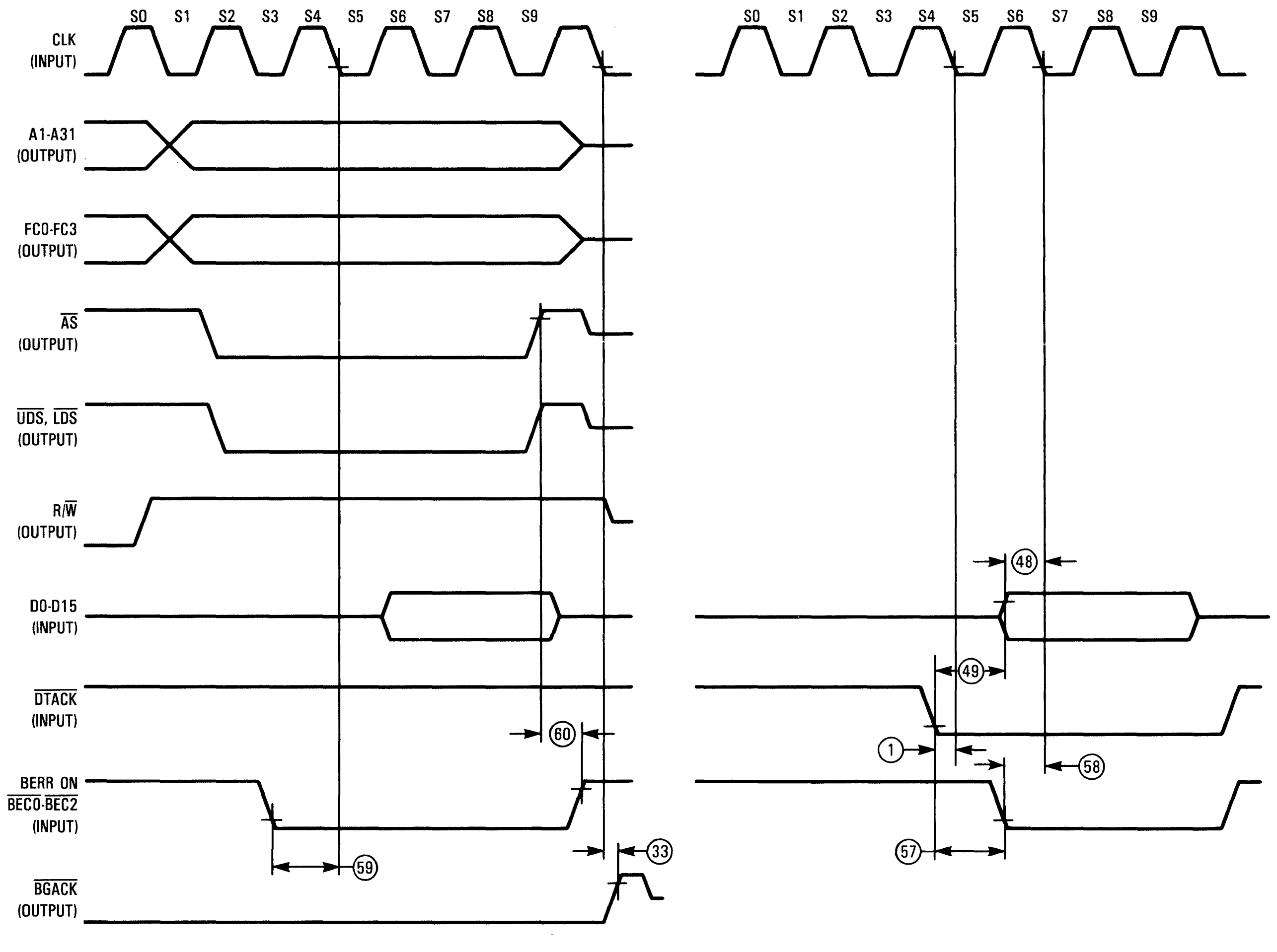


CASE 1: If DTACK satisfies (1), then (48) and (58) are required; if DTACK is active but does not satisfy (1), then (49) and (57) are required.
CASE 2: If DTACK is not active, then (59) is required for the exception active setup time. Parameter (61) is always required for the exception inactive setup time.

Figure 10-7. XPC Read Cycle with Retry Timing Diagram

Figure 10-6. Write Cycle Timing Diagram

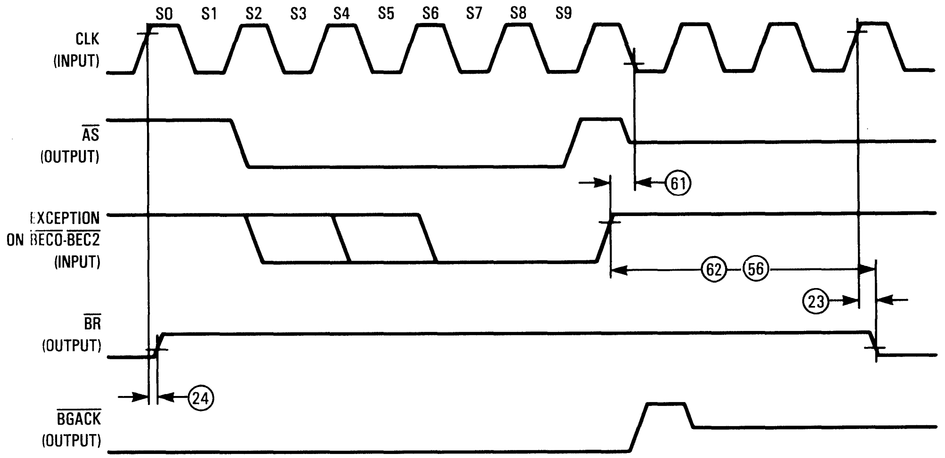
Figure 10-7. XPC Read Cycle with Retry Timing Diagram



EARLY ASYNCHRONOUS EXCEPTION
$\overline{\text{DTACK}}$ INACTIVE

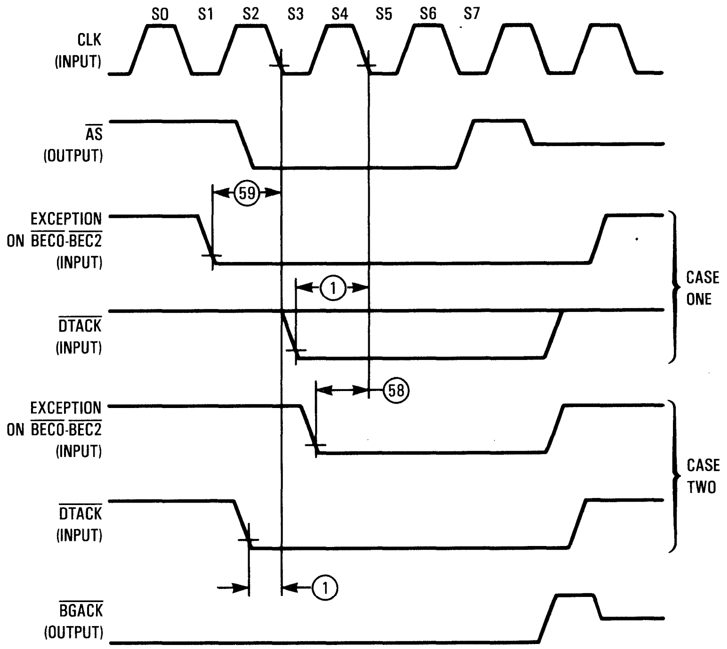
LATE SYNCHRONOUS EXCEPTION
$\overline{\text{DTACK}}$ ACTIVE

Figure 10-8. Read Cycle with Bus Error Timing Diagram



The above occurs when the XPC requires the bus cycle after a previous exception.

Figure 10-9. $\overline{\text{BR}}$ After Previous Exception Timing Diagram



Two alternatives of $\overline{\text{DTACK}}$ and exception. Case one has $\overline{\text{DTACK}}$ occur after exception and case two has exception occur after $\overline{\text{DTACK}}$. Note that a HALT cycle can be terminated only by $\overline{\text{DTACK}}$.

Figure 10-10. Short Exception Cycle Timing Diagram

Figure 10-8. Read Cycle with Bus Error Timing Diagram

Figure 10-9. $\overline{\text{BR}}$ After Previous Exception Timing Diagram

Figure 10-10. Short Exception Cycle Timing Diagram

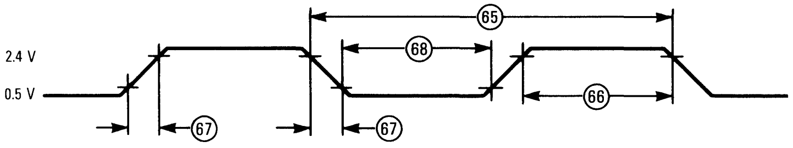


Figure 10-11. Clock (CLK) Timing Diagram

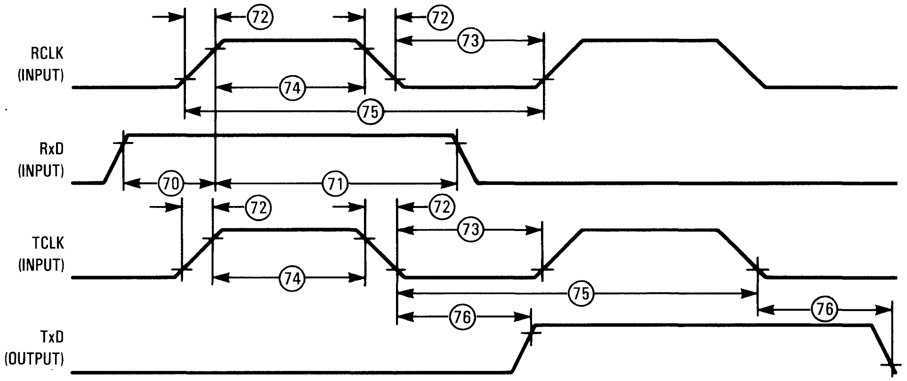


Figure 10-12. XPC Serial Data RxD, TxD, and Serial Clocks (RCLK, TCLK) Timing Diagram

Figure 10-11. Clock (CLK) Timing Diagram

Figure 10-12. XPC Serial Data RxD, TxD, and Serial Clocks (RCLK, TCLK) Timing Diagram

MOTOROLA

Literature Distribution Centers:

USA: Motorola Literature Distribution; P.O. Box 20912; Phoenix, Arizona 85036.
EUROPE: Motorola Ltd.; European Literature Center; 88 Tanners Drive, Blakelands, Milton Keynes, MK14 5BP, England.
ASIA PACIFIC: Motorola Semiconductors H.K. Ltd.; P.O. Box 80300; Cheung Sha Wan Post Office; Kowloon Hong Kong.
JAPAN: Nippon Motorola Ltd.; 3-20-1 Minamiazabu, Minato-ku, Tokyo 106 Japan.