EK-DUP11-MM-003

DUP11
Bit Synchronous Interface Maintenance Manual

digital

EK-DUP11-MM-003

DUP11
Bit Synchronous Interface Maintenance Manual

Prepared by Educational Services
of
Digital Equipment Corporation

Preliminary Edition, December 1975
2nd Edition (Rev), December 1976
3rd Edition (Rev), August 1982

Printed in U.S.A.

This document was set on DIGITAL's DECset-8000 computerized typesetting system.

EDUCATIONAL SERVICES
DEVELOPMENT AND PUBLISHING
UPDATE NOTICE

DUP11 BIT SYNCHRONOUS INTERFACE MAINTENANCE MANUAL
EK-DUP11-MM-CN1
OCTOBER 1983

Insert this Update Notice Page in the manual, directly following the Title Page, as a means of maintaining an up-to-date record of changes to the manual.

INSTRUCTIONS

The following appendix is to be placed as the final appendix in your manual.

CONTENTS

CONTENTS (Cont)

CONTENTS (Cont)

CONTENTS (Cont)

FIGURES

FIGURES (Cont)

TABLES

CHAPTER 1
INTRODUCTION

1.1 SCOPE

This manual provides the user with the information necessary to install, operate and maintain the DUP11 Synchronous Line Interface. The manual is organized into five chapters and three appendices:

Chapter 1 - Introduction
Chapter 2 - Installation
Chapter 3 - Register Description
Chapter 4 - Theory of Operation
Chapter 5 - Maintenance
Appendix A - PDP-11 Memory Organization and Addressing Conventions
Appendix B - Logic Symbology
Appendix C - Integrated Circuit Descriptions

This chapter provides a general description of the DUP11 and a general discussion of the Synchronous Data Link Control (SDLC) protocol and Digital Data Communications Message Protocol (DDCMP). Some background material on Cyclic Redundancy Checking (CRC) methods is provided also.

1.2 DUP11 GENERAL DESCRIPTION

The DUP11 provides a data path between a synchronous modem and the Unibus. It operates under the discipline of SDLC, ADCCP, DDCMP, and other similar protocols. Protocols of the BISYNC family can be used with some loss of efficiency due to the additional software decisions required.

The DUP11 provides parallel-to-serial conversion of data to be transmitted and serial-to-parallel conversion of received data. Logic is provided to create a transparent data stream and to compute a CRC check character during transmission. All information is handled in 8-bit bytes and VRC parity is not provided. CRC error detection is provided during reception. Modem control and level conversion logic is provided also.

Interrupt control logic is used to generate requests for the transfer of data between the DUP11 and the PDP-11 system memory via the Unibus. No direct memory access (DMA) logic is contained in the DUP11.

The DUP11 contains logic to perform the following functions:

1. Program control of secondary station address recognition. Primary station operation is used as the default condition (SDLC protocol family only).

2. Programmable SYN character recognition (DDCMP and BISYNC protocol families).

3. CRC characters computation and error detection (SDLC and DDCMP protocol families).

4. Automatic transmission of flag characters initiated by the program (SDLC protocol family only).

5. Program control of transmission of abort sequence and 16 zero sequence (SDLC protocol family only).

6. Hardware detection of received flags and abort sequences (SDLC protocol family only).

The DUP11, including level conversion, is contained on a hex module. The DUP11 is connected to the modem via a BC05C cable and BC02 cable that support RS232-C specifications only. Current mode operation is not supported by the DUP11 and it is not compatible with the DF11 series options.

The modem control logic is compatible with Bell 201, 208, and 209 series modems. There is no interlock between the transfer of data and modem control. The program controls handshaking with the modem, if it is required. Once the handshaking has been completed, the program can initiate the transfer of data. The modem control logic includes secondary receive and transmit leads. These leads can be redefined by the Field Service engineer at the user's request.

1.3 SDLC AND DDCMP PROTOCOLS

1.3.1 Introduction

This discussion provides a general description of the SDLC and DDCMP protocols. It is the prerequisite to a thorough understanding of the operation of the DUP11. Details of the SDLC, DDCMP, ADCCP and BISYNC protocols are found in the following documents:

Digital Data Communications Message Protocol (Digital Equipment 130-959-007-02)
IBM Binary Synchronous Communications General Information (GA27-3004-2)
ADCCP ANSI X3S34/475 DR7
ADCCP ANSI X3S34/584 DR1

1.3.2 General Information

Although the mentioned protocols are not identical, they are similar enough to operate with the DUP11. The program directly controls the DUP11 operation through the use of control and status registers. The program must provide a continuous flow of data to be transmitted. No intra-message fill characters are allowed. The program must also service the receiver data buffer within the prescribed time.

When transmitting in the SDLC or DDCMP family of protocols, the program must form the address and command fields plus any other header information that is required. The program must maintain the transmitter data buffer and set marker bits to delimit the transmitted message. When receiving in the SDLC or DDCMP family of protocols, the program must interpret the header information, service the receiver data buffer, and monitor the status bits associated with the received data.

Protocols such as BISYNC that achieve transparency by using special control characters are less efficient than SDLC and DDCMP when used with the DUP11. This occurs because of the increased program involvement required to maintain transparency and compute the CRC character. The CRC control logic in the DUP11 is not suited to protocols in which special control characters appear within the body of the message. For these protocols, the CRC logic should be disabled by setting the NO CRC bit (PARCSR bit 9).

1.3.3 SDLC Protocol Description

1.3.3.1 Message Format – The SDLC message format is shown in Figure 1-1. This format is called a frame and is the standard structure for all transmissions.

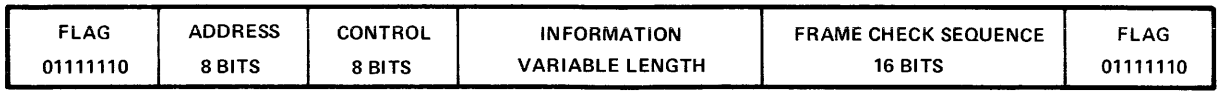


Figure 1-1 SDLC Message Format

The frame starts with the 8-bit Flag sequence, 01111110, followed in order by the Address sequence, Control sequence, Information sequence (if present), Frame Check sequence, and ends with another Flag sequence. In some applications, the Flag is preceded by a sequence of 16 zeros.

Each sequence in the frame is discussed below with emphasis on related operational features of the DUP11, if applicable.

Flag Character
The flag character is a unique 8-bit character of the form 01111110. Flag characters are used to delimit the message. They can be used to fill in between messages but cannot be used as fillers within messages. When the transmitter initiates the start of a message by asserting the TSOM (transmitter start of message) bit, the initial flag character is automatically transmitted. If the TSOM bit is still asserted at the end of the first flag character, another flag character is transmitted. When the TXDONE (transmitter done bit) is asserted by the DUP11 subsequent to the program's asserting of the TSOM bit, the program may respond by loading data into the TXDBUF (transmitter data buffer) low byte, or leave the TSOM bit asserted and send another flag.

In some applications, the TSOM and TEOM bits are used to initiate a sequence of 16 zeros. This sequence can be initiated only from the idle state. To transmit this sequence, SEND must be asserted and TXACT must be cleared. With these requirements met, the program simultaneously sets TSOM and TEOM and the 16 zeros are transmitted. When the first zero bit is presented to the serial output, TXDONE is set. Now, the program should clear TEOM and on the next transition of TXDONE the program should clear TSOM. The first data character can be loaded now. This point marks the start of the initial flag character. the first data character is transmitted subsequent to the current flag character.

When the last character of a message has been loaded into the TXDBUF, that character is then transmitted. Subsequent to loading the last character, the TXDONE bit is asserted again by the DUP11. This marks the start of the transmission of the last character. At this time, the TEOM (transmitter end of message) may be asserted in the upper byte of the TXDBUF. The character currently being serialized (i.e., the last character of the message) is transmitted and followed by a CRC check character and the terminating flag character. This concludes the message.

When the receiver logic is enabled by the software, it searches for flag characters. If the basic SDLC or ADCCP message format is followed and the receiver is programmed to operate in the secondary mode, the following actions occur.

The eight bits after the last received flag are compared to the secondary station address. If a match is not found, the receiver continues to hunt for a flag. When the next flag character is located, this comparison of addresses is reiterated.

If the character subsequent to the flag character matches the secondary station address, characters received subsequent to the address character cause the RXDONE (receiver done) bit to be asserted. The RSOM (receiver start of message) bit is presented to the program along with the first data character.

When the secondary station receiver is actively transferring data, the following events occur when a terminating flag character is detected. The receiver logic automatically resumes the address search as cited earlier. Also, a status entry is made into the receiver data buffer, the REOM bit is asserted and the CRC error bit is set if an error was detected. The lower byte of data in this entry is invalid.

When the receiver logic is programmed to operate as a primary station, all characters subsequent to the last received flag character cause the RXDONE bit to be asserted. The first character of the frame is accompanied by the RSOM bit.

When the terminating flag character of a message is received, primary station operation is the same as cited above for secondary station operation. When the next data character is received, the receiver logic again sets the RSOM and RXDONE bits. The last two bytes preceding the flag were the receiver CRC bytes.

Address Character
The address character appears subsequent to the flag character and is eight bits long. This format supports a maximum of 256 addresses. The protocol has provisions for the recursive expansion of the number of addresses. This feature is not supported by the DUP11 hardware. It must be maintained by the program. In the secondary station mode, the program must load the address of the receiving station into the low byte of the PARCSR.

Control Field
The 8-bit control field follows the address character. This field is controlled by the program and is encoded to indicate the commands and responses to control the data link. This field has three formats as described below.

1. Nonsequenced Format – used by the primary station primarily for data link management. Such duties include activating and initializing secondary stations, controlling the response mode of secondary stations, and reporting procedural errors.
2. Supervisory Format – does not contain an information field but it is an adjunct to the information format. It is used by the primary station to poll the secondary stations. The secondary stations use this format to provide acknowledgment to the primary station.
3. Information Format – used by primary and secondary stations for the transfer of information fields.

Information Field
This field is used for the transmission of data or status information. This field contains an arbitrary number of characters as specified by the documents covering the protocols. **The DUP11 handles the data in this field as eight bit characters.**

When one character is transmitted from the transmitter shift register, another character is taken from the data buffer. If the data buffer is empty, the transmitted data lead goes to a mark hold state. Also, a status bit is asserted to indicate the data underrun condition in SDLC or ADCCP and an Abort character is automatically transmitted.

There are no restrictions on bit patterns that appear between flags in an SDLC frame. Therefore, the transmitted data may contain six or more contiguous 1s and this pattern could be interpreted as a flag which would inadvertently terminate an incomplete frame. To prevent this action and to maintain data

transparency, the DUP11 contains 0 insertion and 0 removal logic that is active on all characters between the flags. During transmission, when five contiguous 1s occur, the transmitter automatically inserts a 0 after the fifth 1. During reception, the 0 after five contiguous 1s is automatically removed. This applies to all fields except the Flag.

Frame Check Sequence (FCS) Field
This 16-bit field follows the information field and is also referred to as the Block Check Character (BCC) or CRC check character. It is used in all SDLC frames to detect errors.

Logic to compute CRC check characters is included in the transmitter logic. Similarly, logic is included in the receiver logic to check the results when the check character is received. This operation of computing and verifying the CRC check is transparent to the program. Any error in the computation of the received check character in SDLC type protocol operation is indicated by a status bit in the receiver data buffer. If DDCMP operation is selected, the program must monitor a status bit to detect the desired accumulated results.

Two CRC polynomials are supported by the DUP11: CRC 16 and CCITT. When the SDLC or ADCCP mode of operation is selected, the CCITT polynomial is used and the internal CRC registers are effectively initialized to all 1s. During transmission, the complement of the accumulated CRC character is sent. If the SDLC or ADCCP mode is not selected, the CRC 16 polynomial is used providing CRC checking is not inhibited.

1.3.3.2 Abort Sequence – An abort is the premature termination of a data line by the transmitting station. An abort is detected by the reception of more than seven contiguous 1s. When the abort sequence is received, the message in progress is terminated. A flag (RXERROR) is set and RXDONE is set also. If the program has set RXITEN, an a interrupt request is generated when RXDONE is set. A transmitting station can send abort sequences under program control by setting the TXABORT bit. If the program response time to the TXDONE bit is excessive, the TXDATLAT bit is set and the transmitter idles abort characters.

1.3.4 DDCMP Protocol
DDCMP (Digital Data Communications Message Protocol) was developed to provide full-duplex message transfer over standard existing hardware.

1.3.4.1 Controlling Data Transfers – The DDCMP message format is shown in Figure 1-2. A single control character is used in a DDCMP message, and is the first character in the message. Three control characters are provided in DDCMP to differentiate between the three possible types of messages:

SOH – data message follows
ENQ – control message follows
DLE – bootstrap message follows.

Note that the use of a fixed-length header and message-size declaration obviates the BSC requirement for extensive message and header delimiter codes.

Figure 1-3 shows a simple example of data exchange between the DUP11/PDP-11 and a data terminal. More efficient procedures can be derived after a study of DDCMP.

SYN	SYN	SOH	COUNT 14 BITS	FLAG 2 BITS	RESPONSE 8 BITS	SEQUENCE 8 BITS	ADDRESS 8 BITS	CRC-1 16 BITS	DATA (ANY NUMBER OF 8-BIT CHARACTERS UP TO 2^{14})	CRC-2 16 BITS

11-2897

Figure 1-2 DDCMP Data Message Format

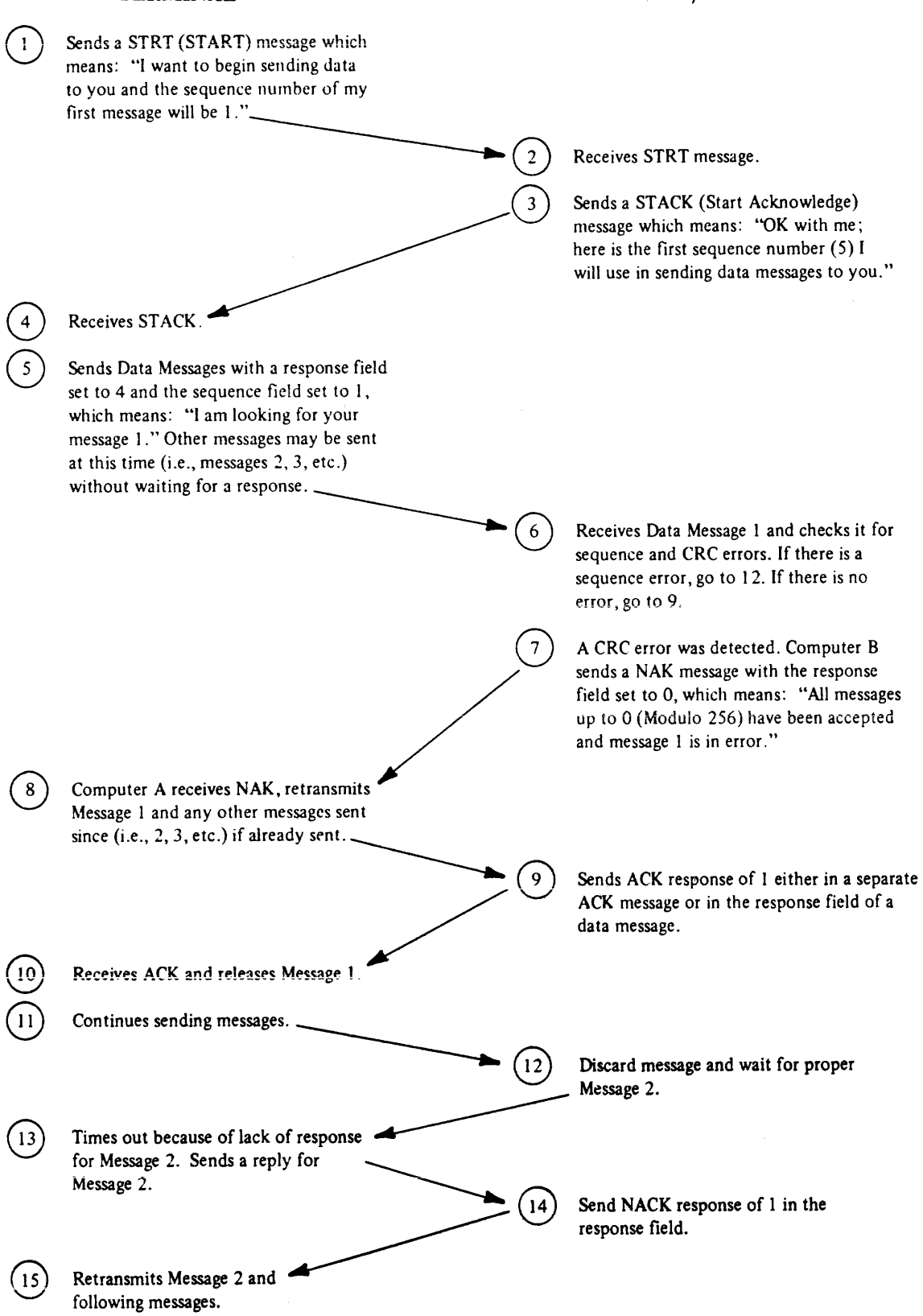


Figure 1-3 DDCMP Sample Handshaking Procedure

1.3.4.2 Error Checking and Recovery - DDCMP uses CRC-16 for detecting transmission errors. When an error occurs, DDCMP sends a separate NAK message. DDCMP does not require an acknowledgment message for all data messages. The number in the response field of a normal header, or in either the special NAK or ACK message, specifies the sequence number of the last good message received. For example, if messages 4, 5, and 6 have been received since the last time an acknowledgment was sent and message 6 is bad, the NAK message specifies number 5 which says "messages 4 and 5 are good and 6 is bad." When DDCMP operates in full-duplex mode, the line does not have to be turned around - the NAK is simply added to the sequence of messages for the transmitter.

When a sequence error occurs in DDCMP, the receiving station does not respond to the message. The transmitting station detects from the response field of the messages it receives (or via time-out) that the receiving station is still looking for a certain message and sends it again. For example, if the next message the receiver expects to receive is 5, but 6 is received, the receiver will not change the response field of its data messages, which contains 4. This says: "I accept all messages up through message 4 and I'm still looking for message 5."

1.3.4.3 Character Coding - DDCMP uses ASCII control characters for SYN, SOH, ENQ and DLE. The remainder of the message, including the header, is transparent.

1.3.4.4 Data Transparency - DDCMP defines transparency by use of a count field in the header. The header is of fixed length. The count in the header determines the length of the transparent information field, which can be 0 to 16,383 bytes long. To validate the header and count field, it is followed by a 16-bit CRC-16 field; all header characters are included in the CRC calculation. Once validated, the count is used to receive the data and to locate the second CRC-16 which is calculated on the datafield. Thus, character stuffing is avoided.

1.3.4.5 Data Channel Utilization - DDCMP uses either full- or half-duplex circuits at optimum efficiency. In the full-duplex mode, DDCMP operates as two dependent one-way channels, each containing its own data stream. The acknowledgments are the only dependency which must be sent in the data stream in the opposite direction.

Separate ACK messages are unnecessary and reduce control overhead. Acknowledgments are simply placed in the response field of the next message for the opposite direction. If several messages are received correctly before the terminal is able to send a message, all of them can be acknowledged by one response. Only when a transmission error occurs, or when traffic in the opposite direction is light (no data message to send) is it necessary to send a special NAK or ACK message, respectively.

In summary, DDCMP data channel utilization features include:

1. Low control character overhead
2. No "character stuffing"
3. No separate ACKs when traffic is heavy - saving on extra SYN characters and inter-message gaps
4. Multiple acknowledgments (up to 255) with one ACK
5. The ability to support point-to-point and multipoint lines.

1.3.4.6 Synchronization - DDCMP achieves synchronization through the use of two ASCII SYN characters preceding the SOH, ENQ, or DLE. It is not necessary to synchronize between messages as long as no gap exists. Gaps are filled with SYN characters. Two sync characters are required but more are usually transmitted. If synchronization between messages is deliberately lost by sending PAD (all 1s) characters, the inter-message interval must be at least 14 character times in length.

1.3.4.7 Bootstrapping - DDCMP has a bootstrap message as part of the protocol. It begins with the ASCII control character DLE. The information field contains the system reload programs and is totally transparent.

1.4 BASICS OF CYCLIC REDUNDANCY CHECKING

1.4.1 Mathematical Background

A cyclic code message consists of a specific number of data bits and a Block Check Character (BCC) that is computed by the CRC logic. Let n equal the total number of bits in the message and k equal the number of data bits; then n-k equals the number of bits in the BCC.

The code message is derived from two polynomials which are algebraic representations of two binary words: the generator polynomial P(X) and the message polynomial G(X). The generator polynomial is the type of code used (CRC-12, CRC-16, CRC-CCITT etc.); the message polynomial is the string of serial data bits. The polynomials are usually represented algebraically by a string of terms in powers of X, such as $X^n ... + X^3 + X^2 + X + X^0$ (or 1). In binary form, a 1 is placed in each position that contains a term; absence of a term is indicated by a 0. The convention followed in this manual is to place the least significant bit (X^0) at the right. For example, if a polynomial is given as $X^4 + X + 1$, its binary representation is 10011 (third and second degree terms are not present).

Given a message polynomial G(X) and a generator polynomial P(X), the objective is to construct a code message polynomial F(X) that is evenly divisible by P(X). It is accomplished as follows:

1. Multiply the message G(X) by X^{n-k} where n-k is the number of bits in the BCC.

2. The resulting product X^{n-k} [G(X)] is divided by the generator polynomial P(X).

3. The quotient is disregarded and the remainder C(X) is added to the product to yield the code message polynomial F(X), which is represented as X^{n-k} [G(X)] + C(X).

The division is performed in binary without carries or borrows. In this case, the remainder is always one bit less than the divisor. The remainder is the BCC and the divisor is the generator polynomial; therefore, the bit length of the BCC is always one less than the number of bits in the generator polynomial.

A simple example is explained below.

1. Given:

   Message polynomial G(X) = 110011 ($X^5 + X^4 + X + X^0$)
   Generator polynomial P(X) = 11001 ($X^4 + X^3 + 1$)
   G(X) contains 6 data bits
   P(X) contains 5 bits and will yield a BCC with 4 bits; therefore, n-k = 4.

2. Multiplying the message G(X) by X^{n-k} gives:

   $$X^{n-k} [G(X)] = X^4 (X^5 + X^4 + X + X^0) = X^9 + X^8 + X^5 + X^4$$

   The binary equivalent of this product contains 10 bits and is 1100110000.

3. This product is divided by P(X)

```
                    100001 ← quotient
              ┌──────────
P(X) → 11001  │1100110000 ← X^(n-k) [G(X)]
              11001
              ─────
                   10000
                   11001
                   ─────
                    1001 ← remainder = C(X) = BCC
```

4. The remainder C(X) is added to X^{n-k} [G(X)] to give F(X) = 1100111001.

The code message polynomial is transmitted. The receiving station divides it by the same generator polynomial. If there is no error, the division will produce no remainder and it is assumed that the message is correct. A remainder indicates an error. The division is shown below.

```
                    100001
              ┌──────────
P(X) → 11001  │1100111001 ← F(X)
              11001
              ─────
                   11001
                   11001
                   ─────
                   00000 ← no remainder
```

1.4.2 Hardware Implementation of CRC

The BCC is computed and accumulated in a shift register during transmission. Another shift register is used during reception to examine the received data and BCC. In each register, the number of stages is equal to the degree of the generating polynomial. In the DUP11, the registers have 16 stages because 16-degree generating polynomials are used. SDLC uses code CRC-CCITT whose generator polynomial is $X^{16} + X^{12} + X^5 + 1$. DDCMP uses code CRC-16 whose generator polynomial is $X^{16} + X^{15} + X^2 + 1$. Both the transmitter and receiver CRC registers have control logic that allows the registers to be configured for the selected CRC code.

When a message and accompanying BCC character have been received, the CRC logic only indicates whether the message is in error or not. It does not correct errors nor does it even enumerate or locate errors. Under protocol discipline, the sending station is requested to re-transmit the message.

1.4.3 CRC Operation in DDCMP Protocol

Under DDCMP protocol control, CRC operation is exactly like that described in Paragraph 1.4.1, Mathematical Background.

The transmitter and receiver CRC registers are initialized to all 0s. At the sending station, the transmitter CRC register processes the information being transmitted and accumulates the BCC. When the last bit of information has been transmitted, the contents of the transmitter CRC register are transmitted.

At the receiving station, the information plus the 16-bit BCC is examined by the receiver CRC register. At the end of the message (information plus BCC), the contents of the receiver CRC register should read 0 if the message is error-free. The CRC error detection logic asserts a flag if the register reads 0. If an error is present, the register reads non 0 and the flag is not asserted. The DUP11 does not count characters so it is the program's responsibility to look for the CRC error flag at the proper time.

1.4.4 CRC Operation in SDLC Protocol

Under SDLC protocol control, CRC operation is slightly different than that described in Paragraph 1.4.1, Mathematical Background. The differences are:

1. The factor X^k ($X^{15} + X^{14} .... + X + 1$) is added to X^{n-k} [G(X)] which corresponds to initializing the transmitter CRC register to all 1s. This function protects against the obliteration of leading flags, which may not be detected if the register is 0.

2. The accumulated BCC, which is called Frame Check Sequence (FCS) in the SDLC mode, is complemented before being transmitted. This results in a unique non-0 remainder (016417_8) at the receiver. This protects against obliteration of terminating flags which may not be detected if the remainder is 0.

3. At the receiving station, the receiver CRC register is initialized to all 1s. The information plus the FCS constitutes the message and it is added to X^k ($X^{15} + X^{14} .... + X + 1$) and divided by P(X) to give 016417_8, if the transmission is error-free. If an error is present, the flag is asserted. This CRC check is performed only when the flag is received.

CHAPTER 2
INSTALLATION

2.1 SCOPE
This chapter provides information for installing and checking out a DUP11 Synchronous Line Interface.

2.2 UNPACKING AND INSPECTION
There is only one version of the DUP11 – the DUP11-DA; it consists of six items (refer to Figure 2-1).

M7867 Bit Synchronous Interface
BC22F-25 Cable
BC08S-10 Cable
H325 Test Connector
H3001 Distribution Panel
74-27292 Bracket*

Inspect these parts for visible damage. Report any damage or shortage immediately to the shipper and the DIGITAL representative.

2.3 TOOLS REQUIRED FOR INSTALLATION
The standard field service tool kit contains all the required tools for the installation of the DUP11.

2.4 PREINSTALLATION SET-UP PROCEDURES
Before installing the DUP11 option, the following five steps must be performed.

1. Examine the eight jumpers (W1 – W8) on the M7867 module. Refer to Figure 2-2 to locate and identify the jumpers. All M7867 modules are shipped with the standard jumper configuration described in Table 2-1. All DUP11 diagnostics must be run on each M7867 utilizing the standard jumper configuration. After successfully completing the diagnostic testing in the shipped configuration, the M7867 may be reconfigured to meet the customer's requirements. MAINDEC CZDPE (DUP11 Quick Verify Test) should then be run to verify operation of the new configuration.

2. The DUP11 device address must be selected in accordance with Paragraph 2.9. For diagnostics, device address default = 760050.

3. The DUP11 vector address must be selected in accordance with Paragraph 2.10. For diagnostics, vector address default = 770.

*Used in configurations not incorporating I/O bulkhead.

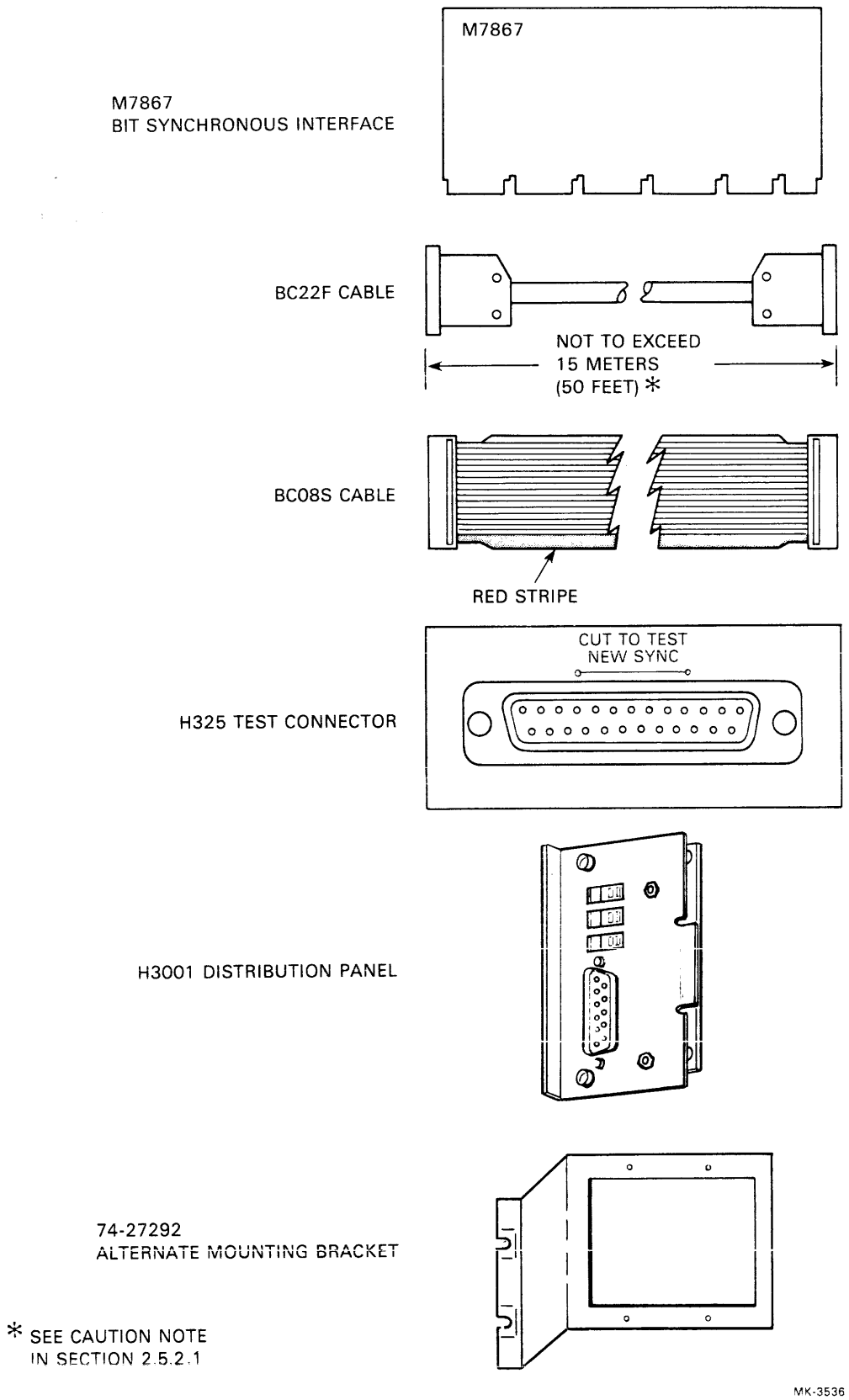


Figure 2-1 DUP11 Parts Diagram

Table 2-1 M7867 Jumper Configuration

Jumper Number	Standard Configuration	Function
W1	Installed	Secondary Receive Enable – With this jumper installed, the state of the data set Secondary Received Data line is received by the DUP11. This jumper is used in conjunction with jumper W2. With this jumper removed, pin JJ of the Berg header is available for some other function.
W2	Removed	Secondary Receive Disable – This jumper must be removed when W1 is installed. Conversely, it must be installed when W1 is removed. When installed, the EIA SEC REC receiver input is grounded; however, this has no effect on the Berg header, cable, or data set.
W3	Installed	Clear option – With this jumper removed, the following bits cannot be directly cleared by DEVICE RESET or BUS INIT. Secondary Transmit Data (RXCSR bit 3) Request to Send (RXCSR bit 2) Data Terminal Ready (RXCSR bit 1) Some data sets may require that these connections be excluded from a device reset function.
W4	Installed	Secondary Transmit Enable – With this jumper installed, the state of the Secondary Transmit Data line is sent to the data set. With this jumper removed, this signal is disconnected at the output of the EIA driver. Some data sets do not use this signal.
W5	Removed	A Data Set Control – With this jumper removed, positive transitions on the Ring line and any transitions on the Clear to Send line set ADAT SET CH. This flag requests a receiver interrupt if the DSITEN bit has been set by the program. With this jumper installed, any transition on three additional lines set ADAT SET CH: Carrier Data Set Ready Secondary Received Data
W6	Installed	A and B Data Set Control – With this jumper installed, transitions on the Carrier, Data Set Ready, and Secondary Received Data lines set BDAT SET CH. This signal is a flag only and does not request interrupts. With this jumper removed, the BDAT SET CH flag (RXCSR bit 0) is inhibited.
W7	Installed	NPR Latency Improvement – With this jumper installed, the NPR latency improvement circuit in the interrupt control logic is enabled. This jumper should be removed only if the DUP11 is installed in a system using a KA11 processor with no KH11 latency reduction option.
W8	Installed	External Clock Enable – Remove for Bell 201A modem.

4. Confirm that a BR5 priority plug is installed in the module. The diagnostics assume a BR5 priority level (see Figure 2-2 to locate and identify the BR5 plug).

5. Set up the H3001 module in accordance with Paragraph 2.7.

2.5 INSTALLATION

Installation of the DUP11 is treated in two paragraphs. Paragraph 2.5.1 contains instructions for installing the M7867 module. Paragraph 2.5.2 contains instructions for installing the H3001 distribution panel.

Examine Figure 2-3. This drawing shows the cabling configuration for the DUP11 installation.

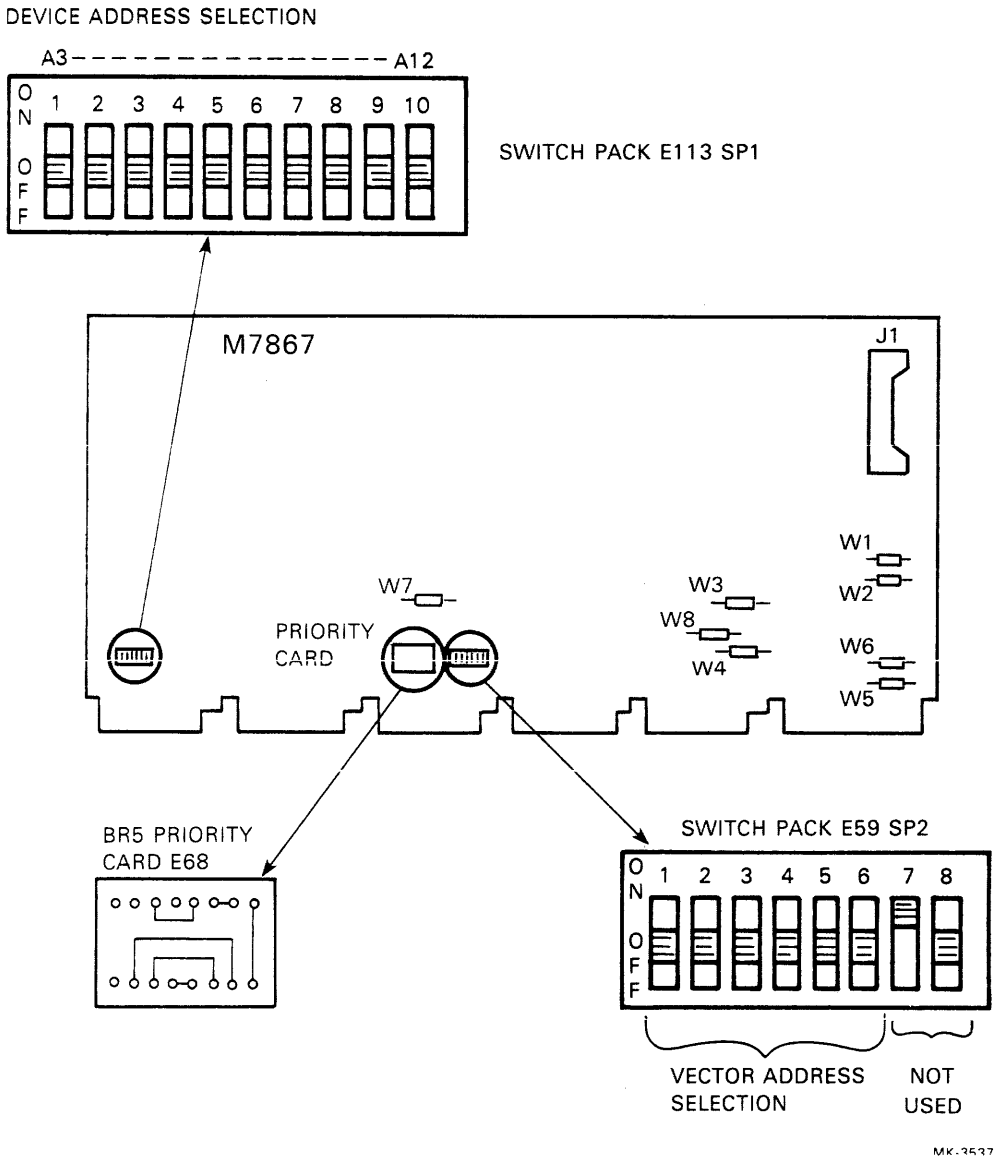


Figure 2-2 Component Location

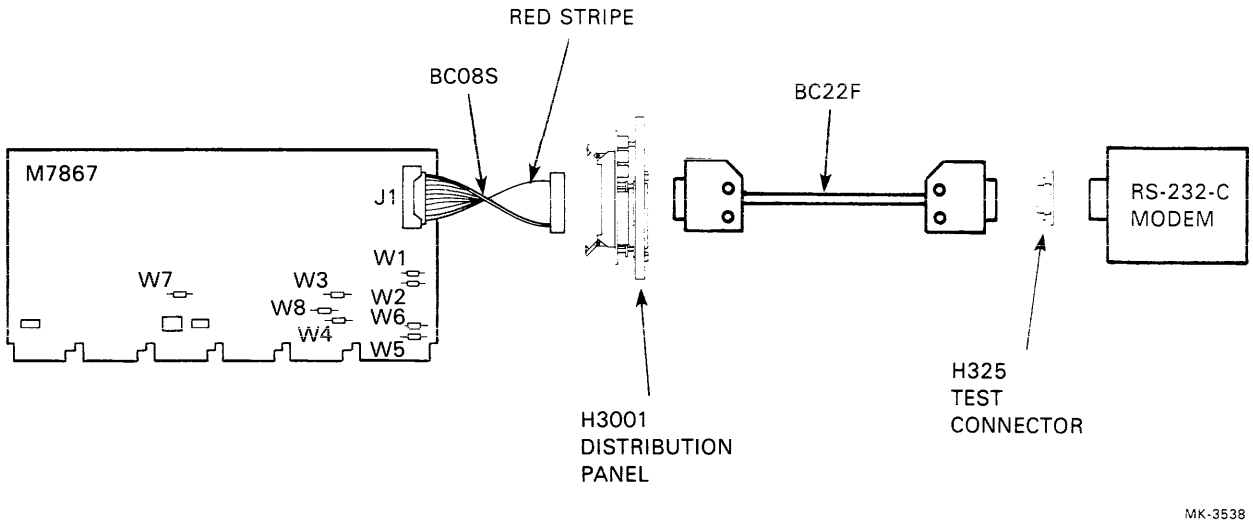


Figure 2-3 DUP11 Cabling

2.5.1 M7867 Module Installation

The DUP11 can be installed in any small peripheral controller (SPC) hex slot in the PDP-11 UNIBUS. Figure 2-4 shows the DD11-B system unit. This unit contains four slots but the DUP11 can only be installed in slots 2 and 3 because of the configuration of the prewired backplane.

WARNING

Turn all power OFF.

CONNECTOR

SLOT	A	B	C	D	E	F
1	UNIBUS IN (NOTE 2)			G727*		
2	M7867 HEX MODULE (NOTE 1)					
3				G727*		
4	UNIBUS OUT (NOTE 3)			G727*		

MODULE SIDE VIEW

*G727 GRANT CONTINUITY MODULE MUST BE INSTALLED IN EACH SLOT IN WHICH AN INTERFACE MODULE IS NOT INSTALLED.

NOTES
1. M7867 CAN BE MOUNTED ONLY IN SLOT 2 OR 3.
2. CAN BE M920 UNIBUS CONNECTOR OR BC11S UNIBUS CABLE.
3. CAN BE M920, BC11A, OR M930 UNIBUS TERMINATOR.

MK-3539

Figure 2-4 DUP11 (M7867 Module) Mounted in DD11-B

The M7867 installation procedure is as follows:

1. Connect the female Berg connector on the BC08S cable (ribbed side up) to the header on the M7867 module.

2. Plug the module into an SPC slot or into slot 2 or 3 of the DD11-B system unit.

2.5.2 H3001 Distribution Panel Installation

Two different approaches to installing the H3001 distribution panel assembly are included in this manual.

FCC regulations necessitate the incorporation of I/O bulkheads in most new installations to limit electromagnetic interference (EMI) leakage. For installations utilizing an I/O bulkhead, follow the steps outlined in Paragraph 2.5.2.1.

Alternate instructions are included for non-FCC compliant cabinets that require a slightly modified installation procedure. If the system does not incorporate an I/O bulkhead, follow the procedures in Paragraph 2.5.2.2.

2.5.2.1 H3001 Installation In an I/O Bulkhead – The following instructions are for cabinets utilizing an I/O bulkhead. If a particular cabinet does not include an I/O bulkhead, omit these steps and follow the instructions in Paragraph 2.5.2.2.

Though there are differences in the orientation and positioning of I/O bulkheads of different levels of the PDP-11, the installation concept is the same. Once the H3001 distribution panel is installed, there should be no openings (panels omitted) left in the I/O frame on the rear of the cabinet which could permit EMI leakage. For this reason, it is important to tighten both mounting screws on the distribution panel. Figures 2-5 and 2-6 depict the various I/O bulkhead types and illustrate the correct approach to each.

1. Gain access to the I/O bulkhead through the door on the rear of the system cabinet and remove one of the 4.5 cm (2 in) wide panels on the bulkhead.

2. Route the remaining BC08S cable through the cabinet and through the opening in the I/O bulkhead at the rear of the cabinet. Keep in mind that the cable must be routed and dressed in a manner compatible with the existing cabinet cabling.

3. Plug the connector on the free end of the BC08S cable into the Berg connector on the rear of the H3001 distribution panel. Make sure that the ribbed side of the cable faces the pins lettered A to UU (not B to VV) of the Berg connector (see Figure 2-9). This assures pin to pin correspondence between the connectors of the M7867 and H3001 modules.

4. Install the panel into the opening of the I/O bulkhead in place of the 4.5 cm (2 in) panel that was removed in Step 1.

NOTE

It is imperative to maintain an interference-free environment outside the cabinet enclosure. Any additional panels that may have been removed to facilitate easier installation of the H3001 must be replaced in order to maintain the integrity of the I/O bulkhead.

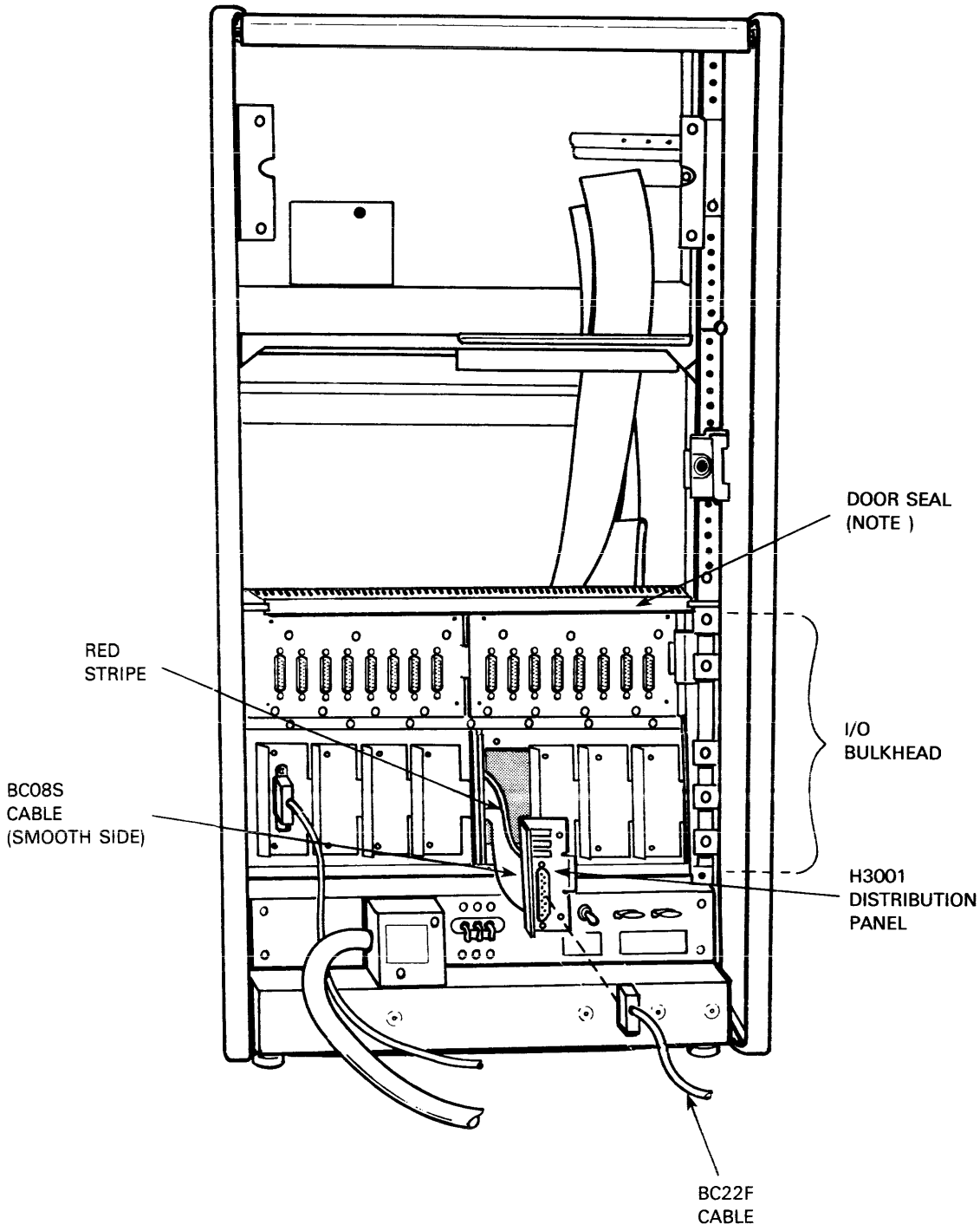


Figure 2-5 H3001 Installation in a Horizontally Oriented I/O Bulkhead

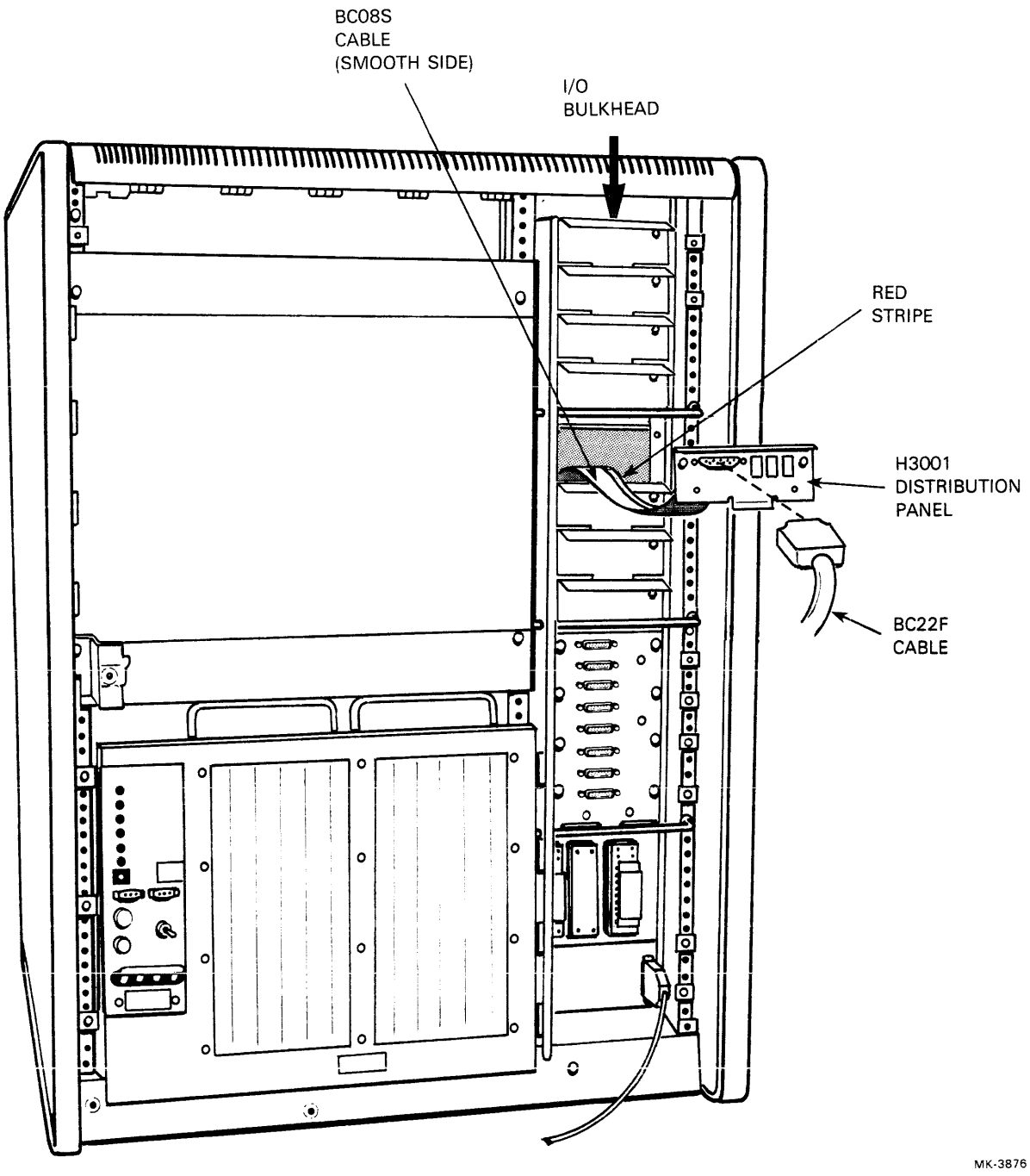


Figure 2-6 H3001 Installation in a Vertically Oriented I/O Bulkhead

5. Connect the female Cinch connector of the BC22F cable to the 25-pin Cinch connector on the rear of the H3001 module. The cable should exit the cabinet with the other signal cables.

CAUTION

BC22F cable lengths in excess of 7.62 meters (25 feet) may exceed the maximum load capacitance defined by the RS-232-C specification. Note, however, that up to 15 meters (50 feet) provides satisfactory DUP11 performance levels.

6. Connect the other end of the BC22F cable to the modem or to the H325 test connector which is the configuration assumed by the diagnostics.

7. Turn power ON.

2.5.2.2 H3001 Installation in Cabinets Without an I/O Bulkhead –

1. Gain access to the rear of the system cabinet and mount the bracket (Part No. 74-27292) to one of the rear side rails as shown in Figure 2-7. Mount the H3001 distribution panel into the bracket.

2. Route the remaining BC08S cable through the cabinet and to the bracket at the rear of the cabinet. Keep in mind that the cable must be routed and dressed in a manner compatible with the existing cabinet cabling.

3. Plug the connector on the free end of the BCO8S cable into the Berg connector on the rear of the H3001 distribution panel. Make sure that the ribbed side of the cable faces the pins lettered A to UU (not B to VV) of the Berg connector (see Figure 2-9). This assures pin to pin correspondence between the connectors of the M7867 and H3001 modules.

4. Connect the female Cinch connector of the BC22F cable to the 25-pin Cinch connector on the rear of the H3001 module. The cable should exit the cabinet with the other signal cables.

CAUTION

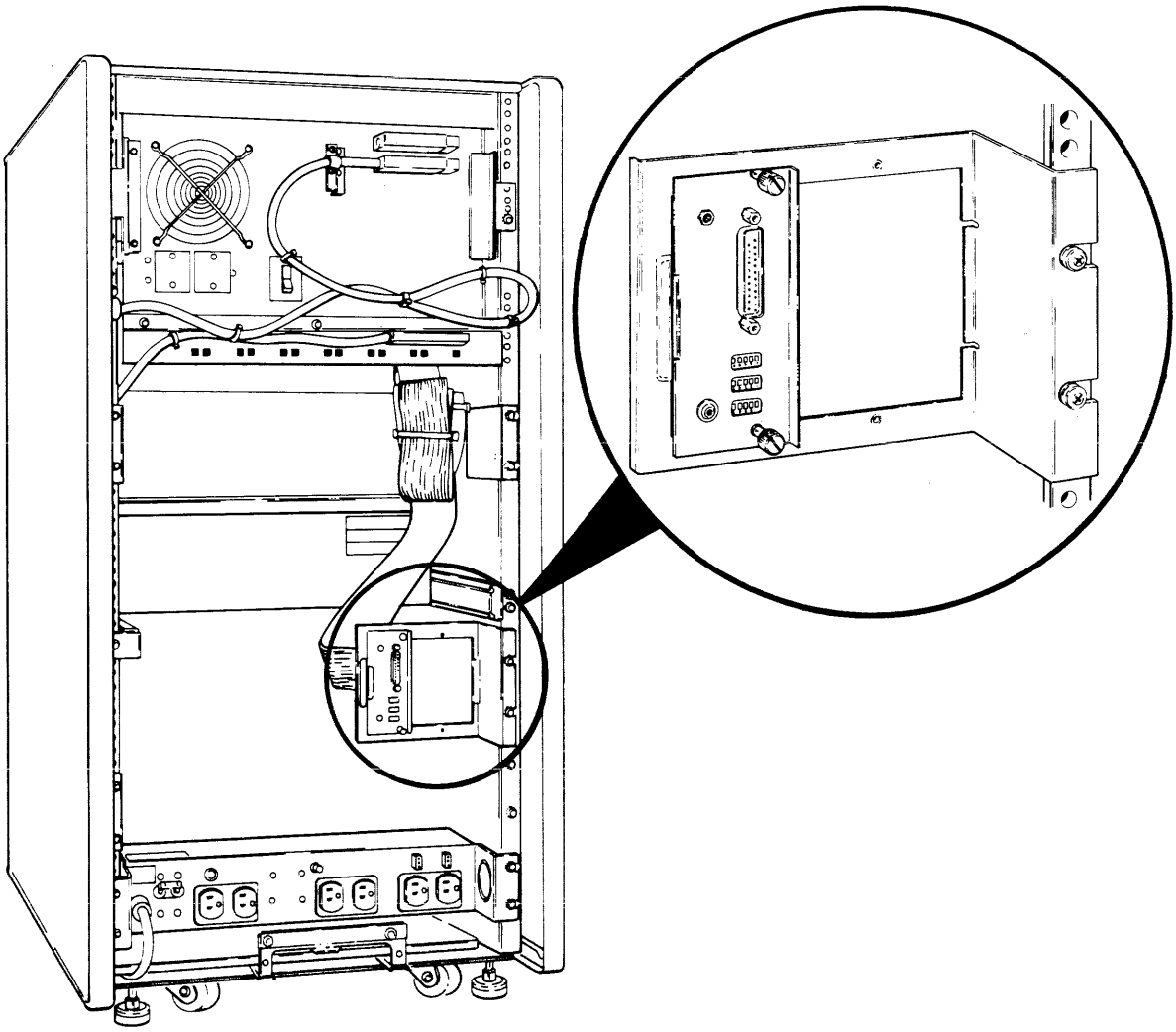
BC22F cable lengths in excess of 7.62 meters (25 feet) may exceed the maximum load capacitance defined by the RS-232-C specification. Note, however, that up to 15 meters (50 feet) provides satisfactory DUP11 performance levels.

5. Connect the other end of the BC22F cable to the modem or to the H325 test connector which is the configuration assumed by the diagnostics.

6. Configure the H3001 panel switches according to the chart in Table 2-2.

7. Turn power ON.

Figure 2-8 is included for convenience. Use this figure for quick reference when installing the DUP11 option.



MK-3880

Figure 2-7 Side Rail Installation of H3001 Distribution Panel

Table 2-2 H3001 Switch Settings

	DF03	BELL 201C	BELL 208B	BELL 209A	DIAGNOSTICS & H325
S1					ON
S2					
S3					
S4					
S5					ON
S6					
S7					
S8					
S9					
S10		*			
S11					
S12					
S13					
S14					
S15					

SWITCHES ARE OFF UNLESS OTHERWISE INDICATED

* ON IF NEW SYNC CONFIGURED ON M7867

MK-3838

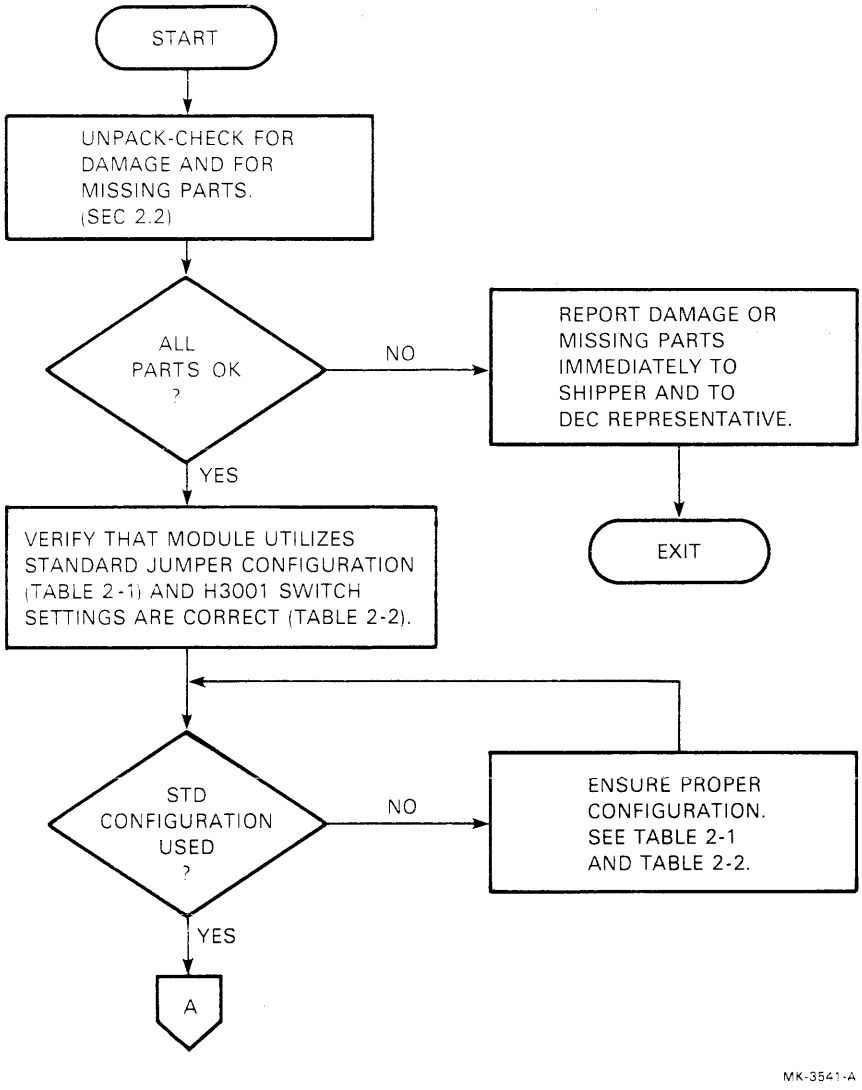


Figure 2-8 Installation Procedure Flowchart (Sheet 1 of 6)

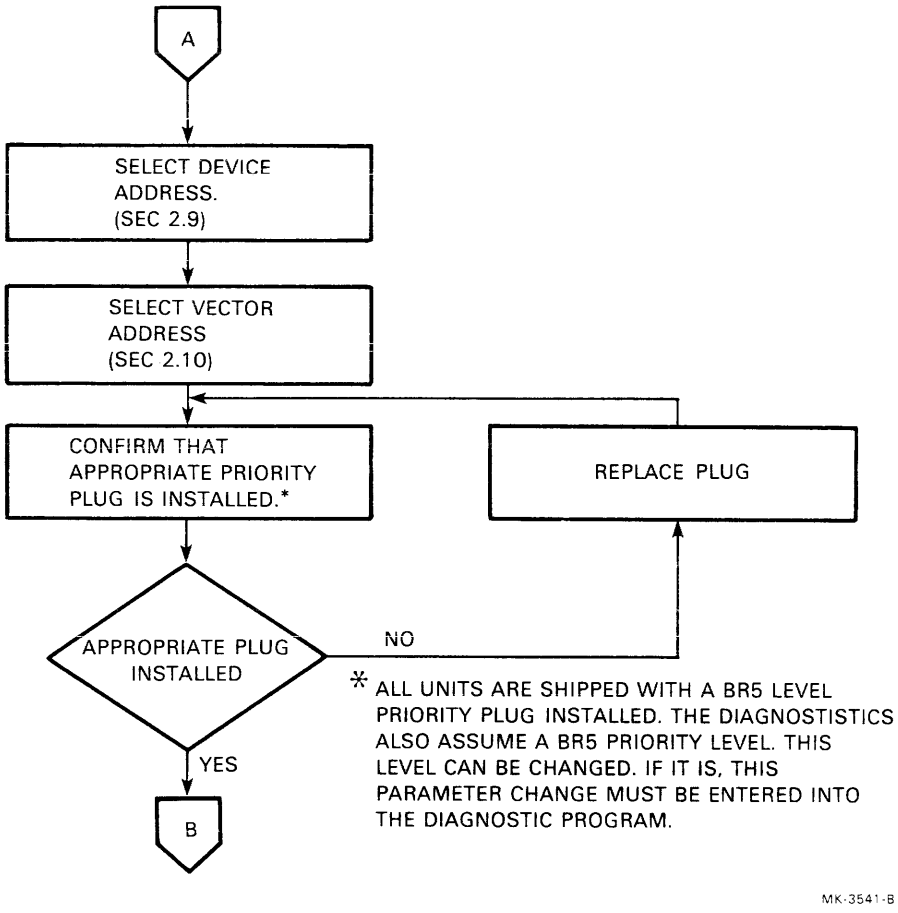


Figure 2-8 Installation Procedure Flowchart (Sheet 2 of 6)

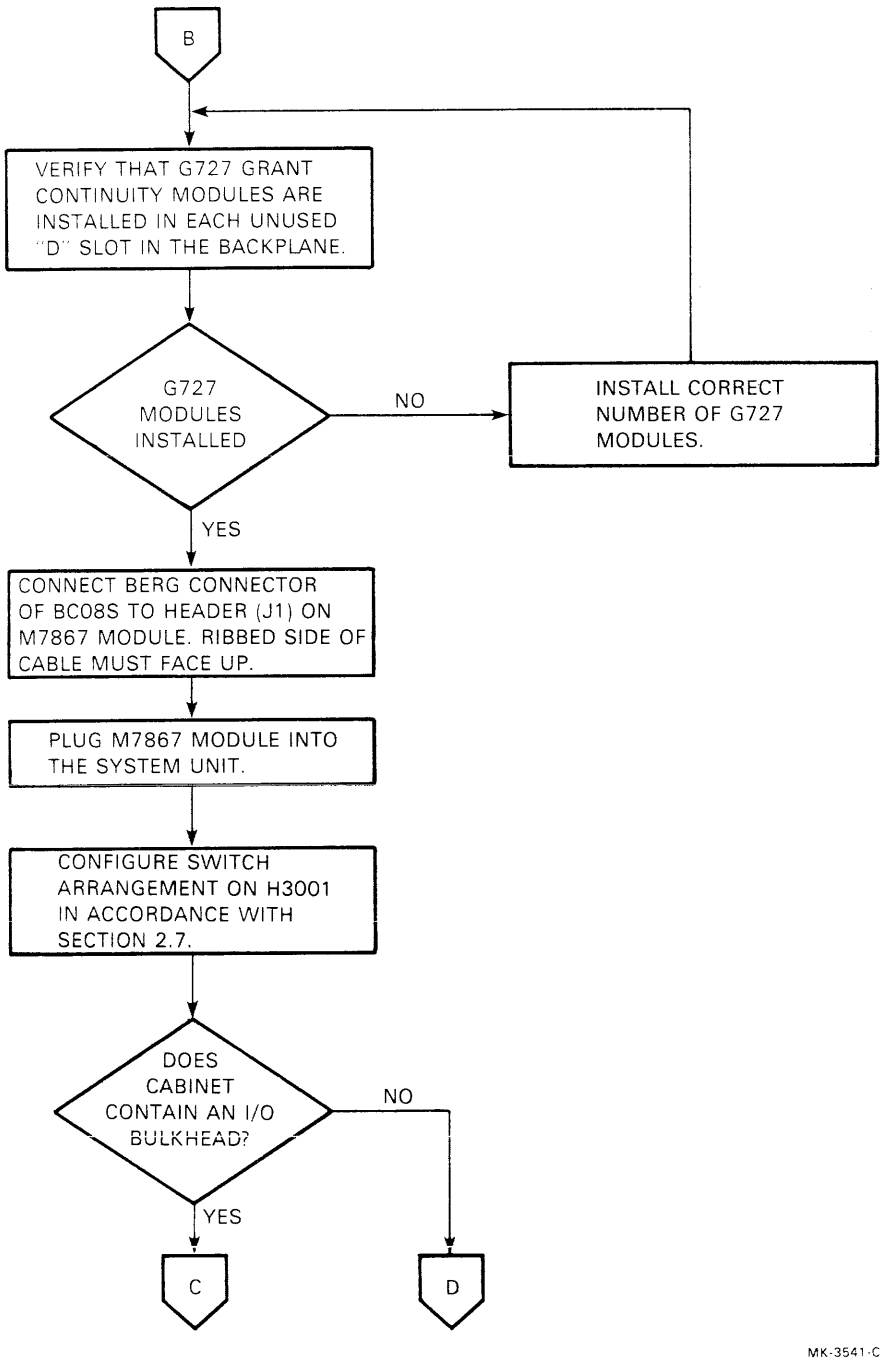


Figure 2-8 Installation Procedure Flowchart (Sheet 3 of 6)

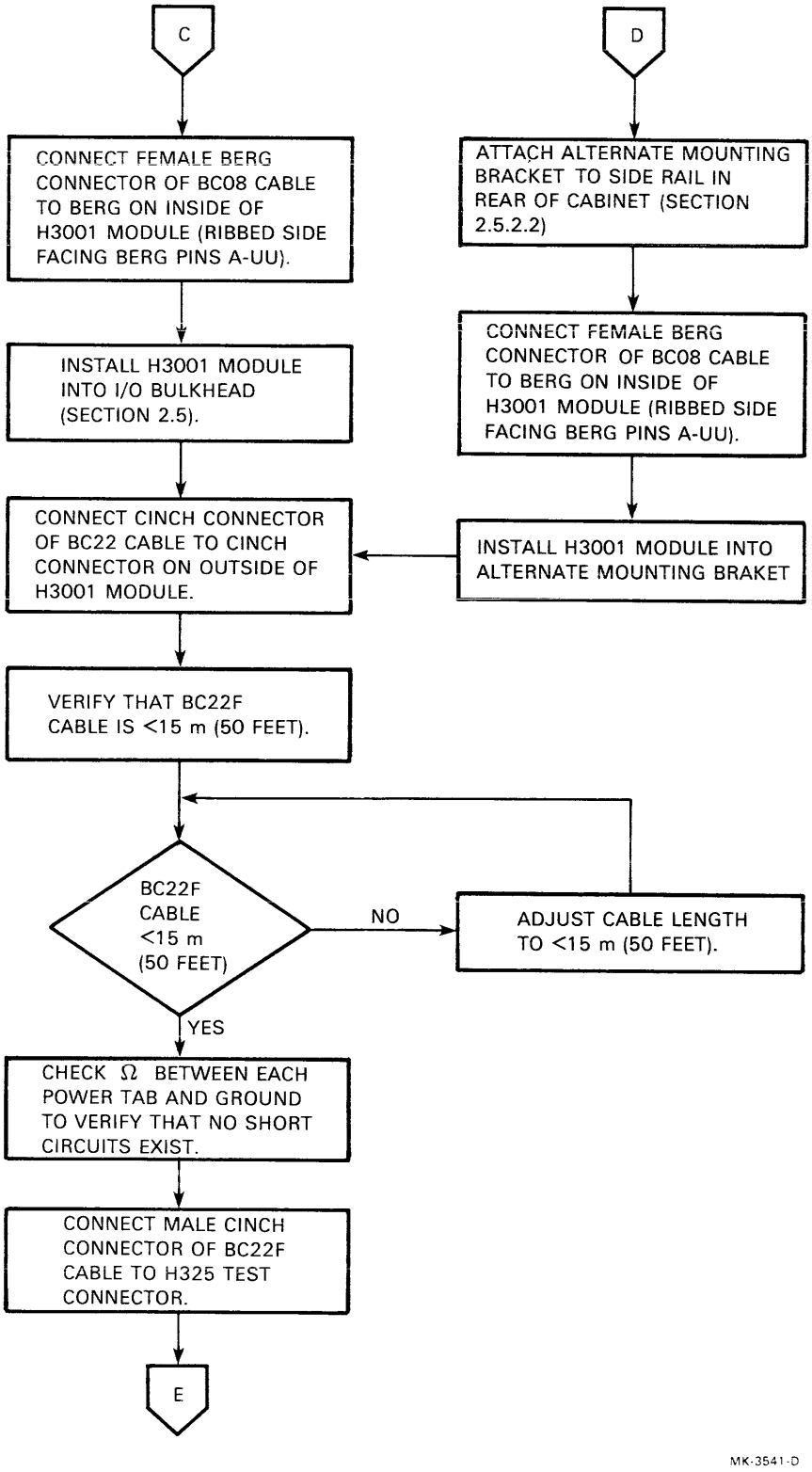


Figure 2-8 Installation Procedure Flowchart (Sheet 4 of 6)

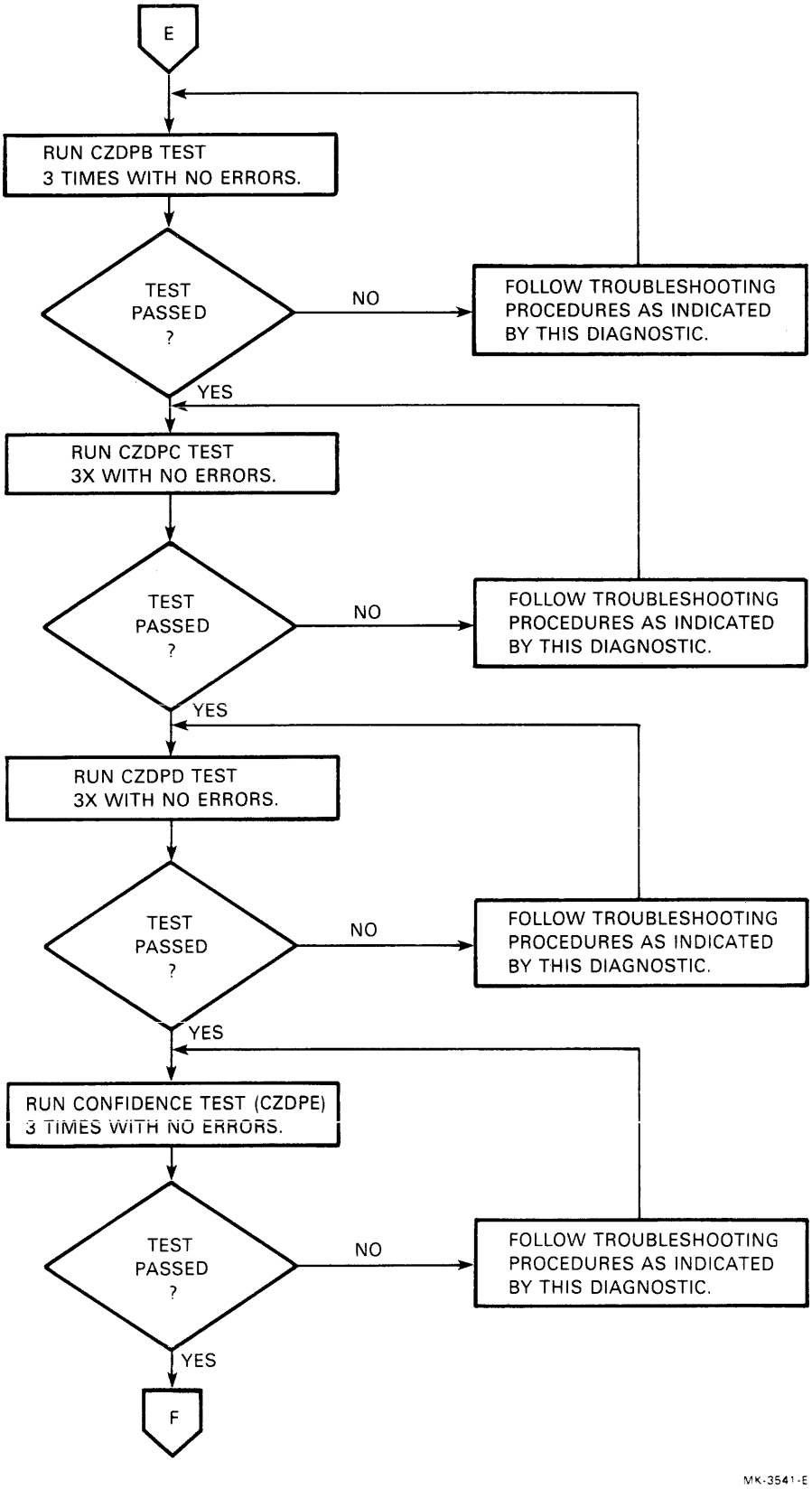


Figure 2-8 Installation Procedure Flowchart (Sheet 5 of 6)

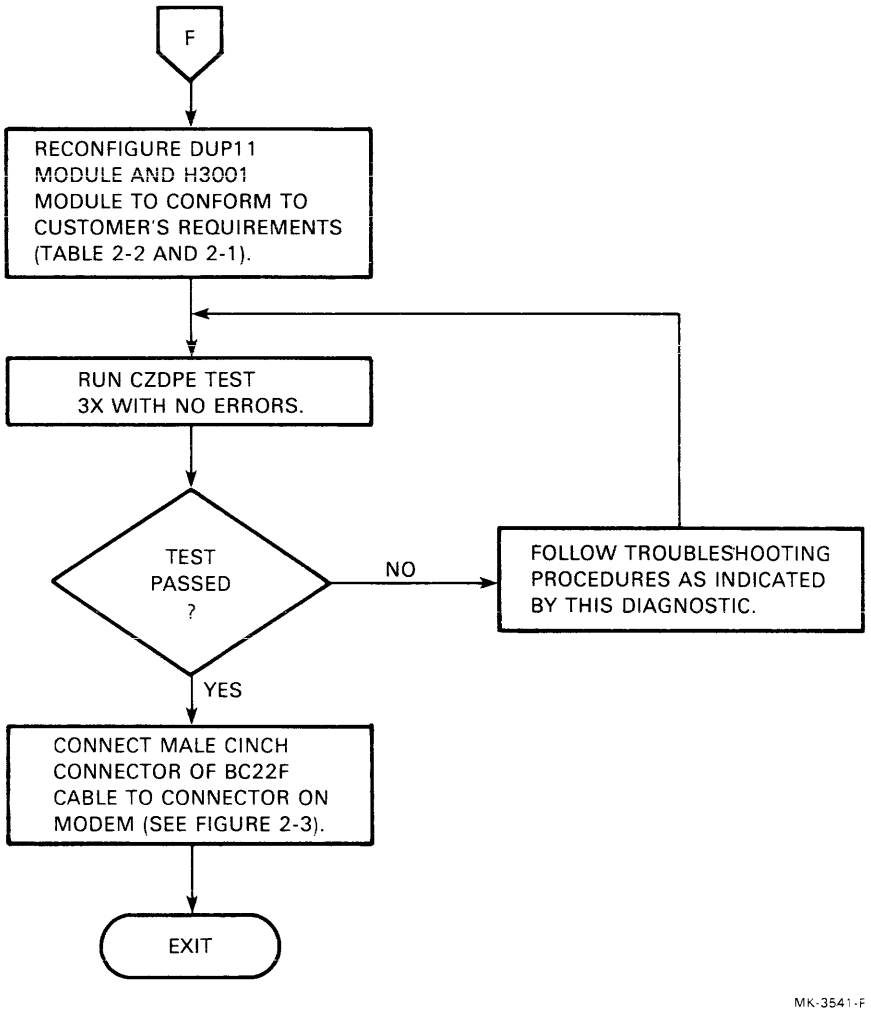


Figure 2-8 Installation Procedure Flowchart (Sheet 6 of 6)

2.6 VERIFICATION OF HARDWARE OPERATION

Verification of proper DUP11 operation is performed by a series of diagnostic programs. A general description of the diagnostics is included in Chapter 5, Maintenance. Details on the content and use of the diagnostics is contained in the diagnostic documentation package supplied with the DUP11.

Proceed as follows:

1. Run the following diagnostics in the following order:

   CZDPB – Basic and Off-line Transmitter Tests

   CZDPC – Off-line and SDLC Receiver Tests and Off-line Modem Control and Interrupt Tests

   CZDPD – Off-line SDLC and DEC MODE Data and Function Tests

2. Run diagnostic CZDPE. This is a confidence test that requires a dialog with the user to ensure proper setting of the DUP11 and system parameters. It offers a quick test to verify that the DUP11 is operational.

Each diagnostic must make three passes without an error.

Reconfigure the H3001 and the DUP11 in accordance with the customer's requirements (Tables 2-1 and 2-2). Then run diagnostic CZDPE (DUP11 quick verify test) to check the final configuration.

System testing consists of running DECX11 module CXDPB to exercise all DUP11s in a system. Run DECX11 until three error-free passes of module CXDPB are obtained. Note that only four DUP11s can be tested with one DECX11 module.

2.7 COMPATIBILITY

The DUP11 is compatible with the DF03 and Bell type 201™, 208™, and 209™ modems or equivalent. In addition, compatibility with these and other modems is enhanced through the incorporation of the H3001 distribution panel.

Adjust the switches on the H3001 to correspond to the settings indicated in Table 2-2 for the particular modem used in your configuration. The H3001 switches are contained in three DIP switch packages grouped together on the H3001 modules. Refer to Figure 2-9 for the location of these switches. The switches are rocker or slide type and are pushed to the desired position.

A schematic of the H3001 distribution panel is included in Chapter 5 (Figure 5-1). Use this figure as an aid in determining the proper switch settings and jumper configuration if the modem used is not listed in Table 2-2.

Jumper W1 (see Figure 2-9) is normally not installed. Install this jumper when RS-232 protective ground (pin 1 of Cinch connector) must be connected to enclosure ground. Note that this may introduce an undesirable ground loop.

Bell 201, 208, and 209 are trademarks of Western Electric.

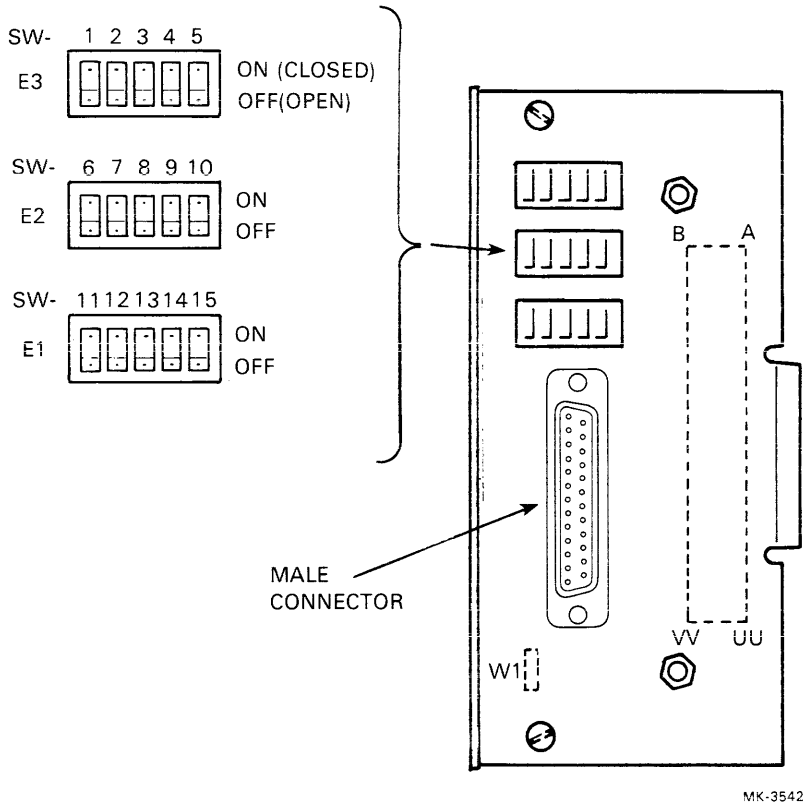


Figure 2-9 H3001 Distribution Panel

NOTE

Due to the extensive variety of modems currently available, DIGITAL cannot guarantee that the DUP11 interface will fully support all features of every modem.

2.8 POWER REQUIREMENTS

The DUP11 requires the following power:

+5 V at 3.6 A
+15 V at 75 mA
−15 V at 75 mA

2.9 DEVICE ADDRESSES

2.9.1 Introduction

Starting with the DJ11, new communications devices are to be assigned floating addresses. The addresses for current production devices are to be retained. The word *floating* means that addresses are not assigned absolutely for the maximum number of each communications device that can be used in a system.

2.9.2 Floating Device Address Assignment

Floating device addresses are assigned as follows:

1. The floating address space starts at location 760010 and extends to location 764000 (octal designations).

2. The devices are assigned in order by type: DJ11, DH11, DQ11, DU11, and DUP11; then the next device is introduced into production. Multiple devices of the same type must be assigned contiguous addresses.

3. The first address of a new type device must start on a modulo 10_8 boundary, if it contains one to four bus-addressable registers. The starting address of the DH11 must be on a modulo 20_8 boundary because the DH11 has eight registers.

4. A gap of 10_8, starting on a modulo 10_8 boundary, must be left between the last address of one type device and the first address of the next type device. A gap must be left for any device on the list that is not used if the device following it is used. The equivalent of a gap should be left after the last assigned device to indicate that nothing follows.

5. A new type device cannot be inserted ahead of a device on the list.

6. If additional devices on the list are to be added to a system, they must be assigned contiguously after the original devices of the same type. Reassignment of other type devices already in the system may be required to make room for the additions.

The following examples show typical floating device assignments for communications devices in a system:

EXAMPLE 1: No DJ11s, 2 DH11s, 2 DQ11s, and 1 DUP11

760010	DJ11 gap
760020	DH11 #0 first address
760040	DH11 #1 first address
760060	DH11 gap
760070	DQ11 #0 first address
760110	DQ11 gap
760120	DU11 gap
760130	DUP11 #0 first address
760140	Indicates no more DUP11s and no other devices follow.

EXAMPLE 2: 1 DJ11, 1 DH11, 2 DQ11s, and DUP11s

760010	DJ11 #0 first address
760020	DJ11 gap
760040	DH11 #0 first address
760060	DH11 gap
760070	DQ11 #0 first address
760100	DQ11 #1 first address
760110	DQ11 gap
760120	DUP11 gap
760130	DUP11 #0 first address
760140	DUP11 #1 first address
760150	Indicates no more DUP11s and no other devices follow.

EXAMPLE 3: 1 DUP11

760010 DJ11 gap
760020 DH11 gap
760030 DQ11 gap
760040 DU11 gap
760050 DUP11 #0 first address
760060 Indicates no more DUP11s and no other devices follow.

2.9.3 Device Address Selection

In the floating address space (760010-764000), bits 13-17 are always 1s (function of PDP-11 processor). Appendix B shows the PDP-11 memory organization and addressing conventions. Bits 3-12 are selected by switches in the address decoding logic (Table 2-3). With the switch ON (closed), the decoder looks for a 0 on the associated UNIBUS address line; conversely, with the switch OFF (open), the decoder looks for a 1 on the associated UNIBUS address line. Bits 1 and 2 are decoded to select one of four registers. They determine the least significant digit (octal) of the device address because bit 0 is not used for address decoding. It is used to select the proper byte during byte transactions.

Table 2-3 Guide for Setting Switches to Select Device Address

Switch No.	10	9	8	7	6	5	4	3	2	1	Device
Bit No.	12	11	10	9	8	7	6	5	4	3	Address
										X	760010
									X		760020
									X	X	760030
								X			760040
								X		X	760050
								X	X		760060
								X	X	X	760070
							X				760100
						X					760200
						X	X				760300
					X						760400
					X		X				760500
					X	X					760600
					X	X	X				760700
				X							761000
			X								762000
			X	X							763000
		X									764000

Notes:
1. X means switch off (open) to respond to logical 1 on the Unibus.
2. Switch numbers are physical positions in switch package 1.

Switch Identification
The device address selection switches are contained in one DIP switch package (E113). Refer to Figure 2-2 for the location of the package. All ten switches in the package are used. The correlation between switch numbers and bit numbers is shown in Table 2-3. The ON and OFF positions and the switch numbers are marked on the package. The switches are rocker or slide type and are pushed to the desired position.

The DUP11 requires four addresses:

76XXX0 Receiver Control and Status Register

76XXX2 Receiver Data Buffer Register (Read Only) and Parameter Control and Status Register (Write Only)

76XXX4 Transmitter Control and Status Register

76XXX6 Transmitter Data Buffer Register

2.10 VECTOR ADDRESSES

2.10.1 Introduction
Communications devices are assigned floating vector addresses. This eliminates the necessity of assigning addresses absolutely for the maximum number of each device that can be used in the system.

2.10.2 Floating Vector Address Assignment
Floating vector addresses are assigned as follows:

1. The floating address space starts at location 300 and proceeds upward to 777. Addresses 500-534 are reserved.

2. The devices are assigned in order by type: DC11; KL11/DL11-A, -B; DP11; DM11-A: DN11; DM11-BB; DR11-A; DR11-C; PA611 Reader; PA611 Punch; DT11; DX11; DL11-C, -D -E; DJ11; DH11; GT40; LPS11; VT20; DQ11; KW11-W; DU11; DUP11; and DV11.

3. If any type device is not used in a system, address assignments move up to fill the vacancies.

4. If additional devices are to be added to the system, they must be assigned contiguously after the original devices of the same type. Reassignment of other type devices already in the system may be required.

2.10.3 Vector Address Selection
Each device interrupt vector requires four address locations (two words) which implies only even-numbered addresses. A further constraint is that all vector addresses must end in 0 or 4. The vector address is specified as a three-digit, binary-coded, octal number using UNIBUS data bits 0-8. Because the vector must end in 0 or 4, bits 1 or 0 are not specified (they are always 0) and bit 2 determines the least significant octal digit of the vector address (0 or 4). The interrupt control logic sends only seven bits (2-8) to the PDP-11 processor to represent the vector address.

The DUP11 is shipped with a BR5 priority selection plug installed in the interrupt control logic. This logic generates two vector addresses: receiver interrupts generate vector addresses of the form XX0, and transmitter interrupts generate vector addresses of the form XX4. For this method of operation, the state of bit 2 is selected by the logic, not by a switch. The two most significant octal digits of the vector address are determined by switches in lines 3-8 (Table 2-4). With the switch OFF (open), a 0 is generated on the associated UNIBUS data line; with the switch ON (closed), a 1 is generated on the associated UNIBUS data line. Also, the NPR jumper (W7) in this logic is left in to improve NPR latency time.

Table 2-4 Guide for Setting Switches to Select Vector Address

Switch No.	1	2	3	4	5	6	Vector
Bit No.	8	7	6	5	4	3	Address
	X			X	X	X	300
	X			X	X		310
	X			X		X	320
	X			X			330
	X				X	X	340
	X				X		350
	X					X	360
	X						370
		X	X	X	X	X	400
		X		X	X	X	500
			X	X	X	X	600
				X	X	X	700

Notes:
1. X means switch off (open) to produce a logical 0 on the Unibus.
2. Switch numbers are physical positions in switch package 2.

Switch Identification

The vector address selection switches are contained in one DIP switch package (E59). Refer to Figure 2-2 for the location of the package. Only six of the eight switches in the package are used. The correlation between switch numbers and bit numbers is shown in Table 2-4. The ON and OFF positions and the switch numbers are marked on the package. The switches are rocker or slide type and are pushed to the desired position.

CHAPTER 3
REGISTER DESCRIPTIONS AND PROGRAMMING INFORMATION

3.1 INTRODUCTION

This chapter describes the bit assignments for the five DUP11 Registers. It also includes some test programs for use in checking specific aspects of the DUP11 operation.

3.2 DUP11 REGISTERS AND DEVICE ADDRESS SELECTION

The five registers used in the DUP11 are shown in Table 3-1. There is no conflict in assigning the same address (76XXX2) to two registers because the RXDBUF is read-only and the PARCSR is write-only.

Communications devices are assigned floating device addresses in the range 760010 to 764000. Rules for assigning floating device addresses are contained in Chapter 2.

3.3 INTERRUPT VECTORS

The DUP11 generates two vector addresses: receiver interrupts (REQ A) generate vector addresses of the form XX0, and transmitter interrupts (REQ B) generate vector addresses of the form XX4.

Communications devices are assigned floating vector addresses in the range 300–777 (500–534 are reserved). Rules for assigning floating vector addresses are contained in Chapter 2.

3.4 PRIORITY SELECTION

The priority selection (BR level) for receiver and transmitter interrupts is selectable on the module via a plug-in priority selection card. The DUP11 is shipped with a priority 5 card installed that establishes BR5 as the bus request level for interrupts.

3.5 REGISTER BIT ASSIGNMENTS

Bit assignments for the five DUP11 registers are shown in Figure 3-1. Each register is described by showing a bit assignment illustration and an accompanying table that discusses each bit in detail.

Table 3-1 DUP11 Registers

Register Name	Mnemonic	Address	Comments
Receiver Control and Status	RXCSR	76XXX0	Word- and byte-addressable. Read/write.
Receiver Data Buffer	RXDBUF	76XXX2	Word-addressable. Read-only.
Parameter Control and Status	PARCSR	76XXX2	Word-addressable. Write-only.
Transmitter Control and Status	TXCSR	76XXX4	Word- and byte-addressable. Read/write.
Transmitter Data Buffer	TXDBUF	76XXX6	Word- and byte-addressable. Read/write.

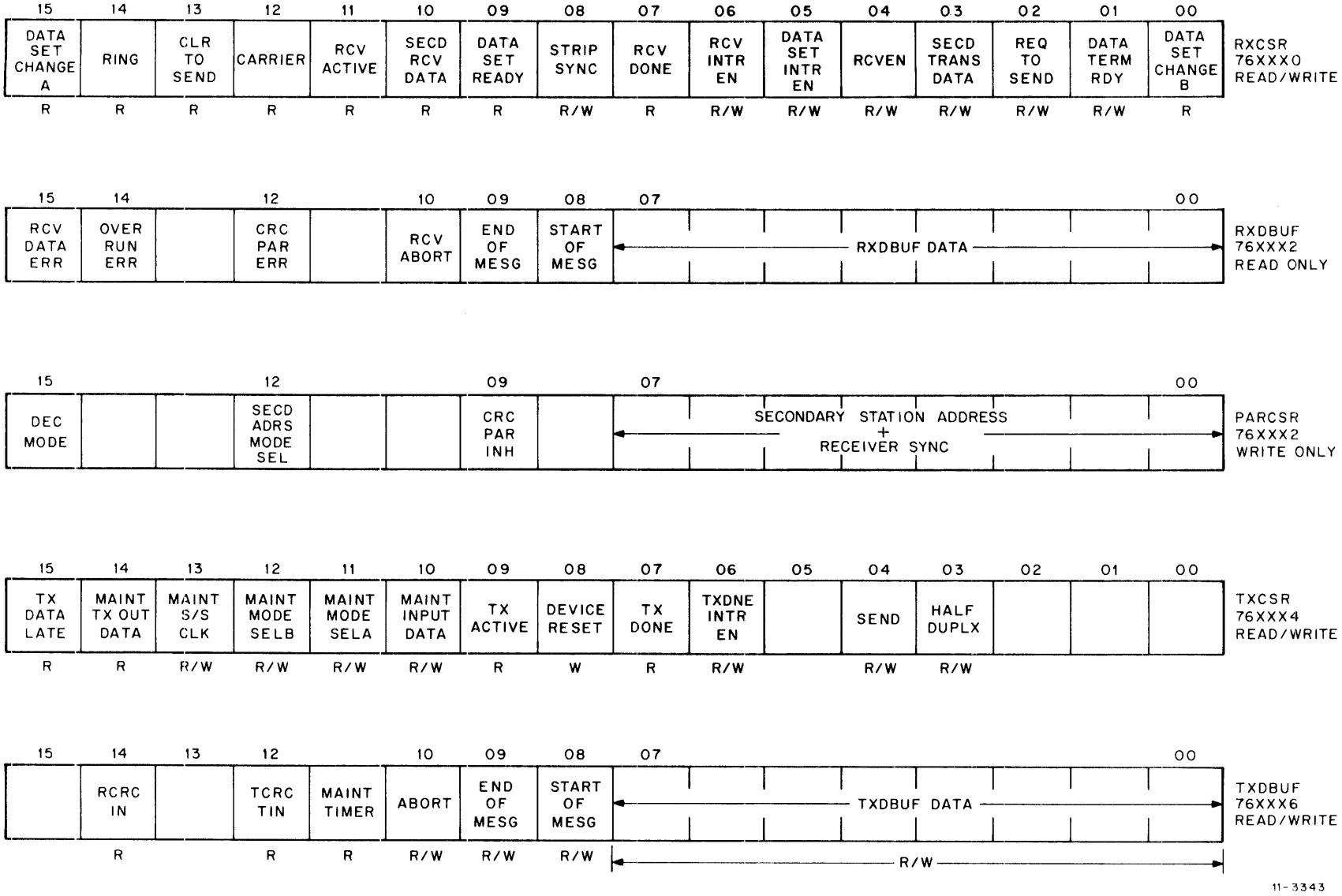


Figure 3-1 DUP11 Register Configurations and Bit Assignments

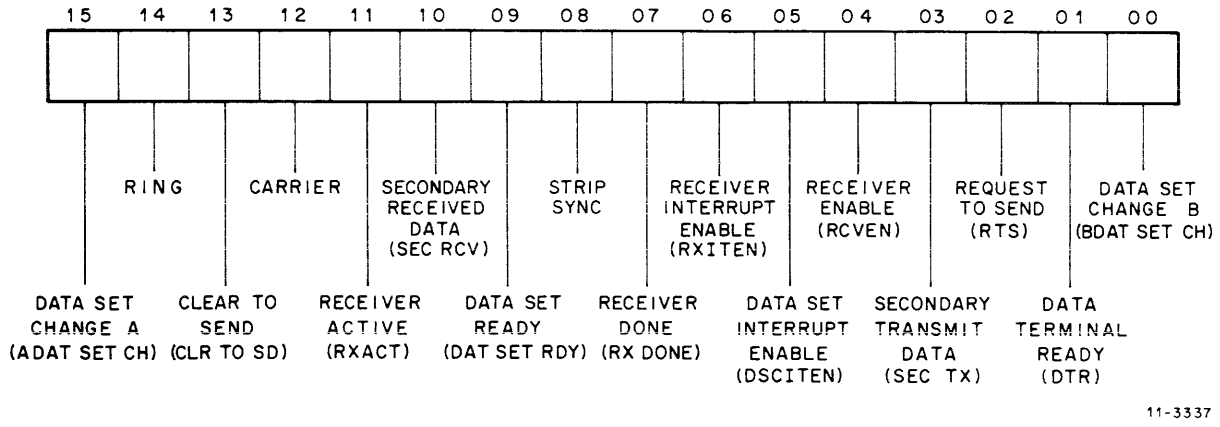


Figure 3-2 Receiver Control and Status Register Format

Table 3-2 Bit Descriptions for Receiver Control and Status Register (RXCSR) (Refer to Figure 3-2)

Bit	Name	Description
15	ADAT SET CH (Data Set Change A)	This bit is set when any of the following transitions occur on the data set control lines. A positive transition on the Ring line greater than 10 ms. Any transition on the Clear to Send line. An optional jumper modification allows this bit to be set by any of the following transitions. This modification is a field installation change supported by diagnostics. Any transitions of the Carrier line Any transitions of the Data Set Ready line Any transitions of the Secondary Received Data line Normally these three transitions cause the Data Set Change B bit to be set in this register. If the jumper modification is made, this bit is disabled. If bit 05 (Data Set Interrupt Enable) of this register is set, the assertion of this bit causes an interrupt to the receiver vector. This bit is program read only and is cleared by INIT, device reset or when the RXCSR is read.
14	RING	This bit reflects the state of the modem Ring line. Any positive transition of this line greater than 10 ms causes the Data Set Change A bit to be set. This bit is program read-only.

Table 3-2 Bit Descriptions for Receiver Control and Status Register (RXCSR) (Cont)
(Refer to Figure 3-2)

Bit	Name	Description
13	CLR TO SD (Clear to Send)	This bit reflects the state of the Clear to Send line of the modem. Any transition of this line causes the Data Set Change A bit to be set. This bit is program read only.
12	CARRIER	This bit is a direct reflection of the modem carrier. Any change in the state of this line causes Data Set Change B to be set unless the data set change jumper modification has been made. (Refer to bit 15 of this register.) This bit is program read only.
11	RXACT (Receiver Active)	The function of this bit is to reflect the current state of the receiver logic in accordance with the selected mode of operation as defined by the contents of the PARCSR. **SDLC or ADCCP Protocol:** In the SDLC or ADCCP mode of operation, (i.e., DEC MODE cleared) this bit is set by the DUP11 logic when the first character of a message frame is being received. CRC computation is performed for all data received, if not inhibited. If the SECD ADRS MS (Secondary Mode Address Select) bit in the PARCSR is cleared, the receiver's operating mode is that of a primary station. In this mode of operation, the RXACT bit stays asserted until the terminating flag character is received. At this time the RXACT bit is cleared and the REOM bit is asserted. The internal receiver CRC register is tested for an error condition and re-initialized at this time in preparation for the next message, if CRC is not inhibited. If an Abort sequence is not received prior to the terminating flag, the RXACT bit is re-asserted when the first data character of the next message is received. The RSOM bit will appear with this character. Any inter-message flag characters are ignored by the receiver. With the Secondary Address Mode Select bit asserted in the PARCSR, the receiving station operates as a secondary station. The major difference between the primary and secondary modes, as far as the DUP11 hardware is concerned, is the character subsequent to the last initial flag character. In secondary mode, this character must match the contents of the low byte of the PARCSR; if not, the RXACT will not be set and the receiver logic searches for the next flag sequence.

Table 3-2 Bit Descriptions for Receiver Control and Status Register (RXCSR) (Cont)
(Refer to Figure 3-2)

Bit	Name	Description
		If extended secondary addressing is implemented in a system, (i.e., 16-bit addresses), the second byte of the address must be recognized by the software. **DDCMP or BISYNC Protocol:** Setting the DEC MODE bit in the PARCSR causes the DUP11 to operate in a manner compatible to the DDCMP or BISYNC family protocols. The low byte of the PARCSR must be loaded with the SYNC character being utilized by the system. This regiter is used only by the receiver logic for comparison purposes. It is not utilized by the transmitter logic. When the RCVEN bit is asserted, the receiver logic searches the received data stream for two consecutive SYNC characters. When two consecutive SYNC characters have been recognized, the receiver is to be considered synchronized to the transmitting station. At this time, all characters subsequent to the two SYNC characters that caused the synchronization are presented to the program (i.e., RXDONE is set) conditional on the character and the state of the STRIP SYNC bit asserted by the program. The RXACT bit is normally asserted when the first character is received subsequent to the synchronization process, unless the STRIP SYNC bit is set. If STRIP SYNC is set, the assertion of RXACT by the DUP11 logic is delayed until the first non-sync character is received. Once RXACT is asserted, the CRC detection logic is activated, provided it is not inhibited in the PARCSR (bit 9) and the STRIP SYNC function is disabled internally. When the completion of the message has been detected, the program must clear the RCVEN bit to re-initialize the receiver. Clearing the RCVEN bit causes the RXACT bit to be cleared also. This bit is program read only and is cleared by INIT, device reset, an off transition of RCVEN, and an ABORT sequence is received in the SDLC or ADCCP mode.
10	SEC RCV (Secondary Received Data)	This bit reflects the state of the Secondary Received Data line from the modem. Any transition on this line causes the Data Set Change B bit to be set unless the data set change jumper modification has been installed. Refer to bit 15 of this register. Used with certain modems only. This bit is program read only.

Table 3-2 Bit Descriptions for Receiver Control and Status Register (RXCSR) (Cont)
(Refer to Figure 3-2)

Bit	Name	Description
9	DAT SET RDY (Data Set Ready)	This bit reflects the state of the Data Set Ready (or interlock) lead from the modem. When asserted, this line indicates that the modem is powered up, is not in test, talk, or dial mode. Any transition of this bit causes the Data Set Change B bit to be asserted unless the data set change jumper modification has been installed. Refer to bit 15 of this register. Program read only.
8	STRIP SYNC	This bit is used only with the DDCMP or BISYNC family protocols. Once the receiver has achieved synchronization, any characters received that match the contents of the low byte of the PARCSR and are contiguous to the initial SYNC characters are not presented to the program (i.e., RXDONE will not be set) if this bit is set. This is useful in stripping off any SYNC characters that are subsequent to the SYNC characters that caused the actual syn chronization of the receiver logic. **NOTE** **This bit must be cleared when the SDLC or ADCCP mode is invoked. Failure to clear this bit disables the receiver logic.** This bit is program read/write and is cleared by INIT or device reset.
7	RXDONE (Receiver Done)	This bit is set by the device when the RXACT bit is set and a character is transferred from the internal receiver shift register into the RXDBUF (receiver data buffer) for the program's acceptance. This bit is also set whenever SYNC characters are received immediately subsequent to the actual synchronization SYNC character, unless the STRIP SYNC bit is set. This applies only to DDCMP or BISYNC modes. In SDLC mode, this bit also is set when an ABORT sequence is received or when the REOM bit is set in the RXDBUF. The RABORT bit in the RXDBUF is asserted when an ABORT sequence is received while the RXACT bit is asserted, or when seven consecutive 1s are detected following a final flag character. In the latter case, RXACT is clear when RXDONE is set and occurs only for the first received ABORT sequence.

Table 3-2 Bit Descriptions for Receiver Control and Status Register (RXCSR) (Cont)
(Refer to Figure 3-2)

Bit	Name	Description
		This bit is program read and is cleared by reading the RXDBUF, INIT, or Device Reset. An interrupt request is generated if the Receiver Done Interrupt Enable bit is set when this bit is asserted.
6	RXITEN (Receiver Interrupt Enable)	When set, this bit allows interrupt requests to be made to the receiver vector if the RXDONE bit is set. All interrupts should be serviced at a processor level equal to or higher than the device Bus Request level which is shipped at level 5. This bit is program read/write and is cleared by INIT or device reset.
5	DSCITEN (Data Set Interrupt Enable)	When set, this bit allows interrupt requests to be made to the receiver vector if the Data Set Change A bit is set. All interrupt requests should be serviced at a processor level equal to or higher than the device interrupt request level which is shipped at level 5. This bit is program read/write and is cleared by INIT or Device Reset.
4	RCVEN (Receiver Enable)	This bit controls the operation of the receiver logic. When initially set, the receiver is enabled to search for synchronization, irrespective of the DUP11's operating mode. Once synchronization has been achieved, the reception of received data and timing is controlled by this bit. Clearing this bit at any time causes all receiver timing and control functions to be reset asynchronously to the modem clock or the data stream currently being received. The RXDONE bit is cleared by the off transition of this bit. This bit is program read/write and is cleared by INIT and device reset.
3	SEC TX (Secondary Transmit Data)	This bit is connected to the Secondary Transmit line of the modem. Supervisory data can be transmitted over this line at a reduced rate. This applies to certain modems only. This bit is program read/write and is optionally cleared by INIT or Device Reset.

Table 3-2 Bit Descriptions for Receiver Control and Status Register (RXCSR) (Cont)
(Refer to Figure 3-2)

Bit	Name	Description
2	RTS (Request to Send)	When set, this bit causes the Request to Send lead to be asserted at the modem interface. This bit is program read/write and is optionally cleared by INIT and Device Reset.
1	DTR (Data Terminal Ready)	When set, this bit causes the Data Terminal Ready lead to be set. For auto dial and manual call origination, it maintains the established call. For auto answer, it allows handshaking in response to a Ring signal. This bit is program read/write and is optionally cleared by INIT or device reset.
0	BDAT SET CH (Data Set Change B)	This bit is asserted when any of the following transitions occur on the respective data set control lines. Any transition of the Carrier line Any transition of the Data Set Ready line Any transition of the Secondary Received Data line Two optional jumper modifications can be made in the field with respect to this bit, the normal jumper configuration is as cited above. Removing the data set change jumper inhibits the setting of the Data Set Change B bit. The Data Set Change B bit is inhibited and the signal transitions cited above are combined with the signal transitions that set Data Set Change A. In this case Data Set Change A is also set by the transitions cited above. These variations are supported by diagnostics. This bit is program read and is cleared by INIT, device reset or by reading the RXCSR.

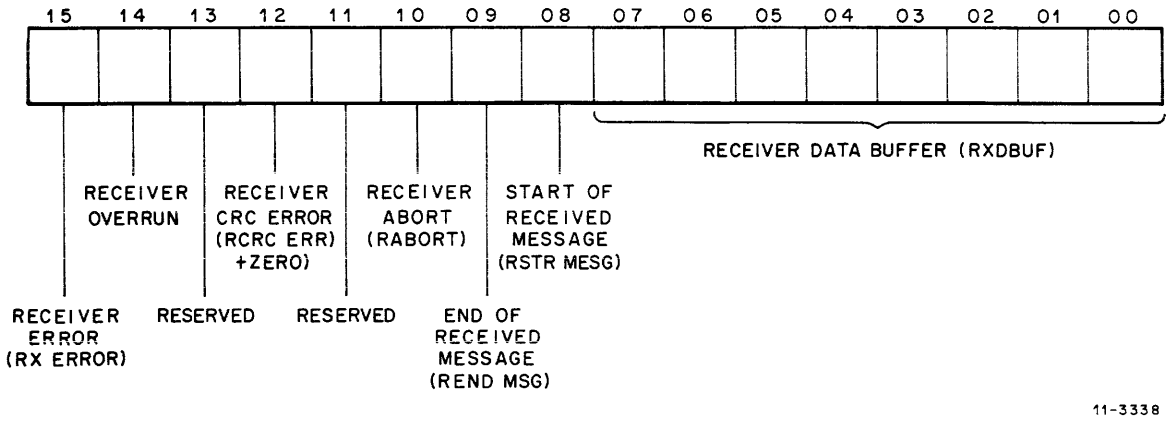


Figure 3-3 Receiver Data Buffer Register Format

Table 3-3 Bit Descriptions for Receiver Data Buffer Register (RXDBUF) (Refer to Figure 3-3)

Bit	Name	Description
15	RX ERROR (Receiver Error)	This bit is set if one of the three error bits in the RXDBUF is set. (Logical OR of bits 14, 12, and 10.) **NOTE** **If the DEC mode bit is set, the setting of bit 12 does not cause this bit to be set.** This bit is program read only and is cleared only when bits 14, 12, or 10 are cleared.
14	REC OVERRUN (Receiver Overrun)	When the receiver logic detects an overrun condition, this bit is set. An overrun is caused primarily by poor program response time. Once the RXDONE bit is set, the program must respond within (1/bps) (8+n) (bit time) sec: if not, overrun occurs. This condition indicates the loss of at least one character. This bit causes the error bit to be set. This bit is set for a minimum of one character time. This bit is cleared within one character time after the overrun condition has been relieved by reading the RXDBUF (i.e., When the next transfer from the internal receiver shift register into the RXDBUF occurs). The Receiver Error bit is set when this bit is set. n = number of inserted zero bits (SDLC or ADCCP only) ($n \leqslant 2$) This bit is program read only and is cleared by INIT, device reset, or by clearing RCVEN.

Table 3-3 Bit Descriptions for Receiver Data Buffer Register (RXDBUF) (Cont)
(Refer to Figure 3-3)

Bit	Name	Description
13		Reserved.
12	RCRC ERR + ZERO (Receiver CRC Error)	When the SDLC or ADCCP mode is selected, this bit is set when the receiver logic detects a CRC error upon termination of a message. The error check is made only when the REOM bit is set in the RXDBUF. When the DDCMP protocol is being used, this bit is set when the internal receiver CRC register is equal to zero. When this bit is set, it stays set for one character time, or until the next transfer is made into the RXDBUF from the internal receiver shift register. The Receiver Error bit is set when this bit is set if the DEC MODE bit is cleared in the PARCSR. This bit is program read and is cleared by INIT, device reset, or by clearing RCVEN.
11		Reserved.
10	RABORT (Receiver Abort)	When an SDLC or ADCCP abort sequence has been received, this bit is set. All receiver timing, internal control and registers are reset. The receiver logic detects ABORT sequences, providing an initial or final flag character has been received. The ABORT sequence is defined as seven or more contiguous 1s. If multiple ABORT sequences are being transmitted the receiver indicates reception of only the first ABORT of the sequence. If the RCVEN bit is left asserted, the receiver resumes searching for a flag synchronization sequence. The RXDONE bit is set when the abort sequence is received and the Receiver Error bit is set also. This bit is cleared by INIT, device reset, clearing RCVEN, or reading the RXDBUF.
9	REND MESG (End of Received Message)	This bit is functional only in SDLC or ADCCP mode. It is set when a terminating flag character is received. This occurs when a flag character is received with the RXACT bit set. When this bit is set, bits 07–00 of this register are invalid. This bit is set for a minimum of one character time. The next transfer from the receiver shift register into the RXDBUF clears this bit. This bit is program read and is cleared by INIT, device reset, or by clearing RCVEN.

Table 3-3 Bit Descriptions for Receiver Data Buffer Register (RXDBUF) (Cont)
(Refer to Figure 3-3)

Bit	Name	Description
8	RSTR MESG (Start of Received Message)	This bit is functional only in SDLC or ADCCP mode. When operating in the primary mode, this bit is set when the first data character is received. When operating in the secondary mode, this bit is set if the character following the last received flag matches the contents of the secondary station address register. This bit is set for a minimum of one character time. The next transfer from the receiver shift register into the RXDBUF clears this bit. This bit is program read and is cleared by INIT, device reset, or by clearing RCVEN.
7–0	RXDBUF (Receiver Data Buffer)	This register contains the data received from the modem. All characters that are presented to the program through this register are eight bits. All characters in this register are right-adjusted with bit 00 being the least significant bit, and bit 07 being the most significant bit. When the End of Received Message bit is set, the data in this register is not valid. If CRC checking is being used, the last two characters that precede the setting of the End of Received Message bit are the CRC check characters that were transmitted. These bits are program read and are cleared by INIT, device reset, RABORT, or by clearing RCVEN.

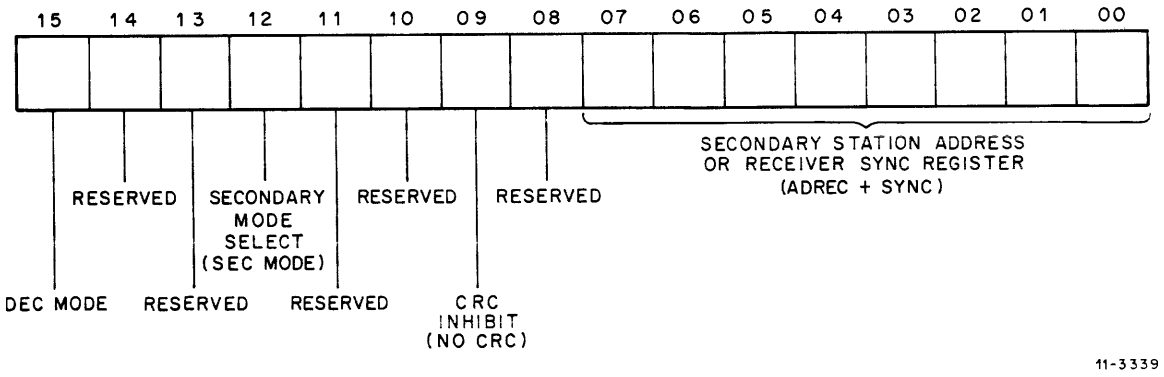


Figure 3-4 Parameter Control and Status Register Format

Table 3-4 Bit Descriptions for Parameter Control and Status Register (PARCSR) (Refer to Figure 3-4)

Bit	Name	Description
		NOTE **The contents of the PARCSR register should be modified only when the transmitter and receiver are in the idle state.**
15	DEC MODE	When this bit is set, the DUP11 logic operates in the manner compatible with the DDCMP or BISYNC protocols. If this bit is clear, the device operates as an SDLC or ADCCP station. As a DDCMP or BISYNC type interface, the receiver logic is synchronized to the transmitting station when two or more consecutive SYNC characters have been recognized. The SYNC character used in the system must be loaded into the low byte of the PARCSR before the RCVEN bit is set. The transmitter logic has no ability to idle SYNC characters from the low byte of the PARCSR. When it is required to transmit SYNC character, the program must load the SYNC character into the TXDBUF. The program should also set the TSOM bit in the TXDBUF when the SYNC character is loaded. This is done to inhibit the inclusion of the SYNC character in the computation of the transmit CRC check character, if CRC is not inhibited. Setting the TSOM also suppresses the setting of the TXDAT LATE (transmit data late) bit. This is useful if the need to idle SYNCs existed. In this case, the program would load the SYNC character into the TXDBUF along with the TSOM bit, and the SYNC character would be transmitted until the program initiated a new operation. During this period, the servicing of the TXDONE could be disregarded without causing the TXDAT LATE error. The bit is program write and is cleared by INIT or device reset.
14, 13		Reserved.
12	SEC MODE (Secondary Mode Select)	Used with SDLC family protocols only. Cleared for DDCMP and BISYNC operation. When this bit is cleared, the device operates as a primary station. All data subsequent to the last received flag character is presented to the program until the termination flag is received. Secondary station operation is in effect when this bit is set. In this mode, only messages that are prefixed with the correct secondary address are presented to the program. The secondary station address must have been loaded into the low byte of the PARCSR before the RCVEN bit was set. The actual address character is not presented to the program in the secondary mode.

Table 3-4 Bit Descriptions for Parameter Control and Status Register (PARCSR) (Cont)
(Refer to Figure 3-4)

Bit	Name	Description
		If extended secondary addresses are used, (i.e., 16-bit address), the first 8 bits of the address can be detected by the hardware. The software must confirm the next 8 bits of address. This bit is program write and is cleared by INIT or device reset.
11, 10		Reserved.
9	NO CRC (CRC Inhibit)	When set, this bit inhibits the transmission of the CRC check character and testing of the CRC error detection logic during reception. This bit is program write and is cleared by INIT or device reset.
8		Reserved.
7–0	ADREC + SYNC (Secondary Station Address Register or Receiver SYNC Register)	This register contains the desired secondary station address when operating in the SDLC or ADCCP secondary mode. The contents of this register is compared to the character received in the shift register (excluding zeros inserted for transparency) subsequent to the last received flag character. If the DEC MODE bit is set, this register must be loaded with the expected SYNC character. This register is used by the receiver logic only. Bit 00 is the least significant and bit 07 is the most significant. These bits are program write and are cleared by INIT or device reset.

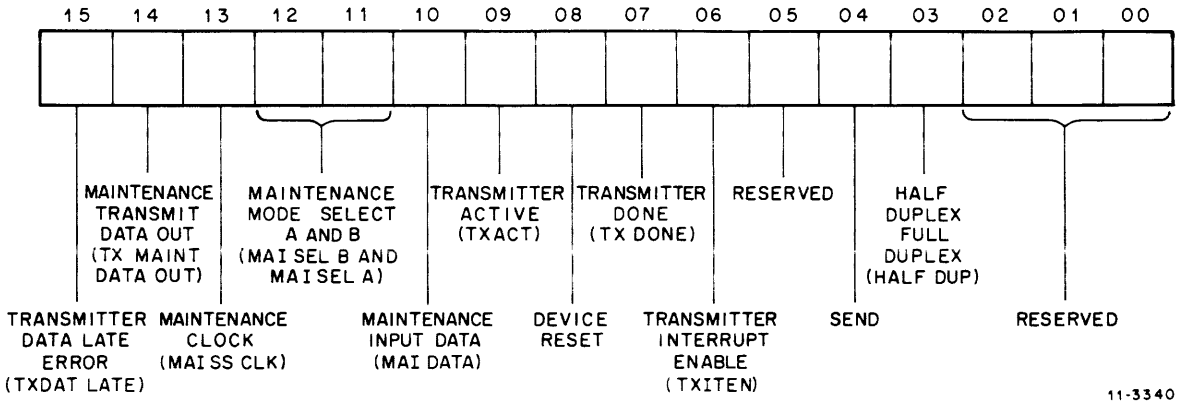


Figure 3-5 Transmitter Control and Status Register Format

Table 3-5 Bit Descriptions for Transmitter Control and Status Register (TXCSR) (Refer to Figure 3-5)

Bit	Name	Description
15	TXDAT LATE (Transmitter Data Late Error)	This bit is set by the transmitter logic when the program response time to the transmitter Done bit is longer than the specified time frame. When this bit is set and the SDLC or ADCCP mode is selected, the transmitter idles abort characters until either a new message is started or the Send bit is cleared. In DDCMP mode, the line is held in the mark state until a new message is initiated. The program must respond to the TXDONE bit by loading the TXDBUF within the following time frame: (1/bps) 8 + N (bit time) N = number of zeros inserted to maintain transparency, SDLC or ADCCP mode only. This bit is program read and is cleared by INIT, device reset, or by setting the TSOM bit.
14	TX MAINT DATA OUT (Maintenance Transmit Data Out)	The function of this bit is to provide a monitoring point of serial output data of the transmitter for the diagnostic program when using the internal maintenance mode. This bit is program read during internal maintenance mode only and is cleared by INIT or device reset.
13	MAI SS CLK (Maintenance Clock)	This bit is used to simulate the transmitter and receiver clock for diagnostic purposes only. Using it in the internal maintenance mode, the diagnostic has the ability to single-step the interface with respect to the handling of data. A 0-to-1 transition of this bit causes the transmitter to transfer one bit of information to the serial line. A 1-to-0 transition of this bit causes the receiver to shift the contents of the receiver shift register and sample the serial input line. This bit must be cleared during the user mode. This bit is program read/write and is cleared by INIT or device reset.

Table 3-5 Bit Descriptions for Transmitter Control and Status Register (TXCSR) (Cont) (Refer to Figure 3-5)

Bit	Name	Description
12, 11	MAI SELB and MAI SELA (Maintenance Mode Select B and A)	These two bits are used together to select the maintenance mode of the interface. **Bit 12 (Select B)** / **Bit 11 (Select A)** 0 0 User mode 0 1 External maintenance mode 1 0 Internal maintenance mode 1 1 System test mode 1 = bit set 0 = bit cleared The user mode is the normal operating mode with all level conversion enabled. The modem is expected to provide all necessary clock signals with a 50/50 duty cycle in accordance with the RS334 standard. The maximum rate is 10 kHz. External maintenance mode provides complete checking of all interface components including level converts and cables. The clocking for this mode is provided by a free-running clock contained within the interface at a 10 kHz ± 20% rate asynchronous to the program. This mode can be used in some circumstances to verify the operation of system's software. When this mode is utilized, the device is disconnected at the modem and a maintenance turn-around connector (H325) is used in place of the modem at the end of the cable. The internal maintenance mode provides a means of analyzing 90 percent of the interface without disconnecting the modem. The interface to the modem cannot be diagnosed when this mode is used (i.e., level converts and cables). Fault isolation is greatly enhanced by this mode since the diagnostic program supplies the data set clocking via the maintenance clock bit. Data being transmitted can be monitored on a bit-by-bit basis at the Maintenance Transmit Data Out bit. The The receiver input can be simulated by either the output of the transmitter or by the Maintenance Input Data bit.

Table 3-5 Bit Descriptions for Transmitter Control and Status Register (TXCSR) (Cont)
(Refer to Figure 3-5)

Bit	Name	Description
		The system's test mode provides asynchronous bus interaction between this device and other devices on the UNIBUS. Data set clocking is simulated by a free-running clock contained on the module at 5 kHz ± 20%. This clocking is asynchronous to the operation of the program. When this mode is used, the device may remain connected to the modem. Receiver and transmitter clocking and data level conversion are inhibited. Modem control signals are not inhibited from being received or transmitted. Transmitted data is internally looped from the transmitter output to the input of the receiver. It is assumed that system test programs will utilize this mode. This bit is program read/write and is cleared by INIT or device reset.
10	MAI DATA (Maintenance Input Data)	When the internal maintenance mode is used, this bit can be used as the receiver serial input. When this bit is set and the maintenance clock bit makes a 1-to-0 transition, a logical 1 is transferred into the receiver shift register. This bit is program read/write and is cleared by INIT or device reset.
9	TXACT (Transmitter Active)	The function of this bit is to indicate the current state of the DUP11 transmitter logic. When the transmitter has been previously in the idle state, (i.e., SEND cleared) and a new message is initiated, this bit is set after a one-half bit time delay, subsequent to the presentation of the first bit to the serial line. When this bit is clear, the transmitter logic is in the idle state and the serial line is held in the mark state. The idle state can be entered by clearing the SEND bit in this same register. The idle state is entered synchronously with the data stream and is also dependent on the DEC MODE, CRC INHIBIT, and TEOM bits. Once the idle state is entered, all transmitter timing and internal control logic is reset.

Table 3-5 Bit Descriptions for Transmitter Control and Status Register (TXCSR) (Cont) (Refer to Figure 3-5)

Bit	Name	Description
		If the SEND bit is cleared and the TEOM bit is not asserted, the character currently being transmitted from the transmitter shift register is completed and the line goes to the mark state. The TXDONE bit is not asserted by the completion of this character. After a one-half bit time delay, the TXACT bit is cleared by the DUP11 hardware. This off transition of TXACT causes the TXDONE bit to be set. The following description assumes that the SEND bit is being cleared within the same character frame as the assertion of TEOM. In the SDLC mode, the transmitter sequence consists of the CRC character (if enabled) and a terminating flag. In the DDCMP mode, the CRC character (if enabled) is transmitted. If these conditions are met, the TXACT bit is cleared 1-1/2 bit times after the last character of the sequence. This transition of the TXACT bit causes TXDONE bit to be set, not the completion of individual characters during the sequence. If DEC MODE is selected and CRC is not inhibited, the character currently being serialized is completed and followed by the automatic transmission of the CRC check character. In this case, the CRC check character is considered the last character of the sequence. If CRC is inhibited, the character currently being serialized is the last character of the sequence. When the DEC MODE bit is cleared and CRC is not inhibited, the character currently being serialized is completed. The CRC check character follows this character. Subsequent to the check character, one terminating flag character is transmitted. This flag character is considered the last character of this sequence. If the CRC INHIBIT bit is asserted, the CRC check character is omitted and the flag character is transmitted subsequent to the character being serialized. The flag character is the last character of this sequence. The one-half bit time delay involved with the assertion of TXDONE in this case is useful in the manipulation of the Clear to Send line. At this time, the Request to Send line can be cleared on most modems without losing the last characer. If the transmitter is left enable (SEND is asserted) and TEOM is also left asserted following the transmission of a sequence, continuous flag characters are transmitted until SEND or TEOM is cleared. The current character being transmitted is completed. This bit is program read and is cleared by INIT or device reset.

Table 3-5 Bit Descriptions for Transmitter Control and Status Register (TXCSR) (Cont)
(Refer to Figure 3-5)

Bit	Name	Description
8	DEVICE RESET	Device reset and the Unibus Initialize signal perform identical functions with respect to the DUP11. When this bit is set, all components of the interface are initialized unless the optional clear jumper is removed. When this jumper is removed, the modem control signals emanating from the device (SEC XMIT, ATS, and DTR) are not affected. This bit is a 1 μs one-shot and self clears. With the optional clear jumper W3 installed, all bits in the interface are cleared with the exception of the transmitter Done bit. Program access should not be made while this bit is set. Both configurations of this jumper are supported by diagnostics. This bit is program write and is cleared by INIT or device reset.
7	TXDONE (Transmitter Done)	This bit is set when the transmitter data buffer is available for a new character. This occurs either as a result of an INIT, device reset, or when a character is transferred from the TXDBUF into the transmit shift register. If the transmitter is entering the idle state, (i.e., SEND is cleared during the current message), the off transition of the TXACT bit causes TXDONE to assert, not the completion of the current character. The TXDONE bit also is set whenever a SYNC, FLAG, or ABORT character has completed transmission, providing the SEND bit is asserted. The TX DATA LATE bit will not assert if a FLAG or ABORT sequence is being transmitted. The transitions of TXDONE can be used to count the number of fill characters transmitted. If this is done, the TXDONE bit can be cleared by reloading the TXDBUF with either TSOM if FLAGS or SYNCS are being used as fill, or TXABORT, if ABORT characters are used. For timing information related to the TXDONE bit and its relationship to the data stream and control bits, refer to the print set. The program must respond to the assertion of this bit within the previously cited time span in order to avoid data under run errors. If the Transmitter Interrupt Enable bit is set, the setting of this bit creates an interrupt request. This bit is program read. It is cleared by writing into the TXDBUF and is set by INIT, device reset, or clearing TXACT.

Table 3-5 Bit Descriptions for Transmitter Control and Status Register (TXCSR) (Cont)
(Refer to Figure 3-5)

Bit	Name	Description
6	TXITEN (Transmitter Interrupt Enable)	When set, this bit allows a program interrupt request to be generated by the TXDONE bit. All interrupt requests should be serviced at a processor level equal to or greater than the device's Bus Request level which is shipped at level 5. This bit is program read/write and is cleared by INIT or device reset.
5		Reserved.
4	SEND	This bit is used to enable the transmitter logic. Once enabled, the transmitter starts transmission of a message when the TSOM bit is detected in the TXDBUF. This bit should remain set until the TEOM bit is loaded into the TXDBUF. IF this bit is cleared at any other time, the current character is finished and the transmitter output goes to a mark hold state. If SEND is cleared while TEOM is still asserted, the current character being transmitted is completed. Following this character, and depending on the protocol being used, any necessary CRC and/or control characters are transmitted. For further information, refer to the TXACT bit in this same register. This bit is program read/write and is cleared by INIT or device reset.
3	HALF DUP (Half Duplex/Full Duplex)	When this bit is set, operation is in half-duplex mode. In half-duplex mode, the receiver is disabled if the SEND bit in the TXCSR is asserted. This bit is read/write and is cleared by INIT or device reset.
2, 1, 0		Reserved.

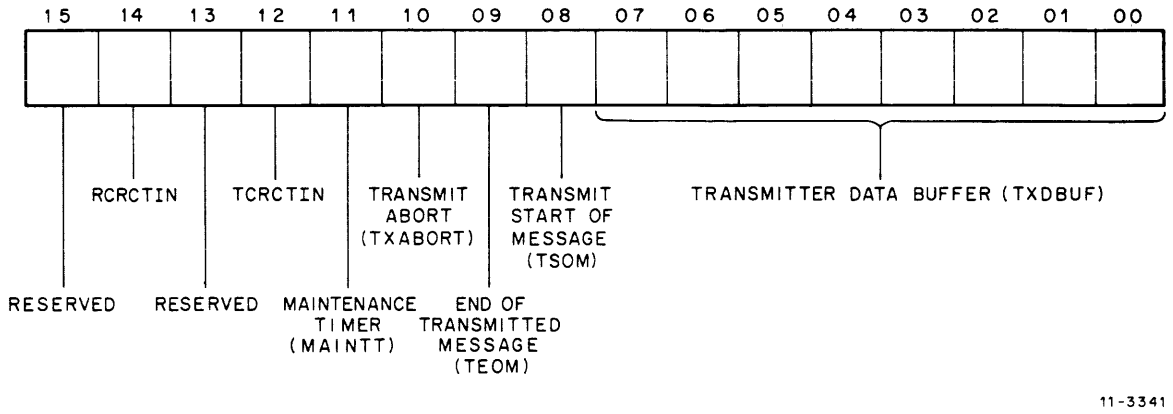


Figure 3-6 Transmitter Data Buffer Register Format

Table 3-6 Bit Descriptions for Transmitter Data Buffer Register (TXDBUF) (Refer to Figure 3-6)

Bit	Name	Description
15		Reserved.
14	RCRCTIN	This bit is provided for maintenance purposes only and is enabled only in internal maintenance mode. The function of this bit is to provide a higher degree of error isolation when diagnosing the receiver CRC register. RCRCTIN is the input to the least significant bit of the receiver CRC register. Refer to Note 1, below, for further information. This bit is program read during the internal maintenance mode only.
13		Reserved.
12	TCRCTIN	This bit is provided for maintenance purposes only and is enabled only in internal maintenance mode. The function of this bit is to provide a higher degree of error isolation when diagnosing the transmitter CRC register. TCRCTIN is the input to the least significant bit of the transmitter CRC register. Refer to Note 1, below, for further information.

Table 3-6 Bit Descriptions for Transmitter Data Buffer Register (TXDBUF) (Refer to Figure 3-6)

Bit	Name	Description
		These bits are program read during the internal maintenance mode only. **NOTE 1** **The true state of these bits is dependent on the protocol being tested.** The RCRCTIN and TCRCTIN bits are XORed inputs to the respective CRC shift register. Data from either the transmitter or receiver data shift registers is presented as a logical 1 being the high state to the XOR gate. The state of data presented to the XOR gate from the most significant bit of either CRC shift register depends on the state of the DEC MODE bit. When DEC MODE is set, a logical 1 output from bit 15 of the respective CRC register is defined as being high. A logical 0 is defined as being low. When DEC MODE is cleared, a logical 1 output from bit 15 of the respective CRC register is defined as being low. A logical 0 is defined as being high.
11	MAINTT (Maintenance Timer)	The function of this bit is to provide a known timing reference for diagnostic programming purposes only. This bit is enabled only in the external or system's test modes. A transition of this bit occurs every 100 μs. The frequency of this clock is 5 kc $\pm$ 20%. This bit is program read in external or system's test mode. It is cleared by INIT or device reset.
10	TXABORT (Transmit Abort)	When this bit is asserted, an Abort sequence is transmitted subsequent to the serialization of the current character, if a character is in process. The SEND bit should be asserted when the Abort sequence is to be transmitted. The TXDONE bit is set at the end of end of each abort character. An abort character is defined as being more than seven contiguous 1 bits. This bit is program read/write and is cleared by INIT and device reset.

Table 3-6 Bit Descriptions for Transmitter Data Buffer Register (TXDBUF) (Cont)
(Refer to Figure 3-6)

Bit	Name	Description
9	TEOM (End of Transmitted Message)	The function of this bit is to terminate the message in progress. How the message is terminated is dependent on the transmitter's mode of operation, as controlled by the information contained in the PARCSR and the state of the SEND bit. If the transmitter is to enter the idle state after the completion of the current sequence, the TEOM bit is set and SEND is cleared by the Program. Refer to the description of the TXACT bit. Termination of a message in DDCMP or BISYNC mode (DEC MODE set) should always cause the transmitter to enter the idle state. Termination of a message in SDLC or ADCCP mode can be achieved in one of two ways. Upon completion, the idle state is entered, or flag characters are idled at completion of the message until the next message is initiated. If upon completion of a message sequence, flags are to be idled, the SEND and TEOM bits should remain set. Flag characters are transmitted until a new message is initiated by clearing the TEOM bit and loading the data. The recommended procedure is to load the new data and clear TEOM in the same operation that accesses the TXDBUF. The contents of the TCRC register always contains all 0s or all 1s after transfer of the CRC character. This bit is program read/write and is cleared by INIT or device reset.
8	TSOM (Transmit Start of Message)	The function of this bit is to initiate the start of a new message if the transmitter is in the Idle state (TXACT = 0). The assertion of this bit causes the internal transmitter CRC register to be re-initialized. The re-initialization of the CRC register and the transfer of the first bit of information occurs within two bit times of the assertion of this bit. If the DEC MODE bit is asserted, the procedure for initiating the start of the message requires that the SYNC character be loaded into the TXDBUF along with the TSOM bit. The character loaded into the buffer is transmitted as the SYNC character until the TSOM bit is cleared. This character is not included as part of the CRC computation. When the TSOM bit is cleared, the SYNC character currently being serialized is finished and is followed by data. All data characters are included in the character computation.

Table 3-6 Bit Descriptions for Transmitter Data Buffer Register (TXDBUF) (Cont)
(Refer to Figure 3-6)

Bit	Name	Description
		If the DEC MODE bit is cleared, the setting of this bit causes the initiation of a message using the SDLC or ADCCP protocols. A flag character is automatically transmitted as long as this bit remains asserted. When data is to be transferred, this bit is cleared by the program and the data is loaded into the TXDBUF. At the completion of the current flag character, the actual transmission of data begins. This bit should not be set when another message is actively being transmitted. Setting this bit also causes the TXDAT LATE bit to be cleared. The TXDONE bit is asserted at the completion of each flag or SYNC character when this bit is asserted. This bit is program read/write and is cleared by INIT or device reset.
7–0	TXDBUF (Transmitter Data Buffer)	This register is loaded with the information to be transmitted. All data is treated as eight bit characters. If the DEC MODE bit is set, the SYNC character to be transmitted must be loaded into this register prior to initiating the synchronization process. The least significant bit of this register is bit 00. Bit 07 is the most significant. These bits are program read/write and are cleared by INIT or device reset.

3.6 TYPICAL TEST PROGRAMS

```
 1
 2
 3
 4                    MAINTENANCE MODE TEST PROGRAM
 5
 6
 7                    THE PURPOSE OF THIS SECTION IS TO PROVIDE THE FIELD SERVICE REP
 8                    WITH A SHORT SERIES OF SOFTWARE TOOLS WHICH WILL ENABLE HIM TO
 9                    TURN A DUP-11 ON AND OFF IN EITHER SDLC OR DEC MODE WITHOUT
10                    USING THE INTERRUPT CIRCUITRY.  THIS SOFTWARE CAN BE USED
11                    WITH OR WITHOUT CRC GENERATION AND,IF IN SDLC MODE,  CAN BE RUN
12                    AS A SECONDARY STATION. THE DATA CAN BE CHANGED OR EXPANDED
13                    EASILY BY THE OPERATOR BY THE ADDITION OF SIMPLE BRANCH
14                    INSTRUCTIONS.  THIS PIECE OF TEST CODE WORKS AS A UNIT ,OR CAN
15                    BE USED IN INDIVIDUAL PARTS TO DEBUG A BOARD AS NEEDED.
16
20
21                    INITIAL START-UP,INCLUDING TURNING ON THE RECEIVER,TRANSMITTER AND
22                    IF NECESSARY, LOADING THE PARCSR IN SDLC MODE.
23
24                    TO RUN AS A SECONDARY STATION OR TO SHUT OFF CRC, INSERT EITHER
25                    OR BOTH OF THE FOLLOWING INSTRUCTIONS AFTER THE CHECK FOR THE
26                    MASTER CLEAR BIT TO GO AWAY:
27                            BIS #10125,@#160052     ;SEC STATION WITH ADRS OF 125
28                            BIS #1000,@#160052      ;CRC INHIBIT
29
```

```
000000  012737  000400  160054  START:  MOV     #400,@#160054   ;DO A MASTER CLR OF DEVICE
000006  032737  000400  160054  1$:     BIT     #400,@#160054   ;WAIT FOR MASTER CLEAR TO GO AWAY
000014  001374                          BNE     1$
000016  052737  004000  160054          BIS     #4000,@#160054  ;TURN ON SYS TEST MODE
000024  052737  000020  160050          BIS     #20,@#160050    ;SET REC ENABLE TO TURN ON THE RECEIVER
000032  052737  000020  160054          BIS     #20,@#160054    ;SET SEND TO TURN ON THE TRANSMITTER


                                AT THIS POINT THE NEXT STEP IS TO SEND AT LEAST ONE FLAG. IF
                                START OF MESSAGE IS LEFT SET,FLAGS WILL BE IDLED
                                FOLLOWING SIXTEEN CONTIGUOUS ONES.


000040  105737  160054          2$:     TSTB    @#160054        ;CHECK FOR DONE
000044  100375                          BPL     2$
000046  052737  000400  160056          BIS     #400,@#160056   ;TURN ON START OF MESSAGE
000054  105737  160054          3$:     TSTB    @#160054        ;WAIT FOR DONE
000060  100375                          BPL     3$


                                LOAD DATA HERE -- IN THIS CASE,SEND TWO CHARACTERS; ANY  MORE
                                WILL CAUSE AN OVERRUN ERROR.


000062  012737  000125  160056          MOV     #125,@#160056   ;TURN OFF START OF MESSAGE AND
                                                                ;LOAD A CHARACTER
                                                                ;;;;;;CHANGE DATA HERE!!!!!!
000070  105737  160054          4$:     TSTB    @#160054
000074  100375                          BPL     4$
000076  012737  000125  160056          MOV     #125,@#160056   ;LOAD A SECOND CHARACTER
000104  105737  160054          5$:     TSTB    @#160054
000110  100375                          BPL     5$


                                AFTER GETTING DONE FOLLOWING THE SECOND DATA CHARACTER,SETTING END OF MESSAGE
                                CAUSES CRC TO BE SENT AND IDLES FLAGS UNTIL START OF MESSAGE IS ASSERTED
                                INSTEAD OF EOM,OR UNTIL SEND IS SHUT OFF--DO NOT SHUT OFF
                                EOM FOR TWO CHARACTER TIMES OR UNTIL AFTER SEND IS CLEARED.


000112  052737  001000  160056          BIS #1000,@#160056      ;SET END OF MESSAGE


                                NOW CHECK THE RECEIVER FOR DATA, TO SEE IF BOTH CHARACTERS AND CRC ARE
                                RECEIVED,AND LOAD THE CSR INTO A MEMORY ADDRESS FOR FUTURE REFERENCE
                                IF NEEDED.


000120  105737  160050          6$:     TSTB    @#160050        ;CHECK FOR RECEIVER DONE
000124  100375                          BPL     6$
000126  013737  160052  000246          MOV     @#160052,@#20$  ;GET THE DATA
000134  105737  160050          7$:     TSTB    @#160050
000140  100375                          BPL     7$
000142  013737  160052  000250          MOV     @#160052,@#20$+2
000150  013737  160050  000256          MOV     @#160050,@#20$+10       ;LOAD RECEIVER CSR
000156  105737  160050          10$:    TSTB    @#160050        ;WAIT FOR FIRST CRC CHARACTER
000162  100375                          BPL     10$
000164  013737  160052  000252          MOV     @#160052,@#20$+4
000172  105737  160050          11$:    TSTB    @#160050
000176  100375                          BPL     11$
000200  013737  160052  000254          MOV     @#160052,@#20$+6
000206  122737  000125  000246          CMPB    #125,@#20$      ;CHECK FOR FIRST DATA CHARACTER
000214  001401                          BEQ     12$             ;BRANCH IF OK
000216  000000                          HALT
000220  122737  000125  000250  12$:    CMPB    #125,@#20$+2    ;CHECK THE NEXT CHARACTER
000226  001401                          BEQ     13$
000230  000000                          HALT


                                CHECK FOR AN ERROR. IF NOT,REPEAT THE SEQUENCE. TO STOP THE PROGRAM,
                                SUBSTITUTE A HALT FOR THE JUMP.
```

```
116
117
118
119  000232  005737  000254          13$:    TST     @#20$+6         ;CHECK FOR ERROR
120  000236  100001                          BPL     14$             ;BRANCH IF NO
121  000240  000000                          HALT                    ;CHECK FOR BIT 12 TO BE SET IN 20$+6
122                                                                  ;IF IT IS ,THAT IS A CRC ERROR IN SDLC MODE
123  000242  000137  000000          14$:    JMP     @#START         ;DO IT OVER
124
125
126  000246  000000                  20$:    .WORD 0
127  000250  000000                          .WORD 0
128  000252  000000                          .WORD 0
129  000254  000000                          .WORD 0
130  000256  000000                          .WORD 0
131
132
133
134
135
136                                  TO USE THE DEVICE IN DEC MODE,YOU MUST SET THE  DEC MODE
137                                  BIT BEFORE LOADING ANY OF THE REGISTERS AFTER THE MASTER CLEAR AT LINE 32.
138                                          BIS #100125,@#160052    ;ENTER DEC MODE AND MAKE SYNC CHAR 125
139                                  FOLLOWING THIS,CHANGE THE INSTRUCTION WHERE TSOM IS SET TO INCLUDE
140                                  THE SYNC CHARACTER.  THIS MUST BE DONE TWICE FOR THE RECEIVER TO
141                                  GO ACTIVE IN DEC MODE.  REPLACE LINE 49 WITH THE FOLLOWING:
142                                          BIS #525,@#160056       ;SEND A SYNC CHARACTER
143                                  15$:    TSTB    @#160054        ;WAIT FOR DONE
144                                          BPL     15$
145                                          BIS     #525,@#160056   ;SEND A SECOND SYNC
146                                  THEN CONTINUE WITH LINE 50.
147
148                                  YOU CAN MODIFY ANY OF THE DATA AND SHORTEN THE PROGRAM BY BRANCHES
149                                  AND JUMPS AS NEEDED TO SOLVE SPECIFIC PROBLEMS.  IF INFORMATION
150                                  ABOUT THE RECEIVER CSR IS REQUIRED,FOR EXAMPLE, CHECKING FOR
151                                  RECEIVER ACTIVE,LOOK IN LOCATION 20$+10 ONCE THE PROGRAM HAS RUN.
152
153
154
155
156
  1
  2
  3
  4
  5
  6
  7                                  INSERT NEW ADDRESSES HERE TO CHANGE THE DEFAULT ADDRESSES
  8
  9
 10  000000  160050                  RXCSR:  160050
 11  000002  160052                  RXDBUF: 160052
 12  000004  160052                  PARCSR: 160052
 13  000006  160054                  TXCSR:  160054
 14  000010  160056                  TXDBUF: 160056
 15  000012  000300                  DUPRVC: 300
 16  000014  000302                  DUPRPS: 302
 17  000016  000304                  DUPTVC: 304
 18  000020  000306                  DUPTPS: 306
 19  000022  000000                  LOOP: 0
 20  000024  000000                  TEMP1:0
 21  000026  000000                  TEMP2:0
 22  000030  000000                  DATA:0
 23
 24
 25
 26                                  THIS ROUTINE IS AN EXAMPLE OF HOW TO MAKE THE
 27                                  DUP11 RUN USING INTERRUPTS IN SDLC MODE.  IT WILL
 28                                  SEND AND RECEIVE 100 OCTAL CHARACTERS AS SPECIFIED
 29                                  BY THE CONTENTS OF BUF.
 30
 31
 32
 33
 34  000032  052777  000400  177746  START:  BIS     #400,@TXCSR
 35  000040  012700  000370                  MOV     #BUF,R0
 36  000044  012701  000570                  MOV     #RBUF,R1
 37  000050  005067  177750                  CLR     TEMP1           ;CLR BYTE COUNT
 38  000054  005067  177746                  CLR     TEMP2           ;DITTO
 39  000060  012767  000340  177710          MOV     #340,PS         ;PS = 7
 40  000066  052777  004000  177712          BIS     #4000,@TXCSR    ;ENTER SYSTEM TEST MODE
 41  000074  004567  000246                  JSR     R5,SETVEC       ;LOAD INTERRUPT VECTORS
 42  000100  000210                          11$                     ;RECEIVER
 43  000102  000274                          12$                     ;TRANSMITER
 44  000104     340     340                  .BYTE   340,340         ;LEVEL
 45  000106  052777  000120  177664          BIS     #120,@RXCSR     ;TURN ON THE RECEIVER
 46  000114  052777  000020  177664  2$:     BIS     #20,@TXCSR      ;TURN ON SEND
```

```
000122  012777  000400  177660          MOV     #400,@TXDBUF    ;TURN ON START OF MESSAGE
000130  105777  177652          3$:     TSTB    @TXCSR          ;WAIT FOR DONE
000134  100375                          BPL     3$              ;BR IF NOT SET
000136  005067  177666                  CLR     DATA
000142  112067  177662                  MOVB    (R0)+,DATA
000146  016777  177656  177634          MOV     DATA,@TXDBUF    ;SEND OUT FIRST DATA CHAR
000154  105267  177644                  INCB    TEMP1           ;UP THE COUNT
000160  052777  000100  177620          BIS     #100,@TXCSR     ;TURN ON TRANSMITTER INT ENABLE
000166  012767  000200  177602          MOV     #200,PS         ;LOWER PROCESOR STATUS
000174  005267  177622          4$:     INC     LOOP            ;DO A WAIT LOOP

000200  001375                          BNE     4$
000202  005067  177614                  CLR     LOOP
000206  000772                          BR      4$




                                ;INTERRUPT SERVICE ROUTINES
                                ;--------------------------



                                THIS SERVICE ROUTINE RECEIVES ALL DATA AND
                                CHECKS FOR END OF MESSAGE AND CRC ERROR

                                ;RECEIVER:
000210  017767  177566  177612  11$:    MOV     @RXDBUF,DATA    ;GET THE REGISTER AND DATA
000216  116721  177606                  MOVB    DATA,(R1)+
000222  105267  177600                  INCB    TEMP2           ;COUNT UP EXPECTED
000226  022767  000100  177572          CMP     #100,TEMP2
000234  001016                          BNE     7$              ;BR IF NO
000236  105777  177536          10$:    TSTB    @RXCSR          ;CHECK FOR DONE
000242  100375                          BPL     10$             ;BR IF NOT YET
000244  032777  001000  177530          BIT     #1000,@RXDBUF   ;CHECK FOR END OF MSG
000252  001001                          BNE     .+4             ;BR IF SET
000254  000000                          HALT
000256  005767  177546                  TST     DATA            ;CHECK FOR CRC ERROR
000262  100001                          BPL     .+4             ;BR IF NO
000264  012716  000032                  MOV     #START,(SP)
000270  000000                          HALT

000272  000002                  7$:     RTI                     ;RETURN



                                THIS SERVICE ROUTINE SEND THE SECOND AND SUBSEQUENT
                                CHARACTERS AND SHUTS DOWN THE  TRANSMITTER AT THE
                                COMPLETION OF TRANSMISSION.

                                ;TRANSMITTER:
000274  112077  177510          12$:    MOVB    (R0)+,@TXDBUF   ;LOAD THE TRANSMITTER BUFFER
000300  105267  177520                  INCB    TEMP1           ;UP THE COUNT
000304  122767  000100  177512          CMPB    #100,TEMP1      ;ARE WE DONE
000312  001014                          BNE     13$             ;BR IF NO
000314  012777  000324  177474          MOV     #22$,@OUPTVC    ;SETUP FOR NEXT PART
000322  000410                          BR      13$             ;LEAVE
000324  012777  001000  177456  22$:    MOV     #1000,@TXDBUF   ;SET END OF MSG
000332  000240                          NOP                     ;STALL
000334  000240                          NOP                     ;DITTO
000336  042777  000120  177442          BIC     #120,@TXCSR ;TURN OFF TRANSMITTER
000344  000002                  13$:    RTI                     ;RETURNS



                                TO CHANGE THE VECTORS,CHANGE THE DEFAULT ADDRESSES


                                ;ROUTINE USED TO SET UP THE INTERRUPT VECTORS
000346  012577  177440          SETVEC: MOV     (R5)+,@DUPRVC
000352  012577  177440                  MOV     (R5)+,@DUPTVC
000356  112577  177432                  MOVB    (R5)+,@DUPRPS
000362  112577  177432                  MOVB    (R5)+,@DUPTPS
000366  000205                          RTS     R5


000370  000100          BUF:    .BLKW 100
000570  000100          RBUF:   .BLKW 100
        000001          .END
```

CHAPTER 4
THEORY OF OPERATION

4.1 INTRODUCTION

This chapter provides a two-level discussion of the DUP11. A functional description, presented first, discusses the DUP11 logic in major functional groups at the block diagram level (Figure 4-1). At this level, the major operating features of the DUP11 are discussed also. The second level of discussion is the detailed description, which covers the complete DUP11 logic at the circuit schematic level, as shown in the DUP11 print set.

4.2 FUNCTIONAL DESCRIPTION

4.2.1 Logic Description

For discussion, the DUP11 logic is divided into nine major sections as shown below.

Title	Paragraph
Registers	4.2.1.1
Device Reset Logic	4.2.1.2
Address Selection Logic	4.2.1.3
Unibus Receivers and Drivers	4.2.1.4
Transmitter Logic	4.2.1.5
Receiver Logic	4.2.1.6
CRC Logic	4.2.1.7
Interrupt Control Logic	4.2.1.8
Data Set Interface Logic	4.2.1.9

4.2.1.1 Registers – The five DUP11 registers are discussed below. They are all 16-bit registers.

Receiver Control and Status Register (RXCSR)
This register contains most of the control and status information pertaining to receiver operation, including the status of the lines to and from the data set. The receiver and data set interrupt enable bits are also contained in this register. The RXCSR is read/write and is word- and byte-addressable.

Receiver Data Buffer Register (RXDBUF)
This register contains the remainder of the receiver status information, including the receiver error flags. Bits 0–7 comprise the 8-bit receiver data buffer that contains the received information to be sent to the PDP-11 system memory. The RXDBUF is read-only and word-addressable.

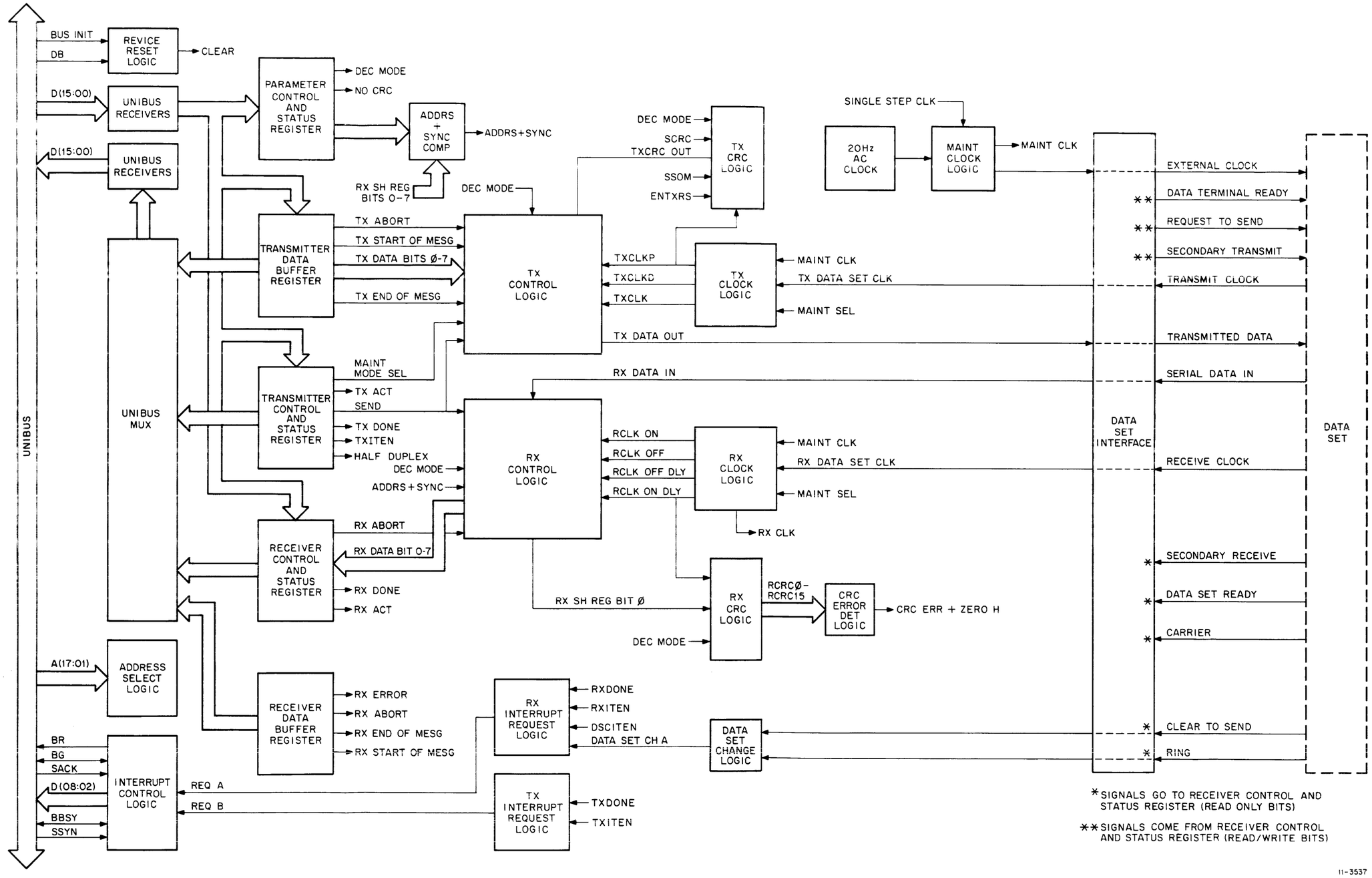


Figure 4-1 DUP11 Simplified Block Diagram

Parameter Control and Status Register (PARCSR)
This register contains the bits that control the DEC mode, secondary address mode, and the enabling of the CRC logic. Its low byte (bits 0–7) contain the 8-bit secondary station address that is used only when the secondary mode is enabled in the SDLC protocol. For DEC MODE operation, this register contains the SYNC character. The PARCSR is write-only and is word-addressable only.

NOTE
The contents of the PARCSR register should be modified only when the transmitter and receiver are in the idle state.

Transmitter Control and Status Register (TXCSR)
This register contains most of the control and status information pertaining to transmitter operation. It also contains the bits to control the DUP11 operation during the maintenance mode. The TXCSR is read/write and is word- and byte-addressable.

Transmitter Data Buffer Register (TXDBUF)
This register contains the remainder of the transmitter control and status information, plus two status bits from the RX and TXCRC registers. Its low byte (bits 0–7) comprise the transmitter data buffer that contains the information to be transmitted. This information comes from the PDP-11 system memory. The TXDBUF is read/write and is word- and byte-addressable.

4.2.1.2 Device Reset Logic – The device reset logic initializes the DUP11 when the Device Reset bit (TXCSR bit 8) is set or the Unibus Initialize signal (INIT H) is asserted. Signal INIT H is gated in to generate the clearing signals. When the Device Reset bit is set, a 1-shot generates a pulse that is converted into the clearing signals.

4.2.1.3 Address Selection Logic – This logic decodes four consecutive addresses to generate control signals that enable the five DUP11 registers. Address 76XXX2 is shared by the RXDBUF and PARCSR registers because the RXDBUF register is read-only and the PARCSR register is write-only.

The basic device address is switch-selectable in the floating device address space (760020–764000). The least significant digit is determined by Unibus address bits A (02:01) and selects the particular register.

The type of operation (DATI, DATO, or DATOB), as determined by Unibus control bits C (01:00), is decoded along with the desired register address to generate the signals to write into or read the contents of the register.

4.2.1.4 Unibus Receivers and Drivers – Multiplexers are used to read the outputs of four registers (RXCSR, RXDBUF, TXCSR, and TXDBUF). The fifth register (PARCSR) is write-only. Because the register outputs are multiplexed, only one set of 16 bus transceivers are required.

Register selection is provided by the multiplexers using Unibus address signals A (02:01) as the select signals. The drivers that put the register contents on the Unibus are enabled by signal DATA→BUS L which is generated by the address selection logic when a DATI (read) operation is requested.

4.2.1.5 Transmitter Logic – The transmitter logic controls the transmission of 8-bit characters with no restrictions on message length or format. The network protocol that is used determines the message format. The logic consists of five functional groups that are described below.

ROMs and Bit Sync Buffer
Three read only memories (ROMs) are the major controlling elements for the transmitter.

The function decode ROM controls the setting of TXDNE and decodes the program inputs which in some cases are synchronized to the data set clock. This information, along with the current state of the logic, determines the next event on a character basis. Six control signals, including the three from the function decode ROM, are stored in the bit sync buffer. After the flag or sync characters have been sent, the buffer is clocked only at the end of a character. This allows the logic to set up for the next character during the present character while not affecting the outputs of the buffer.

The data path control ROM formats the SDLC control characters and controls transmitter data path multiplexing.

The data decode ROM multiplexes the data received from the TX shift register, TXCRC register, and data path control ROM. It also controls the timing of transfers from the TXDBUF register to the TX shift register and the clocking of most of the transmit data path.

Clock Logic
This logic uses an input clock signal to derive a series of clock signals for the transmitter logic. During the user mode, the input is the data set transmitter clock signal TXDAT SET CLK H. During two of the three maintenance modes (system test and internal maintenance), the input comes from the DUP11 internal RC clock as signal MAI ICLK H. In the system test mode, MAI ICLK H is a 5 kHz signal. In the internal maintenance mode, MAI ICLK H is single-stepped using TXCSR bit 13. During the third maintenance mode (external maintenance), RC clock signal MAI EXT CLK (1) H is looped back via the H325 to become TXDAT SET CLK H.

TXDAT FLIP-FLOP AND T1BC COUNTER
The serialized information that is to be transmitted comes from the output of the data decode ROM as signal TXDT H. It is loaded in the TXDAT flip-flop whose output is converted to EIA logic levels and then sent to the modem. The output of the TXDAT flip-flop is also sent to the T1BC counter, which keeps track of the number of consecutive 1s transmitted. In the SDLC mode, this information is used to make protocol decisions when transmitting control characters and to maintain data transparency. Two outputs of the T1BC counter are fed back to the data path control ROM as decision making signals. Counting consecutive 1s allows the control logic to decide whether the character in process is an abort character, flag character, or five consecutive 1s in the data stream. When five consecutive 1s appear in the data stream during a data transfer, the logic stuffs a 0 as the next bit to prevent the sending of a flag configuration (01111110) in the data stream. The receiving station automatically removes the stuffed 0s.

Transmitter Character Serialization Counter (TCSC)
This counter counts the number of bits in each data character, exclusive of stuffed 0s. It also counts the number of bits in a control character. It counts to 16 for the CRC character and a SPACE sequence and to 8 for all others. At the last bit of the character, the counter generates a pulse (TCSC MAX H) that synchronizes the current action of the transmitter data path to the program interface.

Shift Register
The shift register is loaded in parallel with the information (8-bit character) to be transmitted. This includes data, SDLC control characters, and DDCMP sync characters. The register serializes the character starting with the LSB. The serial data (TXSER OUT H) from the shift register goes to the data decode ROM.

4.2.1.6 Receiver Logic - The receiver logic controls the reception of 8-bit characters in accordance with the network protocol.

In the SDLC or ADCCP protocols, the receiver may operate as a primary or secondary station. In the primary mode, all received messages are presented to the program. In the secondary mode, only those messages prefixed with the applicable secondary station address are presented to the program. This address is stored in the low byte of the PARCSR register.

In the DDCMP or BISYNC protocols, the low byte of the PARCSR register must be loaded with the SYNC character being used in the network.

ROMs and RX Control Flags
Two read only memories (ROMs) are the major controlling elements for the receiver.

The receiver decode ROM enables the receiver shift register and interprets the received data according to the network protocol. In the SDLC mode, it recognizes flag characters, data characters, abort characters, and stuffed zeros. In the DDCMP mode, it recognizes sync characters and data characters.

The receiver function ROM asserts signal RSR→RXDBUF H, which sets RXDONE and loads the contents of the shift register into the RXDBUF register. It also sets FRM, RXACT, and MESG ACT. These three signals are called receiver control flags and are stored in the RX control flags flip-flop.

Clock Logic
This logic uses an input clock signal to derive a series of clock signals for the receiver logic. During user mode, the input is the data set receiver clock signal RXDAT SET CLK H. During two of the three maintenance modes (system test and internal maintenance), the input comes from the DUP11 internal RC clock as signal MAI ICLK H. In the system test mode, MAI ICLK H is a 5 kHz signal. In the internal maintenance mode, MAI ICLK H is single-stepped using TXCSR bit 13.

During the third maintenance mode (external maintenance), RC clock signal MAI EXT CLK (1) H is looped back to become RX DATSET CLK H.

Enable R1BC Flip-Flop and R1BC Counter
The EN R1BC flip-flop looks at the received data before it is shifted into the R1BC counter. During SDLC protocol operation, the switch from the idle state to the active state must start with a 0 to signal the first bit of the SDLC flag character (01111110). The EN R1BC flip-flop enables the R1BC counter only if this action occurs. This action locks the EN R1BC flip-flop in the set state. It remains set until it is directly cleared when an ABORT character is received, or the program clears the RCVEN bit in the RXCSR.

The R1BC counter counts consecutive 1s and is cleared when a received 0 is detected. The state of some of its outputs are used as inputs to the decode ROM to recognize a flag character (six consecutive 1s), an abort character (eight consecutive 1s), or a stuffed 0 (five consecutive 1s). This counter provides no function during DDCMP protocol operation.

Receiver Character Serialization Counter (RCSC)
The RCSC counter counts the number of bits (8) in the data and control characters, exclusive of stuffed 0s. At the last bit, it generates signal RCSC MAX H which goes to the function ROM. This indicates that the assembled character should be loaded into the RXDBUF data register and RXDONE should be set to tell the program that a received character is ready for transfer to the PDP-11 system memory.

Shift Register
The shift register is loaded in serial form with 8-bit received characters. The actual input is the LSB (R1BC0 H) of the R1BC counter. The 8-bit parallel output of the shift register is loaded into the RXDBUF register (bits 7–0) and then on to the PDP-11 system memory via the multiplexed Unibus selectors.

4.2.1.7 CRC Logic – The CRC logic is the circuit implementation of the cyclic redundancy checking method of encoding and decoding messages for error detection. It consists of a transmitter CRC register, receiver CRC register, and error detection logic. The SDLC protocol uses code CCITT and the DDCMP protocol uses code CRC-16. Both codes generate 16-bit CRC check characters.

In a typical operation, the sending station's TXCRC register looks at the information being transmitted and accumulates a CRC check character. This character is transmitted at the end of the message. The receiving station's RXCRC register looks at the received message plus the CRC check character.

At this point, the message is errorless if the RXCRC register contains octal 016417 for the SDLC protocol or 0 for the DDCMP protocol.

Flag signal CRC ERROR + ZERO H (RXDBUF bit 12) tells the program if the message is in error. For the SDLC protocol, if the RXCRC does not contain 016417 after reception of the CRC check character, CRC ERROR + ZERO H is asserted to denote that the message contains one or more errors.

For the DDCMP protocol, the flag is used in a different way. The DUP11 does not count characters so the arrival of the CRC check character cannot be predicted. It is left for the program to check the RXCRC register at the correct time. Therefore, the error detection logic asserts signal CRC ERROR + ZERO H any time the RXCRC register reads zero at the end of a character. If the register is checked at the correct time and it does not read zero, CRC ERROR + ZERO H is not asserted, which indicates an error.

4.2.1.8 Interrupt Control Logic – The interrupt control logic is functionally equivalent to the BR half of the M7821 Interrupt Control module. The DUP11 communicates with the PDP-11 system via interrupts – it does not use NPR transactions.

The interrupt control logic responds to interrupt requests from the transmitter and receiver. Signal REQ A H is the receiver request signal and is associated with vector address XXO. Signal REQ B H is the transmitter request signal and is associated with vector address XX4. Both requests are at level BR5; however, if they occur simultaneously, the receiver request is honored first.

Signal REQ B H is generated when TXDONE is set and the transmit interrupt enable bit (TXITEN) is asserted.

Signal REQ A H is generated when either of two pairs of signals are asserted, provided the RXCSR is not being read. In one case, REQ A H is generated when RXDONE is set and the receive interrupt enable bit (RXITEN) is asserted. In the other case, REQ A H is generated when ADAT SET CH is set and the data set interrupt enable bit (DSC ITEN) is asserted.

The interrupt control logic contains switches for selecting the vector address.

4.2.1.9 Data Set Interface Logic – The data set interface logic allows program control of the initiation of communications with the data set and allows monitoring of status signals from the modem. It also provides logic level conversion for all signals between the DUP11 and the data set. The data set uses EIA logic levels and the DUP11 uses TTL logic levels.

The control lines to the data set, the status lines from the data set, and two flag bits (Data Set Change A and Data Set Change B) are part of the RXCSR and can be read by the program. Normally, the Data Set Change A flag is set by a transition on the following data set status lines: Clear to Send and Ring. The Data Set Change A flag can request a receiver interrupt if the data set interrupt enable bit is asserted. The Data Set Change B flag is set by a transition of the following data set status lines: Carrier, Data Set Ready, and Secondary Received Data. A change can be made using jumpers to inhibit the operation of the Data Set Change B flag and to allow all of the above mentioned status lines to set the Data Set Change A flag.

Logic is also provided to control the flow of transmitted data during the user mode and to control the external clock during the external maintenance mode.

4.2.2 Major Operating Features

4.2.2.1 Introduction – This paragraph discusses the major operating features of the DUP11 at the functional level. The discussion is divided into four sections as shown below.

Title	Par. No.
Modem Control	4.2.2.2
Transmitter Section	4.2.2.3
Receiver Section	4.2.2.4
Interrupt Handling	4.2.2.5

This discussion assumes that the DUP11 is operating in the user mode. The maintenance modes, which are used to service the DUP11, are described in Chapter 5, Maintenance.

4.2.2.2 Modem Control – The modem control of the DUP11 is provided by the program. The initiation of communications with the modem is achieved by asserting the appropriate bits in the receiver status and control register (RXCSR).

The modem control and status lines are monitored along with a flag to indicate a change in any line's state since the last time it was monitored by the program. Using these indicators, the program must determine when it can transmit data. Once this has been established, the transmitter is enabled and transmission begins when the first character is loaded into the data buffer.

In some systems, this handshaking is not necessary. The transmitter is simply enabled and transmission starts when the program sets the TSOM bit in the transmit data buffer register (TXDBUF). The receiver may start searching for sychronization without first having been rung.

The flow of data is not interlocked with signals received from the modem. When modem control is being used in a system, the program must monitor the received modem control signals contained in the RXCSR register.

The secondary receive and transmit leads are also included in the modem control section. While these leads have no function in the Bell 201, 208 and 209 modems, they can be redefined for other user purposes at installation time.

All clock signals used in conjunction with data received from or transmitted to the modem must emanate from the modem and be in accordance with EIA 232-C at a rate $\leqslant$ 10 kHz. No external clock to the modem is supplied by this interface. Maintenance mode clocking used to facilitate checkout and diagnostic engineering is provided. This clocking is under program control and is intended to be used only when the DUP11 is being serviced.

4.2.2.3 Transmitter Section

The transmitter section provides the following functions:

1. Buffers and serializes parallel data.
2. Generates CRC check characters.
3. Creates transparent data stream for SDLC and ADCCP protocols.
4. Transmits flag and abort sequences and leading zero sequences.

The transmitter section is enabled when the program asserts the SEND bit in the TXCSR transmitter control and status register (TXCSR). The actual state of the transmitter logic is indicated by the TXACT bit in this register.

A character or control bit may be loaded into the TXDBUF whenever the TXDONE bit is asserted. This occurs after an initialize pulse, device reset by the program, or when the logic completes transmission of a character. This bit is cleared when a character is loaded into the TXDBUF.

Transmit Operation Under SDLC Family Protocol Discipline

Assuming that the transmitter section is in the idle state (TXACT = 0) and is enabled by the SEND bit, transmission of a message sequence begins when the TSOM bit in the TXDBUF is detected. Upon detecting this bit, the transmitter automatically transmits the initial flag character.

When starting from the idle state, the first bit of the flag character is delayed for a period equal to two bit times. The TXDONE and TXACT bits are asserted when this first bit is transferred to the serial line. At this point, the first data character may be loaded into the TXDBUF. If the TSOM bit is still asserted at the completion of the flag character currently in progress, another flag character is transmitted. Flags are sent until TSOM is cleared.

The transmitter goes into the transparent state immediately following the transmission of the last initial flag characters and also begins the accumulation of the CRC check character, providing that the CRC inhibit bit is cleared in the PARCSR. When the transmitter is in the transparent state, the logic automatically inserts 0 following five consecutive 1s, to preserve the properties of the flag and abort characters.

Termination of a message is accomplished by asserting the TEOM bit in the TXDBUF. The character currently being serialized in the transmitter shift register is completed. If CRC is not inhibited, the computer CRC check character is automatically transmitted and followed by a terminating flag character. If CRC is inhibited, the terminating flag character follows the character currently being serialized. The TEOM bit must remain asserted until transmission of the terminating flag character starts. If the SEND bit is asserted when TEOM is asserted, the start of the transmission of the flag is identified by the next transition of TXDONE. These events occur simultaneously.

At this point, the program has three options:

1. Clear SEND, which allows the flag character currently being transmitted to be finished. At the end of the flag there is a delay of one and one-half bit times, after which the transmitter enters the idle state.

2. Leave TEOM asserted, which allows continuous flag characters to be transmitted until option 1 or 3 is executed. In this option, Data Late errors do not occur. The number of flag characters sent can be determined by counting the transitions (set state) of TXDONE. This bit is set when the TXDBUF register is loaded. In this case, the program should keep TEOM set.

3. Initiate another data transfer by clearing TEOM and loading the data byte into the TXDBUF register. If TEOM is cleared during transmission of the first terminating flag, only one flag separates the SDLC frames. If TXACT is asserted, it is not necessary to set TSOM to start a subsequent message. The recommended procedure is to load the new data and clear TEOM in the same operation that accesses the TXDBUF. The TSOM bit should be used to initiate a message only when TXACT is cleared.

   If the SEND bit is cleared between the time that the TEOM bit is asserted and the transmission of the terminating flag has started, the transition of TXDONE is deferred until the transmitter returns to the idle state.

This delay of one-half a bit time in most cases ensures the integrity of the last character before attempting to turn the line around in half-duplex situations; however, this is modem-dependent and each modem manual should be referenced for applicability.

If SEND is cleared and no transmitter control bits are set, the current character is transmitted and the transmitter is shut down after a one and one-half bit delay.

Because of system or timing restriction, it may be necessary in some instances to transmit a given number of ABORT characters subsequent to the last flag characters. This can be accomplished by asserting the TXABORT bit at the appropriate time. The SEND bit must remain asserted for this operation.

The TXABORT bit can be asserted when the last required flag character has begun transmission. The earliest possible time that the TXABORT bit can be asserted (for this purpose) is immediately after the second assertion of the TXDONE bit subsequent to the setting of the TEOM bit.

The first assertion of the TXDONE bit subsequent to the setting of the TEOM bit occurs when the serialization of CRC check information begins. It is necessary to clear the TXDONE bit before the end of the transmission of the CRC character so that the second transition of the TXDONE bit can occur. This transition marks the start of the transmission of the terminating flag character. This second transition can be created by again setting the TEOM bit. The TXABORT bit can be set when the TXDONE bit asserts, marking the beginning of the terminating flag character.

Transmit Operation Under DDCMP or BISYNC Protocol Discipline

Assuming that the transmitter is enabled by the SEND bit in the TXCSR, transmission starts when the TSOM bit is set in the TXDBUF by the program. When the TSOM bit is asserted, the SYNC character being used must be present in the lower byte of the TXDBUF. All transmitted SYNC characters must be loaded into the TXDBUF. The TSOM bit must remain asserted until the start of the last SYNC character. The TXDONE bit is asserted at the completion of each SYNC character. When it is necessary to count the number of transmitted SYNC characters, the TXDONE bit can be cleared by again setting the TSOM bit. This allows the next transition of TXDONE at the end of the character.

When the last SYNC character is transmitted, the TSOM bit is cleared and the first character of data is entered into the TXDBUF. This character and all subsequent data characters are included in the transmitter's CRC computation.

For protocols such as BISYNC, the CRC feature of this device must usually be inhibited. Protocols such as DDCMP can make efficient use of the CRC logic. These protocols are characterized by the fact that no special control characters are embedded within the message that are not included as part of the CRC computation.

The accumulated CRC check character is transmitted subsequent to the assertion of the TEOM bit. When the character currently being transmitted is complete, the CRC check character is sent next if the TEOM bit is asserted. The TXDONE bit is asserted at the start of the CRC character transmission. This can be ignored or cleared by again setting the TEOM bit.

The DUP11 does not provide the feature of automatically idling SYNC characters without program intervention.

Data must be presented to the TXDBUF as 8-bit characters. The message length is not restricted.

Transitter CRC Character Generation
To enable the CRC logic, the program must clear NO CRC (PARCSR bit 9). Two CRC codes are used: the SDLC family protocols use code CCITT which is represented by polynomial $X^{16} + X^{12} + X^5 + 1$; the DDCMP family protocols use code CRC-16 which is represented by polynomial $X^{16} + X^{15} + X^2 + 1$. With CRC enabled, the accumulated CRC check character is a result of all data loaded by the program into the TXDBUF. With the SDLC and ADCCP protocols, 0s that are inserted into the message to maintain transparency are not included as part of the calculation.

The transmitter CRC register is initialized (cleared) when TSOM is asserted, which is internally synchronized. When initialized, the register reads all 0s in the DDCMP mode and all 1s in the SDLC mode. Initializing the register to all 1s provides protection against obliterating leading zeros which may not be detected if the register is zero. In the DDCMP mode, the CRC check character is transmitted as is; in the SDLC mode, it is complemented before it is transmitted.

Transmitter Latency
The specifications of the SDLC, ADCCP and DDCMP protocols require that the flow of data be contiguous; that is, no intra-message fill characters are allowed.

Since the DUP11 is double-buffered, one character time (and in some cases more) is allowed before data late errors are encountered. The data late condition is indicated whenever the TXDBUF is not serviced within the appropriate response time before assertion of the TXDONE bit.

This time can be expressed as follows:

(1/baud rate) 8 + n (secs), where n = number of inserted zeros (applicable only for SDLC family protocols)

$n \leqslant 2$

When the current character has been transmitted, the absence of valid data in the TXDBUF causes the TXDAT LATE bit to be asserted in the TXCSR, unless the TEOM bit was asserted in the TXDBUF. This indication suggests re-transmission of the message. When this occurs, the transmitter automatically transmits abort characters in the SDLC or ADCCP modes until a new message is presented to the transmitter, providing the SEND bit is asserted. In the DDCMP or BISYNC modes, the line is held in the mark hold state. Assertion of the TSOM bit clears the TXDAT LATE bit.

4.2.2.4 Receiver Section

The receiver section provides the following functions:

1. Buffers and converts received serial data to parallel data.
2. Interprets transparent data stream in SDLC or ADCCP protocols.
3. Recognizes flag and abort sequences.
4. Recognizes secondary station addresses (SDLC mode) and SYNC characters (DDCMP mode).
5. Detects CRC errors.

The receiver section is capable of operating as either a secondary or primary station when the SDLC and ADCCP protocols are selected. This is controlled by the state of the SEC MODE bit in the PARCSR. When operating as a primary station, all received messages are presented to the program. In the secondary mode, only messages that are prefixed with a secondary station address that matches the contents of the low byte of the PARCSR are presented to the program.

If the DDCMP or BISYNC mode of operation is selected, the low byte of the PARCSR must be loaded with the SYNC character being used by the system. All data received is handled as 8-bit characters.

The receiver logic is controlled by the RCVEN bit in RXCSR. The state of the receiver is indicated by the RXACT bit in the same register.

Receive Operation Under SDLC Family Protocol Discipline

Once the initialization and any necessary modem handshaking have been completed, the RCVEN bit can be set. When the RCVEN bit is asserted, the receive logic searches for initial flag characters. When operating in the primary mode, all data received subsequent to the last initial flag character is presented to the program. The first character is accompanied by the RSOM in the RXDBUF. When operating in the secondary mode, the character subsequent to the last flag character is compared to the contents of the low byte of the PARCSR (any bits inserted for transparency are stripped prior to performing the compare). If the comparison is not true, the search for a flag is reiterated. When this comparison is true, the RXACT bit is asserted in the RXCSR. The RSOM bit in the RXDBUF is set to indicate the beginning of a new message. The received address character is not presented to the program; the character subsequent to it is the first character to be presented to the program, along with the RSOM bit. When this character is transferred to the receiver data buffer, the RXDONE bit is also asserted. Any character subsequent to this causes the RXDONE bit to be asserted with the receiver interrupt enable bit asserted; the assertion of the RXDONE bit creates an interrupt request. When the program accesses the RXDBUF, the RXDONE bit is cleared.

When the terminating flag character is received, the REOM bit is asserted in the RXDBUF and the RXDONE bit is asserted in the RXCSR. The low-order byte of the RXDBUF is invalid when the REOM bit is set.

If CRC checking is not inhibited, an error would be indicated only when the REOM bit is asserted. The check is performed on all data received, beginning with the secondary station address, up until the first check character is received. When CRC checking is implemented, the last two bytes of information received by the program are the CRC check characters.

Messages in progress can be aborted if the sending station transmits an ABORT sequence. When the receiver detects the ABORT sequence, the RABORT bit in the RDBUF is set along with the RXDONE bit in the RXCSR. The receiver logic detects one ABORT sequence after receiving an initial or final flag character.

All data received is handled and presented to the program in 8-bit characters.

Transparency is maintained at the receiver by searching the data stream for zeros inserted by the transmitter and removing them.

Receive Operation Under DDCMP or BISYNC Protocol Discipline
The DDCMP and BISYNC family of protocols are also handled by this device. In most BISYNC applications, the software overhead is increased. This occurs because of the extra amount of character recognition or processing of message header information required. Also, in some cases, the CRC feature of the DUP11 may have to be inhibited because of control characters embedded within the data stream.

The low byte of the PARCSR must be loaded with the SYNC character being utilized by the system before the receiver is enabled. This character is loaded into the PARCSR and is used only by the receiver logic for comparison with data being received. This register is not used by the transmitter logic.

Once the initialization and any necessary modem handshaking have taken place, the program may assert the RCVEN bit. With the RCVEN bit asserted, the first operation of the receiver logic is to search the data stream for two consecutive SYNC characters. When two consecutive SYNCs have been recognized, the receiver logic is considered synchronized and any subsequent information is assembled as 8-bit characters. When an 8-bit character is assembled, it is transferred into the RXDBUF, and the RXDONE bit is asserted conditional on the character being received, and the state of the STRIP SYNC bit and RXACT bit. If the program has asserted the STRIP SYNC bit, the character received matches the contents of the PARCSR low byte, and the RXACT bit has not been set by the logic, the RXDONE bit is not asserted.

When the logic has located the first non-SYNC character, the RXACT is asserted by the logic. This character and all subsequent characters are included in the receiver's CRC computation. At this point, the function af the STRIP SYNC bit is internally disabled.

The RCVEN bit in the RXCSR should be left asserted for the entire message and cleared at the end of the message. Clearing this bit re-initializes the receiver logic.

Receiver CRC Character Checking
CRC error detection is performed by the receiver logic if the NO CRC bit is cleared. The method of detecting errors and the polynomial used vary according to the mode of operation as controlled by the DEC MODE bit in the PARCSR.

If the DEC MODE bit is cleared, the CCITT polynomial ($X^{16} + X^{12} + X^5 + 1$) is used and the logic operates in a manner compatible with the SDLC and ADCCP protocols. The receiver CRC error detection logic is effectivelv set to all ones when the RXACT bit is cleared.

The contents of the receiver CRC register are tested when a terminating flag is received in SDLC or ADCCP mode. The register is tested for the following bit pattern: LSB 1 111 000 010 111 000 MSB. The absence of this bit pattern causes assertion of the RCRC ERR + ZERO bit in the RXDBUF. The REOM bit is also asserted at this time. This bit pattern is the result of all data received between the last initial flag character and the terminating flag, excluding inter-message flags and zeros inserted for transparency.

The last two bytes (16 bits) of data presented to the program through the RXDBUF comprise the received CRC check character. The data received in the RXDBUF when the REOM bit is asserted is meaningless and should be disregarded.

DDCMP-compatible operation is enabled by asserting the DEC MODE bit. In this mode, the CRC 16 ($X^{16} + X^{15} + X^2 + 1$) polynomial is used to generate the receiver check character. The receiver CRC register is initialized to all zeros when the RXACT bit is cleared.

Once the receiver has been synchronized, the data received and presented to the program is included in the computation of the check character. This occurs when the RXACT bit is asserted. Characters received during the reception of a message (after RXACT is asserted) are included in the CRC computation even if they match the contents of the PARCSR. The function of STRIP SYNC is internally disabled when RXACT is asserted.

The RCRC ERR + ZERO bit is asserted in this mode when during the reception of a message the CRC register is equal to zero coincidental with the end of the current character.

The program should only test the RCRC ERR + ZERO bit when the expected number of bytes including CRC information have been received. It is entirely possible that during the reception of a message this bit may be asserted without having received the actual CRC check character. Normally the RCRC ERR + ZERO bit is presented to the program along with the second CRC check character.

Receiver Latency
The program must respond to the RXDONE bit within a specified time frame in order to avoid overrun errors.

If the program has not read the contents of the RXDBUF within this time frame, the OVRUN FRR bit in the RXDBUF is set. The contents of the data buffer contain the last received character. This error suggests retransmission of the message.

Because this device is double-buffered, the time lag in which the program must respond is as long as a full character time and can be expressed as the following:

(1/baud rate) 8 + n (secs), where n = number of inserted zeros (SDLC family protocols only)

$n \leqslant 2$

Interrupt Requests
In the RXCSR there are two interrupt request enable bits; in the TXCSR, there is one. These bits can be used to selectively allow interrupt requests that occur asynchronous to the operation of the program.

The Data Set Interrupt Enable bit (DSCITEN) allows interrupt requests to be generated on the receiver interrupt vector if the DATA SET CHANGE A bit is asserted. The Receiver Interrupt Enable bit (RXITEN) also allows interrupts to be generated on this same vector if the RXDONE bit is asserted. If both interrupt conditions exist simultaneously on the receiver vector, the interrupt requests occur back-to-back and there is no fixed scheme in which the requests should be serviced.

There is only one interrupt request made on the transmitter interrupt vector. This request is made if the Transmitter Interrupt Enable bit (TXITEN) is set and the logic asserts the TXDONE bit.

All interrupt requests should be serviced at a processor status level equal to or above that of the device interrupt priority level. If simultaneous interrupt requests are generated on both the receiver and transmitter vectors, the receiver request is honored first.

Half-Duplex Operation
The program can select half-duplex operation by asserting the HALF-DUPLEX bit in the TXCSR. In this mode of operation, the receiver logic does not transfer data. It is completely disabled if the SEND bit is asserted in the TXCSR. All other characteristics of the interface are maintained.

4.3 DETAILED DESCRIPTION

4.3.1 Introduction
The detailed description of the DUP11 logic is divided into 10 major sections.

Title	Paragraph
Registers	4.3.2
Device Reset Logic	4.3.3
Address Selection Logic	4.3.4
Unibus Receivers and Multiplexed Unibus Drivers	4.3.5
Transmitter Logic	4.3.6
Receiver Logic	4.3.7
CRC Logic	4.3.8
Interrupt Control Logic	4.3.9
Data Set Interface Logic	4.3.9
Typical Operations	4.3.11

4.3.2 Registers
This discussion covers the operation of the five DUP11 registers listed below:

Register	Paragraph
Receiver Control and Status (RXCSR)	4.3.2.1
Receiver Data Buffer (RXDBUF)	4.3.2.2
Parameter Control and Status (PARCSR)	4.3.2.3
Transmitter Control and Status (TXCSR)	4.3.2.4
Transmitter Data Buffer (TXDBUF)	4.3.2.5

Only the register operation is discussed in these paragraphs. A functional description of each bit is contained in Chapter 3.

4.3.2.1 Receiver Control and Status Register (RXCSR) – The receiver control and status register (RXCSR) is a 16-bit register that is word- and byte-addressable. All bits are used; however, several bits are an integral part of the data set interface logic and are described in Paragraph 4.3.10. These bits are: 0–3, 9, 10, and 12–15. The remainder of the bits (4–8 and 11) are discussed in this section.

Bit 11 – RXACT (Logic Sheet BSI3)
The state of the RXACT bit is stored in one section of the 74175 quad flip-flop which contains the receiver control flags. The D input of the RXACT flip-flop (EN RXACT H) comes from the receiver function ROM. It is clocked by RCLK OFF DLY L which is a 50 ns negative pulse that occurs once each bit time. It is cleared directly by RCV CLR L which goes low when the abort bit is set [RABORT (1) H goes high] or the receiver enable bit is cleared [RCVEN (0) H goes high]. The RXACT bit is read-only.

Bit 8 - STRIP SYNC (Logic Sheet BSI1)
This program read/write bit is used with DDCMP or BISYNC protocols only. The state of the STRIP SYNC bit is stored in a 7474 flip-flop. The D input is DB 08 H which is the output of the bus receiver associated with Unibus data line D08. It is clocked by LD RXCSR HB H which is generated by the register selection logic when the RXCSR is written into (word or high byte). It is directly cleared by CLR L.

Bit 7 - RXDONE (Logic Sheet BSI3)
This bit is hardware write/program read. The state of the RXDONE bit is stored in a 7474 flip-flop. The preset input (pin 10) is connected to the output of a 7400 NAND gate (low level presets flip-flop). One input of the NAND gate is RCLK OFF H which a 250 ns positive pulse that occurs once each bit time. The other input is an ORed function of RABORT (1) H and RSR → RX DBUF H. When clock pulse RCLK OFF H occurs with either of these signals asserted, the RXDONE flip-flop is directly set. Signal RSR → RX DBUF H is asserted by the receiver function ROM to load the RX data buffer. Signal RABORT (1) H is asserted when an abort character is received.

Bits 6, 5, and 4 (Logic Sheet BSI1)
These bits are identified as follows:

Bit	Name
6	Receiver Interrupt Enable (RXITEN)
5	Data Set Interrupt Enable (DSCITEN)
4	Receiver Enable (RCVEN)

They are contained in a 74175 quad flip-flop. The common clock for these bits is LD RXCSR LB H which is generated by the register selection logic when the RXCSR is written into (word or low byte). They are directly cleared by CLRL. All three bits are program read/write.

4.3.2.2 Receiver Data Buffer Register (RXDBUF) - The receiver data buffer register (RXDBUF) is a 16-bit read-only register. Bits 13 and 11 are not used.

Bit 15 - RX ERROR (Logic Sheet BSI3)
The detection of any one of three receiver errors causes assertion of the receiver error flag (RX ERROR H). The errors are: receiver overrun (REC OVERUN), receiver CRC error (RXCRC ERR), and receiver abort (RABORT) which are bits 14, 12, and 10 of this register, respectively. These bits are contained in flip-flops which are set when clocked with the error detected. Two gates are used to provide a logical OR function of these three signals which is inverted to generate RX ERROR H. Signal RXCRC ERR (1) H is ANDed with DEC MODE (0) H in a 7408 AND gate whose output goes to one input of a 3-input 7427 NOR gate. This arrangement prevents the RXCRC ERR bit from generating RX ERROR H when the DEC MODE bit is set (DUP11 operating in DDCMP protocol). The other two inputs to the 7427 gate are REC OVERUN (1) H and RABORT (1) H. Assertion of any one of these bits drives the output of the 7427 gate low. This signal is inverted by a 7404 inverter to assert RX ERROR H.

Bits 14, 12, 9, and 8 (Logic Sheet BSI3)
These bits are identified as follows:

Bit	Name
14	OVERRUN
12	Receive CRC Error (RXCRC ERR)
9	End of Received Message (REND MESG)
8	Start of Received Message (RSTR MESG)

They are contained in a 74174 hex D-type flip-flop. The other two sections of this flip-flop contain bits 0 and 1 of this register. The common clock signal for these bits is LD RXBF H which is the result of ANDing RSR → RXDBUF H and RCLK OFF H. Signal RSR → XDBUF H is generated by the receiver function ROM. Signal RCLK OFF H is a 250 ns positive pulse that is generated once per bit time. All bits are cleared by CLR L which goes low when the receiver enable (RCVEN) bit is cleared.

Bit 10 - RABORT (Logic Sheet BSI3)
This bit is set by the hardware when an abort sequence is received (SDLC and ADCCP protocols only).

The state of the RABORT bit is stored in a 7474 flip-flop. Its preset input (pin 4) is connected to the output of a 7400 NAND gate (preset enabled when low). The inputs to this gate are ADRS + SYNC RCVD H and FLG RCVD H which are outputs of the receiver decode ROM. They are asserted simultaneously only when an abort function is in progress.

The D input is connected to ground. Whenever the RXDBUF register is read, RD RXBUF L is generated by the register selection logic. The positive transition of this signal clocks the RABORT flip-flop and clears it. This bit is directly cleared by RCVEN (0) L which is low when the receiver enable bit is cleared.

Bits 7-0 (Logic Sheet BSI3)
These eight bits are the receiver data buffer and contain the data received from the modem and are loaded in parallel from the output of the receiver shift register. Bits 02-07 are stored in a 74174 hex flip-flop and bits 00 and 01 are stored in two sections of the 74174 hex flip-flop that also contains bits 14, 12, 9, and 8 of this register. They are clocked by LD RXBUF H and cleared by CLR L.

4.3.2.3 Parameter Control and Status Register – The parameter control and status register (PARCSR) is a 16-bit register that is word-addressable. Bits 14, 13, 11, 10, and 8 are not used.

NOTE
The parameter control and status register is write only and should be accessed only when the transmitter and receiver are in the idle state.

Bits 15, 12, and 9 (Logic Sheet BSI1)
These bits are identified as follows:

Bit	Name
15	DEC MODE
12	Secondary Mode Select (SEC MODE)
9	CRC Inhibit (NO CRC)

They are contained in a 74174 hex flip-flop. The common clock for these bits is LD PARCSR H which is generated by the register selection logic when the PARCSR is written into. They are directly cleared by CLR L.

Bits 7-0 (Logic Sheet BSI1)
These bits serve as the secondary station address register when operating in the secondary mode in the SDLC or ADCCP protocols. They serve as the sync register in the DDCMP protocol (Figure 4-2).

Bits 05-00 are contained in a 74174 hex flip-flop. Bits 07 and 06 are contained in two sections of the 74174 flip-flop that also contains bits 15, 12, and 9. All bits are clocked by LD PARCSR H and directly cleared by CLR L. The D inputs are signals DB 07-DB 00 which are outputs of the bus

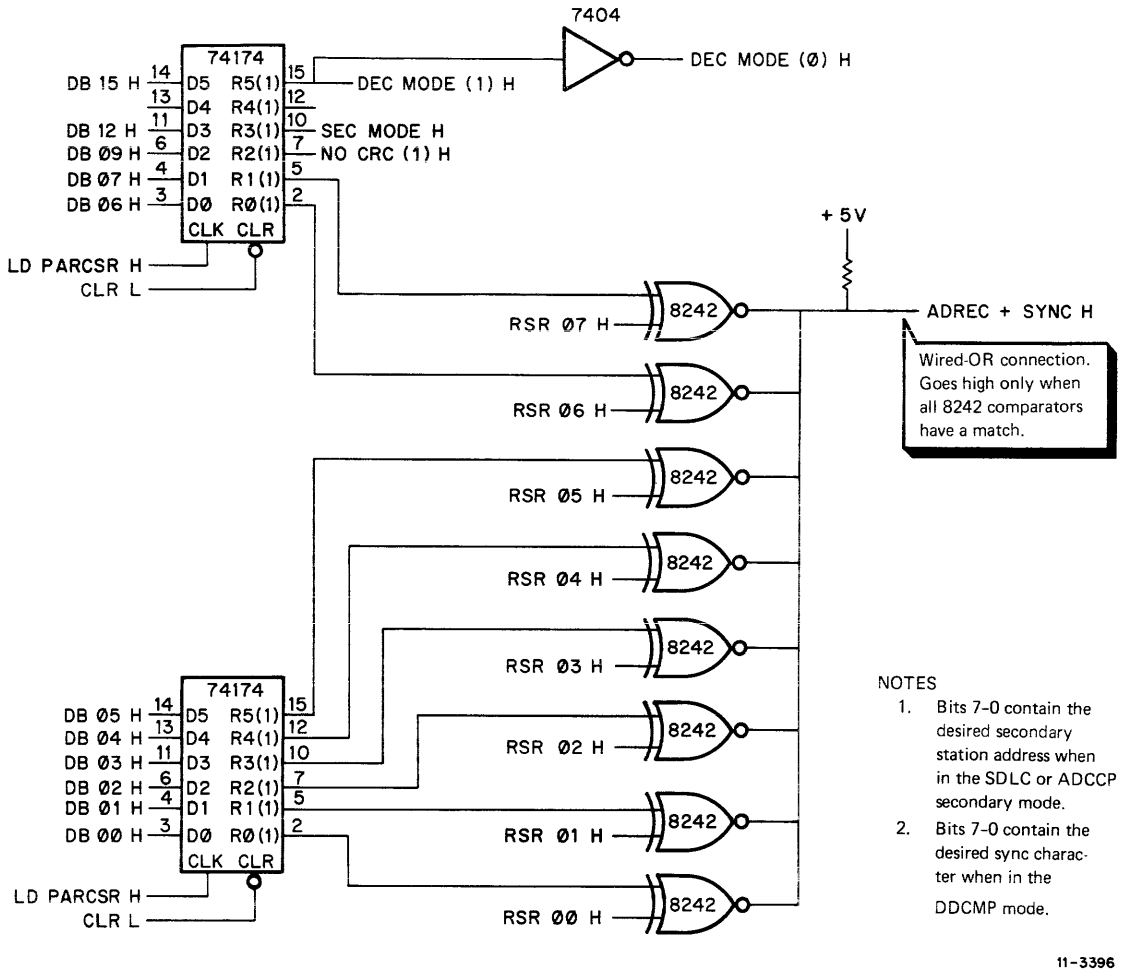


Figure 4-2 PARCSR Bits 7-0

receivers associated with Unibus data lines D (07-00). In the SDLC or ADCCP protocols, these bits are loaded with the desired secondary station address. In the DDCMP protocol, they are loaded with the desired sync character.

Each flip-flop output goes to one input of an 8242 exclusive-NOR gate that is used as a comparator. The other input of each 8242 gate goes to the associated output of the receiver shift register (signals RSR 07 H-RSR 00 H). When the 8242 inputs match, the output goes high. All 8242 outputs are joined together to form a wired-OR connection that is called ADREC + SYNC H. This signal goes high when the correct secondary address is received (SDLC or ADCCP protocols) or the desired sync character is received (DDCMP protocol).

4.3.2.4 Transmitter Control and Status Register (TXCSR) - The transmitter control and status register (TXCSR) is a 16-bit register that is word- and byte-addressable. Bits 0, 1, 2, and 5 are not used. Bit 0 is available as a read/write bit - it can be written into by using the reserved section of a 74175 quad flip-flop. It can be read by using the reserved input in the multiplexed bus driver logic.

Bit 15 - TXDAT LATE (Logic Sheet BSI2)
During transmission, if TXDONE is still set at the end of the current data character, the hardware sets the TXDAT LATE bit. This indicates that the TXDBUF register has not been loaded with the next character.

The state of the TXDAT LATE bit is stored in a 7474 flip-flop (Figure 4-3). The D input of this flip-flop is connected to the output of a 7408 AND gate. One input of the AND gate is TX DONE (1) H which is asserted if TX DONE is set. The other input comes from a 7402 NOR gate. Its inputs are SABORT (1) H and SFLG (1) H which are both low when the TXABORT bit (TXDBUF bit 10) is cleared and a flag bit is not being transmitted. With both of these signals low, the other input of the 7408 AND gate is high also which puts a high on the D input of the TXDAT LATE flip-flop. The TXDAT LATE flip-flop cannot be set if an SDLC flag or abort sequence is being transmitted. When the last bit of the current character is counted, the TCSC counter asserts TCSC MAX H which clocks the TXDAT LATE flip-flop and sets it. The 1 output of this flip-flop goes to the multiplexed bus drivers and is read by the program. Acknowledgement of the TXDAT LATE flag usually results in retransmission of the message. Depending on the protocol being used, the DUP11 transmits abort characters or idle characters until retransmission is started. When the transmit start of message bit (TSOM) is set, signal TSOM (1) H is asserted. It is inverted by a 7402 NOR gate and directly clears the TXDAT LATE flip-flop. The other input to this NOR gate is CLR H which is asserted during the device reset or bus initialization operations to clear the DUP11 registers.

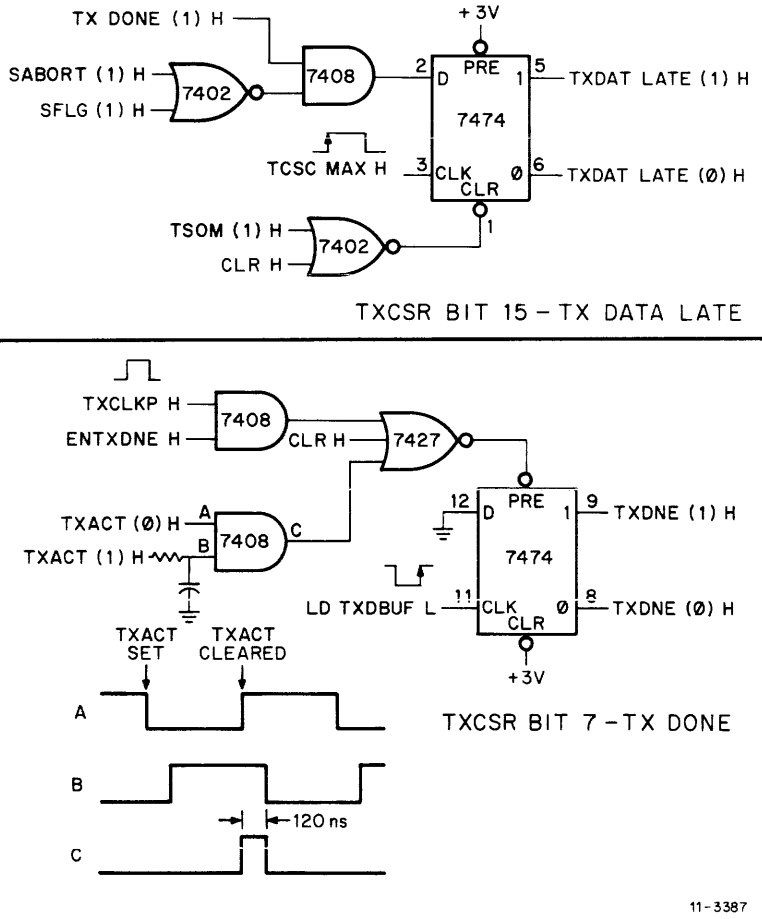


Figure 4-3 TXCSR Bits 15 and 7

Bit 14 - MAINT TX DATA OUT (Logic Sheet BSI3)
This bit is read-only and is discussed under the Internal Clock Logic heading in Paragraph 4.3.6.2.

Bits 13-10 (Logic Sheet BSI1)
These bits are identified as follows:

Bit	Name
13	Maintenance Clock (MAI SS CLK)
12, 11	Maintenance Mode Select A and B (MAI SELA and MAI SELB)
10	Maintenance Input Data (MAI DATA)

They are contained in a 74175 quad D-type flip-flop and are program read/write. The common clock signal for these bits is LD TXCSR HB H. They are cleared by CLR L.

Bit 9 - TX ACTIVE (Logic Sheet BSI2)
Bit 9 is hardware write/program read. The state of the TX ACTIVE bit is stored in a 7474 flip-flop. Its D input is connected to SENTXAC (1) H which comes from the bit sync buffer. The state of this signal is controlled by ENTXAC H from the function decode ROM. Signal TXCLK H clocks the TX ACTIVE flip-flop and sets it when SENTXAC (1) H is asserted. It is directly cleared by CLR L.

Bit 8 - DEVICE RESET (Logic Sheet BSI1)
This bit is program write only and is discussed in Paragraph 4.3.3, Device Reset Logic.

Bit 7 - TXDONE (Logic Sheet BSI2)
This bit is hardware write/program read. The transmitter logic sets TXDONE when the TXDBUF register is available for the next character.

The state of the TXDONE bit is stored in a 7474 flip-flop (Figure 4-3). The CLR input (pin 13) is inhibited by connecting it to +3 V. The D input is connected to ground so it can be cleared only by the positive transition of the clock signal LD TXDBUF L. The preset input (pin 10) is connected to the output of a 7427 3-input NOR gate. One input is connected to CLR H; therefore, during DUP11 initialization the TXDONE flip-flop is set.

The second input of the 7427 NOR gate is connected to the output of a 7408 AND gate. This output represents the ANDing of TXCLKP H and ENTXDNE H. During transmission of flag and data characters, the function decode ROM asserts ENTXDNE H to indicate the availability of the TXDBUF register. Signal TXCLKP H goes high for 300 ns during each bit time and if ENTXDNE H is asserted during this period the TXDONE flip-flop is set.

The TXDONE flip-flop is set one-half bit time after the last bit of the terminating flag character, provided the SEND bit is cleared. This indicates the end of the message and the transmitter goes to the idle state (send MARKs). The transmitter logic clears the the TXACTIVE flip-flop. The 1 and 0 outputs of the TXACTIVE flip-flop are sent to a pulse generator (7408 AND gate and RC delay network) to generate a 120 ns positive pulse when TXACTIVE is cleared. This pulse is the third input of the 7427 NOR gate. It is inverted and directly sets the TXDONE flip-flop.

Bits 6, 4, and 3 (Logic Sheet BSI1)
These bits are identified as follows:

Bit	Name
6	Transmitter Interrupt Enable (TXINTEN)
4	SEND
3	Half Duplex/Full Duplex (HALF DUP)
0	Not Assigned

They are contained in a 74175 quad-type flip-flop and are program read/write. The fourth section of this flip-flop is reserved for bit 0 (signal DB 00 H is connected to its D input). The common clock for these bits is LD TXCSR LB H. They are cleared by CLR L.

4.3.2.5 Transmitter Data Buffer Register (TXDBUF) - The transmitter data buffer register (TXDBUF) is a 16-bit register that is word- and byte-addressable. Bits 13 and 15 are not used.

Bits 14 and 12 (Logic Sheet 4)
These bits are used only during the internal mode to diagnose the receiver and transmitter CRC registers. Bits 14 (RCRCT IN) and 12 (TCRCT IN) represent the input to the LSB position of the RXCRC and TXCRC registers, respectively. Each one of these signals is picked off the output of a 7408 AND gate that is enabled by MIA EN H (Figure 4-4). This signal is high only during the internal maintenance mode.

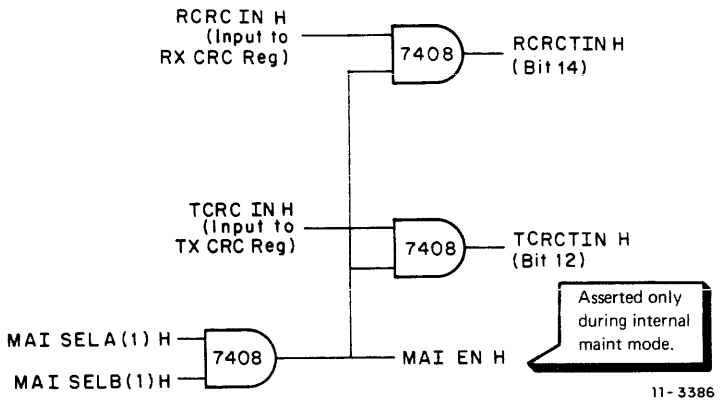


Figure 4-4 TXDBUF Register Bits 12 and 14

Bit 11 - MAINT TIMER (Logic Sheet BSI1)
This signal is a timing reference that can be used during the external or system test modes only. It is read-only during these modes and is represented by signal MAINTT (1) H. It is picked off the second least significant bit of the 74191 counter in the internal clock logic. Its frequency is 5 kHz ± 20%; therefore, it produces a transition every 100 μs.

Bits 10, 9, and 8 (Logic Sheet BSI2)
These bits are identified as follows:

Bit	Name
10	Transmit Abort (TXABORT)
9	End of Transmitted Message (TEOM)
8	Transmit Start of Message (TSOM)

They are contained in a 74175 quad D-type flip-flop and are program read/write. The fourth section of the flip-flop is not used. The common clock signal for these bits is LD TXDBUF HB H. They are cleared by CLR L.

Bits 7–0 (Logic Sheet BSI2)
These bits are loaded with the information that is to be transmitted as an 8-bit character and are program read/write. Bits 0–5 are stored in a 74174 hex D-type flip-flop. Bits 6 and 7 are stored in another 74174 flip-flop. The other four sections of this flip-flop are unused. The inputs (DB 07 H–DB 00 H) come from the bus receivers associated with Unibus data lines D(07:00) and are loaded by the positive transition of signal LD TXDBUF LB H which clocks the 74174 flip-flops. They are cleared by CLR L. The outputs (TBUF 07 H–TBUF 00 H) are sent in parallel to the transmitter shift register for serialization.

4.3.3 Device Reset Logic (Logic Sheet BSI1)
Bit 8 of the TXCSR controls the device reset function. When it is set, all bits in the DUP11 are cleared with the exception of the TXDONE bit. A jumper is provided in the data set interface logic that can be removed to prevent the following bits from being cleared:

RXCSR BIT 1 - Data Terminal Ready (DTR)
RXCSR BIT 2 - Request to Send (RTS)
RXCSR BIT 3 - Secondary Transmit Data (EIA SEC XMIT)

The device reset logic is shown in Figure 4-5. The signals that actually perform the reset function are CLR L and CLR H. They are generated by setting the DEVICE RESET bit or asserting the Unibus initializing signal (BUS INIT L).

The heart of the device reset logic is the 74123 one-shot. When triggered, it generates complementary 1 μs pulses. Its clear input (pin 3) is inhibited by connecting it to +3 V. Input pin 1 of the one-shot is connected to the inversion of signal DB08 H which is the output of the bus receiver connected to Unibus data line D08 (logic sheet BSI8). When the program desires to clear the DUP11, it sets bit 8 (DEVICE RESET) in the TXCSR. Setting this bit asserts signal DB08 H which puts a low on pin 1 of the one-shot. This qualifies the one-shot so that a positive transition on its other input (pin 2) triggers the one-shot. This transition is obtained when the TXCSR is loaded. A 7400 NAND gate and a 7404 inverter form the input logic to pin 2. Prior to decoding the address of the TXCSR for the loading operation, signal DADR SEL L is high. This signal comes from the addre ss selection logic (sheet BSI6) and goes to one input of the 7400 NAND gate. When the TXCSR address is decoded, signal LD TXCSR HB H is asserted at the other NAND gate input. The output of this gate goes low and is inverted. The resulting positive transition triggers the one-shot. Approximately 50 ns after LD TXCSR HB H is asserted, DADR SEL L goes low which corresponds with the assertion of BUS SSYN by the address selection logic. The 0 output of the one-shot generates a 1 μs negative pulse that goes to one input of the first 8881 driver. The other input of this driver is high because BUS INIT L is not asserted; therefore the pulse is inverted by the 8881 driver and appears as CLR H. This pulse is inverted by the second 8881 driver to become CLR L. The CLR H and CLR L pulses are sent throughout the DUP11 to perform the reset function.

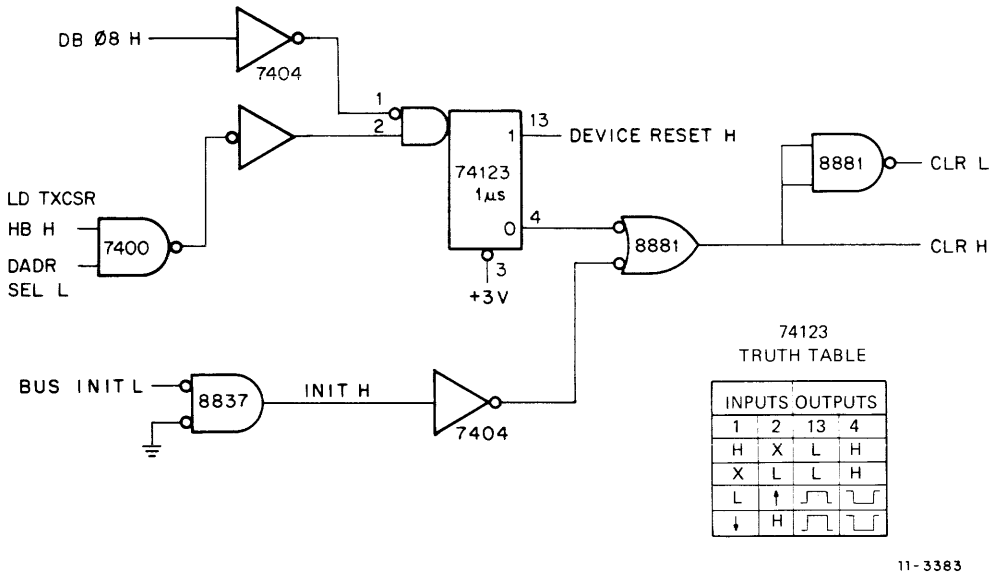


Figure 4-5 Device Reset Logic

If the program uses the Unibus initialize signal to perform the reset function, signal INIT H is asserted and generates CLR L and CLR H using the same 8881 drivers.

4.3.4 Address Selection Logic (Logic Sheet BSI6)

4.3.4.1 Address Assignments and Format - Each DUP11 is assigned four consecutive addresses that are decoded to generate control signals to enable five registers in the DUP11. The RXDBUF and PARCSR registers share the same address (76XXX2) because the RXDBUF register is read-only and the PARCSR register is write-only

A specific number of memory addresses in each PDP-11 system are reserved for communications devices. The space that includes the DUP11 device addresses extends from 760020-764000 (octal designation). These locations are termed floating addresses. The conventions used for assigning floating addresses are discussed in Chapter 2.

When the program desires to read from or write into a DUP11 register, it must address the register and indicate the type of operation to be performed. This is accomplished by placing the proper binary information on Unibus address lines A(17:00) and Unibus control lines C(01:00) and asserting BUS MSYN L. These signals are decoded by the address decoding logic to generate the enabling signal for the addressed register. This allows data from Unibus data lines D(15:00) to be written into the register, or it allows the contents of the register to be placed (read) onto the Unibus data lines.

Bits C(01:00) and A00 are decoded to indicate the type of operation or Unibus transaction to be performed (Table 4-1).

Table 4-1 Unibus Transactions for DUP11

Name	Mnemonic	Unibus Bits C01	Unibus Bits C00	Unibus Bits A00	Function
Data In	DATI	0	0	X	Data transferred from DUP11 to processor on Unibus bits D(15:00)
Data Out	DATO	1	0	X	Data transferred from processor to DUP11 on Unibus bits D(15:00).
Data Out, Byte	DATOB	1	1	0	Data transferred from processor to DUP11 on Unibus bits D(07:00) which is low byte.
		1	1	1	Data transferred from processor to DUP11 on Unibus bits D(15:08) which is high byte.

Bits A(17:03) are decoded to indicate the device address of the DUP11. Each DUP11 in a system has a different device address. The device address is selected by switches in the address decoding logic that are associated with bits A(12:03). Bits A(02:01) are decoded to select the desired register in the DUP11. Bit A00 is not used in address selection. It should be assumed always to be 0; therefore, bits A(02:01) determine the least significant octal address digit which must be 0, 2, 4, or 6. Bit A00 is used during a DATOB operation to select the upper or lower byte. The address word format is shown in Figure 4-6.

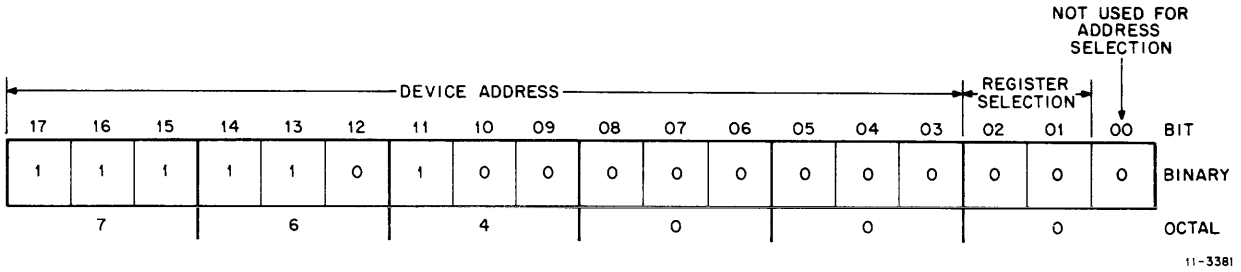


Figure 4-6 Address Word Format

Before discussing the details of the address selection logic, two prerequisite items are discussed briefly.

One item concerns the Unibus logic conventions. The Unibus uses negative logic for all signals except BG(07:04) and NPG. For clarification, the definitions of positive and negative logic are shown below:

Negative Logic

Signal Asserted: Low = 0 V = Logical 1
Signal Not Asserted: High = +3 V = Logical 0

Positive Logic

Signal Asserted:	High = +3 V = Logical 1
Signal Not Asserted:	Low = 0 V = Logical 0

The second item concerns the PDP-11 reserved memory space. A PDP-11 processor with memory management uses 18 address bits which provides a maximum memory size of 256K bytes or 128K words. The highest 8K bytes (address locations 760000–777777) are reserved for internal general registers and peripheral devices and are not programmable. A PDP-11 processor without memory management uses 16 address bits which provides a maximum memory size of 64K bytes (32K words). Logic in the processor relocates the top 8K locations accessible by the Unibus so that devices addresses in this area (760000–777777) can be generated. Programmable memory is limited to 56K bytes or 28K words. The relocation is implemented by the processor which forces address bits A(17:16) to 1s if bits A(15:13) are all 1s when the processor is master. Refer to Appendix A for more details concerning this function.

4.3.4.2 Address Decoding – The logic in this discussion is shown in logic sheet BSI6.

Device Address Decoding
Address bits A(17:03) specify the device address. Bits A(12:03) are each sent to one input of an 8837 bus receiver. The other input of each receiver is connected to ground. The inverted output of each receiver is sent to one input of an 8242 exclusive-NOR gate. The other input of each 8242 gate is connected to +5 V through a resistor and to ground through a switch. The connections are made so that with the switch OFF (open) approximately +3 V is applied to the 8242. With the switch ON (closed), this 8242 input is at ground. The 8242 gates are used as digital comparators and produce a high output only when the inputs are identical (both high or both low). In this way, the switches are used to select a specific address. With the switch OFF (open), the 8242 responds to a 1 on the Unibus. With the switch ON (closed), the 8242 responds to a 0 on the Unibus. The 8242s have open-collector outputs and all the outputs are connected together and returned to +5 V via an external resistor to form a wired-OR function. This common output is high only when all the 8242 gates find a match and produce a high output. Typical operation of an 8242 is shown in Figure 4-7.

The wired-OR connection is sent to one input of the 7437 NAND gate. The other input comes from the output of a 7314 7-input NOR gate. Its inputs are ground, BUS MSYN L, and Unibus address bits A(17:13). When the processor asserts a device address, bits A(17:13) are low and BUS MSYN L is asserted. This drives the 7314 output high, and along with the high from the wired-OR connection, drives the 7437 output low. This signal is ADR SEL L which indicates that the device address has been properly decoded by the DUP11. This signal is used in the logic that decodes the specified register bits A(02:01) and type of Unibus transaction (bits C00 and C01) in order to generate the register enabling signals.

Signal ADR SEL L is inverted, delayed, and inverted again to generate two signals. One is BUS SSYN L, which is returned to the processor to indicate that the selected register can be written into or that it has data available to be read. The other signal is DADR SEL L, which goes to the device reset logic (sheet BSI1).

Register and Transaction Decoding
The decoding of Unibus bits A(02:00) and C(01:00) along with signal ADR SEL L are used to generate the enabling signals to perform a read or write operation on the selected register.

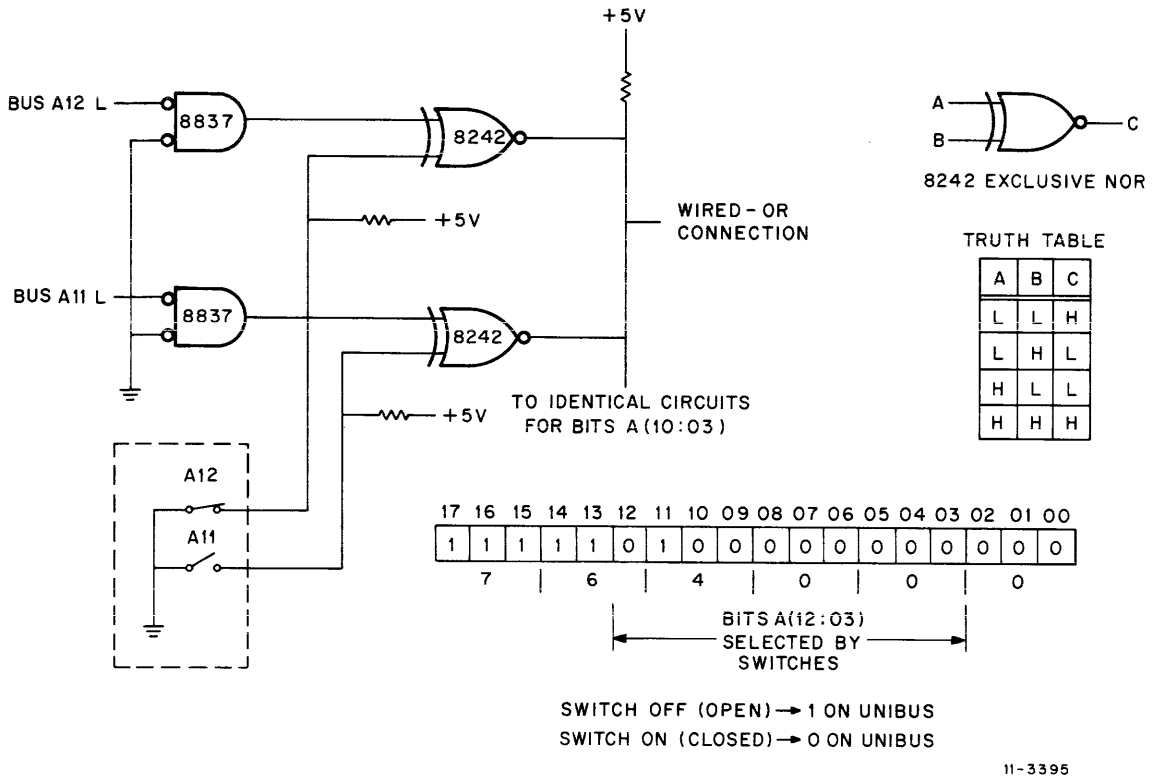


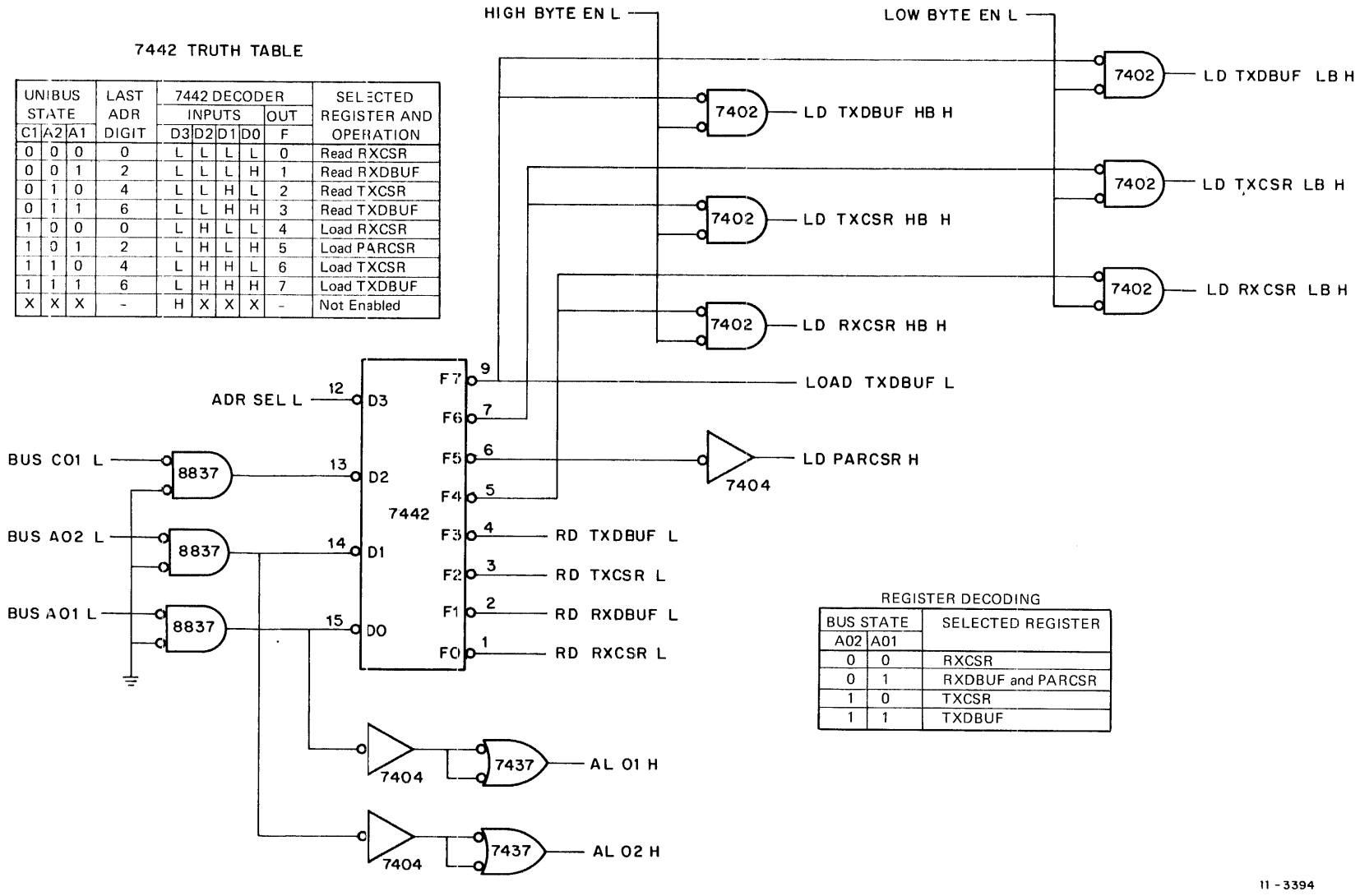
Figure 4-7 Typical Operation of an 8242 Comparator

The 7442 decoder is connected as a 3-wire binary to octal decoder (Figure 4-8). Three of the four inputs (D0, D1, and D2) decode the binary code and the fourth (D3) is the strobe which must be low to enable the decoder. Inputs D0, D1, and D2 are connected to the outputs of the Unibus receivers for BUS A01 L, BUS A02 L, and BUS C01 L, respectively. The strobe signal is ADR SEL L which is low when the DUP11 device address is decoded. The 7442 truth table in Figure 4-8 shows the selected register and operation to be performed.

Outputs FO-F3 of the 7442 decoder are selected for read operations on the RXCSR, RXDBUF, TXCSR, and TXDBUF registers, respectively. The actual output signals are not used because program access to these registers is made via the multiplexed UNIBUS drivers. Only one signal is required to enable the drivers. The signal is DATA → BUS L and it is enabled by the register selection logic when ADR SEL L goes low and BUS C01 L is a 0.

7442 TRUTH TABLE

UNIBUS STATE			LAST ADR	7442 DECODER INPUTS				OUT	SELECTED REGISTER AND
C1	A2	A1	DIGIT	D3	D2	D1	D0	F	OPERATION
0	0	0	0	L	L	L	L	0	Read RXCSR
0	0	1	2	L	L	L	H	1	Read RXDBUF
0	1	0	4	L	L	H	L	2	Read TXCSR
0	1	1	6	L	L	H	H	3	Read TXDBUF
1	0	0	0	L	H	L	L	4	Load RXCSR
1	0	1	2	L	H	L	H	5	Load PARCSR
1	1	0	4	L	H	H	L	6	Load TXCSR
1	1	1	6	L	H	H	H	7	Load TXDBUF
X	X	X	-	H	X	X	X	-	Not Enabled



REGISTER DECODING

BUS STATE		SELECTED REGISTER
A02	A01	
0	0	RXCSR
0	1	RXDBUF and PARCSR
1	0	TXCSR
1	1	TXDBUF

11-3394

Figure 4-8 Register Decoding Logic

Decoder outputs F4–F7, are selected for write (load) operations. Output F5 is active when the PARCSR register is selected. This output signal is inverted to become LD PARCSR H which enables both bytes of the PARCSR register. Outputs F4, F6, and F7 are selected for write operations on the RXCSR, TXCSR, and TXDBUF registers, respectively. Differentiation between loading words (DATO) and bytes (DATOB) is made by using additional gates. Signals LOW BYTE EN L and HIGH BYTE EN L are both sent to one input of three 7402 NOR gates to provide a set of three high byte enabling gates and a set of three low byte enabling gates. Decoder outputs F4, F6, and F7 are sent to each set of gates to provide low and high byte enabling signals for the RXCSR, TXCSR, and TXDBUF registers. These signals are listed below.

Enable Low Byte	**Enable High Byte**
LD TXDBUF LB H	LD TXDBUF HB H
LD TXCSR LB H	LD TXCSR HB H
LD RXCSR LB H	LD TXCSR HB H

To write a word in a specified register, the low byte and high byte enabling signals are asserted simultaneously. In addition, decoder output F7 (LOAD TXDBUF L) is used to clock the TXDONE flip-flop which clears this flip-flop because its D input is permanently connected to ground.

The outputs of the drivers for bits BUS A02 L and BUS A01 L are buffered to generate AL 02 H and AL 01 H which are the select inputs for the Unibus multiplexers.

4.3.5 Unibus Receivers and Multiplexed Unibus Drivers (Logic Sheet BSI8)

The outputs of four DUP11 registers are multiplexed onto the Unibus data lines through one set of 16 bus transceivers. This is accomplished by using 8 dual 4-line to 1-line multiplexers. The registers are: RXCSR, RXDBUF, TXCSR, and TXDBUF. The fifth register (PARCSR) is write-only so it cannot be read by the program.

A typical 2-bit slice of the bus multiplexer logic is shown in Figure 4-9. The 74153 multiplexer has two identical sections that select 1 of 4 inputs. Each section has its own strobe or enabling input, but common address or select lines are used. Both strobe inputs (STB1 and STB0) are connected to ground which keeps both sections of the MUX constantly enabled. The select inputs S1 and S0 are connected to signals AL 02 H and AL 01 H, respectively. These signals are derivatives of Unibus address lines A02 and A01. These bits determine which register is selected by specifying the least significant digit of the device octal address (0, 2, 4, or 6). Input selection is shown in the truth table that appears in Figure 4-9. Input A is the RXCSR, input B is the RXDBUF, input C is the TXCSR, and input D is the TXDBUF. Each mux output goes to one input of the driver of the 8641 bus transceiver. Each transceiver contains four driver/receiver combinations plus a common driver enabling gate. The enabling signal is DATA BUS L which is generated by the address selection logic when the device address is decoded and Unibus control bit C01 is a 0 which indicates that a DATI (read) operation is to be performed.

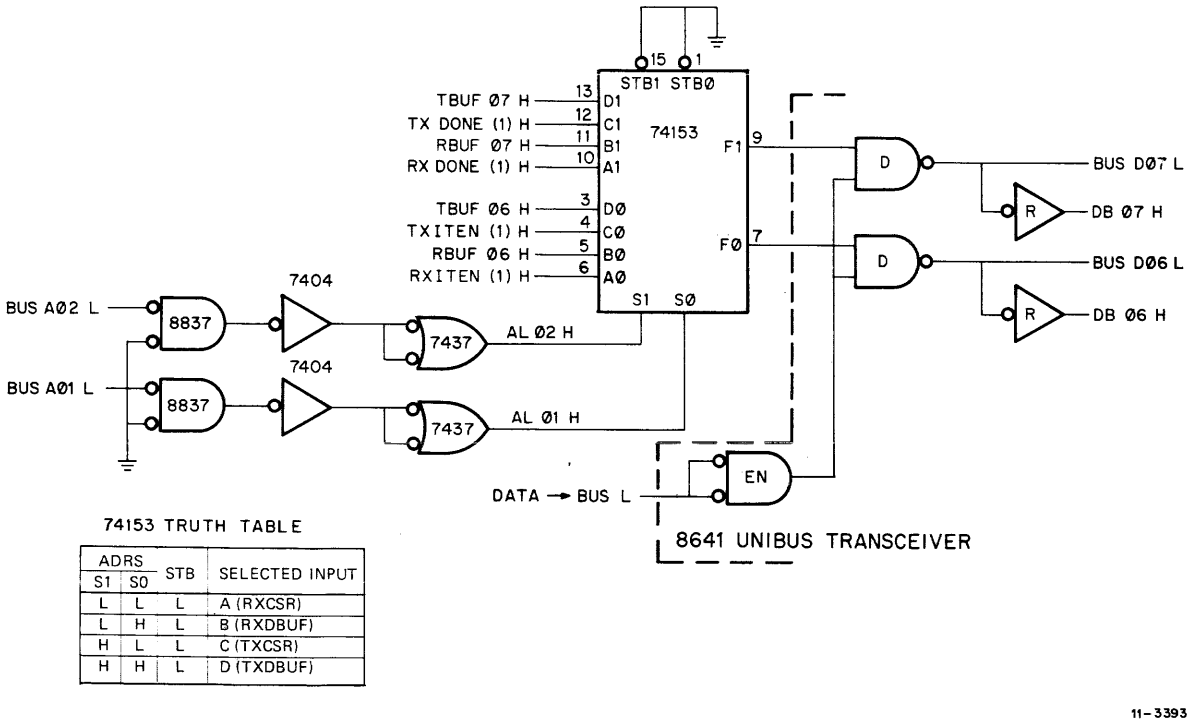


Figure 4-9 Typical 2-Bit Slice of Unibus Multiplexer Logic

4.3.6 Transmitter Logic (Logic Sheet BSI2)

The detailed discussion of the transmitter logic is divided into the following five parts:

Title	Paragraph
ROMs and Bit Sync Buffer	4.3.6.1
Clock Logic	4.3.6.2
TXDAT Flip-Flop and T1BC Counter	4.3.6.3
TCSC Counter	4.3.6.4
Shift Register	4.3.6.5

4.3.6.1 ROMs and Bit Sync Buffer

General

Three read only memories (ROMS) are the major controlling elements for the transmitter. Each one is a 1024-bit TTL ROM organized as 256 words of 4 bits each. Type 74187 ROMs are used but other equivalent programmable ROMs (5603) can be used. Both enabling inputs (pins 13 and 14) are held low to keep the ROM enabled constantly. The inputs represent an 8-bit, binary-coded address that selects any one of the 256 words (addresses 0–255). The most significant input is pin 15 and the least significant input is pin 5. Each word is preprogrammed and is unalterable. When addressed, a speciffic word always produces the same states at the four outputs. As control elements, the ROMs act as compact logic arrays that replace a large amount of distributive logic.

The three ROMs are the function decode ROM, the data path control ROM, and the data decode ROM. A listing for each ROM is contained in the DUP11 print set. The listing contains input/output binary equivalents for each address, along with a brief note of what the address represents. Many addresses form combinations of inputs that are functionally meaningless or are not allowed. These addresses are defined as illegal. For the function decode ROM and the data path control ROM, all illegal addresses generate no ROM outputs. For the data decode ROM, all illegal addresses assert only one output (TXDT H) which indicates that the transmitting line is in the idle state. The circuit schematic for the ROMs and associated logic is contained in logic sheet BSI2. A simplified diagram is shown in Figure 4-10.

The data path control ROM has its enabling inputs (pins 13 and 14) connected directly to ground. The other two ROMs are enabled by signal GR TEST PT. This signal is generated by a 7404 inverter whose input is connected to +5 V; therefore, signal GR TEST PT is held low. Signal GR TEST PT can be manipulated only by the module tester to simulate a disabled ROM.

In the following discussion, the source and destination of the signals associated with the ROMs and bit sync counter are described. Not all signals are described functionally. Some signals, specifically ROM inputs, have relevance only when viewed as a group during a specific point in a transmit operation. This aspect is covered in Paragraph 4.3.11.2 of this chapter which describes a typical transmit operation.

Function Decode ROM

The function decode ROM (FDROM) responds to the program and to the transmitter logic. This ROM controls the setting of TXDNE and decodes the program inputs, which in some cases are synchronized to the data set clock. This information, along with the current state of the logic, determines the next event on a character basis. The program controls the following FDROM inputs directly from outputs of the PARCSR, TXCSR, and TXDBUF registers as shown below.

Input Signal	Input Pin	Signal Source
DEC MODE (1) H	15	PAR CSR bit 15
NO CRC (1) H	1	PARCSR bit 9
SEND (1) H	2	TX CSR bit 4
TEOM (1) H	6	TXDBUF bit 9

TXDBUF bit 8 (TSOM) is also used via a gating network to control input 7 of the FDROM. Signal TSOM (1) H from bit 8 of the TXDBUF register is ORed with SCRC (1) H to generate signal SEND FLG H which is FDROM input 7. Signal SSOM (1) H represents the synchronization of TSOM (1) H with the data set TX clock at the output of the bit sync buffer. Signal SCRC (1) H is also an output of the bit sync buffer. It represents the synchronization of signal TCRC H from the FDROM.

Two of the remaining signals, SFLG (1) H (input 4) and SENTXAC (1) H (input 3), are outputs of the bit sync buffer also. They represent synchronization of FDROM outputs TFLG H and ENTXAC H, respectively.

The remaining signal is SKPLD H (input 5) which is the AND function of LOAD H and SKP SEQ (1) H. It is used to prevent the loss of a data character without error indication if the program accesses the TXDBUF too late during the transition between sending a control character and a data character. Signal SKP LD H inhibits the setting of TXDNE and allows an extra control character to be sent. The character that was loaded late is sent after the extra control character.

Three of the four outputs from the function decode ROM are sent to the bit sync buffer. The fourth output (ENTXDNE H) goes to the preset input gating for the TXDONE flip-flop. This signal allows the TXDONE flip-flop to be set directly at specific times during a transmit sequence.

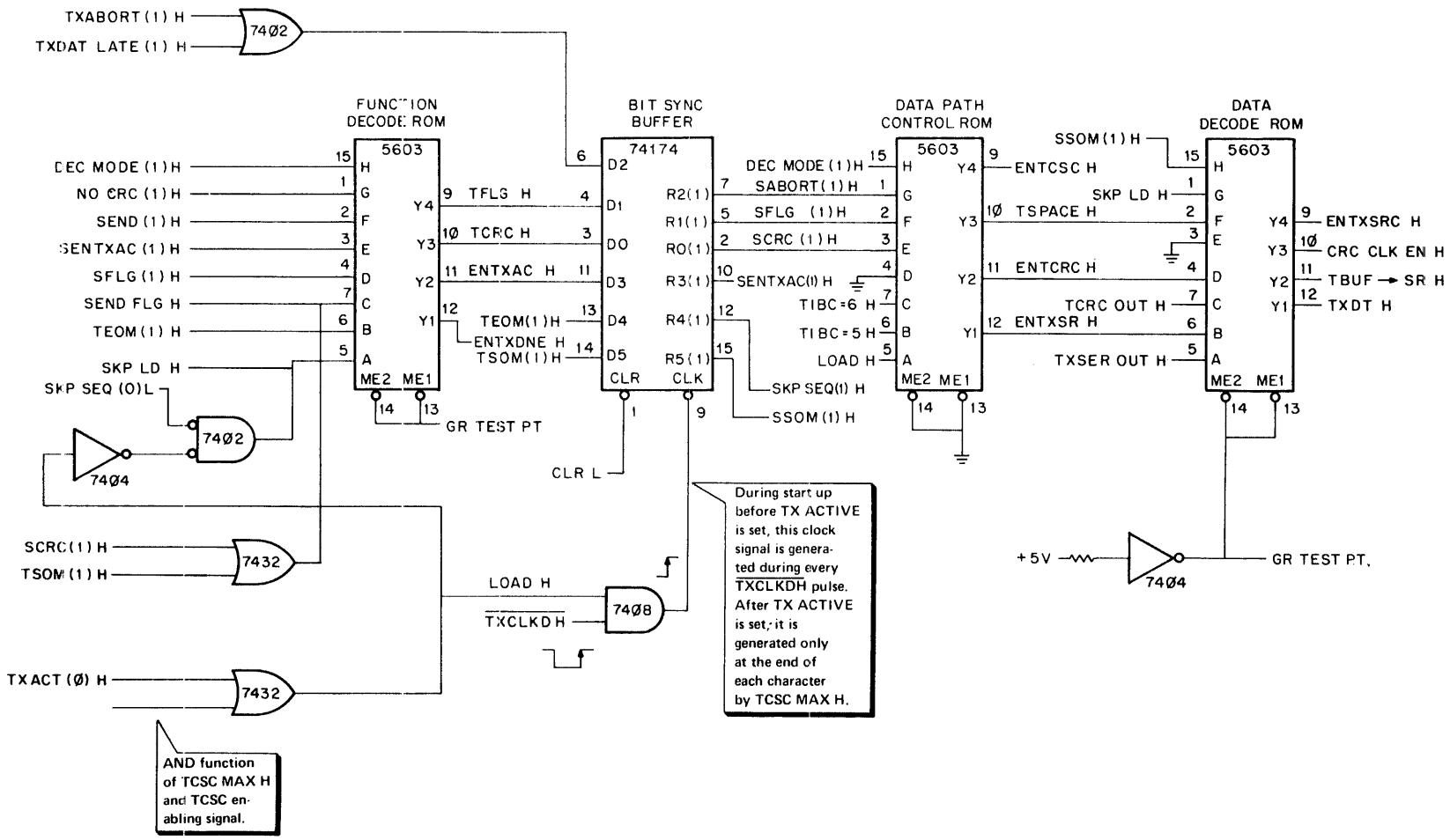


Figure 4-10 Transmitter ROMs and Associated Logic

Bit Sync Buffer

The bit sync buffer provides storage for six transmitter control logic signals (including three from the function decode ROM). After the flag or sync characters have been sent, the bit sync buffer is clocked only at the end of a character. This allows the logic to set up for the next character during the present character while not affecting the outputs of the bit sync buffer. The bit sync buffer is a 74174 hex D-type flip-flop. It has six individual D inputs each of which has a single-rail output (Q=H=1). It has common clock and clear inputs. Three inputs come from the FDROM: TFLG H, TCRC H, and ENTXAC H. The corresponding bit sync buffer outputs are: SFLG (1) H, SCRC (1) H, and SEN-TXAC (1) H. The fourth and fifth inputs are: TXDNE (1) H (TXCSR bit 7) and TSOM (1) H (TXDBUF bit 8). The corresponding outputs are: SKP SEQ (1) H and SSOM (1) H. The sixth input is the result of ORing signals TXABORT (1) H and TXDAT LATE (1) H. Signal TXABORT (1) H is TXDBUF bit 10 which is program-controlled. Signal TXDAT LATE (1) H is TXCSR bit 15 which is hardware-controlled. The corresponding BSB output is SABORT (1) H. The bit sync buffer is cleared by signal CLR L from the device reset logic.

Three gates and several signals are used to provide the clock signal for the bit sync buffer. During idle, and up to the time that the first bit of the flag is transmitted, TXACTIVE is cleared. As a result, signal TXACT (0) H is high. This signal is sent to one input of a 7432 OR gate to produce LOAD H at its output. LOAD H, which is a level signal at this time, is ANDed with clock signal $\overline{\text{TXCLKD H}}$ to clock the bit sync buffer on every positive transition of $\overline{\text{TXCLKD H}}$. These transitions occur about 250 ns after each trailing edge (positive) transition of clock signal TXCLK H. (Details of the TX clock logic are covered in Paragraph 4.3.6.2.) As long as TXACT is cleared, the bit sync buffer is clocked on every positive transition of $\overline{\text{TXCLKD H}}$.

When TXACT is set by the hardware, LOAD H is driven low and only goes high again for a short time when the TCSC counter reaches the last bit of a character. At this time, the TCSC counter asserts positive pulse TCSC MAX H which is ANDed with the TCSC counter enabling signal (ENTCSC H) to generate LOAD H. Pulse TSCSMAX H times out when the counter overflows (all outputs go to 0). With LOAD H asserted, the positive transition of the $\overline{\text{TXCLKD H}}$ pulse associated with the last bit of the current character clocks the bit sync buffer. The outputs of the bit sync buffer can change only when the buffer is clocked. They are considered to be synchronized to the data set transmitter clock because the buffer is clocked by $\overline{\text{TXCLKD H}}$ which is a derivative of the data set transmitter clock. Except during the startup phase of transmission, the bit sync buffer is clocked only at the end of a character. This allows the program to change the inputs to the function decode ROM in anticipation of the next event without interfering with the current event, thus synchronizing the program and data set.

Data Path Control ROM

The data path control ROM formats the SDLC control characters and controls transmitter data path multiplexing. The data path control ROM inputs are shown below:

Input Signal	Input Pin	Signal Source
DEC MODE (1) H	15	PAR CSR bit 15
SABORT (1) H	1	FD ROM output
SFLG (1) H	2	FD ROM output
SCRC (1) H	3	FD ROM output
Ground	4	
T1BC=6H	7	T1BC Counter output
T1BC=5H	6	T1BC Counter output
LOAD H	5	TX logic

Three of the four data path control ROM outputs go to the data decode ROM. These outputs are: TSPACE H, ENTCRC H, and ENTXSR H. The fourth output (ENTCSC H) is the enabling signal for the TCSC counter. Signal ENTCSC H must be high to allow the counter to count. This signal is not asserted (goes low) when a 0 is being stuffed into the transmitted data stream; therefore, the TCSC counter is inhibited to prevent the stuffed 0 from being counted.

Data Decode ROM
The data decode ROM multiplexes the data received from the TX shift register, TXCRC register, and data path control ROM. It also controls the timing of transfers from the TXDBUF register to the TX shift register and the clocking of most of the transmit data path. The data decode inputs are shown below.

Input Signal	Input Pin	Signal Source
SSOM (1) H	15	BSB output
SKPLD H	1	TX logic
TSPACE H	2	DPCROM output
Ground	3	
ENTCRC H	4	DPCROM output
TXCRC 15 H	7	TX CRC Register output bit 15
ENTXSR H	6	DPCROM output
TXSER OUT H	5	TX shift register output

The four data decode ROM outputs are: ENTXSRC H, CRC CLK EN H, TBUF→SR H, and TXDT H.

Signal ENTXSRC H is a qualifying input for the transmitter shift register clock. This register can be clocked only when ENTXSRC H is asserted. This signal is not asserted during the following conditions: transmission of SDLC control characters or CRC information; when a bit is stuffed; and when TXDBUF→SR is asserted.

Signal CRC CLK EN H is a qualifying input for the TXCRC register clock. The TXCRC register can be clocked only when CRC CLK EN H is asserted. This signal is not asserted when control or sync characters are being transmitted, or when a 0 is stuffed.

Signal TBUF→SR H must be asserted to allow bits 0–7 of the TXDBUF register to be loaded into the TX shift register. (This is information to be transmitted.)

Signal TXDT H goes to the TXDT H flip-flop. This signal represents the state of the transmitter output data line (EIA XMIT DATA). It is the multiplexed output of the transmit data path. When TXDT H is high, the line is in the MARK state; when TXDT H is low, the line is in the SPACE state. (A more detailed description of the relationship between signal TXDTH and EIA XMIT DATA is contained in Paragraph 4.3.6.3.)

4.3.6.2 Clock Logic – This discussion covers two separate but related clock circuits. One is the internal RC clock or single-step clock that is enabled only during maintenance operation. The other circuit consists of the logic that transforms the data set transmitter clock (or internal clock) into the various clock signals required for transmitter operation.

Internal Clock Logic
(Refer to Figure 4-11.) The source for the clock logic is a 20 kHz RC clock consisting of two 7404 inverters, feedback capacitors and resistors. It is a free-running clock that starts when power is applied to the DUP11 and stops only when power is removed.

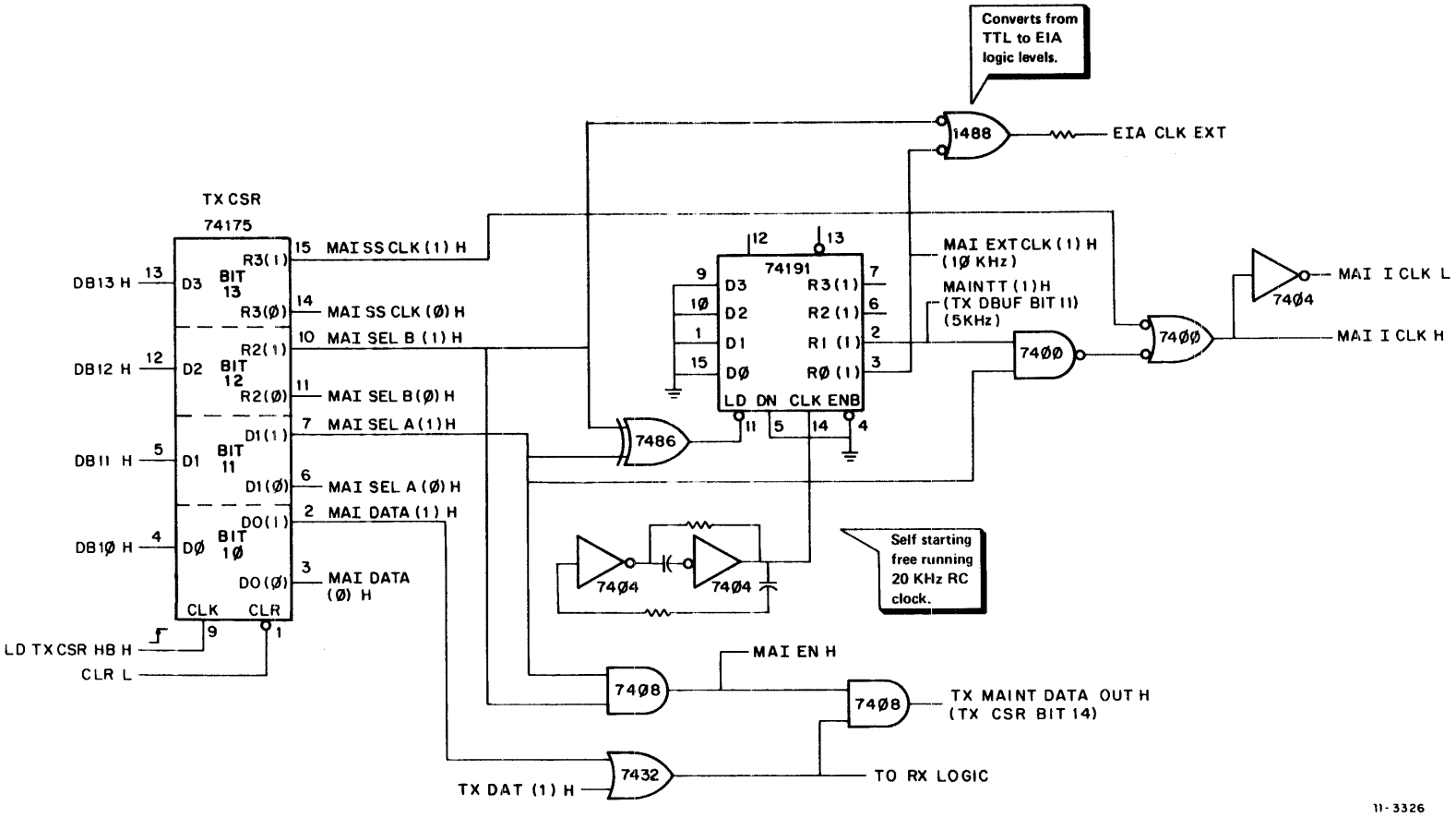


Figure 4-11 Internal Clock Logic

The clock logic is enabled only during the three categories of maintenance operation: system test, external maintenance, and internal maintenance. The RC clock is the source during the system test and external maintenance modes while the single-step clock is used during the internal maintenance mode. During user operation, this logic is disabled and the DUP11 uses the clocks supplied by the modem.

The main element of the clock logic is the 74191 synchronous counter. It is a 4-bit binary counter that is permanently enabled to count up only. This is accomplished by connecting the DN and ENB inputs to ground permanently.

The output of the RC clock is connected to the CLK input of the counter. This means that the outputs of the counter represent the division by 2, 4, 8, and 16 of the RC clock output. The second least significant bit of the counter (pin 2) provides a 5 kHz ±20% signal that is sent to the output gates to become MAI ICLK H which is used as a free-running clock during the system test mode. This signal is also single-stepped by the program during the internal maintenance mode using, TXCSR bit 13 (MAI SS CLK). The least significant bit of the counter (pin 3) is a 10 kHz ±20% signal that is used as a free-running clock [MAI EXT CLK (1) H] during the external maintenance mode.

Selection of the proper clock signal for the user or maintenance modes is controlled by the states of bits 12 (MAINT MODE SELB) and 11 (MAINT MODE SELA) of the TXCSR. These bits are under program control. The selection is made as shown in Table 4-2.

Table 4-2 Clock Signal Selection for Maintenance Modes

TXCSR Bits		Mode	Clock Signal
12	11		
0	0	User	No clock (supplied externally)
0	1	System Test	MAI ICLK H (5 kHz)
1	0	External Maint.	MAI EXT CLK (1) H (10 kHz)
1	1	Internal Maint.	MAI ICLK H (single-step using TXCSR bit 13)

1 = Bit Set
0 = Bit Cleared

Bits 12 and 11 of the TXCSR are contained in a 74175 quad D-type flip-flop. The 1 outputs of these flip-flops [MAI SELB (1) H and MAI SELA (1) H] are exclusive-ORed and the result is sent to the load (LD) input of the counter. When the LD input is low, the counting function is inhibited and the counter outputs change to agree with the inputs. In this application, the outputs always go to all 0s during a load operation because the inputs are permanently connected to ground.

The output of the X-OR gate goes low only when its inputs are identical. This occurs during the user mode and the internal maintenance mode to hold the counter LD input low and prevent the generation of free-running clock pulses. However, during the internal maintenance mode, the diagnostic program toggles TXCSR bit 13 (MAINT SS CLK) which allows signal MAI SS CLK (1) H to single-step clock signal MAI ICLK H.

During the system test mode, the counter is operating. The 5 kHz output of the counter is signal MAINTT (1) H which is TXDBUF bit 11. This signal is sent to bit 11 of the multiplexed bus drivers where it can be monitored. Signal MAINTT (1) H also goes to a NAND gate. The other input of this gate is MAI SEL A (1) H which is high because TXCSR bit 11 is set in this mode. This produces an inverted 5 kHz signal at the NAND gate output. This signal is sent to another NAND gate (shown as a logically equivalent negated-input OR gate). The other input of this gate is signal MAI SS CLK (1) H which is TXCSR bit 13 (MAINT SS CLK). When this bit is set [MAI SS CLK (1) H asserted], the output of the gate (MAI ICLK H) is a 5 kHz free-running clock that is in phase with the counter output.

In the external maintenance mode, the counter is operating. The 10 kHz output [MAI EXT CLK (0) L] of the counter goes to one input of a 1488 line driver that converts the DUP11 TTL levels to EIA logic levels. The other input of the driver is MAI SELB (0) L which is high because TXCSR bit 12 is set for this mode. Therefore, the output of the driver (EIA CLK EXT) is a 10 kHz free-running clock at EIA logic levels. Signal MAI SELA (1) H is low because TXCSR bit 11 is cleared for this mode. This signal inhibits the MAI ICLK H clock. Figure 4-12 shows the signal states at the output NAND gates for the various modes.

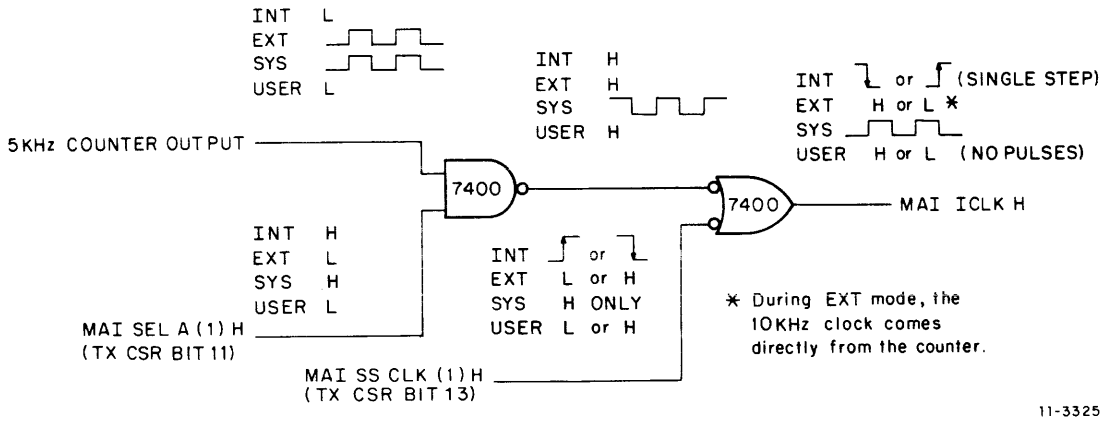


Figure 4-12 Signal States for Maintenance Clock

TXCSR bit 14 (MAINT TX OUT DATA) is associated with the internal maintenance mode and is discussed here for convenience. This read-only bit allows the diagnostic program to monitor the output of the transmitter during the internal maintenance mode. The bit is monitored at the output of a 2-input AND gate. The output signal (TX MAINT DATA OUT H) is sent to bit 14 of the multiplexed bus drivers where it is read by the program. One input of this AND gate is high only when TXCSR bits 12 and 11 are set, which occurs only in the internal maintenance mode. The other input is the result of the ORing of TX DATA (1) H and MAI DATA (1) H. Signal TXDAT (1) H is the serialized transmitter output.

Either input [MAI DATA (1) H or TXDAT (1) H] can be used as the source of data to the receiver serial input line. MAI DATA (1) H is TXCSR bit 10 (MAINT INPUT DATA) and it can be toggled by the diagnostic program. If this input is used, the transmitter control logic is inhibited and TXDAT (1) H is cleared.

If the transmitter control logic is to be tested, the MAINT INPUT DATA bit must be cleared [MAI DATA (1) H is low] and TXDAT (1) H is the source.

In either case, the information is sent to the receiver serial input line and is clocked into the receiver shift register by the maintenance clock (MAI ICLK H) which is controlled by the diagnostic program, using the single-step maintenance clock (TXCSR bit 13).

Transmitter Clock Logic
(Refer to Figure 4-13.) The transmitter clock is supplied by the data set during the user mode. A logic circuit uses the data set clock or internal RC clock during system test and external maintenance modes to derive a group of clock signals for the transmitter logic. This circuit can be thought of as a timing generator. During the internal maintenance mode, a single-step clock is controlled by the program, using TXCSR bit 13.

The input to this circuit is a 7450 dual AND-OR-invert gate. The inputs to one AND input are MAI SEL A (0) H and TXDAT SET CLK H. The inputs to the other AND input are MAI SEL A (1) H and MAI ICLK H. In the user mode, TXCSR bit 11 (MAINT MODE SEL A) is cleared; therefore, MAI SELA (1) H is low and MAI SEL A (0) H is high. In this case, the TXDAT SET CLK H signal is gated into the circuit and the other 7450 AND input is disqualified. Signal TXDAT SET CLK H is the transmitter clock from data set lead EIA XMIT CLK after it has been converted to TTL levels and double inverted. Signal MAI ICLK H is the maintenance internal clock and it is gated into the circuit during the system test mode or the internal maintenance mode when TX CSR bit 11 is set [MAI SEL A (1) H is high.]

Assume that the DUP11 is transmitting in the user mode. Signal TXDAT SET CLK H is a symmetrical square wave pulse train. It is inverted by the 7450 gate to become TXCLK L. This signal clocks the TCSC counter, the T1BC counter, and the TXACT flip-flop. In the illustrations, TXCLK L is shown as the logically equivalent signal $\overline{\text{TXCLK H}}$. It is also designated $\overline{\text{TXCLK H}}$ in the print set.

In the circuit, TXCLK L is inverted by the 8881 gate to become TXCLK H which clocks the TX shift register. TXCLK H is delayed about 250 ns to become TXCLKD H. This signal is inverted to generate TXCLKD L which clocks the bit sync buffer. (Throughout the TX control logic, TXCLKD L is shown as the logically equivalent signal $\overline{\text{TXCLKD H}}$.) Signals TXCLK H and TXCLKD L are ANDed in the 7408 gate to generate TXCLKEG H. The AND gate, inverter, and delay form a pulse generator that asserts TXCLKEG H as a 250 ns positive pulse only on the positive-going edge of TXCLK H. Signal TXCLKEG H is used to control the timing of the TX shift register.

The positive-going edge of TXCLKD H triggers a 74123 one-shot to generate TXCLKP H which is a positive pulse of 300 ns duration. This signal is used in the gating for the preset input of the TXDONE flip-flop. It is also the clocking signal for the TXCRC register. The positive-going trailing edge of the inversion (TXCLKP L) of this signal clocks the TXDAT flip-flop.

4.3.6.3 TXDAT Flip-Flop and T1BC Counter – The TXDAT flip-flop and T1BC counter are discussed together because they are closely related. The conversion of the transmitter data (TXDT H) from TTL to EIA logic levels is also discussed.

TXDAT Flip-Flop (BSI5)
The TXDAT flip-flop is a D-type (7474) whose D input is connected to the TXDT H output of the data decode ROM. The logical state of signal TXDT H is the same as the transmitted data at the output of the DUP11 after signal conversion to EIA logic levels. TXDT H is controlled by the data decode ROM and latched in the TXDAT flip-flop. The 1 output of this flip-flop is sent through some gating logic and is converted to EIA logic levels and leaves the DUP11 as EIA XMIT DATA (Figure 4-14).

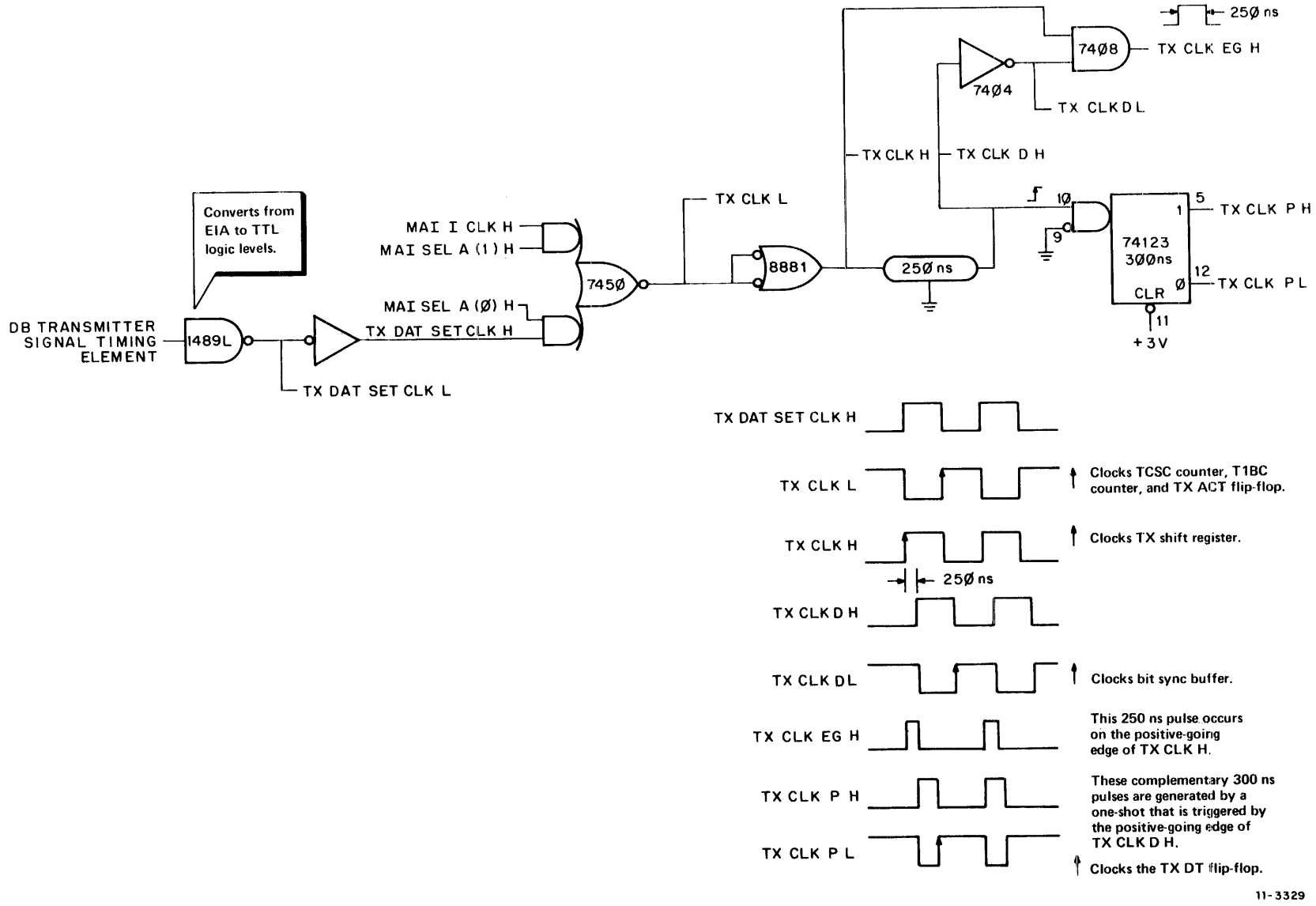


Figure 4-13 Transmitter Clock Logic and Timing Diagram

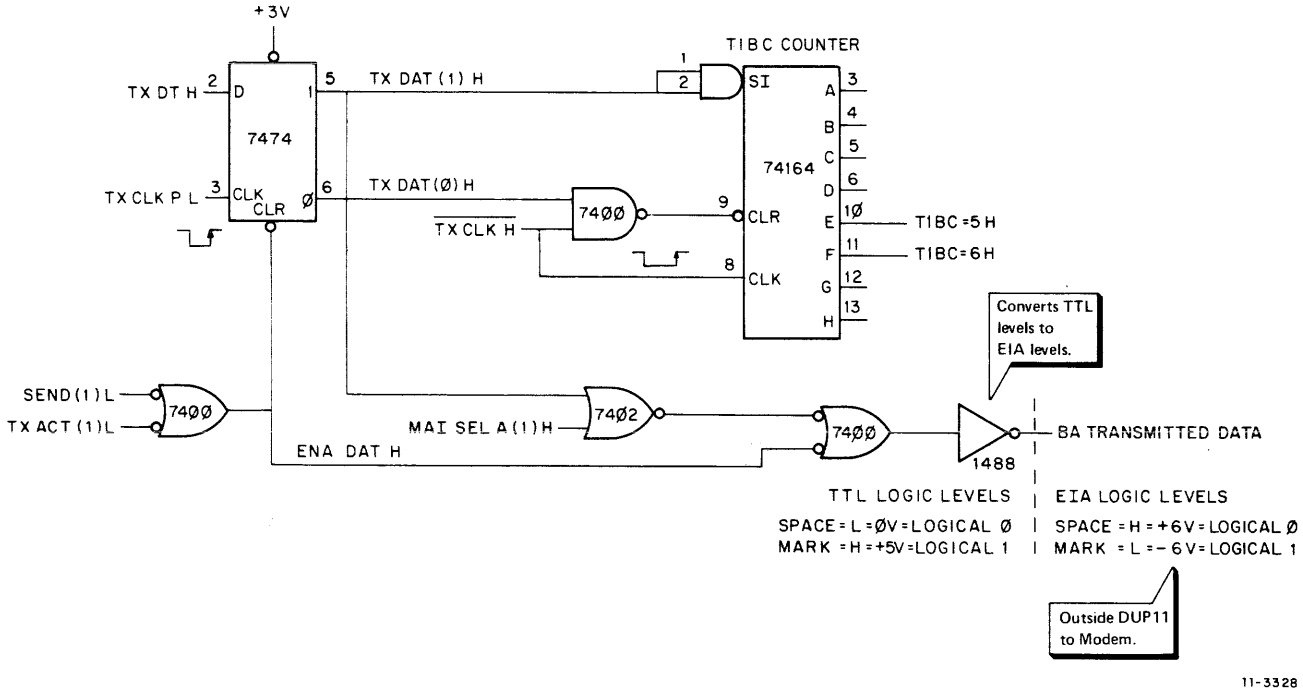


Figure 4-14 TXDAT Flip-Flop and T1BC Counter

The following definitions clarify the level conversion process.

Outside the DUP11 (reference signal TRANSMITTED DATA at the output of level converter):

SPACE = H = +3 V = logical 0
MARK = L = −3 V = logical 1

Inside the DUP11 (reference signal TXDT H at output pin 12 of the TX data decode ROM):

SPACE = L = 0 V = logical 0
MARK = H = +5 V = logical 1

Reference the listing for the TX data decode ROM. When it is desired to put the transmit line in the MARK state (idle mode or logical 1), signal TXDT H (ROM output pin 12) is logical 1. To put the transmit line in the SPACE state (logical 0), signal TXDT H is logical 0.

A typical example is traced through the conversion logic. Assume that the DUP11 is transmitting in the user mode and signal TXDT H is logical 1 which calls for a MARK to be transmitted. This signal is clocked into the TXDAT flip-flop by TXCLK P L. The 1 output of this flip-flop [TXDAT (1) H] is high and remains in this state, even if the ROM changes TXDT H, until the next TXCLKP L positive-going edge. Signal TXDAT (1) H is ORed with MAI SELA (1) H at the 7402 NOR gate. Signal MAI SELA (1) H (TXCSR bit 11) is low because this bit is cleared in the user mode. TXDAT (1) H is gated through the 7402 in inverted form and is sent to the 7400 NAND gate. The other input of the 7400 gate is high (EN DAT H asserted) because TXACT and SEND are set [TXACT (1) L and SEND (1) L both low]. The output of the 7402 gate is gated through the 7400 gate and is inverted again in the process. The output of the 7400 gate is sent to the 1488 line driver. The signal is inverted and converted to EIA logic levels. The signal is now called TRANSMITTED DATA and has been inverted three times, so it is low. This means that it is a MARK (logical 1) when sent to the corresponding EIA receiver.

When the DUP11 is not actively transmitting, TXACT and SEND are both cleared. This drives EN DATA H low which holds the transmit line BA TRANSMITTED DATA in the idle or MARK state. Signal EN DATA H is also sent to the clear input of the TXDAT flip-flop. When EN DATA H goes low, the TXDAT flip-flop is cleared which inhibits the T1BC counter. (This action is described below.)

T1BC Counter

Functionally, the T1BC counter keeps track of the number of consecutive 1s transmitted. This information is used in the SDLC mode (DEC MODE bit cleared) to make protocol decisions when transmitting control characters and to maintain data transparency.

The T1BC counter is a 74164 8-bit parallel-out serial shift register. Both serial inputs (pins 1 and 2) are connected to the 1 output of the TXDAT flip-flop; therefore, with TXDAT (1) H high, a 1 is shifted in when the T1BC counter is clocked by the positive-going edge of TXCLK H. This clock signal is ANDed at a 7400 NAND gate with the 0 output [TXDAT (0) H] of the TXDAT flip-flop. When the TXDAT flip-flop is cleared [TXDAT (0) H is high] and the clock pulse (TXCLK H) goes high, the NAND gate output goes low and the T1BC counter is cleared. In this mode of operation, the T1BC counter counts consecutive 1s and is cleared when a 0 is sensed at its serial input.

When the DUP11 is not actively transmitting, signal EN DATA H goes low and directly clears the TXDAT flip-flop. The 0 output [TXDAT (0) H] goes high and is ANDed with TXCLK H to keep the counter cleared during every TX data set clock cycle.

Only two of the eight outputs of the T1BC counter are used. They are the 5th most significant output (pin 10) identified as T1BC = 5H and the 6th most significant output (pin 11) identified as T1BC = 6H. Signal T1BC = 5H goes high when five consecutive 1s are shifted in and T1BC = 6H goes high when six consecutive 1s are shifted in (T1BC = 5H remains high).

These two signals are inputs to the data path control ROM. During the transmission of a flag character, they control the selection of the 7th and 8th bits. During the transmission of all other characters, they force the insertion of a 0 after a series of five consecutive 1s.

Transmission of a flag character (01111110) is controlled by the transmitter logic. When the fifth 1 (MARK) is detected, the T1BC counter asserts T1BC = 5H. The data path control ROM responds by indicating that a 1 (MARK) is to be sent next. The next MARK (7th bit) is the last 1 in the flag character. When it is detected, the T1BC counter asserts T1BC = 6H. The ROM responds by indicating that a 0 (SPACE) is to be sent as the last bit of the flag character.

The SDLC protocol frame begins and ends with a flag character. All characters between the flags must not contain a flag bit pattern. The transmitter inserts (stuffs) a 0 after a sequence of five contiguous 1s that occur within the frame so that a flag pattern (01111110) cannot be transmitted by chance. Under these conditions when five consecutive 1s are detected, the T1BC counter asserts T1BC = 5H. The data path control ROM responds by indicating that a 0 (SPACE) is to be sent next. When the T1BC counter detects this 0, it is cleared.

The use of a 74164 shift register as a 1s counter and 0 stuffer is certainly unique. This distinctive device should be called the Zereski Zero Stuffer (ZZS).

4.3.6.4 Transmitter Character Serialization Counter (TCSC) – The TCSC counter counts the number of bits in each data character that is transmitted exclusive of stuffed 0s (Figure 4-15). It also counts the number of bits in a control character. It counts to 16 for the CRC character and SPACE sequence and to 8 for all others. At the last bit of the character, it generates a pulse (TSCS MAX H) that synchronizes the current action of the transmitter data path to the program interface.

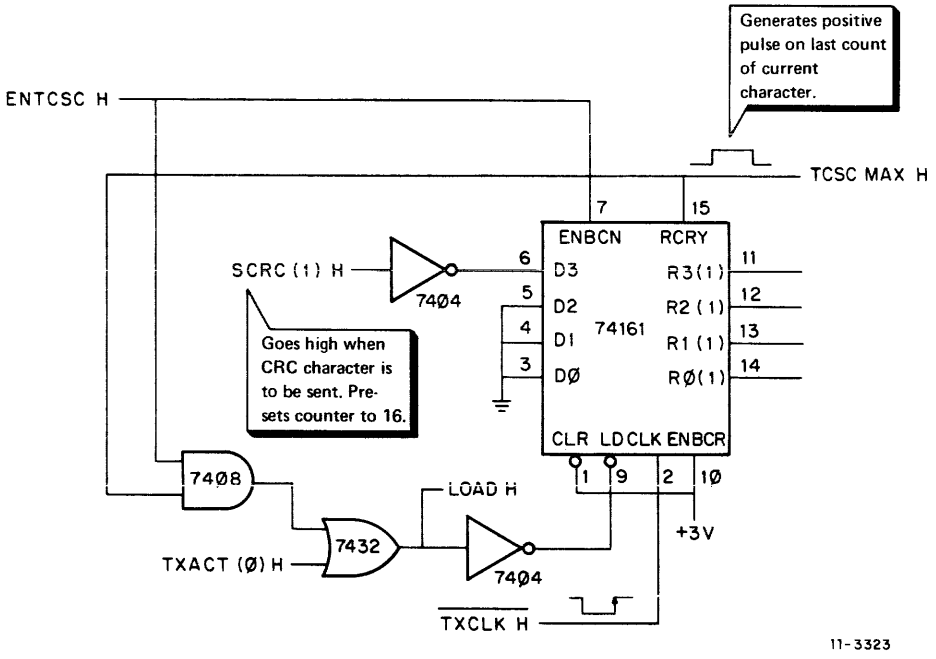


Figure 4-15 TCSC Counter

It is a 74161 synchronous 4-bit counter. Both count enable inputs ENB CN (pin 7) and ENB CR (pin 10) must be high to enable the counter. ENB CR (pin 10) is permanently connected to +5 V and ENB CN (pin 7) is controlled by signal ENTCSC H. The clear (CLR) input is inhibited by connecting it to +5 V. The counter data outputs are not used. A low signal on the load (LD) input inhibits incrementing the counter and causes the outputs to agree with the inputs after the next clock pulse. When the counter reaches its maximum counts (all 1s or count 15), the carry out pulse (TCSC MAX H) is generated. This positive pulse is of the same duration as the positive portion of the LSB output (pin 14). It times out when the next clock pulse arrives and the counter overflows (all outputs go to 0).

The counter can be preset during the load operation to count up to 8 or 16. The CRC character or SPACE sequence is 16 bits long and when it is coming up for transmission, signal SCRC (1) H is high. This signal is inverted which puts a low on the counter MSB input (pin 8). The other three counter inputs are always held low because they are connected to ground. When the LD input goes low, the counter is preset to 0 so it counts from 0 to 15 (16 counts) and then overflows.

During the transmission of all other data characters which are 8 bits long signal SCRC (1) H is low and the counter MSB input is high. When the LD input goes low, the counter is preset to 8 so it counts from 8 to 15 (8 counts) and then overflows.

As previously mentioned, the counter is enabled by signal ENTCSC H which is controlled by the DP CROM. The counter is enabled (ENTCSC H is asserted) during each data character bit exclusive of stuffed 0s. When a 0 is to be stuffed, the DPCROM drives ENTCSC H low to inhibit the counter. The counter is clocked by the positive-going edge of TXCLK H which occurs once each bit time. When the counter finishes counting a character, it overflows and generates TCS MAX H. This signal is fed back via gating logic to the load input of the counter. When TCSC MAX H is asserted, it is ANDed with ENTCSC H at the 7408 AND gate. The output of this gate goes to a 7432 OR gate to generate LOAD H. This high signal is inverted and sent to the load input of the counter to preset it for the next data character count.

4.3.6.5 Transmitter Shift Register – The transmitter shift register (TXSR) is a 74165 parallel-load 8-bit shift register that serializes the information to be transmitted (Figure 4-16). This includes data, SDLC control characters, and DDCMP sync characters.

The TXSR inputs are TBUF 00 H–TBUF 07 H which are the outputs of the low byte of the TX data buffer register. The complementary serial outputs (TXSER OUT L and TXSER OUT H) are picked off input 6 which is the LSB of the TX data buffer register. The 8 bits in the TXSR are serialized starting with bit 0. The TXSR also has a serial input (pin 10) which is connected to pin 9 and is the true serial output (TXSER OUT H).

The clock inhibit input (pin 15) is disabled by connecting it to ground. This allows the TXSR to be clocked by a positive-going edge at its clock input (pin 2). This input is connected to the output of the 7408 AND gate. One input to this gate is ENTXSRC H which is asserted by the data decode ROM when data is to be transmitted. The other input is clock signal TCLKEG H which is a 250 ns positive pulse that occurs once each bit time.

When the TXSR load input (pin 1) is low, the clock is inhibited and the low byte of the TX data buffer is parallel-loaded into the TXSR. The load input is connected to the output of the 7400 NAND gate. One input to this gate is TBUF→SR H which is asserted by the data decode ROM when the TX data buffer is to be loaded. The other input is clock signal TCLKEG H. When both of these signals are asserted, the load input goes low.

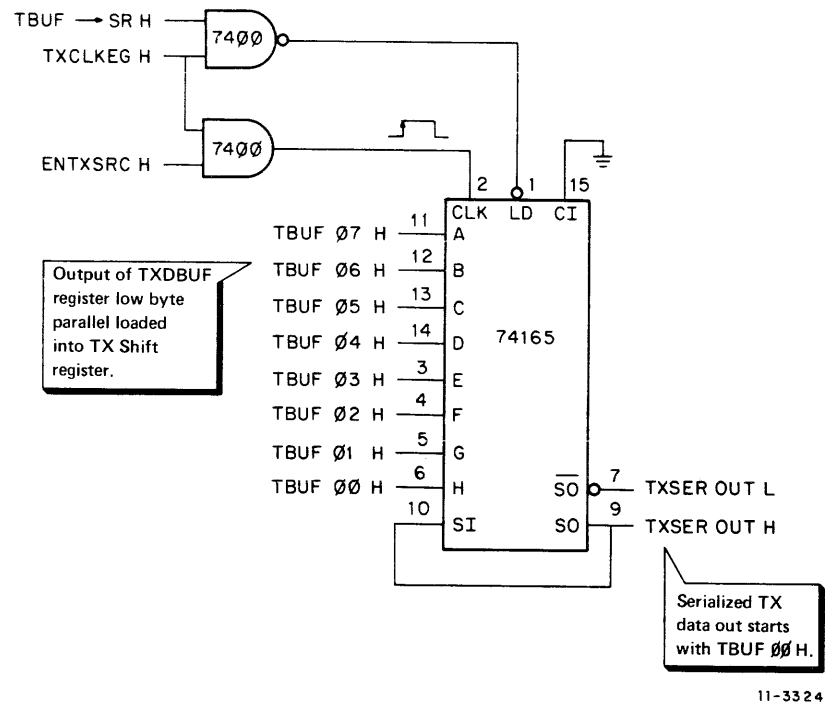


Figure 4-16 Transmitter Shift Register

If the program accesses the TXDBUF too late during the transition between sending a control character and a data character, signal SKP LD H inhibits the setting of TXDNE and allows an extra control character to be sent. The character that was late is sent after the extra control character. This prevents loss of a data character without an error indication. In the case of a SYNC character, it is circulated in the TX shift register.

4.3.7 Receiver Logic

The detailed discussion of the receiver logic is divided into five parts:

Discussion	Paragraph
ROMs and RX Control Flags	4.3.7.1
Clock Logic	4.3.7.2
Enable R1BC Flip-Flop and R1BC Counter	4.3.7.3
Character Serialization Counter	4.3.7.4
Shift Register and Data Buffer	4.3.7.5

4.3.7.1 ROMs and RX Control Flags – Two read-only memories (ROMs) are the major controlling elements for the receiver. Each one is a 1024-bit TTL ROM (74187) organized as 256 words of 4 bits each. These ROMs are interchangeable with 5603 PROMs. Both enabling inputs are held low to keep the ROM enabled constantly. The inputs represent an 8-bit binary-coded address that selects any one of the 256 words (addresses 0–255). The most significant input is pin 15 and the least significant input is pin 5. Each word is preprogrammed and is unalterable. When addressed, a specific word always produces the same states at the four outputs. As control elements, the ROMs act as compact logic arrays that replace a large amount of distributive logic.

The two ROMs are the decode ROM and the function ROM. Listings for the ROMs are contained in the print set. The listing contains input/output binary equivalents for each address along with a brief note of what the address represents. Many addresses form combinations of inputs that are functionally meaningless or that are not allowed. These addresses are defined as illegal. For the decode ROM, all illegal addresses generate output 0. For the function ROM, all illegal addresses generate outputs 0 or 1. Remember that in both of these ROMs there are addresses that generate a legitimate 0 and 1. The circuit schematic for the ROMs and associated logic is contained in logic sheet BSI3. A simplified diagram is shown in Figure 4-17.

Both ROMs have their enabling inputs (pins 13 and 14) enabled by signal GR TEST PT which is low when power is applied. GR TEST PT is the output of a 7900 NAND whose input is connected to +5 V. Signal GR TEST PT can be manipulated only by the module tester to simulate a disabled ROM.

In the following discussion, the source and destination of the signals associated with the ROMs and receiver flags flip-flop are described. Not all signals are described functionally. Some signals, specifically ROM inputs, have relevance only when viewed as a group during a certain point in a receive operation. This aspect is covered in the discussion of a typical receive operation in this chapter.

Decode ROM

The decode ROM responds to the program and to the receiver logic. The program controls two inputs: signal DEC MODE (1) H which is PARCSR bit 15 and SEC MODE (1) H which is PARCSR bit 12. Signal DEC MODE (1) H is high when PARCSR bit 15 is set and the DUP11 operates in accordance with the DDCMP or BISYNC protocols. Signal SEC MODE (1) H is high when PARCSR bit 12 is set and the DUP11 operates in the secondary station mode in accordance with the SDLC protocol. In this case, the DEC MODE bit (PARCSR bit 15) is cleared.

Input ADREC + SYNC H comes from the output of the receiver comparator logic. When operating in the SDLC or ADCCP secondary mode, this signal is high when the correct secondary station address is received. In the DDCMP mode, it is high when the desired sync character is received.

Input FRM (1) H comes from the receiver control flags flip-flop. Actually, it comes from the function ROM and is latched in the flip-flop.

The remaining four inputs R1BC 7 H, R1BC 6 H, R1BC 5 H, and R1BC 0 H come from the received 1s bit counter (R1BC) and are used by the decode ROM to differentiate between a flag character and an abort character.

Two outputs of the decode ROM (EN FRM H and ADRS + SYNC RCVD H) are sent directly to the function ROM. Output EN RSRC H goes to the shift register clock input logic and the RCSC counter output logic. The fourth output, FLG RCVD H, is ORed with STRIP SYNC (1) H as a function ROM input.

Function ROM

The program controls three inputs to the function ROM. Two are SEC MODE (1) H (PARCSR bit 12) and DEC MODE (1) H (PARCSR bit 15) which are discussed above in the decode ROM description. The third input is the ORed function FLG RCVD H + STRIP SYNC (1) H. Signal FLG RCVD H comes from the decode ROM. Signal STRIP SYNC (1) H is RXCSR bit 8. It is used to strip sync characters (after synchronization) when in the DDCMP or BISYNC protocols.

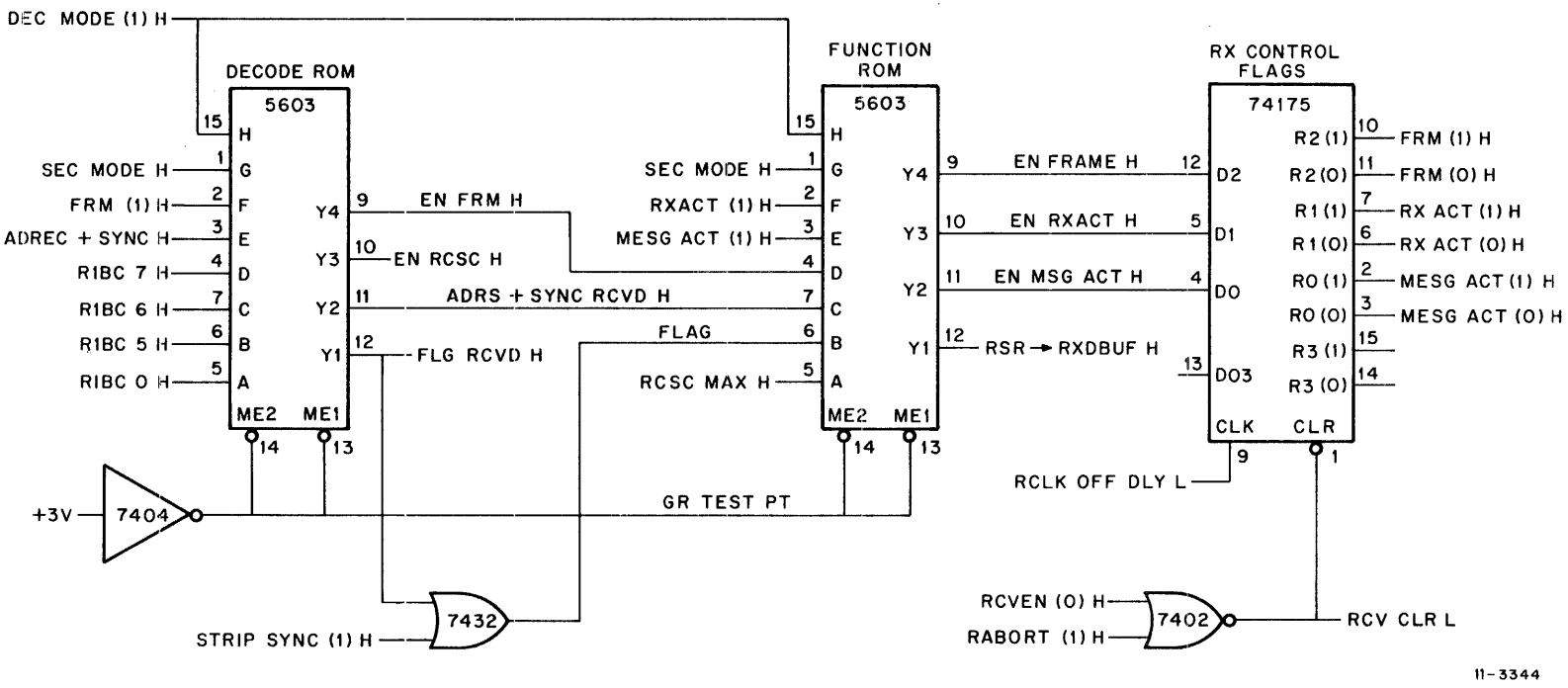


Figure 4-17 Receiver ROMs and Associated Logic

Inputs EN FRM H and ADRS + SYNC RCVD H come from the decode ROM. Input RCSC MAX H comes from the RSCS counter and is asserted when the last bit of a character is counted. Input MESG ACT (1) H comes from the output of this ROM via the receiver control flags flip-flop. Input RXACT (1) H is RXCSR bit 11. This signal comes from the output of this ROM via the receiver control flags flip-flop. It denotes the state of the receiver logic in accordance with the protocol selected and the mode of operation as determined by the PARCSR register.

Three outputs of the decode ROM are sent to the receiver control flags flip-flop. They are: FRAME H, EN RXACT H, and EN MESG ACT H. The fourth output, RSR → RXDBUF H is a qualifying signal for the clock input of receiver data buffer bits 0–7, 8, 9, 12, and 14. It is also used as a qualifying signal in the preset logic for the RXDONE flip-flop.

Receiver Control Flags Flip-Flop
The receiver control flags flip-flop is a 74175 quad D-type. It has four individual D-inputs with corresponding double-rail outputs. It has common clock and clear inputs. Only three sections are used. Inputs FRAME H, EN RXACT H, and EN MESG ACT H all come from the function ROM. These signals are latched into the flip-flop by clocking it with RCLK OFF DLY L once each bit time. Clocking occurs on the positive-going edge of this 70 ns negative pulse.

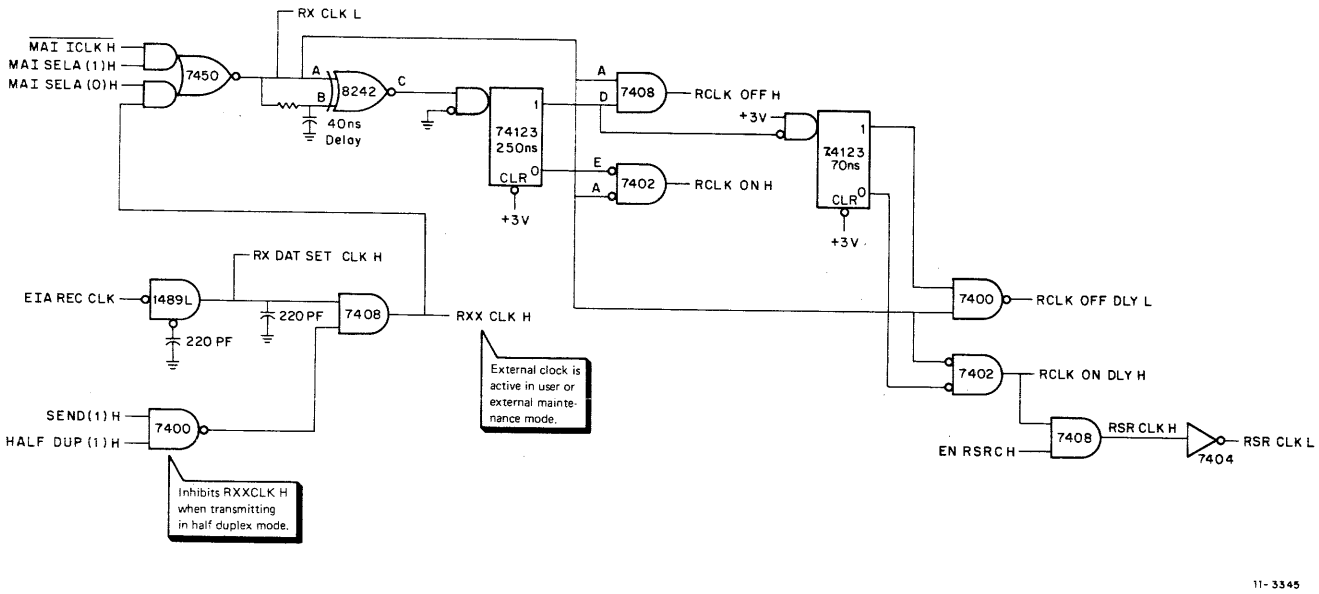
The outputs are FRM (1) H, RXACT (1) H, and MESG ACT (1) H and their complements. Signal FRM (1) H is used as an input to the decode ROM. Signal MESG ACT (1) H is used as an input to the function ROM and as the controlling input for the start of received message bit (RSOM) which is bit 8 of the RXDBUF register. Signal RXACT (1) H is RXCSR bit 11.

4.3.7.2 Clock Logic – (Refer to Figure 4-18.) The receiver clock is supplied by the data set. This logic uses the data set clock (RXDAT SET CLK H) or the maintenance clock (MAI ICLK H) to derive a group of clock signals for the receiver logic. In the user mode, the data set clock is gated into the logic. In the system test or internal maintenance modes, clock MAI ICLK H is gated into the logic. This is accomplished by using the states of the MAINT MODE SEL A bit (TXCSR bit 11) to gate the input clock signal through a 7450 dual AND-OR-invert gate.

The inputs to one AND section of the 7450 are $\overline{\text{MAI ICLK H}}$ and MAI SEL A (1) H. The inputs to the other AND section are RX XCLK H and MAI SELA (0) H. In the system test or internal maintenance mode, the MAINT MODE SEL A bit is set; therefore, MAI SELA (1) H is asserted which gates in the maintenance clock MAI ICLK H to the receiver clock logic. The other 7450 AND input is inhibited by MAI SELA (0) H which is low.

In the user mode, the MAINT MODE SEL A bit is cleared; therefore MAI SELA (0) H is asserted which gates in the data set clock RXCLK H. There are other qualifying conditions for the gating of this clock signal. Signal RXCLK H comes from a 7408 AND gate. One input is RXDAT SET CLK H which is the receive data set clock after it has been converted to TTL logic levels and inverted by a 1489 receiver. The other input is the ANDing of SEND (1) H and HALF DUP (1) H in a 7400 NAND gate.

SEND (1) H is the SEND bit (TXCSR bit 4) and is asserted only during transmitter operation. HALF DUP (1) H is TXCSR bit 3 and selects full- or half-duplex operation. (It is set to select half-duplex operation.) When transmitting in the half-duplex mode, both of these signals are asserted which inhibits the receiver clock by putting a low on one input of the 7408 AND gate. Disabling the receiver is necessary during half-duplex operation to prevent the receiver logic from copying the message being transmitted. This would increase the software overhead.



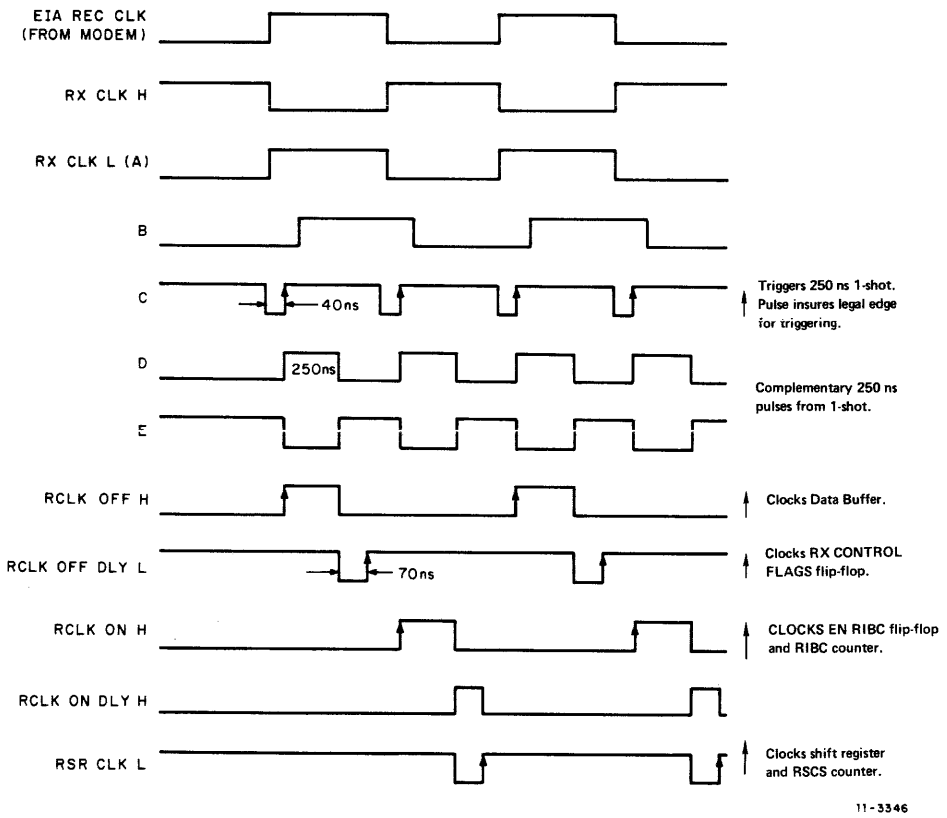


Figure 4-18 Receiver Clock Logic and Timing Diagram

The output of the 7450 is RXCLK L. Assume that we are in the user mode so that this signal represents the data set clock. RXCLK L is sent to one input of an 8242 exclusive-NOR gate. RXCLK L also goes through an RC delay network and to the other input of the 8242. This arrangement generates a 40 ns negative pulse at the output of the 8242 gate on each transition (negative or positive) of RXCLK L. The trailing edge (positive-going) of this pulse triggers the one-shot. The 40 ns delay provides the required setup time for the 74123 one-shot so that it can identify the trailing edge of the pulse as a legitimate positive-going transition. It also provides the timing necessary to set up the steering gates that are fed by the one-shots. When triggered, the one-shot produces complementary pulses of 250 ns duration. The negative pulse is ANDed with RXCLK L to generate RCLK ON H only on the positive-going edge of RXCLK H. The positive pulse is ANDed with RXCLK L to generate RCLK OFF H only on the negative-going edge of RXCLK H.

When the one-shot times out, the negative transition of its 1 output triggers a second one-shot that generates complementary pulses of 70 ns duration. The negative pulse is ANDed with RXCLK L to generate RCLK ON DLY H only on the trailing edge of RCLK OFF H. These pulses are shown in a timing diagram in Figure 4-18.

4.3.7.3 EN R1BC Flip-Flop and R1BC Counter – (Refer to Figure 4-19.) The R1BC counter counts consecutive 1s and is cleared when a received 0 is detected. The states of some of its outputs are used as inputs to the decode ROM. They are used to detect a stuffed 0, flag character, or an abort character. The EN R1BC flip-flop looks at the received data before it is shifted into the R1BC counter.

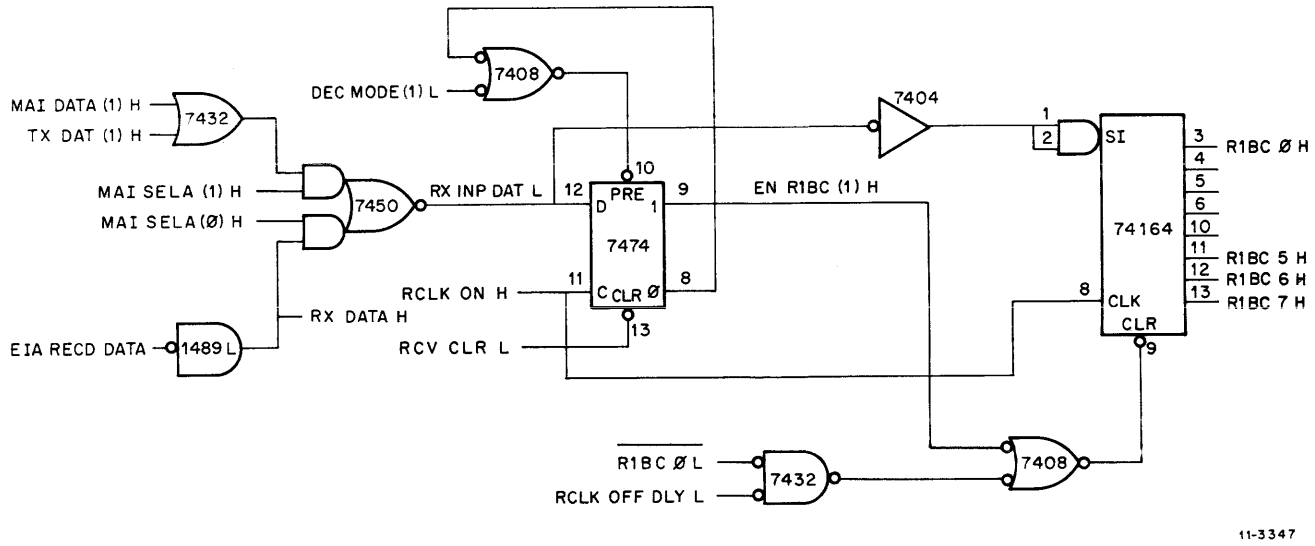


Figure 4-19 ENR1BC Flip-Flop and R1BC Counter

From the idle state, during SDLC protocol operation, the switch to active reception must start with a 0 to signal the first bit of the SDLC flag character (01111110). The EN R1BC flip-flop enables the R1BC counter only if this action occurs.

The input to the EN R1BC flip-flop is a 7450 dual AND-OR-invert gate. The inputs to one AND section of the 7450 are RX DATA H and MAI SELA (0) H. Signal RX DATA H is the received data after it has been converted to TTL logic levels and inverted by a 1489 receiver. In the user mode, the MAINT MODE SEL A bit is cleared; therefore MAI SELA (0) H is asserted which gates in the received data RX DATA H.

The other AND section inputs are MAI SEL A (1) H and the OR function MAI DATA (1) H + TXDAT (1) H. Signal MAI DATA (1) H is TXCSR bit 10 and is called maintenance input data. It is program read/write and is used as the serial data input to the receiver in the internal maintenance mode. In this mode, TXDAT (1) H goes low and MAI SEL A (1) H goes high. Under these conditions, serial data is single-stepped into the receiver using MAI DATA (1) H and TXCSR bit 13 which is the single-step maintenance clock.

The following definitions are provided to clarify the logic levels referred to in this document.

Outside the DUP11 (reference signal is RECEIVER DATA at input of level converter):

SPACE = H = +3 V = logical 0
MARK = L = -3 V = logical 1

Inside the DUP11 (reference signal is RX DATA H at output of level converter):

SPACE = L = 0 V = logical 0
MARK = H = +3 V = logical 1

Assume that the DUP11 is operating in the SDLC protocol user mode and that the line is in the idle state (sending MARKs). The EN R1BC flip-flop has been cleared directly by RCV CLR L but now the receiver enable bit has been set and RCV CLR L is high.

In the idle state (all MARKs), RXDAT H is asserted and keeps the 7450 output (RX INP DAT L) low. This signal goes to the D input of the EN R1BC flip-flop. The flip-flop is clocked by RCLK ON H once each bit time but it does not change state (remains cleared). The 7450 output is also inverted and sent to the serial input of the R1BC counter as RX INP DAT H. This counter is a 74164 8-bit parallel-out shift register with a clear output. It is clocked by RCLK ON H also; however, a 1 is not shifted in because the counter is held cleared by the low [EN R1BC (1) H] from the 1 output of the EN R1BC flip-flop.

Assume that the transmitting station starts to send a flag character (01111110). When the line goes to a 0, the D input (RX INP DAT L) of the EN R1BC flip-flop goes high and the serial input (RX INP DAT H) of the R1BC counter goes low. When RCLK on H goes high, both the flip-flop and the counter are clocked. Nothing happens to the counter (it remains cleared); however, the flip-flop is set. The high from its 1 output goes to one input of a 7408 AND gate (shown as the logically equivalent negated -OR gate). This drives the clear input (pin 9) of the R1BC counter high. The low from the 0 output of the EN R1BC flip-flop is fed back to its preset input (pin 10) via a 7408 gate. This locks the flip-flop in the set state until it is directly cleared by RCV CLR L going low. This occurs when an abort character is received or the program clears RCVEN.

The counter also removes stuffed 0s. Near the end of the bit time, pulse RCLK OFF DLY L is generated for 70 ns. It is ANDed with $\overline{\text{R1BC 0 L}}$, which is also low, at a 7432 OR gate (shown as the logically equivalent negated AND gate) to clear the counter. Nothing happens because the counter is already clear. The counter is always cleared when a 0 is detected and the RCLK OFF DLY L pulse is generated.

When the line goes to a 1 (the first 1 of the flag character), a 1 is shifted into the counter and R1BC 0 H goes high. $\overline{\text{R1BC 0 L}}$ is high also because it is the logical equivalent of R1BC 0 H. Now when RCLK OFF DLY L comes along, the counter is not cleared. Operating in this manner, the R1BC counter counts 1s to recognize a flag character (six consecutive 1s), an abort character (eight consecutive 1s) or a stuffed 0 (five consecutive 1s).

4.3.7.4 Character Serialization Counter – (Refer to Figure 4-20.) The character serialization counter (RCSC) counts the number of bits in a character, exclusive of stuffed 0s. At the last bit, it generates signal RCSC MAX H which goes to the function ROM as a control input to indicate that the RX data buffer should be loaded and RX DONE should be set to tell the program that a received character is ready for transfer.

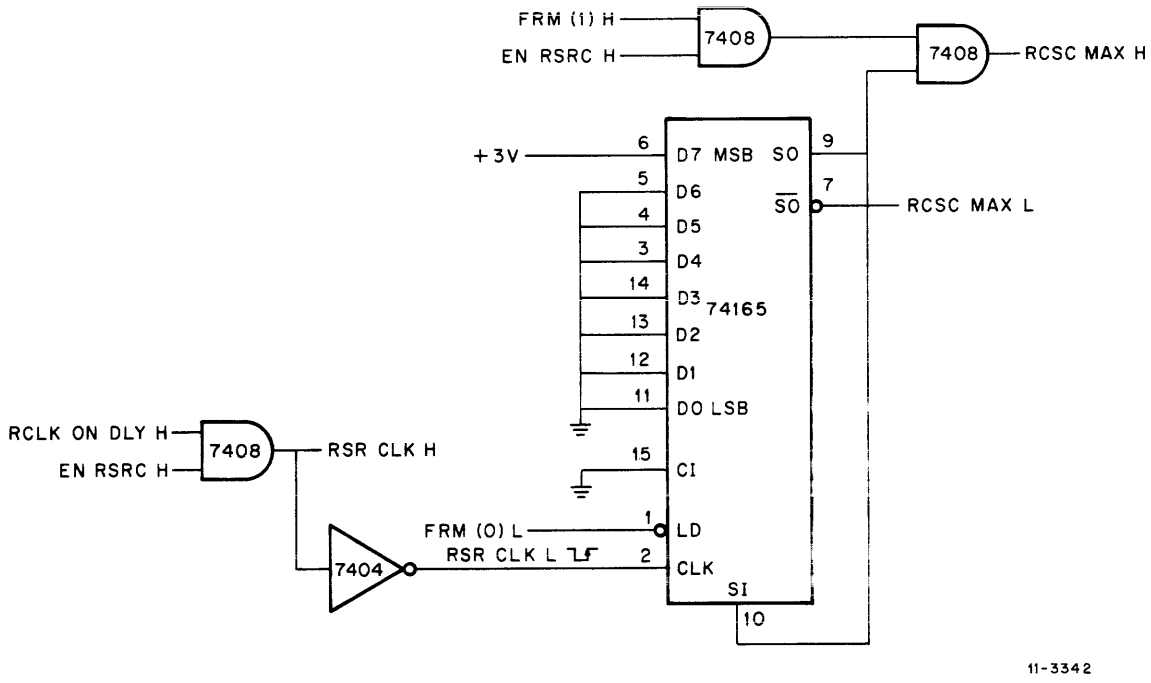


Figure 4-20 Received Character Serialization Counter

The counter is actually a 74165 parallel-load, 8-bit shift register. Its MSB is connected to +3 V and all the other inputs are connected to ground. When the load input (pin 1) is low (FRM cleared), the clock is inhibited and a 1 is loaded into the MSB position (pin 6). The clock inhibit input (pin 15) is disabled by connecting it to ground. This allows the RCSC to be clocked by a positive-going edge at its clock input (pin 2). Complementary serial outputs are picked off the MSB position. The true serial output (pin 9) is fed back to the serial input (pin 10). The serial input is fed to the LSB position (pin 11).

In operation, the loaded 1 is recirculated back through the counter and after eight bits are counted it resides in the MSB position again. At this time, it asserts RCSC MAX H which is used in the receiver function ROM to indicate the end of a received character. Details of this operation are described below.

Assume that the DUP11 is in the idle state under SDLC protocol discipline. The RX control flags flip-flop is cleared; therefore, signal FRM (1) H and its logical equivalent FRM (0) L are both low. Signal FRM (1) H is ANDed with ENRSRC H and the result, which is low, goes to one input of another AND gate. The other input of this gate is the true output (pin 9) of the RCSC counter. The output is RCSC MAX H which is high at present.

Clock signal RCLK ON DLY H is ANDed with ENRSCR H to generate RSR CLK H. This signal is inverted to generate RSR CLK L which is used to clock (shift) the RCSC counter each bit time. Later each bit time, clock signal RCLK OFF DLY H clocks the RX control flags flip-flop.

Assume that the DUP11 is in the idle state under SDLC protocol control. The RX control flags flip-flop is cleared; therefore, signal FRM (1) H and its logical equivalent FRM (0) L are both low. With FRM (0) L low, the RCSC counter is loaded with a 1 in its MSB input (pin 6) and 0s in all other inputs. Signals FRM (1) H and ENRSRC H, which are both low, are ANDed in a 7408 gate. The output of this gate goes to one input of another 7408 gate. The other input of this gate is the true serial output (pin 9) of the RCSC counter. The output of this gate is RCSC MAX H which is low.

The line goes active and an SDLC flag character is received. When the flag is recognized, the control logic asserts FRM (1) H. This releases the RCSC counter load function and activates the clock input. Signal EN RSRC H is also asserted at this time. With ENRSRC H and FRM (1) H both asserted, RCSC MAX H goes high. This action produces an illegal address at the function ROM. The data buffer is not loaded (RSR RXDBUF H not asserted).

When the first bit of the first data character is received, it is clocked into the shift register. The RCSC counter is also clocked. This moves the 1 from the MSB (pin 6) to the LSB (pin 11) which drives RCSC MAX H low. The character is assembled serially bit-by-bit in the shift register and the 1 in the RCSC counter is shifted toward the MSB position bit-by-bit. At the eighth bit, the 1 is in the MSB position and RCSC MAX H goes high. This signal goes to the function ROM which asserts RSR → RXDBUF H to load the assembled data character into the data buffer.

4.3.7.5 Receiver Shift Register and Data Buffer – (Refer to Figure 4-21.) The receiver shift register is loaded in serial form with an 8-bit character. The 8-bit parallel output of the shift register is loaded into bits 7–0 of the receiver data buffer register and then on to the PDP-11 system memory via the multiplexed bus selectors.

The shift register is a 74164 8-bit parallel-out serial shift register. Both serial inputs are connected to signal R1BC 0 H. This is the LSB of the R1BC counter and represents the state of the received information. The register is cleared by RCV CLR L which goes low when the receiver enable bit is cleared by the program [signal $\overline{\text{RCVEN}}$ (0) H is high] or the abort bit is set by the program [signal RABORT (1) H is high]. The clock signal (RSR CLK) is the AND function of ENRSRC H and RCLK ON DLY H. With ENRSRC H enabled, each negative transition of RCLK ON DLY H shifts in a bit of the received character.

The outputs of the shift register are RSR 07 H – RSR 00 H and are sent to bits 07–00 of the receiver data buffer register. These bits are contained in one 74174 hex flip-flop and two sections of a second 74174 and are called the data buffer. Signal FRM (0) L clears the data buffer whenever the RX control flags flip-flop is cleared (during abort or at end of message) or when the function ROM does not assert EN FRAME H.

The data buffer is clocked by signal LD RXBF H which is the AND function of RCLK OFF H and RSR RXDBUF H. Signal RSR → RXDBUF H is generated by the function ROM when it is time to load a character into the data buffer. The outputs of the data buffer (RBUF BIT 7 H – RBUF BIT 0 H) are sent to the multiplexed bus selectors and then to Unibus data lines D (07:00) for transfer to the PDP-11 system memory.

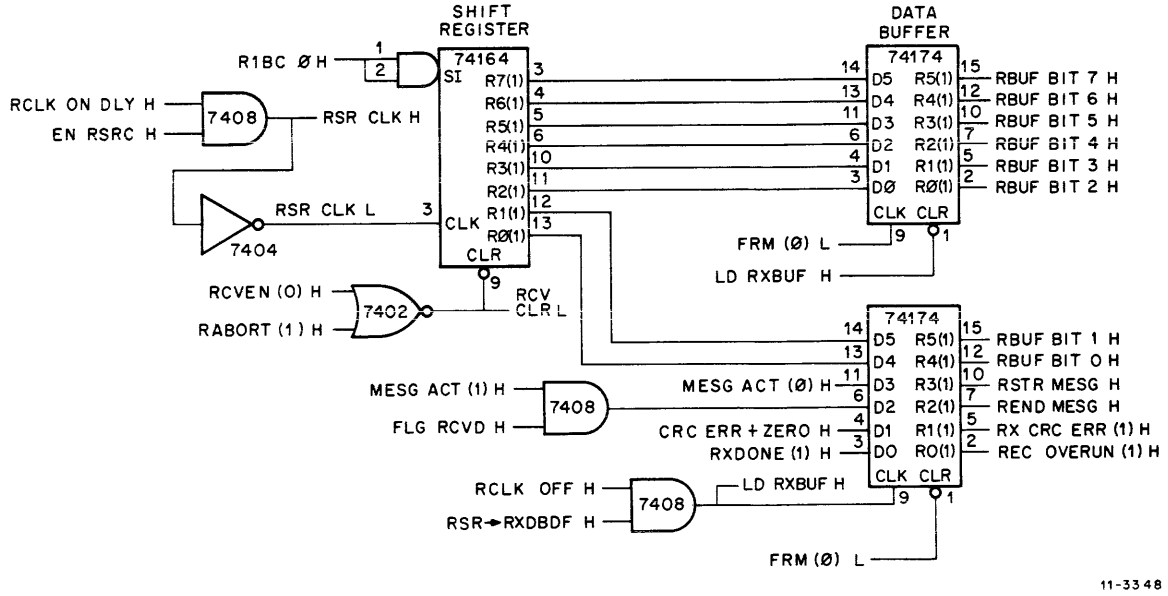


Figure 4-21 Receiver Shift Register and Data Buffer

4.3.8 CRC Logic

4.3.8.1 General – This discussion covers the operation of the transmitter and receiver cyclic redundancy checking (CRC) logic. Functionally, the logic is divided into four sections that are discussed in the order shown below.

1. Error detection logic
2. Transmitter CRC register
3. Receiver CRC register
4. Typical transmitter and receiver CRC computations

Chapter 1 contains background information on cycle redundancy checking.

4.3.8.2 Error Detection Logic – The error detection logic is used only by the receiver. Both the sending and receiving stations must have their CRC logic enabled. The sending station computes a CRC character for the message and transmits it at the end of the message.

In the SDLC mode, the DEC MODE bit (PARCSR bit 15) is cleared. The receiver CRC register receives the message and the CRC character which is called the Frame Check Sequence (FCS). At this point, the receiver CRC register must contain the octal value 016417. If it does not, the error detection logic asserts signal CRC ERROR + ZERO H which indicates that the received message contains one or more errors. CRC ERROR + ZERO H is bit 12 of the RXDBUF register. This logic only indicates that the received message is in error; it does not determine the number or location of the errors, nor does it have an error correcting capability. In response to the error flag, the program requests that the message be retransmitted.

In the DEC mode, the receiver CRC register receives the message and CRC character. At this point, the contents of the receiver CRC register must be zero. If so, the error detection logic asserts CRC ERROR + ZERO H to indicate that the register contains zero and the message is errorless. If an error is present (register contents not zero), CRC ERROR + ZERO H is not asserted. This procedure is used because in DEC mode the CRC character is the last character in the message and is not followed by a flag character to locate it (as in SDLC mode). The DUP11 has no way of counting message characters, therefore, the arrival of the CRC character cannot be predicted by the DUP11 logic. Instead, the error detection logic asserts CRC ERROR + ZERO H any time the receiver CRC register reads zero at the end of a character. It is left to the program to check the register at the correct time for the presence of this indication.

Operation of the error detection logic in both DEC mode and SDLC mode is discussed below.

NOTE

To facilitate the use of different CRC codes in the DDCMP and SDLC protocols, the designations of logical 0 and 1, with respect to the CRC register, are different in each protocol. They are as follows:

	DDCMP	SDLC
Logical 0	Low	High
Logical 1	High	Low

The CRC error detection logic is shown in logic sheet BSI4 of the print set. It is also shown in Figure 4-22 and the gates are identified with letters and numbers for convenience in following the discussion.

Example 1 - DDCMP Mode With No CRC Error

In this example, assume that the message is errorless, the CRC character has been received, and the receiver CRC register reads zero.

Remember that in DDCMP mode a logical 0 is a low and when the receiver CRC register reads zero at the end of the current character, signal CRC ERR + ZERO H is asserted to show that there is no error.

Receiver CRC register bits 4–7, 9, 13, and 14 go to NAND gate A and negated-input AND gate B. Bits 1–3, 8, and 10–12 go to negated-input AND gate E. All these bits are low; therefore, the output of B is high and the output of E (ALWAYS ZERO H) is high. The output of A is high. For this discussion, the output of A can be ignored. Bits 0 and 15 are ANDed at gate C which drives its output high. The high outputs of B, C, and E go to 4-input NAND gate F. The fourth input is DEC MODE (1) H which is high because PARCSR bit 15 (DEC MODE) is set during DDCMP protocol operation. With all four inputs high, the output of F goes low. This signal goes to negative OR gate K whose output goes low and is sent to negated-input AND gate L. The path to the other input of L comes from gates G and H. The output of negated-AND gate G is high because its three inputs are low as shown below.

1. MESG ACT (1) L is low because MESG ACT remains set through the duration of the CRC character.

2. NO CRC (0) L is low because the NO CRC bit is cleared (CRC is active).

3. RCSC MAX L is low because it indicates the end of the CRC character. (In DEC mode, the status of the receiver CRC reqister is checked for zero at the end of each character.)

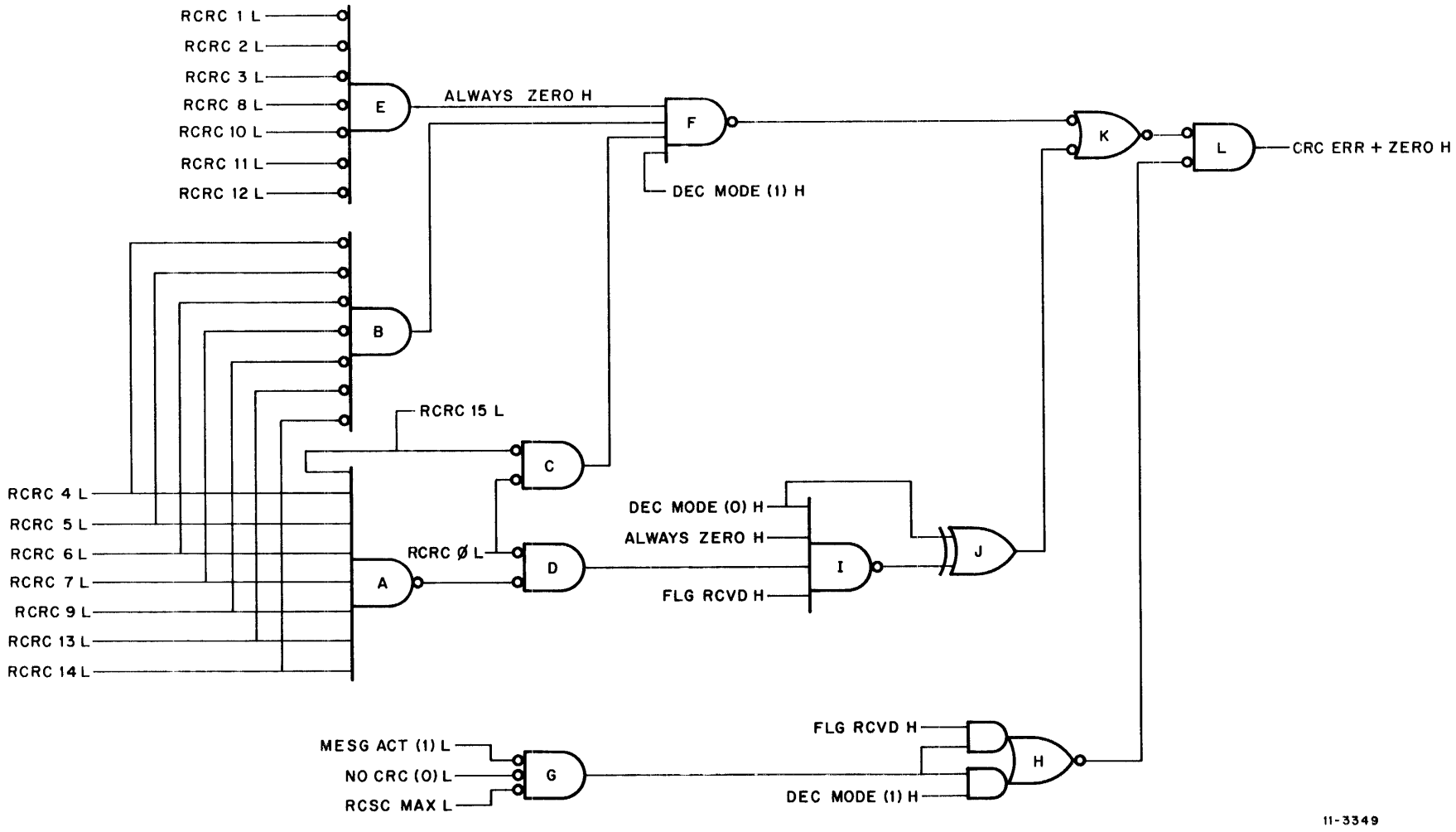


Figure 4-22 CRC Error Detection Logic

The high from G is ANDed with DEC MODE (1) H which is asserted to drive the output of gate H low.

With both inputs to gate L low, signal CRC ERR + ZERO H is asserted at its output. This signal is loaded into the RXDBUF register and presented to the program along with the last byte of the CRC character.

In summary, the message and CRC character have been received and the receiver CRC register reads zero which means that the message is errorless (CRC ERROR + ZERO H is not asserted).

Example 2 - DDCMP Mode With Message Error
If the receiver CRC register is not zero after the CRC character is received, it means that at least one bit of the accumulated CRC character is a 1 (high) due to an error in the message. This means that at least the output of one of three gates (B, C, or E) is low. This drives the output of F high; therefore, one input of negative OR gate R is high. The other input of this gate is held high because DEC MODE (0) H and FLG RCVD H are not asserted during DDCMP mode. With both inputs of K high, its output goes high and drives the output of gate L low (CRC ERROR + ZERO H is not asserted). Therefore, if the program checks the status of the receiver CRC register flag at this time and finds CRC ERROR + ZERO H not asserted, it concludes that the message contains an error.

Example 3 - SDLC Mode With No Error
In this example, assume that the message is errorless, the CRC character has been received, and the receiver CRC register reads octal 016417 (rightmost digit is LSB). When the receiver CRC register reads 016417, signal CRC ERR + ZERO H is not asserted to show that the message is errorless. Remember that in the SDLC mode, a logical 0 is a high and a logical 1 is a low. Converting 016417 to a 16-bit binary number reveals that bits 12, 11, 10, 8, 3, 2, 1, and 0 each are logical 1 or low; and bits 15, 14, 13, 9, 7, 6, 5, and 4 each are logical 0 or high.

With this bit configuration, the output of gate A is low and the output of gate E (ALWAYS ZERO H) is high. The output of A is ANDed with RCRC 0 H (which is low also) at D and its output goes high. This signal is one input to 4-input NAND gate I. The other three inputs are high also as shown below.

1. DEC MODE (0) H is high because the DEC MODE bit is cleared during SDLC protocol operation.

2. ALWAYS ZERO H is high because it is the output of gate E which tests for 0 bits.

3. FLG RCVD H is high because the final flag has been received. This terminating flag character correctly positions the received CRC character in the receiver CRC register.

With all four inputs high, the output of I is low. This signal goes to one input of X-OR gate J. The other input to this gate is DEC MODE (0) H which is high. With unlike inputs, the output of J goes high and is sent to one input of negative OR gate R. The other input to R is high also because DEC MODE (1) H is low. This drives the output of K high and holds the output of gate L low to prevent the assertion of CRC ERR + ZERO H.

In summary, the message and CRC character have been received and the receiver CRC register reads 016417 which means that the message is errorless (CRC ERR + ZERO H is not asserted).

Example 4 - SDLC Mode With Message Error
If the receiver CRC register is not 016417 after the CRC character is received, it means that at least one bit of the accumulated CRC character is in error due to a message error. Assume that bit 3 is logical 0 (high) instead of logical 1 (low). This makes the output of gate E low. ALWAYS ZERO H is not asserted and it is sent to I which holds its output high. This signal goes to one input of J. The other input of this gate is DEC MODE (0) H which is also high. With identical inputs, the output of X-OR gate J is low. This drives the output of K low which puts a low on one input of gate L. The path to the other input of gate L comes from gates G and H. The output of G is high because its three inputs are low as shown below.

1. MESG ACT (1) L is low because MESG ACT remains set through the duration of the CRC character and the following flag character.

2. NO CRC (0) L is low because the NO CRC bit is cleared (CRC is active).

3. RCSC MAX L is low because it indicates the end of the current character (terminating flag) at which point the receiver CRC register is checked for 016417.

The high from G is ANDed with FLG RCVD H which is asserted because the receiver control logic has received the terminating flag character. This drives the output of gate H low. With both inputs to gate L low, signal CRC ERROR + ZERO H is asserted at its output. This signal is loaded into the RXDBUF register and becomes RXCRC ERR (1) H. At this point, both CRC bytes have been presented to the program. The error flag now appears in the RXDBUF register along with the assertion of REND MESG H.

In summary, the receiver CRC register did not contain 016417 after reception of the message and CRC character. When the program checks the status of the receiver CRC register flag at this time and finds CRC ERROR + ZERO H asserted, it concludes that the message contains an error.

4.3.8.3 Transmitter CRC Register

General
The transmitter CRC register consists of a 16-bit shift register, input/feedback control logic, and the appropriate number of X-OR gates to operate with the selected CRC codes. Two codes are used: CRC-16 for the DDCMP protocol and CRC-CCITT for the SDLC protocol. CRC-16 requires three X-OR gates; one each for the input (bit 0), bit 2, and bit 15. CRC-CCITT also requires three X-OR gates; one each for the input (bit 0), bit 5 and bit 12. A total of five X-OR gates are used in the register. The one for the input (bit 0) is used by both protocols. For a given CRC code, the input/feedback control logic selects the required X-OR gates to provide the feedback path. The other X-OR gates act as non-inverting gates. For example, in the SDLC mode, the X-OR gates for the input (bit 0), bit 5 and bit 12 provide X-OR functions. The X-OR gates for bits 2 and 15 act as non-inverting gates and only shift data from one stage to the next.

The transmitter CRC register is shown in the print set. It is also shown in Figure 4-23. In this illustration, the register stages are shown symbolically as numbered squares. Actually, they are D-type flip-flops that are clocked simultaneously and cleared simultaneously. The clock signal is CRC CLK H which is the AND function of CRC CLK EN H and TXCLKP H. Signal CRC CLK EN H is asserted by the transmitter data decode ROM when the CRC function is enabled. TXCLXP H is a 300 ns pulse from the transmitter clock logic that occurs once each bit time. The clear signal is TXCRC INT L and when it is low it clears the TCRC register. When cleared, the register reads all 0s in the DDCMP mode and all 1s in the SDLC mode. Initializing the TCRC register to all 1s in the SDLC mode provides detection of the addition or delection of 0s at the leading edge of the message due to erroneous flag characters.

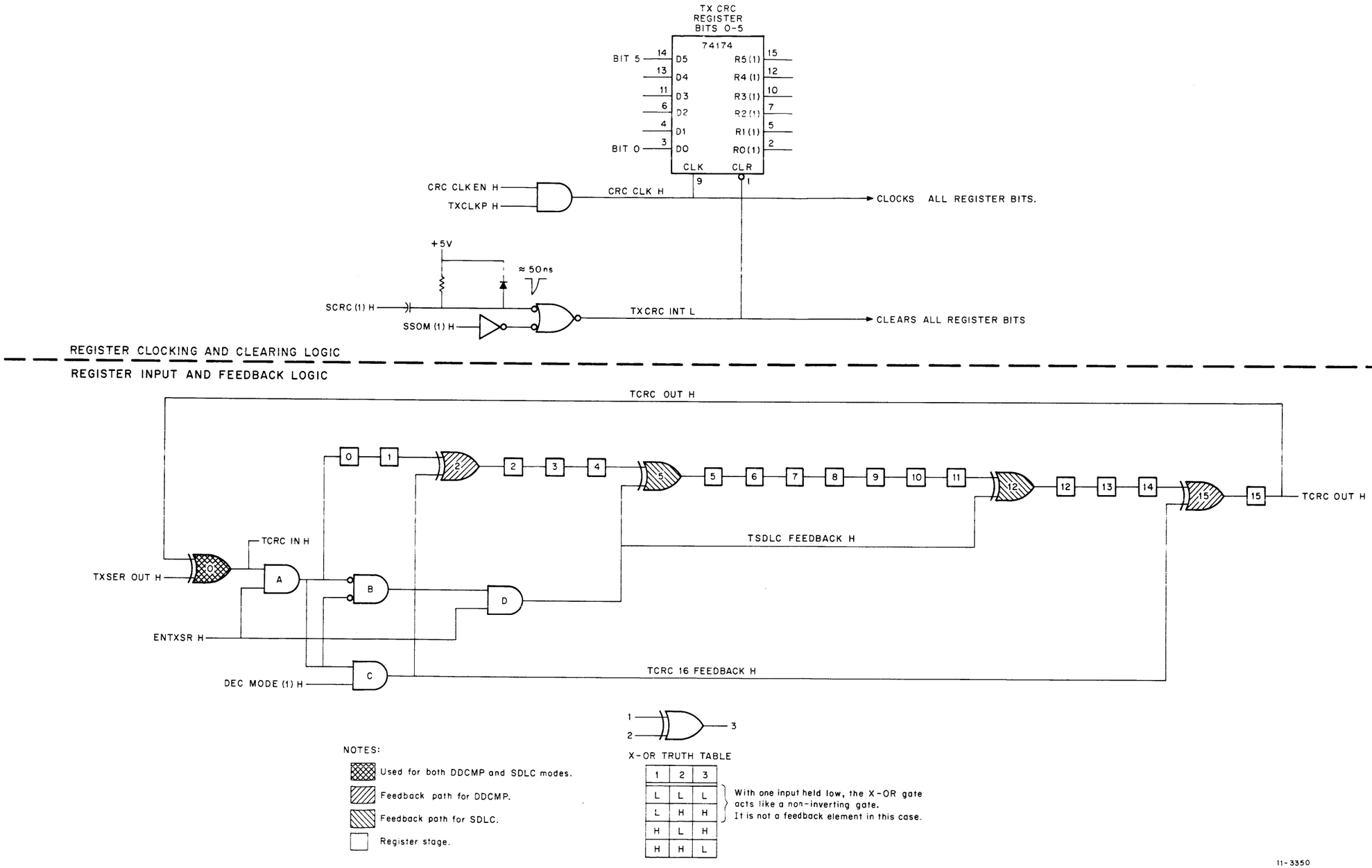


Figure 4-23 Transmitter CRC Register

Clear signal TXRC INT L is generated in two ways. At the start of a message, the program sets TSOM which results in signal SSOM (1) H being asserted by the bit sync buffer in the transmitter logic. This signal is inverted to generate TXCRC INT L.

After the accumulated 16-bit CRC character has been transmitted, the transmitter logic clears signal SCRC (1) H. When SCRC (1) H goes low, the AC-coupled edge detector produces a negative pulse with a duration of 50–90 ns that generates TXCRC INT L. This function is used in the DDCMP mode to clear the TCRC register after the CRC character has been transmitted. In the DDCMP mode, it is necessary to force clear the register at this time because the next character may be part of a sequence that requires computation of another CRC character.

Operation of the input/feedback control logic in both modes is discussed using Figure 4-23 as reference.

Input/Feedback Control Logic (DDCMP Mode)
In the DDCMP mode, the DEC MODE bit is set by the program; therefore signal DEC MODE (1) H is asserted. This put a high on one input of AND gate C and negated-input AND gate B (Figure 4-23). The other input of gates B and C is the output of AND gate A which is the AND function of ENTXSR H and TCRC IN H. Signal ENTXSR H is asserted by the transmitter data path control ROM when the CRC function is enabled. When the CRC character is being transmitted, ENTXSR H is held low which means that the input to the CRC register is low (logical 0) at this time.

Signal TCRC IN H comes from X-OR gate 0 and is the X-OR function of TXSER OUT H and TCRC OUT H. Signal TXSER OUT H is the output of the transmitter shift register and is the data being transmitted. Signal TCRC OUT H is the output of the CRC register (bit 15). The X-ORed states of TCRC IN H are sent to the first stage (bit 0) of the register. They also pass through gates A and C to X-OR gates 2 and 15. The output of gate B (and hence the output of gate D) is held low by DEC MODE (1) H. This inhibits the X-OR function of X-OR gates 5 and 12 and they act as non-inverting gates.

In summary, the CRC register starts cleared (all 0s) by the assertion of SSOM (1) H. The state of DEC MODE (1) H enables X-OR gates 2 and 15 which sets up the register to operate with code CRC-16. Each bit to be transmitted (TX SER OUT H) is X-ORed with the state of CRC register bit 15 (TCRC OUT H). The result is fed to the input of the first CRC register stage (bit 0). It is also sent to X-OR gates 2 and 15. At X-OR gate 2, this feedback signal is X-ORed with the output of register bit position 1. At X-OR gate 15, this feedback signal is X-ORed with the output of register bit position 14. X-OR gates 5 and 12 act as non-inverting gates so all bit positions except 0, 2, and 15 receive the data from the previous position in a straight shift operation. The feedback path is set up by the state of TCRC IN H prior to the register being clocked. When all the transmitted message data has been operated on, the register contains the CRC character. At this point, the transmitter control logic drives ENTXSR H low and the CRC character is transmitted immediately after the last data character. Remember that while the data is being operated on by the CRC register to accumulate a CRC character, it is being transmitted simultaneously without alteration. The CRC character is transmitted by being serially shifted from the output (TCRC OUT H) of the transmitter CRC register to the Data Decode ROM. All X-OR gates are disabled and the existing data is not modified. When the CRC character has been transmitted, the high-to-low transition of SCRC (1) H clears the transmitter CRC register. The only contribution that the CRC register makes to the transmitted data is the CRC character which the receiving station uses to determine whether or not the message has been received errorless.

Input/Feedback Control Logic (SDLC Mode)
In the SDLC mode, the DEC MODE bit is cleared by the program. With DEC MODE (1) H now low, gate B is qualified and gate C is disqualified. Qualification of B, with ENTXSR H asserted, sets up the feedback path for operation with code CRC-CCITT by enabling X-OR gates 5 and 12. The low output from C goes to X-OR gates 2 and 15 and they operate as noninverting gates.

The CRC register starts cleared (all 1s in the SDLC mode). All bit positions except 0, 5, and 12 receive the data from the previous stage of the TCRC register without modification by the X-OR function. Except for the change in the feedback path, operation is the same as that described in the DDCMP example. The transmitter control logic complements the CRC character before sending it. This is a requirement of using CRC-CCITT in the SDLC mode to ensure that the received errorless message results in a unique non-zero remainder at the receiver. This allows detection of the erroneous addition or deletion of 0s at the trailing edge of the message due to errors.

4.3.8.4 Receiver CRC Register – The configuration of the receiver CRC register is exactly like the transmitter CRC register with respect to the X-OR feedback paths. The input/feedback control logic, clocking logic, and clearing logic are different. The receiver CRC register is shown in logic sheet BSI4 of the print set. It is also shown in Figure 4-24.

The clock signal for the receiver CRC register is RSR CLK L which is the inversion of the AND function of EN RSRC H and RCLKON DLY H. Signal EN RSRC H is asserted by the receiver decode ROM when information is being received. Signal RCLKON DLY H is a 70 ns pulse from the receiver clock logic that occurs once each bit time.

The register is cleared by CLR RCRC INIT L which is the output of a 7450 AND-OR-invert gate. Two signal paths are available to generate the clearing signal.

In the SDLC mode, the DEC MODE bit is cleared and signal DFC MODE (0) H is high. At the start of the message, the RXACT bit is cleared so signal RXACT (0) H is high. This drives the output of the 7450 low and clears the CRC register (all bits are 1s). After the first data bit is received, the RXACT bit is set. Signal RXACT (0) H goes low and inhibits the register clear input.

In the DEC mode, SYNC characters are clocked into the receiver CRC register. However, they are not included in the CRC computation. This means that the CRC register must be cleared after every SYNC character and the clear signal must be inhibited at the end of the last SYNC character. The RXACT bit starts in the cleared state and is set after the last SYNC character is received. With the RXACT bit cleared, signal RXACT (0) H is high and is sent to the 7450 gate. At the last bit of the SYNC character, the RCSC counter asserts RCSC MAX H. This signal is ANDed with RCLK ON H which drives the 7450 output low and clears out the SYNC character that had been shifted into the CRC register. At the beginning of the first data character, the RXACT bit is set. This drives signal RXACT (0) H low and inhibits the register clear input. For subsequent messages, this register is cleared after the program clears RCVEN.

Operation of the input/feedback control logic in both modes is discussed using Figure 4-24 as reference.

Input Feedback Control Logic (DDCMP Mode)
In the DDCMP mode, the DEC MODE bit is set by the program; therefore, signals DEC MODE (1) H and DEC MODE (0) L are both high. Signal DEC MODE (1) H goes to gate C and when it is high allows the X-OR function to be performed by X-OR gates 2 and 15. This sets up the feedback path for code CRC-16 in the DDCMP mode. Signal DEC MODE (0) L goes to gate D and when it is high holds the output of D low which makes X-OR gates 5 and 12 perform as non-inverting buffers.

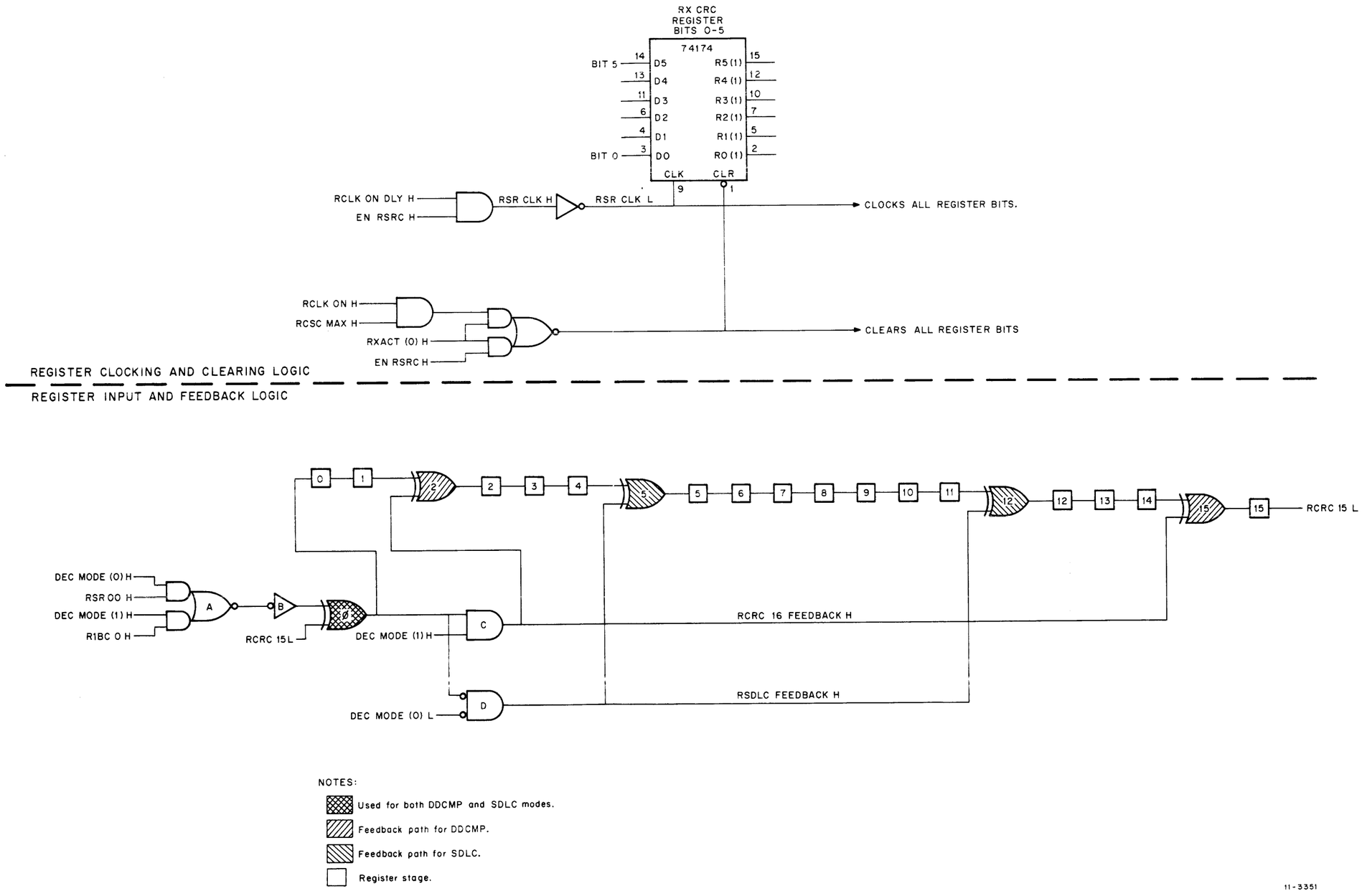


Figure 4-24 Receive CRC Register

Signal DEC MODE (1) H is also ANDed with R1BC 0 H at gate A whose output is inverted and sent to X-OR gate 0. This gate performs the X-OR function of bit 15 of the receiver CRC register (RCRC 15 L) and the received data (R1BC 0 H). The output of X-OR gate 0 goes to the input of the register (bit 0) and to gate C and hence to X-OR gates 2 and 15 as the feedback path. Signal R1BC 0 H is bit 0 of the R1BC counter. It is used as the received data input rather than bit 0 of the shift register to ensure that the last eight bits of the received CRC character are included in the CRC computation as they are received, which eliminates the need of trailing PAD characters to position the CRC character. In the DDCMP mode, the CRC character is the last character of the message. In an errorless message, the CRC ERR + ZERO bit is asserted and presented to the program along with this character. In the SDLC mode, the received data is picked off bit 0 of the shift register because a flag character follows the CRC character which ensures that the last eight bits of the received CRC character are included in the CRC computation as the flag character is received. This flag character acts to position the CRC character in the RCRC register.

In summary, the state of the DEC MODE bit conditions the input logic to set up the feedback path to conform to code CRC-16. All data and the received CRC character are included in the CRC computation. At the end of the message, the receiver CRC register should read all 0s. This indicates reception of an errorless message.

Input Feedback Control Logic (SDLC Mode)
In the SDLC mode, the DEC MODE bit is cleared by the program. With DEC MODE (0) L now low, X-OR gates 5 and 12 are activated via gate D. With DEC MODE (1) H now low, X-OR gates 2 and 15 are biased to act like non-inverting buffers via gate C. This sets up the feedback path for code CRC-CCITT in the SDLC mode.

Signal DEC MODE (0) H, which is high, is ANDed with RSR 0 H at gate A. Signal RSR 00 H is the output of bit 0 of the shift register and is the received data input to the receiver CRC register. It is double-inverted by gates A and B and sent to X-OR gate 0 whose other input is RCRC 15 L. The output of X-OR gate 0 goes to bit 0 of the CRC register and to X-OR gates 5 and 12 in the feedback path.

In summary, the state of the DEC MODE bit conditions the input logic to set up the feedback path to conform to code CRC-CCITT. All data and the CRC character (called FCS character in SDLC protocol) are included in the CRC computation. The sending station complements the FCS character before transmitting it. After receiving all the data and the FCS character, the receiver CRC register must read 016417 to indicate an errorless message.

4.3.8.5 Typical CRC Accumulation – Figure 4-25 shows typical transmit and receive CRC accumulations in the SDLC. Remember the following facts concerning CRC operation in the SDLC mode:

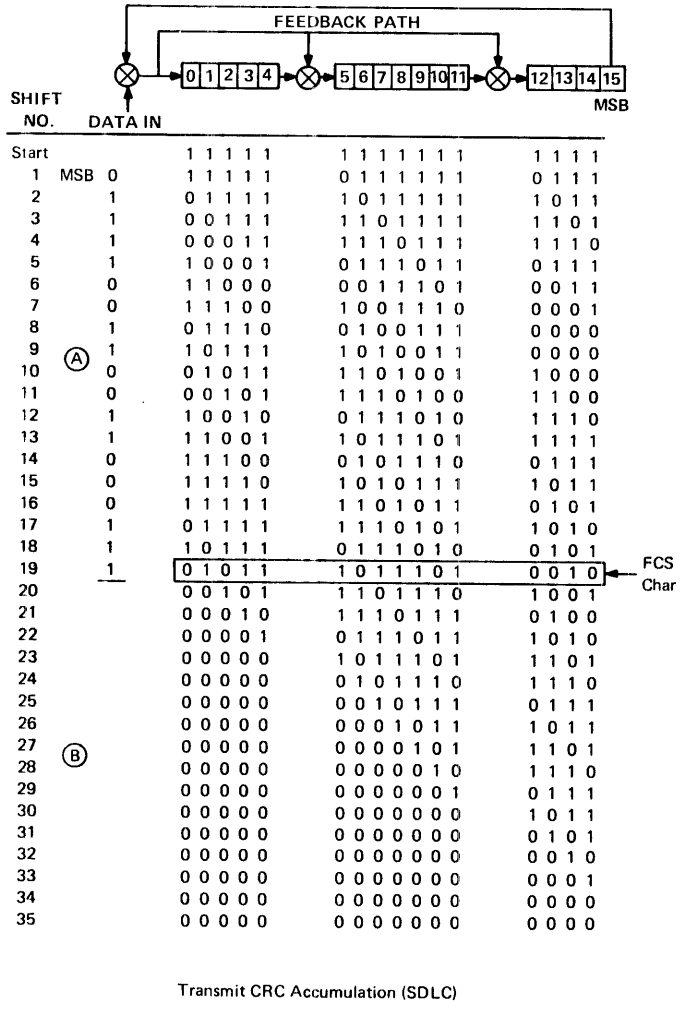
1. In the receiver and transmitter CRC registers, a high signal represents logical 0 and a low signal represents logical 1.

2. Both registers start cleared (all 1s).

3. After the CRC check character (FCS) has been accumulated in the transmit mode, it is transmitted in complementary form.

4. In the receive mode, after reception of the data and FCS character, the receiver CRC register must read octal 016417 (LSB right-justified) or else the message is in error.

In Figure 4-25, the receiver and transmitter registers are shown in symbolic form with the feedback path and X-OR functions identified. Each stage is numbered and the logical state of each stage is shown directly under it. A 19-bit data word is used.

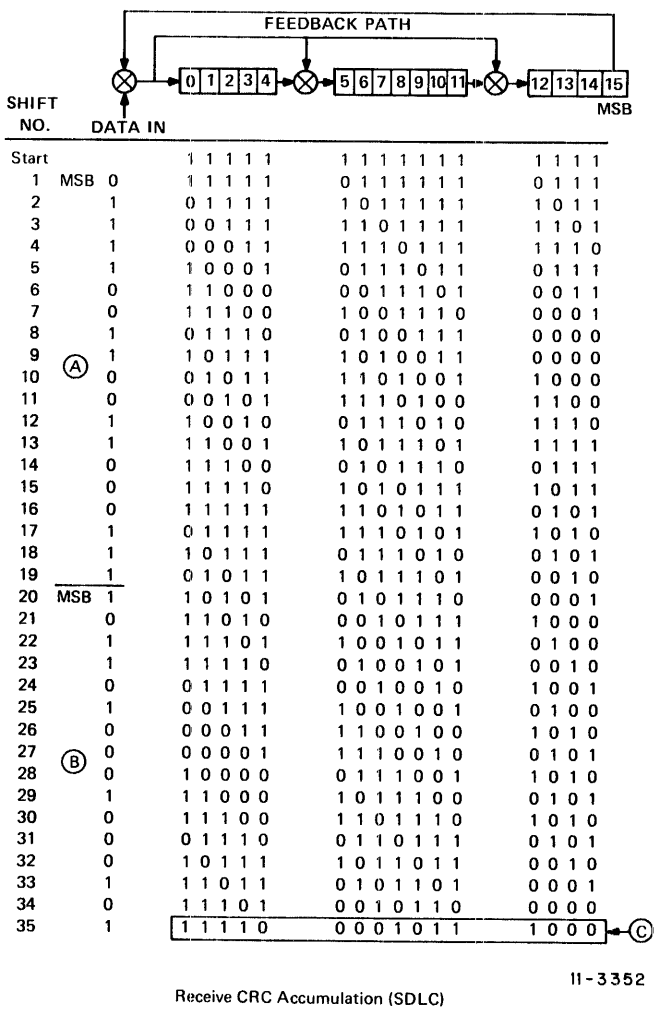


Transmit CRC Accumulation (SDLC)

NOTES

(A) Transmission of 19 bit data character showing accumulated FCS character.

(B) Transmission of 16 bit FCS character showing 0s being shifted into register.



Receive CRC Accumulation (SDLC)

NOTES

(A) Reception of 19 bit data character.

(B) Reception of complement of FCS character.

(C) Contents of register at end of message. Contents equal 016417_8 (LSB right justified).

Figure 4-25 Typical Transmit and Receive CRC Accumulation

4.3.9 Interrupt Control Logic

4.3.9.1 General – The interrupt control logic is functionally equivalent to the BR half of the M7821 Interrupt Control module including the NPR latency time improvement feature.

Physically, there are some differences between the DUP11 interrupt control logic and the M7821 module:

1. The DUP11 uses switches rather than jumpers to select the states of bits 3–8 of the vector address.

2. There is no switch associated with bit 2 of the vector address. Its state is controlled by the interrupt control logic to allow generation of two vector addresses (XX0 and XX4).

3. The interrupt request section of the logic contains two request flip-flops (REQ A and REQ B) that condition a third flip-flop (V2) which controls the state of vector address bit 2. REQ A is associated with receiver interrupts and REQ B is associated with transmitter interrupts. If REQ A is requesting, the vector address (octal) is XX0; if REQ B is requesting, the vector address is XX4. The two most significant digits (XX) are determined by switches in vector address lines 3–8.

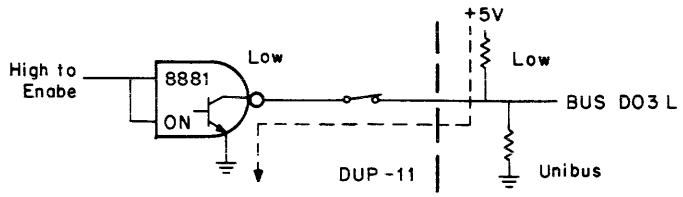
4.3.9.2 Generation of the Vector Address – Every hardware device capable of interrupting a PDP-11 processor has a unique set of memory locations (two consecutive words) reserved for its interrupt vector. The first word contains the location of the device's service routine, and the second, the Processor Status Word that is to be used by the service routine.

Communications devices are assigned floating-vector addresses. This eliminates the necessity of assigning addresses absolutely for the maximum number of each device that can be used in a system. The floating-address space starts at location 300 and proceeds upward to 777 (locations 500–534 are reserved). The devices are assigned in order by type. (See Chapter 2.)

Each device interrupt vector requires four memory locations (two words) which means only even-numbered addresses ending in 0 or 4. The vector address is specified as a three-digit, binary-coded octal number using Unibus data bits D(08–00). Because the vector must end in 0 or 4, bits D01 and D00 are not specified (they are always 0) and bit D02 determines the least significant octal digit of the vector address (0 or 4). The DUP11 interrupt control logic sends only seven bits (D08–D02) to the PDP-11 processor to represent the vector address (XX0 or XX4). When DUP11 interrupt requests are being serviced, the processor status level should be set equal to or higher than the priority level of the DUP11.

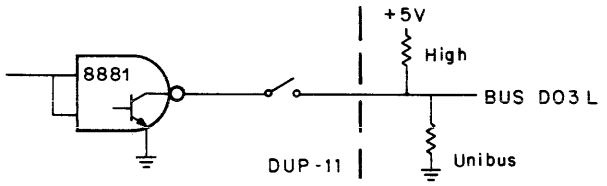
Vector address bits 08–02 are placed on the Unibus data lines via type 8881 open-collector bus drivers. Bits 08–03 are connected to the Unibus via switches in the driver output lines. With the switch open, a 0 is placed on the Unibus; with the switch closed, a 1 is placed on the Unibus. In this way, the first two octal digits of the vector address can be selected.

Figure 4-26 shows the selection of a 1 and 0 on the Unibus using vector address bit 03 as a typical example.



Enabling Vector Bit 03 With Switch ON

High enabling signal drives output of 8881 low (last stage) transistor is ON). +5 V goes to ground which holds BUS D03L low. This is a logical 1 on the Unibus.



Enabling Vector Bit 03 With Switch OFF

With the switch OFF, the 8881 driver is not connected to the Unibus. The +5 V applied to the terminator resistive divider holds BUS D03L high. This is a logical 0 on the Unibus.

11-3388

Figure 4-26 Selecting State of Vector Address Bits

To select a 1, the switch is closed (ON) which connects the 8881 driver output to Unibus data line BUS D03 L. When the vector address is to be put on the Unibus, the interrupt control logic enables the drivers. This is done by putting a high signal on the dual inputs of each driver. The driver output goes low. Being an open-collector device, the last stage transistor in the device turns on. Its emitter is connected to ground in the device so it pulls the voltage on the Unibus data line to the level of the device saturation voltage (approximately 0.8 V max. at the collector). This low on Unibus line BUS D03 L represents a logical 1.

To select a 0, the switch is opened (OFF) which disconnects the 8881 driver output from Unibus data line BUS D03 L. The +5 V applied to the terminator resistive divider holds the line high which represents a logical 0.

Figure 4-27 shows the determination of vector addresses for two DUP11s in a system. It is desired to have the first DUP11 generate 300 for receiver interrupts and 304 for transmitter interrupts. The second DUP11 follows with receiver and transmitter interrupt vectors 310 and 314, respectively. Using the first DUP11 as the example, the switch selections are as follows: switch OFF for bits 8, 5, 4, and 3 and switch ON for bits 7 and 6. This selects 30 as the first two digits of the vector address. The state of bit 2 determines whether the last digit is 0 or 4. Assume that the receiver is requesting the interrupt which generates vector address 300. Flip-flop V2 is set and its 0 output puts a low on one input of the bit 02 driver. The high enabling signal is sent to the other input when the interrupt control logic sets the BBSY flip-flop. The driver output goes high which represents a logical 0 on the Unibus; hence, vector 300 is placed on the Unibus data lines. Details of the logic that controls the state of bit 2 are discussed in a subsequent paragraph. With bits 1 and 0 always 0, addresses 301, 302, and 303 cannot be generated. The only other address that can be generated with this configuration is 304.

The second DUP11 has a different switch configuration for bits 8–3 which allows it to generate vectors 310 and 314.

Note that the binary equivalent of the vector address is represented by logic states on the Unibus data lines. Negative logic conventions are used for these lines; that is, a low is 0 V and it represents a logical 1 while a high is +5 V and it represents a logical 0.

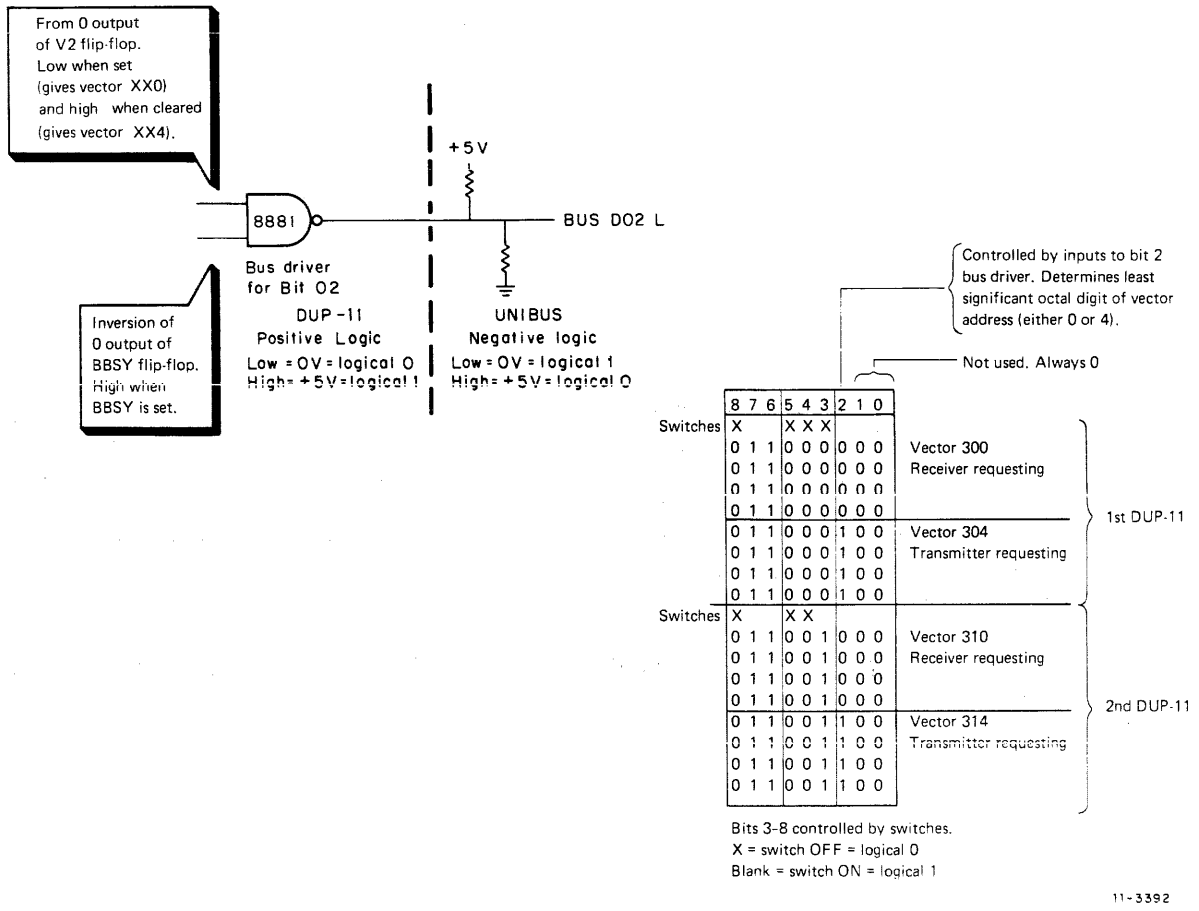


Figure 4-27 Selecting Vector Addresses

4.3.9.3 BR Priority Selector Card – In the PDP-11 system, an interrupting device requests bus mastership via one of four bus request (BR) lines. They are referred to as BR4–BR7 with BR4 being the lowest priority. For devices connected to the same BR line, the device electrically closest to the processor has the highest priority.

A particular device is shipped hard-wired for a recommended BR level; however, the device can be made to interrupt on any BR4 level. Physically, this can be accomplished by changing a small plug-in type printed circuit card. This example uses level BR4; however, the DUP11 is shipped with a level BR5 card installed.

A 16-pin socket is permanently attached to the module and the priority card plugs into it. A separate card is used for each interrupt level (BR4–BR7). Etched jumpers on the card configure it for a specific BR level. Figure 4-28 shows the configuration of a priority card for level BR4. There are no connections made to bus request levels BR5–BR7. Jumpers are used to allow bus grants (BGs) from the processor on levels 5–7 to pass through the device to succeeding devices using these levels. The bus request signal (BR REQ L) from the interrupt control logic goes to pin 12 of the card and is jumpered to pin 13 (BUS BR4 L) which is Unibus signal line BR4. The path for the processor bus grant signal for level 4 under two conditions is described.

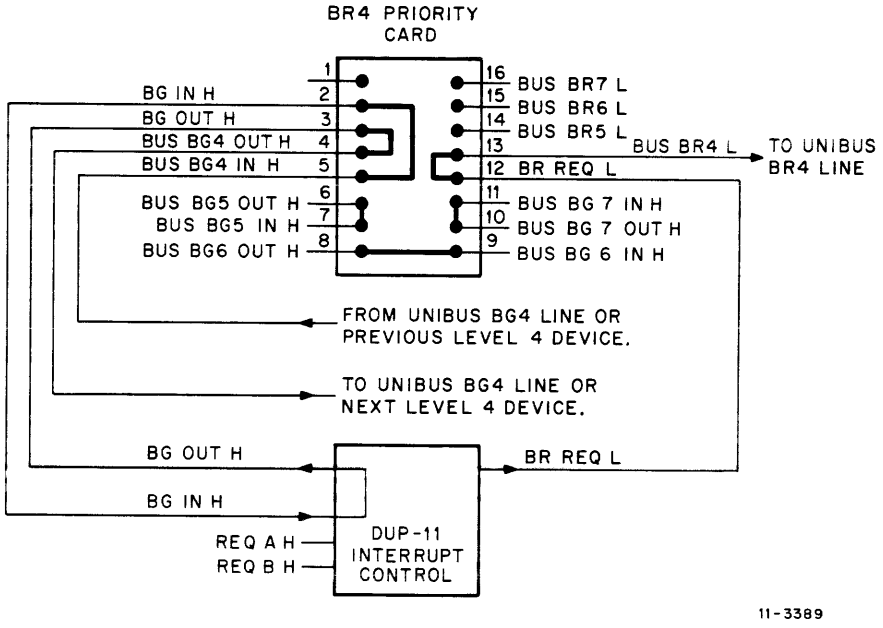


Figure 4-28 Configuration of the BR4 Priority Card

Condition 1 assumes that the processor has issued a bus grant (BG) signal in response to a request from another level 4 device further down the bus. The BG4 signal enters pin 5 of the priority card (BUS BG4 IN H). It is jumpered to pin 2 and leaves as BG IN H. It enters the DUP11 interrupt control logic, and because this device is not requesting, it leaves as BG OUT H. From here, it re-enters the priority card on pin 3, is jumpered to pin 4, and leaves as BUS BG4 OUT H. It continues on to the next level 4 device. Eventually, the grant signal is stopped at the requesting device.

Condition 2 assumes that the processor has issued a bus grant signal in response to a request (BR4) from the DUP11. The path of the BG4 signal is the same up to the point where it enters the DUP11 interrupt control logic as BG IN H. Because the DUP11 is requesting, the grant signal (BG4) is stopped here.

In summary, each unique priority card (BR4–BR7) allows only the assigned level request signal to be connected to the associated Unibus BR line. Grant signals for other than the assigned level are passed through the priority card. The grant signal for the assigned level is routed through the priority card and interrupt control logic of the device, if it is not requesting. If the device is requesting, the assigned level bus grant signal is stopped by the interrupt control logic.

4.3.9.4 Typical Interrupt Transaction

Introduction
The DUP11 communicates with the Unibus via program transfers (DATI, DATO, and DATOB transactions) and interrupt requests. The DUP11 does not use NPR transactions.

The reader should be aware that the Unibus uses negative logic for all signals except grant signals (BG4 - BG7 and NPG). The definitions of positive and negative logic are shown below.

Negative Logic

Signal Asserted: Low = Logical 1 = 0 V
Signal at Rest: High = Logical 0 = +3 V

Positive Logic

Signal Asserted: High = Logical 1 = +3 V
Signal at Rest: Low = Logical 0 = 0 V

The interrupt control logic responds to interrupt requests from the transmitter and receiver. Signal REQ A H is the receiver request signal and it is associated with vector address XX0. Signal REQ B H is the transmitter request signal and it is associated with vector address XX4. Both requests are at level BR4; however, if they occur simultaneously, the receiver request (REQ A) is honored first.

REQ A and REQ B (Logic Sheet BSI7)
The request for a transmitter interrupt (REQ B H) is generated at the output of a 7408 AND gate when TXDNE (1) H and TXITEN (1) H are both asserted. Signal TXITEN (1) H is asserted by the program when a transmitter interrupt is desired. This signal is the transmitter interrupt enable bit (TXCSR bit 06).

Signal TXDNE (1) H, which is the transmitter done bit (TXCSR bit 07), is set by the hardware when the transmitter data buffer is available for a new character. (See Chapter 3 for details concerning the setting of TX DONE under other specific conditions.)

The request for a receiver interrupt (REQ A H) is generated at the output of a 7408 AND gate and delayed 100 ns before being sent to the REQ A flip-flop in the interrupt control logic. The other input of this AND gate is $\overline{\text{RD RXCSR H}}$ which is the equivalent of RD RXCSR L. Signal RD RXCSR L is generated by the address selection logic when the RXCSR is being read. Therefore, when the RXCSR is not being read, RD RXCSR L or its equivalent $\overline{\text{RD RXCSR H}}$ is high which qualifies the 7408 gate to assert REQ A H when its other input is high. This input comes from the output of a 7432 OR gate. Its inputs are the AND function of RXDNE (1) H and RXITEN (1) H and the AND function of A DAT SET CH (1) H and DSC ITEN (1) H. Assertion of either pair of these signals asserts REQ A H, provided $\overline{\text{RD RXCSR H}}$ is high.

Signal RXITEN (1) H, which is the receiver interrupt enable bit (RXCSR bit 06), is asserted by the program if it is desired to request a receiver interrupt when the RXDONE bit is set. Signal RXDNE (1) H, which is the receiver done bit (RXCSR bit 07), is set by the hardware when RXACT is set and a character is transferred into the receiver data buffer. (See Chapter 3 for details concerning the setting of RXDONE under other specific conditions.)

Signal DSCITEN (1) H, which is the data set interrupt enable bit (RXCSR bit 05), is asserted by the program if it is desired to request a receiver interrupt when the ADAT SET CH (1) H bit is set. Signal ADAT SET CH (1) H, which is the data set change A bit (RXCSR bit 15), is set by any transition on the CLEAR TO SEND line or a positive transition on the RING line greater than 10 ms.

Typical Example of an Interrupt Transaction
This example describes a basic interrupt transaction and deals primarily with the interrupt control logic in the DUP11. It does not focus on the details of the PDP-11 processor interaction. For these details, refer to the *PDP-11 Peripherals Handbook (1975)*, Chapter 5, Unibus Theory and Operation.

The logic for this example is shown in sheet BSI7.

1. Initially, no request is pending (BR REQ L is high) and all flip-flops in the interrupt control logic are cleared. The vector address drivers are inhibited. Signals BG IN H, BG OUT H, BUS SACK L, and BUS BBSY L are not asserted.

2. Assume that REQ A H is asserted. This action drives BR REQ L low. This signal enters the BR4 priority card and leaves as BUS BR4 L which goes to the processor.

3. The processor examines BUS BR4 L and if it has the highest priority and BUS SACK L is clear asserts BUS BG4 IN H. This signal enters the BR4 priority card and leaves as BG IN H which goes to the interrupt control logic.

4. When BG IN H goes high, the GRANT flip-flop is clocked; however, because REQ A H is asserted its D input is low and the flip-flop does not change state (assumed to start cleared). Signal BG OUT H is low which blocks the bus grant signal and prevents it from reaching any following devices at level BR4 on the Unibus.

5. The assertion of BG IN H puts a low on the D input of the SACK flip-flop. Signal BG IN H is delayed 50 ns after assertion and clocks the SACK flip-flop which sets it because it is redefined. This action asserts BUS SACK L and clears BR REQ L.

6. The processor receives BUS SACK L and clears BG IN H which prevents the issuance of further grants from the processor during this interrupt transaction.

7. Clearing BG IN H sets up the interrupt control logic for the subsequent clocking of the V2 flip-flop.

8. When the current bus master completes its transaction, it clears BUS SSYN L (probably already cleared) and BUS BBSY. The following actions occur:

   a. The BBSY flip-flop is set, thus the DUP11 asserts BUS BBSY L.

   b. The SACK flip-flop is cleared directly.

   c. The V2 flip-flop is clocked and it is set. This puts a low on one input of the bit 2 driver which holds its output (BUS DB02 L) high. Signal BUS DB02 L is a Unibus signal that uses negative logic conventions. A high corresponds to a logical 0 which produces the address for a receiver interrupt (XX0).

9. When the BBSY flip-flop is set, its 0 output enables the drivers for bits 3–8 and asserts BUS INTR L. The DUP11 is now bus master and the receiver vector address (XX0) is placed on the Unibus data lines (BUS DB02 L–BUS DB08 L). The first two digits are selected by switches in the lines for bits 3–8.

10. The processor receives the assertion of BUS INTR L, reads the vector address, and responds by asserting BUS SSYN L. The assertion of BUS SSYN L clocks the REQ A and REQ B flip-flops. REQ A (which was requesting) is set, which holds it in the set position, thus inhibiting further requests from the receiver (REQ A). The program must negate the receiver request (drive REQ A H low) in order to clear the REQ A flip-flop and qualify the logic to respond to additional requests from the receiver. (This hold-set feature is described in a subsequent paragraph.)

11. In response to the assertion of BUS SSYN L, the logic directly clears the BBSY flip-flop. This clears BUS BBSY L, BUS INTR L, and the vector address. This constitutes active release of the bus to the processor which clears BUS SSYN L when BUS INTR L is cleared. The processor goes to the interrupt service routine at the specified vector address (XX0).

Interaction Among REQ A, REQ B, and V2 Flip-Flops

Figure 4-29 shows the REQ A, REQ B, and V2 flip-flops along with some associated interrupt control logic.

The REQ A flip-flop responds to a receiver request for interrupt (signal REQ A H asserted). The REQ B flip-flop responds to a transmitter request for interrupt (signal REQ B H asserted).

Initially, the request flip-flop (REQ A or REQ B) must be cleared. In this state, the high from its 0 output goes to one input of a 7400 NAND gate. The other NAND gate input comes from the associated request signal (REQ A H or REQ B H). When the request signal is asserted, the NAND gate output goes low which is required to generate BR REQ L.

The state of the REQ A flip-flop and signal REQ A H determine whether the V2 flip-flop is set or cleared during the interrupt transaction. The V2 flip-flop is redefined which means that it is set when clocked with its D input low. This is the case when the receiver is requesting the interrupt; that is, REQ A H is asserted and the REQ A flip-flop is cleared. At the point in the transaction when the current bus master clears BUS BBSY L, the V2 flip-flop is set. The low from its 0 output goes to one input of the bit 2 driver. The other input of this driver is high because the BBSY flip-flop is set. The driver output (BUS DB02 L) goes high. This is a Unibus signal in which a high is equivalent to logical 0. This establishes the last digit of the vector address as a 0 which identifies it as a receiver vector (XX0).

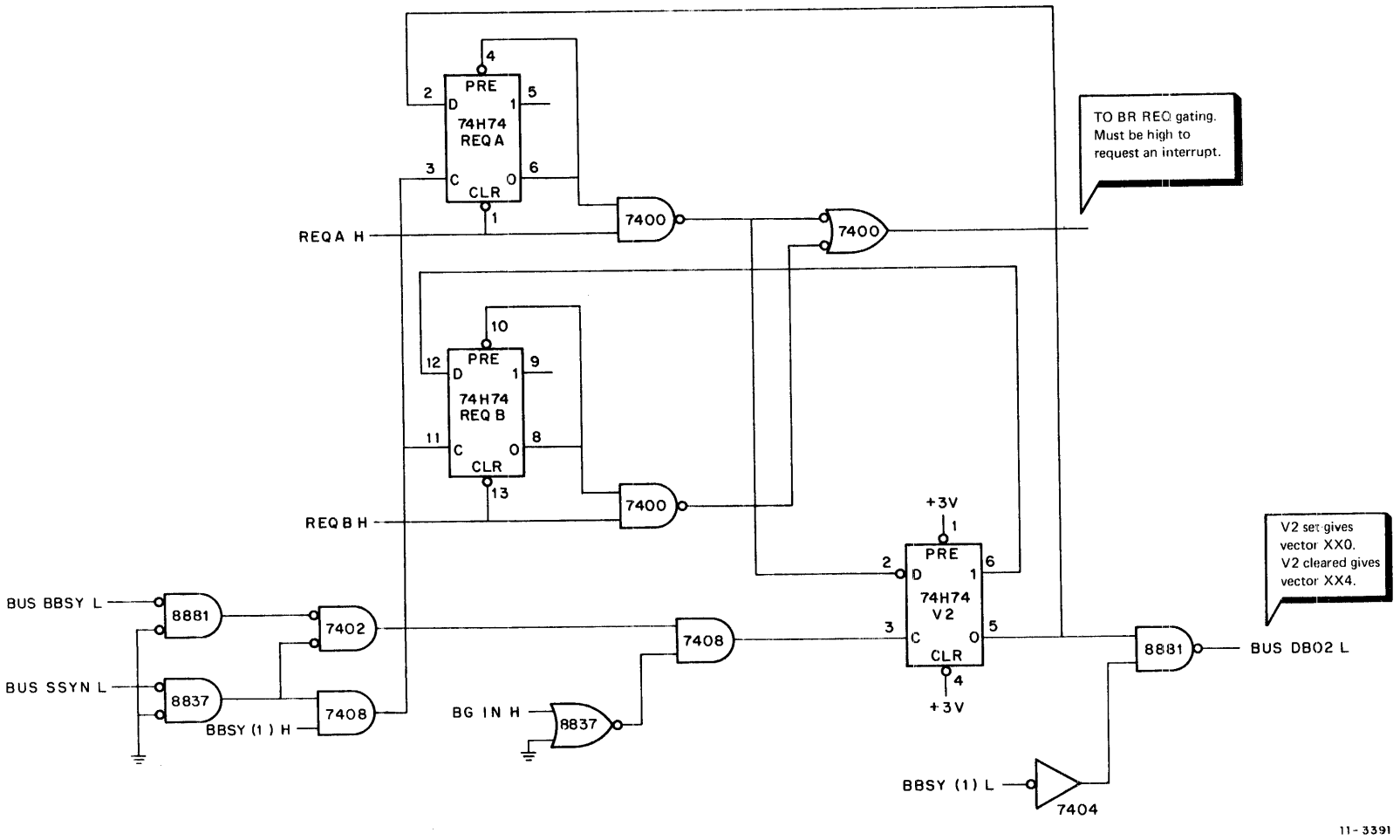


Figure 4-29 REQ A, REQ B and V2 Flip-Flops and Associated Interrupt Control Logic

The 1 output of the V2 flip-flop is sent to the D input of the REQ A flip-flop and its 0 output is sent to the D input of the REQ B flip-flop. In the situation described above (REQ A H asserted and V2 set), the D input of the REQ B flip-flop is low and the D input of the REQ A flip-flop is high. Later in the transaction, when both of these flip-flops are clocked simultaneously, the REQ B flip-flops remain cleared but the REQ A flip-flop is set. The low from the 0 output of the REQ A flip-flop inhibits the ability of the interrupt control logic to respond to another request for an interrupt by the receiver; that is, signal BR REQ L cannot be driven low by asserting REQA H because the REQ A flip-flop is set. (This action is also performed by the REQ B flip-flop if the transmitter had requested the interrupt.) The 0 output of the REQ A flip-flop is fed back to its preset input. With the flip-flop set, the preset input is low and the flip-flop is held set as long as REQ A H remains high.

When the BBSY flip-flop is cleared and the bus is released to the processor, it is important to unlock the hold-set condition on the REQ A flip-flop so additional interrupt requests can be honored. The program forces signal REQ A H low to unlock the hold-set condition by reading the RXCSR register. Signal REQ A H is connected to the clear input of the REQ A flip-flop. When it goes low, the REQ A flip-flop tries to go to the clear state. Because its preset input is also low, only its 0 output changes (goes high). This releases the preset action and the flip-flop goes to the clear state and is held there as long a REQ A H remains low.

4.3.9.5 NPR Latency Improvement Circuit

Introduction
NPR latency is the delay measured from the time an NPR device issues a bus request to the time that it starts to use the Unibus for a data transfer. In a system, it is a function of current bus activity and the type and arrangement of devices on the Unibus.

The BR portion of the M7821 Interrupt Control Logic or its equivalent contains a special circuit to improve the NPR latency in a system containing BR and NPR devices.

Functional Description
Assume that a BR device with the NPR latency improvement circuit is not requesting an interrupt but another BR device at the same priority level farther down the bus has asserted a bus request. Assume further that an NPR device asserts a bus request before the processor issues a bus grant (BG) to the requesting BR device. When the non-requesting BR device receives the NPR, followed by the BG for the same level device farther down the bus, it blocks the BG and asserts SACK. When the processor receives SACK, it clears the BG and prevents the issuance of further grants. When the current bus master completes its data transaction (clears BBSY and SSYN), the non-requesting BR device clears SACK. This allows the processor to assert NPG in response to the NPR. The NPR device now uses the data section of the Unibus before the BR device that had asserted its request first.

Detailed Logic Description
This discussion covers the operation of the NPR latency improvement circuit in the DUP11 interrupt control logic, under the conditions described in the Functional Description.

This circuit consists of an 8837 NOR gate (shown as the logically equivalent negated input AND gate) and a jumper. One input of the 8837 is connected to +5 V and to one side of the jumper. The other side of the jumper is grounded. The DUP11 is shipped with the jumper installed which holds this input of the 8837 low. In this configuration, the NPR latency improvement signal is active. The other input of the 8837 is BUS NPR L. The output of the 8837 goes to a 7402 NOR gate which in turn goes to the D input of the GRANT flip-flop.

Initial Conditions – The DUP11 is not asserting a bus request; however, another BR device at the same priority level farther down the bus has asserted a bus request. At this point, a bus grant (BG) has not been issued, nor has BUS NPR L been asserted.

1. The following conditions exist in the DUP11 interrupt control logic (sheet BSI7).

    a. The DUP11 is not requesting (BR REQ L is high).

    b. A bus grant has not been issued by the processor (BG IN H is low and BG OUT H is low). The GRANT flip-flop is cleared and its D input is high.

    c. BUS SACK L is high. The SACK flip-flop is cleared and its D input is high.

New Conditions – The DUP11 receives the assertion of BUS NPR L from some other device, followed by the assertion of BG IN H granted to the BR device at the same priority level farther down the bus.

2. As a result of receiving the assertion of BUS NPR L, the D input of the GRANT flip-flop goes low.

3. As a result of receiving the assertion of BG IN H, the following events occur:

    a. The D input of the redefined SACK flip-flop goes low.

    b. The GRANT flip-flop is clocked but it remains cleared because its D input is low. BG OUT H remains low which means that the grant is blocked.

    c. After a delay, the assertion of BG IN H clocks the SACK flip-flop. With its D input low, this redefined flip-flop is set which asserts BUS SACK L.

4. When the processor receives BUS SACK L, it clears BG IN H and prevents the issuance of further grants.

5. When the current bus master completes its transaction, it clears BUS BBSY L and BUS SSYN L. This action directly clears the SACK flip-flop which clears BUS SACK L. This action also clocks the BBSY flip-flop but it remains cleared. Its direct clear input is held low because the DUP11 is not requesting an interrupt.

6. Because the DUP11 did not assert BUS BBSY L, the processor asserts BUS NPG H in response to BUS NPR L as soon as it receives the cleared BUS SACK L signal. The NPR device becomes bus master and uses the data section of the Unibus before the BR device that had asserted its request first.

7. If the BR from the device farther down the bus is still pending, it will be arbitrated again when the NPR device clears BUS SACK L.

4.3.10 Data Set Interface Logic

4.3.10.1 General – The data set interface logic is shown in logic sheet BSI5 and is divided into three functional groups as described below.

1. Control and status signals from the data set are converted to TTL logic levels. Some signals are sent to the RX and TX control logic and others are used as indicators, and in some cases, as requests for an RX interrupt. This logic includes seven RXCSR bits (numbers 0, 9, 10, and 12–15).

2. Three control/status signals are sent to the data set. They are bits 1, 2, and 3 of the RXCSR.

3. Logic is provided to control the flow of transmitted data during the user mode and to control the external clock signal during the external maintenance mode.

4.3.10.2 Logic for Signals from Data Set – Eight signals are sent from the data set to the interface logic where they are converted to TTL logic levels and then used to perform specific tasks. Each signal is converted from EIA logic levels to TTL logic levels by a 1489 receiver which inverts the signal in the process. The voltage levels at the input and output of the 1489 receiver are shown below:

1489 Input (EIA Levels)	**1489 Output (TTL Levels)**
6 V to 1.5 V	0.8 V to 0 V
0 V to -6 V	2.4 V to 5 V

Each 1489 has an external response control input that is connected to one side of a capacitor. The other side of the capacitor is connected to ground. The ungrounded side of the capacitor is also connected to one of a pair of terminals. The other terminal is connected to -15 V. If desired, a resistor can be connected across the terminals to change the threshold voltage for the level conversion. The DUP11 is shipped without resistors installed.

RECEIVER SIGNAL TIMING ELEMENT (receive data set clock) is converted and becomes RX DATA SET CLK H which is sent to the RX clock logic.

TRANSMITTER SIGNAL TIMING ELEMENT (transmit data set clock) is converted and becomes TX DATA SET CLK L which is inverted to produce TX DAT SET CLK H. This signal goes to the TX clock logic.

RECEIVED DATA (serial data in) is converted and then inverted to become RX DATA H which is sent to the RX 1s counter.

Four other signals are converted and inverted to become read-only bits of the RXCSR register:

RXCSR Bit No.	**DUP11 Signal Name**	**EIA Signal Name**
9	DAT SET RDY	DATA SET READY
10	SEC RCV	SECONDARY RECEIVE DATA
12	CARRIER	RECEIVED LINE SIGNAL DETECTOR
13	CLR TO SD	CLEAR TO SEND

Each one of these RXCSR signals goes to a pulse generator that produces a 2.6 μs pulse whenever a transition (positive or negative) is detected on the line (Figure 4-30). The pulses from the SEC RCV, DAT SET RDY, and CARRIER lines directly set RXCSR bit 0 (BDAT SET CH). Bit 0 is a read-only bit that serves as a flag to indicate that a transition has occurred on one of the three aforementioned lines. Physically, RXCSR bit 0 is a 7474 flip-flop. It is set directly via its preset input by the 2.6 μs negative pulse from any one of the three lines. Its D input is connected to ground permanently. When the RXCSR is read, RD RXCSR L is generated by the address selection logic. When the register has been read, RD RXCSR L goes high again and clocks the flip-flop which clears it.

The CLR TO SD line directly sets RXCSR bit 15 (ADAT SET CH). Bit 15 is a read-only bit that serves as a flag to indicate that a transition has occurred on the CLR TO SD or RING lines. (The RING circuit is discussed later in this section.) If the data set interrupt enable bit (RXCSR bit 5) has been set by the program, a receiver interrupt request is generated when ADAT SET CH is set. Bit 15 is cleared when the RXCS register is read.

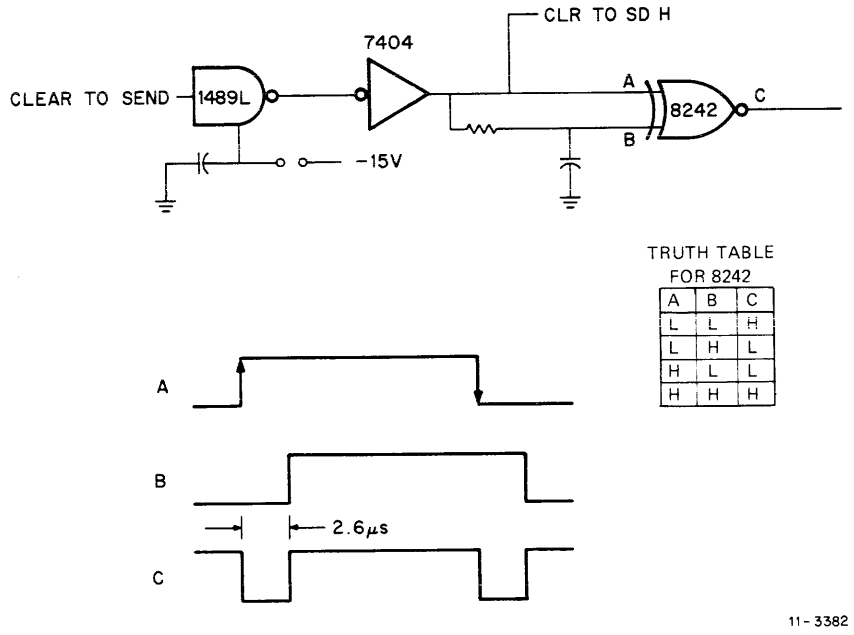


Figure 4-30 Pulse Generator for SEC RCV, DSR, CARRIER and CLR TO SD Lines

In two areas of this logic, jumpers are provided to alter functional operation.

The first area concerns the preset inputs to the ADAT SET CH and BDAT SET CH flip-flop. The standard jumper configuration is shown in the print set. It allows the RING and CLR TO SD lines to set ADAT SET CH and it allows CARRIER, DAT SET RDY, and SEC RCV lines to set BDAT SET CH. (This configuration is described above.) Removing the standard jumper and installing one in the other set of plated-through holes (PTHs) allows all of the above mentioned lines to set ADAT SET CH. The BDAT SET CH flip-flop cannot be set in this configuration. This makes the DUP11 compatible with the DU11. A third configuration would be to remove the B DAT SET CH jumper and only monitor these inputs.

The second area concerns the entry of SECONDARY RECEIVE into the DUP11. The standard jumper configuration is shown in the print set. It allows signal SECONDARY RECEIVE to be received by the DUP11. This signal is not used on Bell 201, 208, or 209 modems. To prevent entry of this signal, the standard jumper is removed and a jumper is installed in the other set of PTHs. Signal SECONDARY RECEIVE now goes to ground instead of entering the DUP11.

The last signal (RING INDICATOR) is converted and becomes bit 14 of the RXCSR. A transition on this line sets ADAT SET CH. The input to the pulse generator for the RING line contains a one-shot to eliminate false setting of the ADAT SET CH flip-flop during the positive transition of the RING signal. The one-shot provides a 10 ms delay after the RING line goes high to ensure that a legitimate level change rather than electrical noise has triggered the one-shot.

The one-shot has three inputs: pin 4 is unused, pin 5 is connected to +3 V, and pin 3 is connected to the output of the 1489 receiver. Only a negative transition at pin 3 triggers the one-shot. This corresponds to the RING line going high. The pulse generator and associated timing diagram are shown in Figure 4-31. A positive or negative transition of the RING line presets the ADAT SET CH flip-flop. The 10 ms delay occurs only on the positive RING transition.

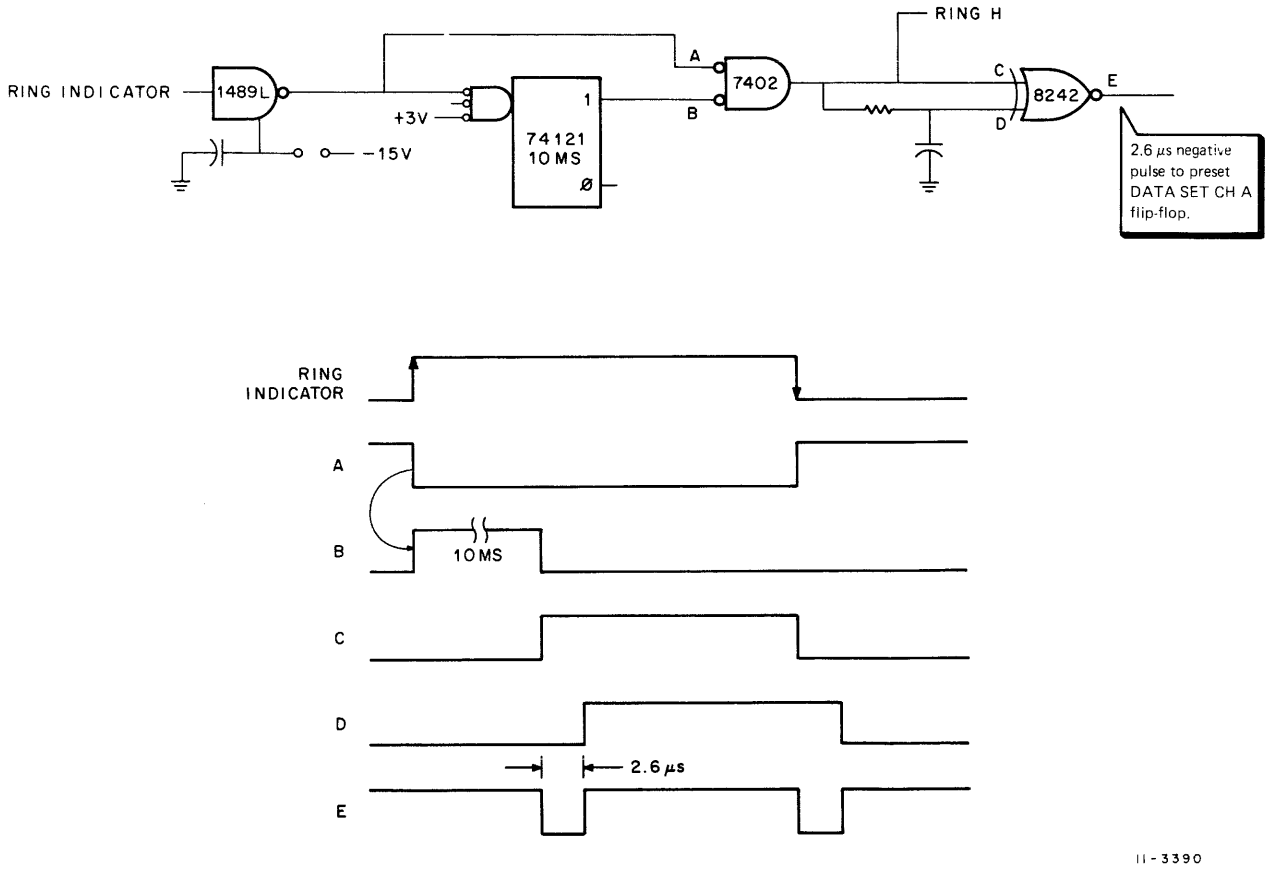


Figure 4-31 Pulse Generator for RING Line

4.3.10.3 Logic for Signals to Data Set - Three RXCSR program read/write bits are used to send information to the data set:

RXCSR Bit No.	DUP11 Signal Name	EIA Signal Name
1	DTR	DATA TERMINAL READY
2	RTS	REQUEST TO SEND
3	SEC TX	SECONDARY TRANSMIT DATA

These three bits are stored in a 74175 quad flip-flop. The common clock signal is LD RXCSR LB H which is generated by the register selection logic when the RXCSR is written into (word or low byte). The common direct clear signal is CLR L. The DUP11 is shipped with a jumper in the CLR input line. It can be removed so that these bits cannot be cleared by DEVICE RESET or BUS INIT. In this case, they must be cleared by the program.

The output of each bit goes to a 1488 driver which inverts the signal and converts it from TTL logic levels to EIA logic levels. The voltage levels at the input and output of the 1488 driver are shown below:

1488 Input (TTL Levels)	1488 Output (EIA Levels)
0.4 V to 0 V	3 V to 6 V
2.4 V to 5 V	-3 V to -6 V

The DUP11 is shipped with a jumper in the SECONDARY TRANSMIT line. Removal of the jumper inhibits this signal. (SEC TX is not used on Bell 201, 208, and 209 modems.)

4.3.10.4 Logic for Transmitted Data and External Maintenance Clock – The logic associated with transmitted data is discussed in detail in Paragraph 4.3.6.3.

A 10 kHz clock is supplied for use during the external maintenance mode. The modem is disconnected and replaced by the H325 test connector which allows checking of the interface including level converters and cables. The 10 kHz output [MAI EXT CLK (1) H] of the RC clock is sent to one input of a 1488 driver. [The signal is shown as its logical inversion MAI EXT CLK (0) L.] The other input of the driver is MAI SELB (0) L which is high when the external maintenance mode is selected. In this case, the clock signal [MAI EXT CLK (0) L] is gated through the 1488 which converts this TTL signal to EIA logic levels. The 1488 output is identified as TRANSMITTER SIGNAL ELEMENT TIMING EXTERNAL which simply means EIA XMIT DATA.

4.3.11 Typical Operations

4.3.11.1 Introduction – In this section, two typical sequenced operations are discussed to familiarize the reader with the transmit and receive ROM control functions. Paragraph 4.3.11.2 discusses a typical SDLC transmit operation and Paragraph 4.3.11.3 discusses a typical SDLC receive operation.

4.3.11.2 Typical Transmit Operation (SDLC) – The following example is a detailed discussion of a typical DUP11 transmit operation using SDLC protocol. The total transmission consists of two messages or frames. The first frame contains the following sequenced characters: flag, address, command, two information or data characters and a 16-bit CRC. In this example, the next character is an intermessage flag that separates the two frames. This flag is followed in sequence by the following characters: address, command, information, CRC, and the terminating flag.

References

Transmitter Timing Diagram (Figure 4-32)
Transmitter Logic [drawing D-CS-M7867-0-1, sheets 2 (BSI2) and 5 (BSI5)]
Listings for the transmitter ROMs (Function Decode, Data Path Control, and Data Decode)
Detailed logic descriptions (Chapter 4) for detailed explanation of specific areas as required

Details of a Typical SDLC Transmit Operation

1. The program initializes the DUP11 by setting DEVICE RESET (TXCSR bit 8) which generates signal CLR H (1 μs pulse) in the device reset logic. The following bits are among those cleared.

   TXACTIVE (TXCSR bit 9)
   SEND (TXCSR bit 4)
   TSOM (TXDBUF bit 8)
   TXITEN (TXCSR bit 6)

   Signal CLR H directly sets TXDONE (TXCSR bit 7).

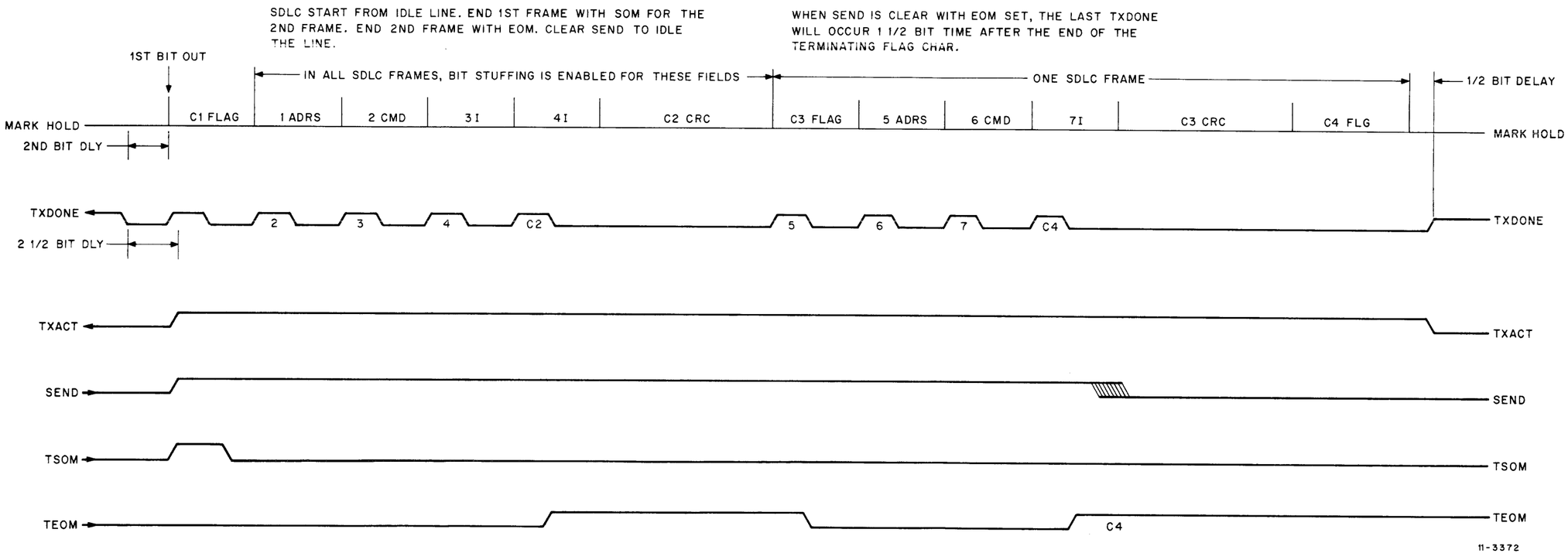


Figure 4-32 Typical SDLC Transmit Operation

2. Clearing TXACTIVE drives TXACT (0) H high. This signal generates LOAD H which is a qualifying input to the 7408 gate that provides the clock input to the Bit Sync Buffer. The other input to this gate is $\overline{\text{TXCLKD H}}$. With LOAD H asserted, the Bit Sync Buffer is clocked on every positive transition of $\overline{\text{TXCLKD H}}$. Signal SKPSEQ (0) L is ANDed with LOAD H to generate SKPLD H.

   Because of DUP11 initialization, signals TXACT (1) L and SEND (1) L are both high. They are ORed in the data set logic (sheet 5) to drive EN DATA H low. This signal is double-inverted to drive EIA XMIT DATA low which is the MARK state.

   Signal EN DATA H (shown as $\overline{\text{EN DATA L}}$) goes to the clear input of the TXDT flip-flop. When it goes low, the TXDT flip-flop is cleared. As long as it is held low, the flip-flop does not respond to the clock signal (TXCLKP L). The D input of the TXDT flip-flop is signal TXDT H from the Data Decode ROM. As long as the flip-flop is held cleared, TXDT H has no effect on the transmitter control logic.

3. The program sets the following bits: SEND, TSOM, and TXITEN. When the program addresses the TXDBUF register to set TSOM, the address decoding logic asserts LD TXDBUF L. When the address is cleared, LD TXDBUF L goes high again and this positive transition clocks the TXDONE flip-flop, which clears it because its D input is permanently connected to ground. This action is asynchronous to the data set clock.

4. When TSOM is set, TSOM (1) H is asserted and goes to the Bit Sync Buffer, which is clocked by the next $\overline{\text{TXCLKD H}}$ pulse and asserts SSOM (1) H. The start of message function is now synchronized to the transmit data set clock. Signal TSOM (1) H is also ANDed with SSOM (1) H to generate SEND FLAG H which is a Function Decode ROM input.

   Summary
   The program has set SEND and TSOM and the hardware has cleared TXDONE. TXACTIVE is cleared. The data out line EIA XMIT DATA is held in the idle (MARK) state. The start-of-message function is synchronized to the transmit data set clock.

5. The Function Decode ROM is in the following state.

Inputs	Adrs	Asserted Outputs
SEND (1) H SEND FLG H LOAD H	37	TFLG H and ENTXAC H. First Clock pulse of the start-up sequence.

6. Signals TFLG H and ENTXAC H go to inputs of the Bit Sync Buffer and when it is clocked by the next $\overline{\text{TXCLK D H}}$ pulse, SFLG (1) H and SENTXAC (1) H are asserted.

7. The Data Path Control ROM and Data Decode ROM are in the following states.

Data Path Control ROM

Inputs	Adrs	Asserted Outputs
SFLG (1) H LOAD H	33	ENTCSC H and TSPACE H. Send one SPACE. This is the first bit of the flag character.

Data Decode ROM

Inputs	Adrs	Asserted Outputs
SSOM (1) H LOAD H TSPACE H	24	No outputs are asserted (set line to 0). TXDT H is low which drives the external line (EIA XMIT DATA) high. This is a SPACE (logical 0).

8. When the Bit Sync Buffer was clocked (step 6), signals SFLAG (1) H and SENTXAC (1) H were fed back as inputs to the Function Decode ROM which puts it in the following state.

Inputs	Adrs	Asserted Outputs
SEND (1) H SENTXAC (1) H SFLG (1) H SEND FLG H SKP LD H	61	TFLG H, ENTXAC H and ENTXDNE H. Send MARK in flag sequence and set TXDONE.

9. Signal ENTXDNE H from the FDROM goes to one input of the 7408 AND gate in the preset input gating for the TXDONE flip-flop. The other input is clock signal TXCLKP H. This 300 ns positive pulse is generated by a one-shot that is triggered by the positive-going edge of $\overline{\text{TXCLKD H}}$ (step 6). Therefore, shortly after the $\overline{\text{TXCLKD H}}$ pulse that set up the transmitter to send the first bit (SPACE) of the flag character, TXCLKP H and ENTXDNE H are ANDed and then inverted to directly set the TXDONE flip-flop.

Summary
The first bit (SPACE) of the flag character has been transmitted. The hardware has set TXDONE. The transmitter logic has been set up to transmit the second bit (MARK) of the flag character. The logic has been qualified to set the TXACTIVE flip-flop.

10. Signal SENTXAC (1), which was asserted in step 8, is sent to the D input of the TXACTIVE flip-flop. The next positive transition of $\overline{\text{TXCLK H}}$ clocks this flip-flop and sets it. Signal TXACT (0) H goes low which drives LOAD H and SKP LD H both low. Now the Bit Sync Buffer can only be clocked at the end of a character when the TCS counter overflows and generates TCSC MAX H which is a positive pulse.

11. At this point, the ROMs are in the following states.

Function Decode ROM

Inputs	Adrs	Asserted Outputs
SEND (1) H SENTXAC (1) H SFLG (1) H SEND FLAG H	60	TFLG H and ENTXAC H. Finish flag character.

Data Path Control ROM
The only asserted input is SFLG (1) H which gives address 32. Only output ENTCSC H is asserted. This signal enables the TCSC counter and the flag continues to be transmitted.

Data Decode ROM
The only asserted input is SSOM (1) H which gives address 128. This asserts TXDT H, which means send a MARK. Neither the TX shift register nor the TXCRC register is enabled to this ROM, but their outputs (TXSER OUT H and TCRC OUT H) are connected to its inputs and their clocks are disabled. As a result, the ROM is programmed to assert TXDT H for all combinations of TXSER OUT H and TCRC OUT H with SSOM (1) H asserted (reference DDROM addresses 128, 129, 132, and 133). This renders the states of TXSER OUT H and TCRC OUT H irrelevant at this time.

12. With TXDONE set, the program can change the inputs to the Function Decode ROM. The program clears TSOM (TXDBUF bit 8). When the TXDBUF register is addressed, signal LD TXDBUF L is asserted. When the address is cleared, LD TXDBUF L goes high again and clocks the TXDONE flip-flop which clears it.

13. Signals SSOM (1) H and SFLG (1) H are latched outputs of the Bit Sync Buffer and cannot be changed until this buffer is clocked again at the end of the flag character. Therefore, the Data Path ROM and Data Decode ROM remain in the states described in step 11 until signals TC=5 H and T1BC=6 H are asserted. The transmitter continues to send MARKs.

14. The Transmit One Bit Counter (T1BC) counts 1s that are being transmitted. The transmitter logic controls the sending of the flag character (01111110). When the fifth 1 (MARK) is detected, the T1BC counter asserts signal T1BC=5 H. This signal is sent to an input of the Data Path Control ROM. Now DPCROM inputs SFLG (1) H and T1BC=5 H are asserted to give address 34. ENTCSC H is asserted. No other outputs are asserted which indicates that a MARK is to be sent. This ROM address identifies the sixth bit of the flag character. Now only Data Decode ROM input SSOM (1) H is asserted, which gives address 128, 129, 132, or 133, depending on the states of DDROM inputs TCRC OUT H and TXSER OUT H. For each of these addresses, only output TXDT H is asserted which makes EIA XMIT DATA a MARK.

15. When the T1BC counter detects the sixth MARK (seventh bit), it asserts T1BC=6 H. The Data Path Control ROM and Data Decode ROM states are as follows.

Data Path Control ROM

Inputs	Adrs	Asserted Outputs
SFLG (1) H T1BC=5 H T1BC=6 H	38	TSPACE H. Last bit of the flag character starts here.

Data Decode ROM
The only asserted inputs are TSPACE H and SSOM (1). The states of inputs TCRC OUT H and TXSER OUT H are irrelevant because the ROM has programmed all combinations of these two signals with TSPACE H asserted (addresses 160, 161, 164, or 165) to yield no DDROM outputs. Output TXDT H is low which makes EIA XMIT DATA a SPACE (0).

16. During the last bit of the flag character, signal TXDT H is low. This drives TXDAT (0) H high and it is ANDed with TXCLK H to clear the T1BC counter. Signals T1BC=5 H and T1BC=6 H are now both low.

17. When the last bit of the flag character is counted on the negative edge of TXCLK H, the TCSC counter overflows and generates TCSC MAX H which is a positive pulse. This signal generates LOAD H and SKPLD H to qualify the AND gate that clocks the Bit Sync Buffer. During the positive transition of $\overline{\text{TXCLKD H}}$ that is associated with the last bit of the flag, the Bit Sync Buffer is clocked. At this time, the states of the ROMs are as follows.

Function Decode ROM

Inputs	Adrs	Asserted Outputs
SEND (1) H SENTXAC (1) H SKPLD H	49	ENTXAC H and ENTDNE H. Send data and set TXDONE.

Data Path Control ROM

Inputs	Adrs	Asserted Outputs
LOAD H	1	ENTCSC H and ENTXSR H. Enable TX shift register to DDROM.

Data Decode ROM

Inputs	Adrs	Asserted Outputs
SKPLD H ENTXSR H	66*	CRC CLK EN H and TBUF→SR H. Enable TXCRC logic and load contents of TXDBUF register into TX shift register. Output TXDT H is low which produces a SPACE (0) on the external EIA XMIT DATA line.

*If TXSER OUT H is asserted, ROM address 67 is selected. This asserts CRC CLK EN H, TBUF→SR H, and TXDT H. The results are the same as that shown for address 66, except that TXDT H is high which produces a MARK (1) on the external EIA XMIT DATA line. Signal CRC CLK EN H enables the TXCRC logic so that it is clocked on each bit by TXCLKP H to accumulate a CRC check character for the transmitted data.

When the TX shift register is loaded, ENTXSRC H is not asserted by the Data Decode ROM. This inhibits the shift function while the TX shift register is being loaded.

The TX CRC register is not enabled to the DDROM by ENTCRC at this time; however, even if its output should go high to produce address 70 or 71, the DDROM has programmed these addresses to perform identically to addresses 66 and 67, respectively.

18. Signal ENTXDNE H is asserted during the last bit of the flag character (step 16) and is sent to the AND gate in the preset gating logic for the TXDONE flip-flop. When TXCLKP H goes high as a result of the first data set clock pulse for the second character, the TXDONE flip-flop is set which indicates the availability of the TXDBUF register to the program second interrupt.

19. Halfway through the first bit of the data character, TCSC MAX H times out and drives LOAD H and SKPLD H low. The addresses of all three ROMs change as follows:

    a. The Function Decode ROM goes to address 48 which still calls for a normal data transfer, except that only output ENTXAC H is asserted.

    b. The Data Path Control ROM goes to address 0 which gives the same outputs (ENTCSC H and ENTXSR H) as the previous address.

    c. The Data Decode ROM goes to address 2 which keeps CRC CLK EN H asserted but asserts ENTXSRC H instead of TBUF→SR H. This allows the loaded shift register to be shifted now. Address 2 means that input TXSER OUT H is not asserted; hence, output TXDT H is not asserted and a SPACE is transmitted. If input TXSER OUT H is asserted, address 3 is selected and TXDT H is asserted; hence, a MARK is transmitted.

Summary

The flag character (01111110) has been transmitted. TXDONE has been set again indicating that the TXDBUF register is available. The TX shift register has been loaded with the second character to be transmitted, which in this example is a secondary station address (treated as data). The TX shift register is enabled to the Data Decode ROM and the first bit of the character is being transmitted. TXACTIVE is kept set and the TXCRC logic is enabled.

20. In the SDL protocol a frame begins and ends with a flag character (01111110). The data between the flags must not contain a flag bit pattern. This means that a 0 must be inserted (stuffed) by the transmitter after a sequence of five contiguous 1s within the frame. Therefore, a flag pattern (01111110) cannot be transmitted by chance.

21. In the case of the second character in this example, if five consecutive 1s are transmitted, a 0 must be stuffed in the next bit position, regardless of the state of the bit that normally follows the fifth 1. This occurs without respect to character boundaries.

    When the fifth consecutive 1 is detected, the T1BC counter asserts T1BC=5 H. This selects Data Path Control ROM address 2 which asserts only output TSPACE H. This indicates that a transfer is in progress and a 0 must be stuffed in the next bit position of the serial data stream. TSPACE H goes to the Data Decode ROM. When a bit is stuffed, only the TXDT flip-flop is clocked. The TX shift register, TXCRC register, and TCSC counter are not clocked. All combinations of TCRC OUT H and TXSER OUT H yield no outputs at the DDROM; therefore, TXDT H is low which makes EIA XMIT DATA a SPACE (0). Possible DDROM addresses under these conditions are 20, 21, 24 or 25 (all send a SPACE). Signal TX DAT (0) H is high because the TXDT flip-flop is cleared. When the next TXCLK H pulse occurs, the T1BC counter is cleared and it starts counting 1s again.

22. Transmission continues with the program loading in each character it desires to send whenever the TXDBUF register is available. They include an address character, command character, and two information characters. During the last information character, the program signals the end of the message by setting TEOM. The last information character is sent and the transmitter automatically sends the 16-bit CRC character and the terminating flag character. In this example, another message (frame) is pending so it starts after the flag character.

23. The discussion continues at the point in the second information character where the program sets TEOM (TXDBUF bit 9). This asserts TEOM (1) H and clears the TXDONE flip-flop via signal LD TXDBUF L, as described in step 12. At this point, the ROMs are in the following states.

    Function Decode ROM

Inputs	Adrs	Asserted Outputs
SEND (1) H SENTXAC (1) H TEOM (1) H	50	TCRC H and ENTXAC H. Keep TXACT and TXCRC enabled. Finish this character and then send the CRC character.

    The Data Path Control ROM and the Data Decode ROM operate as described in step 19; that is, they continue to transmit data from the TX shift register which, in this case, is the second information character.

24. During the last bit of the second information character, the TCSC counter overflows and generates the TCSC MAX H pulse which in turn generates LOAD H and SKPLD H. With LOAD H asserted, TXCLKD H clocks the Bit Sync Buffer and asserts SCRC (1) H which is sent to the DPC ROM and is fed back to the FD ROM.

    At this point, the ROMs are in the following states.

    Function Decode ROM

Inputs	Adrs	Asserted Outputs
SEND (1) H SENTXAC (1) H SEND FLAG H TEOM (1) H SKPLD H	55	TFLG H and ENTXAC H. Inter-message flag coming up. Keep TXACTIVE set.

    Data Path Control ROM

Inputs	Adrs	Asserted Outputs
SCRC (1) H LOAD H	17	ENTCSC H and ENTCRC H. Enables TXCRC register to the DDROM. This is the first bit of the CRC character.

    Data Decode ROM
    Inputs SKPLD H and ENTCRC H are asserted. Under these conditions, the TXCRC register is enabled to the DDROM. Its output is TCRC OUT H which is an input to the DDROM also. To avoid interference by the output of the TX shift register (TXSER OUT H) which is a DDROM input, four addresses are provided to represent the four combinations of TCRC OUT H and TXSER OUT H.

    If TCRC OUT H is a 0, address 72 or 73 is selected which does not assert any DDROM outputs. Hence, TXDT H is low which makes EIA XMIT DATA a SPACE (0). If TCRC OUT H is a 1, address 76 or 77 is selected which asserts TXDT H and makes EIA XMIT DATA a MARK (1).

25. The first CRC bit has been transmitted and now LOAD H times out.

    At this point, the ROMs are in the following states.

    Function Decode ROM

Inputs	Adrs	Asserted Outputs
SEND (1) H SENTXAC (1) H SEND FLG H TEOM (1) H	54	TFLG H and ENTXAC H. Transfer CRC character and prime logic for flag character.

    Data Path Control ROM
    Only input SCRC (1) H is asserted which selects address 16. This asserts ENTCSC H and ENTCRC H which keeps the output of the TXCRC register enabled to the Data Decode ROM.

Data Decode ROM
Only input ENTCRC H is asserted which means that the TXCRC register (TCRC OUT H) is enabled to the DDROM. To avoid interference by the output of the TX shift register (TXSER OUT H), four addresses are provided to represent the four combinations of TCRC OUT H and TXSER OUT H. If TCRC OUT H is a 0, address 8 or 9 is selected which makes EIA XMIT DATA a SPACE (0). If TCRC OUT H is a 1, address 12 or 13 is selected which makes EIA XMIT DATA a MARK (1). In each case, DDROM output CRC CLK EN H is asserted to keep the TX CRC logic operating.

26. Stuffing 0s during transmission of the CRC character is required. If five consecutive 1s are detected during transmission of the 16-bit CRC character, the T1BC counter asserts T1BC=5 H. This signal, along with SCRC (1) H, selects Data Path Control ROM address 18. This asserts only DPCROM output TSPACE H which indicates that a CRC transfer is in progress and a 0 must be stuffed in the next bit position of the data stream. TSPACE H goes to the Data Decode ROM. All combinations of TCRC OUT H and TX SER OUT H (addresses 32, 33, 36, or 37) yield no DDROM outputs; therefore, TXDT H is low which makes EIA XMIT DATA a SPACE (0).

27. The transmitter continues sending the CRC character with stuffed 0s as required. At the start of the last (16th) CRC bit, the TCSC counter overflows and generates LOAD H. At this point, the ROMs are in the following states.

Function Decode ROM

Inputs	Adrs	Asserted Outputs
SEND (1) H SENTXAC (1) H SEND FLAG H TEOM (1) H SKPLD H	55	TFLG H and ENTXAC H. Inter-message flag coming up. Keep TXACTIVE set.

Data Path Control ROM

Inputs	Adrs	Asserted Outputs
SFLG (1) H LOAD H	33	ENTCSC and TSPACE H. Send SPACE which is first bit of flag character.

Data Decode ROM

Inputs	Adrs	Asserted Outputs
SKPLD H TSPACE H	96	No asserted outputs. Send SPACE. First bit of inter-message flag.

Summary
The address, command, and two information characters have been loaded by the program and transmitted by the DUP11. All these characters are treated like data so the 0 stuffing logic and CRC logic are enabled during their transmission. Early in the transmission of the CRC character, the program cleared TEOM and set TSOM. The 16-bit CRC character has been transmitted also (with 0 stuffing enabled). The first bit (SPACE) of the inter-message flag character is being transmitted.

28. Signal ENTXDONE H is asserted during the last bit of the CRC character (step 27) and is sent to the preset input gating for the TXDONE flip-flop. When TXCLKP H goes high as a result of the first data set clock pulse for the inter-message flag, the TXDONE flip-flop is directly set.

29. When LOAD H times out, the ROMs are in the following states.

Function Decode ROM

Inputs	Adrs	Asserted Outputs
SEND (1) H SENTXAC (1) H SFLG (1) H	56	ENTXAC H. Keep TXACTIVE set. Finish flag character, then transmit data.

Data Path ROM
The only asserted input is SFLG (1) H which gives address 32. Only output ENTCSC H is asserted which means that a flag character is being transmitted.

Data Decode ROM
The only asserted input is SSOM (1) H which gives address 128. Only output TXDT H is asserted which means that the second bit (MARK) of the inter-message flag character is being transmitted.

30. The transmitter sends the inter-message flag character and then continues with the remaining characters. In this example, the inter-message flag is followed by an address character, command character, and an information character, which are all under program control. They are automatically followed by the CRC character and terminating flag character, after which the line is set to the idle (MARK) state. Because another message is not pending, the line is forced to the MARK hold state, TXACTIVE is cleared and TXDONE is set.

31. The discussion continues at the point at which the last bit of the terminating flag character has been transmitted. The following conditions exist.

    TX ACTIVE is set
    TX DONE is cleared
    TSOM is cleared
    TEOM is set
    SEND is cleared
    LOAD H is asserted

    With LOAD H asserted, $\overline{\text{TXCLKD H}}$ clocks the Bit Sync Buffer and drives SENTXAC (1) H and SFLG (1) H low. The next $\overline{\text{TXCLK H}}$ pulse clocks the TXACTIVE flip-flop and clears it because its D input [SENTXAC (1) H] is low. The 1 and 0 outputs of the TXACTIVE flip-flop go to a pulse generator (7408 AND gate and RC delay network) to generate a 120 ns pulse when TXACTIVE is cleared. This pulse is inverted and directly sets the TXDONE flip-flop.

    When the TXACTIVE flip-flop is cleared, signal TXACT (1) L goes high. It is a qualifying input to the transmitter output data line interface logic (sheet 5). TXACT (1) L is ORed with SEND (1) L, which is also high, to drive EN DATA H low. This signal drives EIA XMIT DATA low which is the MARK state.

Signal ENDATA H (shown as $\overline{\text{EN DATA L}}$) goes to the clear input of the TXDT flip-flop. When it goes low, the TXDT flip-flop is cleared. As long as it is held in this state by the low signal on the clear input, it does not respond to the clock signal (TXCLKP L).

Summary
The terminating flag character has been transmitted which indicates the end of the message. One bit time after the last flag bit, the hardware clears TXACTIVE which drives the transmit line to a MARK hold state and sets TXDONE.

4.3.11.3 Typical Receive Operation (SDLC)

The following example is a detailed discussion of a typical DUP11 receive operation using SDLC protocol with the receiver as a primary station.

The message consists of the following sequenced characters: three flags, address, command, data, two CRC, and closes with two flags.

The flag and CRC characters are controlled by the logic in the transmitting DUP11. The address, command, and data characters are controlled by the program.

In the SDLC primary mode, all characters subsequent to the last starting flag character are presented to the program.

In the SDLC secondary mode, the character subsequent to the last starting flag is the address of the secondary receiving station to which the message is being sent. This address is stored in bits 0–7 of the PARCSR where it is compared with the received address. If it does not match, the receiver keeps searching for a flag. If it does match, RXACT is set to indicate the start of a new message. The received address character is not presented to the program. The character following the address is the first character presented to the program.

References

Receiver Timing Diagram (Figure 4-33) and Timing for First Recognized Flag Character (Figure 4-34)

Receiver Logic [drawing D-CS-M7867-0-1, sheet 2 (BSI2)]

Listings for the receiver ROMs (Decode and Function ROMs)

Detailed logic descriptions (Chapter 4) for detailed explanation of specific areas as required

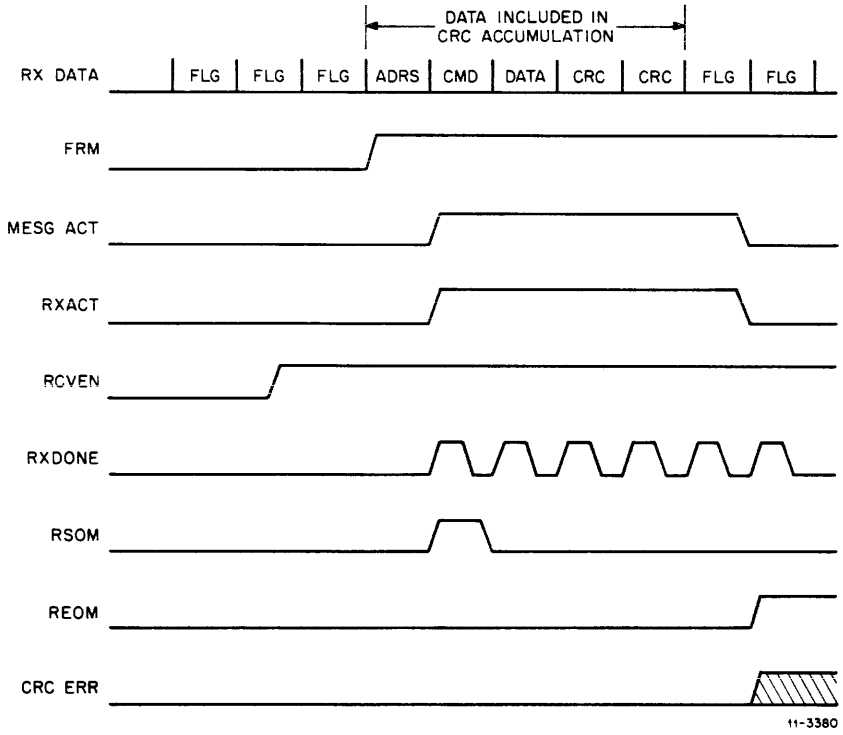


Figure 4-33 Typical SDLC Receive Operation

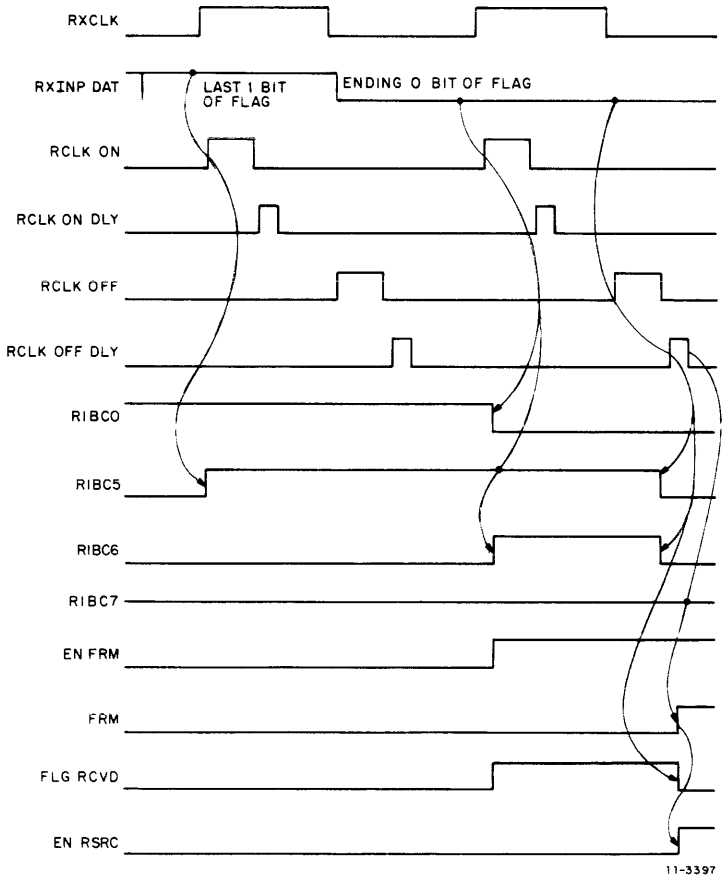


Figure 4-34 Timing for First Flag Character

Details of a Typical SDLC Receive Operation

1. At the beginning, the receiver enable bit (RCVEN) is cleared which resets the receiver timing and control functions. At this point the receiver does nothing.

2. Approximately halfway through the second flag character, the program sets the RCVEN bit. The first and second flags go unrecognized but the third flag is recognized.

3. At the start of the second flag, no inputs to the decode ROM are asserted; hence, no outputs are asserted. As the flag is received, it is not until the eighth bit that the decode ROM asserts an output as shown below.

   Decode ROM

Asserted Inputs	Adrs	Asserted Outputs
RIBC5 H	6	EN FRMH, FLG RCVD H, and ENRSRC H, which indicates reception of the second flag character.

   Decode ROM output ENRSRC H goes to the AND gate in the clock signal line for the receiver shift register. It is ANDed with RCLKON DLY H to produce RSR CLK H. This signal is inverted to generate RSR CLK L which clocks the shift register once each bit time.

   Decode ROM outputs EN FRM H and FLG RCVD H are inputs to the Function ROM. At the time of the eighth flag bit, signal RCSC MAX H is also asserted. This signal is a Function ROM input also; therefore, the ROM goes to the following state.

   Function ROM

Asserted Inputs	Adrs	Asserted Outputs
EN FRM H FLG RCVD H RCSC MAX H	11	FRAME H which indicates reception of the second flag character.

   Signal FRAME H goes to the receiver control flags (RCF) flip-flop and is clocked in by RCLK OFF DLY L. At the output of the $\overline{\text{RCF}}$ flip-flop, this signal is designated $\overline{\text{FRM}}$ (1) H and it is fed back as an input to the Decode ROM.

4. The third flag character is being received now. The discussion picks up at the eighth bit of the third flag character. The situation is very similar to that shown in step 3. The outputs of the Decode ROM are the same; however, the address is 38 because FRM (1) H is asserted. This indicates reception of the third flag. The Function ROM is in the same state as that shown in step 3. During the flag eighth bit, RIBC0H is low; therefore, RCLK OFF DLY L clears the RIBC counter. All outputs of the RIBC counter go low and the only asserted input of the decode ROM is FRM (1) H.

5. At this point, which is the start of the address character, the ROMs are in the following state:

Decode ROM

Asserted Inputs	Adrs	Asserted Outputs
FRM (1) H	32*	EN FRM H and EN RSRC H, which keeps the receiver framed and the shift register enabled.

*Address 32 is selected if the received data, as represented by the state of R1BC 0 H, is a 0. If R1BC 0 H is a 1, address 33 is selected and the ROM outputs are the same. If R1BC 0 H is a 0 that must be removed, R1BC 5 H is also asserted and address 34 is selected. ROM output EN RSRC H is not asserted in this case, which removes the stuffed 0 from the data.

Function ROM

Asserted Inputs	Adrs	Asserted Outputs
EN FRM H	8	FRAME H

These are the conditions that prevail during a normal data transfer.

6. At the eighth bit of the address character, RCSC MAX H is asserted. The function ROM is in the following state now:

Function ROM

Asserted Inputs	Adrs	Asserted Outputs
EN FRM H RCSC MAX H	9	FRAME H, EN RXACT H, EN MESG ACT H, and RSR→RXDBUF.

Signals FRAME H, EN RXACT H, and EN MESG ACT H go to the receiver control flags (RCF) flip-flop. They are clocked in by the trailing edge of RCLK OFF DLY L. However, just prior to this clock signal, RCLK OFF H is ANDed with RSR→RXDBUF to generate LD RXDBUF H. This loads the address character into the RXDBUF and asserts RSTR MESG H which is bit 8 of the RXDBUF register. Signals RSR→RXDBUF and RCLK OFF H are also ANDed to preset the RXDONE flip-flop [RXDONE (1) H is asserted]. Because of propagation delays, RXDONE (1) H is set just after the RXDBUF register is clocked to avoid setting the receiver overrun bit [REC OVERUN (1) H]. When the program accepts the address character by reading the RXDBUF register, the RXDONE bit is cleared. If the program does not accept the character, the RXDONE bit remains set, and when the RXDBUF register is loaded again, the receiver overrun bit is set.

Before summarizing the receiver operation (primary mode in SDLC protocol), the discussion digresses to show in general what would happen at the end of the address character if the SDLC secondary mode was enabled.

If the secondary mode is selected, signal SEC MODE H is asserted. No change in operation is noted up to and including step 5. That is, the Decode ROM and Function ROM outputs are the same although the addresses are different. At step 6 (eighth bit of address), ADRS+SYNC RECD is asserted and the Function ROM only asserts FRAME H and EN RXACT. This keeps FRM enabled and sets TXACT at the end of the address character. Signal RSR→RXDBUF is not asserted so the address character is not presented to the program. The next character is the first one presented to the program. Data handling and message ending are the same as those described in the primary mode.

Summary

Two flag characters have been recognized, clocked into the receiver shift register, but have not been loaded into the RXDBUF register. The receiver looks at flag characters as synchronizing characters, but the program has no interest in them. The program has accepted the address character. If the CRC logic is enabled, the receiver CRC starts its accumulation with the address character because RXACT is set and the DUP11 is operating in the SDLC mode.

7. The receiver is now in the state that takes care of normal data transfers. All characters between the beginning and ending flags (address, command, data, and CRC) are treated as data and are included in the CRC computation, if CRC is enabled. This state continues until reception of the first terminating flag.

   In the normal data transfer state, the Decode ROM keeps EN FRM H and EN RSRC H asserted (address 32 or 33) which keeps the receiver framed and the receiver shift register clock input enabled. However, if a stuffed 0 is detected, input RIBC5 H is asserted, along with FRM (1) H, which gives address 34. Under these conditions, the only output asserted is EN FRM H. With EN RSRC H not asserted, the receiver shift register and receiver CRC register are not clocked; hence, the stuffed 0 is removed.

   The function ROM keeps FRAME H, EN RX ACT H, and MESG ACT H asserted (address 56). At the end of each character, RCSC MAX H is asserted and the Function ROM goes to address 57. In addition to asserting the three outputs mentioned above, the ROM also asserts RSR→RXDBUF H to load the accumulated character into the receiver data buffer and set the RXDONE bit.

   The DUP11 is operating in the primary mode (SEC MODE H is cleared); however, it is possible for the binary equivalent of the secondary address or SYNC character of the receiving DUP11 to appear. As a result, ADRS+SYNC RCVD H could be asserted as an input to the Function ROM. This action produces an illegal ROM address which asserts RSR→RXDBUF and allows the character to be loaded. Once RXACT is set, this action should be treated as normal. The appearance of the secondary address or SYNC character does not affect the operation of the DUP11 in the SDLC primary mode.

8. Assume now that the first of the terminating flag characters has been received. During the last bit of the flag, signals RIBC5 H, RIBC6 H, and RCSC MAX H are asserted. The ROMs are in the following state:

Decode ROM

Asserted Inputs	Adrs	Asserted Outputs
FRM (1) H RIBC6 H RIBC5 H	38	EN FRM H, EN RSRC H, and FLG RCVD H, which denotes reception of the closing flag.

These Decode ROM outputs represent the reception of the closing flag. They also occur via address 6 at the start of the message (step 3).

Function ROM

Asserted Inputs	Adrs	Asserted Outputs
RX ACT (1) H MESG ACT (1) H EN FRM H FLG RCVD H RCSC MAX H	59	FRAME H, EN MESG ACT H, and RSR →RXDBUF. This denotes normal message ending.

Signals FLG RCVD H and MESG ACT (1) H, which are asserted, are ANDed to become the input to the receiver end of the message bit in the RXDBUF register. When the register is clocked, this bit is set (REND MESG H is asserted) to denote the end of the message.

At the end of the first terminating flag, the receiver assumes the state of searching for additional flags. Signals MESG ACT and RXACT are cleared and REOM is set. Conditions are similar to those at the start of the message, except that FRM is set. The last data received by the program are the two CRC characters.

Summary

The message has been received and the receiver looks at flags until more data is seen, at which time RXACT is set again. At the end of the terminating flag the CRC error flag [RX CRC ERR (1) H] is asserted by the hardware if the message contained an error. The receiver can be shut down at any time during the message via the program by clearing the receiver enable bit.

CHAPTER 5
MAINTENANCE

5.1 SCOPE
This chapter lists required test equipment and provides a complete description of DUP11 preventive and corrective maintenance procedures.

5.2 MAINTENANCE PHILOSOPHY
Basically, DUP11 maintenance consists of preventive and corrective maintenance procedures, diagnostic programs, and a maintenance log. The preventive maintenace procedures are performed regularly in an attempt to detect any deterioration due to aging and any damage caused by improper handling of the module. The corrective maintenance procedures are performed to isolate and repair faults in module circuitry only after it has been determined that the module is faulty. The maintenance log is used to record all maintenance activities for future reference and analysis; hopefully, the log will facilitate future maintenance action and aid in detecting any component failure pattern that may develop.

5.3 PREVENTIVE MAINTENANCE
There is no specific DUP11 PM schedule. A general check of voltages and connections should be done when system PM is performed. After handling DUP11 modules or cables, a complete checkout of the device is required. This includes running all diagnostics and, if possible, the interprocessor test.

5.4 TEST EQUIPMENT REQUIRED
Maintenance procedures for the DUP11 require the standard test equipment and diagnostic programs listed in Table 5-1, in addition to standard hand tools, cleaners, test cables, and probes.

Table 5-1 Test Equipment Required

Equipment	Manufacturer	Designation
Multimeter	Triplett or Simpson	Model 630-NA or 260
Oscilloscope	Tektronix	Type 453
X10 Probes (2)	Tektronix	P6008
Module Extender	DEC	W904 or equivalent
Diagnostics	DEC	CZDPB-CXDPE

5.5 CORRECTIVE MAINTENANCE

The corrective maintenance procedures are designed to aid the maintenance technician in isolating and repairing faults within the DUP11 module. Hence, the technician must be otherwise equipped to determine that the DUP11 is in fact at fault.

The diagnostic programs are used by the technician to locate and isolate faults. The diagnostics exercise the DUP11 in three distinct maintenance modes and provide printouts indicating the results. The printouts direct the technician to a particular logic area such as the XMTR or RCVR logic. The technician uses standard test equipment (scope and probe) to further isolate the fault to a specific circuit component.

5.5.1 Maintenance Modes

The DUP11 can be operated in three maintenance modes during servicing:

Internal maintenance
System test
External maintenance

5.5.1.1 Internal Maintenance – When the internal maintenance mode is selected, the DUP11 may remain attached to the modem. The entire DUP11 can be tested with the exception of the attached cables, H3001 connector module, modem control logic, and the level converters.

Single-step clocking is provided by the diagnostic program through the toggling of the maintenance clock bit (MAI SS CLK) which is bit 13 of the TXCSR.

The data is looped internally from the transmitter output to the receiver input. Actually, two inputs are available. One is maintenance input data (MAI DATA) which is TXCSR bit 10. It can be toggled by the diagnostic program. If this input is used, the transmitter control logic is inhibited and the transmitter output TXDAT is cleared. The other input is TXDAT which is used if the transmitter control logic is to be tested. In this case, input MAI DATA must be cleared.

The looped transmitter output is monitored by the diagnostic program by reading TX MAINT DATA OUT which is TXCSR bit 14.

The received data and clock signals from the modem are disabled at their point of input. The transmitter data lead at the modem is held in the MARK state.

Caution should be exercised in dealing with the modem control and status lines because they are not inhibited.

5.5.1.2 System Test – The systems test mode is provided to supply clocking to the receiver and transmitter logic that is asynchronous to the computer system under test. This closely simulates the conditions encountered when operating with the actual modem. This clocking is supplied by a free-running clock contained within the DUP11 at a 5 kHz ± 20% rate. When this mode is selected, the data is internally looped from the output of the transmitter to the receiver input. The maintenance test point between the receiver and the transmitter is not enabled in this mode. As in the internal maintenance mode, all received data and clock signals from the modem are disabled. The transmitter output to the modem is also inhibited. The BC22F cable may remain connected to the modem. System's test programs such as DECX11 should utilize this mode when peripherals are being interactively tested.

5.5.1.3 External Maintenance – The function of the external maintenance mode is to check the complete interface, including level converters, H3001 distribution panel, and cables – it can be used in some cases to evaluate system's software. The DUP11 is physically disconnected at the modem end of the BC22F cable. The modem is replaced by the H325 test connector (Figure 5-1).

The simulated data set clocking is supplied by a free-running clock contained within the DUP11 at a 10 kHz rate ± 20%. This clock is supplied at RS-232-C levels from the SCTE lead (external clock) of the BC22F cable. The H325 connector returns this signal to the SCT (transmit clock) and SCR (receiver clock) leads on the DUP11. This produces the same asynchronous clocking effect as in the system's test mode.

The transmitted data is sent over the SD (send data) lead of the BC22F cable and returned on the RD (received data) lead at RS-232-C levels.

All modem control signals can also be tested to ensure that the proper level conversion and cable paths exist. The paths are shown below:

Request to Send provides source for Clear to Send and Carrier
Data terminal ready provides source for Ring and Data Set Ready
Secondary Transmit provides source for Secondary Receive

All paths between the interface and the modem can be verified in this mode.

5.5.2 Diagnostics

Four diagnostics are used to verify proper operation of the DUP11. Detailed discussions of the content, use, and interpretation of each diagnostic is documented separately. The documents and diagnostics are shipped with the DUP11. The diagnostics are identified as CZDPB-CZDPE. Refer to Table 2-2 for H3001 switch configuration when using diagnostics that exercise the cables and/or H325 test connector.

The diagnostic document for each diagnostic is structured as follows:

1. Abstract
2. List of test by test number and line number. Includes concise statement of test function.
3. Table of contents for specific diagnostic document which includes list of tests by title and line number.
4. Description of monitor which interacts with operator and sets up program parameters.
5. Details of the tests.
6. Cross-reference table.

A brief description of each diagnostic is given below.

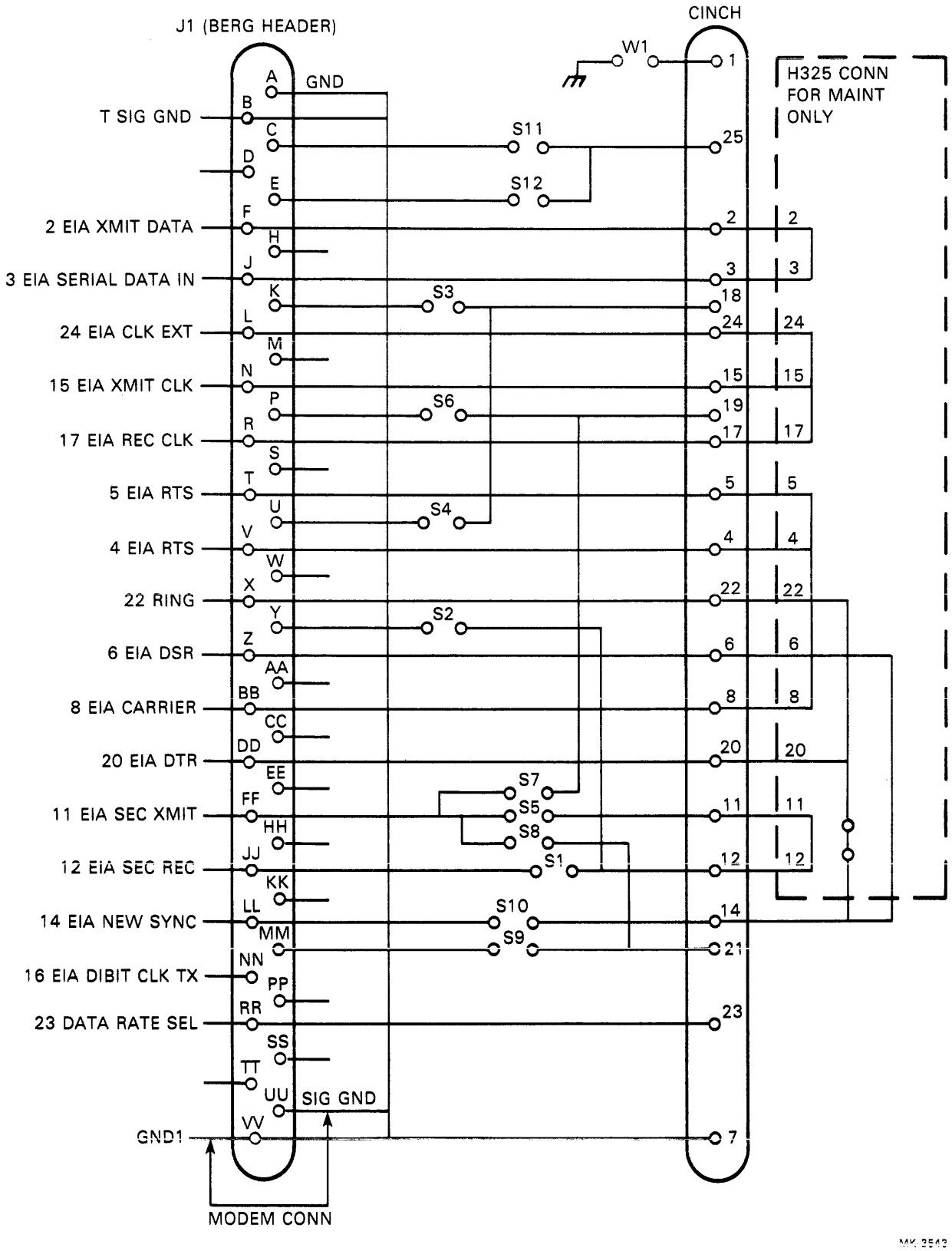


Figure 5-1 Schematic of H3001 Distribution Panel with H325 Test Connector

CZDPB Basic and Off-Line SDLC Transmitter Tests – Verifies that the DUP11 works on the Unibus and that all registers are cleared by an Initialize signal from the Unibus and a DUP11 DEVICE RESET signal. Verifies that there is no dual addressing of the register. Each register is tested one bit at a time and all bits at once. Tries to write into read-only bits and RXDBUF; tries to read write-only bits and PARCSR. Tests all read/write bits. Verifies all SDLC transmitter functions in internal maintenance mode; that is, the program clocks the device using the PK CLK routine. Verifies bit stuffing by the transmitter logic. In checking data, the program emulates the hardware and compares the emulated software bit with the hardware bit. The transmitter CRC logic is verified using the CCITT polynomial. The CRC character is checked like data except that it is calculated in a special routine and compared with the hardware-generated CRC character.

CZDPC Off-Line SDLC Receiver and Modem Control and Interrupt Tests– Verifies all SDLC receiver functions in the internal maintenance mode. Test data and receiver CRC character. Uses same emulation methods employed in diagnostic CZDPB. It also verifies modem control functions, provided the test connector is installed. Verifies interaction of Data Set Change A and Data Set Change B bits in responding to signals from the modem, provided that jumpers W5 and W6 are installed correctly. Verifies that RXDONE and ADAT SET CH each generate a receiver interrupt request. Verifies that TXDONE generates a transmitter interrupt request. Verifies that the receiver interrupts before the transmitter.

CZDPD Off-Line SDLC and DEC MODE Data and Function Tests – Verifies the interrupt control logic. Operates the device in SDLC in system test mode and external mode, if the test connector is installed. Verifies SDLC functions, secondary station operation, and abort sequences. Checks long and short data tests to verify bit stuffing and CRC operation. It also operates the device in DEC MODE (DDCMP protocol) in system test mode and external mode, if the test connector is installed. Verifies all DEC MODE functions. Uses CRC-16 polynomial to check CRC operation.

CZDPE Confidence Test – This brief test requires a dialog with the operator to give the DUP11 parameters. It verifies that the DUP11 operates on both SDLC and DDCMP protocols and that the EIA level conversion logic is operational.

Diagnostics CZDPB-CZDPD should be run in sequence; then diagnostic CZCPE should be run. Each diagnostic must make three passes without an error.

System testing consists of running DECX11 module CXDPB to exercise all DUP11s in a system. Three passes without an error should be made. Only four DUP11s can be tested with one DECX11 module.

In diagnostics CZDPB and CZDPC, the routine for the software clock in the internal maintenance mode is called the PK CLK routine. It is accomplished by using a TRAP call followed by an argument which is a number. The routine uses the argument to determine the number of half-clocks to implement. At the end of the specified number of half-clocks, control is returned to the test that made the call.

In diagnostic CZDPB, during all register testing, the same HALT routine is used. The following assignments are used:

R3 is used for the failing register
R4 is used for the unknown found
R5 is used for the expected results.

A comparison is made, and if a match does not occur, the information is presented to the operator.

CZDPOB – This is an overlay program for use with the ITEP monitor. This ITEP module allows the operator to perform on-line testing of the DUP11 to determine the status of the modem and line.

APPENDIX A
PDP-11 MEMORY ORGANIZATION AND ADDRESSING CONVENTIONS

The PDP-11 memory is organized in 16-bit words consisting of two 8-bit bytes. Each byte is addressable and has its own address location: low bytes are even-numbered and high bytes are odd-numbered. Words are addressed at even-numbered locations only and the high (odd) byte of the word is automatically included to provide a 16-bit word. Consecutive words are therefore found in even-numbered addresses. A byte operation addresses an odd or even location to select an 8-bit byte.

The Unibus address word contains 18 bits identified as A(17:00). Eighteen bits provide the capability of addressing 256K memory locations, each of which is an 8-bit byte. This also represents 128K 16-bit words. In this discussion, the multipler K equals 1024 so that 256K represents 262,144 locations and 238K represents 131,072 locations. The maximum memory size can be used only by a PDP-11 processor with a memory management unit that utilizes all 18 address bits. Without this unit, the processor provides 16 address bits which limits the maximum memory size to 64K (65,536) bytes or 32K (32,768) words.

Figure A-1 shows the organization for the maximum memory size of 256K bytes. In the binary system, 18 bits can specify 2^{18} or 262,144 (256K) locations. The octal numbering system is used to designate the address. This provides convenience in converting the address to the binary system that the processor uses as shown below.

The highest 8K address locations (760000–77777) are reserved for internal general registers and peripheral devices. There is no physical memory for these addresses; only the numbers are reserved. As a result, programmable memory locations cannot be assigned in this area; therefore, the user has 248 bytes or 124K words to program.

A PDP-11 processor without the memory management unit provides 16 address bits that specify 2^{16} or 65,536 (64K) locations (Figure A-2). The maximum memory size is 65,536 (64K) bytes or 32,768 (32K) words. Logic in the processor forces address bits A(17:16) to 1s if bits A(15:13) are all 1s when the processor is master to allow generation of addresses in the reserved area with only 16-bit control.

17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00	Address Bit
0	0	1	0	0	1	1	1	1	1	1	0	0	0	0	0	1	0	Binary
1			1			7			6			0			1			Octal

Address Word Format

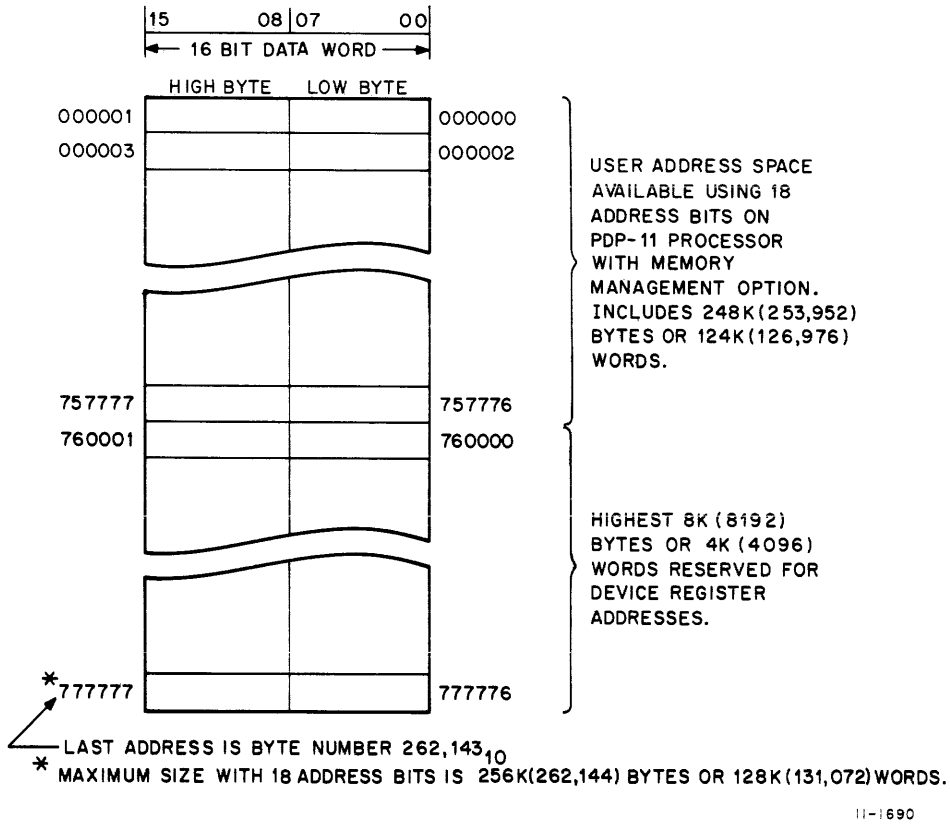


Figure A-1 Memory Organization for Maximum Size Using 18 Address Bits

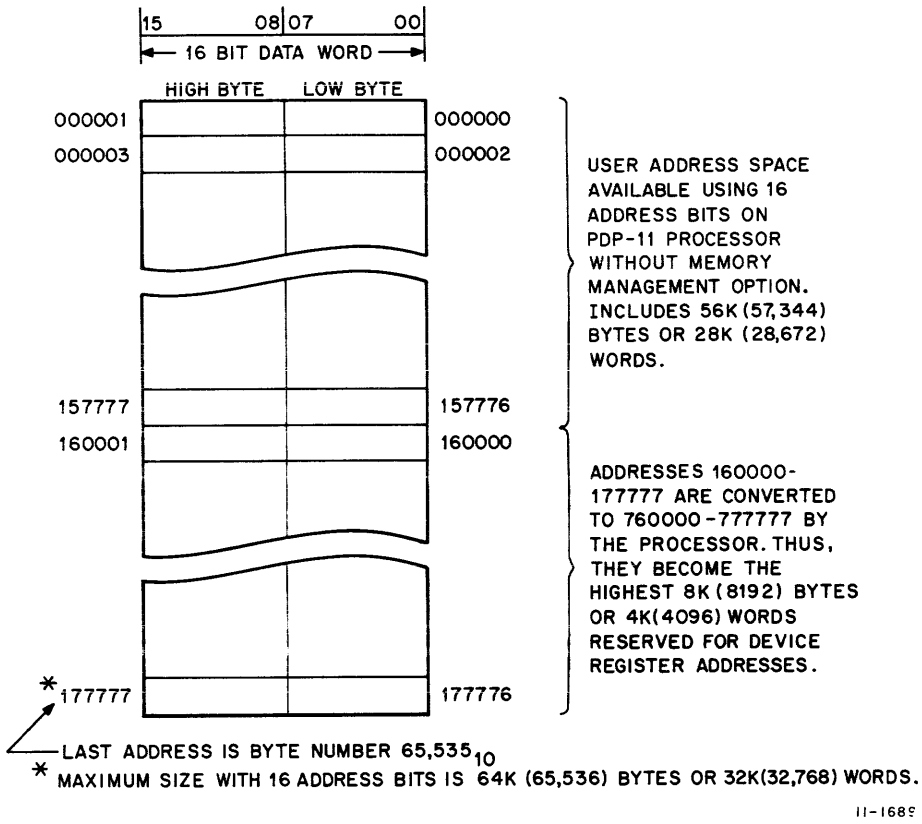


Figure A-2 Memory Organization for Maximum Size Using 16 Address Bits

Bit 13 becomes a 1 first at octal 160000 which is decimal 57,344 (56K). This is the beginning of the last 8K bytes of the 64K byte memory. The processor converts locations 160000–177777 to 760000–777777, which relocates these last 8K bytes (4K words) to the highest locations accessible by the bus. These are the locations that are reserved for internal general register and peripheral device addresses; therefore, the user has 57,344 (56K) bytes or 28,672 (28K) words to program.

Memory capacities of 56K bytes (28K words) or under do not have the problem of interference with the reserved area, because designations less than 160000 do not have a binary 1 in bit A13. No addresses are converted and there is no possibility of physical memory locations interfering with the reserved space.

PDP-11 memories are available in a variety of increments. The highest location of various size memories are shown below.

Memory Size		Highest Location
K-Words	**K-Bytes**	**(Octal)**
4	8	017777
8	16	037777
12	24	057777
16	32	077777
20	40	117777
24	48	137777
28	56	157777

APPENDIX B
LOGIC SYMBOLOGY

B.1 INTRODUCTION

The logic symbology used in the PDP-11 manuals and engineering logic is generally consistent with MIL-STD-806B Graphic Symbols for Logic Diagrams. Certain symbols are modified by DEC to allow direct reading of logic functions in detailed logic diagrams that show explicit electrical connections between logic elements. The modifications and other conventions are explained in the following paragraphs.

B.2 UNIBUS SIGNAL LEVELS

The Unibus has 56 dedicated signal lines. Negative logic is used for 51 lines and the remaining 5, BG(7:4) and NPG, use positive logic.

The definitions of positive and negative logic are:

Positive Logic

Signal Asserted: High = Logical 1 = +3 V
Signal at Rest: Low = Logical 0 = 0 V

Negative Logic

Signal Asserted: Low = Logical 1 = 0 V
Signal at Rest: High = Logical 0 = +3 V

In the logic diagrams, the signal name mnemonic is followed by an H or L to indicate the asserted state (logical 1) of the signal to be high (+3 V) or low (ground or 0 V). Using this convention, a grant line is called BUS BG2 H and a data line is called BUS D12 L.

B.3 EQUIVALENT GATE SYMBOLS

In the detailed logic diagrams, the gate symbols show the active state of the gate output. A small circle at the output shows that the active state is low (L). Absence of a small circle at the output shows that the active state is high (H).

A large number of NAND and NOR gates are used in DEC logic. The symbols for the NAND and NOR gates show an active low output. Frequently, an active high output is required from a NAND or NOR gate. In this case, a logically equivalent symbol is used to retain the concept of direct reading of logic functions.

For the NAND gate, the logically equivalent negated-input OR gate is used to show the active high output. For the NOR gate, the logically equivalent negated-input AND gate is used to show the active high output. These gate symbols and associated truth tables are shown in Figure B-1.

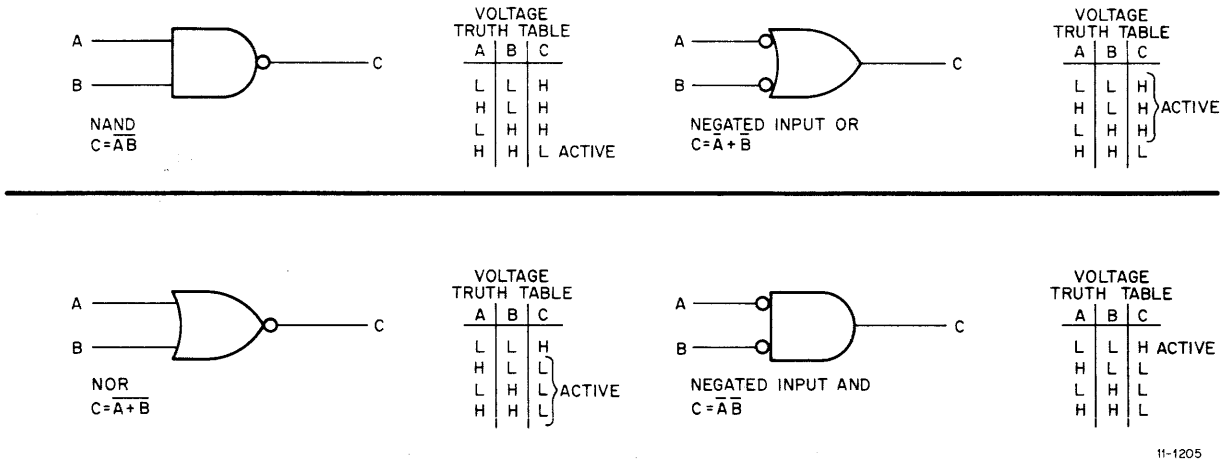


Figure B-1 Logically Equivalent Gates

B.4 4-OUTPUT TERMINAL FLIP-FLOP SYMBOLOGY

The 7474 D-type flip-flops in the engineering logic diagrams are shown as 4-output terminal devices. Most other users (and the IC manufacturers) show them as 2-output terminal devices, which represents only the physical output connections. Both the 4-output symbol and the 2-output symbol are shown in Figure B-2.

The flip-flop is a 2-state device with a pair of complementary outputs. The 4-output terminal symbology defines the polarity of the outputs for each state of the flip-flop. In this discussion, the states are set and reset which are obtained by clocking the flip-flop with its D input high or low, respectively. These states can also be obtained by enabling the PRESET and CLEAR inputs which override the clock. In Chapter 4 of this manual, as in most other DEC manuals, the distinction between set/reset and preset/clear is not maintained. A flip-flop is said to be either set or cleared, regardless of the method used to obtain the state.

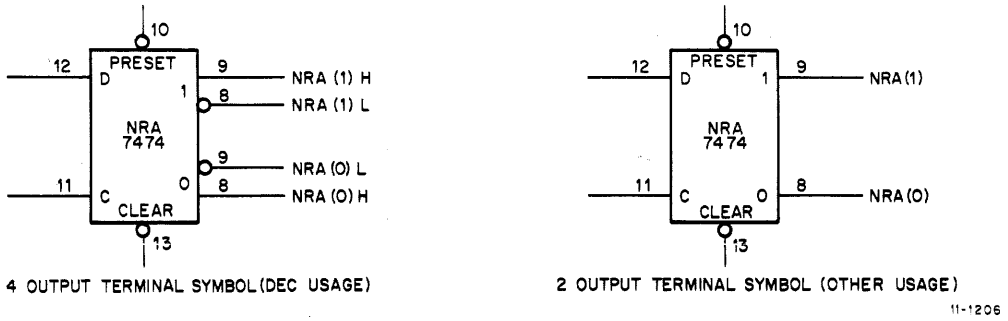


Figure B-2 Flip-Flop Logic Symbology

In the 4-output symbol (Figure B-2), the flip-flop name is NRA and the output signal designations contain the name, state (asserted or not asserted), and polarity (high or low). The state is either 1 for asserted (set) or 0 for non-asserted (reset). The polarity is either H (high or logical 1) or L (low or logical 0); i.e., assuming positive logic conventions in which H = 1 = +3 V and L = 0 = 0 V. For example, NRA (0) L means that this output is low when the flip-flop is reset (cleared). Usually the PRESET input is placed near the 1-output because it directly sets the flip-flops; the CLEAR input is placed near the 0-output because it directly resets the flip-flop. Physically, nothing has changed: flip-flop operation is still the same and there are two electrical outputs (pin 9 for the 1-output and pin 8 for the 0-output).

The 4-terminal symbol and signal designators allow direct reading of logic functions in detailed logic diagrams that show explicit electrical connections between the flip-flop outputs and other logic elements. The correct output polarity, pin number, and flip-flop state are read at a glance. This is not possible with the 2-terminal symbol without mental translations. A comparative example is shown in Figure B-3.

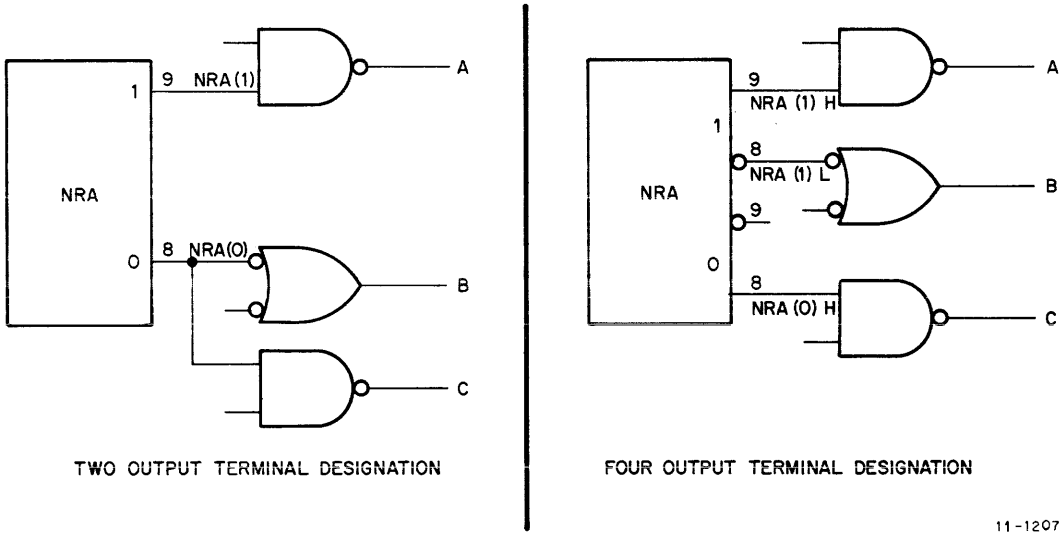


Figure B-3 Electrical Connections to Outputs of 2-Terminal and 4-Terminal Flip-Flops

B.5 REDEFINED 4-OUTPUT TERMINAL FLIP-FLOPS

Logically speaking, a redefined flip-flop is asserted (set) when clocked with a low signal on its D-input. Graphically, this is accomplished by reversing the output pin assignments and placing a circle on the D-input. The PRESET and CLEAR input pin designations are interchanged because their logical functions are reversed: PRESET directly resets the flip-flop, and CLEAR directly sets the flip-flop. Physically, the flip-flop operation is still the same. Redefinition is used to retain consistency in graphically representing the asserted state of a flip-flop in a detailed logic diagram; specifically, to produce the asserted state with a low signal on the D-input.

Pin designations and outputs for a standard 4-output terminal flip-flop and a redefined 4-output terminal flip-flop are shown in Figure B-4.

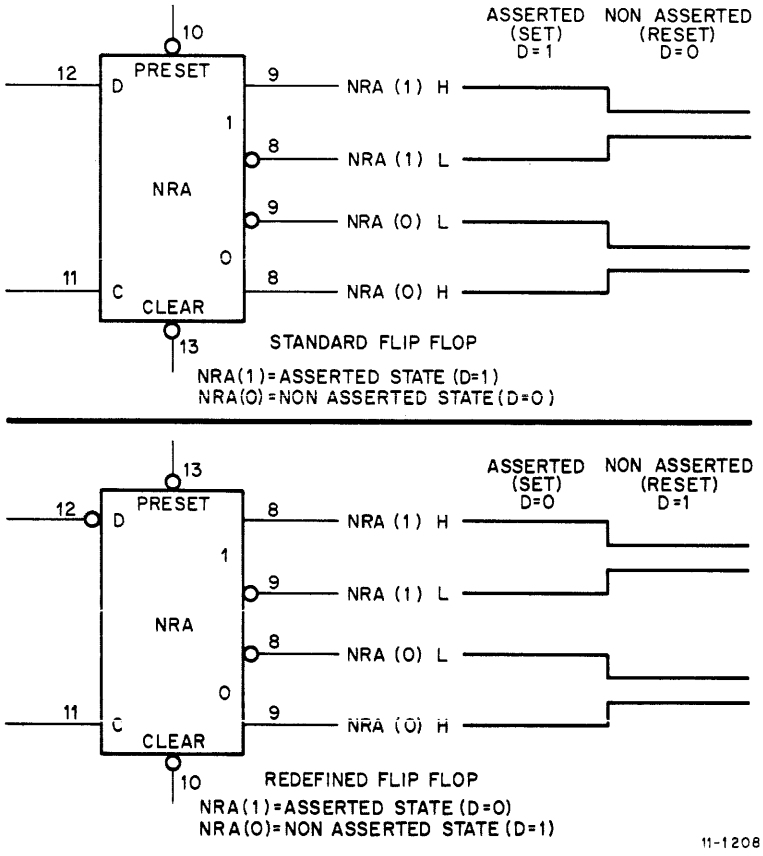


Figure B-4 Standard and Redefined 4-Terminal Flip-Flops

APPENDIX C
INTEGRATED CIRCUIT DESCRIPTIONS

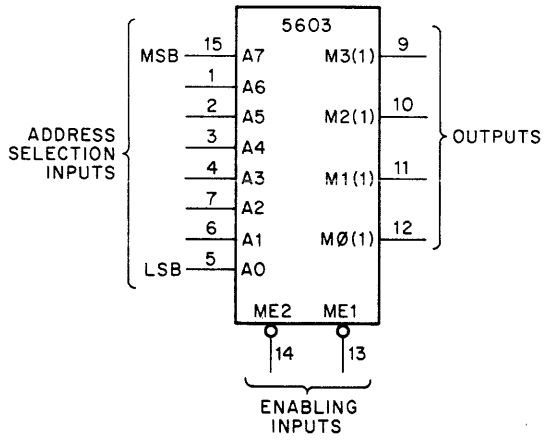
The MSI integrated circuits (ICs) shown in the engineering drawings are described in the following pages. The descriptions include one or more of the following items: pin/signal designations, equivalent logic schematic, and truth table. This information is a maintenance aid for troubleshooting to the IC level. The ICs are listed in Table C-1.

Table C-1 Integrated Circuits

Manufacturer's Part Number	Name
5603	1024-Bit Read-Only Memory
7442	4-Line to 10-Line BCD to Decimal Decoder
7474	Dual D-Type Edge Triggered Flip-Flop
8641	Quad Bus Transceiver
74123	Retriggerable Monostable Multivibrator with Clear
74153	Dual 4-Line to 1-Line Multiplexer
74161	Synchronous 4-Bit Counter
74164	8-Bit Parallel Out Serial Shift Register
74165	Parallel-Load 8-Bit Shift Register
74174	Hex D-Type Flip-Flop
74175	Quad D-Type Flip-Flop
74191	Synchronous Up/Down Counter

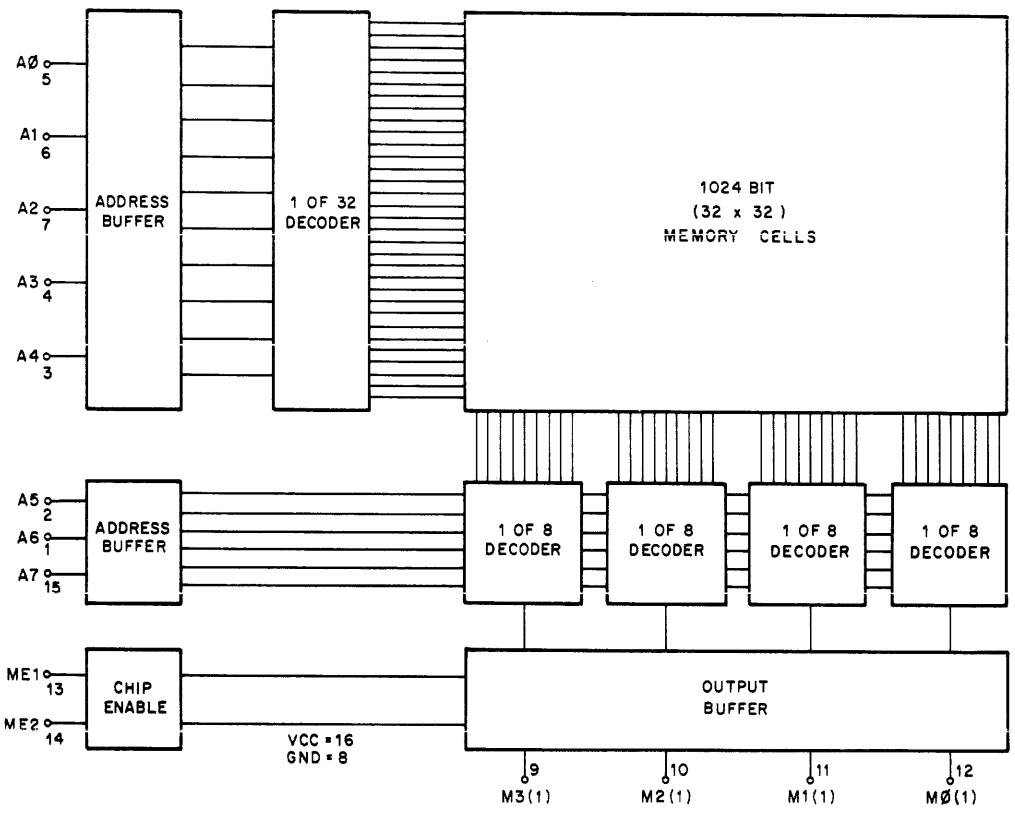
5603 PROGRAMMABLE READ ONLY MEMORY

The 5603 is a 1024-bit, bipolar programmable ROM that is organized into 256 words of 4 bits each. This ROM is equivalent to the 74187 ROM. The ROM can be read when both enabling inputs are held low. When enabled, the 4-bit output corresponds to the data programmed in the selected word.



IC-5603A

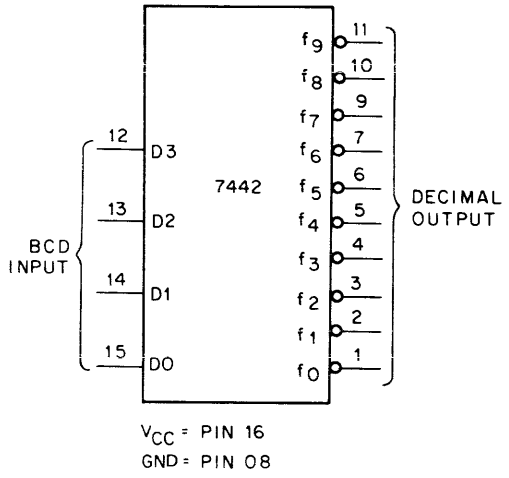
BLOCK DIAGRAM



IC-5603B

7442 4 LINE TO 1 LINE DECODER

These BCD-to-decimal decoders consist of eight inverters and ten 4-input NAND gates. The inverters are connected in pairs to make BCD input data available for decoding by the NAND gates.



7442
TRUTH TABLE

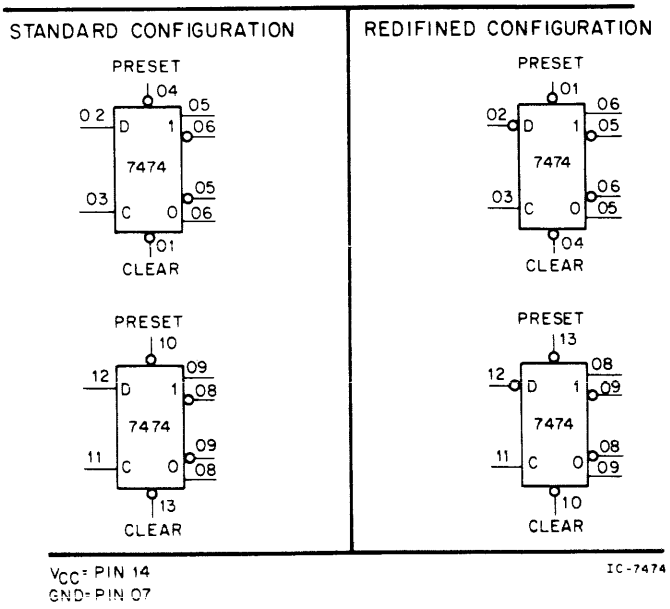
BCD Input				Decimal Output									
D3	D2	D1	D0	f0	f1	f2	f3	f4	f5	f6	f7	f8	f9
0	0	0	0	0	1	1	1	1	1	1	1	1	1
0	0	0	1	1	0	1	1	1	1	1	1	1	1
0	0	1	0	1	1	0	1	1	1	1	1	1	1
0	0	1	1	1	1	1	0	1	1	1	1	1	1
0	1	0	0	1	1	1	1	0	1	1	1	1	1
0	1	0	1	1	1	1	1	1	0	1	1	1	1
0	1	1	0	1	1	1	1	1	1	0	1	1	1
0	1	1	1	1	1	1	1	1	1	1	0	1	1
1	0	0	0	1	1	1	1	1	1	1	1	0	1
1	0	0	1	1	1	1	1	1	1	1	1	1	0
1	0	1	0	1	1	1	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1	1	1	1	1	1
1	1	0	0	1	1	1	1	1	1	1	1	1	1
1	1	0	1	1	1	1	1	1	1	1	1	1	1
1	1	1	0	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1

7474 Dual Flip-Flop

TRUTH TABLE FOR
7474 STANDARD CONFIGURATION
(EACH FLIP-FLOP)

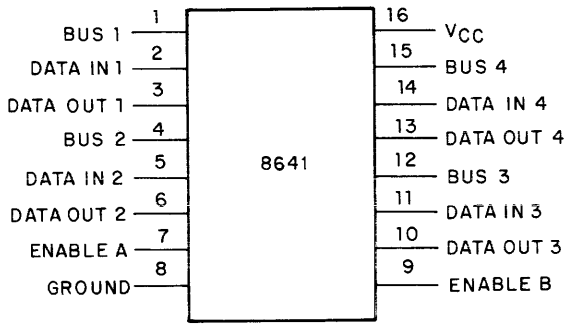
t_n			t_n+1	
Preset Pin 4(10)	Clear Pin 1(13)	D Input Pin 2(12)	1 Side Pin 5	0 Side Pin 6
High	High	Low	Low	High
High	High	High	High	Low
High	Low	X	Low	High
Low	High	X	High	Low
Low	Low	X	High	High

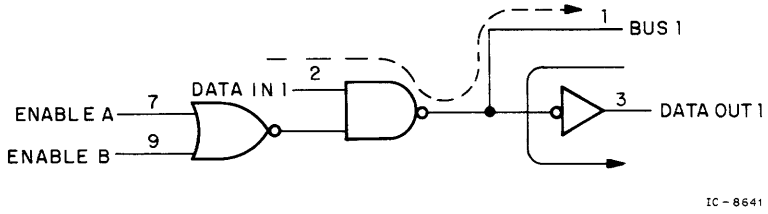
t_n = bit time before clock pulse.
t_n+1 = bit time after clock pulse.
X = irrelevant



8641 QUAD BUS TRANSCEIVER

The 8641 consists of four identical receiver/drivers and a single enabling gate in one package for interfacing with the PDP-11 Unibus. The transceiver drivers are enabled when ENABLE A and ENABLE B are both low. The other input of each driver is connected to the data to be sent to the Unibus. For example, when enabled, DATA IN 1 (pin 2) is read to the Unibus via BUS 1 (pin 1). During a write operation, data comes from the Unibus as BUS 1 (pin 1) and is passed through the receiver to the device as DATA OUT 1 (pin 3).

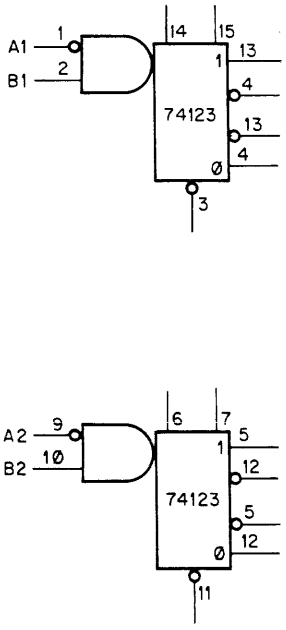




74123 RETRIGGERABLE MONOSTABLE MULTIVIBRATOR

The 74123 Monostable Multivibrator provides d-c triggering from gated low-level active (A) and high-level active (B) inputs. Overriding direct clear inputs and complementary outputs are also provided.

By triggering the input before the output pulse is terminated, the output pulse may be extended. The overriding clear capability permits any output pulse to be terminated at a predetermined time, independently of the external timing components.



+5V = PIN 16
GND = PIN 8

IC-74123A

TRUTH TABLE

INPUTS		OUTPUTS	
A	B	1	0
H	X	L	H
X	L	L	H
L	↑	⎍	⎴
↓	H	⎍	⎴

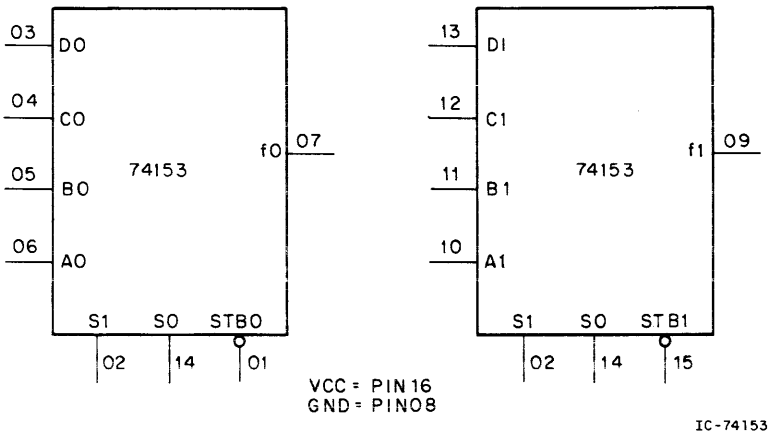
NOTE: H = high level (steady state), L = low level (steady state), ↑ = transition from low to high level, ↓ = transition from high to low level, ⎍ = one high-level pulse, ⎴ = one low-level pulse, X = irrelevant (any input, including transitions).

IC-74123B

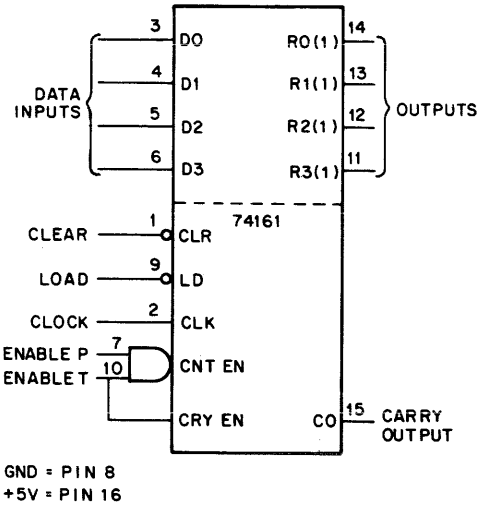
74153 DUAL 4 TO 1 MULTIPLEXER

ADDRESS INPUTS		DATA INPUTS				STROBE	OUTPUT
S1	S0	A	B	C	D	STB	f
X	X	X	X	X	X	H	L
L	L	L	X	X	X	L	L
L	L	H	X	X	X	L	H
L	H	X	L	X	X	L	L
L	H	X	H	X	X	L	H
H	L	X	X	L	X	L	L
H	L	X	X	H	X	L	H
H	H	X	X	X	L	L	L
H	H	X	X	X	H	L	H

Address inputs S0 and S1 are common to both sections.
H = high level, L = low level, X = irrelevant.



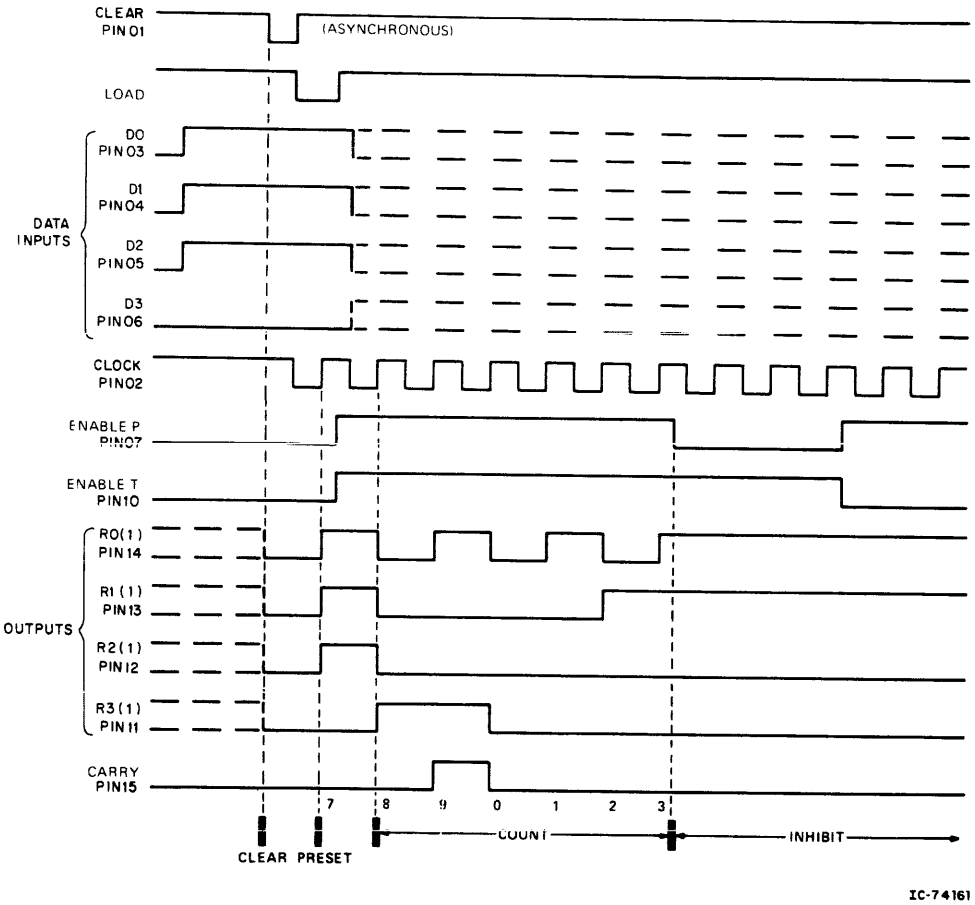
74161 Synchronous 4 Bit Counter



typical clear, preset, count, and inhibit sequences for 74161

Illustrated below is the following sequence:

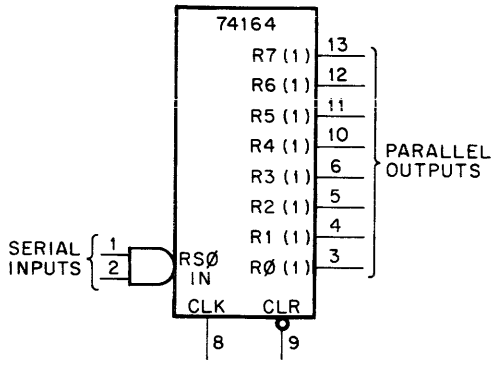
1. Clear outputs to zero.
2. Preset to BCD seven.
3. Count to eight, nine, zero, one, two, and three.
4. Inhibit



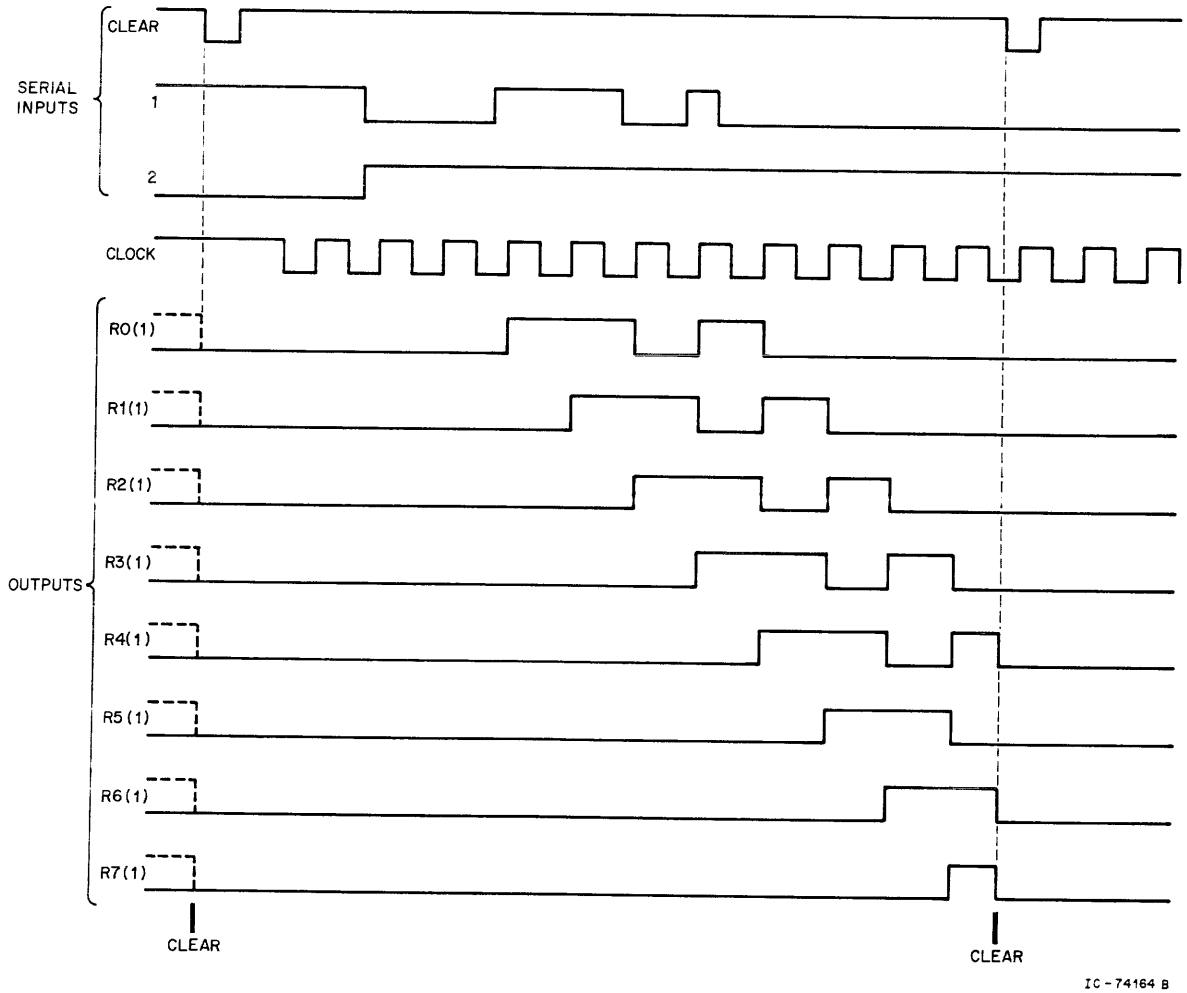
IC-74161

74164 8-BIT PARALLEL-OUT SERIAL SHIFT REGISTER

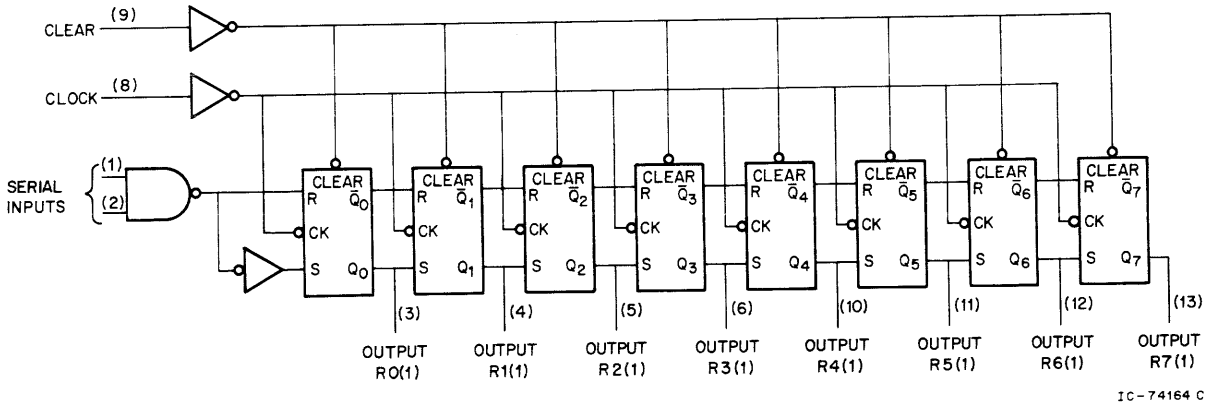
The 74164 is an 8-bit parallel-out serial shift register with gated serial inputs and an asynchronous clear. The register is clocked on the positive-going transition of the clock input.



IC-74164A

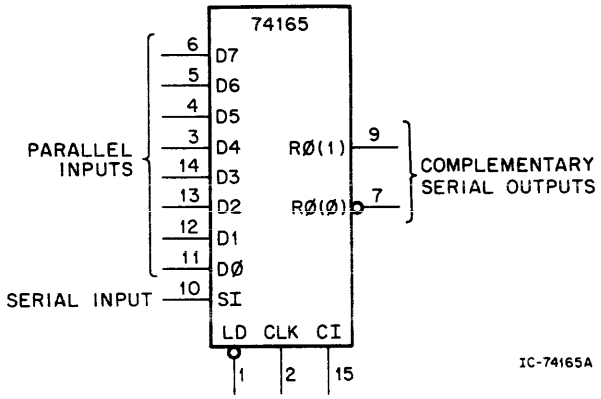


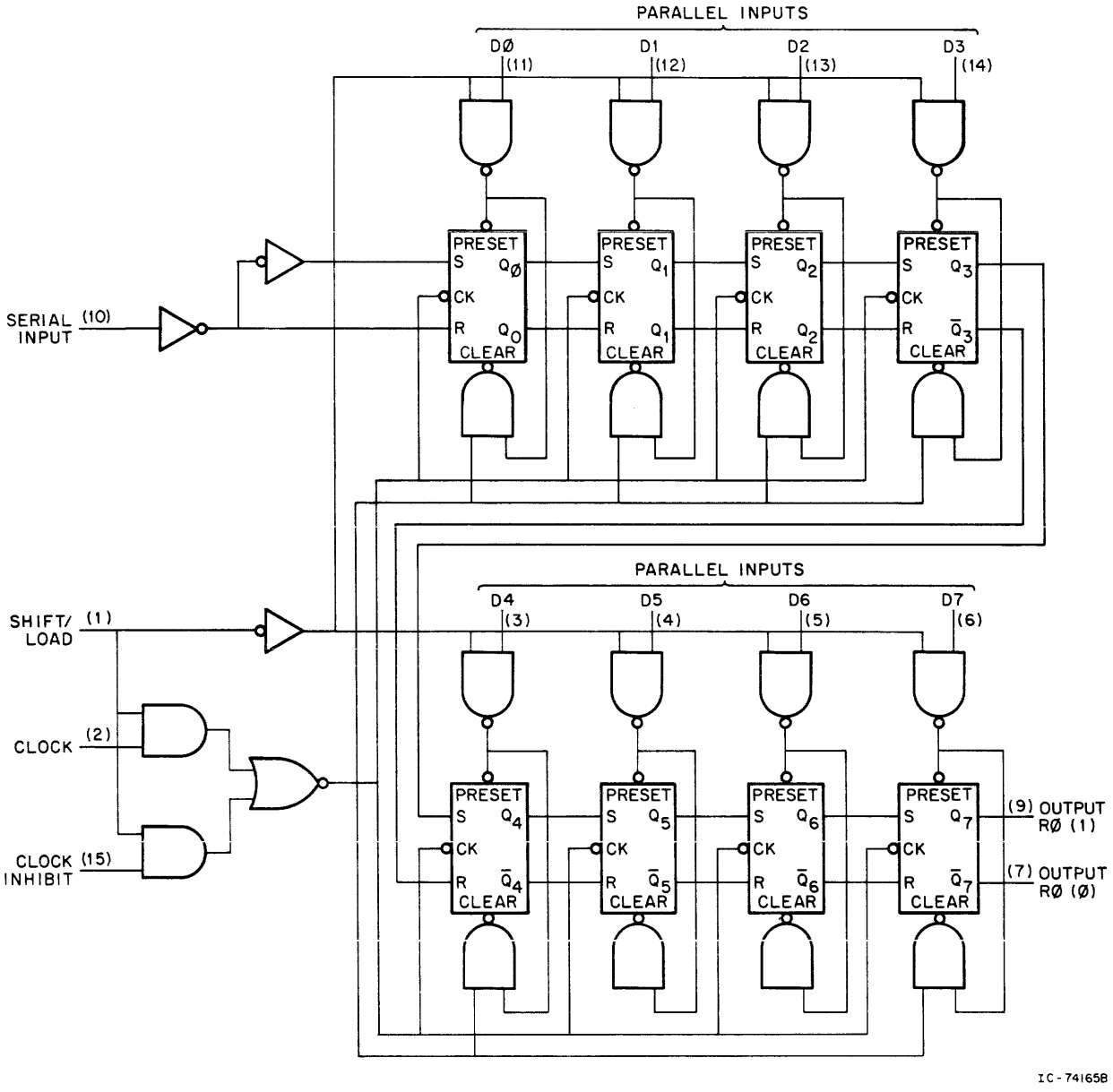
IC-74164 B

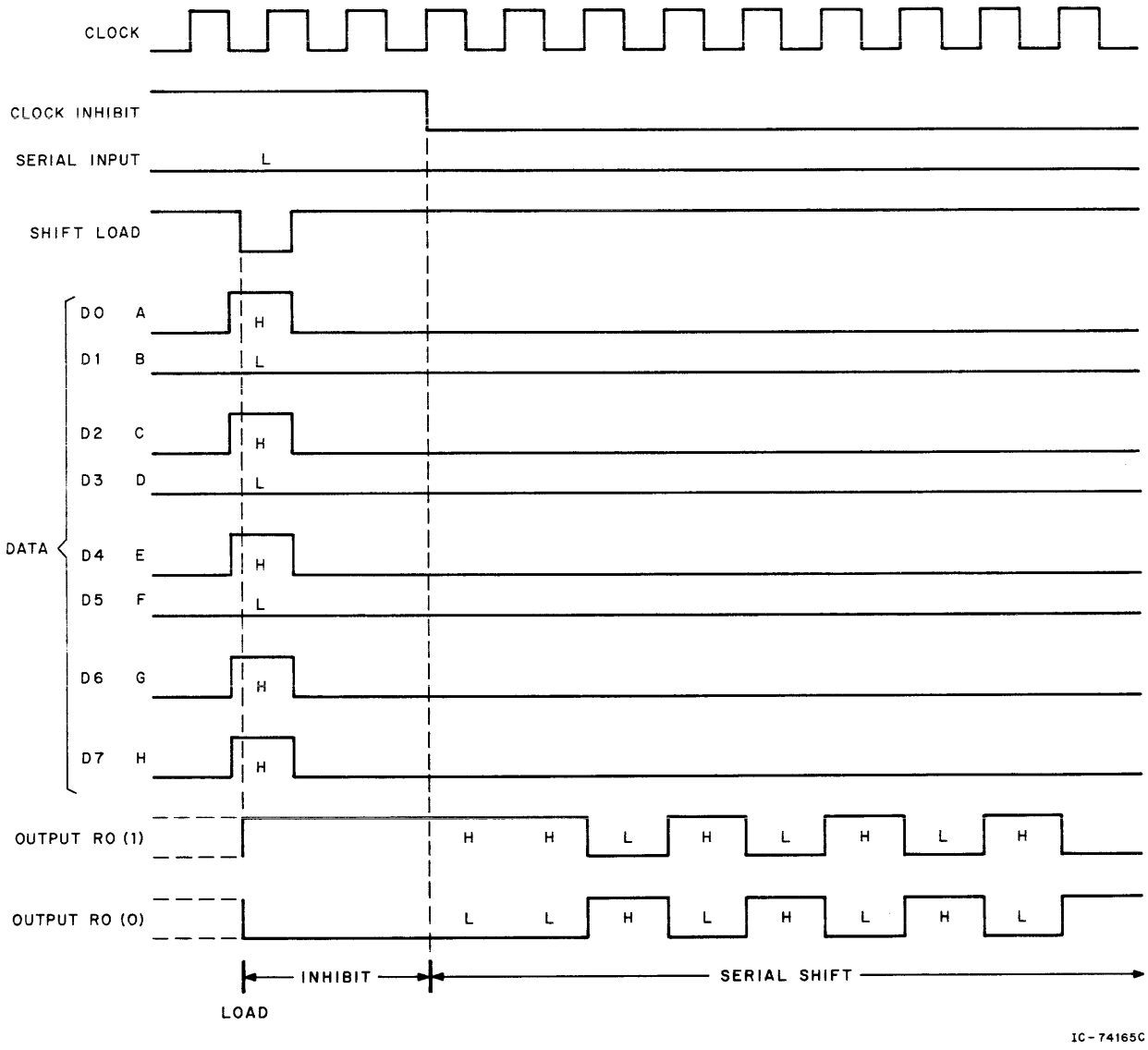

CLEAR
(9)
CLOCK
(8)
SERIAL
INPUTS
(1)
(2)
CLEAR
R
CK
S
$\bar{Q}_0$
Q_0
$\bar{Q}_1$
Q_1
$\bar{Q}_2$
Q_2
$\bar{Q}_3$
Q_3
$\bar{Q}_4$
Q_4
$\bar{Q}_5$
Q_5
$\bar{Q}_6$
Q_6
$\bar{Q}_7$
Q_7
(3)
(4)
(5)
(6)
(10)
(11)
(12)
(13)
OUTPUT R0(1)
OUTPUT R1(1)
OUTPUT R2(1)
OUTPUT R3(1)
OUTPUT R4(1)
OUTPUT R5(1)
OUTPUT R6(1)
OUTPUT R7(1)
IC-74164 C

74165 PARALLEL LOAD 8-BIT SHIFT REGISTER

Data is parallel-loaded into the 74165 on a negative-going transition of the load input (pin 1). This action is independent of the state of the clock, clock inhibit, or serial input. The register is clocked on the positive-going transition of the clock input (pin 2). The prerequisites for clocking are that the load input (pin 1) must be high and the clock inhibit input (pin 15) must be low.




PARALLEL INPUTS
DØ (11)
D1 (12)
D2 (13)
D3 (14)
PRESET
S
CK
R
CLEAR
SERIAL INPUT (10)
PARALLEL INPUTS
D4 (3)
D5 (4)
D6 (5)
D7 (6)
SHIFT/ LOAD (1)
CLOCK (2)
CLOCK INHIBIT (15)
(9) OUTPUT RØ (1)
(7) OUTPUT RØ (Ø)
IC-74165B


CLOCK
CLOCK INHIBIT
SERIAL INPUT
L
SHIFT LOAD
DATA
D0 A
H
D1 B
L
D2 C
H
D3 D
L
D4 E
H
D5 F
L
D6 G
H
D7 H
H
OUTPUT RO (1)
H H L H L H L H
OUTPUT RO (0)
L L H L H L H L
INHIBIT
SERIAL SHIFT
LOAD
IC-74165C

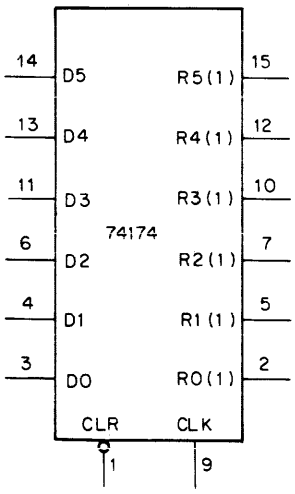
74174 HEX D FLIP-FLOP REGISTER

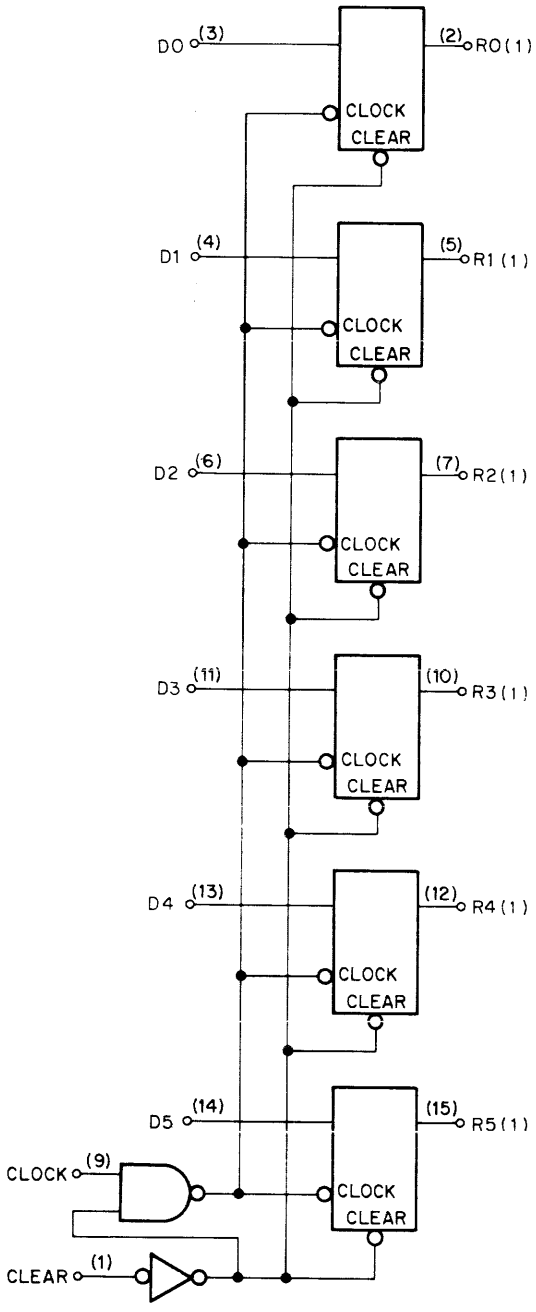
TRUTH TABLE

INPUT t_n D	OUTPUT t_{n+1} R(1)
H	H
L	L

t_n = Bit time before clock pulse.

t_{n+1} = Bit time after clock pulse.





Pin (16) = V_{CC}, Pin (8) = GND

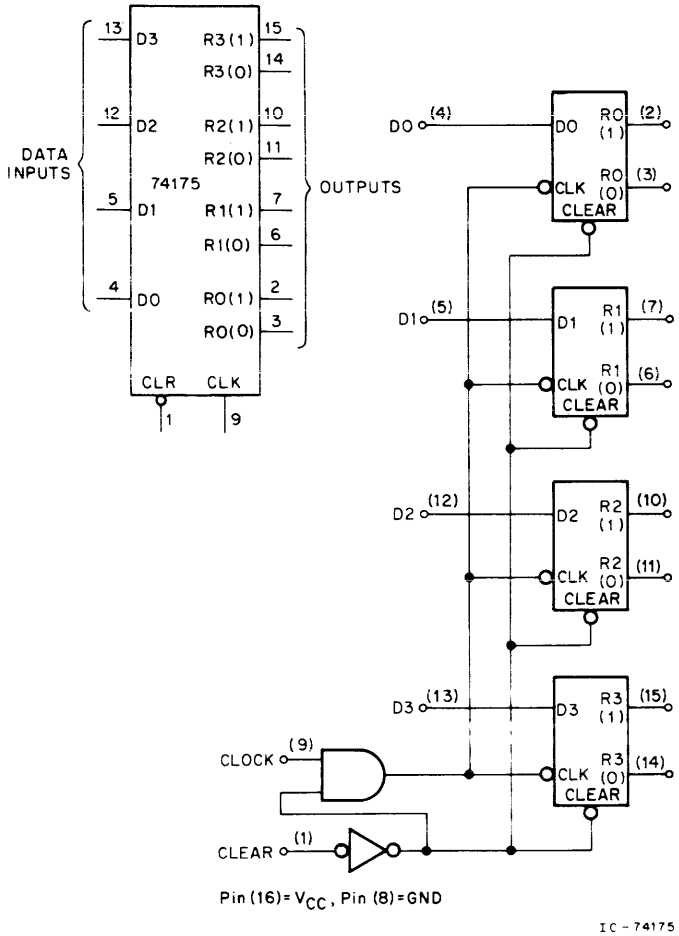
IC-74174

74175 Quad Storage Register

TRUTH TABLE

INPUT t_n	OUTPUTS t_{n+1}
D	R(1) R(0)
H	H L
L	L H

t_n = Bit time before clock pulse.
t_{n+1} = Bit time after clock pulse.



Pin (16) = V_{CC}, Pin (8) = GND

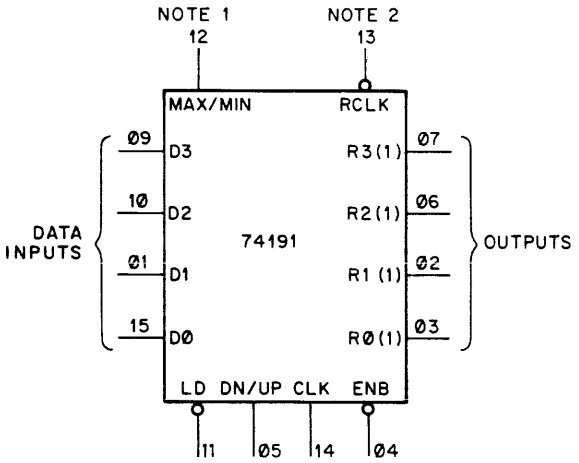
IC - 74175

74191 4 BIT UP/DOWN COUNTER

The 74191 is a 4-bit binary counter that counts in BCD or binary and can operate as an up or down counter. The counter can be preset by the load control and uses a ripple clock output for cascading.

DOWN/UP	ENABLE	LOAD	MODE
X	X	L	Parallel Load
X	H	H	No Change
L	L	H	Count Up
H	L	H	Count Down

H = high level L = low level X = irrelevant



VCC = PIN 16
GND = PIN 08

NOTES:

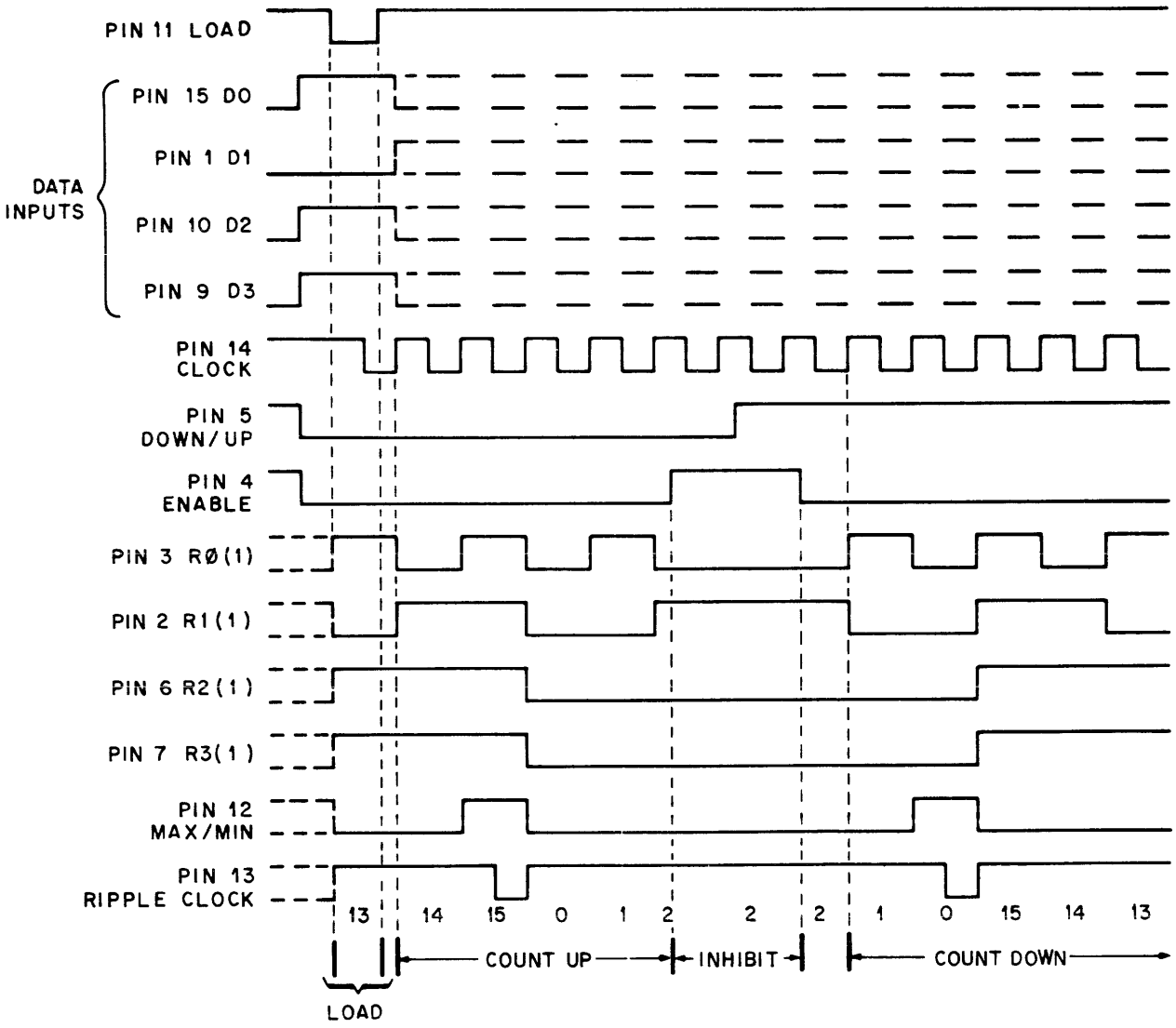
1. MAX/MIN produces a high level output pulse when the counter overflows or underflows.
2. Ripple clock produces a low level output pulse when an overflow or underflow condition exists.

IC-74191A

typical load, count, and inhibit sequence:

Illustrated below is the following sequence.

1. Load (preset) to binary thirteen.
2. Count up to fourteen, fifteen (maximum), zero, one and two.
3. Inhibit
4. Count down to one, zero (minimum), fifteen, fourteen, and thirteen.



IC-74191B

APPENDIX D
DUP11 OPTION DESIGNATIONS

D.1 INTRODUCTION

This appendix lists the option variations and cabinet kits available for the DUP11 Bit Synchronous Interface. The method for assigning DUP11 option designations is also described.

The communications option designations enable DIGITAL customers to obtain communication options that are tailored to their particular needs. FCC regulations require that all system cabinets manufactured after October 1, 1983 and intended for use in the United States, be designed to limit electromagnetic interference (EMI). Since both shielded and unshielded cabinets currently exist in the field, DIGITAL provides separate communication options for each cabinet type.

D.2 OPTION DESIGNATION CONVERSION

Most older DUP11 configurations are discontinued or changed to MAINTENANCE ONLY status. Therefore, the new option designations must be specified to obtain the necessary equipment. Table D-1 can be used to determine which communication option designations are necessary when designing or expanding upon a computer system.

Communication options may be ordered by customers either at the time a system is purchased (a factory-installed system option) or as an upgrade to an existing system (a field upgrade).

Table D-1 Option Compatibility Cross Reference

OLD OPTION	EQUIVALENT NEW OPTION		
	Field Upgrade		System Option
	Base Option	Cabinet Kit	
DUP11-DA	DUP11-M	CK-DUP11-A(*)[1]	DUP11-AP[2]

NOTES

1. **The last character of the cabinet kit (*) varies depending on which kit is required (refer to Table D-3).**

2. **The last character of the system option designation is always "P". This specifies that the option is to be factory installed.**

D.2.1 Factory-Installed System Options

A factory-installed system option is identified by a single option designation. When this designation is specified (see Table D-1), the appropriate module(s), cable(s), and I/O connector panel(s) are installed in the particular system being constructed.

D.2.2 Field Upgrade Options

A field upgrade is identified by two option designations. The two option designations are:

- A base option designation, and
- A cabinet kit designation.

Refer to Table D-1 to determine which new option designation to specify when additional replacement equipment is required.

D.2.2.1 Base Options – The base option designation specifies which component parts make up the base option. The component parts specified are:

- The electronic module(s),
- The turnaround test connector(s), and
- The option documentation.

D.2.2.2 Cabinet Kits – The cabinet kit designation specifies which component parts are included in the cabinet kit. The component parts specified are:

- The internal cable(s),
- The I/O connector panel(s), and
- An adaptor bracket (not always included) for installing the I/O connector panels in a non-FCC compliant (unshielded) cabinet.

NOTE
External cables needed to connect to a modem or other external device are usually not included.

D.3 OPTION CONFIGURATION SUMMARY

This section describes the method used to assign communication option designations.

Communication option designations ensure that the proper cable(s), I/O connector panel(s), and adaptor brackets (if necessary) are shipped with each base option.

Communication options may be obtained by customers either at the time of system purchase (a factory-installed system option) or as an upgrade to an existing system (a field upgrade).

The basic designations identify:

- System options (factory installed).
- Base options and cabinet kits (field upgrades).

System options are installed at the factory and are configured for the particular cabinet in which the option is being installed.

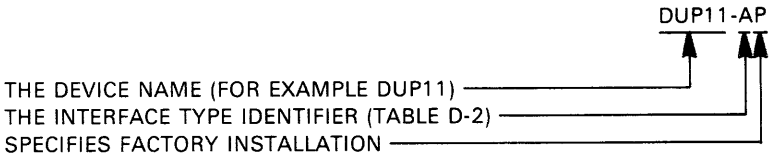
Base options and cabinet kits are ordered as upgrades to existing systems. A complete field upgrade option must include a base option and a cabinet kit.

NOTE

A field upgrade option alone does not make an unshielded cabinet FCC compliant. Shielded cabinets are specially constructed to limit EMI.

D.3.1 System Option Designations

System option designations provide the following information:



D.3.2 Base Option Designations

Base option designations provide the following information:

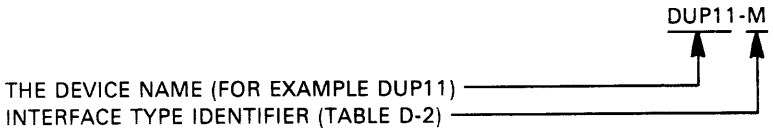


Table D-2 Electrical and Mechanical Interface Type

Identifier	Interface Type
A	RS-232-C (with full modem control)
M	Base option – Module, documentation, and test connector

D.3.3 Cabinet Kit Designations

Cabinet kit designations enable customers to obtain communication options that are tailored to their particular cabinet(s). Cable lengths, I/O connector panels, and method of installation may vary depending on the cabinet kit obtained.

Cabinet kits are individually tailored to specific cabinet types. This enables customers to install communication options in both shielded (FCC compliant) and unshielded (non-FCC compliant) cabinets.

Cabinet kits for *shielded* cabinets include:

- Internal cable(s).
- I/O connector panel.

The internal cable connects the module to the I/O connector panel that is installed in a shielded I/O bulkhead.

NOTE

Typically, cables required to connect to a modem or other external device are not supplied with most cabinet kits.

Cabinet kits for *unshielded* cabinets include:

- Internal cable(s).
- I/O connector panel.
- 74-27292-01 adaptor bracket (may also be included).

The internal cables connect the module to the I/O connector panel or may connect directly to the data communications equipment. When the I/O connector panel is provided, it should be installed in the 74-27292-01 adaptor bracket. When the I/O connector panel is not provided, the cable should be connected directly to the data communications equipment.

Cabinet kit designations provide the following information:

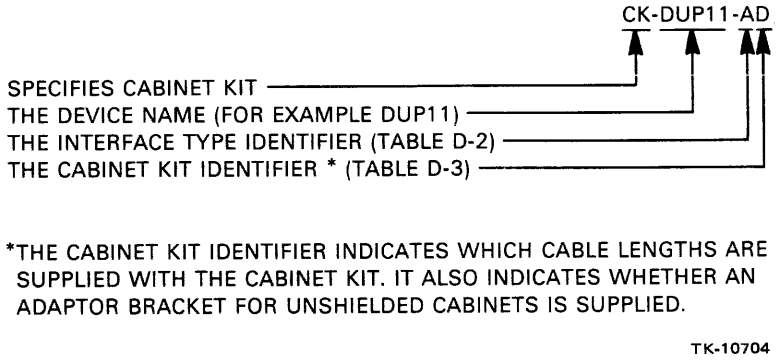


Table D-3 Cabinet Kit Components

Cabinet Identifier	Component Parts Supplied
Letters Indicate Shielded Cabinets	
D	• A 3.05 m (10 ft) BC08S internal cable • An H3001 I/O connector panel
E	• A 2.13 m (7 ft) BC08S internal cable • An H3001 I/O connector panel
Numbers Indicate Unshielded Cabinets	
1	• A 3.05 m (10 ft) BC08S internal cable • An H3001 I/O connector panel • A 74-27292-01 adaptor bracket **NOTE** **The 74-27292-01 adaptor bracket has space for three H3001 connector panels.**
3	• A 7.62 m (25 ft) BC08S internal cable (this cable connects directly to the data communications equipment).

D.4 DUP11 OPTION CONFIGURATIONS

Refer to Table D-4 for reference to the component parts of various DUP11 Bit Synchronous Interface configurations.

Table D-4 DUP11 Option Configurations

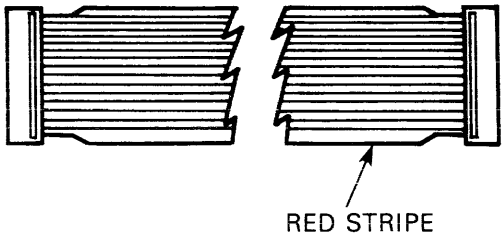
OPTION CONFIGURATIONS	M7876 Module	BC08S-10 Internal Cable (see Figure D-1)	BC08S-07 Internal Cable (see Figure D-1)	BC08S-25 Internal Cable (see Figure D-1)	H3001 I/O Connector Panel (see Figure D-2)	H325 Test Connector (see Figure D-3)	74-27292-01 Adaptor Bracket (see Figure D-4)	Option Documentation
FACTORY INSTALLED OPTIONS (see Note 1)								
• DUP11-AP	•	*	*		•	•		•
FIELD UPGRADE OPTIONS (see Notes 2 and 3)								
• DUP11-M (Requires one of the following four cabinet kits)	•					•		•
- CK-DUP11-AD		•			•			
- CK-DUP11-AE			•		•			
- CK-DUP11-A1		•			•		•	
- CK-DUP11-A3				•				

• Equipment supplied with option.

* Cables used depend on cabinet configuration.

1. Factory-installed options may only be obtained when the system is being originally configured.
2. Upgrades may be obtained for existing systems.
3. Upgrade options require a base option and cabinet kit.

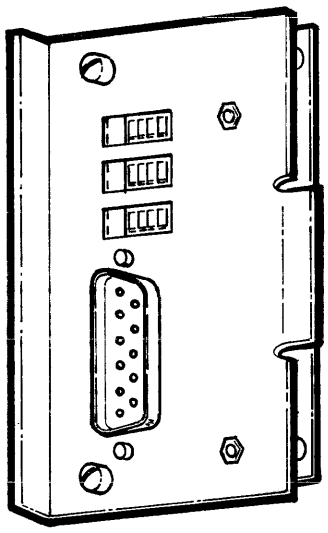
TK-10703



NOTE: LENGTH VARIATION (**) DEPENDS ON CABINET KIT OBTAINED

TK-10706

Figure D-1 BC08S Internal Cable



TK-10719

Figure D-2 H3001 I/O Connector Panel

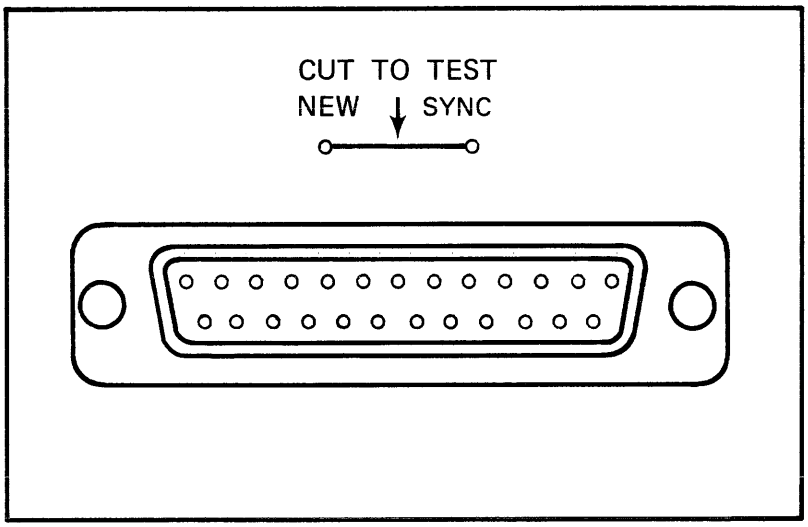


Figure D-3 H325 Test Connector

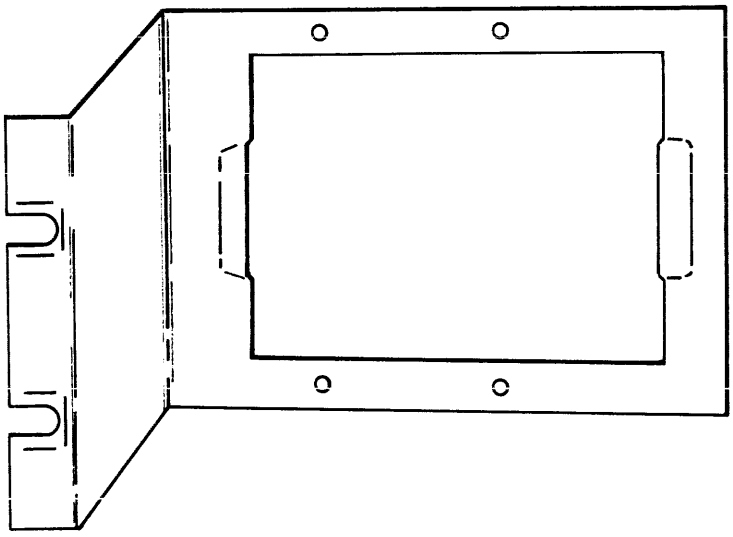


Figure D-4 74-27292-01 Adaptor Bracket

DUP11 Bit Synchronous Interface
Maintenance Manual
EK-DUP11-MM-003
(MK)

Reader's Comments

Your comments and suggestions will help us in our continuous effort to improve the quality and usefulness of our publications.

What is your general reaction to this manual? In your judgement is it complete, accurate, well organized, well written, etc? Is it easy to use? _____

What features are most useful? _____

What faults or errors have you found in the manual? _____

Does this manual satisfy the need you think it was intended to satisfy? _____

Does it satisfy *your* needs? _____ Why? _____

☐ Please send me the current copy of the *Technical Documentation Catalog*, which contains information on the remainder of DIGITAL's technical documentation.

Name _____ Street _____
Title _____ City _____
Company _____ State/Country _____
Department _____ Zip _____

Additional copies of this document are available from:

Digital Equipment Corporation
146 Main St.
Maynard, MA 01754

Attention: Printing and Circulation Services (NR2/M15)
Customer Services Section

Order No. **EK-DUP11-MM-003** _____

Fold Here

Do Not Tear – Fold Here and Staple

digital

No Postage
Necessary
if Mailed in the
United States

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 33 MERRIMACK, NH

POSTAGE WILL BE PAID BY ADDRESSEE

Digital Equipment Corporation
Educational Services/Quality Assurance
12 Crosby Drive, BU/E08
Bedford, MA 01730

Digital Equipment Corporation • Bedford, MA 01730