pdp11

DU11 single line programmable synchronous interface user's manual

digital

EK-DU11-OP-001

DU11 single line programmable synchronous interface user's manual

digital equipment corporation • maynard, massachusetts

1st Edition, November 1976

The material in this manual is for informational purposes and is subject to change without notice.

Digital Equipment Corporation assumes no responsibility for any errors which may appear in this manual.

Printed in U.S.A.

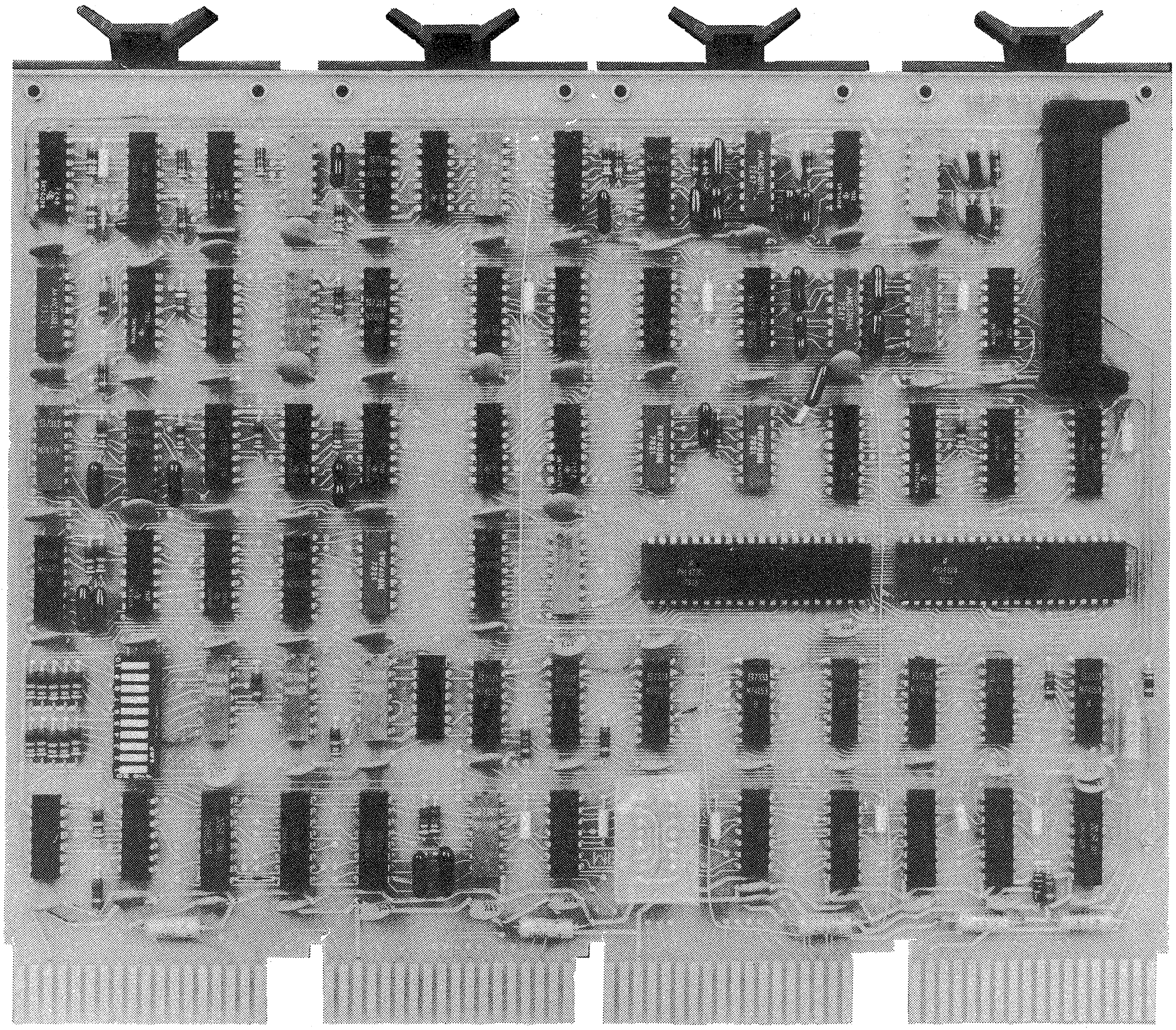
This document was set on DIGITAL's DECset-8000 computerized typesetting system.

CONTENTS

CONTENTS (Cont)

ILLUSTRATIONS

TABLES



6731-1

DU11 Programmable Synchronous Interface

CHAPTER 1
INTRODUCTION

1.1 SCOPE

This manual provides a complete description of the DU11 Line Interface, including installation and programming. The level of discussion assumes that the reader is familiar with basic digital computer theory.

This chapter contains introductory information. It includes a description of data communication techniques and systems, a general description of the DU11, a physical description of the DU11, DU11 specifications, and an explanation of engineering drawing conventions.

1.2 DATA COMMUNICATION TECHNIQUES AND SYSTEMS

1.2.1 Data Communication Techniques

There are several techniques used for the transfer of data communication signals. Each has its particular advantages and disadvantages.

1.2.1.1 Pulse Coding – Standard data communication messages are sent in some form of pulse code. There are several varieties of pulsed codes used in the transferral of data in digital form. Binary signals, by their very nature, are natural elements for digital data codes. Such codes are said to be in "binary format."

A formatted binary code can represent different symbols only by allowing sufficient binary elements for each symbol. If we think of one binary digit (or "bit") representing each symbol, we have only two choices: one symbol represented by the "on" state, the other represented by the "off" state. With such an arrangement, we could let the "on" or one state represent "no" and the "off" or zero state represent "yes." While it would be difficult with such an arrangement, we could convey messages of a very limited nature from a remote station (such as the answer to "Is the temperature at your station over 70° F?").

If, instead of using one binary digit for our character, we use two, we have more characters to choose from. Our choice for a one-bit code was limited to two: 0 or 1. Our choice for a two-bit code is four: 00, 01, 10, or 11. If we choose a three-bit code, our choice is eight: 000, 001, 010, 011, 100, 101, 110, and 111. It can be shown that for a code with a character makeup of n bits, the number of characters available will be 2^n. In communications parlance, instead of calling these codes one-bit codes, two-bit codes, etc., they are called one-level codes, two-level codes, etc. Although any arbitrary meaning can be assigned to a code character, it is more practical for the majority of operations to let the characters represent numbers, punctuation marks, spaces, and letters of the alphabet. In addition to these, some special codes use characters for other meanings.

1.2.1.2 Pulse Code Transmission – In order to transmit code characters, it is necessary to arrange their elements in a way that will allow their reception without uncertainty. There are several techniques by which this may be done; these techniques fall into two broad categories: serial data transmission and parallel data transmission.

Because the DU11 is a serial communication interface, only serial data transmission techniques will be discussed.

There are two basic techniques of serial data transmission: asynchronous and synchronous. These two techniques as well as a third, isochronous, will be discussed in the following paragraphs.

1.2.1.2.1 Asynchronous Serial Transmission – This technique enables data to be transferred as it becomes available. This is possible by framing each data character with a begin signal (START bit) and an end signal (STOP bit), so that the equipment receiving the data (the interface receiver) knows when a data character is being presented on the communication line and when the line is inactive.

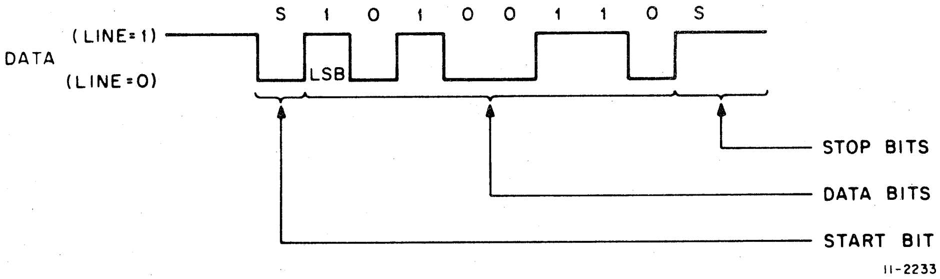


Figure 1-1 Asynchronous Technique Format

Hence, each character consists of three parts: a START bit, the data bits, and a STOP bit (Figure 1-1). A START bit is a line state (usually a zero) that lasts for 1 bit time. The data bits represent the actual binary character being transferred. In many applications the characters are 8 bits long with the least significant bit being sent out and received first. A STOP bit is a line state (usually a one) that lasts for 1, 1.42, or 2 bit times; it indicates that character transmission is complete. The STOP bit enables the interface receiver to check synchronization after each character transmission. If the STOP bit is not received properly, i.e., it is not presented on the line immediately after the last data bit, the character received is considered erroneous and re-transmission is necessary.

Clocking for the interface transmitter and interface receiver during asynchronous transmission is provided by two different sources that are asynchronous to one another. The transmitter clock is enabled when data is available for transmission and clocks the character onto the line. The receiver clock is enabled when a START bit is detected on the line and samples the data bits as they are presented on the line. The receiver is also equipped with a counter that counts the character bits received. When a complete character and a STOP bit are received (the receiver must know the number of bits per character), the receiver clock is disabled until the next START bit is detected.

The asynchronous serial data transmission technique has the following advantages:

a. Can be generated easily by electromechanical equipment (e.g., Teletype® keyboard).

b. Can be used easily to drive mechanical equipment (e.g., Teletype printer).

c. Characters can be sent asynchronously (as they become available) because each character has its own synchronizing information.

The disadvantages of the asynchronous serial data transmission technique are:

a. Separate timing required for both transmitter and receiver.

b. Distortion sensitive because the receiver depends on incoming signal sequences to become synchronized. Any distortion in these sequences will affect the reliability with which the character is assembled.

c. Speed limited because a reasonable amount of margin between characters must be built in to accommodate distortion.

d. Inefficient because at least 10 bit times are required to send 8 bits of data. If a 2 bit time STOP bit is used, it takes 11 bit times to transfer 8 bits of data.

1.2.1.2.2 Synchronous Serial Transmission – This technique does not use START and STOP bits to accomplish synchronization. Instead, the entire block of data (message) is preceded on the line by a synchronizing code. When the interface receiver recognizes this code (henceforth referred to as sync characters), it locks in and, using a counter, assembles the data characters which follow. Hence, as in the asynchronous technique, the receiver must know the number of bits per character.

This technique requires that the clocking for the interface transmitter and interface receiver be provided by a common clock source. The clock signal is provided to the transmitter and receiver on lines separate from the data line. At the transmitter, the clock signal serves to clock the data onto the line. At the receiver, the clock signal gates the data in. Figure 1-2 illustrates the timing for a synchronous communication system using modems.

®Teletype is a registered trademark of Teletype Corporation.

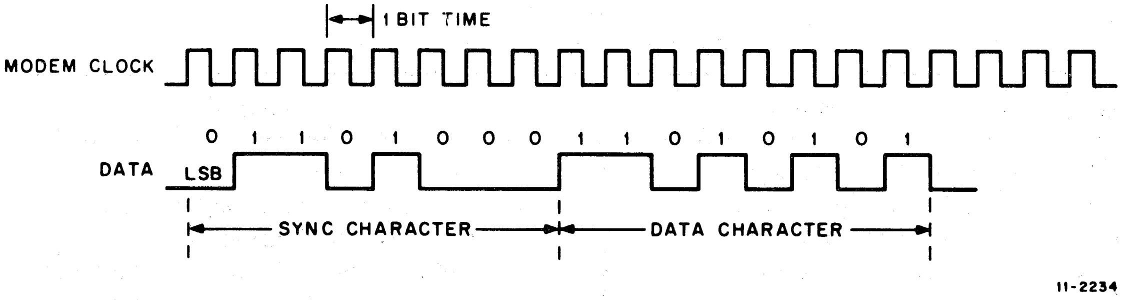


Figure 1-2 Synchronous Format

As shown in Figure 1-2, the modem provides the clock, the transmitter presents the data to the line on the positive going edge of the clock and the receiver samples the data on the negative going edge. If the transmitter pauses at any time and fails to inhibit the clock, the receiver will continue to sample the line, synchronization is lost and the remainder of the message will be erroneous.

The advantages of the synchronous serial data transmission technique are:

a. Modem timing sources can be used for both transmitter and receiver.

b. Interface receiver does not require clock-synchronizing logic as the asynchronous technique does.

c. Highly efficient because there are no bit times wasted with the use of START and STOP bits. All bits on the line are data, with the exception of the sync characters at the beginning of the bit stream.

d. Low distortion sensitivity because the timing is provided along with the data.

e. Higher speeds are achievable because of the low distortion sensitivity.

The disadvantages of the synchronous serial data transmission technique are:

a. Characters must be sent synchronously, not asynchronously (asynchronous transmission is desirable for most real time and mechanical applications).

b. One bit time added to or missing from the data-bit stream can cause the entire message to be faulty.

c. The common-carrier equipment required to accommodate this mode of operation is more expensive than the equipment required for asynchronous modes of operation.

d. Mechanical equipment cannot transmit or receive this format directly.

1.2.1.2.3 Isochronous Serial Transmission – This technique is essentially the transmission of asynchronous data over a synchronous modem. Character synchronization is achieved via START and STOP bits; a common timing source is used for both the transmitter and receiver.

The isochronous technique does have advantages over the asynchronous technique. Clocking for isochronous operations emanates from the modems and is synchronous to the data; hence, the receiver does not require clock-synchronizing logic and distortion sensitivity is low making higher speeds possible.

1.2.2 Data Communication Systems

1.2.2.1 Synchronous Systems – Synchronous modulator-demodulators (modems) have permitted a higher rate of data transmission than asynchronous modems over a voice grade facility. The nature of these transmission techniques has also resulted in higher efficiency by eliminating the need for synchronizing information with every character.

The logic design of interfaces to a synchronous modem is considerably easier than the design of an asynchronous interface because there is no need for bit synchronization and sampling hardware. Most synchronous modems supply all the timing necessary to receive each bit as it is made available from the modem. The difficulty in designing a synchronous modem interface is to design the capability of communicating in the message formats used in synchronous communications.

Table 1-1
Representative Message Codes

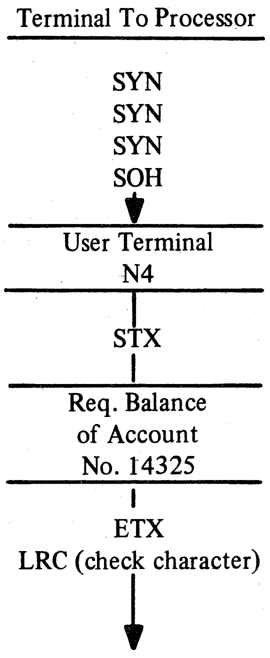
Character	Meaning	Function
SYN	Synchronizing signal	Establish character framing
SOH	Start of heading signal	Precedes block message heading characters
STX	Start of text signal	Precedes block of text characters
ETX	End of text signal	Terminates a block of characters started with STX
ACK	Acknowledge signal	*Affirmative acknowledgment of message received
NAK	Negative acknowledge signal	*Negative acknowledgment of message received

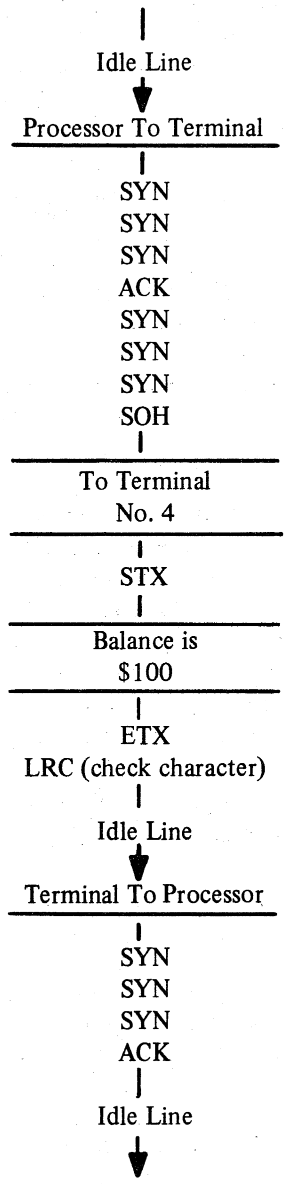
*ACK and NAK are sent by the station that received the message to the station that originated the message.

It is not the purpose of this manual to discuss the format for synchronous communication in detail. However, a brief description of these formats is outlined below to facilitate the reader's understanding of synchronous interface design.

Because the synchronous transmission technique provides only bit recovery timing, there must be a way to establish character framing and message framing. This is accomplished by using codes (usually ASCII) that are assigned for synchronous message formatting purposes. Representative message codes are listed in Table 1-1.

A typical message that might be sent between two devices (a terminal and a processor) follows.





1.2.2.2 **Computer Application** – Electronic computers are often connected into communication systems to help transmit and process digital data. By using computer systems to concentrate data from many low-speed terminals over one voice grade facility, significant improvements can be made in the efficiency of a data communication system. Since most long-range communication systems are connected through common carrier facilities, a communication system using a computer should be interfaced to the correct type facility. There are two basic types of common carrier facilities to which computers must be interfaced: asynchronous serial and synchronous serial. We have already pointed out the advantages and disadvantages of these two types of facilities. Based on these advantages and disadvantages, Table 1-2 shows typical speeds and applications of these two techniques.

As shown in Table 1-2, there are three basic communication applications to be solved by the computer communications engineer:

Low speed terminal equipment, such as Teletypes.
Medium speed terminal equipment.
Intercomputer communications.

1.3 GENERAL DESCRIPTION

The DU11 interface is a single line, program controlled, double-buffered communication interface. It provides serial to parallel and parallel to serial data conversion, EIA* to TTL (transistor-transistor logic) and TTL to EIA voltage level conversion and modem control for full or half duplex communication systems.

The DU11 is compatible with all PDP-11 family computers and is available in two models. Model DU11-DA is the basic version and is completely contained on the M7822 module. The basic version is compatible with the Bell 201 synchronous modem or equivalent. Model DU11-EA is simply the basic version adapted to current mode operation. The DU11-EA version consists of the basic M7822 module plus the DF11-G current mode converter. The DU11-EA is compatible with the Bell 303, wide band, synchronous modem or equivalent. A typical communication system using the DU11 is shown in Figure 1-3.

Interface operation is completely program controlled. The mode of operation (synchronous or isochronous), character length (5, 6, 7, or 8 bits plus parity if selected), parity enable and sense (odd or even), sync character configuration, and duplex mode (full or half) are all selected via the program.

1.4 PHYSICAL DESCRIPTION

The DU11 interface is completely contained on a single M7822 Quad Integrated Circuit module (Figure 1-4). This module can be mounted easily in the PDP-11 processor small peripheral controller slot (exceptions noted in Chapter 2) or in one of four slots in a DD11-A or DD11-B peripheral mounting panel.

All DU11 operating power is provided by the mounting panel in which it is installed. The power is taken from the mounting box power supply. For proper operation, the module requires +5 V @ 2.2 A, -15 V @ 0.17 A, and +15 V @ 0.07 A.

The mounting panel also connects the DU11 to the Unibus. All Unibus input/output signals enter and leave the module via the mounting panel pins. Refer to Chapter 2 for Unibus to mounting panel connection.

Table 1-2
Computer Communications Applications

Speed	Asynchronous	Synchronous
Low 0 to 300 baud	Electromechanical terminals such as keyboard printers and Teletypes.	Operations tend to be asynchronous at these speeds.
Medium 300 to 3000 baud	Unbuffered terminals such as paper tape readers and punches, card readers and line printers.	Buffered terminals such as displays, buffered card readers, and line printer configurations.
High 5000 baud and up	Not frequently used.	Intercomputer communications.

*EIA – A standardized set of signal characteristics (time duration, voltage, and current) specified by the Electronic Industries Association.

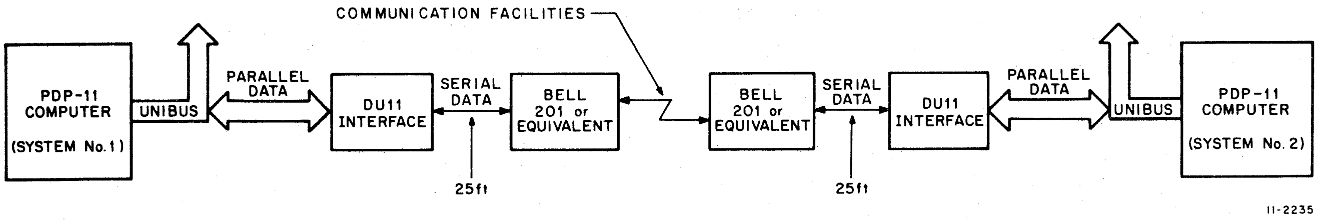


Figure 1-3 Typical Communication System Using the DU11 Interface

Major DU11 components are also labeled on Figure 1-4: the rocker switches which are used to select the interface Unibus address, the priority plug which determines the bus request (BR) priority level of the interface (BR5 plug normally installed at factory), the SAR (receiver) and SAT (transmitter) chips, and jumpers W2, W4–W6, and W9–W16 (a complete description of the jumpers is provided in Paragraph 2.1.4).

1.5 SPECIFICATIONS

Environmental, electrical, and performance specifications for the DU11 are contained in the following paragraphs.

1.5.1 Environmental

Ambient temperature
10° to 50° C (50° to 122° F)

Relative humidity
20% to 95% (without condensation)

1.5.2 Electrical

DC voltage requirements
+5 V @ 2.2 A
- 15 V @ 0.17 A
+15 V @ 0.07 A

Electrical Characteristics

Electrical characteristics of this interface meet EIA standard RS-232C and PDP-11 Unibus Interface specifications.

1.5.3 Performance

The following paragraphs discuss the baud rate limitations of the DU11 and related program response time.

1.5.3.1 Baud Rates for Synchronous Communications – EIA/CCITT* baud rate (10K baud maximum) is limited by modem and data set interface level converters. Current mode operation (100K baud maximum) is possible only with the DU11-EA. Current mode speed is limited by DU11 logic.

Even though the DU11 can receive and transmit information at such a high rate, it may, in most cases, be impractical. Since the service of the data buffers relies solely on the program, little time if any would be left for other events. This problem would be compounded if the interface were operating in full duplex mode.

1.5.3.2 Baud Rates for Isochronous Communications – EIA/CCITT baud rate (10K baud maximum) is limited by data set interface level converters. Current mode operation baud rate (100K baud maximum) is limited by DU11 logic.

1.6 ENGINEERING DRAWINGS

A complete set of engineering drawings entitled DU11 Line Interface, Engineering Drawings is provided with each interface. The general logic symbols used on these drawings are described in the *DEC Logic Handbook,* 1972. Specific symbols and conventions are also included in the PDP-11 system manuals. The following paragraphs describe the signal nomenclature conventions used in the drawing set.

1.6.1 Basic Signal Names

Signal names in the DU11 print set are in the following basic form:

SOURCE SIGNAL NAME POLARITY

SOURCE indicates the drawing number of the print from which the signal originates. The drawing number of a print is located in the lower right-hand corner of the print title block (D1, D2, D3, D4, D5, and D6).

*CCITT – The Consultive Committee International Telegraph and Telephone is an advisory committee established under the United Nations to recommend worldwide standards.

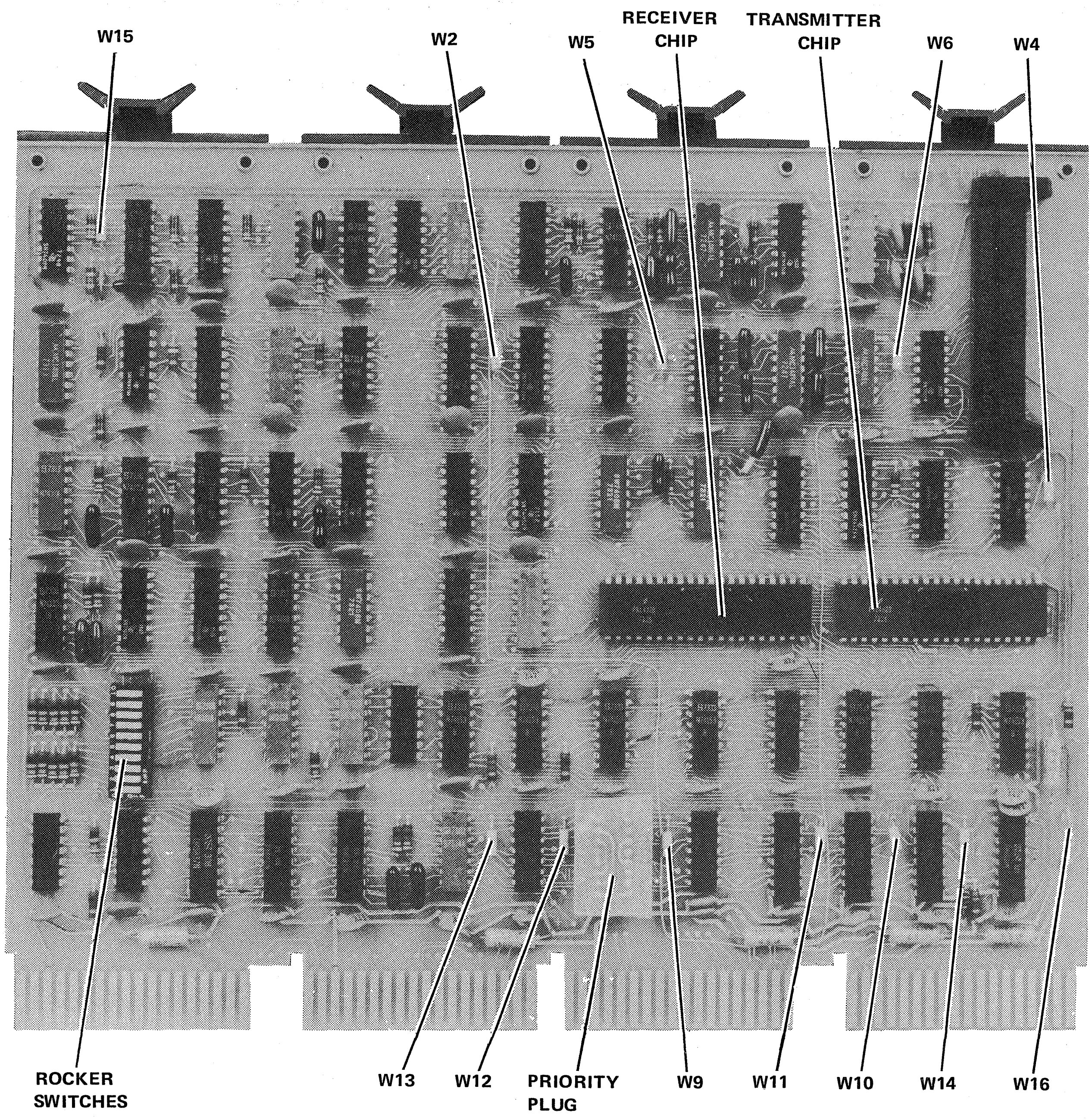


Figure 1-4 DU11 Major Components

SIGNAL NAME is the proper name of the signal. The names used on the print set are also used in this manual for correlation between the two.

POLARITY is either H or L to indicate the voltage level of the signal: H means +3 V; L means ground.

For example, the signal

D5- TX DONE H

originates on sheet 5 of the engineering drawings and is read, "When TX DONE is true, this signal is at +3 V."

Unibus signal lines do not carry a SOURCE indicator. These signal names represent a bidirectional wire-ORed bus; as a result, multiple sources for a particular bus signal exist. Each Unibus signal name is prefixed with the word BUS.

Interface signals fed to or received from the Bell 201 modem via the Berg connector on the M7822 module are preceded by the jack and pin number in parentheses:

(J1-DD) EIA DATA TERM RDY (This signal is shown on engineering drawing D6.)

Interface signals fed to or received from the Bell 303 modem via the mounting panel backpanel wiring and DF11-G level converter are preceded by the M7822 module pin number:

AF1 D6-DTR (1) H

1.6.2 Flip-Flop Signal Names

Flip-flop signal names add an extra dimension. Although flip-flops have only two outputs, four signal names are possible (Figure 1-5). The two real outputs are RX DONE (1) H on pin 5 and RX DONE (0) H on pin 6. The two additional outputs are simply the real outputs reidentified. RX DONE (1) L is electrically the same as RX DONE (0) H, and RX DONE (0) L is electrically the same as RX DONE (1) H.

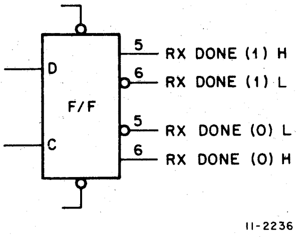


Figure 1-5 Flip-Flop Signal Names

CHAPTER 2
INSTALLATION

2.1 INSTALLATION

2.1.1 Mounting the DU11 in the Computer

There are two DU11 installation configurations:

a. The standard configuration, in which the DU11-DA interfaces with the Bell 201 synchronous modem or equivalent.

b. The current mode configuration, in which the DU11-EA interfaces with the Bell wide band 303 modem or equivalent.

2.1.1.1 Standard Configuration – In this configuration, the DU11-DA can be mounted in the small peripheral controller slot in the PDP-11/05, 10, 35, 40, 45, and 50 processors or in any one of four slots in the DD11-A or DD11-B peripheral mounting panels (Figure 2-1). The DD11-A mounting panel (Figure 2-2) is used in the PDP-11/15 and 20 computers, while the DD11-B (Figure 2-3) is used in the PDP-11/05, 10, 35, 40, 45, and 50 computers.

NOTE

The DU11-DA cannot be mounted in the small peripheral controller slot in the PDP-11/15 and 20 processors.

DD11-A and DD11-B mounting requirements are somewhat different. When using the DD11-B mounting panel, the DU11 is simply installed in the mounting panel; however, when using the DD11-A, jumper W16 (engineering drawing D1) and module G8000 must also be installed. Jumper W16 bypasses a voltage dropping resistor and the G8000 module converts the full-wave rectified +8 V/rms mounting panel input signal to a positive dc voltage which is used to drive the EIA level converters. To install the G8000 module, proceed as follows:

1. Install the G8000 module in slot A02 of the DD11-A.
2. Connect a wire between A03V2 and A02V2.
3. Connect a wire between A02N2 and CXXU1, where XX is to slot location of the M7822 module.

NOTE

Jumper W16 must not be installed if the DU11 is being installed in the DD11-B mounting panel.

2.1.1.2 Current Mode Configuration – In this configuration, the DU11-EA must be installed in the DD11-B mounting panel as shown in Figures 2-4 and 2-5.

NOTE

The DD11-A mounting panel cannot be used in the current mode configuration because it will not accommodate the DF11-G level converter.

2.1.2 Installing the Modem Cable Harness

A different cable is required to connect the DU11 to the Bell 201 modem than to the Bell 303 modem. The BC05C-25 cable harness (Figure 2-6) is used for the DU11-DA configuration; the BC01W-25 cable harness (Figure 2-7) is used for the DU11-EA configuration.

To install the cables, refer to Figure 2-8 and proceed as follows:

1. Position the Berg connector such that the connector name and pin number markings are visible and mate it fully and squarely with the Berg connector on the DU11 module (DU11-DA configuration) or the DF11-G connector module (DU11-EA configuration).

2. Align the Cinch connector (DU11-DA configuration) or the Burndy connector (DU11-EA configuration) to the receptacle located on the rear of the modem.

3. Mate the connector and tighten the two hold-down screws using a screwdriver.

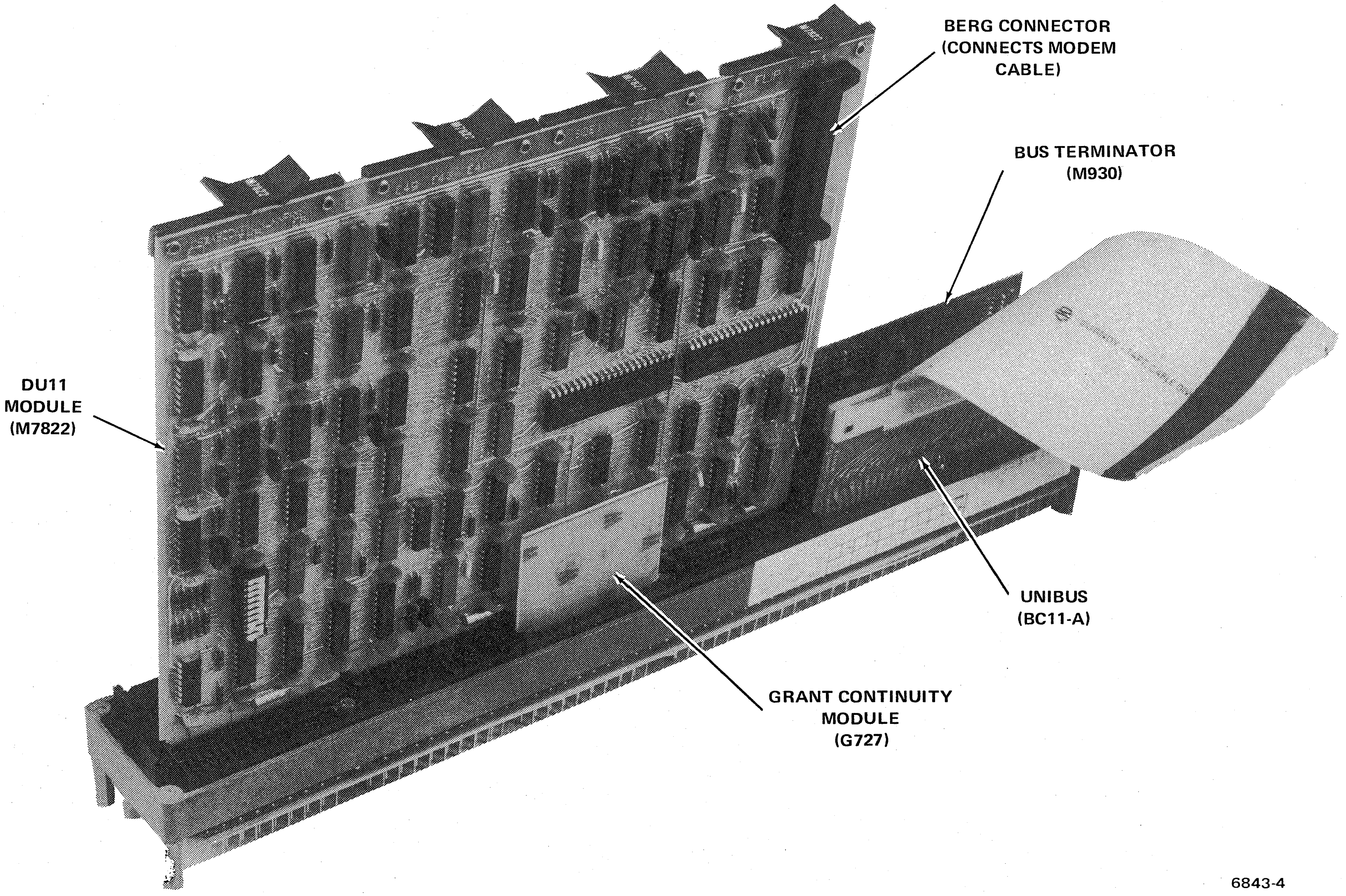
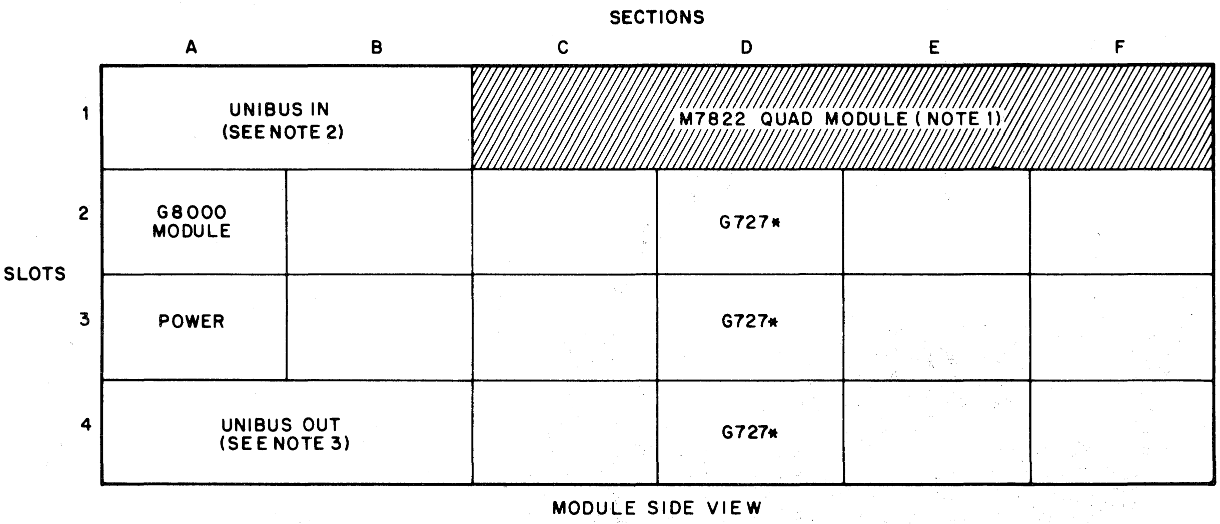


Figure 2-1 Standard Configuration (DU11-DA) Using DD11-B Mounting Panel



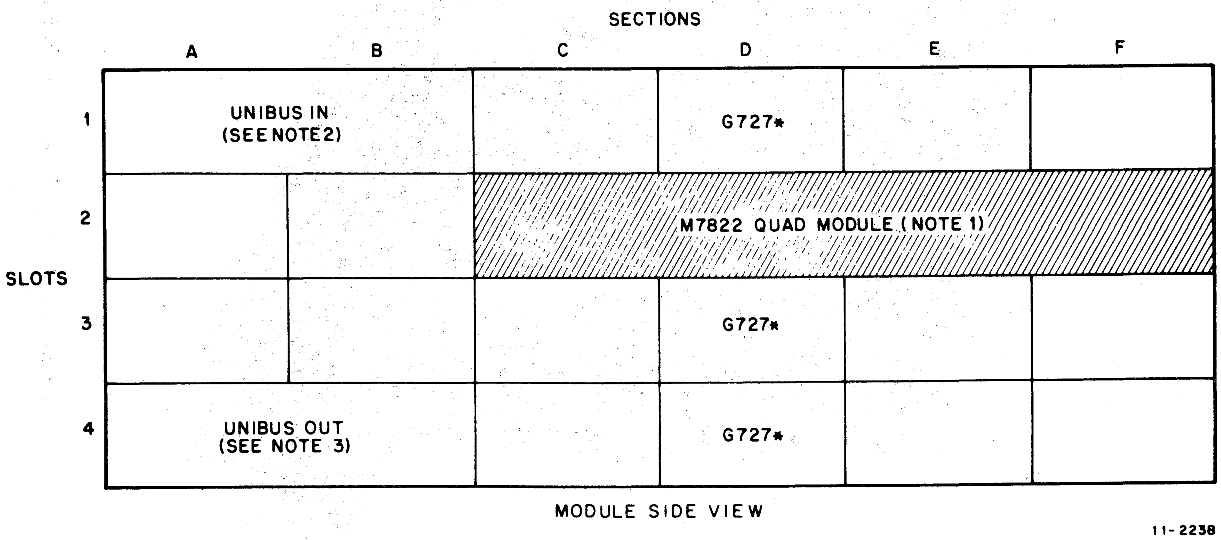
SLOTS	A	B	C	D	E	F
1	UNIBUS IN (SEE NOTE 2)		M7822 QUAD MODULE (NOTE 1)			
2	G8000 MODULE			G727*		
3	POWER			G727*		
4	UNIBUS OUT (SEE NOTE 3)			G727*		

***GRANT Continuity Module (G727) must be installed in each slot that does not receive an interface logic module.**

NOTES:

1. **Can be mounted in slots 1, 2, 3, or 4**
2. **Can be M920 or BC11-A**
3. **Can be M920, BC11-A or M930**

Figure 2-2 DU11-DA (M7822 Module) Mounted in DD11-A



SLOTS	A	B	C	D	E	F
1	UNIBUS IN (SEE NOTE2)			G727*		
2			M7822 QUAD MODULE (NOTE 1)			
3				G727*		
4	UNIBUS OUT (SEE NOTE 3)			G727*		

***Grant Continuity Module (G727) must be installed in each slot that does not receive an interface logic module.**

NOTES:

1. **Can be mounted in slots 1, 2, 3, or 4**
2. **Can be M920 or BC11-A**
3. **Can be M920, BC11-A or M930**

Figure 2-3 DU11-DA (M7822 Module) Mounted in DD11-B

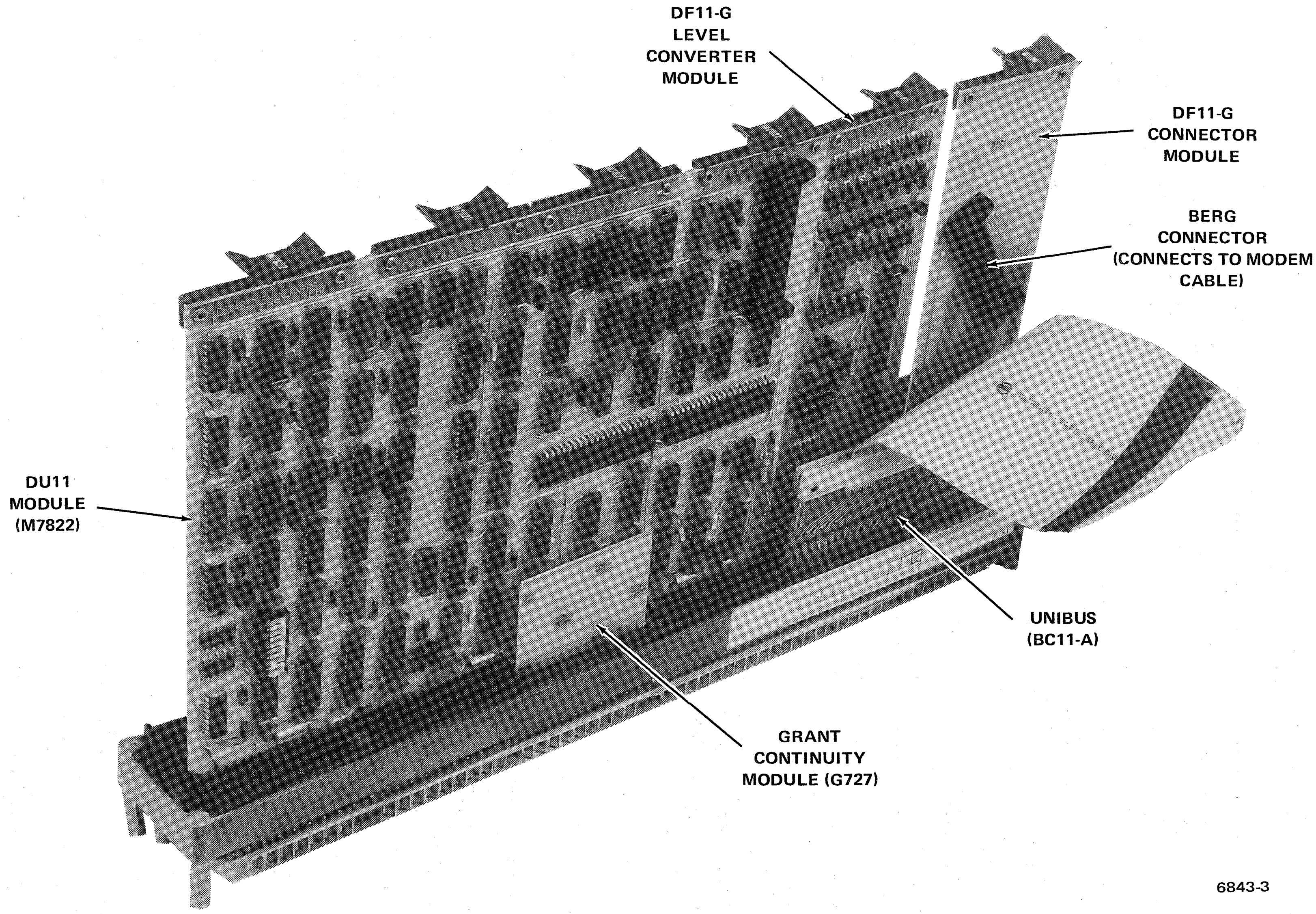
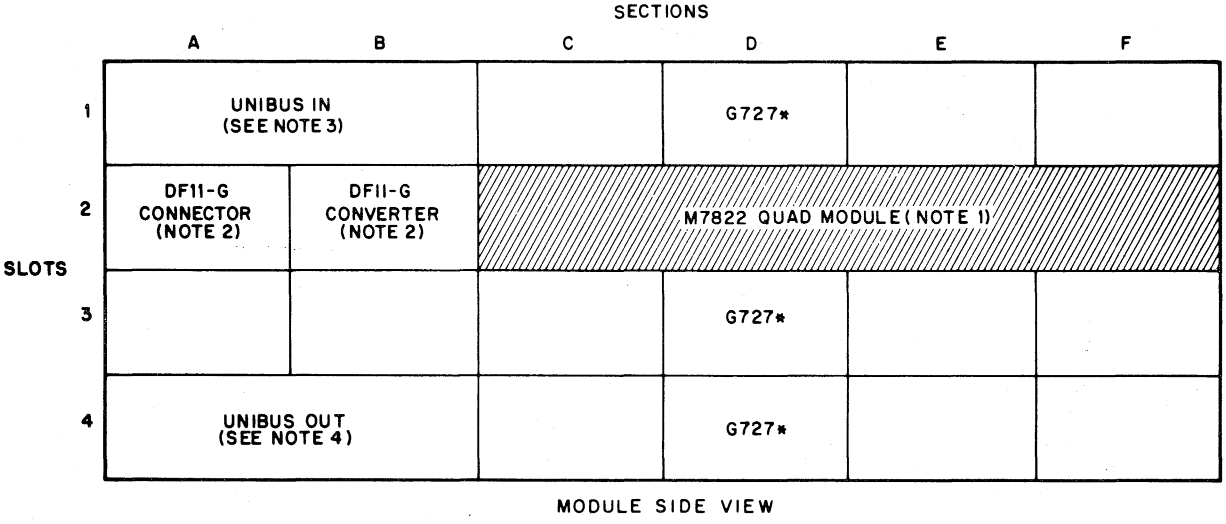


Figure 2-4 Current Mode Configuration (DU11-EA) Using DD11-B Mounting Panel



SLOTS	SECTIONS A	B	C	D	E	F
1	UNIBUS IN (SEE NOTE 3)			G727*		
2	DF11-G CONNECTOR (NOTE 2)	DFII-G CONVERTER (NOTE 2)	M7822 QUAD MODULE(NOTE 1)			
3				G727*		
4	UNIBUS OUT (SEE NOTE 4)			G727*		

MODULE SIDE VIEW

11-2239

***Grant Continuity Module (G727) must be installed in each slot that does not receive an interface logic module.**

NOTES:

1. **Can only be mounted in slots 2 or 3**
2. **The DF11-G connector and converter must be mounted in the same slot as the M7822 module.**
3. **Can be M920 or BC11-A**
4. **Can be M920, BC11-A or M930**

Figure 2-5 DU11-EA (M7822 Module and DF11-G Converter) Mounted in DD11-B

2.1.3 Unibus and Interrupt Vector Address Assignments

The Unibus and interrupt vector addresses must be determined prior to operating the DU11. The Unibus address is switch selectable; the interrupt vector addresses are jumper selectable (Figure 1-4 for physical location).

The Unibus address (also referred to as the device address) is controlled by ten rocker switches located in the address selection and mode control logic. The position of these switches determines the required address state (0 or 1) of bus address bits 12–03. If a rocker switch is set to ON, the switch contacts are closed and an address state of 0 is required on the related address bit to address the DU11. Hence, electrically the DU11 can have any device address within the range of 760000 to 777777; however, Digital Equipment Corporation software requires that the device address fall within the floating address range of 760010 to 763776. Refer to Appendix B for a complete discussion of DU11 address assignments.

NOTE

If a device address is selected which falls outside the floating address range, the software must be modified accordingly.

The interrupt vector addresses are also floating and are established at the factory in accordance with the vector addressing scheme described in Appendix B. If it is necessary to change the vector address, simply change jumpers W9–W14 as required. Jumpers are cut to obtain a logical zero. Jumpers W9–W14 are located in the interrupt control logic (engineering drawing D4). These jumpers control vector address bits 08–03; hence, vector addresses can be generated within the range of 000 to 774; however, software requires that the vector address fall within the floating address range of 300 to 777.

NOTE

If a vector address is selected which falls outside the floating address range, the software must be modified accordingly.

2.1.4 Jumper Assignments

Jumpers are used at various points in the DU11 circuitry to increase flexibility and to meet the floating vector address requirement described in Appendix B. For a complete description of the DU11 jumpers, refer to Table 2-1.

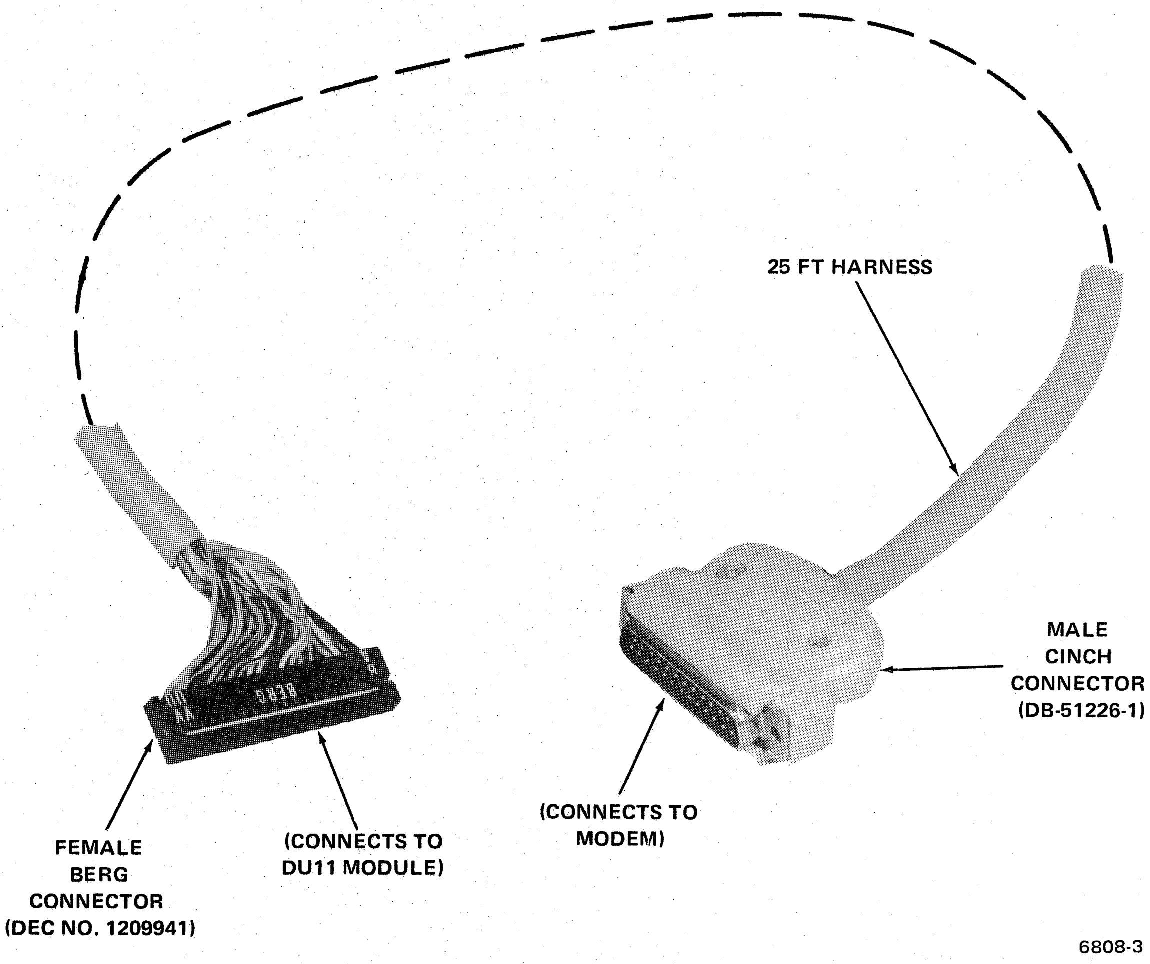

25 FT HARNESS
MALE CINCH CONNECTOR (DB-51226-1)
(CONNECTS TO MODEM)
FEMALE BERG CONNECTOR (DEC NO. 1209941)
(CONNECTS TO DU11 MODULE)
BERG
6808-3

Figure 2-6 BC05C-25 Cable Harness Used to Connect DU11-DA to Bell 201 Modem

2.1.5 Priority Assignment

The priority level is determined by the priority plug located on the DU11 module. The DU11 normally has a priority level of BR5. However, the priority may be changed by simply replacing the BR5 plug with a plug wired for a different priority level.

NOTE

If the priority level is changed, the software must be modified accordingly.

2.2 INITIAL TESTING

The DU11 must be tested prior to placing the unit into operation. For initial test procedures, refer to the engineering specification, A-SP-DU11-0-4, which is provided with each DU11 delivered.

NOTE

Before running diagnostics on interface model DU11-DA, disconnect the modem cable (BC05C-25) from the rear of the modem and install the modem test connector as shown in Figure 2-9.

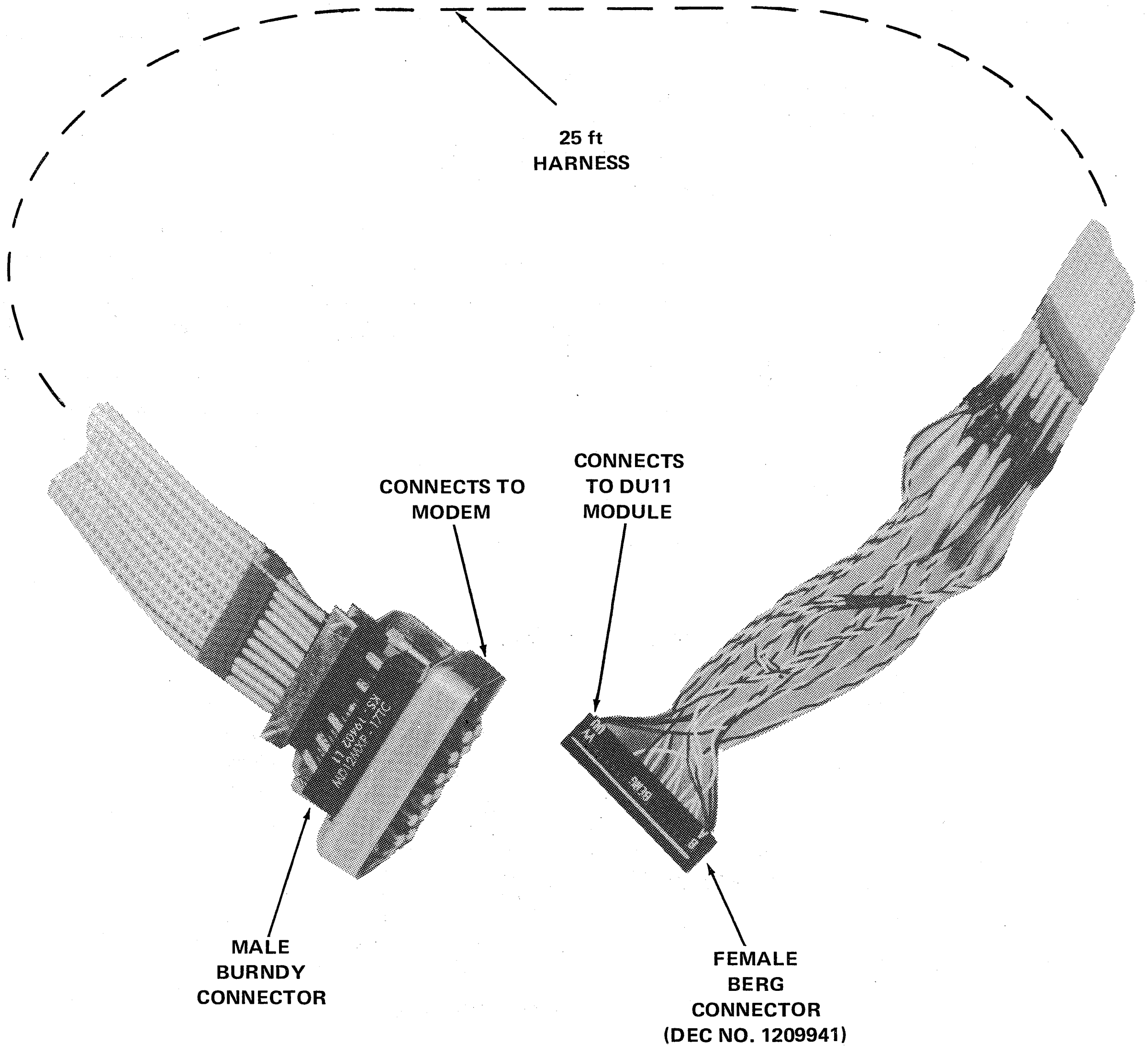
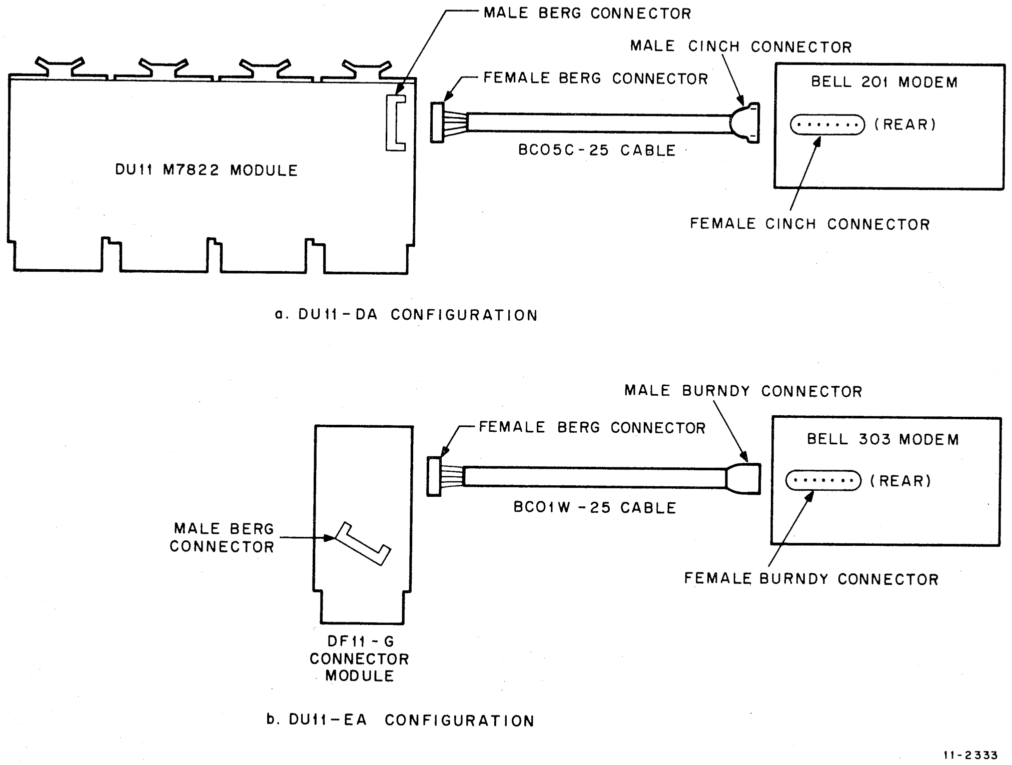


Figure 2-7 BC01W-25 Cable Harness Used to Connect DU11-EA to Bell 303 Modem



a. DU11-DA CONFIGURATION

b. DU11-EA CONFIGURATION

Figure 2-8 DU11 to Modem Connection

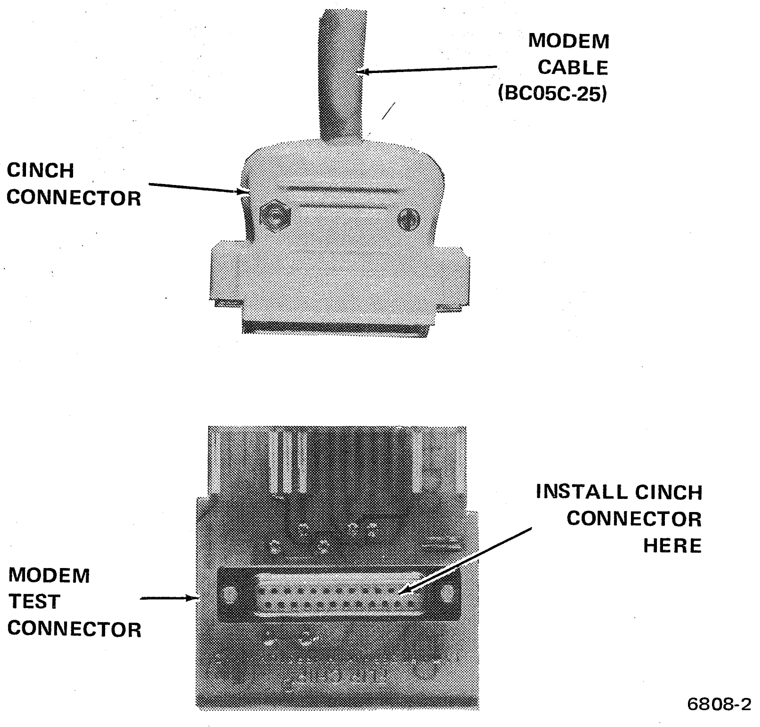


Figure 2-9 Modem Test Connection Installation

Table 2-1
Jumper Assignments

Jumper No. and Location	Normal Configuration	Function
W2/D5	Removed	This jumper may be installed to enable the receiver to synchronize internally upon receiving just one sync character, thereby negating the normal requirement of receiving two contiguous sync characters to achieve synchronization in the internal synchronous mode.
W4/D6	Installed	This jumper may be removed to disable CLR OPT L (Clear Option), thereby preventing clearing of bits 03, 02, and 01 in the RXCSR (see Chapter 3 for bit descriptions).
W5, W6/D6	Installed	These jumpers may be removed to disconnect the secondary data channel between the modem and the DU11. Removed at customers request.
W9–W14/D4	Floating	These jumpers control the receiver and transmitter interrupt vector address (Paragraph 2.1.3) bits: **Jumper** — **Address Bit** W9 — BUS D03 W10 — BUS D04 W11 — BUS D05 W12 — BUS D06 W13 — BUS D07 W14 — BUS D08
W15/D4	Installed	This jumper may be removed to inhibit the BUS NPR L input to the interrupt control logic. (Removed only if PDP-11/20 processor is used without KH option.)
W16/D1	Removed	This jumper must be installed if the DU11 is mounted in a DD11-A peripheral mounting panel (Paragraph 2.1.1.1).

CHAPTER 3
DEVICE REGISTERS AND INTERRUPT REQUESTS

3.1 SCOPE

This chapter provides a complete description of the DU11 device registers and the interrupt requests employed to service those registers.

3.2 DEVICE REGISTERS

All software control of the DU11 is performed by means of five device registers. These registers have been assigned bus addresses and can be read or loaded (with the exceptions noted) using any PDP-11 instruction referring to their addresses. Address assignments can be changed via the rocker switches to correspond to any address within the floating address range of 160010 to 163776.

3.2.1 Register Address Assignments

The five device registers and associated DU11 addresses are listed in Table 3-1.

3.2.2 Register Title and Bit Assignments

Each of the five device registers plays a specific role in controlling and monitoring DU11 operation. Register titles, bit titles, and read/write capability labeling are intended to facilitate the programmer's understanding of the purpose of each register relative to interface operation and to simplify software preparation.

3.2.2.1 Title Assignments – Register titles and functions are listed below:

a. RXCSR – programmed and monitored (read/write) to control the RCVR (receiver) portion of the interface; to communicate interface status, requests, and supervisory data to the modem; and to monitor status and supervisory data inputs from the modem.

b. RXDBUF – monitored (read only) to detect interface RCVR status flags and RCVR parallel data outputs.

c. PARCSR – programmed (write only) to establish the overall operating parameters of the DU11, i.e., the mode of operation (synchronous or isochronous), word length (5, 6, 7, or 8 bits plus parity), parity (enabled or disabled), parity sense (odd or even), and sync character configuration.

Table 3-1
DU11 Register Address Assignments

Register	Mnemonic	Address	Program Capability
Receiver Status Register	RXCSR	16XXX0	Read/Write
Receiver Data Buffer	RXDBUF	16XXX2	Read Only
Parameter Status Register	PARCSR	16XXX2	Write Only
Transmitter Status Register	TXCSR	16XXX4	Read/Write
Transmitter Data Buffer	TXDBUF	16XXX6	Write Only

XXX = Selected in accordance with floating device address scheme described in Appendix B.

d. TXCSR – programmed and monitored (read/write) to control the XMTR (transmitter) portion of the interface, to control the resetting and initialization of the interface, and to control and monitor the maintenance mode operation of the interface.

e. TXDBUF – programmed (write only) to provide parallel data to the interface XMTR for serial transmission to the modem.

3.2.2.2 Bit Assignments – The bit names indicate the function of the bit. The bits that are defined as "not used" or "write-only" are always read as 0. In the same respect, attempts to program the "not used" bits or "read-only" bits have no effect on the bit.

The following figures and tables describe register content. Figures 3-1 through 3-5 illustrate the register formats. Tables 3-2 through 3-6 list bit descriptions.

The mnemonic INIT is used frequently in the following tables and refers to the initialization signal generated by the processor. The processor will issue an INIT signal for any one of the following conditions:

a. A programmed RESET instruction is processed.
b. The processor START switch is pressed.
c. The power fail sequence occurs.

During a power fail sequence, INIT is asserted when power is going down and again when power is coming up.

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
DATA SET CH	RING	CLR TO SEND	CAR-RIER	REC ACT	SEC REC DATA	DATA SET RDY	STRIP SYNC	RX DONE	RX INTEB	DATA SET INTEB	SCH SYNC	SEC XMIT DATA	REQ TO SD	DATA TERM RDY	NOT USED
R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	

11-2244

Figure 3-1 Receiver Status Register (RXCSR)

Table 3-2
Receiver Status Register Bit Description

Bit	Name	Description
15	DAT SET CH (Data Set Change)	When set, this bit indicates a modem status change. This bit is set by a transition of any of the following lines: ● Ring ● Clear To Send ● Carrier ● Secondary Received Data ● Data Set Ready If bit 05 of this register is set, the setting of this bit will cause a RCVR interrupt. Read-only bit; cleared by INIT, Master Reset, and the DTI SEL 0 (RXCSR read strobe).
14	RING (Ring)	This bit reflects the state of the modem Ring line. When set, this bit indicates that a Ring signal is being received from the modem. Read-only bit.

Table 3-2 (Cont)
Receiver Status Register Bit Description

Bit	Name	Description
13	CLR TO SD (Clear to Send)	This bit reflects the state of the Clear to Send line from the modem. When set, this bit indicates that the modem is on and ready to accept data from the interface for transmission. Read-only bit.
12	CARRIER (Carrier)	This bit reflects the state of the modem carrier. When set, this bit indicates the Carrier is up. Read-only bit.
11	REC ACT (Receiver Active)	When the internal synchronous mode is selected, this bit is set when the proper number of contiguous sync characters (either 1 or 2, normally set for 2) have been received. If external synchronous or isochronous mode is selected, this bit follows the state of the Search Sync bit (bit 04 of this register). See Paragraph 4.3 for RCVR synchronization information. Read-only; cleared by INIT, Master Reset, and SCH SYNC (1) H (Search Sync) making 1 to 0 transition.
10	SEC RCV DAT (Secondary Receive Data)	This bit reflects the state of the Secondary Receive Data line from the modem. This bit provides a receive channel for supervisory data from the modem to the processor. Read-only bit.
09	DAT SET RDY (Data Set Ready)	This bit reflects the state of the Data Set Ready line from the modem. When set, this bit indicates that the modem is powered up and ready to transmit and receive data. Read-only bit.
08	STRIP SYNC (Strip Sync)	This bit determines whether sync characters received from the modem are to be presented to the program for reading. When this bit is set, receive characters that match the contents of the Sync register do not cause a RCVR interrupt provided no errors are detected, i.e., bit 15 of the RXDBUF is clear. Read/write bit; cleared by INIT and Master Reset.
07	RX DONE (Receiver Done)	This bit is set when synchronization has been achieved and a character has been loaded into the RXDBUF, provided the STRIP SYNC bit is not set. If the STRIP SYNC bit is set and the received character is a sync character without errors, i.e., bit 15 of the RXDBUF is clear, this bit will not be set. When set, this bit will cause a RCVR interrupt request provided bit 06 of this register is set. Read-only bit; cleared by INIT, Master Reset, and the DTI SEL 2 (RXDBUF read strobe).

Table 3-2 (Cont)
Receiver Status Register Bit Description

Bit	Name	Description
06	RX INTEB (Receiver Interrupt Enable)	When set, allows a RCVR interrupt request to be generated when the RX DONE bit is set. Read/write bit; cleared by INIT and Master Reset.
05	DAT SET INTEB (Data Set Interrupt Enable)	When set, allows a RCVR interrupt request to be generated when the DAT SET CH bit is set. Read/write bit; cleared by INIT or Master Reset.
04	SCH SYNC (Search Sync)	When set in the internal synchronous mode, enables the RCVR synchronization logic and causes the RCVR to start comparing incoming data bits to the contents of the Sync register in an attempt to recognize a sync character. When set in the isochronous mode, enables the RX DONE flag generation logic. When set in the external synchronous mode, enables the RX DONE flag generation logic and causes the RCVR to start framing incoming characters. Read/write bit; cleared by INIT and Master Reset.
03	SEC XMIT (Secondary Transmit Data)	This bit reflects the state of the Secondary Transmit Data line to the modem. This bit provides a transmit channel for supervisory data from the modem to the processor. Read/write bit; optionally cleared by INIT or Master Reset.
02	REQ TO SD (Request to Send)	When set, this bit causes the Request to Send line to the modem to be asserted. The Request to Send line is a control lead to the modem. This line must be asserted before the interface can transmit data to the modem. Read/write bit; optionally cleared by INIT and Master Reset.
01	DATA TERM RDY (Data Terminal Ready)	When set, this bit indicates the interface is powered up, programmed, and ready to receive data from the modem. Setting this bit causes the Data Terminal Ready line to the modem to be asserted. The Data Terminal Ready line is a control lead for the modem communication channel. When asserted, it permits the interface to be connected to the channel. Read/write bit; optionally cleared by INIT and Master Reset.

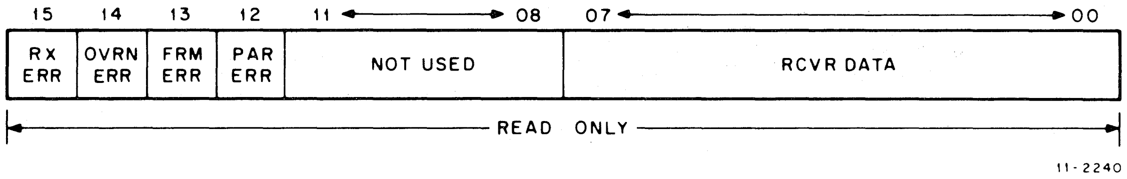


Figure 3-2 Receiver Data Buffer (RXDBUF)

Table 3-3
Receiver Data Buffer Bit Description

Bit	Name	Description
15	RX ERR (Receiver Error)	This bit is set whenever one of the three receiver error bits is set (logical OR of bits 14, 13, and 12). Read-only bit; cleared only when bits 14, 13, and 12 are cleared.
14	OVRN ERR (Overrun Error)	When set, this bit indicates that the processor has failed to service the RX DONE flag within the time required to load another character into the RXDBUF, i.e., (1/baud rate) X (bits per character) seconds. Hence, the previous character was over-written (lost). Read-only bit; cleared by INIT, Master Reset, and DTI SEL 2 (RXDBUF read strobe).
13	FRM ERR (Framing Error)	When set, indicates that character received was not followed by a valid STOP bit. This error only occurs in the isochronous mode of operation. Read-only bit; cleared by INIT, Master Reset, and DTI SEL 2.
12	PAR ERR (Parity Error)	When set, indicates that the parity of the received character does not agree with the parity programmed (odd or even). If parity is not programmed, this bit is always cleared. Read-only bit; cleared by INIT, Master Reset, and DTI SEL 2.
07–00	RCVR DATA (Receiver Data)	This register holds the received character for transfer to the program. The buffer is right justified for 5, 6, 7, or 8 bits. If parity is received it is also loaded into the buffer at the next vacant higher order bit position. Therefore, if a 5-bit character plus parity is framed by the RCVR, the parity bit would be loaded into bit position 05 in the RXDBUF and presented to the program for reading. If an 8-bit character plus parity is framed, the parity bit would not be presented to the program for reading. Read-only buffer; cannot be cleared, INIT or Master Reset sets the buffer to all ones.

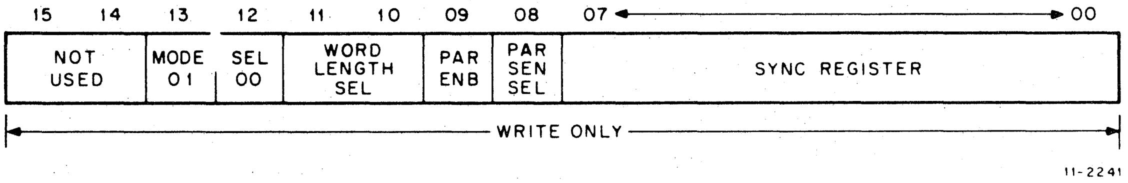


Figure 3-3 Parameter Status Register (PARCSR)

Table 3-4
Parameter Status Register Bit Description

<table>
<tr><th>Bit</th><th>Name</th><th>Description</th></tr>
<tr><td>13 and 12</td><td>MODE SEL
(Mode Select)</td><td>These bits control the mode of operation. Modes are selected as follows:
<table>
<tr><th>Mode</th><th>Bit 13</th><th>Bit 12</th></tr>
<tr><td>Internal Synchronous</td><td>1</td><td>1</td></tr>
<tr><td>External Synchronous</td><td>1</td><td>0</td></tr>
<tr><td>Isochronous</td><td>0</td><td>0</td></tr>
</table>
Any other mode select bit combinations will produce errors in the interface.
Write-only bits.</td></tr>
<tr><td>11 and 10</td><td>WORD LEN SEL
(Word Length Select)</td><td>These bits control the length of characters received and transmitted by interface. Word length (not including parity) is selected as follows:
<table>
<tr><th>Bits per Character</th><th>Bit 11</th><th>Bit 10</th></tr>
<tr><td>5</td><td>0</td><td>0</td></tr>
<tr><td>6</td><td>0</td><td>1</td></tr>
<tr><td>7</td><td>1</td><td>0</td></tr>
<tr><td>8</td><td>1</td><td>1</td></tr>
</table>
Write-only bits.</td></tr>
<tr><td>09</td><td>PAR ENB
(Parity Enable)</td><td>If this bit is set, parity will be generated by the XMTR and checked by the RCVR. If character length is less than eight bits, the parity bit is loaded into the RXDBUF for reading by the program. If bad parity is detected at the RCVR, the parity error flag is set (bit 12 of the RXDBUF).
Write-only bit.</td></tr>
</table>

Table 3-4 (Cont)
Parameter Status Register Bit Description

Bit	Name	Description
08	PAR SEN SEL (Parity Sense Select)	When the Parity Enable bit (bit 09 of this register) is set, the sense of the parity (odd or even) is controlled by this bit. When this bit is set, even parity is generated by the XMTR and checked for by the RCVR (the program does not have to provide a parity bit to the XMTR). When this bit is cleared, odd parity is generated and checked. Write-only bit.
07–00	Sync Register	This register contains the sync character. The sync character is used by the RCVR to detect received sync characters and thereby achieve synchronization. The sync character is used as a fill character by the XMTR when operating in the synchronous mode. Fill characters are transmitted when the program fails to provide characters to the XMTR fast enough to maintain continuous transmission, i.e., (1/baud rate) X (bits per character) seconds - 1/2 (bit time).

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
DNA	MAINT DATA	SS CLK	MS 01	MS 00	RX INP	NOT USED	MST RST	TX DONE	TX INTEB	DNA INTEB	SEND	HALF DUP	NOT USED		BREAK
R	R/W	R/W	R/W	R/W	R		W	R	R/W	R/W	R/W	R/W			R/W

11-2242

Figure 3-4 Transmitter Status Register (TXCSR)

Table 3-5
Transmitter Status Register Bit Description

Bit	Name	Description
15	DNA (Data Not Available)	This bit is set by the XMTR when a fill character is transmitted. This applies only to the synchronous mode of operation and is caused by late program response to a TX DONE interrupt request. The processor response to TX DONE must be within (1/baud rate) X (bits per character) seconds - 1/2 (bit time). If not, the fill character is transmitted. If bit 05 of this register is set, setting this bit causes an XMTR interrupt request. Read-only bit; cleared by INIT, Master Reset, and DTI SEL 4 (TXCSR read strobe).

Table 3-5 (Cont)
Transmitter Status Register Bit Description

Bit	Name	Description
14	MAINT DATA (Maintenance Data)	This bit is used in the internal loop and external loop maintenance modes by the diagnostic program to simulate an input to the RCVR. Read/write bit; cleared by INIT and Master Reset.
13	SS CLK (Single Step Maintenance Clock)	This bit is used in the internal loop and external loop maintenance modes by the diagnostic program to simulate the XMTR and RCVR clocks. Read/write bit; cleared by INIT or Master Reset.
12 and 11	MS01/MS00 (Maintenance Mode Select 01 & 00)	These bits are used to select the normal mode of operation or one of three maintenance modes. Modes are selected as follows: Mode / Bit 12 / Bit 11 Normal / 0 / 0 Internal Maintenance Loop / 0 / 1 External Maintenance Loop / 1 / 0 System Test / 1 / 1 Read/write bits; cleared by INIT and Master Reset.
10	RX INP (Receiver Input)	This bit monitors the RCVR input in the internal loop and external loop maintenance modes. Read-only.
08	MSTRST (Master Reset)	This bit is used to generate a CLR (Clear) pulse, which initializes the registers and the XMTR and RCVR and inhibits the BUS SSYN L (Slave Sync) signal. Write-only.
07	TX DONE (Transmitter Done)	This bit is set by INIT and Master Reset and when the first bit of the character contained in the XMTR register is placed on the XMTR output line. If bit 06 of this register is set when this bit is set, an XMTR interrupt request is generated. Read-only bit; cleared by LD TXDBUF (TXDBUF load strobe).

Table 3-5 (Cont)
Transmitter Status Register Bit Description

Bit	Name	Description
06	TX INTEB (Transmitter Interrupt Enable)	When set, this bit allows an XMTR interrupt request to be generated by the TX DONE bit. Read/write bit; cleared by INIT and Master Reset.
05	DNA INTEB (Data Not Available Interrupt Enable)	When set, this bit allows a XMTR interrupt request to be generated by the DNA bit. Read/write bit; cleared by INIT and Master Reset.
04	SEND (Send)	When set, this bit enables the XMTR and transmission will start when a character is loaded into the TXDBUF. This bit must remain set until the entire message is transmitted. If not, transmission of the character currently in the XMTR register is completed and the XMTR will enter the idle state. Read/write bit; cleared by INIT and Master Reset.
03	HALF DUP (Half Duplex)	When this bit is set, operation will be in the half duplex mode. In this mode the RCVR is disabled whenever bit 04 of this register is set. Read/write bit; cleared by INIT and Master Reset.
00	BREAK (Break)	When this bit is set, the serial XMTR output is held in the space (constant LOW) condition; otherwise, operation is normal. This bit is used by the diagnostic program in the internal loop or external loop maintenance modes to inhibit the XMTR output while inputting data to the RCVR via bit 14 of this register. Read/write bit; cleared by INIT and Master Reset.

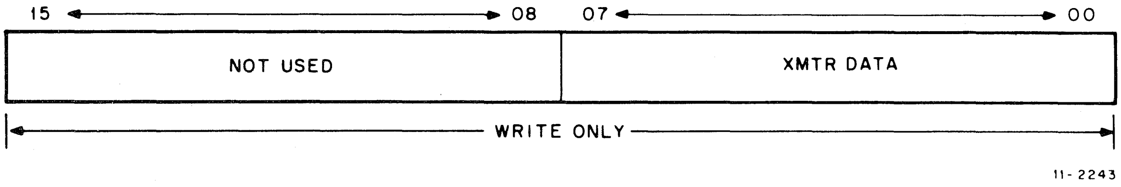


Figure 3-5 Transmitter Data Buffer (TXDBUF)

Table 3-6
Transmitter Data Buffer Bit Description

Bit	Name	Description
07–00	XMTR DATA (Transmitter Data)	This register is loaded by the program with the character to be transmitted. Character length is from 5 to 8 bits. The character is right-hand justified. If a parity bit is programmed, it is generated by the interface. Write-only bits; an INIT or Master Reset places all ones in this register.

3.3 INTERRUPT REQUESTS

The DU11 uses BR interrupts to gain control of the bus and cause a program interrupt, thereby causing the processor to branch to a subroutine.

The interface uses two interrupt vectors: one for the RCVR section and one for the XMTR section. If simultaneous RCVR and XMTR interrupt requests occur, the RCVR has priority.

Both the XMTR and RCVR sections of the interrupt control logic handle interrupt requests from two sources. A XMTR interrupt request is generated by the setting of the TX DONE bit or the DNA bit provided the TX INTEB and the DNA INTEB bits are set. A RCVR interrupt request is generated by setting the RX DONE bit or the DAT SET CH bit, provided the RX INTEB and DAT SET INTEB bits are set.

The standard DU11 interrupt priority level is BR5. However, the priority level can be changed by replacing the priority plug.

The DU11 interrupt vector addresses are floating. Floating vector addresses are used for all options and are assigned according to the scheme described in Appendix B. The vector addresses can be changed via jumpers W9–W14 in the interrupt control logic.

NOTE

If the DU11 priority plug or an interrupt vector address is changed, all DEC programs or other software referring to the standard priority level or interrupt vector addresses must also be changed.

CHAPTER 4
PROGRAMMING REQUIREMENTS AND RECOMMENDATIONS

4.1 INTRODUCTION

To program the DU11 in the most efficient manner, the programmer must understand fully the control signal and timing requirements of the device. The following paragraphs discuss DU11 operation from a programming point of view and describe recommended programming methods. It is beyond the scope of this manual to provide detailed programming information. For more detailed information on programming in general, refer to the *Paper-Tape Software Programming Handbook,* DEC-11-XPTSA-A-D, and the individual program listings.

4.2 PROGRAMMING THE TRANSMITTER IN THE SYNCHRONOUS MODE

4.2.1 Loading the PARCSR

Once the transmitter is initialized via the BUS INIT pulse or MSTRST, the PARCSR register must be programmed (loaded) to select the mode of operation (synchronous in this case), character length, and parity. At this point the Sync register will contain all ones. Before any necessary handshaking is done with the modem, the program must load the Sync register with the desired character. When the Sync register is loaded, the character will be used for both XMTR and RCVR operation.

4.2.2 Enabling the Transmitter

Once handshaking is complete, the program can assert the SEND bit in the TXCSR. When SEND is asserted, the XMTR is enabled but will not start transmitting data until the first character is loaded into the TXDBUF. If SEND is cleared during transmission, the character currently being transmitted will be completed, the transmit line will go to a mark hold state, the internal XMTR logic will enter the idle state, and synchronization with the RCVR will be lost. When SEND is cleared, there is no guarantee that the TX DONE bit will be asserted when current character transmission is complete.

4.2.3 Detecting the Last Character of the Message

When it is necessary to know when the entire message has been transmitted, the DU11 may be programmed as follows:

a. Just prior to loading the last character of the message into the TXDBUF, clear the TX DONE INTEB bit and set the DNA INTEB bit.

b. When the last character is loaded into the register, the TX DONE bit will set but will not cause an interrupt request.

c. After the last character is transmitted, the transmitter will transmit the sync character and assert DNA, which causes an interrupt request.

d. The DNA interrupt is notification to the program that the entire message has been transmitted.

4.2.4 Transmitting Initial Sync Characters to Establish Synchronization

The transmission of initial sync characters can be accomplished in one of two ways:

a. The program may arrange its data buffer such that the required number of sync characters precede any messages. In this case, the Sync register may or may not contain the sync character. If the Sync register is not loaded, it will contain all ones subsequent to a BUS INIT or MSTRST.

Assuming that any necessary handshaking has been completed and that SEND has been asserted, the program can commence transmission by loading a sync character into the TXDBUF. When the first data bit is transferred to the communication line, the TX DONE bit will be asserted. If the TX INTEB bit is set, an interrupt request will be generated, and the program must load another sync character into the TXDBUF.

If the sync character was not initially loaded into the Sync register, then synchronization cannot be guaranteed unless the program response time to the TX DONE bit is less than (1/baud rate X bits per char) seconds - 1/2 (bit time). This can be verified by the absence of the DNA bit in the TXCSR.

b. The program can also enable transmission of initial sync characters from the Sync register. Assuming any necessary handshaking is complete and SEND is asserted, the program loads the Sync register with a sync character, sets the DNA INTEB bit, and clears the TX INTEB bit. The program then loads a sync character into the TXDBUF and transmission begins. The TX DONE interrupt is inhibited so the contents of the Sync register are transferred to the XMTR register upon the completion of transmission of the first sync character. The second sync character is then transmitted and a DNA interrupt is generated notifying the program. The program then allows the transmission of sync characters to continue by simply monitoring the DNA until the desired number have been transmitted. Note that DNA is reset each time the program reads the TXCSR and set again when the first bit of the next sync character is placed on the communication line.

NOTE

It is suggested that a minimum of five sync characters be transmitted. In systems that are prone to error because of lost synchronization, as many as twelve sync characters may be desirable.

When the desired number of sync characters have been transmitted, the program sets the TX INTEB bit, thereby enabling the TX DONE interrupt, and responds to the interrupt by loading a message character into the TXDBUF.

4.2.5 Transmitting Sync Characters to Maintain Synchronization

After synchronization has been achieved, it can be maintained by the program by inserting sync characters into the message or by ignoring the TX DONE bit, thereby allowing sync characters from the Sync register to be transmitted.

If the latter method is chosen, it can be programmed in one of two ways. The first way would be to set the DNA INTEB bit and clear the TX INTEB bit. The program would then ignore the TX DONE bit and the XMTR would transmit a sync character and assert DNA. The program would monitor the DNA bit and, when the desired number of sync characters are transmitted, set the TX INTEB bit thereby enabling the TX DONE interrupt. The program would then respond to the TX DONE interrupt by loading a message character into the TXDBUF, thereby terminating the transmission of sync characters. The second way would be to clear the TX INTEB and DNA INTEB bits for a given period of time during message transmission thereby allowing sync characters from the sync register to be transmitted.

NOTE

The SEND bit in the TXCSR must remain set for the duration of the message; any on to off transition will cause the XMTR to enter an idle state after completion of current character transmission.

4.3 PROGRAMMING THE RCVR IN THE INTERNAL SYNCHRONOUS MODE

Once the program has completed any necessary handshaking, the receiver logic can be enabled. The program enables the receiver logic by setting the SCH SYNC (Search Sync) bit in the RXCSR. Assuming a sync character has been loaded into the Sync register (this must be done in the internal synchronous mode), the receiver begins to compare incoming character bits with the character held in the Sync register.

NOTE

For the receiver to become synchronized with XMTR either one or two contiguous sync characters must be received. The number of sync characters required is jumper selectable. The standard configuration requires two sync characters.

NOTE

Though the DU11 may be jumpered to synchronize on two contiguous sync characters, there is a situation which, if it develops, will prevent RCVR synchronization on only two contiguous sync characters. If, while the DU11 is searching for synchronization, it recognizes a sync character that is not followed contiguously by a second sync character, the RCVR internal logic resets, thereby inhibiting the RCVR bit detection logic for two bit times. Should the first bits of a proper sync character sequence occur during that two bit time period, the RCVR will fail to achieve synchronization.

When two contiguous sync characters are received, the REC ACT (Receiver Active) bit is set and any characters received after that will cause RX DONE interrupt requests, provided the RX INTEB bit is set and the STRIP SYNC bit is cleared.

NOTE

The SCH SYNC bit must remain set for the duration of the message. If not, the character being received at the time of the on to off transition will be lost along with synchronization.

If the programmer wishes the RCVR to discard all sync characters after synchronization is achieved, the STRIP SYNC bit in the RXCSR must be set. The STRIP SYNC bit inhibits the RX DONE interrupt whenever a sync character is received with no errors; however, the sync character is still held in the RXDBUF until the next character is received.

If the program fails to read the RXDBUF in response to a RX DONE interrupt, overrun errors will occur. When the RXDBUF is not serviced in the time required to receive the next character, i.e., (1/baud rate X bits per character) seconds, the character presently being held in the RXDBUF is overwritten by the next received character and the OVRN ERR (overrun error) bit is set in the RXCSR.

NOTE

The information in the following paragraph must be strictly adhered to or RCVR synchronization problems will be encountered.

If the DU11 is configured to achieve synchronization on two contiguous sync characters then receiver operation may be terminated (after the entire message is received) by simply clearing the SCH SYNC bit in the RXCSR. However, if only one character is required to achieve synchronization, receiver termination is a little more complex. If the SCH SYNC bit is cleared while a sync character is present in the RXDBUF, false synchronization will occur when the receiver is enabled (SCH SYNC bit set) to receive the next message. The program must ensure that this does not happen by transmitting a pad character, i.e., a non-sync character, immediately after the transmission of the terminating control character.

4.4 PROGRAMMING THE RCVR IN THE EXTERNAL SYNCHRONOUS MODE

The external synchronous mode enables the RCVR logic to set to the synchronize state immediately upon the assertion of the SCH SYNC (1) H input. This mode is designed for use with communication equipment capable of accomplishing synchronization external to the DU11. When the program sets the SCH SYNC bit, the REC ACT bit sets and the RCVR starts framing characters on the very next bit received. When the selected number of bits are received, the received character is transferred into the RXDBUF and the RX DONE bit is set causing an interrupt request. All other features and parameters of the internal synchronous mode apply to this mode also.

4.5 PROGRAMMING THE XMTR IN THE ISOCHRONOUS MODE

4.5.1 Loading the PARCSR

Once the XMTR is initialized via BUS INIT or MSTRST, the PARCSR must be programmed to select the mode of operation (isochronous in this case), character length, and parity. It is not necessary to load the sync register in this mode as sync characters are not required to achieve synchronization and the transmitter is not required to transmit continuously.

4.5.2 Enabling the XMTR

When the required handshaking is complete, the program sets the SEND and TX INTEB bits and loads a character into the TXDBUF. The XMTR adds the START and STOP bits and transmits the character to the modem. As soon as the first character bit is placed on the communication line by the XMTR, the TX DONE bit is asserted and remains asserted until the XMTR services the TXDBUF or clears the SEND bit.

4.6 PROGRAMMING THE RCVR IN THE ISOCHRONOUS MODE

RCVR operation is initiated by the assertion of SCH SYNC. When the program sets the SCH SYNC bit, the REC ACT bit sets and the RCVR starts framing characters upon receipt of the START bit from the XMTR. When the selected number of character bits are received, the RCVR tests the line for a valid STOP bit, transfers the received character into the RXDBUF, and sets the RX DONE bit. If the STOP bit is not detected, the FRM ERR (Framing Error) bit is also set. If the program fails to service the RXDBUF before the next character is framed, the OVRRN ERR bit is set.

APPENDIX A
REPRESENTATIVE MODEM FACILITIES AVAILABLE

Manufacturer	Model	Speed (Maximum)	Half or Full Duplex	Sync or Async	Type of Line	Comments
Bell System	103A	300 baud	Full Duplex	Async	DDD	
Bell System	103E	300 baud	Full Duplex	Async	DDD	Similar to 103A
Bell System	103F	300 baud	Full Duplex	Async	Private	
Bell System	113A	300 baud	Full Duplex	Async	DDD	Originate Only
Bell System	113B	300 baud	Full Duplex	Async	DDD	Answer Only
Bell System	201A	2000 baud	Either	Sync	DDD	Full Duplex on 2 calls
Bell System	201B	2400 baud	Either	Sync	Private	
Bell System	202B	1800 baud	Either	Async	DDD	
Bell System	202C	1200 baud	Either	Async	DDD	Full Duplex on 2 calls
Bell System	202D	1800 baud	Either	Async	Private	
Bell System	205B	600 baud 1200 baud 2400 baud	Full Duplex	Sync	Private	
Bell System	202E Series	1200 baud	Trans Only	Async	DDD Private	
Bell System	301B	40,800 baud	Either	Sync	Private Wide Band	
Bell System	303B, C, D, E	19,000 to 230,400 baud	Either	Sync	Private Wide Band	
Bell System	811B	110 baud	Either	Async	TWX Network	
Western Union	118-1A	180			Telegraph	
Western Union	1601-A	600			Voice	
Western Union	2121-A	1200			Voice Broad Band	
Western Union	2241-A	2400	Either	Either	Broad Band	
Western Union	100	200	Either	Async	Voice	

Manufacturer	Model	Speed (Maximum)	Half or Full Duplex	Sync or Async	Type of Line	Comments
Western Union	100	2400	Either	Async	Voice	
Western Union	300	18,000 40,000	Either	Sync	Broad Band	
Rixon	FM-12	1200	Either	Either	Voice Bell 4A	
Rixon	Sebit 48	4,800	Either	Sync	Voice Bell 4C	
General Electric	TDM 220	2,400	Either	Either	Private Bell 4B	

APPENDIX B
ADDRESS ASSIGNMENTS

B.1 FLOATING VECTORS

There is a floating vector convention used for communications (and other) devices that interface with the PDP-11 computer. These vector addresses are assigned, in order, starting at 300 and proceeding upwards to 777. Table B-1 shows the assigned sequence. It can be seen that the first vector address, 300, is assigned to the first DC11 in the system. If another DC11 is used, it would then be assigned vector address 310, etc. When the vector addresses have been assigned for all the DC11s (up to a maximum of 32), addresses are then assigned consecutively to each unit of the next highest ranked device (KL11 or DP11 or DM11, etc.), then to the other devices in accordance with the priority ranking (Table B-1).

If any of these devices are not included in a system, the vector address assignments move up to fill the vacancies. If a device is added to an existing system, its vector address must be inserted in the normal position and all other addresses must be moved accordingly. If this procedure is not followed, DEC software cannot test the system.

NOTE

The floating vectors range from addresses 300 to 777, but addresses 500 through 534 are reserved for special bus testers. In addition, address 1000 is used for the DS11 Synchronous Serial Line Multiplexer. Refer to Appendix A of the *PDP-11 Peripherals Handbook*, 1973-1974, for a complete discussion of Unibus addresses.

B.2 FLOATING DEVICE ADDRESS

There is a floating address convention for communication (and other) devices interfacing with PDP-11 computers. These addresses are assigned in order starting at 760010 and proceeding upward to 763776. Refer to Table B-2 for floating address sequence.

Table B-1
Priority Ranking for Floating Vectors
(starting at 300 and proceeding upwards)

Rank	Device	Vector Size (in octal)	Max. No.
1	DC11	10	32
2	KL11, DL11-A, DL11-C	10	16
3	DP11	10	32
4	DM11-A	10	16
5	DN11	4	16
6	DM11-BB	4	16
7	DR11-A	10*	32
8	DR11-C	10*	32
9	PA611 Reader	4*	16
10	PA611 Punch	4*	16
11	DT11	10*	8
12	DX11	10*	4
13	DL11-C, DL11-D, DL11-E	10	31
14	DJ11	10	16
15	DH11	10	16
16	GT40	10	1
17	LPS11	30*	1
18	DQ11	10	16
19	KW11-W	10	1
20	DU11	10	16

*The first vector for the first device of this type must always be on a $(10)_8$ boundary.

Table B-2
Floating Address Sequence

Rank	Device	Address Boundary Starting at 760010
1	DJ11	10 X (N) + 2
2	DH11	20 X (N) + 2 (go to next 20, 40, 60, or 100 boundary)
3	DQ11	10 X (N) + 2 (go to next 10, 20, 30, 40, 50, 60, 70, or 100 boundary)
4	DU11	10 X (N) + 2 (go to next 10, 20, 30, 40, 50, 60, 70, or 100 boundary)

N = number of each device

Table B-3
Floating Device Address Assignments

Device	First Address	Number of Register Addresses
DJ11 (GAP)	760010	–
DH11 #0	760020	8
DH11 #1	760040	8
DH11 (GAP)	760060	–
DQ11 #0	760070	4
DQ11 #1	760100	4
DQ11 (GAP)	760110	–
DU11 #0	760120	4
DU11 #1	760130	4

If, for example, a communication system is to contain two DH11s, two DQ11s, two DU11s and no DJ11s. the floating addresses would be assigned as shown in Table B-3.

If a DU11 in a system is not preceded by other devices in the floating vector area, it must have a starting address of 1600 for zero.

DU11 SINGLE LINE PROGRAMMABLE
SYNCHRONOUS INTERFACE
USER'S MANUAL
EK-DU11-OP-001

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 judgment is it complete, accurate, well organized, well written, etc.? Is it easy to use? ______________________

____

____

____

What features are most useful? ______________________

____

____

____

What faults do you find with the manual? ______________________

____

____

____

Does this manual satisfy the need you think it was intended to satisfy? ______________________

Does it satisfy *your* needs? ____________ Why? ______________________

____

____

____

Would you please indicate any factual errors you have found. ______________________

____

____

____

Please describe your position. ______________________

Name ______________________ Organization ______________________

Street ______________________ Department ______________________

City ____________ State ____________ Zip or Country ____________

CUT OUT ON DOTTED LINE

Fold Here

Do Not Tear - Fold Here and Staple

Digital Equipment Corporation
Technical Documentation Department
Digital Park, PK3-2
Maynard, Massachusetts 01754

digital

digital equipment corporation

Printed in U.S.A.