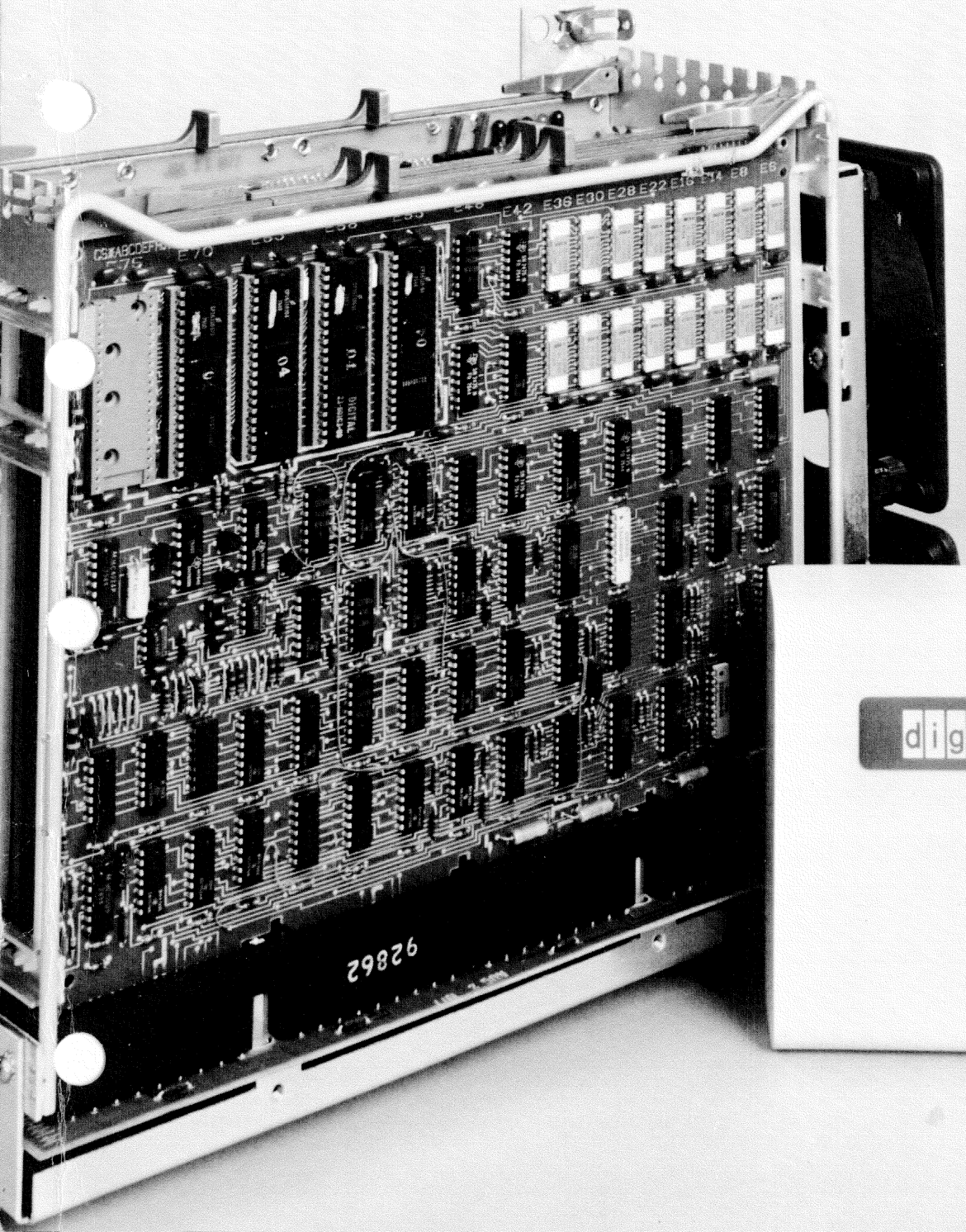
DLV11-E and DLV11-F asynchronous line interface user's manual



digital PDP 11/03-L

EK-DLV11-OP-001

DLV11-E and DLV11-F asynchronous line interface user's manual

digital equipment corporation • maynard, massachusetts

1st Edition, June 1977

Printed in U.S.A.

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

CONTENTS

CONTENTS (CONT)

CONTENTS (CONT)

FIGURES

FIGURES (CONT)

TABLES

TABLES (CONT)

CHAPTER 1
INTRODUCTION

1.1 PURPOSE AND SCOPE

The DLV11-E and DLV11-F are asynchronous line interface modules that interface the LSI-11 bus to any of several standard types of serial communications lines. The modules receive serial data from peripheral devices, assemble it into parallel data, and transfer it to the LSI-11 bus. They accept data from the LSI-11 bus, convert it into serial data, and transmit it to the peripheral devices. The two modules differ in that the DLV11-E offers full modem control, whereas the DLV11-F supports either 20 mA current loop or EIA-standard lines, but does not include modem control.

This manual describes these modules to the user. It treats the two modules together for those functions common to both, and separately for those areas in which they differ. It is assumed that the reader has a general familiarity with the operation of the LSI-11 computer and with the requirements of the peripheral equipment. Refer to *Microcomputer Handbook,* EB 06583 76, for detailed information about the LSI-11.

1.2 OPERATING FEATURES

Each asynchronous line interface is constructed on a single 21.6 cm × 122.7 cm (8.5 in × 5.0 in) dual-height module. The module mounts in any slot in the LSI-11's backplane. Both the DLV11-E and the DLV11-F have the following features:

- Jumper- or program-selectable crystal-controlled baud rates: 50, 75, 110, 134.5, 150, 300, 600, 1200, 1800, 2000, 3600, 4800, 7200, and 9600.
- Provisions for user-supplied external clock inputs for baud rate control.
- Jumper-selectable parity and data bit formats.
- LSI-11 bus interface and control logic for interrupt processing and vectored addressing of interrupt service routines.
- Control, status, and data buffer registers directly accessible via processor instructions.
- Program and peripheral connector plug compatible with the PDP-11 DL11 series of asynchronous line interface modules.

The DLV11-E is designed to interface data sets (modems with control capability) such as Bell models 103, 202C, and 202D.

The DLV11-F is designed for either 20 mA current loop equipment or EIA-standard "data leads only" (no modem control) operation. Flexibility is achieved by the use of wire wrap jumpers. Table 1-1 compares the features of the DLV11-E and DLV11-F with those of the DLV11 and the DL11 series.Refer to Paragraph 4.4, Timing Considerations, for further information.

Table 1-1 Feature Comparison

(NOTE: X indicates feature available.)

Features	DL11-A through D	DL11-E	DLV11	DLV11-F	DLV11-E
Programmable Baud Rates (Write Only Bits)				X	X
Modem Control		X			X
EIA "Data Leads Only"	X		X	X	X
20 mA Current Loop	X		X	X	
Jumper Selectable Active or Passive 20 mA Current Loop			X	X	
Error Flags	X	X		X	X
BREAK Generation Bit	X	X	X	X	X
Receiver Active Bit	X	X		X	X
Maintenance Bit	X	X		X	X
On-board Clocks for Split Speed Operation	X	X		X	X
Halt on Framing Error			X	X	X
Boot on Framing Error				X	X
UART Cleared by INIT	X	X		X	X
UART Cleared by DCOK			X		
No Trap on Write to RBUF	X	X		X	X
1.5 STOP BITS	X	X			
Modem Status Bit			X		

1.3 MODULE SPECIFICATIONS

The following specifications and particulars are for informational purposes only and are subject to change without notice.

Physical Characteristics

Dimensions

	Circuit Card	Circuit Card Plus Handles
Length:	21.6 cm (8.5 in)	22.8 cm (8.9 in)
Height:	12.7 cm (5.0 in)	13.2 cm (5.2 in)
Width:	1.3 cm (0.5 in)	1.3 cm (0.5 in)

Cable Connection — One 40-pin header connector

Mounting Requirements — Plugs directly into any dual-height slots on the LSI-11 backplane or LSI-11 expansion box backplane.

Electrical Characteristics

Module Type
- DLV11-E: M8017
- DLV11-F: M8028

Power Requirements
- 1.0 A (nominal) @ +5 V ±5%, 5.0 W
- 150 mA (nominal) @ +12 V ±5, 1.8 W

LSI-11 Bus Loading
- Presents one bus load.

Environmental Characteristics

Temperature

Operating	5° C to 50° C (41° F to 122° F)
Nonoperating	–40° C to 66° C (–40° F to 151° F)

Humidity (Operating and Nonoperating)
- 10% to 95%, maximum wet bulb 32° C (90° F) and minimum dew point 2° C (35° F)

Altitude

Operating	2.4 km (8,000 ft)
Nonoperating	9.1 km (30,000 ft)

1.4 MAINTENANCE

This manual explains the normal operation of the asynchronous line interface modules. This information and the diagnostic maintenance programs will aid the user when analyzing trouble symptoms to determine necessary corrective action. A set of engineering drawings is available for each of the two modules. Refer to DLV11-E Asynchronous Line Interface, Circuit Schematics (DIGITAL part number D-CS-M8017-0-1) or DLV11-F Asynchronous Line Interface, Circuit Schematics (DIGITAL part number D-CS-M8028-0-1).

Signal names in the DLV11-E and DLV11-F print sets are in the following basic form:

SOURCE SIGNAL NAME POLARITY

SOURCE indicates the drawing number of the print set where the signal originates. The drawing number of a print (K-3, K-4, K-5, etc.) is located above the title block.

SIGNAL NAME is the proper name of the signal. The names used in the print set are also used in this manual.

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

As an example, the signal:

(K-3) INIT H

originates on sheet K-3 of the drawings and means "when INIT is true, this signal is at approximately +3 V."

LSI-11 bus signal lines do not carry a SOURCE indicator. These names represent a bidirectional wire-ORed bus. As a result, multiple sources for a particular bus signal exist. The LSI-11 bus signal names begin with a "B" for "bussed."

The DLV11-E module is shipped with an H315 modem test connector included. This is plugged into the interface cable in place of a data set when running maintenance programs. The DLV11-F does not use this test connector.

A paper tape diagnostic maintenance program is shipped with the module for checkout and maintenance. The following programs are available:

DLV11-E: MAINDEC-11-DVDVA
DLV11-F: MAINDEC-11-DVDVC

CHAPTER 2
GENERAL DESCRIPTION

2.1 GENERAL

The DLV11-E is designed to interface equipment that transmits and receives data over communications lines and conforms to EIA Standard RS232C and CCITT Recommendation V.24. The DLV11-E is used by the program to control a communications data set through the use of control signals and handshake sequences.

The DLV11-F supports either EIA-compatible data lines or 20 mA current loop data lines. When configured for EIA support, the DLV11-F transmits and receives bipolar levels over the data lines to the device. This operation does not include control lines. When configured for 20 mA current loop operation, the DLV11-F can support either active or passive current loop devices. Figure 2-1 illustrates several applications of the modules.

2.2 MODULE FUNCTIONS

The DLV11-E and DLV11-F asynchronous line interface modules take data from the LSI-11 and convert it to the speed, character format, and signal levels required by the user's peripheral devices. Conversely, they assemble inputs from the peripheral devices into the format required for transfer to the computer. The computer program can address any of four registers in the interface modules to transfer data or status information. It can also enable the interface modules to generate interrupts. When a peripheral device requires service, the interface module will, if enabled, interrupt the program and vector to the necessary service routine.

Data passes through three main circuits on its way to and from the peripheral device (Figure 2-2). During computer output operations, parallel data is taken off the LSI-11 bus by a bus interface circuit and placed on the module's internal three-state bus. The data on the three-state bus enters a data buffer, where it is serialized and formatted for the peripheral device. From there it goes to a peripheral interface circuit that changes it from TTL to either EIA-compatible bipolar levels (DLV11-E or DLV11-F) or 20 mA current loop signals (DLV11-F only). The data then leaves the module on an interface cable and goes to the user's peripheral device. Data coming into the computer from the peripheral device goes through this process in reverse order.

The control functions within the interface module are carried out by circuits that handle I/O transfers, interrupt requests, and control and status information. The DLV11-E interfaces control signals as well as data between the LSI-11 and the peripheral. The extent of this interaction is determined by the program and the type of peripheral being supported.

The DLV11-E and DLV11-F also have a self-test function. When the computer program places the module in the maintenance mode, parallel data travels through the bus interface and the data buffer, is serialized, and then loops back through the data buffer, is converted back to parallel, and travels through the bus interface to the computer to be checked for accuracy.

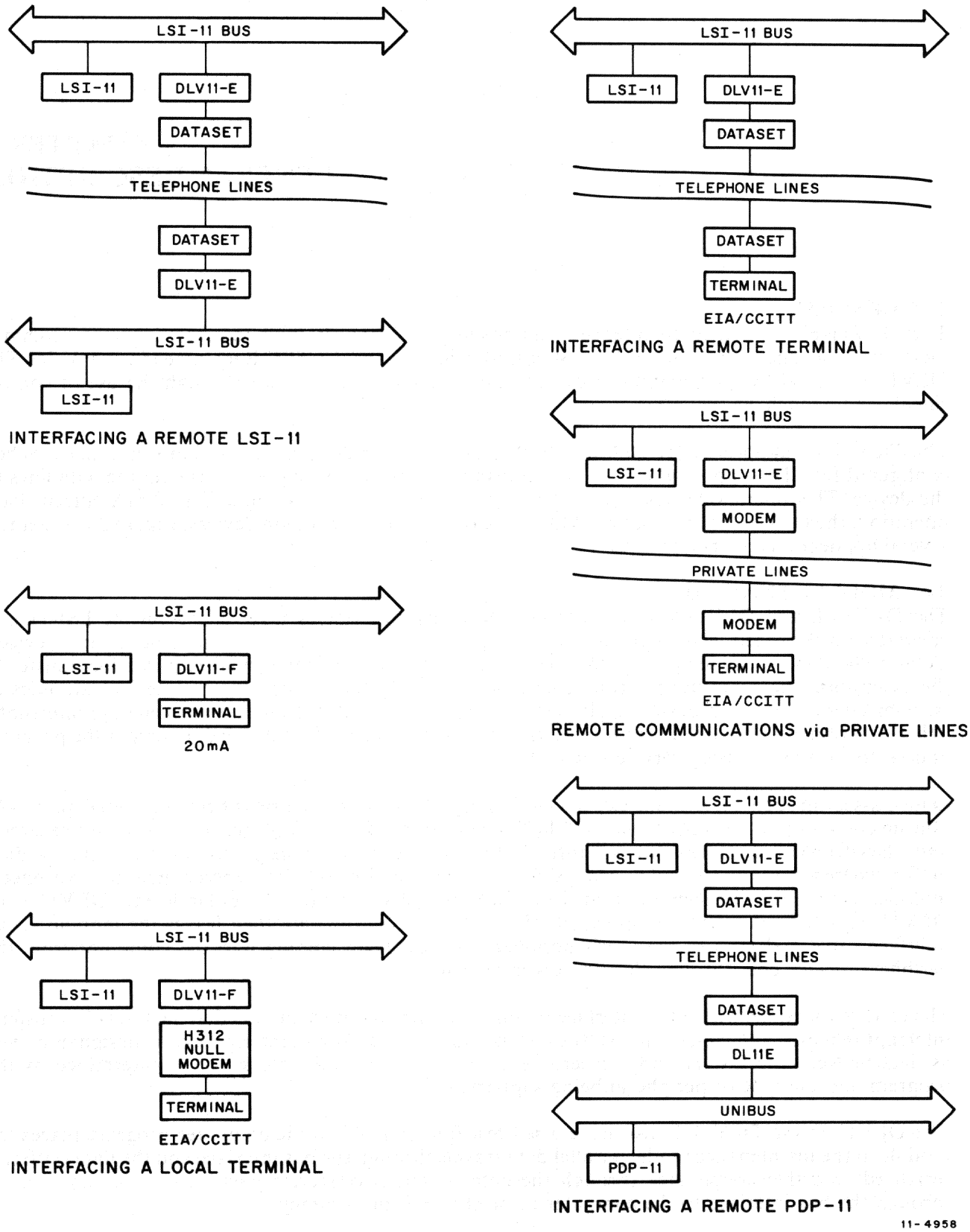


Figure 2-1 Interfacing Examples

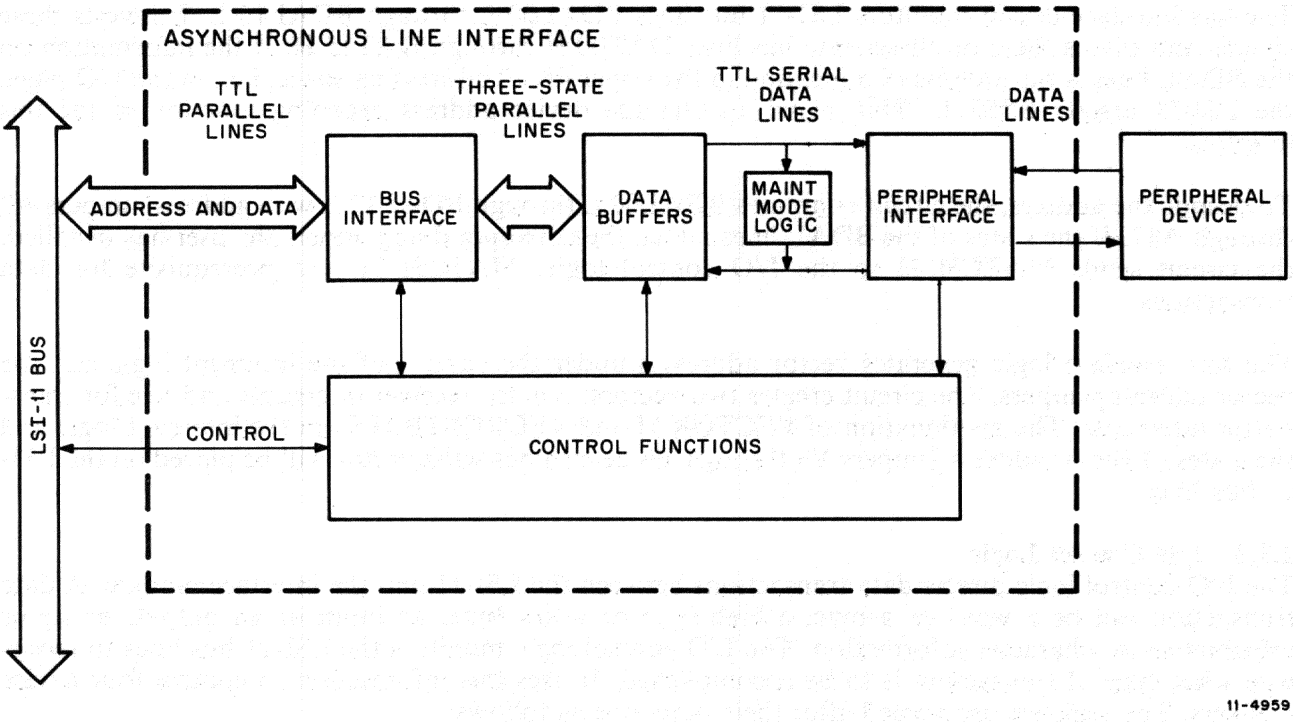


Figure 2-2 DLV11-E and DLV11-F Data Flow, Simplified Block Diagram

2.3 CIRCUIT FUNCTIONS

2.3.1 General

This section discusses the circuits on a functional level and is keyed to Figure 2-3. For a more detailed coverage of circuit operation, refer to Chapter 5.

2.3.2 Bus Interface

The bus interface circuit performs three basic functions:

1. It converts signal levels of data moving between the LSI-11 bus and the interface module's internal three-state bus.
2. It decodes the device address and produces an address match (MATCH H) signal.
3. It generates interrupt vectors and places them on the LSI-11 bus.

The LSI-11 signals are standard TTL levels. The module's internal three-state bus, however, has three signal conditions. It has TTL high and low states, and also a disabled state. When a bus interface transceiver output is disabled, it goes to a high impedance condition that does not affect other devices connected to the same line. This permits the lines to be used in both directions by high speed, low power devices.

The bus interface is normally enabled to receive from the LSI-11 bus. It can be switched to transmit onto the LSI-11 bus by either the I/O control logic or the interrupt logic. The signals received from the LSI-11 bus are ignored unless the address decoding function is enabled.

The bus interface circuit monitors LSI-11 bus lines BDAL00 L through BDAL15 L. It inverts these signals and places them on three-state bus lines DAT00 H through DAT15 H. If the information on the BDAL lines is the address of a location in the upper 4K of addressing space, i.e., in the I/O page, the LSI-11 asserts BBS7 L. This signal enables the device address decoding function in the bus interface.

To decode the address, the circuit compares BDAL03 L through BDAL12 L with address jumpers A3 through A12. If the states of the BDAL lines match the corresponding jumpers the user has installed, the circuit sends MATCH H to the I/O control logic. MATCH H is a prerequisite for data transactions.

The bus interface logic generates vector addresses under the control of the interrupt logic and the vector address jumpers. The circuit creates two vectors; one for receiver interrupts and one for transmitter interrupts. The combination of VECTOR H and VECRQSTB H from the interrupt logic and the states of vector address jumpers V3 through V8 determines what vector will be placed on the LSI-11 bus lines.

2.3.3 I/O Control Logic

The I/O control logic directs data transactions between the LSI-11 and the interface module. A data transaction can be a word or a byte, a high byte or a low byte, an input or an output, or status information or character information. The I/O control logic monitors the LSI-11 bus lines to recognize what type of transaction is to be accomplished. It uses this information to control four device registers. The registers are named after their functions as follows:

Receiver Control/Status Register	(RCSR)
Transmitter Control/Status Register	(XCSR)
Receiver Buffer	(RBUF)
Transmitter Buffer	(XBUF)

These four registers are described in subsequent paragraphs of this chapter.

An I/O operation begins with the LSI-11 addressing the interface module. The bus interface decodes the address, asserts MATCH H to the I/O control logic, and places the address on the three-state bus lines. The I/O control logic decodes the three least significant bits of the three-state bus lines (DAT00 H through DAT02 H) and the LSI-11 bus control signals. The circuit develops register selection and byte selection signals to enable the correct data paths between the computer and the appropriate device register. It also controls INWD L, which determines whether the bus interface transceivers are transmitting or receiving. When data becomes available, the I/O control logic gates it to its destination (from the LSI-11 bus to the three-state bus for an output transfer, or from the three-state bus to the LSI-11 bus for an input transfer).

2.3.4 Control/Status Registers

The DLV11-E and DLV11-F each have two control/status registers: the RCSR and the XCSR. The computer writes control bits out of these registers and reads status bits in from them. The registers consist of a series of latches, data selectors, and gating circuitry. During data transactions involving control and status information, the I/O control logic enables the XCSR or RCSR to either latch in control bits or gate out status bits.

When status information is to be read into the computer, the LSI-11 addresses the device register containing the desired information. The bus interface and I/O control logic decode the address and enable the contents of the selected register to be placed on the bus and transferred into the computer. When control information is to be written out to the interface modules, the computer addresses the device register that is to be loaded. The bus interface and I/O control logic decode the address and enable the register to load the control information when it is placed on the bus.

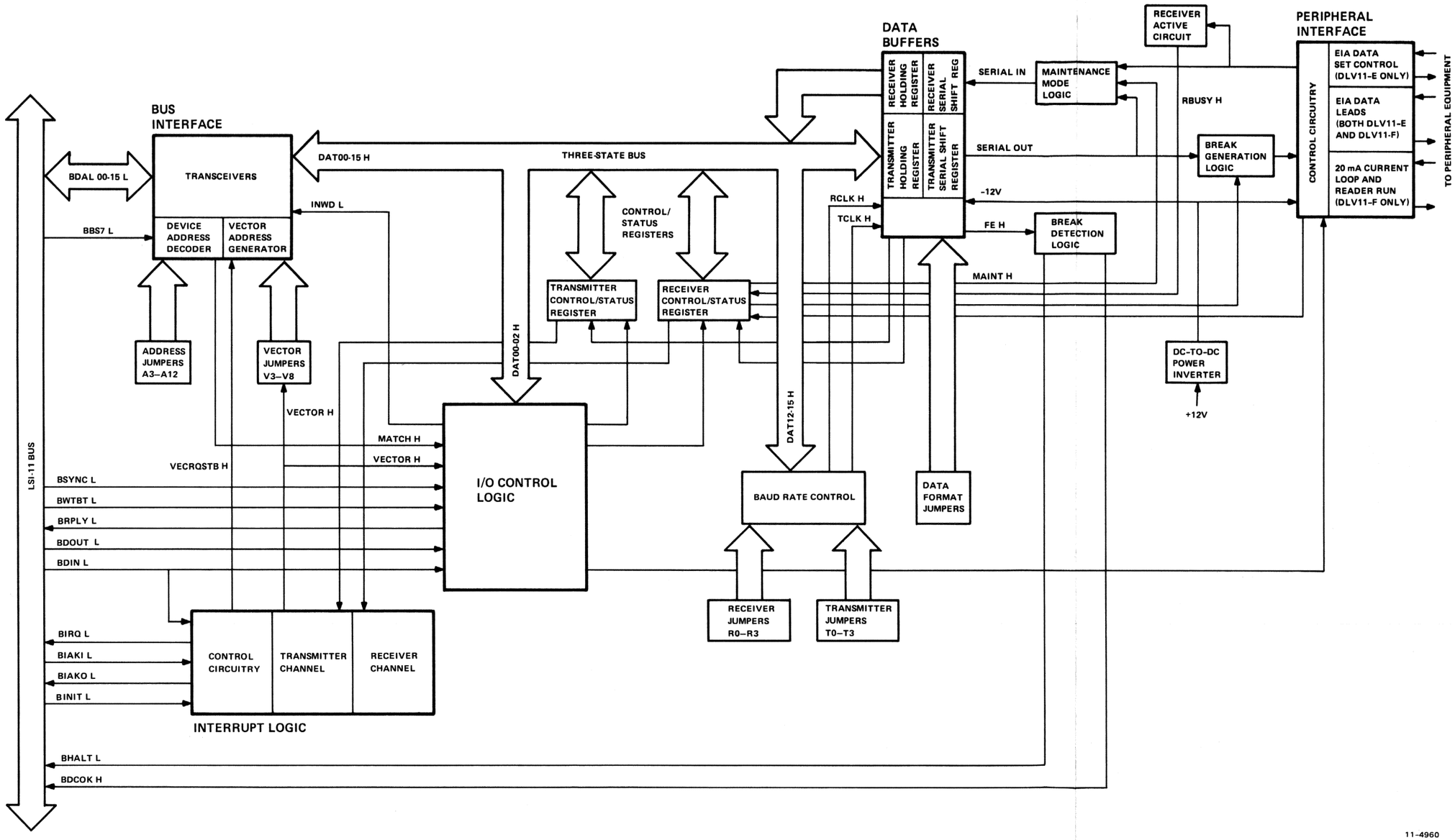


Figure 2-3 DLV11-E and DLV11-F Functional Block Diagram

Not all control and status bits are both read and write; some are read-only bits and some are write-only bits. A detailed description of each bit is given with the programming information in Chapter 4.

2.3.5 Data Buffers

The DLV11-E and the DLV11-F each have two data buffers: one for receive data (RBUF) and one for transmit data (XBUF). Both data buffers handle data by bytes. The RBUF also holds error flag bits pertaining to the status of the received data.

The data received from the peripheral device is transferred serially from the peripheral interface circuit into a receive shift register in the data buffer. From there it is transferred in parallel to a holding register. At the appropriate time, the buffer control circuitry places the parallel data, along with error information, onto the module's internal three-state bus. The bus interface then transfers the data to the computer.

Data to be transmitted to the peripheral device is taken off the three-state bus in parallel by the XBUF and then shifted serially out to the peripheral interface circuit.

Both the RBUF and the XBUF provide "double-buffering" of the data. The buffering is double in that the circuits each have both a serial shift register and a parallel holding register. This allows one character to be held while another is being moved into or out of the buffer.

2.3.6 Receiver Active Circuit

The receiver active circuit monitors the serial received data line from the peripheral interface and a receiver done (RDONE H) status bit from the RBUF. The circuit generates a busy signal (RBUSY H) to indicate that the receiver is active. This signal sets the RCVR ACT bit in the RCSR.

2.3.7 Interrupt Logic

When a peripheral device interfaced by a DLV11-E or DLV11-F needs service, the module can, if enabled, interrupt the computer program and vector to a service routine. The interrupt logic can initiate two types of interrupts: a receiver interrupt and a transmitter interrupt. These interrupts are handled through separate receiver and transmitter channels.

For an interrupt transaction to occur, first the program sets the interrupt enable bit in the control/status register. Next, the interrupt logic recognizes the condition requiring service and asserts the interrupt request line (BIRQ L) to the computer. When the interrupt is acknowledged by the computer, the interrupt logic enables the bus interface to place the vector on the bus lines.

There are two vectors: one for a receiver interrupt and one for a transmitter interrupt. The interrupt logic uses VECRQSTB H to indicate which vector is enabled.

The LSI-11's interrupt acknowledge signal (BIAKI L/BIAKO L) is daisy-chained through the devices on the LSI-11 bus. A device's priority is established by its position in the interrupt acknowledge daisy-chain. The interrupt acknowledge chain goes through both the receiver section and the transmitter section of the module's interrupt logic. It goes through the receiver section first, thereby giving the receiver channel priority over the transmitter channel.

A receiver interrupt is initiated when the RBUF has received and assembled a character of data and is ready to transfer it to the computer. A transmitter interrupt is initiated when the XBUF's holding register is empty and is ready for another data input from the computer.

The DLV11-E differs from the DLV11-F in that it recognizes a second condition requiring a receiver interrupt. The DLV11-E initiates a receiver interrupt when the data set that it is interfacing signals for a handshake. The computer program can read the DLV11-E's RCSR to determine whether the receiver interrupt is for a handshake or for another character of data.

2.3.8 Baud Rate Control

The baud rate control circuit generates clock signals that control the speeds at which the RBUF and XBUF move serial data. The circuit can provide a common clock to both data buffer circuits (common speed operation) or separate transmit and receive clocks (split speed operation).

In common speed operation, both transmit and receive baud rates are either set by wire wrap jumpers R0 through R3 or programmable by three-state bus lines DAT12 H through DAT15 H. In split speed operation, the transmit baud rate is set by jumpers T0 through T3, while the receive baud rate remains under the control of either R0 through R3 or the computer program.

Should it be desired to use a baud rate not available from the baud rate control's crystal-controlled clock generator, the module has provisions for external inputs for both the transmit and receive clocks.

2.3.9 Break Logic

A BREAK signal is a continuous spacing condition on the serial data line. The DLV11-E and DLV11-F can receive BREAK signals from a peripheral device (normally the console device) and can transmit BREAK signals to a peripheral device (normally another processor). Either operation can be enabled or inhibited by wire wrap jumpers.

When the interface module receives a BREAK signal from the serial data line, it interprets the absence of STOP bits as a framing error. It can respond to this apparent error (or to an actual error) in one of three ways:

1. It can ignore it the apparent error.
2. It can place the LSI-11 in the HALT mode.
3. It can cause the LSI-11 to re-boot.

Which action the module takes is controlled by wire wrap jumpers. To place the computer in the HALT mode, the break logic asserts BHALT L. To cause the computer to reload a bootstrap, the break logic negates BDCOK H. Refer to Paragraph 5.9 for further information.

2.3.10 Maintenance Mode Logic

The DLV11-E and DLV11-F have a maintenance mode for verifying the operation of the modules' data paths up to (but not including) the peripheral interface circuitry. This mode is controlled by the computer program, but is used only for checking the interface module, not the computer. In maintenance mode, data from the computer is transferred from the bus interface to the XBUF and serialized, as in normal operation. But then, in addition to going to the peripheral interface circuit, a sample of the XBUF's serial output is also routed back to the RBUF's serial input. There it is converted to parallel, placed on the three-state bus to the bus interface, and transferred back into the computer. The program can then compare the received data with the transmitted data to check for errors.

2.3.11 DLV11-E Peripheral Interface

The peripheral interface circuitry converts the DLV11-E's data and modem control signals from TTL levels to EIA-standard bipolar levels for the peripheral device. Likewise, it converts the peripheral's data and control lines from EIA levels to TTL levels for the interface module.

The circuit can receive four modem control signals (RING, CARRIER, CLEAR TO SEND, and SECONDARY RECEIVED DATA) and can transmit four modem control signals (DATA TERMINAL READY, REQUEST TO SEND, FORCE BUSY, and SECONDARY TRANSMITTED DATA). The control signals are routed through the control/status registers. The interrupt logic uses the received control signals to initiate data set interrupts. The program uses the transmitted control signals to perform handshakes with the data set. Refer to Paragraph 5.11 for an example of a handshake sequence.

2.3.12 DLV11-F Peripheral Interface

The DLV11-F peripheral interface operates in one of two possible modes:

1. EIA Data Leads Only – This type of operation supports terminals that use EIA levels, but do not require control signal interaction.

2. 20 mA Current Loop – This operation supports terminals that use either active or passive current loops. It also controls the paper tape reader on DIGITAL-modified TTY units that have a reader run relay.

When interfacing EIA-level equipment, the module performs the TTL-to-EIA and EIA-to-TTL level conversion on the transmit and receive data leads only. During data leads only operation, the module does not monitor incoming control signals. Outgoing control signals (REQUEST TO SEND, FORCE BUSY, and DATA TERMINAL READY) are held by driver circuits in a continuous TRUE condition.

When the DLV11-F interfaces a 20 mA current loop peripheral device, it can be jumpered to operate in either active or passive configuration. In the active configuration, the peripheral interface supplies the current for the loop; in the passive configuration, the current is supplied by the peripheral device. In either case, the receive data line from the peripheral is optically isolated from the DLV11-F's internal data path.

The 20 mA current loop transmitter operates in either the active or passive configuration. The transmit data lines are optically isolated from the DLV11-F's internal data path only in the passive configuration.

A Reader Run signal is produced for a peripheral device that has a reader run relay. When enabled by the program, the peripheral interface circuit supplies current to the relay, causing the reader to advance the paper tape.

2.3.13 DC-to-DC Power Inverter

Both the DLV11-E and DLV11-F need -12 V for the data buffers and the peripheral interface. This voltage is produced on the module by a small power inverter. The inverter uses the +12 V power available on the LSI-11 backplane to produce a regulated -12 V for the data buffers and peripheral interface circuits.

CHAPTER 3
INSTALLATION

3.1 GENERAL

This chapter describes the jumper configuration, the installation requirements, and the checkout of the DLV11-E and DLV11-F asynchronous line interface modules. The wire wrap jumper functions are defined and application examples are presented. Wire wrapping instructions are presented in Appendix B.

3.2 CONFIGURATION

Before installing the module, ensure that it is configured for your application. The jumper locations are depicted in Figures 3-1 and 3-2. Their functions are defined in Tables 3-1, 3-2, and 3-3. Table 3-4 explains the configuration in which the modules are shipped from the factory. Table 3-5 lists common applications of the DLV11-E and DLV11-F; Figures 3-3 and 3-4 illustrate examples of typical cabling requirements.

The DLV11-F is shipped from the factory with capacitor C29 installed (Figure 3-2). This capacitor is provided for applications using Teletype® terminals. For applications using DIGITAL terminals, remove capacitor C29.

3.3 MODULE INSTALLATION

The DLV11-E or DLV11-F module can be installed in any slots in the LSI-11 backplane, except the first four slots (the LSI-11 processor always occupies the first slots). Do not leave any unused option locations between the processor and the DLV11-E or DLV11-F. An open slot would break the interrupt acknowledge daisy chain. The priority of the module is determined by its proximity to the processor on the bus (refer to Figure 3-5). The closer the slot is to the processor module, the higher the interface module's priority.

Determine the appropriate slot for the module. For example, if a DLV11-E is interfacing communications lines from a host computer, it would normally be placed in the slot closest to the processor module, followed by the module interfacing the console terminal. Refer to *Microcomputer Handbook* (DIGITAL part number EB 06583 76) for system considerations.

®Teletype is a registered trademark of Teletype Corporation.

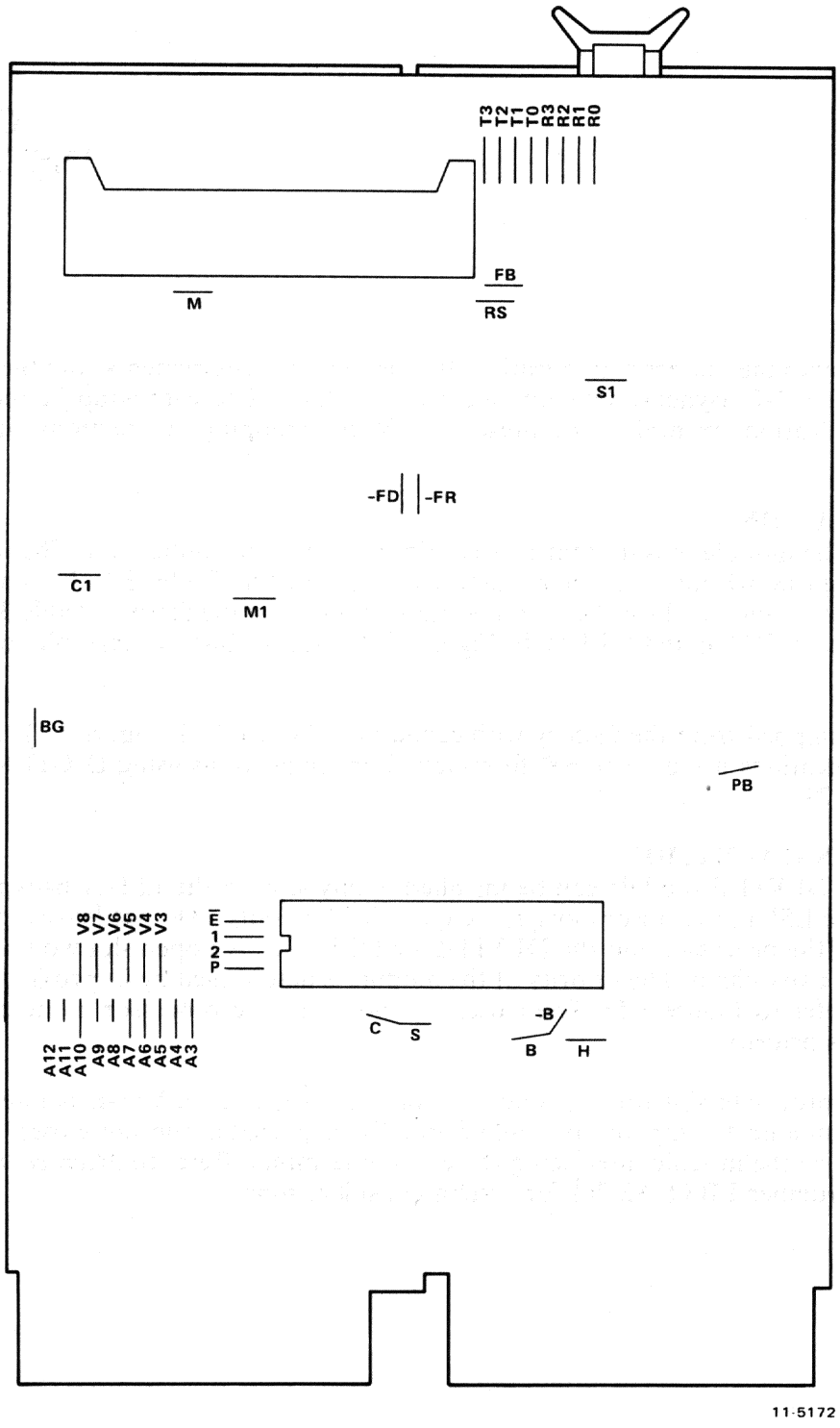


Figure 3-1 DLV11-E Jumper Locations

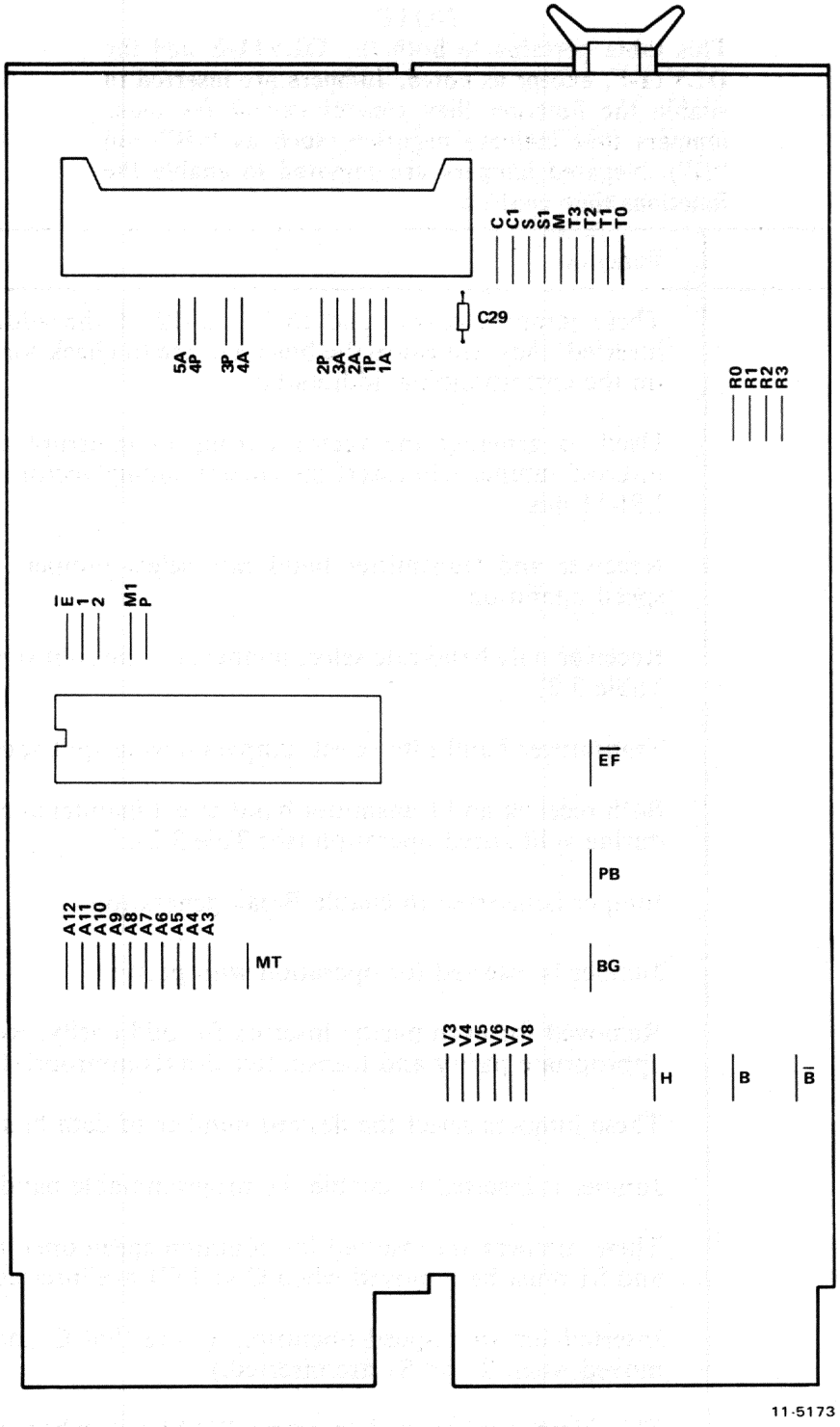


Figure 3-2 DLV11-F Jumper Locations

Table 3-1 Jumper Definitions

NOTE
This table pertains to both the DLV11-E and the DLV11-F, except as noted. Jumpers are inserted to enable the function they control except for those jumpers that indicate negation (such as "-B" and "$\overline{B}$"). Negated jumpers are removed to enable the functions they control.

Jumper	Function
A3–A12	These jumpers correspond to bits 3–12 of the address word. When inserted, they will cause the bus interface to check for a True condition on the corresponding address bit.
V3–V8	Used to generate the vector during an interrupt transaction. Each inserted jumper will assert the corresponding vector address bit on the LSI-11 bus.
R0–R3	Receiver and transmitter baud rate select jumpers, during common speed operation. Receiver only baud rate select jumpers during split speed operation (see Table 3-2).
T0–T3	Transmitter baud rate select jumpers during split speed operation. Both receiver and transmitter baud rate if maintenance mode is entered during split speed operation (see Tale 3-2).
BG	Jumper is inserted to enable Break generation.
P	Jumper is inserted for operation with parity.
$\overline{E}$	Removed for even parity; inserted for odd parity. Receiver checks for appropriate parity and transmitter inserts appropriate parity.
1, 2	These jumpers select the desired number of data bits (see Table 3-3).
PB	Jumper is inserted to enable the programmable baud rate capability.
C, C1	These jumpers are inserted for common speed operation. (Note that S and S1 must be removed when C and C1 are inserted.)
S, S1	Inserted for split speed operation. (Note that C and C1 must be removed when S and S1 are inserted.)
H	This jumper is inserted to assert BHALT L when a framing error is received, except when the Maintenance bit is set. This places the LSI-11 in the halt mode.

Table 3-1 Jumper Definitions (Cont)

Jumper	Function
B, -B (DLV11-E) B, $\overline{B}$ (DLV11-F)	Jumper B is inserted to negate BDCOK H when a BREAK signal or framing error is received, except when the Maintenance bit is set. This causes the LSI-11 to reload the bootstrap. (Jumper -B or $\overline{B}$ must be removed when B is inserted.)
-FD (DLV11-E only)	Jumper is removed to force DATA TERMINAL READY signal on.
-FR (DLV11-E only)	Jumper is removed to force REQUEST TO SEND signal on.
RS (DLV11-E only)	This jumper is inserted to enable normal transmission of the REQUEST TO SEND signal.
FB (DLV11-E only)	Inserted to enable transmission of the FORCE BUSY signal (for Bell model 103E data sets).
1A, 2A, and 3A (DLV11-F only)	These three jumpers are inserted to make the 20 mA current loop receiver active. (Jumpers 1P and 2P must be removed when 1A, 2A, and 3A are inserted.)
1P, 2P (DLV11-F only)	These jumpers are inserted to make the 20 mA current loop receiver passive. (Jumpers 1A, 2A, and 3A must be removed when 1P and 2P are installed.)
4A, 5A (DLV11-F only)	Inserted to make the 20 mA current loop transmitter active. (Jumpers 3P and 4P must be removed when 4A and 5A are inserted.)
3P, 4P (DLV11-F only)	Inserted to make the 20 mA current loop transmitter passive. (Jumpers 4A and 5A must be removed when 3P and 4P are inserted.)
$\overline{EF}$ (DLV11-F only)	Jumper is removed to enable the error flags to be read in the high byte of the Receiver Buffer.
MT (DLV11-F only)	When inserted, enables maintenance bit.
M, M1	These are test jumpers used during the manufacture of the module. They are not defined for field use.

Table 3-2 Baud Rate Selections

	Bit	Bit	Bit	Bit	Bit	
Program Control	15	14	13	12	11*	
Receive Jumpers	R3	R2	R1	R0		Baud
Transmit Jumpers	T3	T2	T1	T0		Rate
	I	I	I	I		50
	I	I	I	R		75
	I	I	R	I		110
	I	I	R	R		134.5
	I	R	I	I		150
	I	R	I	R		300
	I	R	R	I		600
	I	R	R	R		1200
	R	I	I	I		1800
	R	I	I	R		2000
	R	I	R	I		2400
	R	I	R	R		3600
	R	R	I	I		4800
	R	R	I	R		7200
	R	R	R	I		9600

I = Jumper Inserted = Program Bit Cleared.

R = Jumper Removed = Program Bit Set.

*Bit 11 of the XCSR (Write Only Bit) must be set in order to select a new baud rate under program control. Also, jumper PB must be inserted to enable baud rate selection under program control.

Table 3-3 Data Bit Selections

Jumpers		Number of Data Bits
2	1	
I	I	5
I	R	6
R	I	7
R	R	8

Table 3-4 Jumper Configuration When Shipped

Jumper Designation	Jumper State DLV11-E	Jumper State DLV11-F	Function Implemented
A3	R	I	Jumpers A3 through A12 implement device address 17561X for the DLV11-E and 17756X for the DLV11-F. The least significant octal digit is hardwired on the module to address the four device registers as follows: X = 0 RCSR; X = 2 RBUF; X = 4 XCSR; X = 6 XBUF
A4	I	R	
A5	I	R	
A6	I	R	
A7	R	I	
A8	R	R	
A9	R	R	
A10	I	R	
A11	R	R	
A12	R	R	
V3	I	I	This jumper selection implements interrupt vector address 300_8 for receiver interrupts and 304_8 for transmitter interrupts on the DLV11-E. On the DLV11-F it selects 60_8 for receiver interrupts and 64_8 for transmitter interrupts.
V4	I	R	
V5	I	R	
V6	R	I	
V7	R	I	
V8	I	I	
R0	I	I	The module is configured to receive at 110 baud.
R1	R	R	
R2	I	I	
R3	I	I	
T0	I	I	The transmitter is configured for 9600 baud if split speed operation is used.
T1	R	R	
T2	R	R	
T3	R	R	
BG	I	I	Break generation is enabled.
P	R	R	Parity bit is disabled.
$\overline{E}$	R	R	Parity type is not applicable when P is removed.
1	R	R	Operation with 8 data bits per character.
2	R	R	
PB	R	R	Programmable baud rate function disabled.
C	I	I	Common speed operation enabled.
C1	I	I	

Table 3-4 Jumper Configuration When Shipped (Cont)

Jumper Designation	Jumper State		Function Implemented
	DLV11-E	DLV11-F	
S	R	R	Split speed operation disabled.
S1	R	R	
H	R	I	Halt on framing error disabled on DLV11-E; enabled on DLV11-F.
B	R	R	Boot on framing error disabled.
-B	I	N/A	
$\overline{B}$	N/A	I	
-FD	I	N/A	The DATA TERMINAL READY signal is not forced continuously True.
-FR	I	N/A	The REQUEST TO SEND signal is not forced continuously True.
RS	I	N/A	The circuitry controlling the REQUEST TO SEND signal is enabled.
FB	R	N/A	The FORCE BUSY signal is disabled.
1A	N/A	I	The 20 mA current loop receiver is configured as an active receiver.
2A	N/A	I	
3A	N/A	I	
1P	N/A	R	
2P	N/A	R	
4A	N/A	I	The 20 mA current loop transmitter is configured for active operation.
5A	N/A	I	
3P	N/A	R	
4P	N/A	R	
$\overline{EF}$	N/A	I	Error flags are enabled on DLV11-E; disabled on DLV11-F.
M	R	R	Factory test jumpers. Not defined for field use.
M1	R	R	
MT	N/A	R	Maintenance bit disabled.

Table 3-5 Module Application Examples

Module/Mode	Equipment Supported
DLV11-E Modem Control	Bell Data Sets, Models: 103 202C 202D 212A
DLV11-F EIA Data Leads Only	Bell Model 103 Data Set (in automode). Teletype Model 37 Teletypewriter
DLV11-F 20 mA Current Loop	Teletype Model 33 and 35 Teletypewriters DIGITAL equipment: LA36 DECwriter (read/write) LA35 DECwriter (read only) VT05B Alphanumeric Terminal VT50 DECscope (12 line) VT52 DECscope (24 line) RT02 Alphanumeric Terminals DF01-A Acoustic Telephone Coupler LT33 Teletypewriter LT35 Teletypewriter

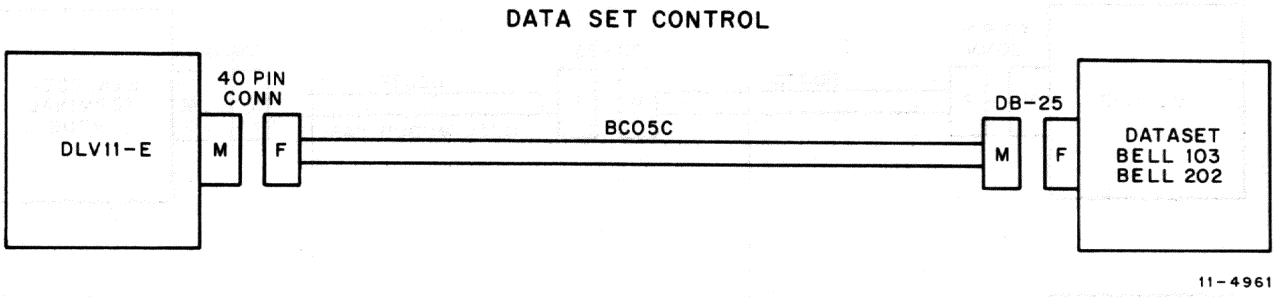


Figure 3-3 DLV11-E Cabling Example

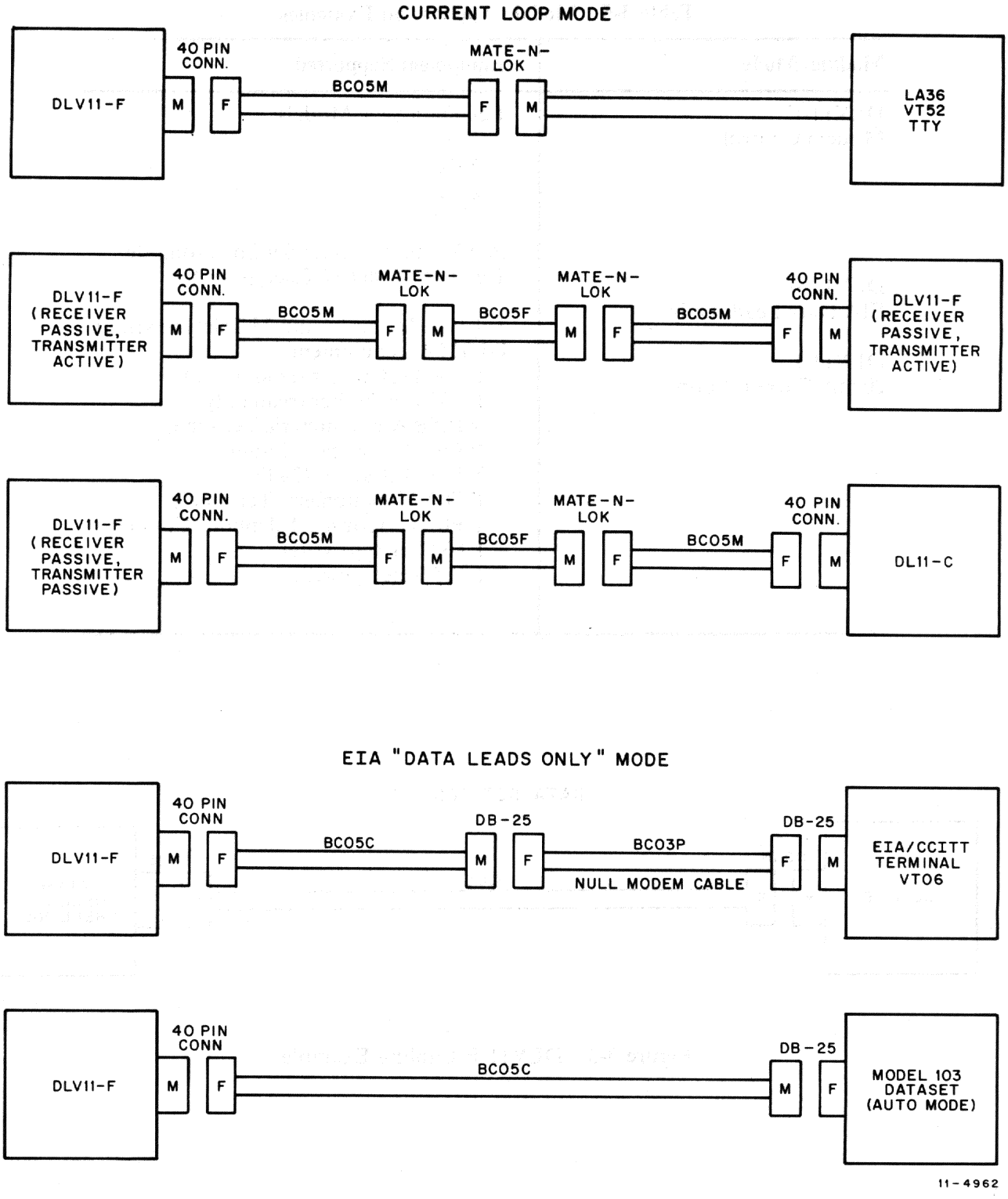


Figure 3-4 DLV11-F Cabling Examples

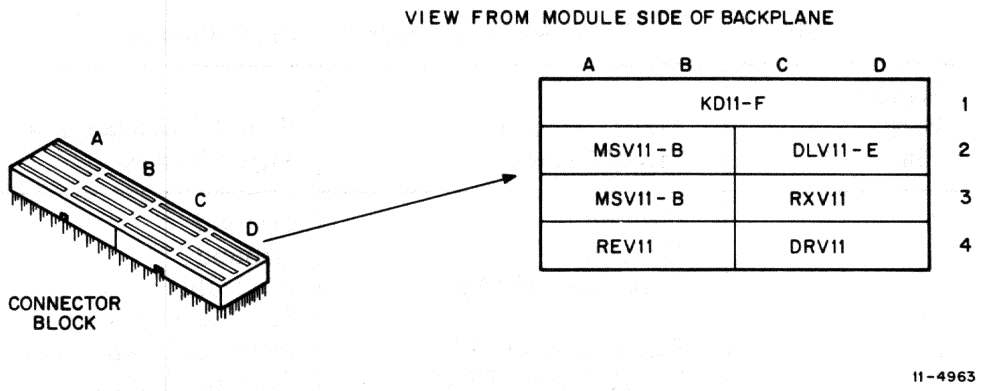


Figure 3-5 Typical Backplane Configuration

After the module has been configured properly and the desired location determined, install it in the computer as follows:

CAUTION

DC power must be removed from the backplane during module insertion and removal.

The module and backplane connector block may be damaged if the module is plugged in backwards.

1. Position the module so that the components side is facing row 1.
2. Slide the module into its slot, taking care that the module fingers mesh correctly with the backplane connector block.
3. Press the module into the connector block, making sure that the deep notch on the module seats against the connector block rib.
4. Next, plug the interface cable into the module's 40-pin header connector.

When the other end of the interface cable is installed, the module can be powered up and checked out. Interface cable installations are shown in Figures 3-3 and 3-4. Interface connector pinning is listed in Tables 3-6 and 3-7. Bus connector pinning is listed in Table 3-8.

3.4 MODULE CHECKOUT

A diagnostic program is shipped with the module, and should be run to verify the proper operation of the module. The program runs on an LSI-11 with the most basic options. Perform the diagnostic checkout as explained in Paragraph 3.4.1 or 3.4.2. If a malfunction is detected, contact the nearest DIGITAL Field Service office.

Table 3-6 DLV11-E 40-Pin Header Connector Pinning

Header Berg Pin	M8017 Module Signal Names	BC05C Modem Cable Signal Names
A	Ground	Ground
B	Ground	Ground
C	Force Busy (EIA)	Force Busy
D		Sec. Clear to Send
E	Serial Input (TTL)	Interlock In *
F	Serial Output (EIA)	Transmitted Data
H		
J	Serial Input (EIA)	Received Data
K		
L		External Clock
M	EIA Interlock	Interlock Out
N		Serial Clock XMIT
P		Sec Request to Send
R		Serial Clock RCVR
S		
T	Clear to Send (EIA)	Clear to Send
U		
V	Request to Send (EIA)	Request to Send
W		- Power
X	Ring (EIA)	Ring
Y		+ Power
Z		Data Set Ready
AA		
BB	Carrier (EIA)	Carrier
CC	External Clock Input (TTL)	
DD	Data Terminal RDY (EIA)	Data Terminal RDY
EE		
FF	Secondary XMIT (EIA)	202 Sec XMIT
HH	External Clock ENB (TTL)	
JJ	Secondary Rec (EIA)	202 Sec RCVR
KK		
LL		EIA Sec XMIT
MM		Signal Quality
NN		EIA Sec RCVR
PP		
RR		
SS	Serial Output (TTL)	
TT	+5 V	
UU	Ground	Ground
VV	Ground	Ground

*This jumper is built into the cable.

Table 3-7 DLV11-F 40-Pin Header Connector Pinning

Header Pin	M8028 Module Signal Names	BC05C Modem Cable Signal Names	BC05M 20 mA Cable
A	Ground	Ground	Ground
B	Ground	Ground	
C	Force Busy (EIA)	Force Busy	
D		Sec Clear to Send	
E	Serial Input (TTL)	Interlock In *	Interlock In *
F	Serial Output (EIA)	Transmitted Data	
H	20 mA Interlock		Interlock Out
J	Serial Input (EIA)	Received Data	
K	Serial Input + (20 mA)		Received Data +
L		External Clock	
M	EIA Interlock	Interlock Out	
N		Serial Clock XMIT	
P		Sec Request to Send	
R		Serial Clock RCVR	
S	Serial Input - (20 mA)		Received Data -
T		Clear to Send	
U			
V	Request to Send (EIA)	Request to Send	
W		- Power	
X		Ring	
Y		+ Power	
Z		Data Set Ready	
AA	Serial Output+(20 mA)		Transmitted Data+
BB		Carrier	
CC	Ext. Clock Input (TTL)		
DD	Data Terminal RDY (EIA)	Data Terminal RDY	
EE	Reader Run - (20 mA)		Reader Run -
FF		202 Sec XMIT	
HH	Ext. Clock Enb (TTL)		
JJ		202 Sec RCVR	
KK	Serial Output		
LL		EIA Sec XMIT	
MM		Signal Quality	
NN		EIA Sec RCVR	
PP	Reader Run+(20 mA)		Reader Run+
RR		Signal Rate	
SS	Serial Output (TTL)		
TT	+5V		
UU	Ground	Ground	Ground
VV	Ground	Ground	Ground

Table 3-8 DLV11-E and DLV11-F Edge Connector Pinning

Mnemonic	Pin	
+5	AA2	
	BA2	
+12	AD2	
BBS7 L	AP2	
BDAL 0 L	AU2	
BDAL 1 L	AV2	
BDAL 2 L	BE2	
BDAL 3 L	BF2	
BDAL 4 L	BH2	
BDAL 5 L	BJ2	
BDAL 6 L	BK2	
BDAL 7 L	BL2	
BDAL 8 L	BM2	
BDAL 9 L	BN2	
BDAL 10 L	BP2	
BDAL 11 L	BR2	
BDAL 12 L	BS2	
BDAL 13 L	BT2	
BDAL 14 L	BU2	
BDAL 15 L	BV2	
BDIN L	AH2	
BDOUT L	AE2	
BHALT L	AP1	
BIAK I L*	AM2	
BIAK 0 L*	AN2	
BINIT L	AT2	
BDMGIL*	AR2	
BDMG0L*	AS2	
BIRQ L	AL2	
BRPLY L	AF2	
BSYNC L	AJ2	
BDC OK H	BA1	
GND	AC2	
GND(	AT1	
GND	BC2	
GND	BT1	
MSPARE A (–12 V)	AK1	**
	AL1	
MSPARE B (EXT R CLK)	BK1	**
	BL1	
SSPARE 4	BC1	
SSPARE 5	BD1	
SSPARE 6	BE1	
SSPARE 7	BF1	
SSPARE 8 (EXT T CLK)	BH1	

*These signals are not bussed, they are daisy-chained.
**This jumper is wired on the backplane.

3.4.1 DLV11-E Checkout

To verify the operation of the DLV11-E, turn off the dc power and remove the interface cable from the data set. Leave the other end connected to the module's header connector. Plug an H315 terminator into the free end of the interface cable. Power up the computer. Load and start MAINDEC-11-DVDVA. When the program has been completed successfully, turn off the dc power and reconnect the interface cable to the data set.

3.4.2 DLV11-F Checkout

The DLV11-F does not require a terminator plug for checkout. Load and start MAINDEC-11-DVDVC. Successful completion of the program indicates the module is acceptable.

CHAPTER 4
PROGRAMMING

4.1 INTRODUCTION

Both the DLV11-E and DLV11-F are program compatible with PDP-11 software. Programs written for PDP-11's using DL11-A through -D interface modules will run on an LSI-11 using a DLV11-F configured for the same application. Programs written for a DL11-E will run with a DLV11-E. Also, the DLV11-F will operate with LSI-11 programs written for the DLV11.

This chapter defines the bits in each of the four device registers, discusses interrupts and timing considerations, and gives programming examples.

4.2 DEVICE REGISTERS

All software control of the DLV11-E or DLV11-F Asynchronous Line Interface is performed by means of four device registers. These registers have been assigned bus addresses and can be read or loaded (with the exceptions noted) using any LSI-11 instruction referring to their addresses. Address assignments can be changed by altering jumpers on the module to correspond to any address within the range of 160000 to 177777. Table 4-1 lists the addresses of the registers when the module is used to interface a console device. The RCSR is at the base address. Each subsequent register is two locations up from the one preceding it.

Table 4-1 Register Addresses for Console Interfacing

Register	Mnemonic	Address
Receiver Control/Status Register	RCSR	177560
Receiver Buffer Register	RBUF	177562
Transmitter Control/Status Register	XCSR	177564
Transmitter Buffer Register	XBUF	177566

The DLV11-E RCSR differs from the DLV11-F RCSR; therefore, the bits for these two RCSRs are defined separately. The DLV11-E and DLV11-F operate identically with respect to the three other device registers. The bit definition for these registers applies to both modules. Figures 4-1 and 4-2 show RCSR bit assignments. Figures 4-3, 4-4, and 4-5 show the RBUF, XCSR, and XBUF, respectively. Tables 4-2 through 4-6 define the bit assignments.

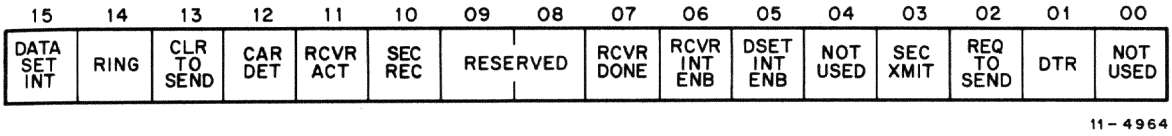


Figure 4-1 DLV11-E RCSR Bit Assignments

Table 4-2 DLV11-E RCSR Bit Assignments

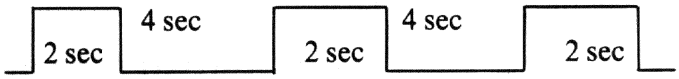
Bit	Name	Meaning and Operation
15	DATA SET INT (Data Set Interrupt)	This bit initiates an interrupt sequence provided the DATA SET INT ENB bit (05) is also set. This bit is set whenever CAR DET, CLR TO SEND, or SEC REC changes state; i.e., on a 0-to-1 or 1-to-0 transition of any one of these bits. It is also set when RING changes from 0 to 1. Cleared by INIT or by reading the RCSR. Because reading the register clears the bit, it is, in effect, a "read-once" bit.
14	RING	When set, indicates that a RINGING signal is being received from the dataset. Note that the RINGING signal is not a level but an EIA control with the cycle time as shown below:  Read-only bit.
13	CLR TO SEND (Clear to Send)	The state of this bit is dependent on the state of the CLEAR TO SEND signal from the data set. When set, this bit indicates an ON condition; when clear, it indicates an OFF condition. Read-only bit.
12	CAR DET (Carrier Detect)	This bit is set when the data carrier is received. When clear, it indicates either the end of the current transmission activity or an error condition. Read-only bit.
11	RCVR ACT (Receiver Active)	When set, this bit indicates that the DLV11-E's receiver is active. The bit is set at the center of the START bit, which is the beginning of the input serial data from the device, and is cleared by the leading edge of R DONE H. Read-only bit; cleared by INIT or by R DONE H (bit 07)

Table 4-2 DLV11-E RCSR Bit Assignments (Cont)

Bit	Name	Meaning and Operation
10	SEC REC (Secondary Received or Supervisory Received Data)	This bit provides a receive capability for the reverse channel of a remote station. A space (≈+10 V) is read as a 1. (A transmit capability is provided by bit 03.) Read-only bit.
9–8	Not Used	Reserved for future use.
07	RCVR DONE (Receiver Done)	This bit is set when an entire character has been received and is ready for transfer to the LSI-11. When set, initiates an interrupt sequence provided RCVR INT ENB (bit 06) is also set. Cleared whenever the receiver buffer (RBUF) is addressed. Also cleared by INIT. Read-only bit
06	RCVR INT ENB (Receiver Interrupt Enable)	When set, allows an interrupt sequence to start when RCVR DONE (bit 07) sets. Read/write bit; cleared by INIT. See Note 1.
05	DSET INT ENB (Data Set Interrupt Enable)	When set, allows an inerrupt sequence to start when DATA SET INT (bit 15) sets. Read/write bit; cleared by INIT. See Note 1.
04	Not Used	Reserved for future use.
03	SEC XMIT (Secondary Transmitted or Supervisory Transmitted Data)	This bit provides a transmit capability for a reverse channel of a remote station. When set, transmits a space (≈+10 V). (A receive capability is provided by bit 10.) Read/write bit; cleared by INIT.
02	REQ TO SEND (Request to Send)	A control lead to the data set which is required for transmission. A jumper on the DLV11-E ties this bit to REQ TO SEND or FORCE BUSY in the data set. Read/write bit; cleared by INIT.

Table 4-2 DLV11-E RCSR Bit Assignments (Cont)

Bit	Name	Meaning and Operation
01	DTR (Data Terminal) Ready)	A control lead for the data set communication channel. When set, permits connection to the channel. When clear, disconnects the interface from the channel. Read/write bit; must be cleared by the program, is not cleared by INIT. (See Note 2.)

NOTES

1. **When clearing an interrupt enable bit, first set the processor to its highest priority [Processor Status Word (PSW) bit 7 = 1]. After the interrupt enable bit is cleared, the processor may be returned to its normal priority (PSW bit 7 = 0).**

   For example:

   ```
   MTPS #200
   BIC #100, CSR
   MTPS #0
   EXIT
   ```

   For further information refer to Paragraph 4.6.

2. **The state of this bit is not defined after power-up.**

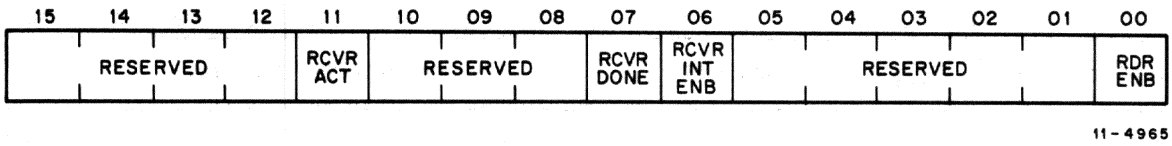


Figure 4-2 DLV11-F RCSR Bit Assignments

Table 4-3 DLV11-F RCSR Bit Assignments

Bit	Name	Meaning and Operation
15–12	Not Used	Reserved for future use.
11	RCVR ACT (Receiver Active)	When set, this bit indicates that the DLV11-F interface receiver is active. The bit is set at the center of the START bit, which is the beginning of the input serial data from the device, and is cleared by the leading edge of RDONE H. Read-only bit; cleared by INIT or by RCVR DONE (bit 07).
10–08	Not Used	Reserved for future use.
07	RCVR DONE (Receiver Done)	This bit is set when an entire character has been received and is ready for transfer to the LSI-11 bus. When set, initiates an interrupt sequence provided RCVR INT ENB (bit 06) is also set. Read-only bit; cleared whenever the receiver buffer (RBUF) is addressed or whenever RDR ENB (bit 00) is set. Also cleared by INIT.
06	RCVR INT ENB (Receiver Interrupt Enable)	When set, allows an interrupt sequence to start when RCVR DONE (bit 07) sets. Read/write bit; cleared by INIT.
05-01	Not Used	Reserved for future use.
00	RDR ENB (Reader Enable)	When set, this bit advances the paper-tape reader in DIGITAL-modified TTY units (LT33-C; LT35-A, -C) and clears the RCVR DONE bit (bit 07). This bit is cleared at the middle of the START bit, which is the beginning of the serial input from an external device. Also cleared by INIT. Write-only bit.

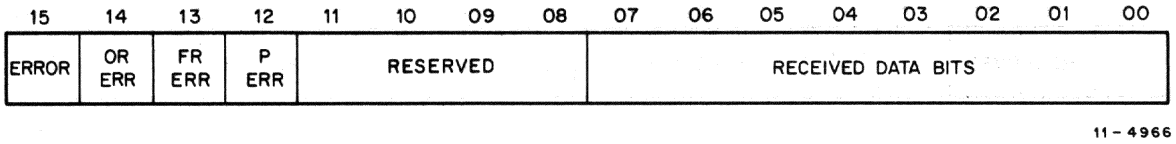


Figure 4-3 DLV11-E and DLV11-F RBUF Bit Assignments

Table 4-4 DLV11-E and DLV11-F RBUF Bit Assignments

Bit	Name	Meaning and Operation
15	ERROR (Error)	Used to indicate that an error condition is present. This bit is the logical OR of OR ERR, FR ERR, and P ERR (bits 14, 13, and 12, respectively). Whenever one of these bits is set, it causes ERROR to set. This bit is not connected to the interrupt logic. Read-only bit; cleared by removing the error-producing condition.
		NOTE **Error indications remain present until the next character is received, at which time the error bits are updated. INIT clears the error bits.**
14	OR ERR (Overrun Error)	When set, indicates that reading of the previously received character was not completed (RCVR DONE not cleared) prior to receiving a new character. Read-only bit. Cleared by INIT.
13	FR ERR (Framing Error)	When set, indicates that the character that was read had no valid STOP bit. Read-only bit. Cleared by INIT.
12	P ERR (Parity Error)	When set, indicates that the parity received does not agree with the expected parity. This bit is always 0 if no parity is selected. Read-only bit. Cleared by INIT.
11–08	Not Used	Reserved for future use.
07–00	RECEIVED DATA BITS	Holds the character just read. If less than eight bits are selected, then the buffer is right-justified into the least significant bit positions. In this case, the higher unused bit or bits are read as 0's. Read-only bits; not cleared by INIT.

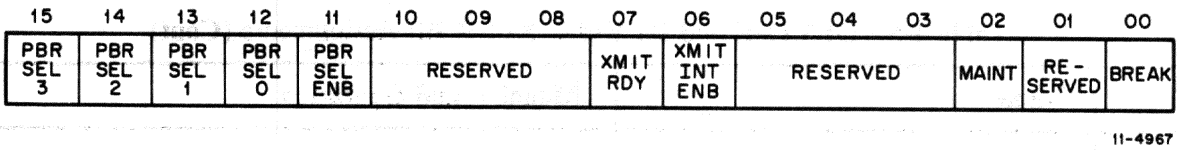


Figure 4-4 DLV11-E and DLV11-F XCSR Bit Assignments

Table 4-5 DLV11-E and DLV11-F XCSR Bit Assignments

Bit	Name	Meaning and Operation
15–12	PBR SEL (Programmable Baud Rate Select)	When set, these bits choose a baud rate from 50–9600 baud. See Table 3-2. Write-only bits.
11	PBR ENB (Programmable Baud Rate Enable)	This bit must be set in order to select a new baud rate indicated by bits 12 to 15. Write-only bits.
10–08	Not Used	Reserved for future use.
07	XMIT RDY (Transmitter Ready)	This bit is set when the transmitter buffer (XBUF) can accept another character. When set, it initiates an interrupt sequence provided XMIT INT ENB (Bit 06) is also set. Read-only bit; set by INIT.
06	XMIT INT ENB (Transmitter Interrupt Enable)	When set, allows an interrupt sequence to start when XMIT RDY (bit 07) is set. Read/write bits; cleared by INIT. See Note.
05–03	Not Used	Reserved for future use.
02	MAINT	Used for maintenance function. When set, connects the transmitter serial output to the receiver serial input while disconnecting the external device from the receiver serial input. It also forces the receiver to run at transmitter baud rate speed when split speed operation is enabled. Read/write bit; cleared by INIT

Table 4-5 DLV11-E and DLV11-F XCSR Bit Assignments (Cont)

Bit	Name	Meaning and Operation
01	Not Used	Reserved for future use.
00	BREAK	When set, transmits a continuous space to the external device. Read/write bit; cleared by INIT.

NOTE

When clearing an interrupt enable bit, first set the processor to its highest priority (PSW bit 7 = 1). After the interrupt enable bit is cleared, the processor may be returned to its normal priority (PSW bit 7 = 0). For example:

```
MTPS #200
BIC #100, CSR
MTPS #0
EXIT
```

For further information refer to Paragraph 4.6.

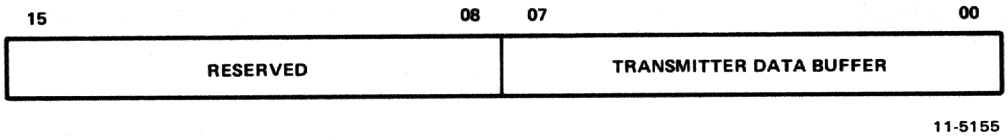


Figure 4-5 DLV11-E and DLV11-F XBUF Bit Assignments

Table 4-6 DLV11-E and DLV11-F XBUF Bit Assignments

Bit	Name	Meaning and Operation
15–08	Not Used	Not defined. Not necessarily read as 0s.
07–00	TRANSMITTER DATA BUFFER	Holds the character to be transferred to the external device. If less than eight bits are used, the character must be loaded so that it is right-justified into the least significant bits. Write-only bits. Not necessarily read as 0s.

The unused and load-only bits are always read as 0's except for the XBUF, in which unused bits are undefined. Loading unused or read-only bits has no effect on the bit position. The mnemonic INIT refers to the initialization signal issued by the processor. Initialization is caused by one of the following: issuing a programmed RESET instruction; pressing "G" while in ODT; or the occurrence of a power-up or power-down condition of the processor power supply.

In the following descriptions, "transmitter" refers to those registers and bits involved in accepting a parallel character from the LSI-11 for serial transmission to the external device; "receiver" refers to those registers and bits involved with receiving serial information from the external device for parallel transfer to the LSI-11.

4.3 INTERRUPTS

Both the DLV11-E and the DLV11-F have two interrupt channels: one for receiver interrupts and one for transmitter interrupts. These two channels operate independently. If, however, simultaneous interrupt requests occur, the receiver channel has priority over the transmitter channel.

In both the DLV11-E and the DLV11-F, a transmitter interrupt can occur only if the interrupt enable bit (XMIT INT ENB) in the XCSR is set. With XMIT INT ENB set, setting the transmitter ready (XMIT RDY) bit initiates an interrupt request. When XMIT RDY is set, it indicates that the XBUF is empty and ready to accept another character from the bus for transfer to the external device.

A receiver data interrupt can occur only if the interrupt enable (RCVR INT ENB) bit in the receiver RCSR is set. With RCVR INT ENB set, setting the receiver done (RCVR DONE) bit initiates an interrupt request. When RCVR DONE is set, it indicates that an entire character has been received and is ready for transfer to the bus. The receiver data interrupt occurs in both the DLV11-E and the DLV11-F. The DLV11-E also has a data set interrupt.

The receiver portion of the DLV11-E handles multisource interrupts. One of the receiver interrupt circuits is activated by RCVR INT ENB and RCVR DONE. The other interrupt circuit can cause an interrupt only if the data set interrupt enable bit (DATA SET INT ENB) in the RCSR is set. With DATA SET INT ENB set, setting the DATA SET INT bit initiates an interrupt request. The DATA SET INT bit can be set by any of four other bits: CAR DET, CLR TO SEND, SEC REC, or RING.

NOTE

When servicing a receiver interrupt from the DLV11-E, if a second receiver interrupt condition develops a second interrupt request may not occur. To avoid missing this second interrupt condition, either all possible receiver interrupt conditions should be checked after servicing the first condition, or else both interrupt enable bits (bits 05 and 06) should be cleared upon entry to the service routine and then set at the end of service.

4.4 TIMING CONSIDERATIONS

When programming the DLV11-E or DLV11-F Asynchronous Line Interface, it is important to consider timing of certain functions in order to use the system in the most efficient manner. Timing considerations for the receiver, transmitter, and break generation logic are discussed in the following paragraphs.

4.4.1 Receiver

The RCVR DONE flag (bit 07 in the RCSR) sets when the receiver has assembled a full character. This occurs at the middle of the first STOP bit. Because the receiver is double buffered, data remains valid until the next character is received and assembled. This permits one full character time for servicing the receiver interrupt.

4.4.2 Transmitter

The transmitter is also double buffered. The XMIT RDY flag (bit 07 in the XCSR) is set after initialization. When the XBUF is loaded with the first character from the bus, the flag clears but then sets again within a function of a bit time. A second character can then be loaded, which clears the flag again. The flag then remains cleared for nearly one full character time.

4.4.3 BREAK Generation Logic

When the BREAK bit (bit 00 in the XCSR) is set, it causes transmission of a continuous space. Because the XMIT RDY flag continues to function normally, the duration of a BREAK can be timed by the pseudo-transmission of a number of characters. However, because the transmitter is double buffered, a null character (all 0's) should precede transmission of the BREAK to ensure that the previous character clears the line. In a similar manner, the final pseudo-transmitted character in the BREAK should be null.

4.4.4 System Reset Timing

A system reset should not be performed immediately after the processor loads a character into the transmitter buffer for serial transmission. If the system is reset before the last character has left the transmitter buffer, the character will be lost when the buffer is cleared by INIT. To avoid this, the program should transmit two null characters after the last character, and then wait for XMIT RDY to return to its true state.

NOTE

Programs developed on the DLV-11 Serial Line Unit (M7940) may not include these null characters, since the DLV-11s transmitter buffer is not cleared by the INIT signal.

4.5 PROGRAMMING EXAMPLES

Table 4-7 is an example of a typical program that can be used as an echo program for a DLV11-E interfacing a Bell Model 103 data set. When a remote terminal dials in, this program answers the call and provides a character-by-character echo. Characters are also copied onto the console device.

Figure 4-6 depicts a DLV11-F program. The program demonstrates the flexibility of programming with interrupts. It is performed on a console terminal interfaced by the DLV11-F. The program exercises the module's full-duplex capability.

Table 4-7 DLV11-E Programming Example

```
                000200                          .=200
000200  000167  001616          START:  IMP     BEGIN           ;JUMP TO BEGINNING
                                                                ;OF PROGRAM
                                :SYMBOL DEFINITIONS

                040000          RING=   040000                  ;BIT 14 OF RCSR, RING
                020000          CTS=    020000                  ;BIT 13 OF RCSR,
                                                                ;CLEAR TO SEND
                000200          RDONE=  000200                  ;BIT 07 OF RCSR,
                                                                ;RECEIVER DONE
                000002          DTR=    000002                  ;BIT 01 OF RCSR, DATA
                                                                ;TERMINAL READY
                000200          XRDY=   000200                  ;BIT 07 OF XCSR,
                                                                ;TRANSMITTER READY

                002000                          .=2000
002000  175610          RCSR:   175610                  ;CSR OF RECEIVER
002002  175612          RBUF:   175612                  ;BUF OF RECEIVER
002004  175614          XCSR:   175614                  ;CSR OF TRANSMITTER
002006  175616          XBUF:   175616                  ;BUF OF TRANSMITTER
002010  177564          CXCSR:  177564                  ;CSR OF CONSOLE
                                                        ;TRANSMITTER
002012  177566          CXBUF:  177566                  ;BUF OF CONSOLE
                                                        ;TRANSMITTER
002014  000000          BUFFER: 0                       ;HOLDS CHARACTER
                                                        ;RECEIVED
002016  000000          DELAY:  0                       ;HOLDS DELAY COUNT,
                                                        ;HIGH ORDER
002020  000000                  0                       ;HOLDS DELAY COUNT,
                                                        ;LOW ORDER
                                ;BEGINNING OF ECHO PROGRAM

002022  005077  177752  BEGIN:  CLR     @RCSR            ;START BY INITIALIZING
                                                        ;ALL BITS TO ZERO

002026  032777  040000  177744  LOOP1:  BIT     # RING,@RCSR   ;CHECK FOR INCOMING CALL
002034  001774                  BEQ     LOOP1         ;BRANCH IF PHONE IS NOT
                                                        ;RINGING
002036  052777  000002  177734          BIS     #DTR,@RCSR    ;PHONE IS RINGING, SO
                                                        ;ANSWER WITH DTR
002044  012767  000005  177744          MOV     #5,DELAY      ;SET UP COUNT FOR DELAY
```

Table 4-7 DLV11-E Programming Example (Cont)

```
002052  032777  020000  177720  LOOP2:  BIT     #CTS,@RCSR      ;CHECK FOR CLEAR
                                                                ;TO SEND
002060  001007                          BNE     LOOP3           ;BRANCH IF ON
002062  162767  000001  177730          SUB     #1,DELAY+2      ;CHECK DELAY
002070  005667  177722                  SBC     DELAY           ;DECREMENT A
                                                                ;TWO-WORD INTEGER
002074  001752                          BEQ     BEGIN           ;BRANCH IF WE HAVE
                                                                ;WAITED TOO LONG
002076  000765                          BR      LOOP2           ;BRANCH AND CONTINUE
                                                                ;TO WAIT FOR CTS
002100  032777  020000  177672  LOOP3:  BIT     #CTS,@RCSR      ;IS CHANNEL STILL
                                                                ;ESTABLISHED?
002106  001745                          BEQ     BEGIN           ;BRANCH IF CTS NOT
                                                                ;PRESENT
002110  032777  000200  177662          BIT     #RDONE,@RCSR    ;CHECK FOR RECEIVED
                                                                ;CHARACTER
002116  001770                          BEQ     LOOP3           ;BRANCH IF NO
                                                                ;CHARACTER RECEIVED
002120  017767  177656  177666          MOV     @RBUF,BUFFER    ;READ RECEIVED
                                                                ;CHARACTER INTO BUFFER
002126  032777  000200  177650  LOOP4:  BIT     #XRDY,@XCSR     ;CHECK FOR TRANSMITTER
                                                                ;READY
002134  001774                          BEQ     LOOP4           ;BRANCH IF NOT READY
002136  016777  177652  177642          MOV     BUFFER,@XBUF    ;TRANSMIT CHARACTER
                                                                ;TO REMOTE TERMINAL
002144  032777  000200  177636  LOOP5:  BIT     #XRDY,@CXCSR    ;CHECK FOR CONSOLE
                                                                ;TRANSMITTER READY
002152  001774                          BEQ     LOOP5           ;BRANCH IF NOT READY
002154  016777  177634  177630          MOV     BUFFER,@CXBUF   ;TRANSMIT CHARACTER
                                                                ;TO CONSOLE
002162  000746                          BR      LOOP3           ;BRANCH AND WAIT FOR
                                                                ;NEXT CHARACTER
```

```
                        ; PROGRAMMING EXAMPLE FOR THE DLV11-F SHOWING
                        ;  THE FLEXIBILITY OF PROGRAMMING WITH INTERRUPTS

        000060                  .=60
000060  001130                  .WORD   INPUT           ; RECEIVER VECTOR
000062  000340                  .WORD   340             ; PRIORITY 7
000064  001236                  .WORD   OUTPUT          ; TRANSMITTER VECTOR
000066  000340                  .WORD   340             ; PRIORITY 7
        000200                  .=200

000200  000167  000574          JMP     START


000204  000000          FLAGS:  0                       ; USED AS THE COMMUNICATION LINK
                                                        ; BETWEEN THE MAIN MODULE AND
                                                        ; THE INTERRUPT ROUTINES.


        000001          INDONE= 1                       ; BIT 0 \ TO BE SET IN FLAGS TO  TELL
        000002          OUTDONE=2                       ; BIT 1 / 'MAIN' WHICH ROUTINE IS DONE.

        000100          INTENB= 100                     ; BIT POSITION OF THE INTERRUPT
                                                        ;  ENABLE BIT IN BOTH THE
                                                        ;  RCSR AND XCSR

        177560          DLADD=  177560
        177560          RCSR=   DLADD
        177562          RBUF=   DLADD+2
        177564          XCSR=   DLADD+4
        177566          XBUF=   DLADD+6

        001000                  .=1000                  ; GIVE THE STACK SOME ROOM

                        ; *********   THIS IS THE MAIN MODULE WHICH CONTROLS THE FLOW OF DATA*****


                        ; INITIALIZE THE STACK

001000                  START:
001000  010706                  MOV     PC,SP           ; SET UP THE STACK POINTER
001002  024646                  CMP     -(SP),-(SP)     ; MAKE SURE IT STARTS BELOW US.

                        ; PRINT INSTRUCTIONS

001004                  INSTR:
001004  005067  177174          CLR     FLAGS           ; START CLEAN
001010  012701  001264          MOV     #INST,R1        ; R1 IS OUR OUTPUT BYTE POINTER
001014  052737  000100  177564  BIS     #INTENB,@#XCSR  ; SET INTERRUPT ENABLE IN XCSR
001022                  1$:
001022  032767  000002  177154  BIT     #OUTDONE,FLAGS  ; WAIT FOR OUTPUT TO FINISH
001030  001774                  BEQ     1$              ; OUTPUT ROUTINE WILL SET BIT
                                                        ;  1 IN FLAGS WHEN DONE
                                ; INSTEAD OF WAITING WE COULD BE DOING OTHER PROCESSING HERE

001032                  RESTRT:
001032  005067  177146          CLR     FLAGS           ; RESTART CLEAN
001036  012701  001347          MOV     #POEM,R1        ; R1 FOR THE OUTPUT BYTE POINTER
001042  005067  000166          CLR     INCOUNT         ; INITIALIZE INPUT COUNTER
001046  012702  001535          MOV     #LINE,R2        ; R2 FOR THE INPUT BYTE POINTER
001052  052737  000100  177560  BIS     #INTENB,@#RCSR  ; SET INTERRUPT ENABLE IN RCSR
001060  052737  000100  177564  BIS     #INTENB,@#XCSR  ;    AND XCSR
```

11-5174

Figure 4-6 DLV11-F Programming Example
(Sheet 1 of 3)

```
                                ;  ALL WE CAN DO NOW IS WAIT AND KEEP CHECKING.
                                ;   BOTH TRANSMITTER AND RECEIVER ARE WORKING AT FULL SPEED.

001066                          2$:
001066  022767  000003  177110          CMP     #INDONE!OUTDONE,FLAGS  ; ARE THEY BOTH DONE?
001074  001374                          BNE     2$                     ; IF NOT, WAIT

                                ; NOW WE HAVE A LINE IN AND THE POEM OUT.

001076  005067  177102                  CLR     FLAGS                  ; CLEAN UP AGAIN
001102  012701  001535                  MOV     #LINE,R1               ; NOW WE WILL PRINT LINE
001106  052767  000100  176450          BIS     #INTENB,XCSR           ; LET IT INTERRUPT

001114                          3$:
001114  022767  000002  177062          CMP     #OUTDONE,FLAGS  ; LINE TYPED YET?
001122  001374                          BNE     3$              ; WE MUST BE PATIENT

001124  000000                          HALT                    ; ALL DONE
001126  000741                          BR      RESTRT          ;  TO DO IT AGAIN


                                ;*********    END OF THE MAIN MODULE **********

001130                          INPUT:
                                        ; THIS IS THE INPUT ROUTINE. IT IS INTERRUPT DRIVEN
                                        ;  AS EACH CHARACTER IS RECEIVED BY THE DLV11-F AN
                                        ;  INTERRUPT IS GENERATED.
                                        ; R2 IS THE GLOBAL POINTER TO THE NEXT LOCATION FOR
                                        ;  STORAGE OF THE INPUT STREAM.
                                        ; WHILE IN THIS ROUTINE THE PRIORITY IS 7
                                        ; WHEN A CARRIAGE RETURN IS SEEN, IT IS STORED AND
                                        ;  A LINE FEED IS INSERTED AFTER IT. SINCE THIS IS
                                        ;  OUR SIGNAL TO STOP WE WILL CLEAR INTERRUPT
                                        ;  ENABLE ON THE RCSR AND SET INDONE IN FLAGS.

        000015                          CR=     15              ;CARRIAGE RETURN
        000012                          LF=     12              ; LINE FEED


001130  113704  177562                  MOVB    @#RBUF,R4       ; SAVE CHARACTER IN REGISTER 4
001134  042704  177600                  BIC     #177600,R4      ; STRIP OFF PARITY AND JUNK
001140  020427  000015                  CMP     R4,#CR          ; IS IT A CARRIAGE RETURN
001144  001414                          BEQ     10$             ; IF SO, WE'RE DONE

001146  026727  000062  000110          CMP     INCOUNT,#72.    ; DO WE HAVE 72 CHARACTERS ON THIS LINE
001154  103404                          BLO     1$              ; NO--SKIP AROUND <CRLF> INSERTION

001156  005067  000052                  CLR     INCOUNT         ; RESET COUNTER
001162  004767  000034                  JSR     PC,CRLF         ; INSERT CRLF

001166                          1$:
001166  110422                          MOVB    R4,(R2)+        ; CHARACTER INTO LINE
001170  005267  000040                  INC     INCOUNT         ; COUNT IT
001174  000411                          BR      11$             ; EXIT

                                ; THIS CODE WRAPS IT UP AFTER RECEIVING A CARRIAGE RETURN

001176                          10$:
001176  004767  000020                  JSR     PC,CRLF         ; FINISH LINE
001202  105012                          CLRB    (R2)            ; APPEND NULL BYTE
001204  042767  000100  176346          BIC     #INTENB,RCSR    ; NO MORE INTERRUPTS
001212  052767  000001  176764          BIS     #INDONE,FLAGS   ; SIGNAL DONE

001220                          11$:
001220  000002                          RTI                     ; EXIT

001222                          CRLF:
001222  112722  000015                  MOVB    #CR,(R2)+       ; CARRIAGE RETURN
001226  112722  000012                  MOVB    #LF,(R2)+       ; LINE FEED
001232  000207                          RTS     PC              ; RETURN

001234  000000                  INCOUNT:0                       ; CHARACTERS PER LINE COUNT
```


Figure 4-6 DLV11-F Programming Example
(Sheet 2 of 3)

```
001236                          OUTPUT:
                                        ; THIS IS THE OUTPUT ROUTINE
                                        ; IT ALSO IS INTERRUPT DRIVEN.
                                        ; THIS ROUTINE IS CALLED WHEN THE TRANSMITTER IS READY
                                        ;  TO OUTPUT ANOTHER CHARACTER.
                                        ; R1 IS OUR GLOBAL POINTER TO THE LINE OF OUTPUT.
                                        ; 'OUTPUT' RUNS AT A PRIORITY OF 7
                                        ; A NULL BYTE IS THE TERMINATION SIGNAL.

001236  112137  177566                  MOVB    (R1)+,@#XBUF    ; TRANSMIT THE CHARACTER
001242  105711                          TSTB    (R1)            ; NEXT BYTE NULL?
001244  001006                          BNE     1$              ; NO-- EXIT AND WAIT

001246  042737  000100  177564          BIC     #INTENB,@#XCSR  ; NO MORE INTERRUPTS
001254  052767  000002  176722          BIS     #OUTDONE,FLAGS  ; TELL THE MAIN ROUTINE

001262                          1$:
001262  000002                          RTI                     ; EXIT


                                ;*******************************************************

                                ; ASCII STORAGE AREA:  INSTRUCTIONS, POEM, AND INPUT LINE.

001264  054524  042520  044440  INST:   .ASCII  /TYPE IN YOUR LINE ENDING /
001272  020116  047531  051125
001300  046040  047111  020105
001306  047105  044504  043516
001314     040
001315     127  052111  020110          .ASCIZ  /WITH A CARRIAGE RETURN./<CR><LF>
001322  020101  040503  051122
001330  040511  042507  051040
001336  052105  051125  027116
001344  005015     000
001347     015  046412  051101  POEM:   .ASCII  <CR><LF>/MARY HAD A LITTLE LAMB/<CR><LF>
001354  020131  040510  020104
001362  020101  044514  052124
001370  042514  046040  046501
001376  006502     012
001401     040  052111  020123          .ASCII  / ITS FLEECE WAS WHITE AS SNOW,/<CR><LF>
001406  046106  042505  042503
001414  053440  051501  053440
001422  044510  042524  040440
001430  020123  047123  053517
001436  006454     012
001441     101  042116  042440          .ASCII  /AND EVERY WHERE THAT MARY WENT/<CR><LF>
001446  042526  054522  053440
001454  042510  042522  052040
001462  040510  020124  040515
001470  054522  055440  047105
001476  006524     012
001501     040  044124  020105          .ASCIZ  / THE LAMB WAS SURE TO GO./<CR><LF>
001506  040514  041115  053440
001514  051501  051440  051125
001522  020105  047524  043440
001530  027117  005015     000
001535     000                  LINE:   .BYTE 0 ; STORAGE FOR INPUT LINE STARTS HERE

        000001                  .END
```

11-5176

Figure 4-6 DLV11-F Programming Example
(Sheet 3 of 3)

4.6 PROGRAMMING NOTES

Several programming considerations are presented below. Additional information is available from program listings and current software manuals.

1. Character Format – Figure 4-7 shows the serial character format. Note that when less than eight data bits are used, the character must be right-justified to the least significant bit. The character format pertains to both the receiver and the transmitter.

2. Maintenance Mode – The maintenance mode is selected by setting the MAINT bit (bit 02) in the XCSR. In this mode, the interface disables the normal input to the receiver and replaces it with the output of the transmitter. The programmer can then load various bits into the transmitter and read them back from the receiver to verify proper operation of the DLV11-E and DLV11-F logic circuits.

3. Clearing Interrupt Enable Bits – Before executing an instruction that will clear the XCSR or RCSR interrupt enable bits, the processor should be set to its highest priority (PS bit 7 = 0). This will prevent the processor from recognizing an XCSR or RCSR interrupt request that occurs during instruction execution and then erroneously acknowledging that request after the instruction has cleared it. If the computer were to acknowledge the interrupt request after the interrupt enable bit has been cleared, it could result in a bus timeout error when the processor attempts to input a vector from the device.

4. Programmable Baud Rate – The baud rate is programmed by loading the desired bits into the high byte of the XCSR and setting bit 11. An example of a program step that does this is:

```
MOVB #130,XCSR+1 ;300 BAUD BIT11 ENABLE PROGRAMMABLE
```

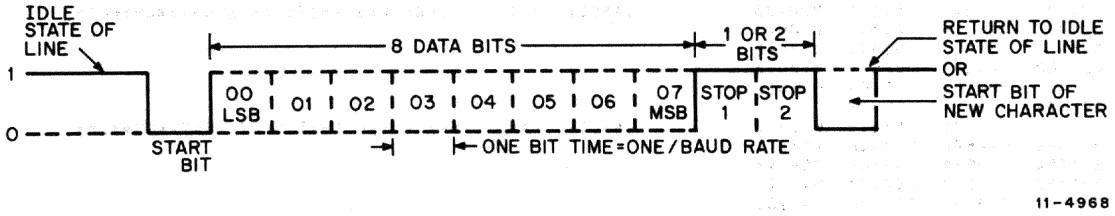


Figure 4-7 Serial Data Format

CHAPTER 5
DETAILED TECHNICAL DESCRIPTION

5.1 GENERAL

This chapter describes, on a detailed functional level, each of the 12 circuits discussed in Chapter 2. For a description of the major LSI chips, refer to Appendix A. For circuit schematics, refer to *DLV11-E Asynchronous Line Interface, Circuit Schematics* (DIGITAL part number D-CS-M8017-0-1) or to *DLV11-F Asynchronous Line Interface, Circuit Schematics* (DIGITAL part number D-CS-M8028-0-1). The functional areas described in this chapter are illustrated individually. It may be helpful, however, to refer back to Figure 2-3 for a general overview.

5.2 BUS INTERFACE

Four DC005 transceiver chips perform the bus interface functions. The chips receive from and transmit to both the computer's Bussed Address/Data Lines (BDALs) and the module's three-state bus data lines (DATs). The chips decode the module's address from the LSI-11 bus and place interrupt vectors on the LSI-11 bus.

5.2.1 Address Decoding

The computer addresses the module for both input and output data transfers. A data transfer occurs in two stages: address time and data time. (These are described further in Paragraph 5.3,I/O Control Logic.) During address time the computer places the address on bus lines BDAL00 L through BADL15 L and asserts the memory bank 7 select signal (BBS7 L). This signal indicates that the address is in the 28–32K range of addressing space, and enables the DC005 transceiver chips to decode the address. The circuit performs a logical inversion on the entire address word and places it on the three-state bus. However, it decodes only bits 03 through 12 (Figure 5-1). Bits 00 through 02 pertain to device register selection, and are routed to the I/O control logic for decoding. Bits 13 through 15 pertain to the selection of addressing space. Their states are already indicated by BBS7 L. Bits 03 through 12 contain the address of the specific DLV11-E or DLV11-F being addressed. The bus interface's address decoding circuitry compares the states of bits 03 through 12 with the conditions set by address jumpers A3 through A12. If a match is decoded, the circuit asserts MATCH H to enable the I/O control logic.

During data time, the transceivers transfer data from the LSI-11 bus lines to the three-state bus lines if the operation is an output data transfer. If the operation is an input data transfer, the I/O control logic asserts the "in word" signal (INWD L), switching the transceivers to their opposite state, in which they transfer data from the three-state bus to the LSI-11 bus.

5.2.2 Vector Addressing

The bus interface circuit can place one of two vector addresses on the BDAL lines when the interrupt function is enabled by the program. Which vector is placed on the bus lines is determined by the interrupt logic. Bit 02 of the vector word (Figure 5-2) is controlled by VECRQSTB H from the interrupt logic. This bit is in a TRUE state for a transmitter interrupt and is negated for a receiver interrupt. Bits 03–08 can be selected by the user by removing or inserting vector jumpers V3 through V8. The remaining bits are all zeros.

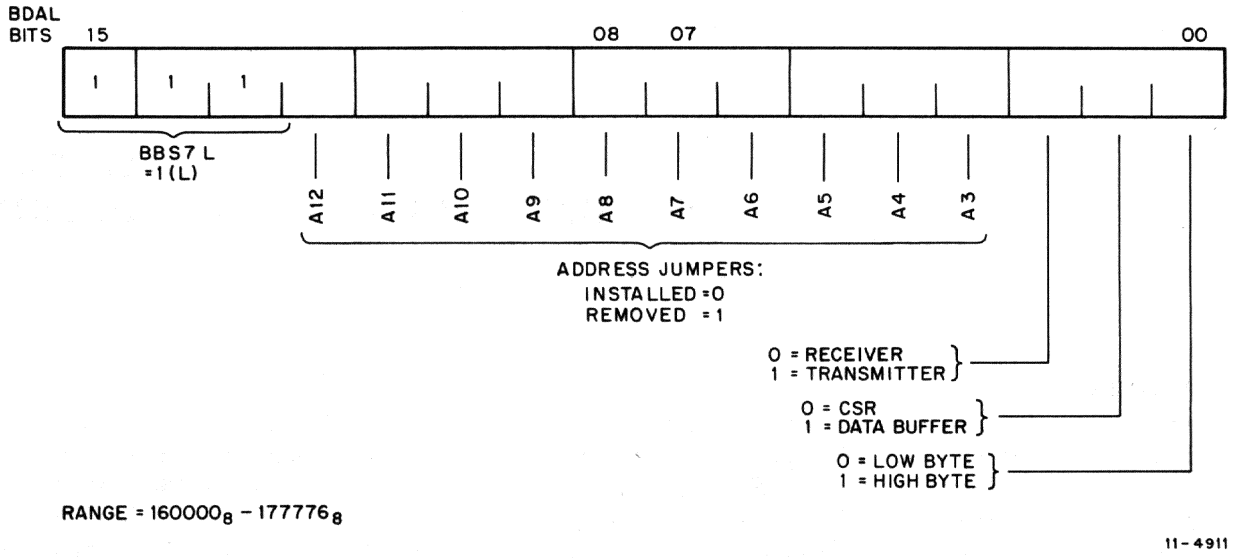


Figure 5-1 DLV11-E and DLV11-F Addresses

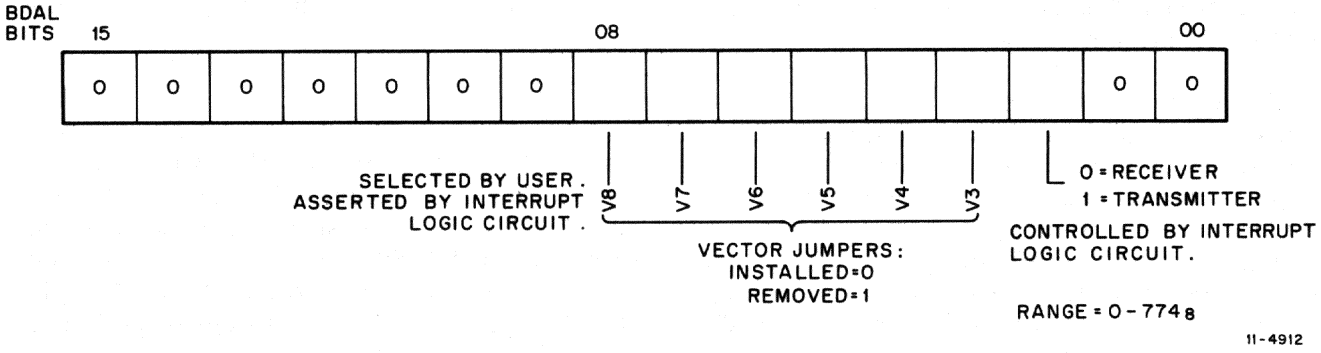


Figure 5-2 DLV11-E and DLV11-F Interrupt Vectors

To place a vector on the bus lines, the interrupt logic asserts VECTOR H. VECTOR H enables those bits whose corresponding vector jumpers have been installed. This action does not require BBS7 L or INWD L.

5.3 I/O CONTROL LOGIC

The I/O control logic monitors LSI-11 bus control signals, decodes the device address from the last three bits of the address word, and controls the flow of data in the module. The major element in the I/O control logic is a DC004 protocol chip. This chip decodes DAT00 H through DAT02 H, monitors VECTOR H from the interrupt logic and MATCH H from the bus interface, and responds to the following LSI-11 bus control lines:

BSYNC L	Bussed Synchronize
BWTBT L	Bussed Write Byte
BDIN L	Bussed Data In
BDOUT L	Bussed Data Out

The chip controls four register select lines for enabling the device registers (Table 5-1). It also generates OUTHB L and OUTLB L signals to control which byte of a register is loaded. The chip produces the "in word" signal (INWD L) to control the direction of data flow through the bus interface transceivers. It also generates a reply (BRPLY L) to the LSI-11 bus.

Table 5-1 Register Selection

DAT02 H	DAT01 H	Select Line Asserted	Register Selected
Low	Low	SEL0 L	RCSR
Low	High	SEL2 L	RBUF
High	Low	SEL4 L	XCSR
High	High	SEL6 L	XBUF

5.3.1 Input Operation

When the LSI-11 program reads data in from the DLV11-E or DLV11-F to the computer (DATI bus cycle), the input data transfer proceeds as follows:

1. The program places the device address on LSI-11 bus lines BDAL00 L through BDAL15 L, and asserts BBS7 L (Figures 5-3 and 5-4). BWTBT L is negated at this time because all input transfers are full words rather than bytes.

2. BBS7 L enables the bus interface logic to decode the address. The circuit decodes the address and sends MATCH H to the I/O control logic. It also inverts the address word and places it on three-state bus lines DAT00 H through DAT15 H. DAT00 H through DAT02 H are applied to the I/O control logic.

3. The computer asserts BSYNC L. The leading edge of BSYNC L latches the states of MATCH H and DAT00 H through DAT02 H into the protocol chip of the I/O control logic. The chip decodes DAT00 H through DAT02 H to recognize the address of the device register, and then asserts the appropriate register select line. It asserts SEL2 L, for example, if the program is addressing the RBUF. The register select signal conditions a gate that will later be enabled for the data transfer.

4. The computer next removes the address from the LSI-11 bus lines, negates BBS7 L, and asserts BDIN L. BDIN L is gated with the register select lines. This enables the selected register to place its contents on the three-state bus. BDIN L is also routed to the protocol chip, which asserts INWD L. INWD L causes the bus interface transceivers to transfer the data from the module's three-state bus to the LSI-11 bus. The chip waits about 150 ns for the data to stabilize on the three-state bus lines and then asserts INWD L and BRPLY L. BRPLY L signals the computer that the data is on the bus.

5. The computer reads in the data and then negates BDIN L.

6. The I/O control logic responds to the negation of BDIN L by negating BRPLY L.

7. The computer terminates the bus cycle by negating BSYNC L.

8. In the absence of a TRUE condition on BSYNC L, the protocol chip releases the register select and INWD L signals. The bus interface reverts to its normal condition of receiving from the LSI-11 bus and transmitting onto the three-state bus.

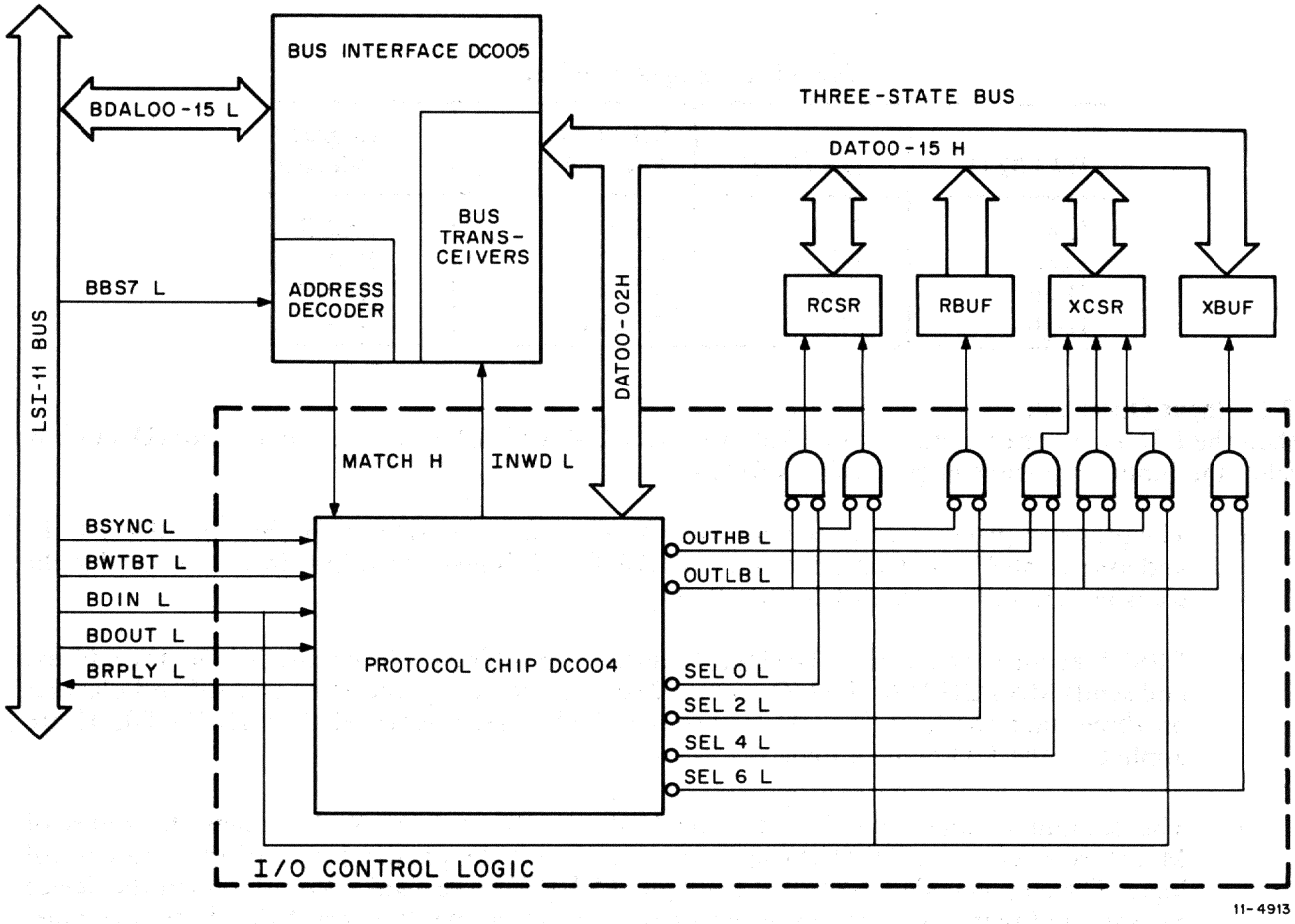
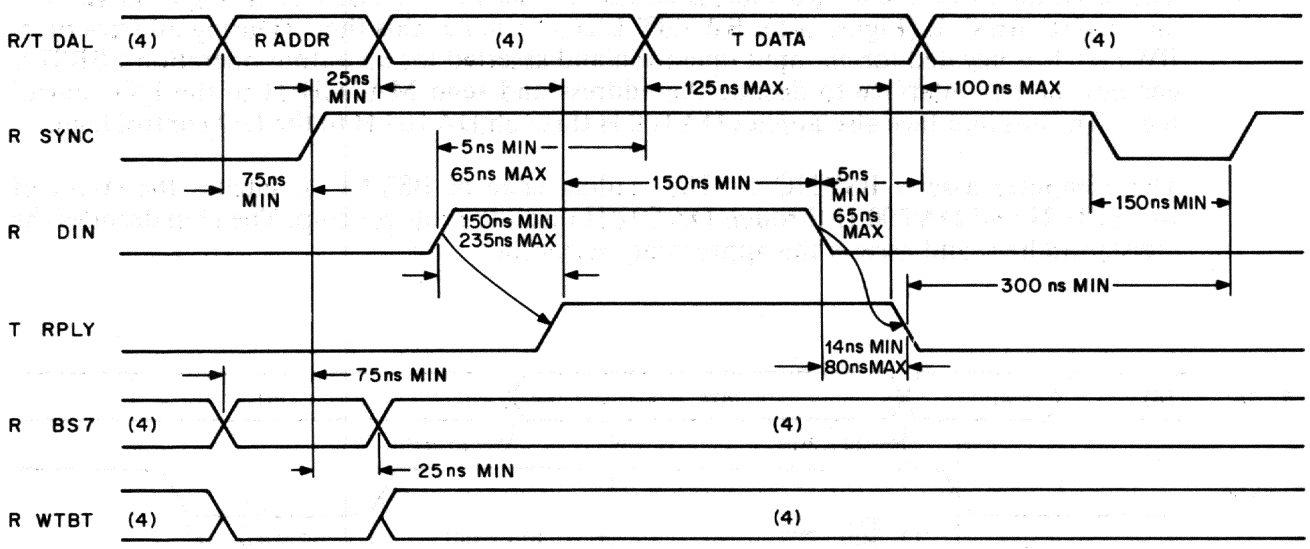


Figure 5-3 I/O Control Logic, Block Diagram



NOTES:

1. Timing shown at Master and Slave Device Bus Driver inputs and Bus Receiver Outputs.
2. Signal name prefixes are defined below:
   T = Bus Driver Input
   R = Bus Receiver Output
3. Bus Driver Output and Bus Receiver Input signal names include a "B" prefix.
4. Don't care condition.

11-4914

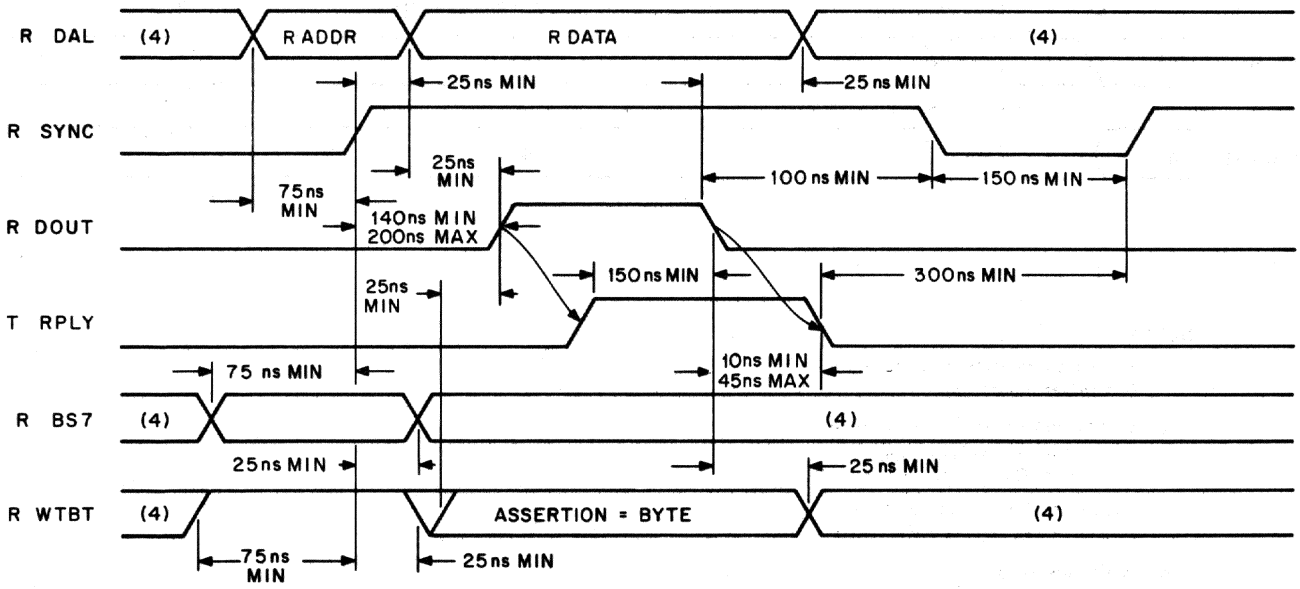
Figure 5-4 Data Input Timing

5.3.2 Output Operation

The DLV11-E and DLV11-F can accept data from the computer in either bytes or words. To write a word out to the interface module, the computer performs a DATO bus cycle; for a byte, a DATOB bus cycle. An output data transfer proceeds as follows:

1. The program places the device address on LSI-11 bus lines BDAL00 L through BDAL15 L, and asserts BBS7 L (Figure 5-5). BWTBT L is asserted at this time. (During address time, BWTBT L is negated for an input operation and asserted for an output operation.) BBS7 L enables the bus interface to decode the address and send MATCH H to the I/O control logic. The bus interface also applies DAT00 H through DAT02 H to the I/O control logic.

2. The computer asserts BSYNC L. The leading edge of BSYNC L latches the states of MATCH H and DAT00 H through DAT02 H into the protocol chip. The chip decodes the register address and asserts the appropriate select line.



NOTES:

1. Timing shown at Master and Slave Device Bus Driver Inputs and Bus Receiver Outputs.
2. Signal name prefixes are defined below:
   T = Bus Driver Input
   R = Bus Receiver Output
3. Bus Driver Output and Bus Receiver Input signal names include a "B" prefix.
4. Don't care condition.

11-4915

Figure 5-5 Data Output Timing

3. The computer removes the address from BDAL00 L through BDAL15 L and negates BBS7 L. If a byte is to be transferred out to the device register, BWTBT L remains asserted. If a word is to be transferred, BWTBT L is negated. At this time the computer places data on the LSI-11 bus lines and asserts BDOUT L. BDOUT L goes to the protocol chip and enables it to decode the states of BWTBT L and DAT00 H. The chip uses these signals to determine the desired byte of the addressed register (Table 5-2). The device registers are configured for output transfers unless switched otherwise by BDIN L. Therefore, BDOUT L is not gated with the register select and byte lines.

Table 5-2 Byte Selection (Output Operations Only)

		Select Line	Byte
BWTBT L	DAT00 H	Asserted	Selected
High	Don't Care	OUTLB L and OUTHB L	Both
Low	Low	OUTLB L	Low
Low	High	OUTHB L	High

4. About 150 ns after it receives BDOUT L, the protocol chip issues BRPLY L to the computer to signal that the module is loading data.

5. The computer removes the data from its bus lines and negates BDOUT L.

6. The protocol chip responds to this by terminating BRPLY L.

7. The computer then terminates the bus cycle by negating BSYNC L and, if applicable, BWTBT L.

8. When BSYNC L is negated, the protocol chip negates the register select and byte select lines.

5.3.3 Vector Operation

The I/O control logic has the additional function of asserting BRPLY L in response to VECTOR H from the interrupt logic. This action is independent of BSYNC L and MATCH H. It is part of the interrupt sequence and is discussed further in Paragraph 5.7.

5.4 CONTROL/STATUS REGISTERS

The RCSR and XCSR each consist of several types of flip-flop latches, rather than single devices. Status bits from various circuits are placed in the registers and then, under the control of the I/O control logic, gated on to the three-state bus for transfer to the computer. Control bits from the computer are loaded into the registers from the three-state bus. While in the registers, they direct the operation of the modules.

5.4.1 CSR Data Flow

RCSR operation differs between the DLV11-E and DLV11-F only in those areas concerned with the peripheral interface requirements (Figures 5-6 and 5-7). Some bits are set by the peripheral interface circuit, receiver active circuit, and the RBUF, while others are set by the program via three-state bus lines DAT00 H through DAT06 H. All RCSR bits except the DLV11-F's Reader Run Enable bit may be read by the program. Refer to Chapter 4 for a listing of how the bits are set and cleared.

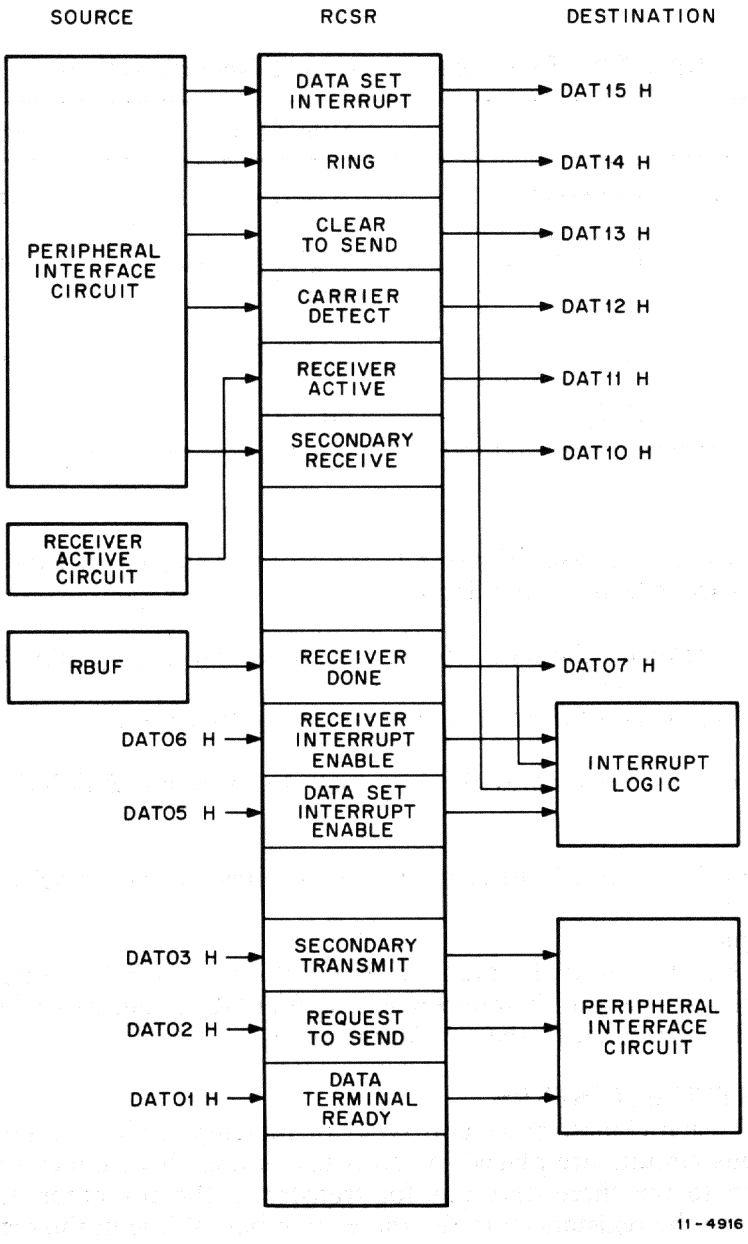


Figure 5-6 DLV11-E RCSR Data Flow

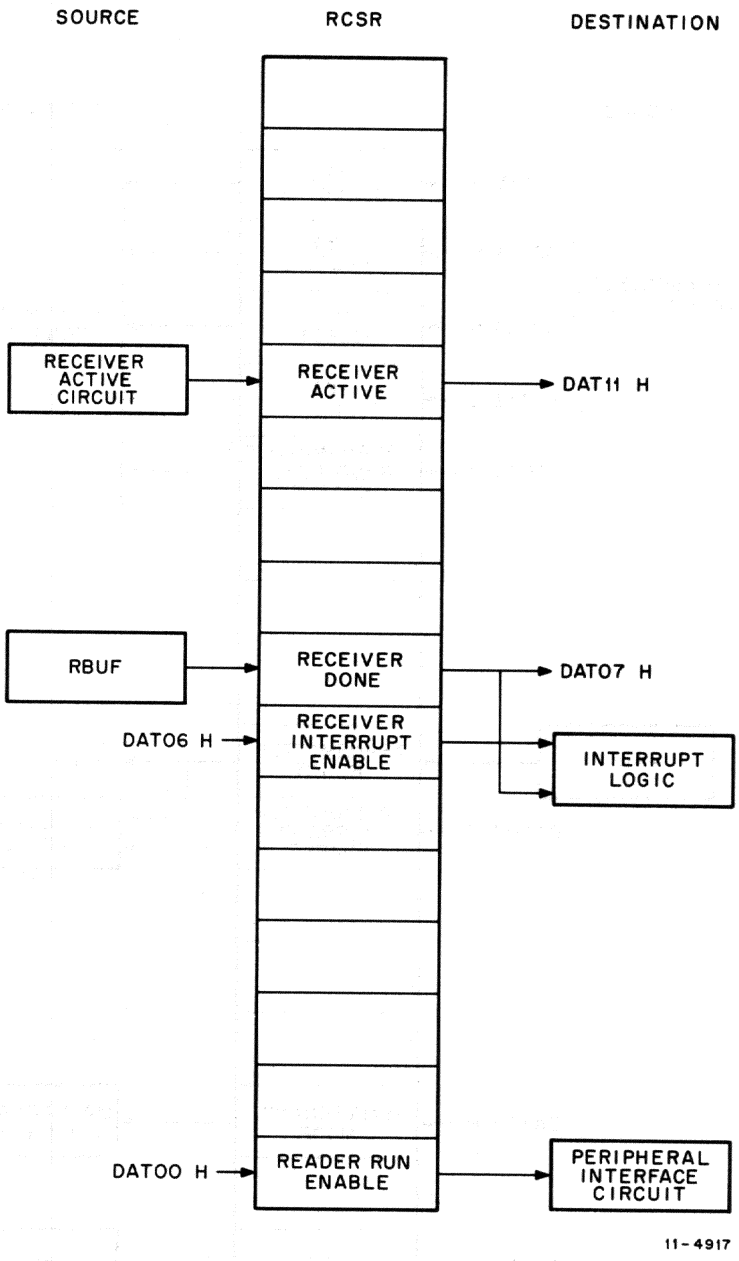


Figure 5-7 DLV11-F RCSR Data Flow

XCSR operation is the same for both the DLV11-E and the DLV11-F (Figure 5-8). The bits for the baud rate control circuit are write only bits. TRANSMITTER READY is a read only bit. The other XCSR bits are both read and write bits.

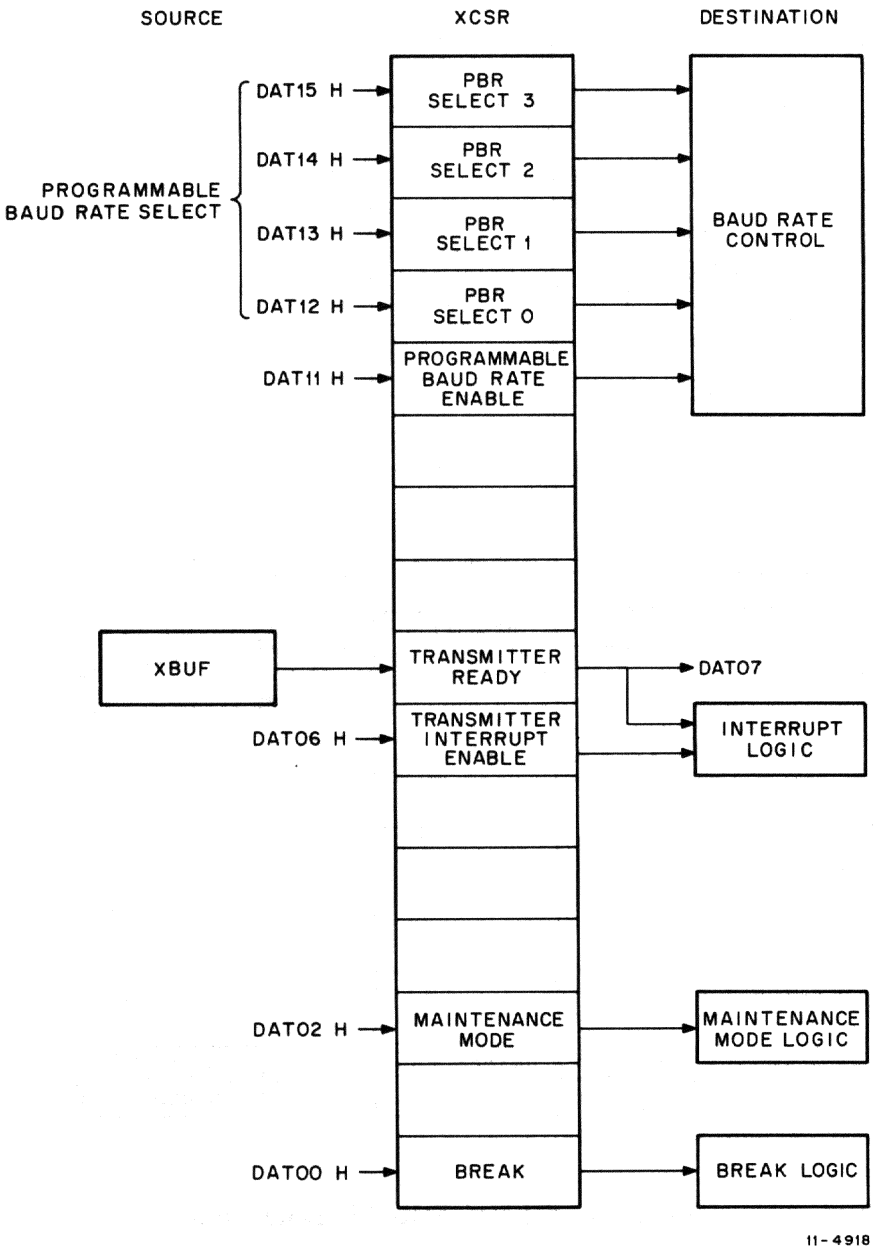


Figure 5-8 DLV11-E and DLV11-F XCSR Data Flow

5.4.2 Input Operation

The contents of the RCSR and XCSR are read into the LSI-11 by an input data transfer (DATI). The computer places the address of the register on the LSI-11 bus and then the bus interface and I/O control logic decode the address. The I/O control logic generates register select signals that switch data selectors to the desired source (Figure 5-9).

The select signals also enable the output of the data selectors and, if the RCSR is addressed, enable bus drivers. The status information leaves the CSRs on the three-state bus. The bus interface circuit then transfers the data to the LSI-11 bus.

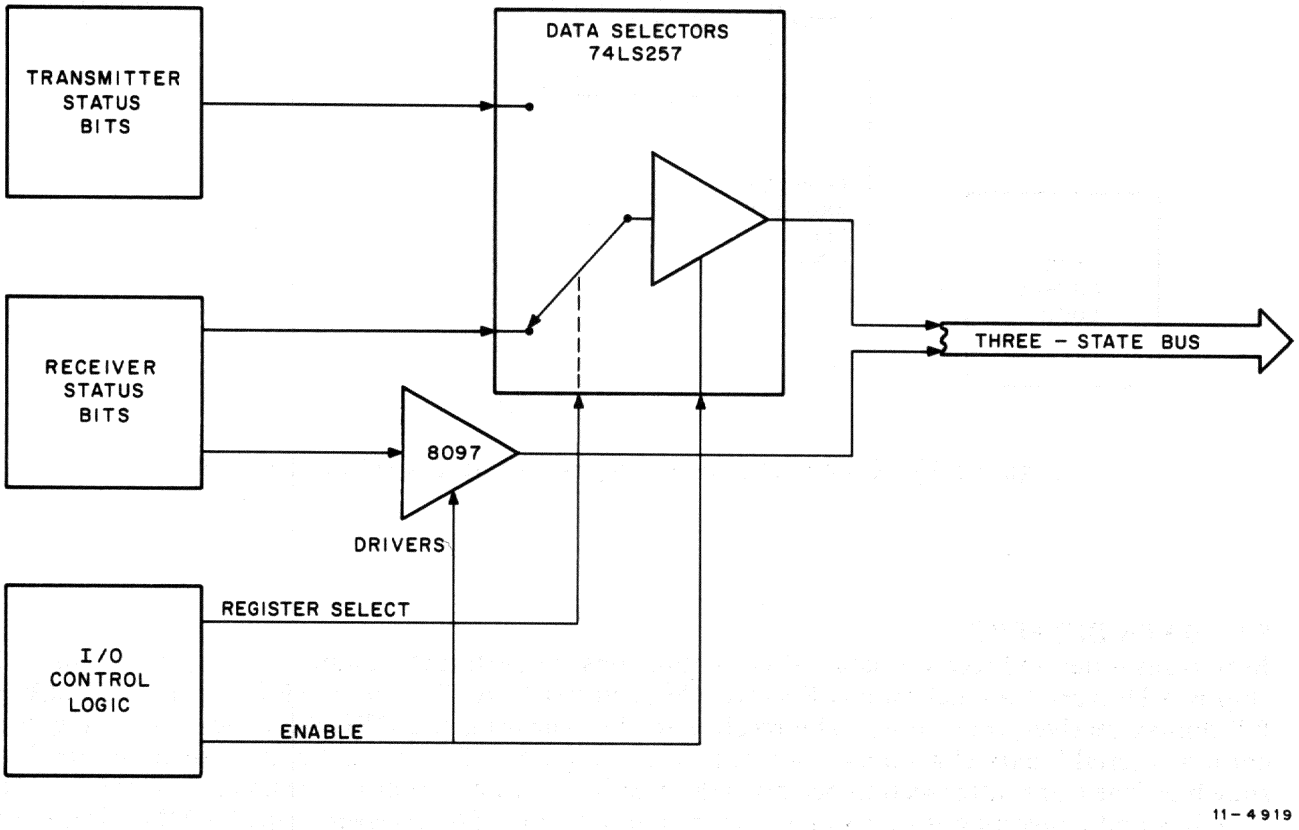


Figure 5-9 Control/Status Registers During DATI

5.4.3 Output Operation

The LSI-11 writes control bits out to the CSRs by an output data transfer (DATO or DATOB). Normally the RCSR is loaded by a DATOB cycle because only the low byte contains control bits. (The bits used in the high byte are all read only status bits.) The XCSR can be loaded by a DATOB cycle if it is desired to load only the high byte (e.g., to change the baud rate), or only a low byte. The computer uses a DATO cycle to transfer a full word to the XCSR.

When the computer addresses the desired register, the bus interface and I/O control logic circuits decode the address. The I/O control logic generates register and byte selection signals that enable the chips comprising the selected register (Figure 5-10). Flip- flops latch in control bits that are held in the register, and data selectors route other bits to latches in the circuits which they control.

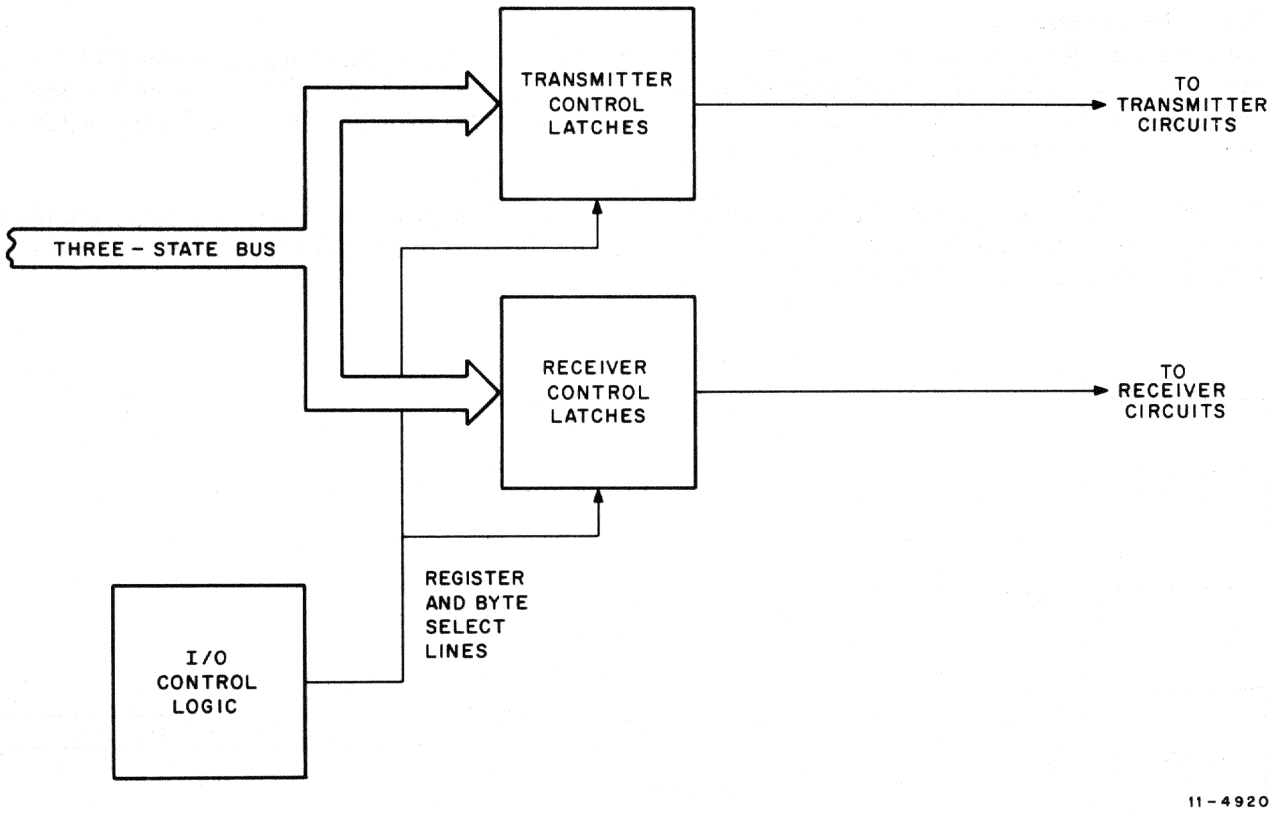


Figure 5-10 Control/Status Registers During DATO or DATOB

5.5 DATA BUFFERS

Both transmitter and receiver data buffering functions are performed mainly by a single LSI chip. The chip is a Universal Asynchronous Receiver/Transmitter (UART). The UART is a double-buffered, full-duplex receiver/transmitter. The receiver section performs the RBUF function, accepting asynchronous serial binary characters, converting them to parallel format, and placing them on the three-state bus. The transmitter section performs the XBUF function, accepting parallel data from the three-state bus and converting it to a serial aysnchronous output. The receiver strips START, STOP, and parity bits off the data coming in from the peripheral device. The transmitter appends START, STOP, and parity bits to the data being transmitted out to the peripheral device. Jumper control of the STOP and parity bits, and the number of data bits, is defined in Chapter 3.

The UART is driven by a clock signal (or two clock signals, for split speed operation) from the baud rate control circuit. The clock speed is 16 times the baud rate of the UART. The UART transmitter internally synchronizes the START bit with the clock input to ensure a full 16-element (clock periods) START bit independent of the time of data loading. The receiver rejects any received START bit that lasts less than one-half of a bit time.

5.5.1 Receiver Operation

Serial data coming in from the peripheral device is converted to TTL levels by the peripheral interface and is applied to the UART's receiver section (Figure 5-11). The UART samples the serial input data line at 16 times the data bit rate. The line is in a continuous marking state when idle. When a START bit arrives, the UART detects the mark-to-space transition and begins loading the received character into the receiver shift register. The character is shifted to have its least significant bit in the lowest bit position of the register. If jumpered for checking parity, the UART checks the total of the received data bits plus the parity bit. (It checks for an even total if even parity has been selected, and an odd total if odd parity has been selected.) A parity error will result in a flag bit (P ERR) being set (Figure 5-12).

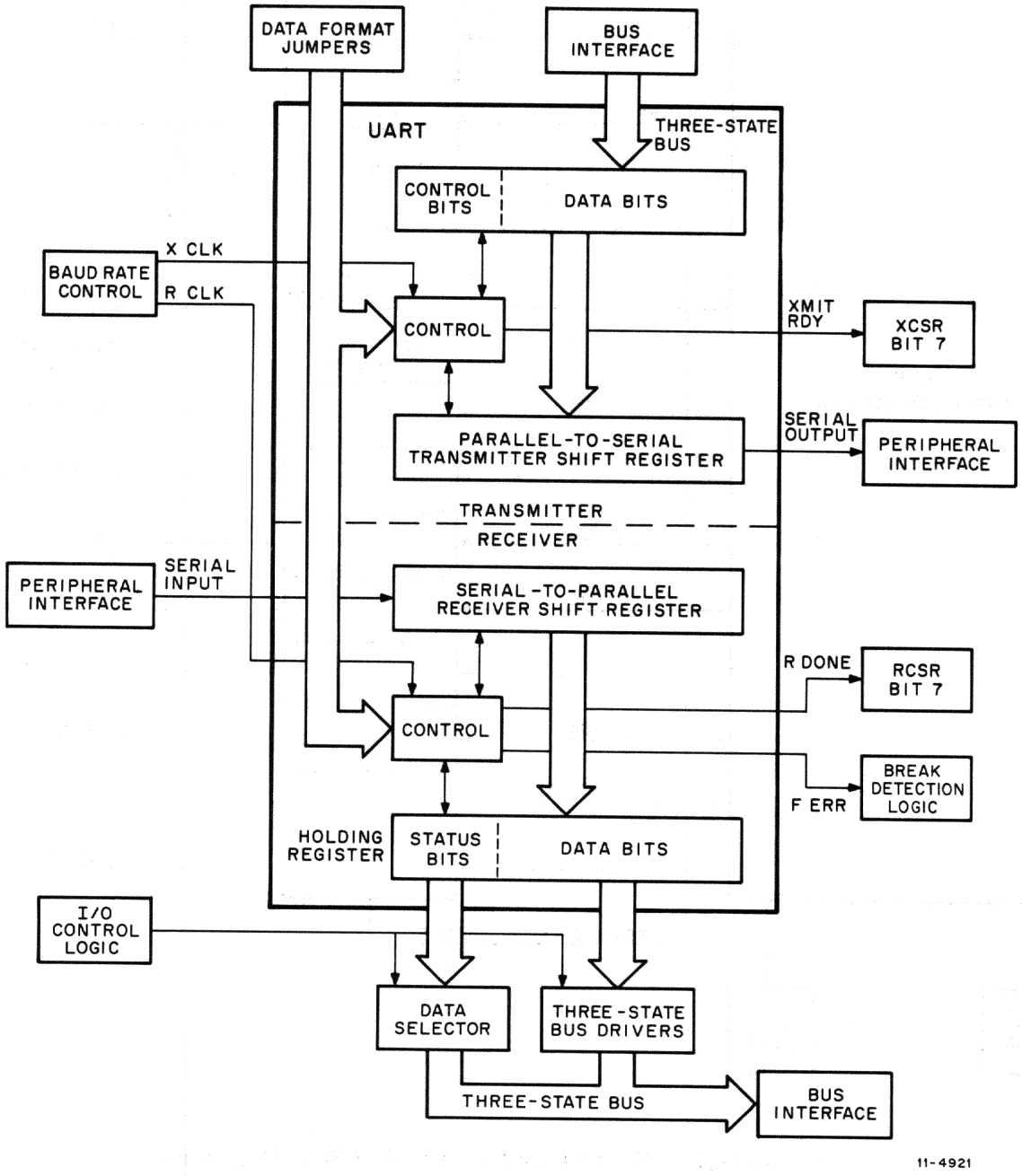


Figure 5-11 UART Signal Flow

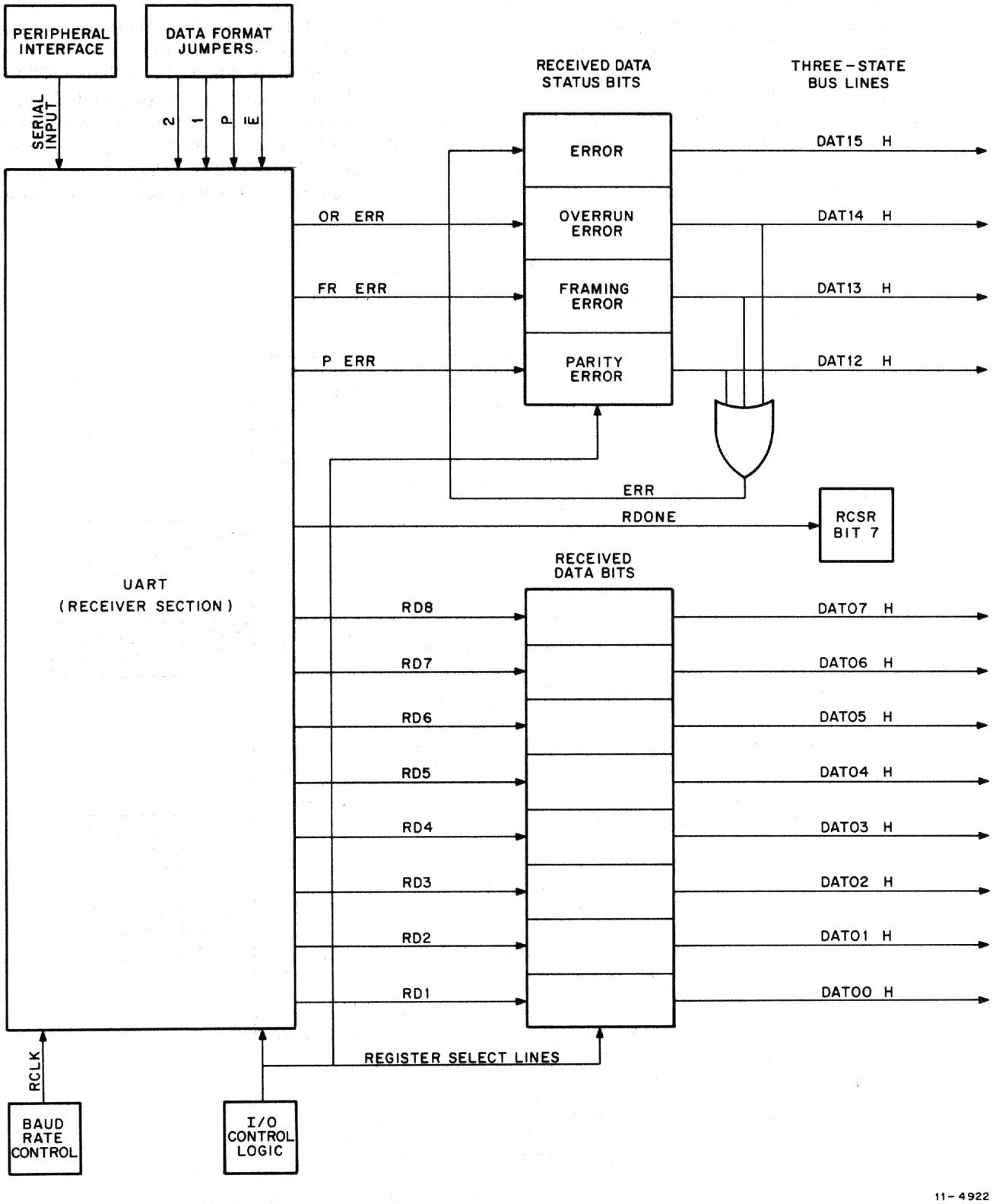


Figure 5-12 DLV11-E and DLV11-F RBUF Data Flow

The UART checks the STOP bit to see if it is marking. If the line is spacing when the UART checks for a STOP bit, the UART sets the framing error flag (FR ERR).

When the UART receives the center of the first STOP bit, it transfers the data in parallel from the receiver shift register to the holding register. At this time, the data and error bits become available for gating on to the three-state bus, and the UART asserts the receiver done (RDONE H) signal. This sets the RECEIVER DONE status bit in the RCSR.

If the receiver interrupt enable bit is set, RDONE H initiates an interrupt request. The LSI-11 then has a full character period to service the interrupt before the next character moves into the holding register. During this time the next character is being assembled in the receiver shift register. After an LSI-11 DATI sequence has taken the data, the I/O control logic resets the receiver done status bit. If the LSI-11 program does not take the data before the next character enters the holding register, RDONE H does not get reset. In this case, the UART sets the data overrun flag (OR ERR). This bit goes with the next received data word to indicate that the old data was lost.

Any of the three error conditions (Overrun Error, Framing Error, or Parity Error) sets an end check error flag (ERR) as well as its own flag. These error bits do not initiate an interrupt request, but they are available in the high byte of the RBUF for the programmer's use.

5.5.2 Transmit Operation

The XBUF consists of two registers and their controlling logic, all of which are contained in the UART chip. A holding register stores the parallel data taken off the three-state bus, and then transfers it in parallel to the transmitter shift register. Next, the data is shifted out serially. The format of the character being transmitted is controlled by the data format jumpers.

During idle time the UART transmits a continuous marking signal and holds the transmitter ready status bit (XMIT RDY) asserted in the XCSR. XMIT RDY initiates a transmitter interrupt request if the transmitter interrupt enable bit is set in the XCSR. If the interrupt function is not enabled, the UART transmitter remains idle until the program requires it.

When the program has data to transmit to a peripheral device, it uses a DATO or DATOB sequence to address the XBUF and place the data on the bus lines. The bus interface moves the data from BDAL00 L through BDAL07 L to DAT00 H through DAT07 H. The I/O control logic enables the XBUF to load the data into its holding register (Figure 5-11). When the data enters the holding register, the UART negates XMIT RDY. The UART then transfers the data in parallel from the holding register to the transmitter shift register and reasserts XMIT RDY. In the transmitter shift register, the UART attaches the selected START, STOP, and Parity bits. The assembled character is then shifted serially out of the XBUF to the peripheral interface circuitry (Figure 5-13).

The time between the leading edge of the register select signal from the I/O control logic and the corresponding mark-to-space transition of the serial output line is within one clock cycle (1/16 of a bit time) if the transmitter has been idle.

XMIT RDY is asserted as soon as a character is transferred from the holding register to the transmitter shift register, thereby indicating that the holding register is empty. The next character may be loaded immediately, even while the first character is still being serially shifted out of the transmitter shift register. Thus, if the holding register and transmitter shift register are both empty, the LSI-11 can parallel-transfer a two-character pair into the XBUF in less time than it takes for a single character to be serially transmitted to the peripheral device. This advantage of double-buffering applies only to the first two characters; that is, if a series of characters is being transmitted each character after the second must wait a serial character period for the XBUF to become ready again. The actual time depends on the baud rate.

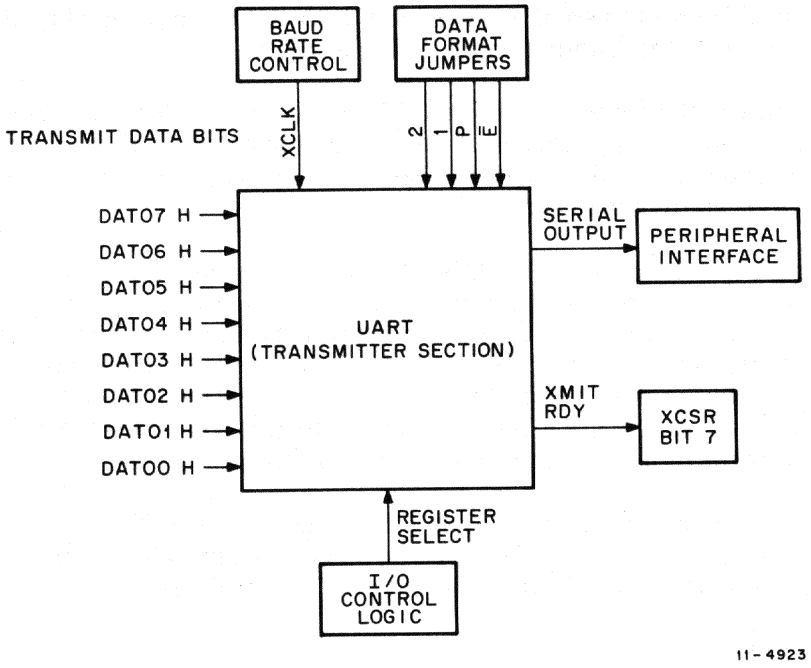


Figure 5-13 DLV11-E and DLV11-F XBUF Data Flow

5.6 RECEIVER ACTIVE CIRCUIT

The receiver active circuit produces a status bit to indicate that the RBUF is receiving a character of data. This status bit, RECEIVER ACTIVE, is set by the START bit of the received data character and cleared by the receiver done (RDONE H) signal from the UART.

During the period between received data characters, SI MARK H from the peripheral interface holds the receiver clock counter in the cleared state (Figure 5-14). When a START bit is received, SI MARK H changes state and releases the CLEAR line to the counter. The counter begins to count receiver clock pulses from the baud rate control circuit. Each RCLK H pulse is 1/16th of a bit time. The counter counts to eight, which places it in the center of the START bit, then asserts RBUSY H. RBUSY H is routed to the RCSR, where it can be read in by the program as RECEIVER ACTIVE. It is also used to stop the counter and to inhibit SI MARK H from clearing the counter. When the RBUF has finished receiving the character, the UART asserts RDONE H. RDONE H clears the counter, thereby negating RBUSY H and returning the circuit to its initial condition. Thus, RECEIVER ACTIVE is set during the time from the center of the START bit to the leading edge of R DONE H.

5.7 INTERRUPT LOGIC

Both transmitter and receiver interrupt functions of the interrupt logic are handled by a single DC003 interrupt chip. This chip is described in Appendix A. The interrupt logic has a receiver interrupt channel, a transmitter interrupt channel, and control circuitry. Figure 5-15 shows the signal flow associated with the interrupt chip.

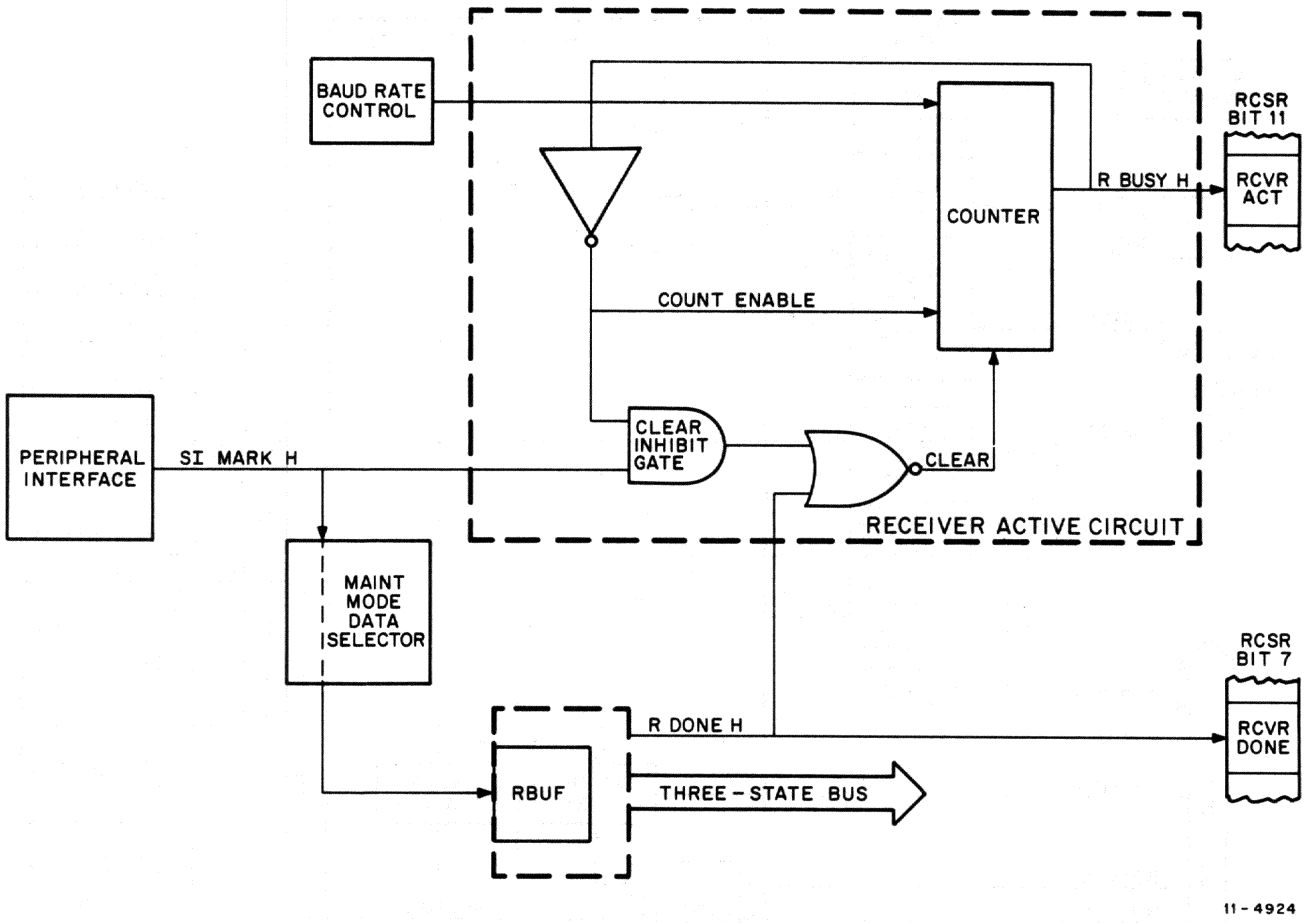


Figure 5-14 Receiver Active Circuit

5.7.1 DLV11-E Receiver Interrupts

In the DLV11-E, a receiver interrupt sequence is started by either the UART or the peripheral interface circuitry. Both cases require that the appropriate enabling bit be set in the RCSR. When the computer program sets RECEIVER INTERRUPT ENABLE (bit 06) in the RCSR, an interrupt can be caused by RDONE H from the UART. The UART asserts RDONE H when the RBUF has received and assembled a character of data. When the program sets DATA SET INTERRUPT ENABLE (bit 05) in the RCSR, an interrupt can be initiated by DATA SET INTERRUPT from the peripheral interface circuit. The peripheral interface sets DATA SET INTERRUPT when it receives control signals from a data set. When either pair of conditions is satisfied, the receiver channel will be enabled to request to interrupt the program. When the interrupt is acknowledged (discussed in Paragraph 5.7.4), the interrupt chip asserts VECTOR H. This signal causes the assertion of the vector address bits that correspond to the vector jumpers which the user has inserted. The bus interface circuit places the bits set by jumpers V3 through V8 onto LSI-11 bus lines BDAL03 L through BADL08 L. All other BDAL's are negated at this time. When the computer locates the service routine, it may check the status bits in the RCSR to determine what condition initiated the interrupt. Refer to Paragraphs 4.3 and 4.6 for notes regarding simultaneous receiver and data set interrupt conditions.

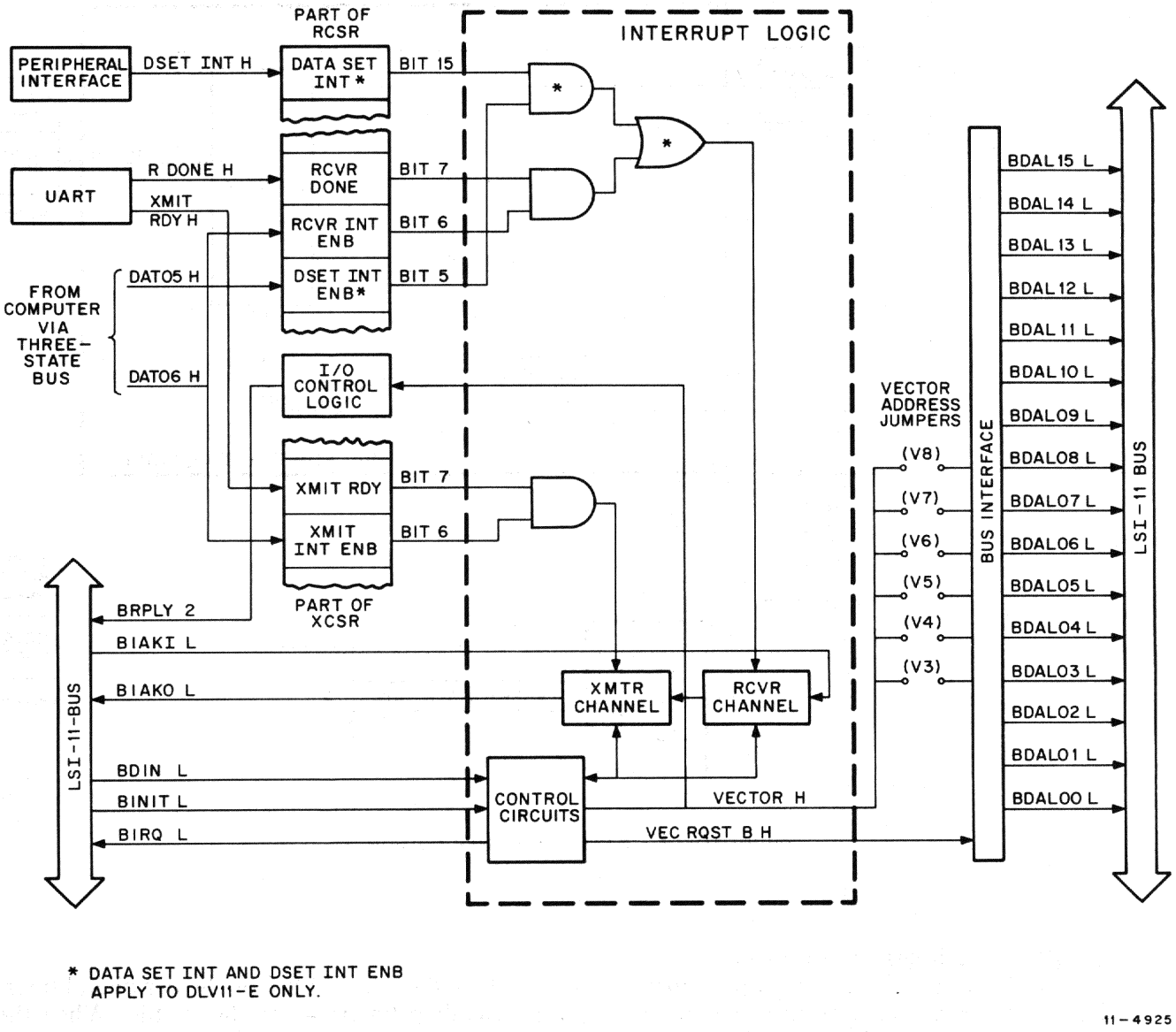


Figure 5-15 Interrupt Vector Signal Flow

5.7.2 DLV11-F Receiver Interrupts

The DLV11-F interrupt vector flow is the same as that of the DLV11-E, with the exception that it has no DATA SET INTERRUPT or DATA SET INTERRUPT ENABLE bits. The module does not support data set control, and therefore produces a receiver interrupt only for servicing the RBUF.

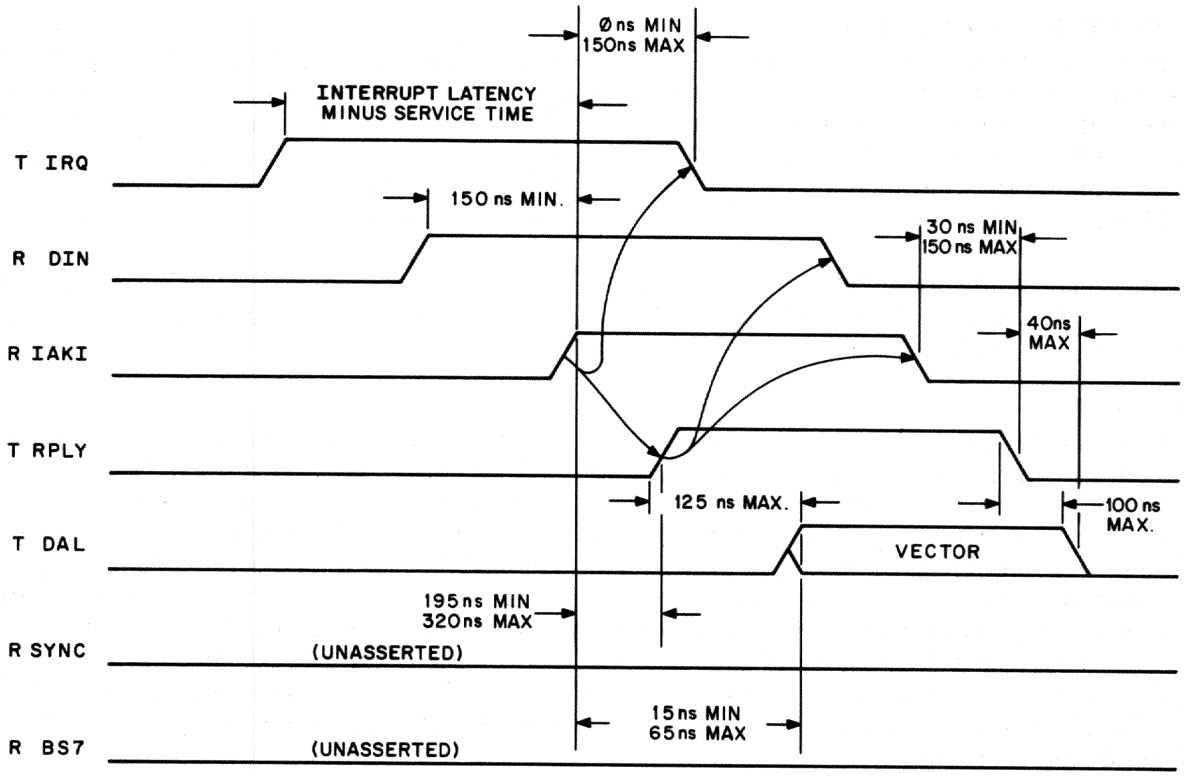
5.7.3 Transmitter Interrupts

The DLV11-E and DLV11-F function alike for transmitter interrupts. The interrupt logic generates a transmitter interrupt when the program sets TRANSMITTER INTERRUPT ENABLE (bit 06) in the XCSR and the UART asserts XMIT RDY H. The UART asserts XMIT RDY H when the XBUF is empty and ready for more data from the computer. When XMIT RDY H and TRANSMITTER INTERRUPT ENABLE are both TRUE, the transmitter channel of the interrupt chip is enabled to request interrupt service. (Although these two signals are functionally bits 06 and 07 of the XCSR, they are physically located in the interrupt chip.) After the computer acknowledges the interrupt logic's interrupt request, the circuit asserts both VECTOR H and VECRQSTB H. VECTOR H is applied to vector address jumpers V3 through V8, the same as for a receiver interrupt vector. In this case, however, VECRQSTB H causes the bus interface to assert BDAL02 L, as well as the other selected bits on BDAL03 L through BDAL08 L. This results in the vector addressing of the transmitter interrupt service routine.

5.7.4 Interrupt Transactions

Either type of interrupt begins with the interrupt logic asserting BIRQ L, the interrupt request line. This is followed by an interchange of control signals and the vector address being placed on the LSI-11 bus lines. The sequence proceeds as follows:

1. The request is initiated by the interrupt logic asserting BIRQ L (Figure 5-16).



NOTES:
1. Timing shown at Requesting Device Bus Driver Inputs and Bus Receiver Outputs.
2. Signal Name Prefixes are defined below:
   T = Bus Driver Input
   R = Bus Receiver Output
3. Bus Driver Output and Bus Receiver Input signal names include a "B" prefix.

11-4926

Figure 5-16 Interrupt Timing

2. The LSI-11 responds to BIRQ L by asserting BDIN L and then BIAKI L.

3. BIAKI L is passed down the priority chain until it reaches the section of the interrupt logic that initiated the request. When the circuit receives both BDIN L and BIAKI L, it asserts VECTOR H (and also VECRQSTB H, if a transmitter interrupt) and negates BIRQ L.

4. VECTOR H causes the I/O control logic to issue BRPLY L to the computer. VECTOR H (and VECRQSTB H, if applicable) also causes the bus interface to place the vector on the LSI-11 bus lines.

5. The computer reads in the interrupt vector and then, as a result of receiving BRPLY L, negates BDIN L. Shortly after this it also negates BIAKI L.

6. The interrupt logic negates VECTOR H (and VECRQSTB H, if applicable).

7. The negation of VECTOR H causes the I/O control logic to negate BRPLY L, and the bus interface to remove the vector from the LSI-11 bus lines.

An interrupt transaction does not require MATCH H, BSYNC L, BBS7 L, or INWD L. The interrupt logic overrides the module's normal I/O protocol. When the computer is initialized, the interrupt logic is cleared by BINIT L.

5.8 BAUD RATE CONTROL

The baud rate control circuit establishes the speeds at which the RBUF and XBUF operate. The circuit consists of two sets of wire wrap jumpers, gating circuitry, an oscillator, and a 5016 dual baud rate generator. The 5016 chip divides the oscillator frequency down to the frequency selected by the jumpers or the program. In the split speed mode of operation, it produces two separate clock frequencies: one for transmit and one for receive. The circuit routes either these clocks or an external clock to the UART to control the baud rate(s) at which the module operates.

Also included in the baud rate control are gates that decode a selection of 110 baud. When this condition is detected, the circuit asserts 110 BAUD H. This signal enables the UART to handle a data format having two STOP bits (Figure 5-17).

5.8.1 Program Control

The 5016 chip has two sections, each of which is driven by a 5.0688 MHz clock from the oscillator. The two sections of the chip each divide the 5.0688 MHz clock by a selectable amount. The selection for section B of the chip is accomplished by jumpers T0 through T3. The frequency in section A, however, can be controlled by either jumpers R0 through R3 or three-state bus lines DAT12 H through DAT15 H. The source of control for section A of the chip is selected by a data selector chip. This data selector is functionally part of the high byte of the XCSR. It is addressed by a combination of the Programmable Baud Rate Enable bit (on DAT11 H) and register select lines from the I/O control logic. If DAT11 H is asserted during a DATO output transaction, the data selector chip will route the logic states of DAT12 H through DAT15 H to the dual baud rate Generator chip to program the frequency. When DAT11 H is not asserted, the data selector chip will select jumpers R0 through R3 to control the dual baud rate generator.

When computer power is first switched on, the assertion of BDCOK L causes the data selector to select jumpers R0 through R3 as the source of the section A frequency control. From that time on the circuit can choose either the jumpers or the XCSR bits, as determined by the state of the Programmable Baud Rate Enable bit. A table of jumper combinations and their corresponding baud rates is presented in Chapter 3.

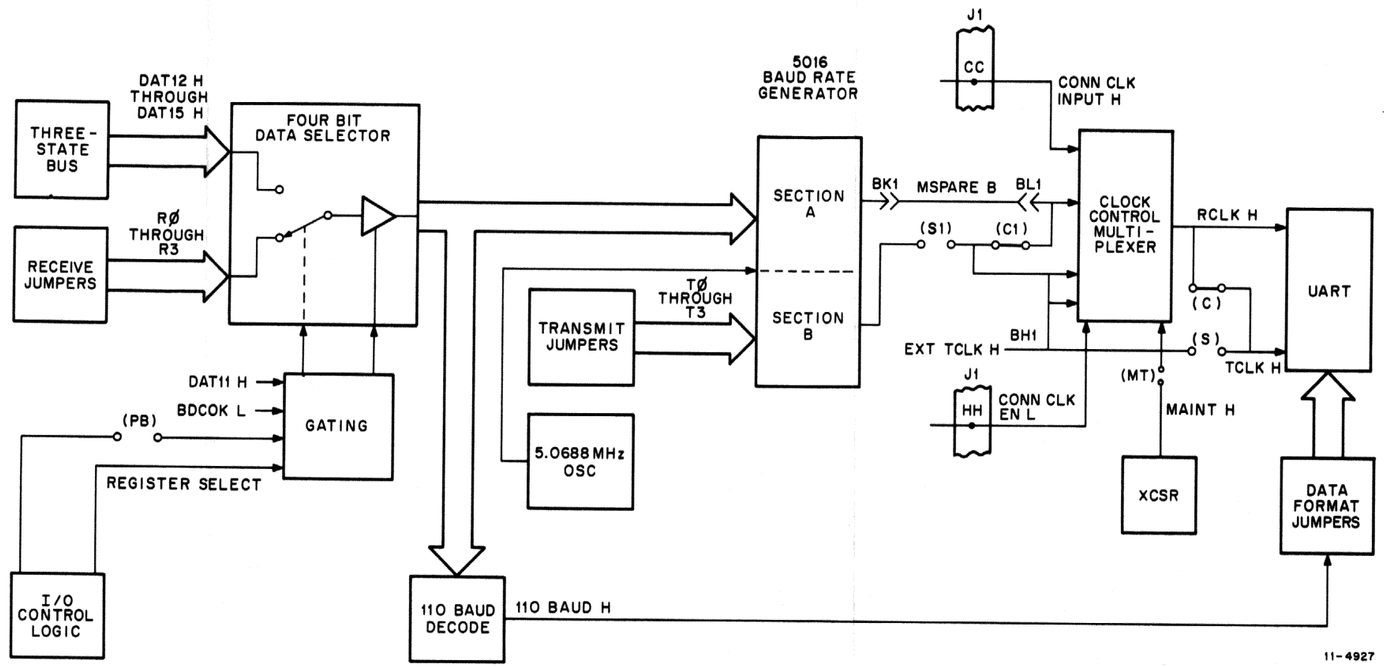


Figure 5-17 Baud Rate Control Signal Flow

5.8.2 Jumper Control

When the Programmable Baud Rate Enable bit is not set, section A of the 5016 chip is controlled by jumpers R0 through R3. This section is used to control the receiver baud rate during split speed operation. During common speed operation, section A (and jumpers R0 through R3) controls both transmitter and receiver baud rates.

Jumpers T0 through T3 always determine the output frequency of section B of the chip. During split speed operation, this establishes the baud rate of the transmitter. During common speed operation, jumpers T0 through T3 and section B of the chip are not used. When the module is operating in its maintenance mode and in split speed, T0 through T3 and section B produce the clock for both the RBUF and the XBUF.

5.8.3 External Control

External clock inputs can be introduced through either the backplane connector or the header connector. Pins BK1 and BL1 are connected together by a jumper (MSPAREB) at each module location on the LSI-11 backplane. The output of section A is routed through this jumper.

The jumper can be cut and an external clock applied to backplane pin BL1. This clock will then drive the receiver in split speed operation, or both the receiver and the transmitter in common speed operation.

An external clock can be used for the transmitter in split speed operation by removing jumper S1 and applying the external clock to backplane pin BH1. External clock frequencies must be 16 times the desired baud rate.

The baud rate can be controlled by an external peripheral device via the cable to the module's header connector. When a TTL logic low enabling signal is applitd to J1 pin HH it causes the clock control multiplexer to select the external clock on pin CC. When the enabling signal is negated the baud rate reverts to its former configuration.

5.8.4 Clock Selection

The receiver and transmitter clock inputs to the RBUF and XBUF timing circuitry (in the UART) are selected by two jumpers and a multiplexer. Normally the multiplexer selects the input from pin BL1 as the receiver clock. The CONN CLK EN L signal, however, causes the multiplexer to select header connector pin CC as the receiver clock. Additionally, during the maintenance mode only, MAINT H causes the multiplexer to choose the transmitter clock as the source of the receiver clock in split speed operation, and the receiver clock as the source of the transmitter clock in common speed operation when jumper MT is installed.

During split speed operation, jumpers S and S1 are inserted and jumpers C and C1 are removed. This routes the receiver clock to the RBUF section of the UART, and the transmitter clock to the XBUF section. For common speed operation, jumpers S and S1 are removed and jumpers C and C1 are inserted. This routes the receiver clock to both the RBUF and XBUF sections of the UART.

Table 5-3 summarizes the possible connections discussed in this section.

5.9 BREAK LOGIC

The break logic performs two functions: it causes a BREAK to be transmitted, and it determines the action taken when a framing error or a BREAK is received.

Table 5-3 UART Clock Sources

Clock Source	Receiver Speed	Transmitter Speed
Dual Baud Rate Generator		
Common Speed	R0–R3	R0–R3
Split Speed	R0–R3	T0–T3
External Clock on Backplane		
Common Speed	BL1	BL1
Split Speed	BL1	BH1
External Clock on Header Connector		
Common Speed Only (Requires Enable on pin HH)	CC	CC

5.9.1 Receive Operation

During normal operation, the UART checks each received character for the proper number of STOP bits. It does this by testing for a marking condition at the appropriate time. If it finds a spacing condition instead, it sets the framing error flag (FR ERR). The BREAK signal is a continuous spacing condition, and is interpreted by the UART as a data character that is missing its STOP bit(s). The UART, therefore, responds to the BREAK signal by asserting FR ERR H (Figure 5-18). MAINT L from the XCSR is gated with FR ERR H to inhibit the framing error signal (FE H) during the maintenance mode.FE H is applied to jumper B, and is inverted and applied to jumper H. If jumper B is inserted and -B (or $\overline{B}$) is removed, FE H will negate control line BDCOK H. BDCOK H indicates to the LSI-11 that dc power is "OK." When FE H negates this signal, it causes the computer to reload its bootstrap.

If jumper B is removed and jumper -B (or $\overline{B}$) is inserted, the computer will not "boot" on a framing error.

If jumper H is inserted, FE H will negate control line BHALT L. This causes the computer to halt when a framing error is received.

CAUTION

If the LSI-11 is using MOS memory, data may be lost when BDCOK H is negated because this action interrupts the memory refresh cycle.

5.9.2 Transmit Operation

To transmit a BREAK signal the program sets the BREAK bit (bit 00) in the XCSR (Figure 5-19). The output of the XCSR latch holding the BREAK bit is used to inhibit the serial data output of the XBUF. This causes the peripheral interface circuitry to transmit a continuous spacing condition (BREAK signal) on the serial communications line.

BREAK generation can be enabled by inserting jumper BG. This allows the state of DAT00 H (BREAK bit) to control the BREAK inhibit gate. When the BREAK bit is set, BREAK(0) L is clocked to a continuous FALSE condition, thus inhibiting the flow of serial data from the XBUF to the peripheral interface.

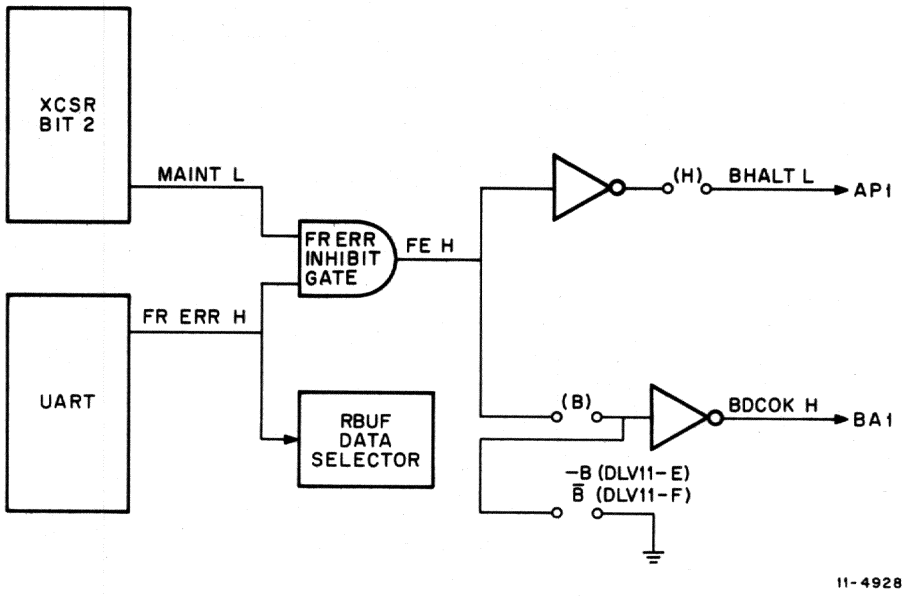


Figure 5-18 Break Logic Receive Signal Flow

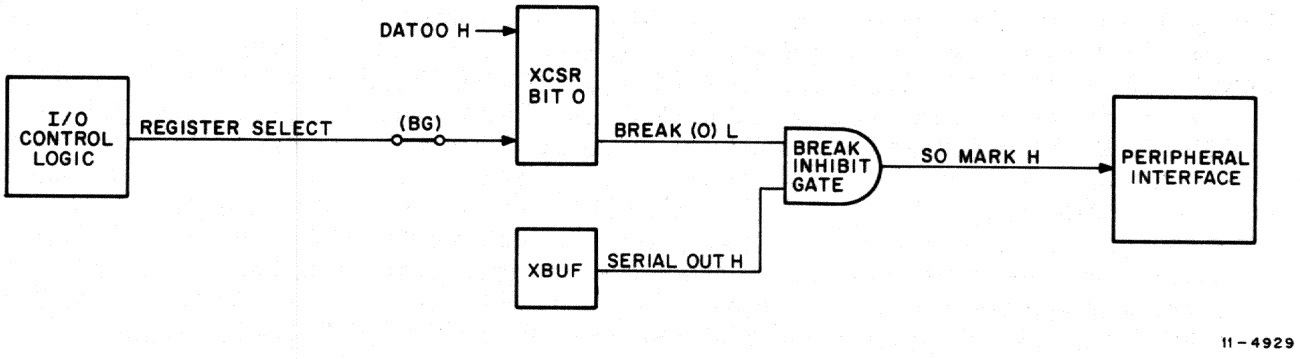


Figure 5-19 Break Logic Transmit Signal Flow

5.10 MAINTENANCE MODE LOGIC

In the maintenance mode, the DLV11-E and DLV11-F modules route their output data back to their input (Figure 5-20). To accomplish this the computer program sets the MAINTENANCE bit in the XCSR. The latch holding this bit has two outputs. One goes to the break logic to prevent the generation of framing error signals during operation in the maintenance mode. The other output is applied to the maintenance mode data selector. The data selector normally routes the incoming data from the peripheral interface to the RBUF. In the maintenance mode, however, it switches its input to the output of the XBUF. This action loops the serial data out of the XBUF back into the RBUF and disconnects the peripheral interface's received data. While in the maintenance mode, the serial output of the XBUF continues to go to the peripheral interface and out to the peripheral device.

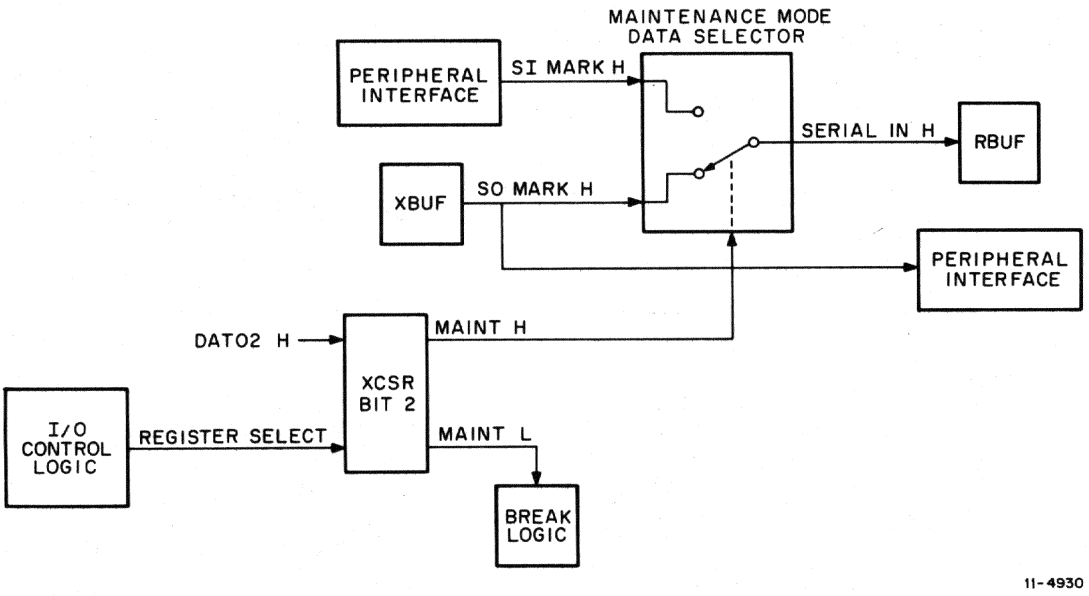


Figure 5-20 Maintenance Mode Logic

5.11 DLV11-E PERIPHERAL INTERFACE

The DLV11-E provides data set control by producing and responding to EIA-compatible control signals. EIA-level receivers in the peripheral interface circuit monitor the following control lines: RING, CLEAR TO SEND, CARRIER, and SECONDARY RECEIVED DATA (Figure 5-21). Each of these control lines is represented by a bit in the RCSR. The peripheral interface will set the DATA SET INTERRUPT bit in the RCSR if RING changes state from a 0 to a 1, or if any of the three other signals changes state from either a 0 to a 1 or a 1 to a 0. Thus, when the computer program has set DSET INT ENB, a signal on any of the incoming EIA control lines can initiate a receiver interrupt. When the interrupt is acknowledged, the program can check the RCSR to determine which signal initiated it. The program can then respond by asserting the appropriate control bits in the RCSR. The peripheral interface responds to a True condition on RCSR bits 1, 2, or 3 (DATA TERMINAL READY, REQUEST TO SEND, and SECONDARY TRANSMITTED DATA, respectively) by transmitting a TRUE condition on the corresponding EIA control line. (If the data set has a FORCE BUSY function, jumper FB should be inserted to drive this control line with the REQUEST TO SEND bit.) The exchange of control signals allows a remote data set to establish a channel of communication with the LSI-11 through the use of a handshake.

A typical handshake sequence proceeds as follows: A remote data set calls the local data set. The local data set sends a RING signal to the DLV11-E asynchronous line interface. The RING signal initiates a receiver interrupt (assuming DSET INT ENB is set). The program reads the RCSR and determines that the interrupt was caused by the RING signal. Then, through a service routine, it issues the DATA TERMINAL READY and REQUEST TO SEND signals. These signals direct the local data set to answer the remote data set by sending it a carrier signal. The remote data set acknowledges the carrier signal by returning its own carrier signal. The local data set detects the remote data set's carrier signal and indicates this to the DLV11-E by asserting its CARRIER control line. This causes another receiver interrupt. Upon recognizing the CARRIER-caused interrupt, the program can either receive or transmit data. The only prerequisites for this handshaking sequence are that the program use appropriate service routines and that the data set interrupt enable bit be set in the RCSR.

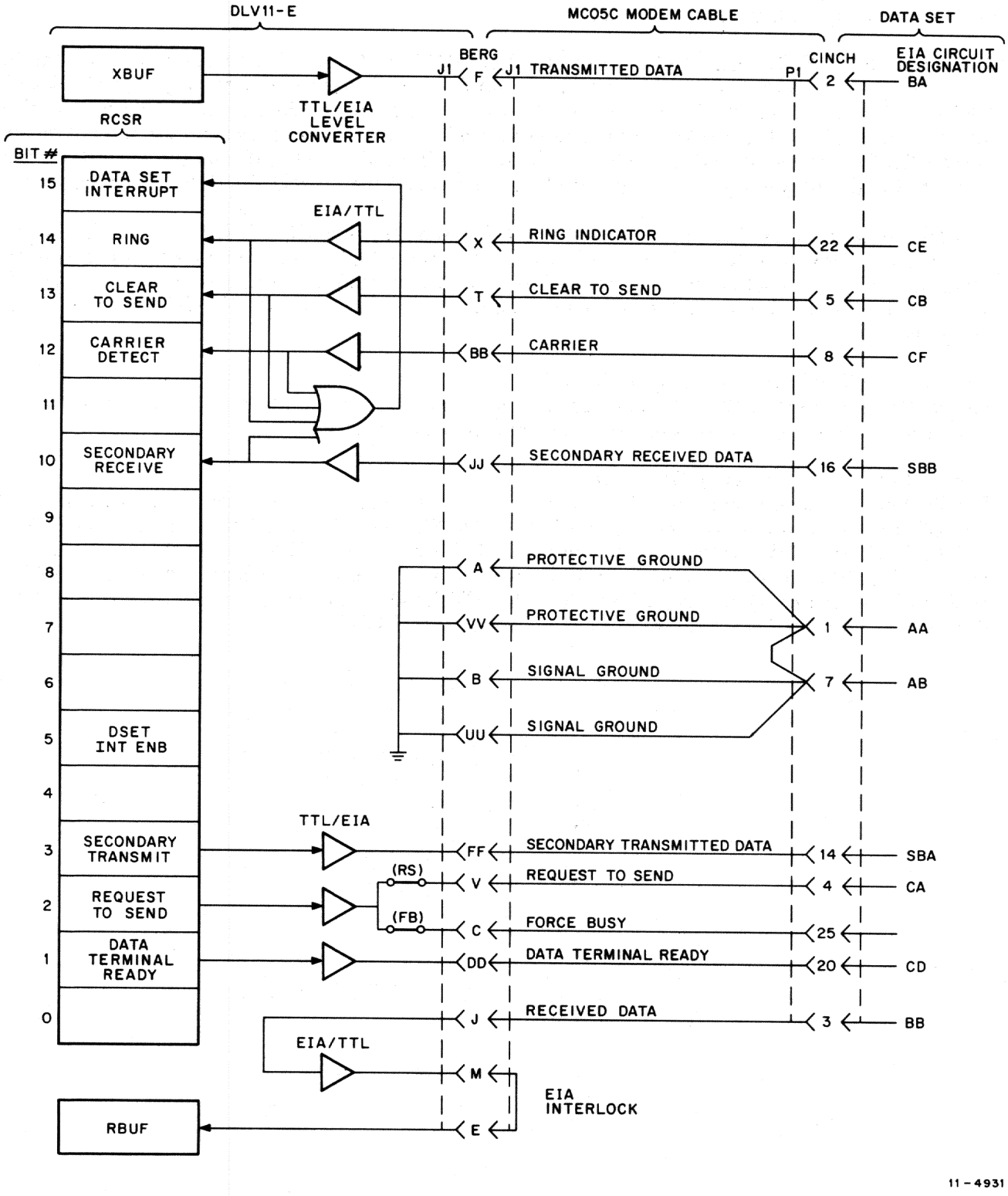


Figure 5-21 DLV11-E Peripheral Interface Signal Flow

Other exchanges involving CLEAR TO SEND and SECONDARY RECEIVED DATA may be programmed, as required, by the equipment.

SECONDARY RECEIVED DATA and SECONDARY TRANSMITTED DATA are provided for the exchange of secondary or supervisory data with data sets having this capability. SECONDARY RECEIVED DATA allows the remote data set to set one bit in the RCSR and to cause a receiver interrupt. SECONDARY TRANSMITTED DATA allows the LSI-11 to transmit the state of one bit in the RCSR to the remote data set. These exchanges involve only two RCSR bits and are independent of normal data exchanges between the peripheral device and the DLV11-E's data buffer registers.

EIA-level data from the data set arrives at the DLV11-E on the RECEIVED DATA line. The peripheral interface converts it to TTL levels and routes it to the RBUF. Data to be transmitted from the computer to the data set is serialized in the XBUF and then routed to the peripheral interface. The interface circuitry converts it to the EIA-levels and transmits it out the TRANSMITTED DATA line to the data set.

5.12 DLV11-F PERIPHERAL INTERFACE

The DLV11-F supports either EIA data leads ("Data Leads Only" operation) or 20 mA current loops. It does not perform handshakes or exchange control signals with data sets.

5.12.1 EIA Data Leads Only Operation

The DLV11-F does not monitor EIA control lines but it does, however, hold three outgoing EIA control lines in a continuous TRUE condition. REQUEST TO SEND, FORCE BUSY, and DATA TERMINAL READY are held continuously TRUE by separate EIA drivers (Figure 5-22). The peripheral interface converts data from TTL levels to EIA levels for transmission on the TRANSMITTED DATA line. Data received over the RECEIVED DATA line is converted from EIA levels to TTL levels and routed through an interlock jumper to the RBUF.

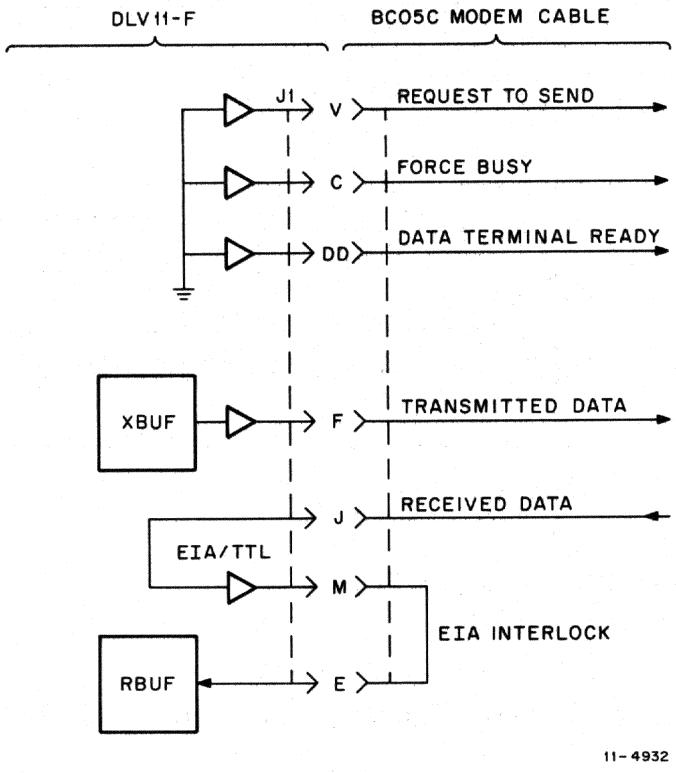


Figure 5-22 Data Lead Only Interface

5.12.2 Current Loop Operation

The peripheral interface directly interfaces terminal devices that use 20 mA current loops. It provides current for receiver and transmitter circuits, and also controls the paper tape reader on teleprinters equipped with a Reader Run relay. Both the transmitter and receiver circuits use neutral current loops, in that current flows in only one direction (as opposed to polar current loops, in which it flows either way).

The transmitter can be jumpered for active operation by inserting jumpers 4A and 5A (Figure 5-23), or for passive operation by inserting jumpers 3P and 4P. In active operation, the transmitter provides 20 mA (nominal) current to loop through the peripheral device. The current is switched on and off by data bits from the XBUF.

In passive operation, data bits from the XBUF are optically isolated from the transmission lines. Through the isolator, the data controls a switching circuit that switches the 20 mA current on and off. In passive operation, the peripheral device provides the power for the current flow.

The reader run circuit supplies a negative voltage (approximately -12 V) and a positive voltage (approximately +5 V) to energize the peripheral device's reader run relay. If the READER ENABLE bit is set in the RCSR, the reader run circuit causes the peripheral terminal's paper tape reader to advance. When the START bit of the next character is received, the Receiver Active circuit asserts RCVR BUSY H. RCVR BUSY H clears the reader enable bit, thereby switching off the current to the peripheral terminal's reader run relay. The reader run bit must be set again by the program before the reader run circuit can drive the relay again.

The receiver circuit can be jumpered to be either active or passive. When configured for active operation (Figure 5-24) the circuit supplies a ground and a positive voltage to the peripheral device. When jumpered for passive operation (Figure 5-25), the receiver uses power supplied by the peripheral device. In either case, current passes through an optical isolator. The isolator produces a TTL output that is electrically isolated from the current loop. The TTL output is routed to the RBUF.

The RBUF accepts TTL inputs from either the EIA interface circuit or the 20 mA circuit. The routing is determined by the cable attached to the 40-pin header connector (Figure 5-26). An EIA modem cable will jumper the output of the 20 mA receiver to the input of the RBUF.

5.13 DC-TO-DC POWER INVERTER

The power inverter operates on +12 V from the LSI-11 power supply and produces -12 V for the UART, the EIA drivers, and, on the DLV11-F, the reader run circuit. The power inverter circuit consists of an oscillator driving a charge pump.

The output of the oscillator is capacitively coupled to a rectifier, which develops a negative-going output. This output pumps up an inductive charge storage network and is Zener-regulated back to -12 V.

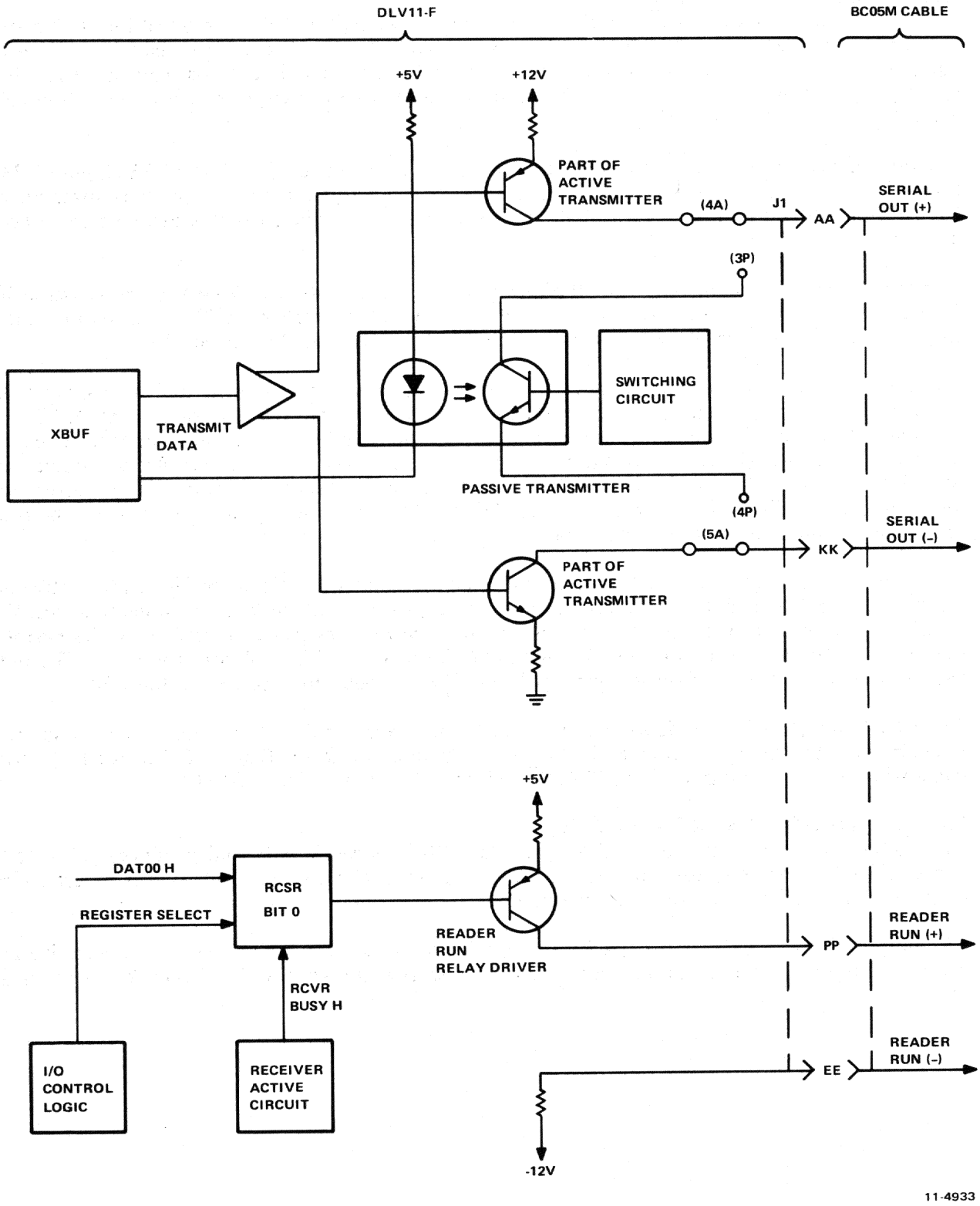


Figure 5-23 20 mA Transmitter and Reader Run Circuit

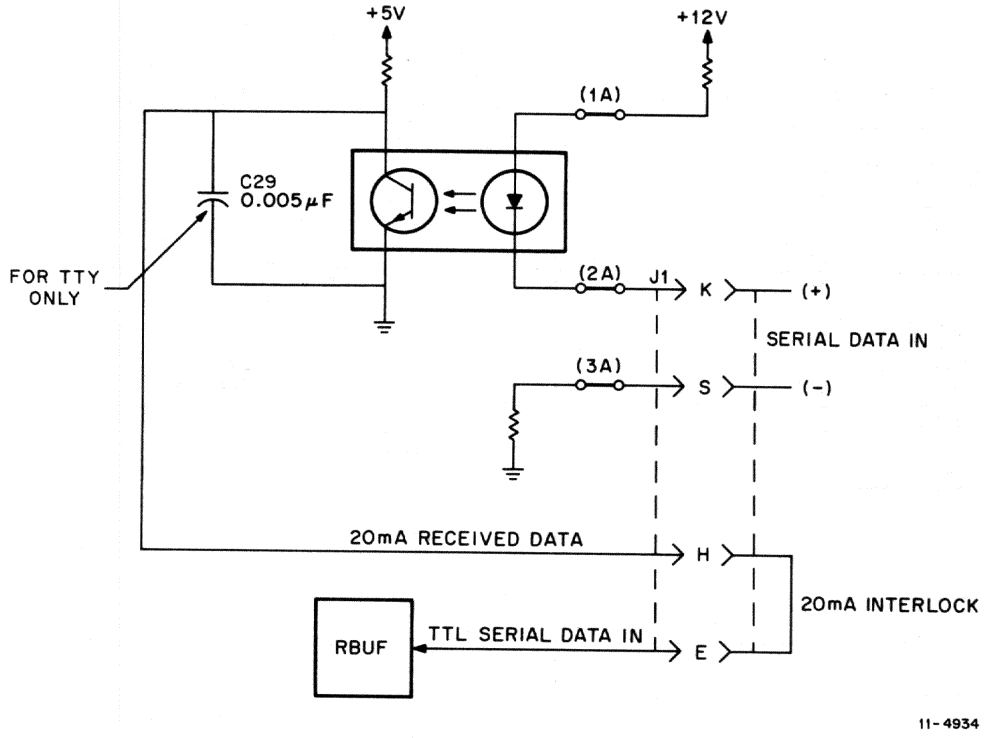


Figure 5-24 Active Receive 20 mA Current Loop

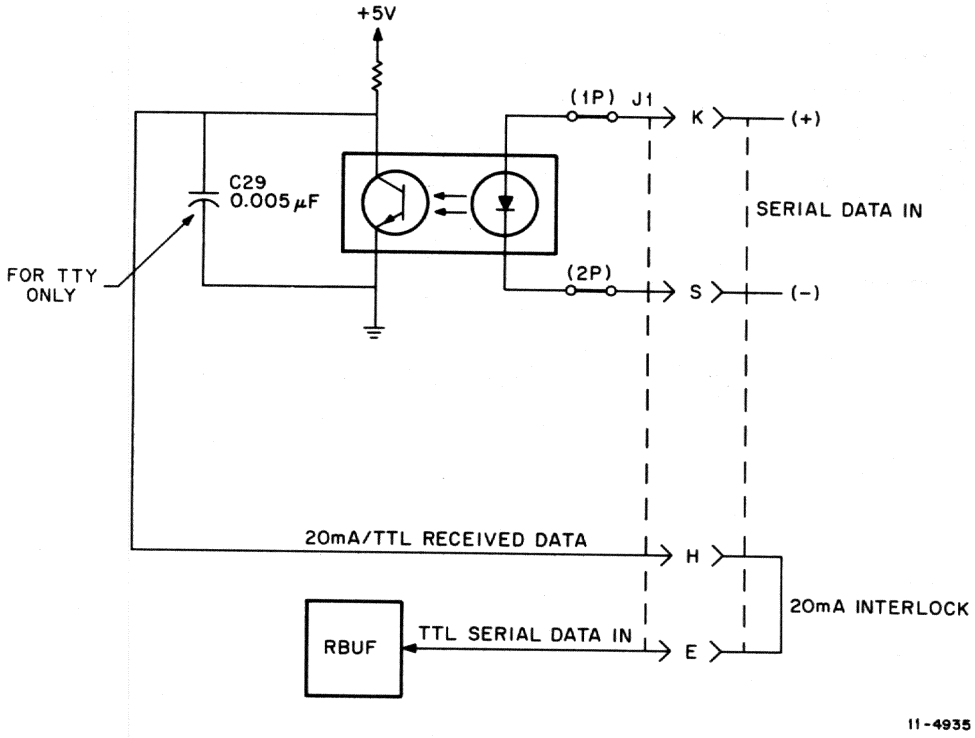


Figure 5-25 Passive Receive 20 mA Current Loop

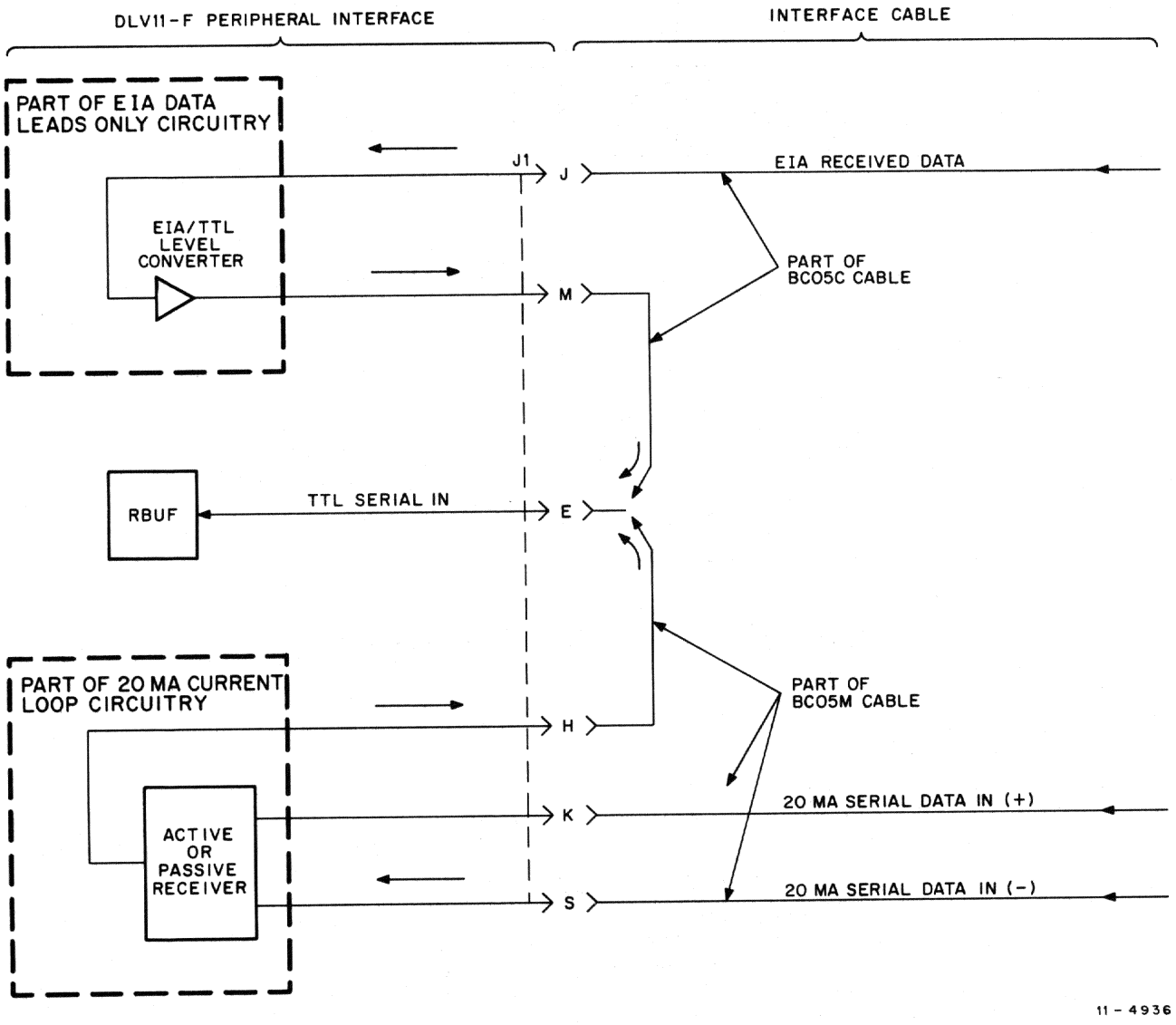


Figure 5-26 Interlock Jumper Data Flow

APPENDIX A
IC DESCRIPTIONS

A.1 DC003 INTERRUPT LOGIC

The interrupt chip is an 18-pin DIP device that provides the circuits to perform an interrupt transaction in a computer system that uses a "pass-the-pulse" type arbitration scheme. The device is used in peripheral interfaces and provides two interrupt channels labeled "A" and "B," with the A section at a higher priority than the B section. Bus signals use high-impedance input circuits or high-drive open-collector outputs, which allows the device to directly attach to the computer systems bus. Maximum current required from the V_{cc} supply is 140 mA.

Figure A-1 is a simplified logic diagram of the DC003 IC. Timing for the A interrupt section is shown in Figure A-2, while Figure A-3 shows the timing for both A and B interrupt sections. Table A-1 describes the signals and pins of the DC003 by pin and signal name.

A.2 DC004 PROTOCOL LOGIC

The protocol chip is a 20-pin DIP device that functions as a register selector, providing the signals necessary to control data flow into and out of up to four word registers (8 bytes). Bus signals can directly attach to the device because receivers and drivers are provided on the chip. An RC delay circuit is provided to slow the response of the peripheral interface to data transfer requests. The circuit is designed such that if tight tolerance is not required, then only an external 1K +20 percent resistor is necessary. External RCs can be added to vary the delay. Maximum current required from the V_{cc} supply is 120 mA.

Figure A-4 is a simplified logic diagram of the DC004 IC. Signal timing with respect to different loads are tabularized in Table A-2 and are shown in Figure A-5. Figure A-6 shows the loading for the test conditions in Table A-2. Signal and pin definitions for the DC004 are presented in Table A-3.

A.3 DC005 TRANSCEIVER LOGIC

The 4-bit transceiver is a 20 pin DIP, low-power Schottky device for primary use in peripheral device interfaces, functioning as a bidirectional buffer between a data bus and peripheral device logic. In addition to the isolation function, the device also provides a comparison circuit for address selection and a constant generator, useful for interrupt vector addresses. The bus I/O port provides high-impedance inputs and high-drive (70 mA) open-collector outputs to allow direct connection to a computer's data bus structure. On the peripheral device side, a bidirectional port is also provided, with standard TTL inputs and 20 mA tristate drivers. Data on this port is the logical inversion of the data on the bus side.

Three address jumper inputs are used to compare against three bus inputs and to generate the signal MATCH. The MATCH output is open-collector, which allows the output of several transceiver's to be wired-anded to form a composite address match signal. The address jumpers can also be put into a third logical state that disconnects that jumper from the address match, allowing for "don't care" address bits. In addition to the three address jumper inputs, a fourth high-impedance input line is used to enable/disable the MATCH output.

Three vector jumper inputs are used to generate a constant that can be passed to the computer bus. The three inputs directly drive three of the bus lines, overriding the action of the control lines.

Two control signals are decoded to give three operational states: receive data, transmit data, and disable.

Maximum current required from the V_{cc} supply is 100 mA.

Figure A-7 is a simplified logic diagram of the DC005 IC. Timing for the various functions is shown in Figure A-8. Signal and pin definitions for the DC005 are presented in Table A-4.

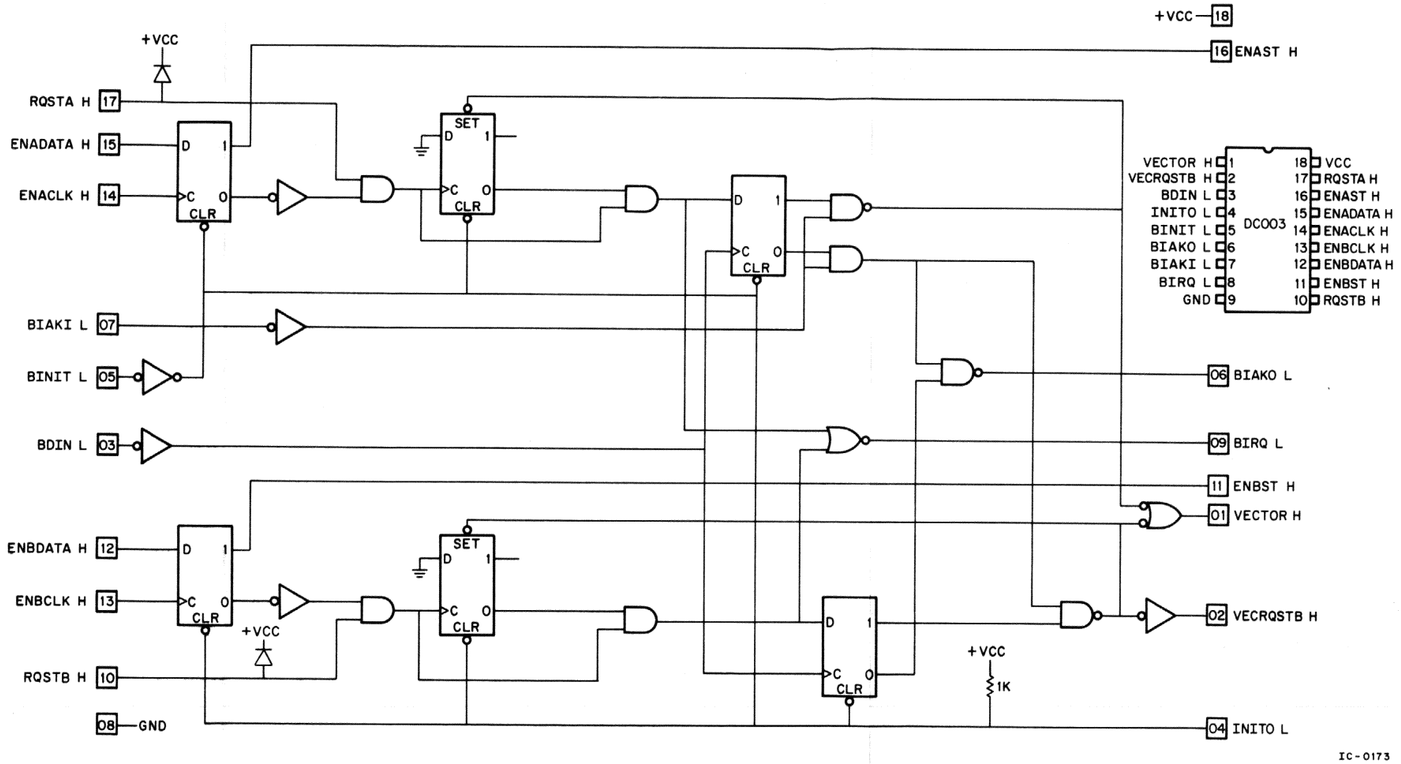


Figure A-1 DC003 Simplified Logic Diagram

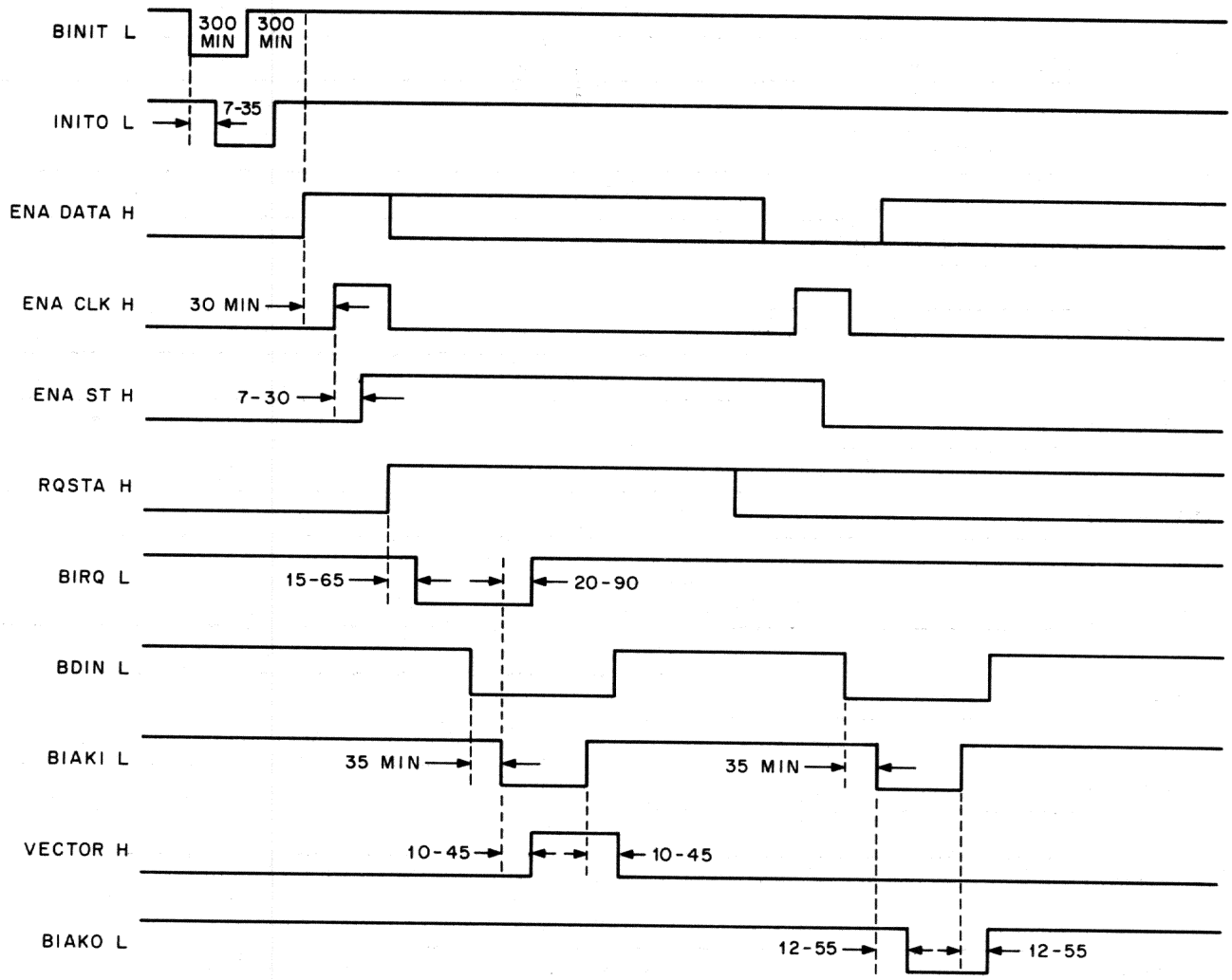


Figure A-2 DC003 "A" Interrupt Section Timing Diagram

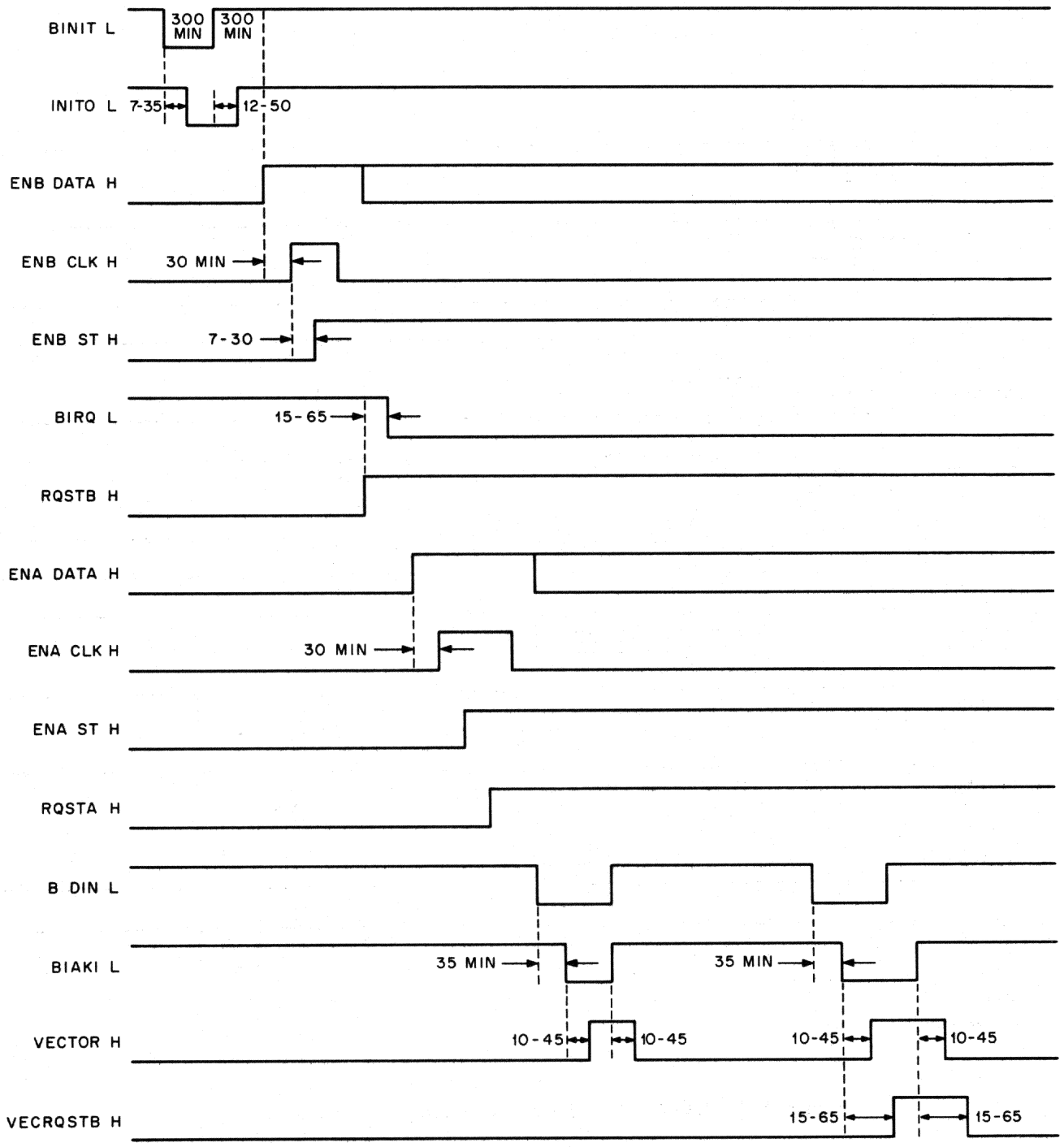


Figure A-3 DC003 "A" and "B" Interrupt Section Timing Diagram

Table A-1 DC003 Pin/Signal Descriptions

Pin	Signal	Description
1	VECTOR H	INTERRUPT VECTOR GATING signal. This signal should be used to gate the appropriate vector address onto the bus and to form the bus signal called BRPLY L.
2	VEC RQSTB H	VECTOR REQUEST "B" signal. When asserted, indicates RQST "B" service vector address is required. When unasserted, indicates RQST "A" service vector address is required. VECTOR H is the gating signal for the entire vector address. VEC RQST B H is normally bit 2 of the vector address.
3	BDIN L	BUS DATA IN. This signal, generated by the processor BDIN, always precedes a BIAK signal.
4	INITO L	INITIALIZE OUT signal. This is the buffered BINIT L signal used in the device interface for general initialization.
5	BINIT L	BUS INITIALIZE signal. When asserted, this signal brings all driven lines to their unasserted state (except INITO L).
6	BIAKO L	BUS INTERRUPT ACKNOWLEDGE signal (OUT). This signal is the daisy-chained signal that is passed by all devices not requesting interrupt service (see BIAKI L). Once passed by a device, it must remain passed until a new BIAKI L is generated.
7	BIAKI L	BUS INTERRUPT ACKNOWLEDGE signal (IN). This signal is the processor's response to BIRQ L true. This signal is daisy-chained such that the first requesting device blocks the signal propagation while nonrequesting devices pass the signal on as BIAKO L to the next device in the chain. The leading edge of BIAKI L causes BIRQ L to be unasserted by the requesting device.
8	BIRQ L	ASYNCHRONOUS BUS INTERRUPT REQUEST from a device needing interrupt service. The request is generated by a true RQST signal along with the associated true interrupt enable signal. The request is removed after the acceptance of the BDIN L signal and on the leading edge of the BIAKI L signal, or the removal of the associated interrupt enable, or due to the removal of the associated request signal.
10 17	REQSTB H REQSTA H	DEVICE INTERRUPT REQUEST SIGNAL. When asserted, with the enable "A" flip-flop asserted, will cause the assertion of BIRQ L on the bus. This signal line normally remains asserted until the request is serviced.

Table A-1 DC003 Pin/Signal Descriptions (Cont)

Pin	Signal	Description
11 16	ENB ST H ENA ST H	INTERRUPT ENABLE "A" STATUS signal. This signal indicates the state of the interrupt enable "A" internal flip-flop, which is controlled by the signal line ENA DATA H and the ENA CLK H clock line.
12 15	ENB DATA H ENA DATA H	INTERRUPT ENABLE "A" DATA signal. The level on this line, in conjunction with the ENA CLK H signal, determines the state of the internal interrupt enable "A" flip-flop. The output of this flip-flop is monitored by the ENA ST H signal.
13 14	ENB CLK H ENA CLK H	INTERRUPT ENABLE "A" CLOCK. When asserted (on the positive edge), interrupt enable "A" flip-flop assumes the state of the ENA DATA H signal line.

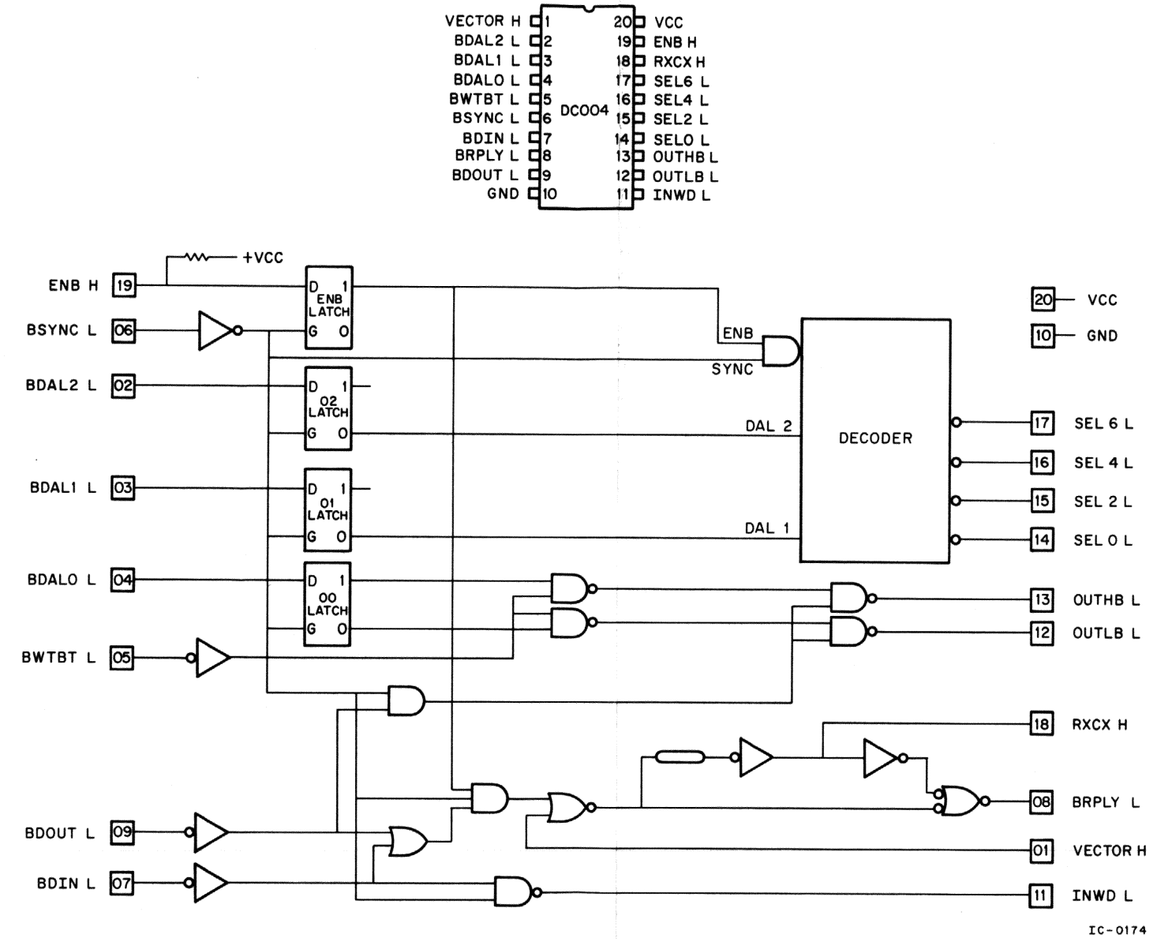


Figure A-4 DC004 Simplified Logic Diagram

Table A-2 DC004 Signal Timing vs Output Loading

	Signal	With Respect to Signal	Test Cond.	Output Being Asserted Min (ns)	Output Being Asserted Max (ns)	Output Being Asserted Min (ns)	Output Being Asserted Max (ns)	Figure A-5 Ref.
	Sel (0,2,4,6) L	BSYNC L	Load B	15	35	5	25	t_5, t_6
			Load C	15	40	5	30	
	OUTLB L	BDOUT L	Load B	5	25	5	25	t_9, t_{10}
			Load C	5	30	5	30	
	OUTHB L	DBOUT L	Load B	5	25	5	25	t_9, t_{10}
			Load C	5	30	5	30	
	INWD L	BDIN L	Load A	5	25	5	25	t_{11}, t_{12}
			Load B	5	30	5	30	
Pin 18 Connection RX = 1K ±5% 350Ω ±5% 15 pf ±5%	BRPLY L (Load A)	OUTLB L (Load B)		20	60	-10	45	t_{13}, t_{14}
	BRPLY L (Load A)	OUTHB L (Load B)		20	60	-10	45	t_{13}, t_{14}
	BRPLY L (Load A)	INWD L (Load B)		20	60	-10	45	t_{13}, t_{14}
	BRPLY L (Load A)	VECTOR H		30	70	0	45	t_{13}, t_{14}
Pin 18 Connection RX = 4.64K ±1%	BRPLY L (Load A)	OUTLB L (Load B)		300	400	-10	45	t_{13}, t_{14}
	BRPLY L (Load A)	OUTHBL (Load B)		300	400	-10	45	t_{13}, t_{14}
CX = 220 pf ±1%	BRPLY L (Load A)	INWD L (Load B)		300	400	-10	45	t_{13}, t_{14}
	BRPLY L (Load A)	VECTOR H		330	430	0	45	t_{13}, t_{14}

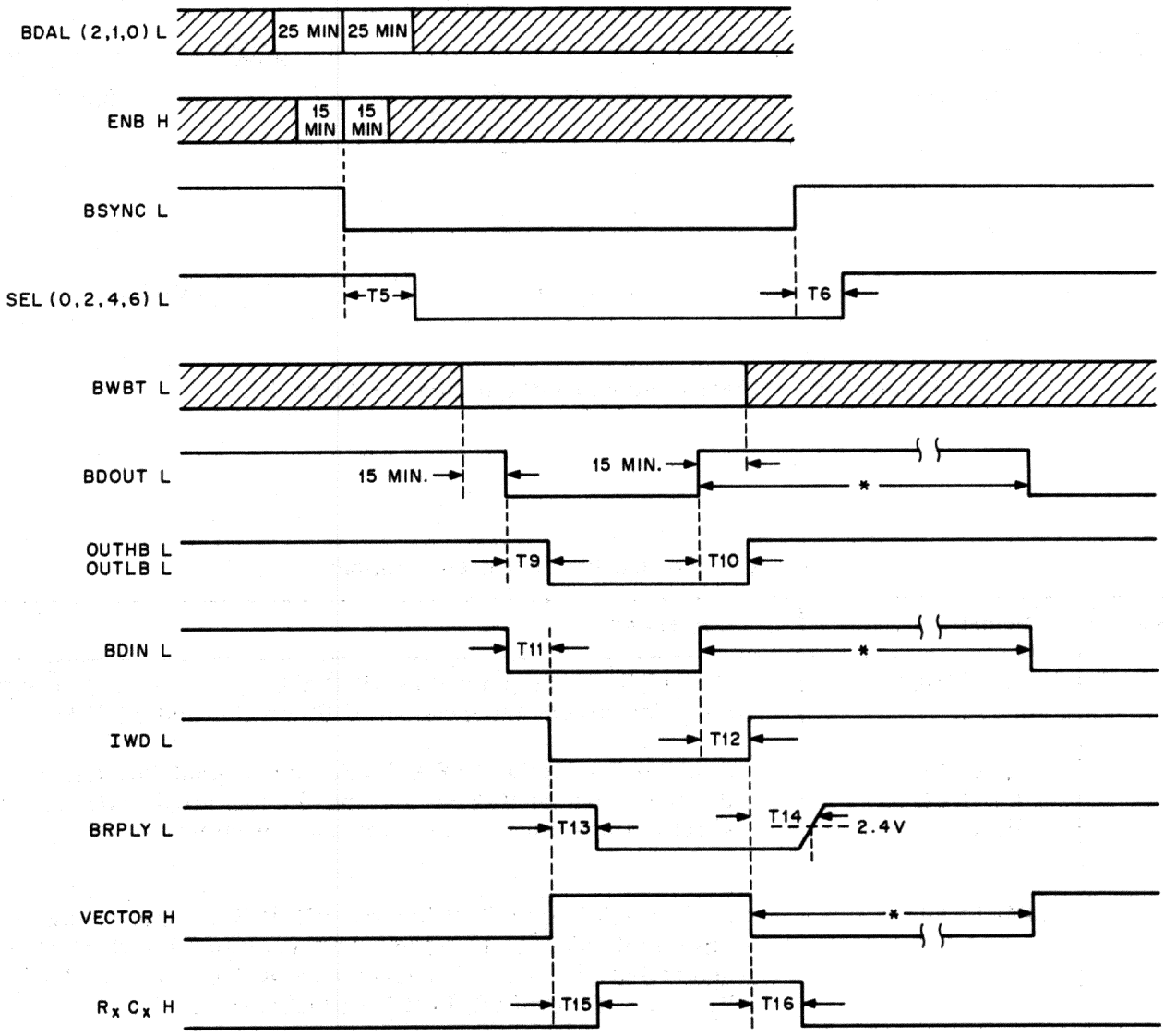


Figure A-5 DC004 Timing Diagram

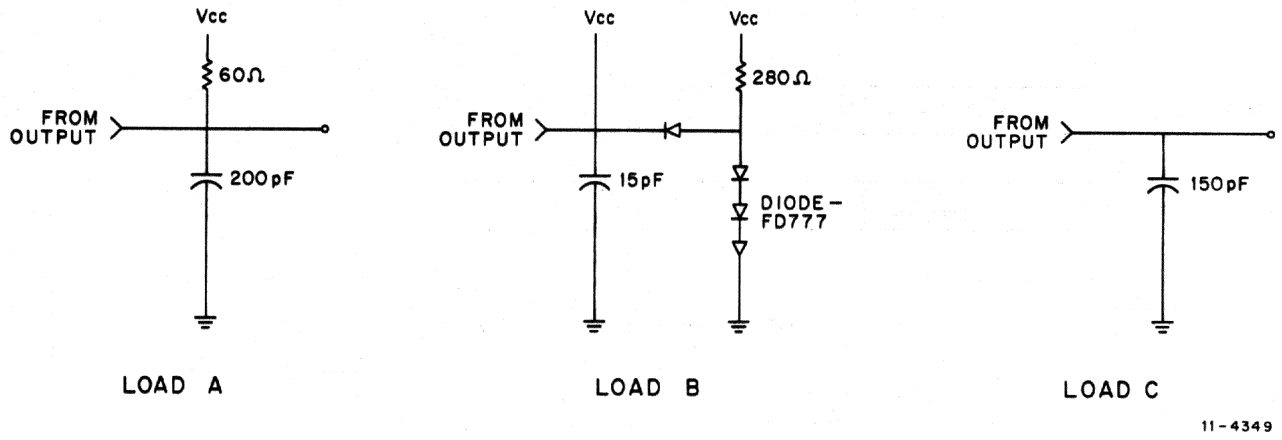


Figure A-6 DC004 Loading Configuration for Table A-2

Table A-3 DC004 Pin/Signal Descriptions

Pin	Signal	Description
1	VECTOR H	VECTOR. This input causes BRPLY L to be generated through the delay circuit. Independent of BSYNC L and ENB H.
2 3 4	BDAL2 L BDAL1 L BDAL0 L	BUS DATA ADDRESS LINES. These signals are latched at the assert edge of BSYNC L. Lines 2 and 1 are decoded for the select outputs; line 0 is used for byte selection.
5	BWTBT L	BUS WRITE/BYTE. While the BDOUT L input is asserted, this signal indicates a byte or word operation: Asserted = byte, unasserted = word. Decoded with B OUT L and latched BDAL0 L to form OUTLB L and OUTHB L.
6	BSYNC L	BUS SYNCHRONIZE. At the assert edge of this signal, address information is trapped in four latches. While unasserted, disables all outputs except the vector term of BRPLY L.
7	BDIN L	BUS DATA IN. This is a strobing signal to effect a data input transaction. Generates BRPLY L through the delay circuit and INWD L.

Table A-3 DC004 Pin/Signal Descriptions (Cont)

Pin	Signal	Description
8	BRPLY L	BUS REPLY. This signal is generated through an RC delay by VECTOR H, and strobed by BDIN L or BDOUT L, and BSYNC L and latched ENB H.
9	BDOUT L	BUS DATA OUT. This is a strobing signal to effect a data output transaction. Decoded with BWTBT L and BDALO to form OUTLB L and OUTHB L. Generates BRPLY L through the delay circuit.
11	INWD L	IN WORD. Used to gate (read) data from a selected register on to the data bus. Enabled by BSYNC L and strobed by BDIN L.
12 13	OUTHB L OUTLB L	OUT LOW BYTE, OUT HIGH BYTE. Used to load (write) data into the lower, higher, or both bytes of a selected register. Enabled by BSYNC L and decode of BWTBT L and latched BDAL0 L, and strobed by BDOUT L.
14 15 16 17	SEL0 L SEL2 L SEL4 L SEL6 L	SELECT LINES. One of these four signals is true as a function of BDAL2 L and BDAL1 L if ENB H is asserted at the assert edge of BSYNC L. They indicate that a word register has been selected for a data transaction. These signals never become asserted except at the assertion of BSYNC L (then only if ENB H is asserted at that time) and, once asserted, are not unasserted until BSYNC L becomes unasserted.
18	RXCX	EXTERNAL RESISTOR CAPACITOR NODE. This node is provided to vary the delay between the BDIN L, BDOUT L, and VECTOR H inputs and BRPLY L output. The external resistor should be tied to VCC and the capacitor to ground. As an output, it is the logical inversion of BRPLY L.
19	ENB H	ENABLE. This signal is latched at the asserted edge of BSYNC L and is used to enable the select outputs and the address term of BRPLY L.

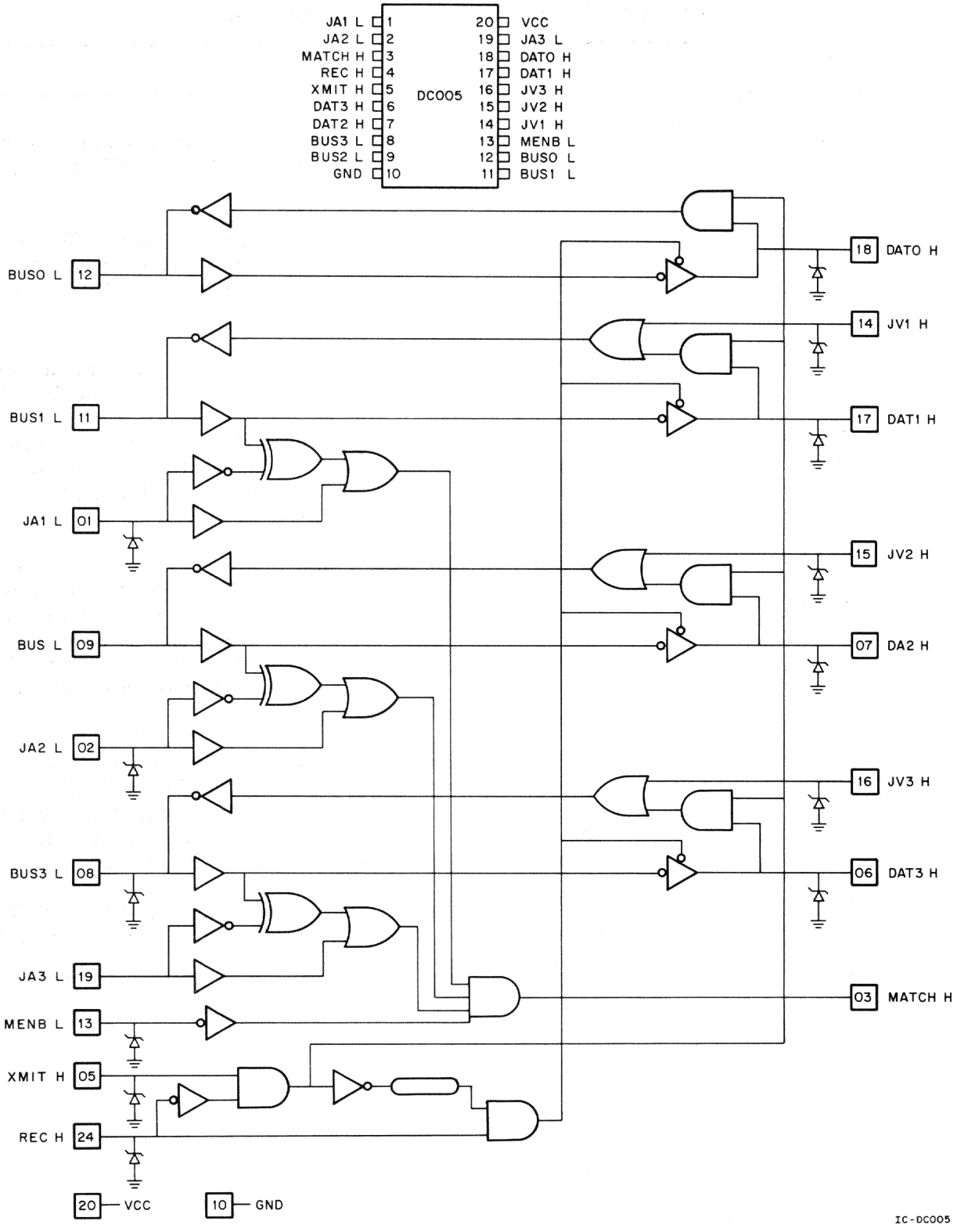


Figure A-7 DC005 Simplified Logic Diagram

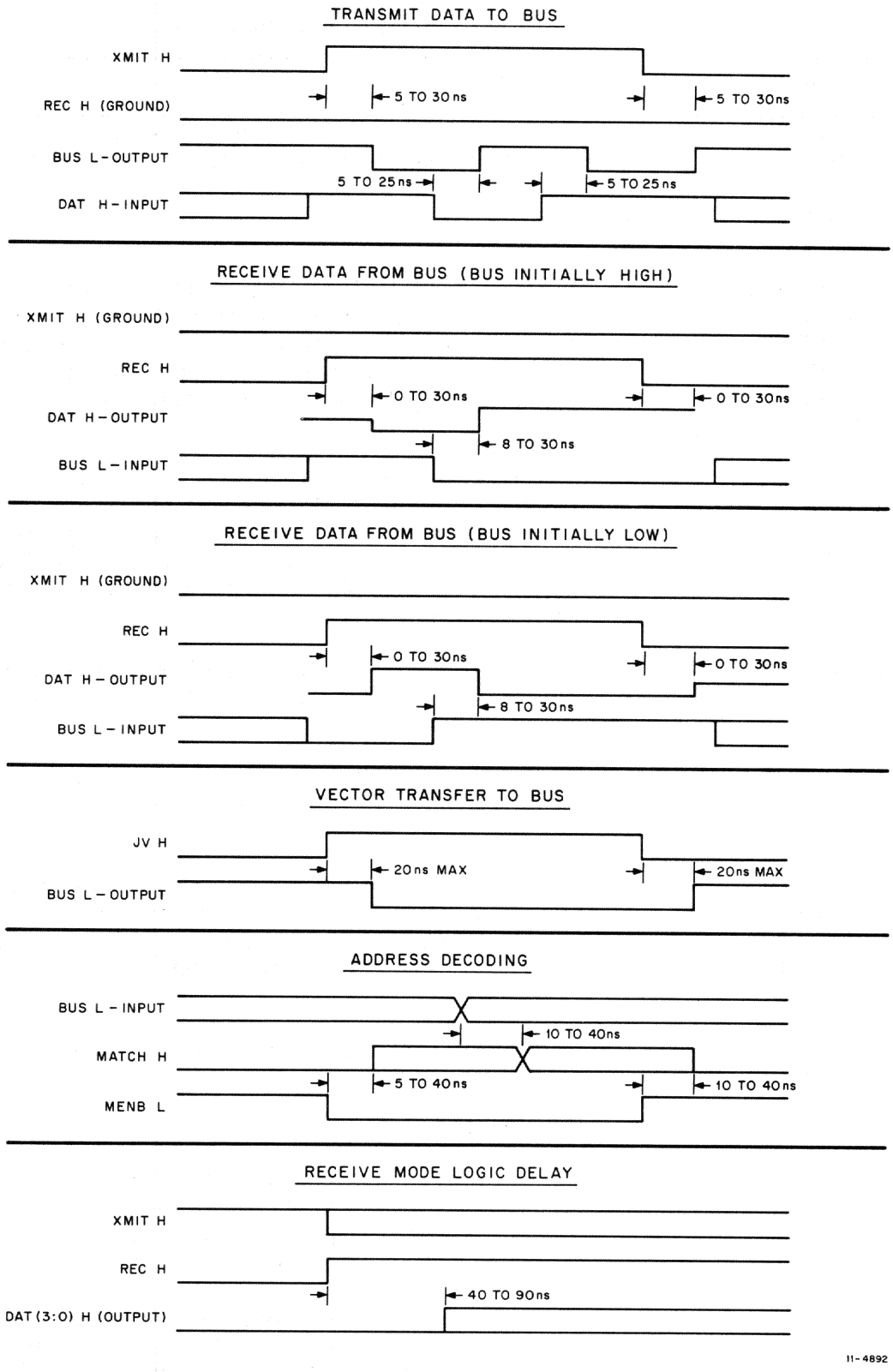


Figure A-8 DC005 Timing Diagram

Table A-4 DC005 Pin/Signal Descriptions

Pin	Name	Function
	BUS(3:0) L	BUS DATA. This set of four lines constitutes the bus side of the transceiver. Open-collector outputs; high-impedance inputs. LOW = 1.
12	BUS0	
11	BUS1	
9	BUS2	
8	BUS3	
	DAT(3:0) H	PERIPHERAL DEVICE DATA. These four tri-state lines carry the inverted received data from BUS (3:0) when the transceiver is in the receive mode. When in transmit data mode, the data carried on these lines is passed inverted to BUS (3:0). When in the disabled mode, these lines go open (hi-z). HIGH = 1.
18	DAT0	
17	DAT1	
7	DAT2	
6	DAT3	
	JV (3:1) H	VECTOR JUMPERS. These inputs, with internal pull-down resistors, directly drive BUS (3:1). A low or open on the jumper pin will cause an open condition on the corresponding BUS pin if XMIT H is low. A high will cause a one (low) to be transmitted on the BUS pin. Note that BUS0 L is not controlled by any jumper input.
14	JV1	
15	JV2	
16	JV3	
13	MENB L	MATCH ENABLE. A low on this line will enable the MATCH output. A high will force MATCH low, overriding the match circuit.
3	MATCH H	ADDRESS MATCH. When BUS (3:1) match with the state of JA (3:1) and MENB L is low, this output is open; otherwise, it is low.
	JA (3:1) L	ADDRESS JUMPERS. A strap to ground on these inputs will allow a match to occur with a one (low) on the corresponding BUS line; an open will allow a match with a zero (high); a strap to V_{cc} will disconnect the corresponding address bit from the comparison.
1	JA1 L	
2	JA2 - L	
19	JA3 - L	
5	XMIT H	CONTROL INPUTS. These lines control the operational of the transceiver as follows:
4	REC H	**REC XMIT** 0 0 DISABLE: BUS, DAT open 0 1 XMIT DATA; DAT Bus 1 0 RECEIVE: BUS DAT 1 1 RECEIVE: BUS DAT To avoid tristate overlap conditions, an internal circuit delays the change of modes between XMIT DATA mode, and delays tristate drivers on the DAT lines from enabling. This action is independent of the DISABLE mode.

A.4 UNIVERSAL ASYNCHRONOUS RECEIVER/TRANSMITTER

The Universal Asynchronous Receiver/Transmitter (UART) is an LSI subsystem that accepts binary characters from either a terminal device or a computer and receives or transmits these characters with appended control and error detecting bits. In order to make this subsystem universal, the baud rate, bits per word, parity mode, and number of STOP bits are selected by external logic circuits.

The UART is a full duplex receiver/transmitter. The receiver section accepts asynchronous serial binary characters and converts them to a parallel format. The transmitter section accepts parallel binary characters from the bus and converts them to a serial asynchronous output with START and STOP bits added.

All UART characters contain a START bit, five to eight DATA bits, one or two STOP bits, and a PARITY bit which may be odd, even, or turned off. The STOP bits are opposite in polarity to the START bit. Refer to Figure A-9.

Both the receiver and transmitter are double buffered. The UART internally synchronizes the START bit with the clock input to ensure a full 16-element (clock periods) START bit independent of the time of data loading. Transmitter distortion (assuming perfect clock input) is less than 3 percent on any bit up to 10K baud. The receiver strobes the input bit within ±8 percent of the theoretical center of the bit. The receiver also rejects any START bit that lasts less than one-half of a bit time.

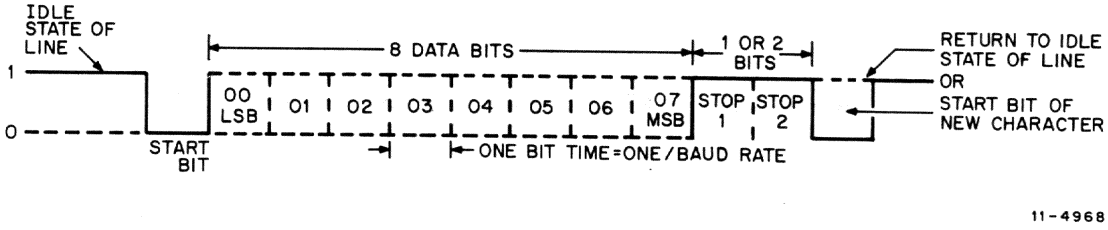


Figure A-9 UART Data Format

A.4.1 Receiver Operation

A block diagram of the UART receiver is shown in Figure A-10. When the receiver is in the idle state, it samples the serial input line (SERIAL IN, pin 20) at the selected clock edges (R CLK, pin 17) after the first mark-to-space transition of the serial input line. If the first sample is a mark (high), the receiver returns to the idle state and is ready to detect another mark-to-space transition. If, however, the first sample is a space (low), then the receiver enters the data entry state.

If the receiver control logic has **not** been conditioned to the **no** parity state (a low on pin 35), then the receiver checks the parity of the data bits plus the parity bit following the data bits and compares it with the parity sense on the parity select line (pin 39). If the parity sense of the received character differs from the parity of the UART control logic, then the receive parity error line (P ERR, pin 13) goes high and causes the P ERR bit in the RBUF register to set.

If the receiver control logic **has** been conditioned to the **no** parity state (a high on pin 35), then the receiver takes no action with respect to parity and maintains the parity error line (P ERR, pin 13) in the FALSE (low) state. When the control logic senses a parity error, it generates a P ERR signal. The DATA AVAILABLE signal updates the parity error indicator.

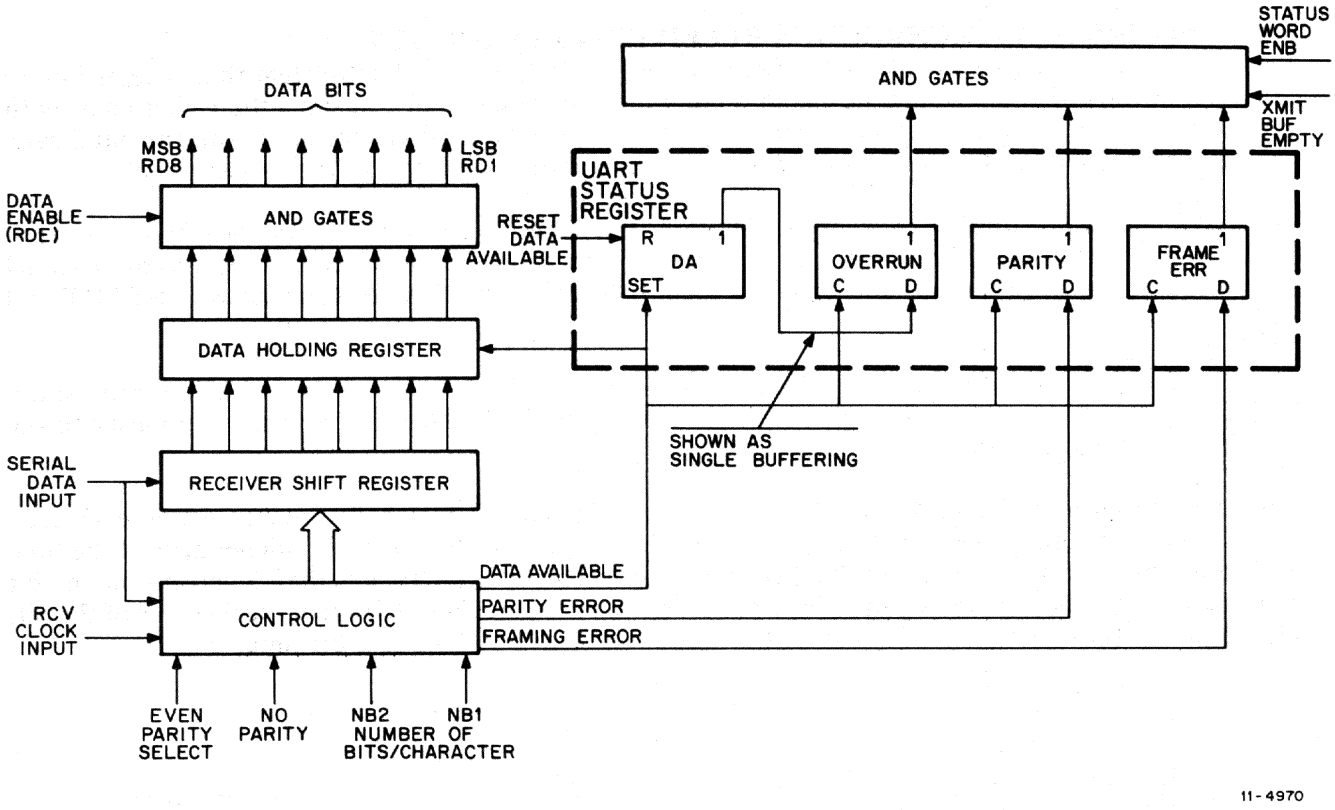


Figure A-10 UART Receiver – Block Diagram

The receiver samples the first STOP bit that occurs either after the PARITY bit, or after the data bits if no parity is selected. If a valid (high) STOP bit exists, no further action is taken. If, however, the STOP bit is FALSE (low), indicating an invalid STOP code, then the UART control logic provides a framing error indication (a high on FR ERR, pin 14).

Because the serial input from the external device is shifted into the UART a bit at a time (SI, pin 20), occurrence of a STOP code indicates that the entire data character has been received and shifted into the receiver shift register. After the STOP bit has been sampled, the receiver control logic parallel transfers the contents of the shift register into the receiver data holding register and then sets the data available (R DONE) flag.

The data available signal also functions as the clock input to the FRAME ERR, PARITY, and OVERRUN flip-flops in the UART status register. At this point, the DA flip-flop is set, the OVERRUN flip-flop is clear but has a high on the data input because of the output from the DA flip-flop, and the PARITY and FRAME ERR flip-flops are set or cleared depending on the signal (TRUE or FALSE) strobed in from the control logic.

An OVERRUN condition indicates that another data character is being sent to the UART before the previous character has been transferred out. If the DA flip-flop is set, indicating a character is stored in the holding register, and the UART control logic attempts to set the DA flip-flop again (indicating a new character has been shifted into the shift register), the DA signal from the control logic provides a clock input to the OVERRUN flip-flop. This flip-flop then sets because the data input is high (DA flip-flop was already set by the previous DA signal).

If the serial input line goes from a mark (high) to a space (low) and remains at the low level, the receiver shifts in one character, which is all spaces, then sets the FR ERR indicator and waits until the input line goes high (marking) before shifting in another character.

A.4.2 Transmitter Operation

A block diagram of the UART transmitter is shown in Figure A-11. When the UART transmitter is in the idle state, the serial output line (pin 25) is a mark (high). When it is desired to transmit data, a parallel character is strobed into the UART transmitter data buffer (lines connected to pins 26-33) by means of the data strobe signal (pin 23). The time between the low-to-high transition of data strobe and the corresponding mark-to-space transition of the serial output line is within one clock cycle (1/16 of a bit time) if the transmitter has been idle.

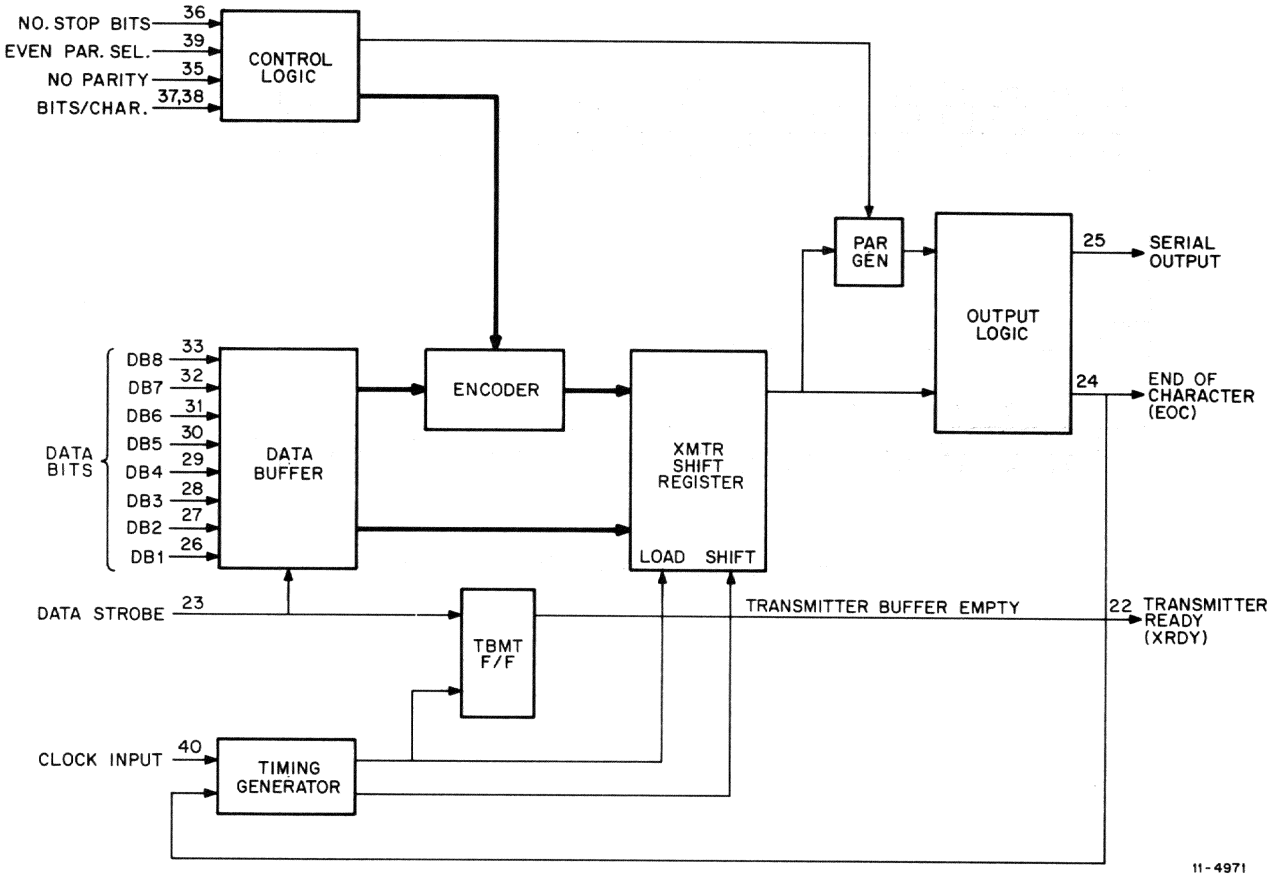


Figure A-11 UART Transmitter - Block Diagram

When the data has been loaded into the UART data buffer, it is next transferred to the transmitter shift register under control of signals from an encoder that selects the format determined by the control logic. This permits selection of parity or no parity (pin 35), the type of parity (pin 39), the number of STOP bits (pin 36), and the number of data bits per character (pins 37 and 38).

The end-of-character (pin 24) signal goes high each time a full character (including STOP bits) is transmitted. If this line goes low, it prevents the timing generator from loading another character into the shift register. The line is normally high when data is not being transmitted and goes low at the start of transmission of the next character.

If the transmitter data buffer is loaded while the previous character is being shifted through to the output line, the START bit of the new character immediately follows the last STOP bit of the previous character.

Figure A-12 shows the pin locations and Table A-5 defines the pin functions.

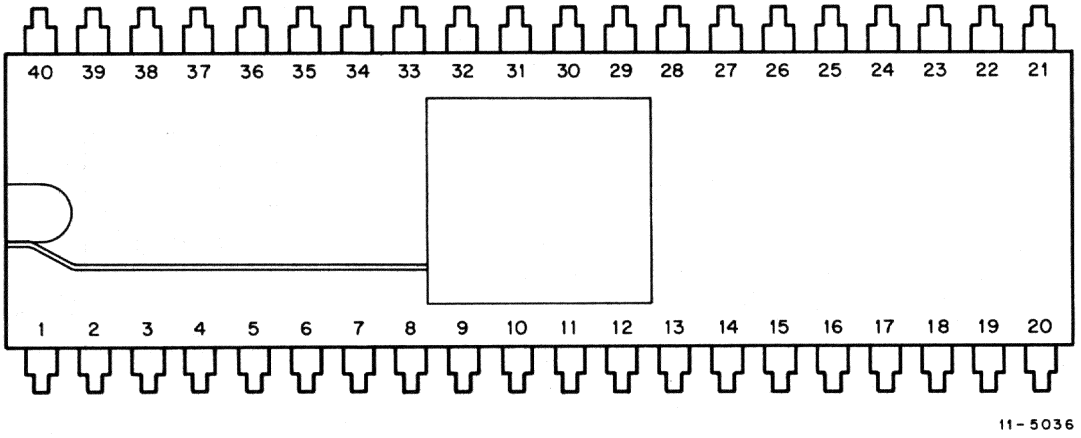


Figure A-12 UART Pin Locations

Table A-5 UART Pin Functions

Pin No.	I/O	Name	Mnemonic	Function
1	I	V_{cc} POWER SUPPLY	V_{cc}	+5 V supply.
2	I	V_{gg} POWER SUPPLY	V_{gg}	-12 V supply.
3	I	GROUND	G	Ground
4	I	RECEIVED DATA ENABLE	RDE	A low on the receiver enable line places the received data onto the output lines.
5–12	O	RECEIVED DATA BITS	RD8–RD1	These are the eight data output lines. These lines may be wire-ORed. When 5, 6, or 7 level code is selected, the most significant unused bits are low. Characters will be right justified into the least significant bits. RD1 (pin 12) is the least significant bit, RD8 (pin 5) is the most significant bit. A high indicates a mark.
13	O	RECEIVE PARITY ERROR	PER	This line goes to a high if the received character parity does not agree with the selected POE.
14	O	FRAMING ERROR	FER	This line goes to a high if the received character has no valid stop bit, i.e., the bit following the parity bit is not marking.
15	O	OVERRUN	OR	This line goes to a high if the previously received character is not read (DA line not reset) before the present character is transferred to the receiver holding register.
16	I	STATUS WORD ENABLE	SWE	A low on this line places the status word bit (PE, DA, TBMT, FE, OR) onto the output lines.
17	I	RECEIVER CLOCK LINE	RCP	This line is for a clock whose frequency is 16 times (16X) the desired receiver baud rate.
18	I	RESET DATA AVAILABLE	RDA	A low on this line will reset the DA line.
19	O	RECEIVED DATA AVAILABLE	DA	This line goes to high when an entire character has been received and transferred to the receiver holding register.

Table A-5 UART Pin Functions (Cont)

Pin No.	I/O	Name	Mnemonic	Function
35	I	NO PARITY	NP	A high on this lead will eliminate the parity bit from the transmitted and received character. The stop bits will immeidately follow the last data bit on transmission. The receiver will not check parity or reception. It will, when asserted, also clamp the PE to a low.
36	I	TWO STOP BITS	2SB	This lead will select the number of stop bits, one or two, to be appended immediately after the parity bit. A low will insert one stop bit and a high will insert two stop bits.
37–38	I	NUMBER OF BITS/CHARACTER	NB2, NB1	These two leads will be internally coded to select either 5, 6, 7, or 8 data bits/character. **NB2 (37) NB1 (38) Bits/Character** 0 (L) 0 (L) 5 0 (L) 1 (H) 6 1 (H) 0 (L) 7 1 (H) 1 (H) 8
39	I	EVEN PARITY SELECT	PEV	The logic level on this pin selects the type of parity that will be appended immediately after the data bits. It also determines the parity that will be checked by the receiver. A low will insert and check odd parity and a high will insert and check even parity.
40	I	TRANSMITTER	TCP	This line is for a clock whose frequency is 16 times (16X) the desired transmitter baud rate.

Table A-5 UART Pin Functions (Cont)

Pin No.	I/O	Name	Mnemonic	Function
20	I	SERIAL INPUT	SI	This line accepts the serial bit input stream. A high must be present when data is not being received. High is a mark. Low is a space.
21	I	EXTERNAL RESET	XR	A high level pulse on this pin will reset TSO, TRMT, and EOC to a high level and RDA, PER, FER, and ROR to a low level.
22	O	TRANSMITTER BUFFER EMPTY	TBMT	The transmitter buffer empty flag goes to a high when the data bits holding register may be loaded with another character.
23	I	DATA STROBE	DS	A low to high transition on this line will enter the data bits into the data bits holding register. Data loading is controlled by the rising edge of DS.
24	O	END OF CHARACTER	EOC	This line goes to a high each time a full character including stop bits is transmitted. It remains at this level until the start of transmission of the next character. Start of transmission is defined as the mark to space transmission of the start bit. It remains at a high when data is not being transmitted.
25	O	SERIAL OUTPUT	SO	This line serially, by bit, provides the entire transmitted character. It remains at a high when no data is being transmitted. High is a mark; low is a space.
26–33	I	DATA BIT INPUTS	DB1-DB8	These are the eight parallel data input lines. If 5, 6, or 7 bits are transmitted, the least most significant bits are used. DB1 is the least most significant bit (pin 26). DB8 is the most significant bit (pin 33). A high input will cause a mark (high) to be transmitted.
34	I	CONTROL STROBE	CS	A high on this lead will enter the control bits (POE, NB1, NB2, SB, NP) into the control bits holding register. This line can be strobed or hard wired to a high level.

A.5 5016 DUAL BAUD RATE GENERATOR

The 5016 is an LSI MOS device containing two independent sections. Each section divides its input clock frequency by one of 16 divisors to produce one of 16 different clock outputs. The divisors are stored in ROMs on the chip. The ROMs are addressed by circuits that latch in and decode the logical states of the address lines (Figure A-13). The address lines may be strobed or held at a dc level. Table A-6 lists the frequencies selected by the address lines. Figure A-14 depicts the 5016 pin locations. Table A-7 defines their functions.

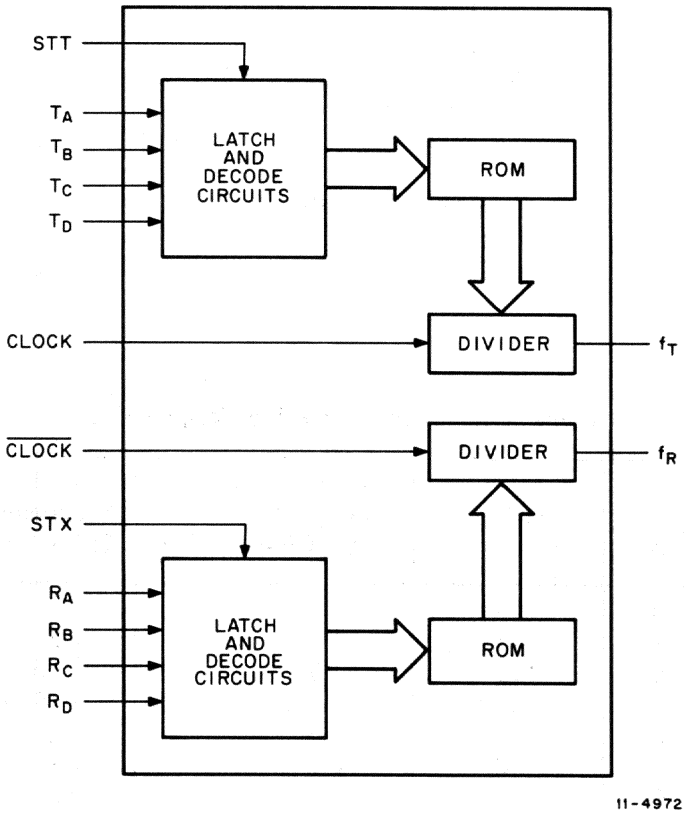


Figure A-13 5016 Block Diagram

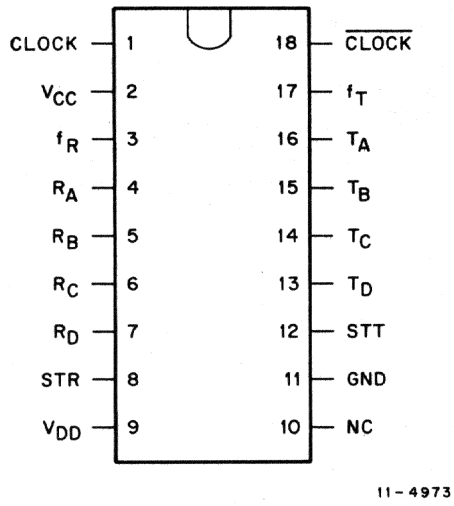


Figure A-14 5016 Pin Locations

Table A-6 5016 Selectable Frequencies

Transmit/Receive Address				Baud Rate	Theoretical Frequency 16X Clock (kHz)	Actual Frequency 16X Clock (kHz)	Divisor
D	C	B	A				
0	0	0	0	50	0.8	0.8	6336
0	0	0	1	75	1.2	1.2	4224
0	0	1	0	110	1.76	1.76	2880
0	0	1	1	134.5	2.152	2.1523	2355
0	1	0	0	150	2.4	2.4	2112
0	1	0	1	300	4.8	4.8	1056
0	1	1	0	600	9.6	9.6	528
0	1	1	1	1200	19.2	19.2	264
1	0	0	0	1800	28.8	28.8	176
1	0	0	1	2000	32.0	32.081	158
1	0	1	0	2400	38.4	38.4	132
1	0	1	1	3600	57.4	57.6	88
1	1	0	0	4800	76.8	76.8	66
1	1	0	1	7200	115.2	115.2	44
1	1	1	0	9600	153.6	153.6	33
1	1	1	1	19200	307.2	316.8	16

Crystal Frequency = 5.0688 MHz

Table A-7 5016 Pin Functions

Pin No.	Mnemonic	Name	Function
1	CLOCK	External Clock Input	This input is either one pin of a crystal oscillator package or one polarity of another external input.
2	V_{CC}	Power Supply	+5 V supply.
3	f_R	Reciever Output Frequency	This output runs at the frequency selected by the receiver address.
4–7	R_A, R_B, R_C, R_D	Receiver Address	The logic levels on these inputs select the receiver output frequency, f_R.
8	STR	Strobe-Receiver Address	A high level input strobe loads the receiver address (R_A, R_B, R_C, R_D) into the latch and decode circuits. This input may be strobed or hard wired to a high level.
9	V_{DD}	Power Supply	+12 V supply.
10	NC	No Connection	
11	GND	Ground	Ground
12	STT	Strobe-Transmitter Address	A high level input strobe loads the transmitter address (T_A, T_B, T_C, T_D) into the latch and decode circuits. This input may be strobed or hard wired to a high level.
13–16	T_D, T_C, T_B, T_A	Transmitter Address	The logic levels on these inputs select the transmitter output frequency, f_T.
17	f_T	Transmitter Output Frequency	This output runs at the frequency selected by the transmitter address.
18	$\overline{\text{CLOCK}}$	Inverted External Clock Input	This input is either one pin of a crystal package or one polarity of another external input.

APPENDIX B
WIRE WRAP INSTRUCTIONS

B.1 PURPOSE

This appendix is intended to assist the user who installs or removes wire wrap jumpers. It describes and illustrates the preferred procedures and standards for producing high-grade solderless wrapped jumper wire connections.

B.2 DEFINITIONS

The following terms are used in discussing wire wrapping:

Solderless wrapped connection– This connection consists of a helix of continuous, solid uninsulated wire tightly wrapped around a wire wrap pin to produce a mechanically and electrically stable connection. In addition to the length of uninsulated wire wrapped around the wire wrap pin, a half turn of insulated wire is wrapped around the pin to ensure better vibration characteristics (Figure B-1).

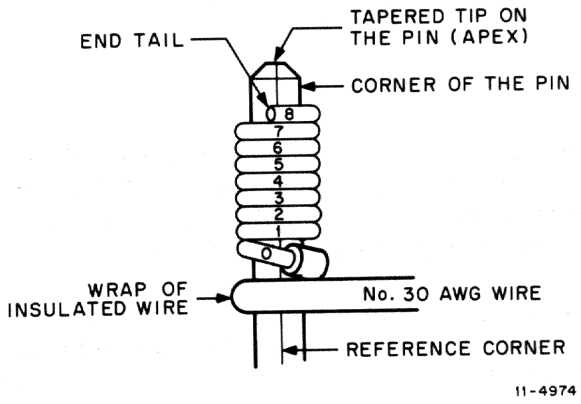
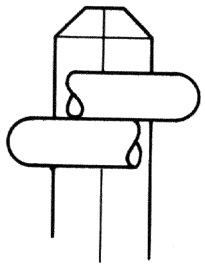


Figure B-1 Solderless Wrapped Connection on Wire Wrap Pin

A turn of wire – A turn of wire consists of one complete, single, helical ring of wire wrapped 360 degrees around a wire wrap pin, intersecting four corners of the pin. Thus, a connection having "n" turns in contact with the wire wrap pin will intersect the reference corner "n + 1" times (Figure B-2).

A half turn of wire – A half turn of wire contacts three of the four corners of a wire wrap pin (Figure B-3).

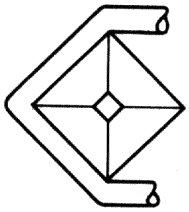
End tail – An end tail is the end of the last turn of wire on the wire wrap pin.



WIRE CONTACTS ALL FOUR CORNERS, AND CONTACTS THE REFERENCE CORNER TWICE.

11-4975

Figure B-2 Full Turn



WIRE CONTACTS THREE CORNERS OF PIN

11-4976

Figure B-3 Half Turn

B.3 CONNECTIONS

Turns are counted along the edge of a reference corner (Figure B-1). There should be seven to nine turns of insulated wire on the wire wrap pin. Each turn should be adjacent to the next turn; one turn should not be wrapped over another turn. The end tail may extend tangentially away from the wire wrap pin, but should not extend more than one wire diameter.

If a second level of wire wrap is placed on a wire wrap pin, the bare wire of the second level wrap should not overlap the first level wrap. The first turn of the insulated wire of the second level wrap may, however, overlap the last turn of the first level wrap (Figure B-4).

The wire used for the jumpers should be good quality wrapping wire. DIGITAL uses the following specifications for the jumpers installed at the factory:

Conductor

Gauge

30 AWG solid

Material

Silver-coated copper

Diameter

0.0257 + 0.0008 cm or -0.0003 cm (0.0101 + 0.0003 in or - 0.0001 in)

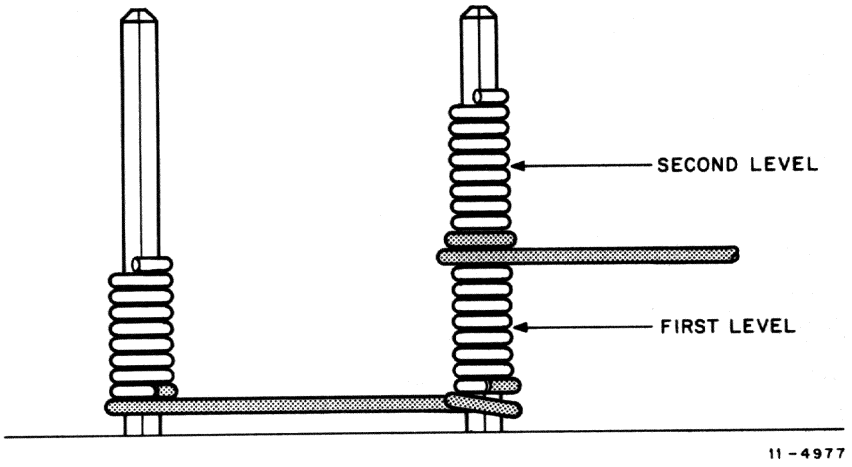


Figure B-4 Two Levels of Wire Wrap

Insulation
Material
Vinylindene flouride

Outside Diameter

0.048 ±0.003 cm (0.018 ±0.001 in)

U.L. Style No.
1423

DC Resistance/304.8 m (1000 ft)
113.6 ohms

NOTE
This wire should not be used for solder applications.

Figures B-1 and B-4 show recommended solderless wrapped connections. Figure B-5 illustrates connections that should be avoided.

B.4 PROCEDURE

To install a wire wrap jumper, proceed as follows:

1. Cut a piece of 30 AWG wire 5.7 cm (2-1/4 in) longer than the distance between the two wire wrap pins.

2. Strip 2.7 cm (1-1/16 in) off each end of the wire.

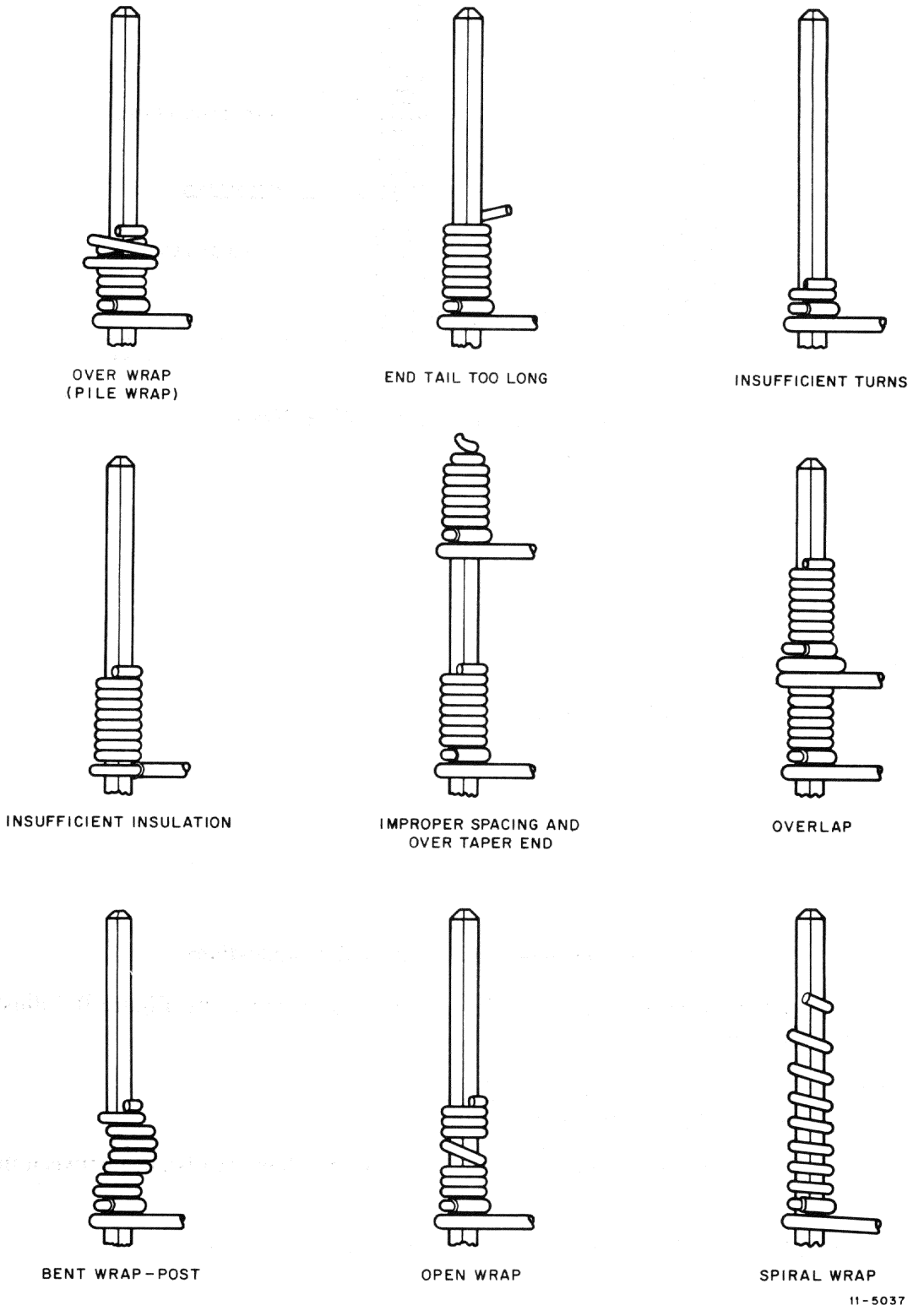


Figure B-5 Defective Wire Wraps

3. Insert the wire into the wire wrap bit far enough for the insulation to enter the feed slot (Figure B-6).

4. Loop the wire through the anchoring notch.

5. Place the tool on the wire wrap pin and actuate the rotating spindle (bit). This should produce eight turns of bare wire and one-half to two turns of insulated wire on the wire wrap pin.

6. Load the free end of the wire into the wire wrap bit and wrap the other wire wrap pin.

Use an unwrapping tool to remove a wire wrap jumper. A jumper may be snipped out to break the electrical connection, but when it is desired to reuse the wire wrap pin the remaining wire should be removed carefully. Pulling the wire off may bend the pin and dent the pin corners. Therefore, it is recommended than an unwrapping tool be used to remove jumper wire wraps.

Place the tool over the wire wrap pin and insert the end tail of the wrap into the unwrapping tool bit. Carefully unwrap the wire and discard it. Jumper wires should not be reused.

If it is desired to place a second level wrap on a wire wrap pin, care should be taken not to overlap the first wrap. If there is insufficient space left on the wire wrap pin for a second level wrap, remove the first level jumper and install a new one lower on the pin. A wire wrap joint that is installed too high on the pin should not be forced to a lower level; it should be unwrapped and replaced with a new one at the lower level.

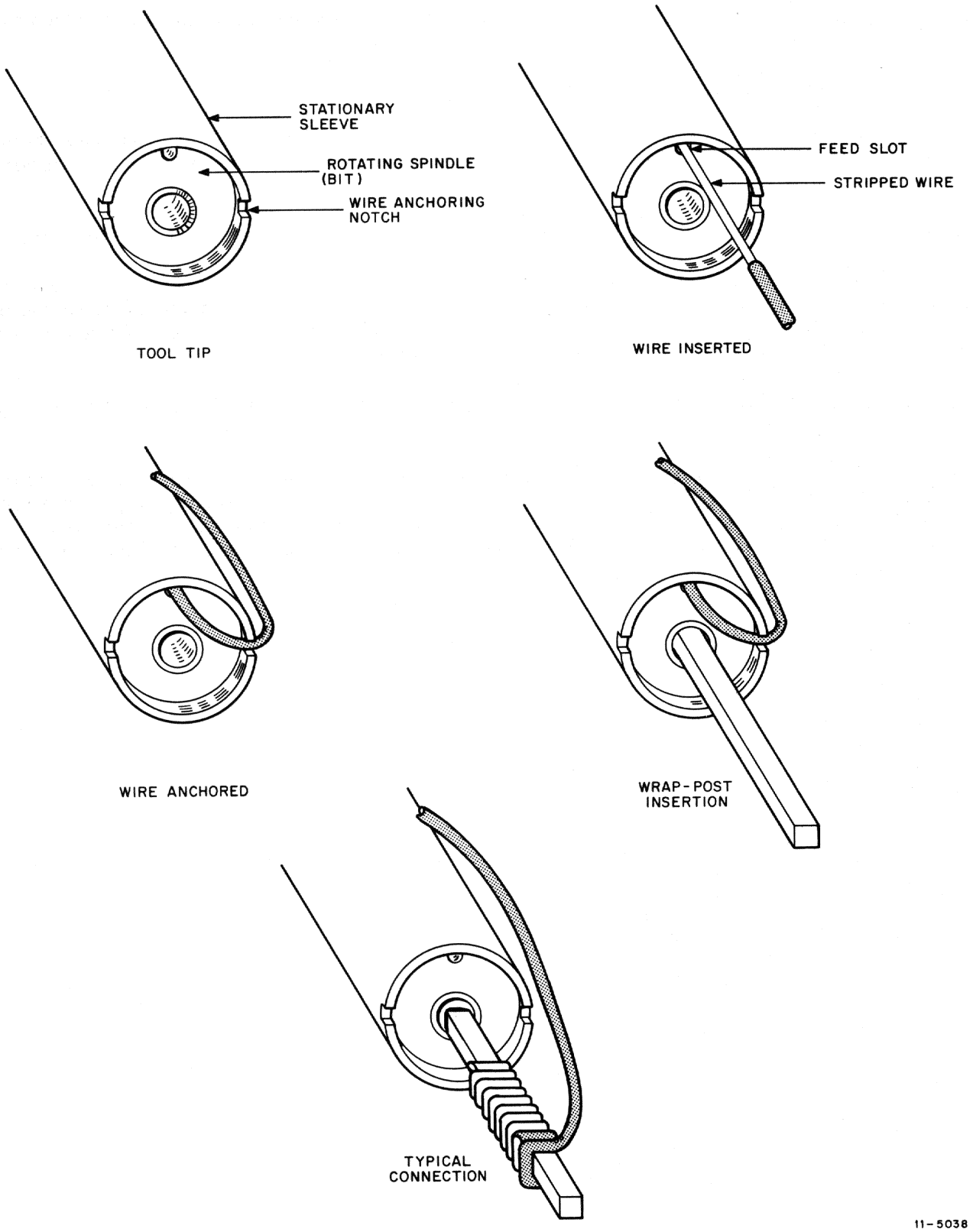


Figure B-6 Loading the Wire Wrapping Kit

DLV11-E AND DLV11-F
ASYNCHRONOUS LINE INTERFACE
USER'S MANUAL
EK-DLV11-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 _____

Fold Here

Do Not Tear - Fold Here and Staple

FIRST CLASS
PERMIT NO. 33
MAYNARD, MASS.

BUSINESS REPLY MAIL
NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

Postage will be paid by:

Digital Equipment Corporation
Technical Documentation Department
Maynard, Massachusetts 01754

digital equipment corporation

Printed in U.S.A.