DATA COMMUNICATIONS MULTIPLEXOR MANUAL

DATA COMMUNICATIONS

MULTIPLEXOR MANUAL

Specifications Subject to Change Without Notice

Note:

This manual covers the following models:

70-5201	16-line multiplexor
70-5202	32-line multiplexor
70-5203	64-line multiplexor
70-5301	Asynchronous RS232 modem line adapter
70-5302	Direct-connection RS232 line adapter
70-5303	Direct-connection current-loop line adapter
70-5304	Direct-connection relay line adapter
70-5305	Synchronous RS232 modem line adapter
70-5306	Binary synchronous communication line adapter

76-5910 Communications Chassis

varian data machines/a varian subsidiary

printed in USA

98 A 9902 251

APRIL 1975

TABLE OF CONTENTS

SECTION 1 INTRODUCTION

SECTION 2 MULTIPLEXOR UNIT

SECTION 3 DIRECT-CONNECTION LINE-ADAPTER

SECTION 4 ASYNCHRONOUS RS232 MODEM LINE-ADAPTER

SECTION 5 SYNCHRONOUS RS232 MODEM LINE-ADAPTER

SECTION 6 BINARY SYNCHRONOUS COMMUNICATION LINE-ADAPTER

LIST OF ILLUSTRATIONS

LIST OF TABLES

SECTION 1
INTRODUCTION

The data communications multiplexor (DCM, figure 1-1) provides a low cost multiport interface between Varian Data Machines computers (with 620-type I/O) and up to 64 full-duplex serial devices. These devices can be serial peripherals and/or serial modems. The DCM consists of a multiplexor unit (MU), multiplexor buses, and several types of line adapter (LAD). The multiplexor bus scheme allows a wide variety of LADs to be added as new needs arise.

The DCM provides communication between the LADs and the computer via programmed I/O and direct-memory access (DMA). The multiplexor logic handles interrupts and trap requests (DMA) internally, and therefore does not require an external priority interrupt module (PIM) or a buffer interlace controller (BIC). For DMA operation, the DCM uses a control table stored in the computer memory. The table contains control characters, output and input block-lengths, output and input buffer-locations, and control information for each line. DCM accesses this table through the computer DMA with individual line addresses as pointers.

The individual lines are set up under program control to begin a transfer-in, transfer-out, or both (full duplex). Once the line is set up the multiplexor inputs or outputs data to/from the line on a demand basis using the DMA port of the computer. The program is interrupted only for a line error, control-character detection, or at completion of the transfer. The data are automatically packed or unpacked by the multiplexor hardware.

The character assembly, disassembly, parity generation, parity checking, modem control and buffering tasks are handled at the LAD level.

Communication between the line adapters and the multiplexor is via the multiplexor bus, which consists of an eight-bit bidirectional bus, 12 control lines, and 12 addressing lines. The multiplexor also provides six bit-rate lines for the asynchronous LADs. These lines go to all LAD locations, and are connected as needed. The frequency of these lines is determined by hardware jumpering options on the multiplexor.

The DCM is usually installed in a 620 or V70 series I/O expansion chassis. It consists of the two multiplexor printed-circuit boards, the LAD complement, and one or two half-backplane assemblies. Systems that handle over 32 lines require two half-backplanes and consequently one entire I/O expansion chassis.

In this manual, section 2 describes the MU and associated buses. Each subsequent section describes one type of LAD.

Each section concludes with a subsection containing applicable programming information. General DCM programming is discussed at the end of section 2. In each LAD section, only programming peculiar to that line adaptor is discussed.

Documents such as logic diagrams, schematics, and parts lists are supplied in a System Maintenance Manual. This manual is assembled when the equipment is shipped, and reflects the configuration of a specific system.

1.1 PHYSICAL DESCRIPTION

The DCM consists of two multiplexor printed circuit boards (MU), one or two special I/O backplanes, and up to 16 LAD boards (each LAD handles up to four lines). Systems with 32 or fewer lines require only one backplane. See (figures 1-2 through 1-5). The DCM backplane(s) is (are) installed in a standard 620 or V70 series I/O expansion chassis. A DCM system larger than 32 lines occupies the entire I/O chassis. If a single-backplane DCM occupies a chassis by itself, a righthand backplane should be used; but if the DCM is to share the chassis with an I/O backplane, a lefthand DCM backplane and a righthand I/O backplane must be used.

LADs are plugged into the backplane starting in slot 9 next to multiplexor board 1. The slot location of the LAD determines its addresses, i.e., the LAD in slot 9 handles lines 0 through 3 and the one in slot 10 handles lines 4 through 7 and so on. If the DCM systems is larger than 32 lines, slot 13 of the second backplane handles lines 32 through 35, etc. Interface to the modems or direct-connection terminals is through two connectors on the back of the LADs (see sections on individual LADs).

The DCM operates at 0 to 50 degrees C and 0 to 90 percent relative humidity (without condensation).

The MU requires a +5V dc ± 5 percent input at 3.5A while a typical LAD requires the following power inputs:

+ 5V dc ± 5 percent at 1.2 A

+ 12V dc ±2.5 percent at 100 mA

- 12V dc ± 2.5 percent at 150 mA

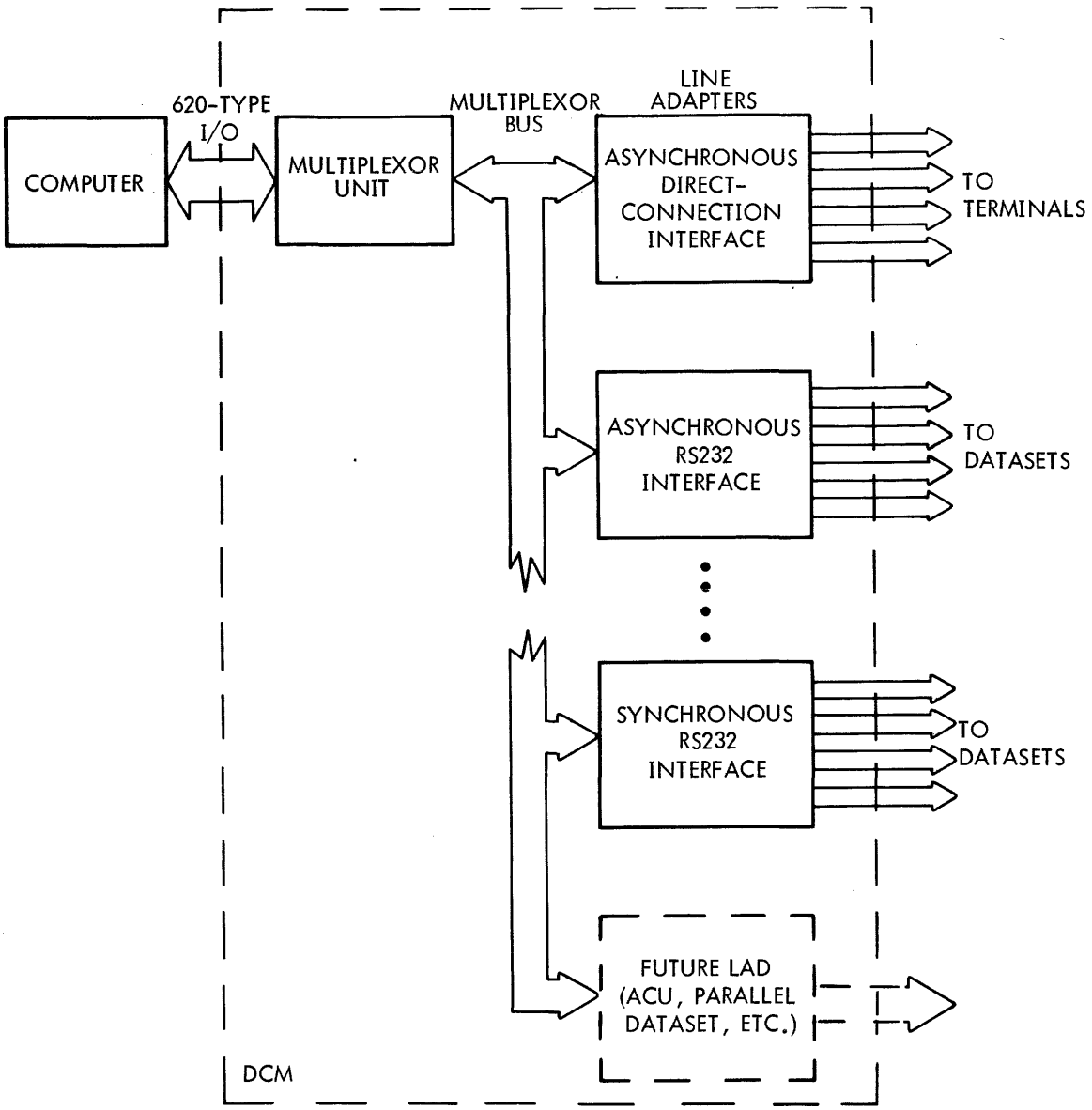
1.2 OPTIONS

The scan-length, scan-counter, and bit-rate lines can be hardwired to provide optimum performance for a particular system. There is also a priority scan in which the DCM resets to line 0 after the second service request. Details on these options are given in the applicable sections of the manual.

1.3 RELATED PUBLICATIONS

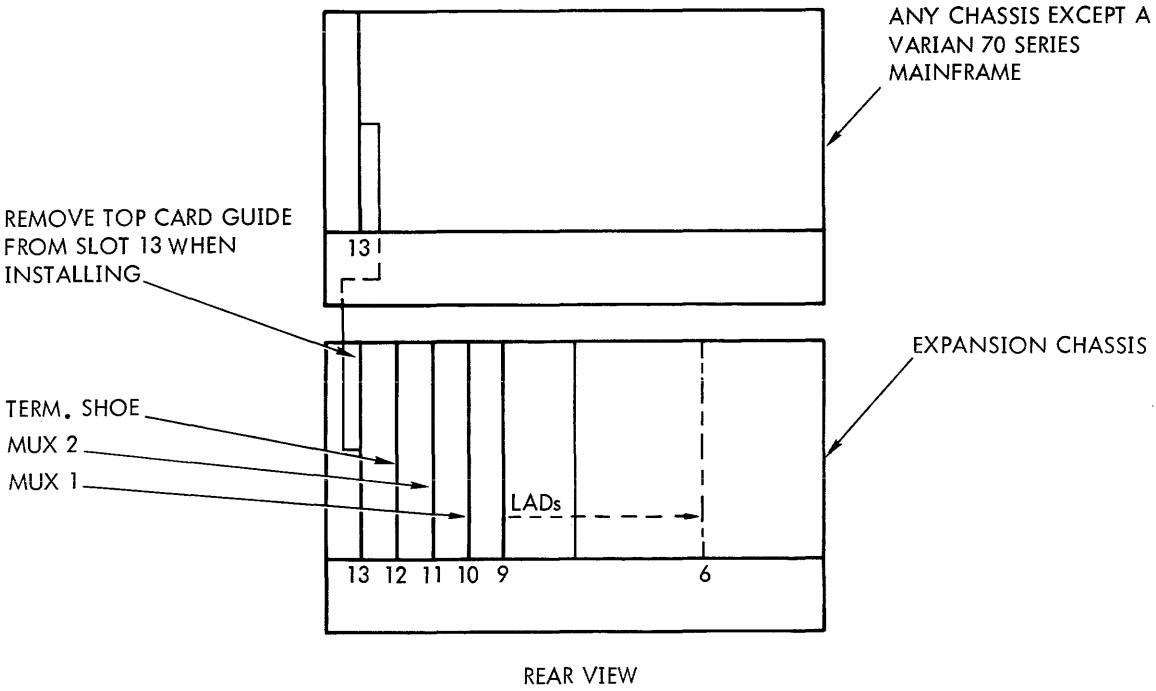
The following publications are aids to understanding the DCM and the systems to which it is applicable.

Varian 73 System Handbook	98 A 9906 01x
Varian 620 100s Computer Handbook	98 A 9905 00x



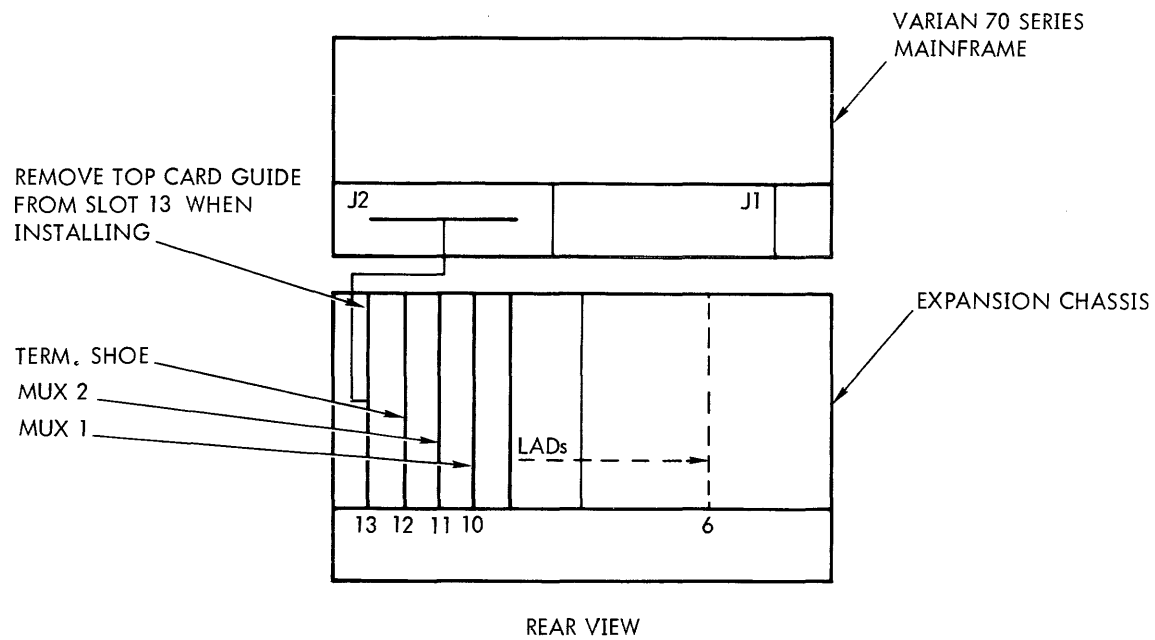
VTII-1678

Figure 1-1. Data Communications Multiplexor (DCM) Block Diagram



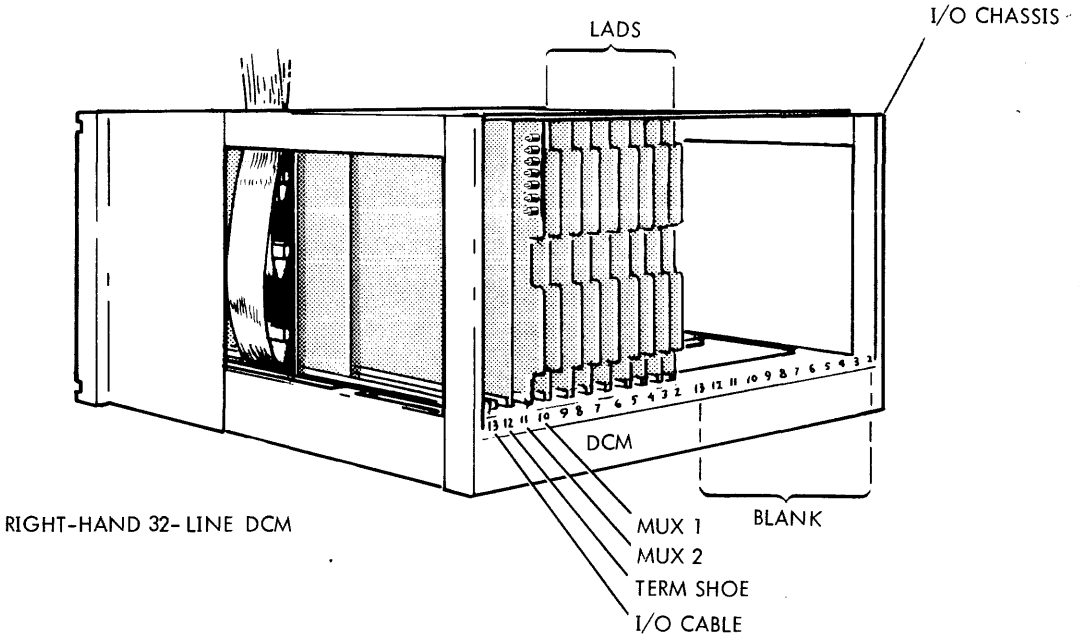
VTII-1751A

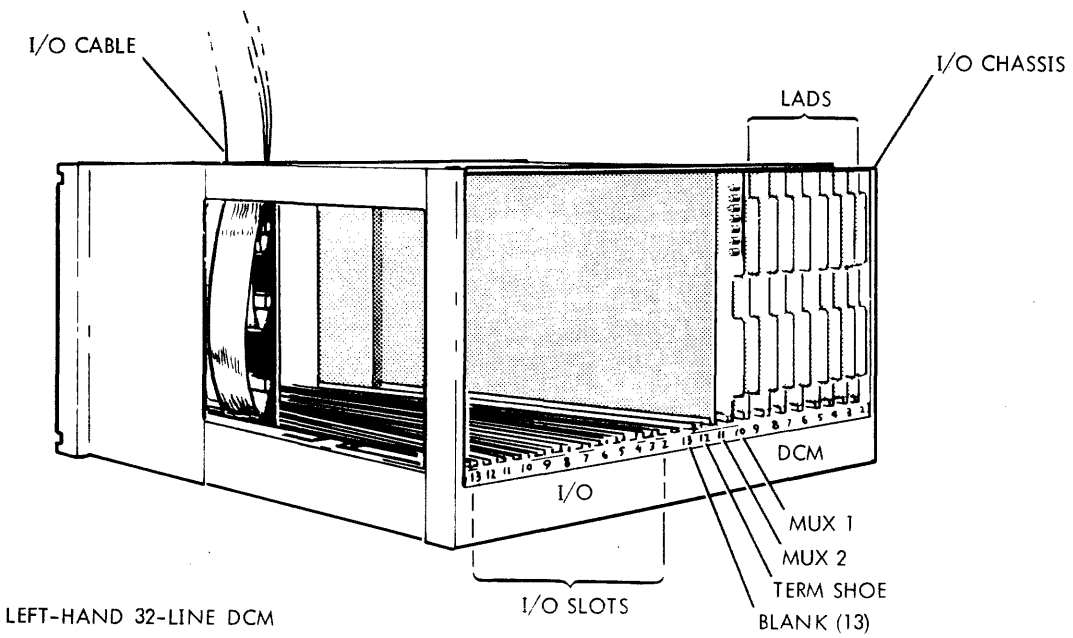
Figure 1-2. Typical installation



VTII-1752A

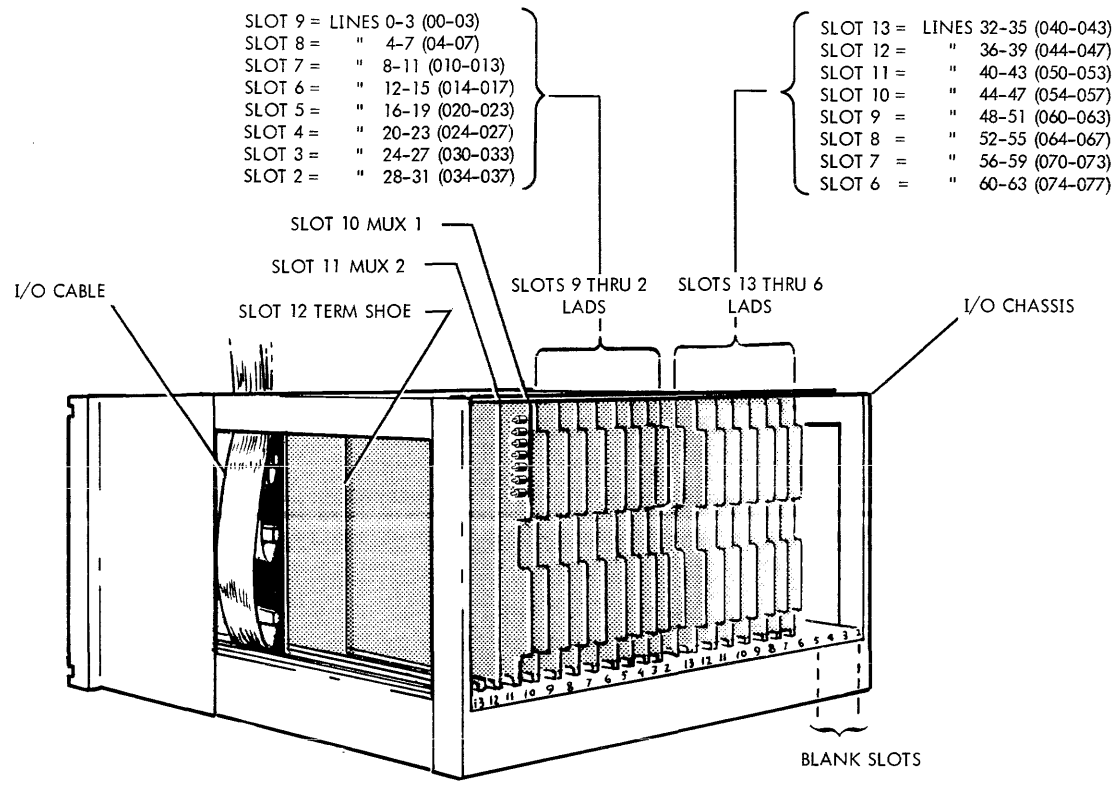
Figure 1-3. Installation with V70 Series Mainframe





VTII-1901

Figure 1-4. 32-Line DCM Configurations



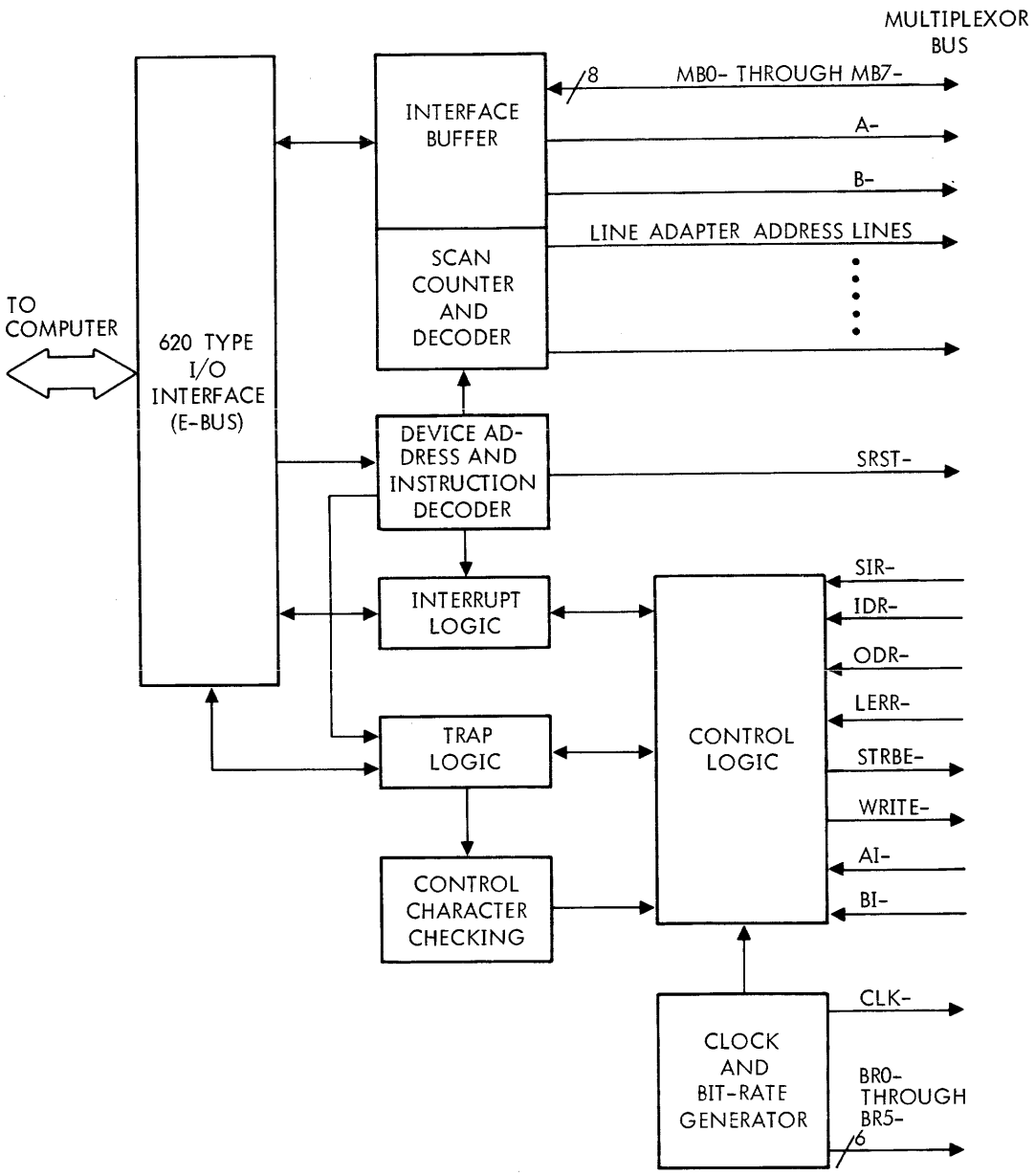
REAR VIEW

VTII-1910

Figure 1-5. 64-Line DCM Configurations

SECTION 2
MULTIPLEXOR UNIT

The multiplexor unit (MU, figure 2-1) is on two printed-circuit boards that fit slots in a special 620-type expansion chassis. It provides communication between the LADs and the computer via the programmed I/O and the DMA. The MU logic handles interrupts and (DMA) trap requests internally and therefore does not require an external priority-interrupt module (PIM) or a buffer-interlace controller (BIC). In DMA operations, the MU uses a control table in the computer memory. The control table contains control characters, output- and input-block lengths, output- and input-buffer addresses and control information for each line. MU access to this table is through the computer DMA, using the individual line addresses as pointers.



VTII-1679A

Figure 2-1. DCM Multiplexor-Unit Simplified Block Diagram

The computer transfers control information to the LADs and monitors line status via programmed I/O. The MU, when not transferring data, scans the individual lines at 614,400 Hz (a new line every 1.63 microseconds) until it detects a service request from a line, or a control request from the computer. Once the request is recognized, the MU stops scanning and generates the proper interrupt to the computer or, in the case of I/O data requests, accesses memory (automatically unpacking and packing bytes) via DMA. Line error information and status is transferred via programmed I/O.

All input data are compared with the control characters (2 per line) stored in the control table. When a match is found for either character, the computer is sent a control-character-detected interrupt.

Table 2-1 gives the specifications of the MU.

2.1 INSTALLATION

The MU has been inspected and packed to ensure its arrival in good working order. To prevent damage, take reasonable care during unpacking and handling. Check the shipping list to ensure that all equipment has been received. Immediately after unpacking, inspect the equipment for shipping damage. If any is found:

a. Notify the transportation company

b. Notify Varian Data Machines

c. Save all packing material

2.1.1 Physical Description

The MU comprises two printed-circuit boards mounted in a specially-wired 620 expansion chassis (figures 1-2 through 1-5). Board 1 is the multiplexor-bus LAD interface and board 2 is the 620-type E-bus I/O interface.

2.1.1.1 Multiplexor Unit Board 1

Board 1 contains the DCM control memory, provides timing for the DCM, and provides the interface to the LADs. It is packaged on a 7-3/4-by-12-3/4-inch multiplexor printed-

Table 2-1. DCM Multiplexor Unit Specifications

Specification	Description
Basic clock frequency	9.8304 MHz
Line-scanning rate	614.4 kHz
Computer interface	
Device code	070 (Standard)
Interrupts	Six interrupts generated by the DCM (no PIM)
DMA (operation)	Accesses memory using table stored in memory for DMA control (no BIC)
Instructions	Six external-control Two transfer
Priority assignment	Interrupt and DMA priority determined by position on the I/O priority chain, communication-line priority by line address
Priority scan	Allows lower-numbered lines to be scanned more often.
Options	Can be strapped to scan 4 to 64 lines (bit rates can be modified by strapping options)
Multiplexor bus	Provides up to six asynchronous data rates (maximum asynchronous or synchronous bit rate/line = 20 kHz)
MU package	Two printed-circuit boards
Power consumption (MU)	+5V dc at 5A
Interconnection	Special I/O backplane that fits a 620 I/O expansion chassis

circuit board that contains 88 integrated circuits. The following paragraphs describe the jumpering options on this board.

Bit-Rate Selection: Board 1 contains two odd-bit-rate counters (variable bit rate counters). Counter number one provides for dividing the input or base frequency by any even number from 2 to 8190, and counter number two divides the base frequency by any even number for 2 to 512 . The output of these counters can then clock the asynchronous LADs. The frequency of the output is varied by adding (or deleting) jumper clips corresponding to the set bits of a binary number determined by the formula:

$$n = \frac{\text{base frequency}}{2 \times \text{desired frequency}} - 1$$

where n is rounded to the nearest whole number and converted to binary; and is less than 4096 for counter one, and less than 256 for counter two.

Tables 2-2 and 2-3 give the jumper locations corresponding to the various bits and to the base-frequencies of counters 1 and 2, respectively. The outputs of counters 1 and 2 are at board-1 points X33 and X34, respectively. Table 2-4 gives values of n for some common frequencies.

Board 1 provides six clocks for the LADs. The frequencies of these clocks are determined by jumpers (table 2-5). The clocks are jumpered into the LADs at the backplane per system requirements.

Table 2-2 Odd-Bit-Rate Counter 1

Jumper Locations

Value of n (Binary Weight) Bit	Jumper Locations (Board 1)
0 (1)	X15 to W15
1 (2)	X16 to W16
2 (4)	X17 to W17
3 (8)	X18 to W18 *
4 (16)	X19 to W19
5 (32)	X20 to W20 *
6 (64)	X21 to W21 *
7 (128)	X22 to W22 *
8 (256)	X23 to W23
9 (512)	X24 to W24 *
10 (1024)	X25 to W25
11 (2048)	X26 to W26 *

Base-Frequency Selection

Base Frequency	Jumper Locations (Board 1)
9.8304 MHz	V19A to V14 *
4.9152 MHz	V20 to V14
2.4576 MHz	V15 to V14
1.2288 MHz	V16 to V14
0.6144 MHz	V17 to V14
0.3072 MHz	V18 to V14

* Indicates standard configuration that provides an output frequency of 1760 Hz (16 times 110 Hz).

Note: Asynchronous LADs require a clock frequency that is 16 times the line bit-rate.

Table 2-3. Odd-Bit-Rate Counter 2

Jumper Locations

Value of n (Binary Weight) Bit	Jumper Locations (Board 1)
0 (1)	X7 to W7
1 (2)	X8 to W8
2 (4)	X9 to W9
3 (8)	X10 to W10
4 (16)	X11 to W11
5 (32)	X12 to W12
6 (64)	X13 to W13
7 (128)	X14 to W14

Base-Frequency Selection

Base Frequency	Jumper Locations (Board 1)
9.8304 MHz	V19B to V13 *
4.9152 MHz	V20 to V13
2.4576 MHz	V15 to V13
1.2288 MHz	V16 to V13
0.6144 MHz	V17 to V13
0.3072 MHz	V18 to V13

* Indicates standard configuration.

Note: Asynchronous LADs require a clock frequency that is 16 times the line-bit rate.

Table 2-4. Values of n for Common Frequencies

Frequency	n (Decimal)	n (Binary)	Base Frequency
800 (16 x 50) Hz	3071	101111111111	4.9152 MHz
1760 (16 x 110) Hz	2792	101011101000	9.8304 MHZ
2152 (16 x 134.5) Hz	2283	100011101011	
14400 (16 x 900) Hz	340	000101010100	
16000 (16 x 1000) Hz	306	000100110010	
28800 (16 x 1800) Hz	170	000010101010	
32000 (16 x 2000) Hz	153	000010011001	
57600 (16 x 3600) Hz	84	000001010100	9.8304 MHz

Table 2-5. Bit-Rate Selection

Frequency Jumper Points

Bit Rate	Jumper Point
153,600 (16 x 9600) Hz	W27
76,800 (16 x 4800) Hz	W28
38,400 (16 x 2400) Hz	W29
19,200 (16 x 1200) Hz	W30
9,600 (16 x 600) Hz	W31
4,800 (16 x 300) Hz	W32
2,400 (16 x 150) Hz	W33
1,200 (16 x 75) Hz	W34
Odd-bit-rate counter #1	X33
Odd-bit-rate counter #2	X34

Bit-Rate Jumper Points

Mnemonic	Jumper Point	Pin Location at LAD Slot
BR0	X27	101
BR1	X28	102
BR2	X29	103
BR3	X30	104
BR4	X31	105
BR5	X32	106

Standard-Bit-Rate Jumper Configuration

Frequency	Mnemonic	Jumper
153,600 (16 x 9600) Hz	BR0	W27 to X27
38,400 (16 x 2400) Hz	BR1	W29 to X28
19,200 (16 x 1200) Hz	BR2	W30 to X29
4,800 (16 x 300) Hz	BR3	W32 to X30
2,400 (16 x 150) Hz	BR4	W33 to X31
* 1,760 (16 x 110) Hz	BR5	X33 to X32

Note: Asynchronous LADs require a clock frequency 16 times the line bit rate.

* Output of odd-bit-rate counter 1

Table 2-5. Bit-Rate Selection (continued)

Line Adapter Clock Inputs (at Backplane)

Line	Input Pin	Remarks
0	92	Speed A for RS232 asynchronous LAD, input for direct-connection LAD, test clock for RS232 synchronous LAD
	94	Speed B for RS232 asynchronous LAD
1	93	Speed A for RS232 asynchronous LAD, input for direct-connection LAD
	91	Speed B for RS232 asynchronous LAD
2	98	Speed A for RS232 asynchronous LAD, input for direct-connection LAD
	95	Speed B for RS232 asynchronous LAD
3	97	Speed A for RS232 asynchronous LAD, input for direct-connection LAD
	96	Speed B for RS232 asynchronous LAD

The RS232 asynchronous LAD requires two clocks per line, the direct-connection LAD requires one clock per line, and the RS32 synchronous LAD requires only one clock input for testing.

Scan-length selection: The number of lines scanned by the multiplexor is adjustable (table 2-6). The standard configuration depends on the DCM model number (i.e., 16, 32 or 64).

Priority-mode selection: The DCM can be put in the priority mode by jumpering point W6 to W5 on board 1. This simply causes the DCM to begin scanning at line zero after alternate service requests (either from a line or from the computer). Normally the DCM continues to scan lines sequentially. When priority mode is selected, the higher-speed lines should have the numerically lower addresses. This mode should only be used when a few high-speed lines are being run concurrently with a large number of low-speed lines. If the line speeds are all relatively close together, this mode should not be used. In priority mode, the scan length should be as close to the actual number of lines in use as possible.

Table 2-6. Scan-Length Selection

Scan Length	Jumpers (Board 1)
4 lines	Points W4, W3, W2 and W1 open
8 lines	X3 to W3
16 lines	X3 to W3; X1 to W1
32 lines	X3 to W3; X1 to W1; X2 to W2
64 lines	X3 to W3; X1 to W1; X2 to W2; X4 to W4

2.1.1.2 Multiplexor Unit Board 2

Board 2 contains the interface to the computer I/O. It is packaged on a 7-3/4-by-12-inch multiplexor printed circuit board that contains 90 integrated circuits. The following paragraphs describe the jumpering options on the board.

Device-Address Selection: The standard device-address is 070, but can be changed to any address from 00 to 077 by jumpers (table 2-7).

Table 2-7. Device-Address Selection

Selection of Less-Significant Portion

Octal Device Address	Jumper
x0	U0 to U8
x1	U1 to U8
x2	U2 to U8
x3	U3 to U8
x4	U4 to U8
x5	U5 to U8
x6	U6 to U8
x7	U7 to U8

Selection of More-Significant Portion

Octal Device Address	Jumper
0x	T0 to T8
1x	T1 to T8
2x	T2 to T8
3x	T3 to T8
4x	T4 to T8
5x	T5 to T8
6x	T6 to T8
* 7x	T7 to T8

* This jumper is etched on the board and must be cut if another configuration is required.

E-Bus Priority Selection: The DCM generates its own interrupts and DMA-transfers. Consequently, it must be wired into the computer I/O priority chain. The priority input signal (PRMX-I) is on pin 3 of board 2 and the priority output signal (PRNX-I) is on pin 4. Board 2 is normally plugged into slot 11 of the DCM main backplane.

Interrupt Address Selection: The DCM generates six interrupts. Their standard addresses are:

Input byte-count zero	060
Output byte-count zero	062
Line error	064
Status change	066
Control-character detected	070
Control	072

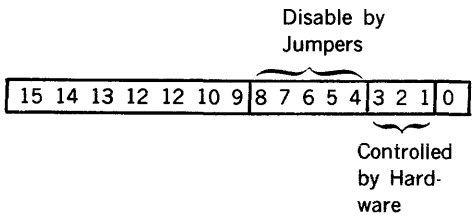
The addresses can be moved in a contiguous block by means of jumpers on board 2 (table 2-8). Bits 4 through 8 are always enabled unless jumpers are added to disable them. This allows location of the interrupt block beginning at any even block-address from 000 to 0760, i.e.: 000, 020, 040, ... 0740, 0760.

Table 2-8. Selection of Interrupt Addresses
Interrupt Address Block Location

Interrupt Address Bit	Disabling Jumper (Board 2)
* Bit 08	W8 to V8
* Bit 07	W7 to V7
* Bit 06	W6 to V6
Bit 05	W5 to V5
Bit 04	W4 to V4

* Standard configuration (interrupt address block 060)

Interrupt Address Format



Control-Table Address Selection (LCB): The DCM control table or line-control block (LCB) requires 8 dedicated memory locations per line (512 for a maximum system). By additions or deletions of jumpers on board 2, the table can be relocated to any 1K boundary of memory. Table 2-9 gives the jumper locations for all possible LCB starting-address locations. Unless otherwise specified, the beginning address is 017000.

High-Speed DMA: The DCM contains logic to interface with the high-speed DMA of the V70 series computers. This option is enabled by moving jumpers on board 2. Table 2-10 lists the jumper configurations for high-speed DMA or normal DMA. The DCM provides three input priority signals: PRMA + I, PRMB + I, and PRMC + I on pins 6, 7, and 5 of the backplane (slot 11); and one output priority signal PRNF + I on pin 2 of the backplane (slot 11) for the high-speed DMA priority chain.

Table 2-9. Control Table (LCB) Beginning-Address Selection

Beginning Address Octal	Jumpers (Board 2) W15 V15	W14 V14	W13 V13	W12 V12	W11 V11	W10 V10	W9 V9	Beginning Address Octal	Jumpers (Board 2) W15 V15	W14 V14	W13 V13	W12 V12	W11 V11	W10 V10	W9 V9
00000	X	X	X	X	X	X	X	40000	X	-	X	X	X	X	X
01000	X	X	X	X	X	X	-	41000	X	-	X	X	X	X	-
02000	X	X	X	X	X	-	X	42000	X	-	X	X	X	-	X
03000	X	X	X	X	X	-	-	43000	X	-	X	X	X	-	-
04000	X	X	X	X	-	X	X	44000	X	-	X	X	-	X	X
05000	X	X	X	X	-	X	-	45000	X	-	X	X	-	X	-
06000	X	X	X	X	-	-	X	46000	X	-	X	X	-	-	X
07000	X	X	X	X	-	-	-	47000	X	-	X	X	-	-	-
10000	X	X	X	-	X	X	X	50000	X	-	X	-	X	X	X
11000	X	X	X	-	X	X	-	51000	X	-	X	-	X	X	-
12000	X	X	X	-	X	-	X	52000	X	-	X	-	X	-	X
13000	X	X	X	-	X	-	-	53000	X	-	X	-	X	-	-
14000	X	X	X	-	-	X	X	54000	X	-	X	-	-	X	X
15000	X	X	X	-	-	X	-	55000	X	-	X	-	-	X	-
16000	X	X	X	-	-	-	X	56000	X	-	X	-	-	-	X
17000*	X	X	X	-	-	-	-	57000	X	-	X	-	-	-	-
20000	X	X	-	X	X	X	X	60000	X	-	-	X	X	X	X
21000	X	X	-	X	X	X	-	61000	X	-	-	X	X	X	-
22000	X	X	-	X	X	-	X	62000	X	-	-	X	X	-	X
23000	X	X	-	X	X	-	-	63000	X	-	-	X	X	-	-
24000	X	X	-	X	-	X	X	64000	X	-	-	X	-	X	X
25000	X	X	-	X	-	X	-	65000	X	-	-	X	-	X	-
26000	X	X	-	X	-	-	X	66000	X	-	-	X	-	-	X
27000	X	X	-	X	-	-	-	67000	X	-	-	X	-	-	-
30000	X	X	-	-	X	X	X	70000	X	-	-	-	X	X	X
31000	X	X	-	-	X	X	-	71000	X	-	-	-	X	X	-
32000	X	X	-	-	X	-	X	72000	X	-	-	-	X	-	X
33000	X	X	-	-	X	-	-	73000	X	-	-	-	X	-	-
34000	X	X	-	-	-	X	X	74000	X	-	-	-	-	X	X
35000	X	X	-	-	-	X	-	75000	X	-	-	-	-	X	-
36000	X	X	-	-	-	-	X	76000	X	-	-	-	-	-	X
37000	X	X	-	-	-	-	-	77000	X	-	-	-	-	-	-

* = Standard configuration.

Note: X indicates that a jumper is required.

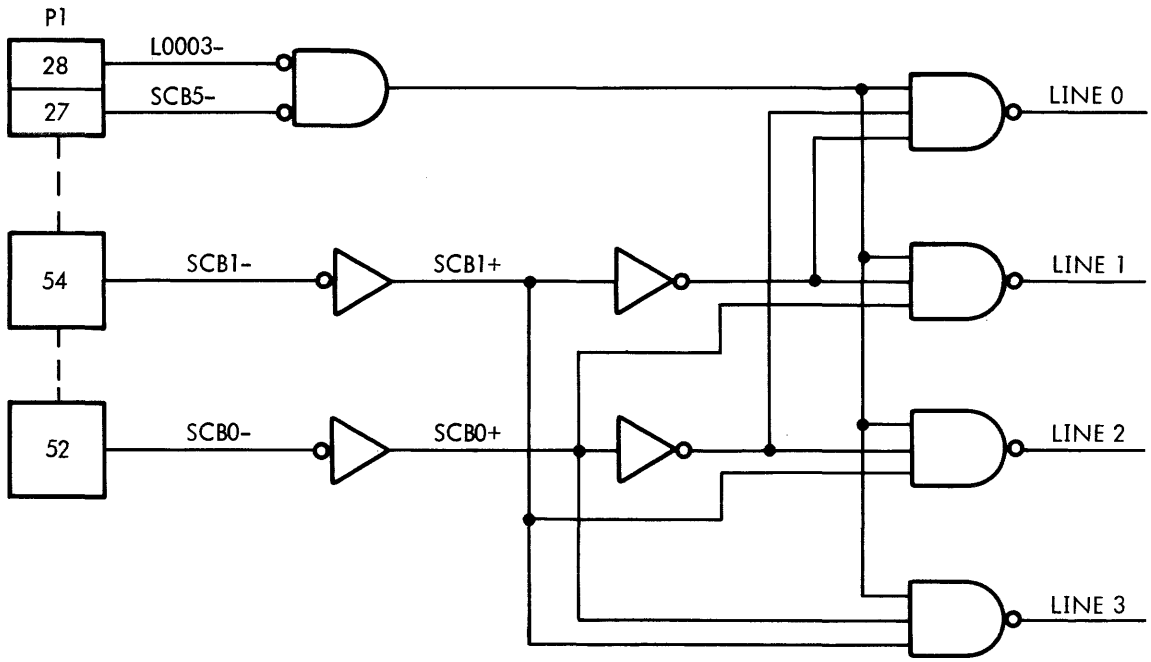
Table 2-10. DMA Mode-Selection Jumpers

High-Speed DMA	Jumpers (Board 2)		Normal DMA	Jumpers (Board 2)	
1	S1 to S2	FTEN–	1	Z1 to Z2	FSTRP–
2	Z3 to Z2	FSTRP–	2	X1 to X2	FSTRP+
3	X3 to X2	FSTRP+	3	Y1 to Y2	FSTDP+
4	Y1 to Y2	FSTDP+			

Note: Remove all existing jumpers when switching from one mode to another.

2.1.2 Interfaces

The MU interfaces with the computer over a standard 620-type I/O interface (E-bus) and with the LADs over the multiplexor bus (figures 1-1 and 2-1). Figure 2-2 illustrates a typical LAD-address decoder.



VTII-1753

Figure 2-2. Typical LAD-Address Decoder

Pin assignments are given on the logic diagrams in the system maintenance manual, and signal levels in table 2-11.

Table 2-11. Signal Levels

E-bus	Negative logic:	False = +2.2 to 3.2V dc
		True = -0.5 to +0.8V dc
Multiplexor bus	Negative logic:	False = +2.2 to +5.0V dc
		True = -0.5 to +0.8V dc
Internal (multiplexor)	Positive logic:	True = +2.4 to +5.0V dc
		False = -0.5 to +0.8V dc

2.1.2.1 E-Bus

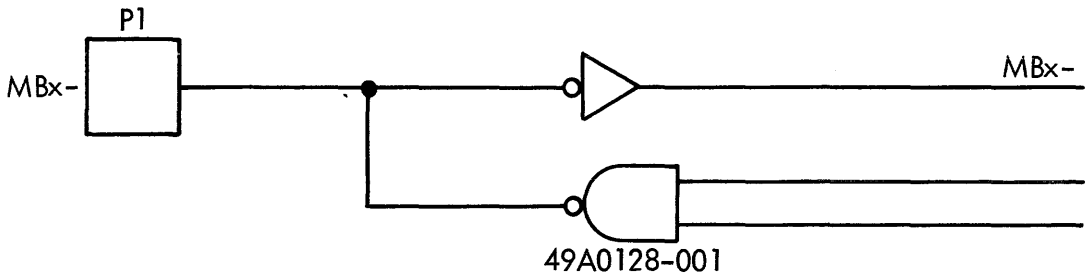
The DCM provides a standard interface to the 620-type I/O bus (see 620/f-100 maintenance manual 98 A 9908 15x for a description of the interface). Since the DCM generates the trap requests and interrupt requests, it must be assigned a position on the I/O priority line. The trap addressing for data is handled by software assignment, but the interrupt addresses, priority assignment, the device code and the line-control-block addresses are determined by strapping.

2.1.2.2 Multiplexor Bus

The MU interfaces with the LADs over a 38-line common bus comprising an eight-bit bidirectional bus (figure 2-3), eight control lines, six bit-rate lines, twelve addressing lines, a clock line, two line-error lines, and a system-reset line (figure 2-1).

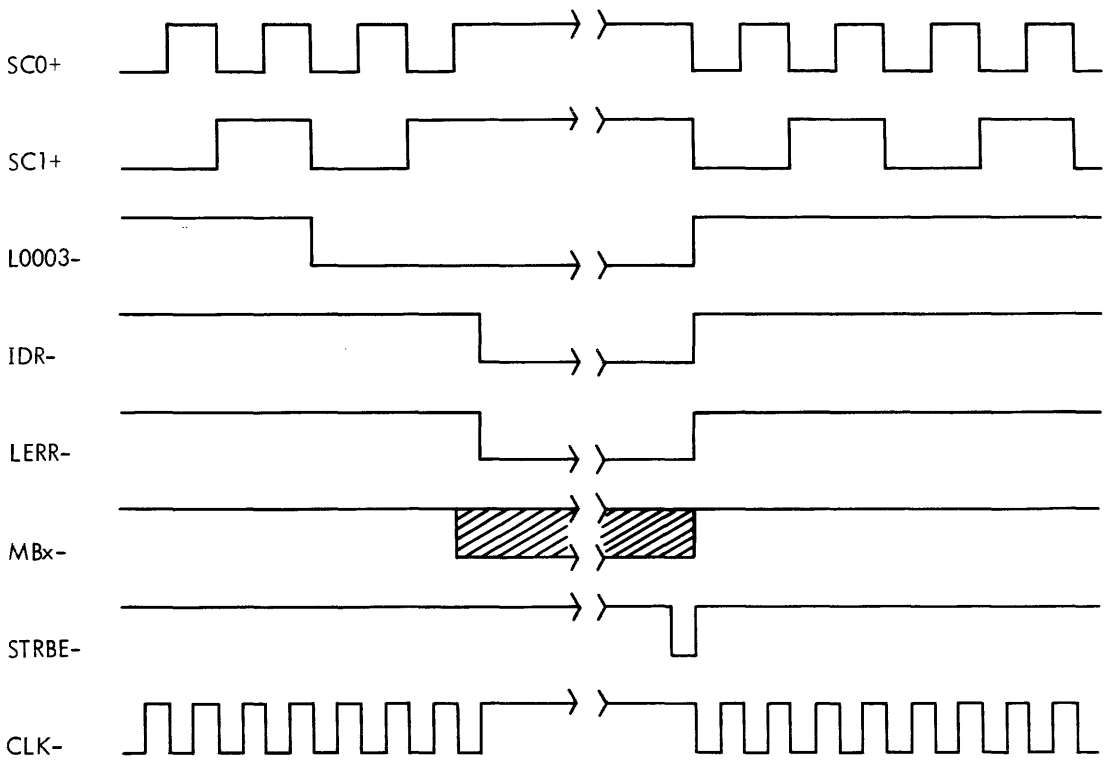
The LAD should present only one high-speed TTL load (2.0 milliamperes) to any of the MU output lines. Except for the bidirectional bus, which requires a 48-milliampere sink, all lines driven by the LAD are driven with open-collector gates capable of sinking 16 milliamperes in the low state.

Figures 2-4, 2-5, and 2-6 show the relative timing for a data-input request, a status-input request, and a data-output request, respectively. Data are stable before the leading edge of the STRBE- pulse.



VTII-1754A

Figure 2-3. LAD Interface to the Eight-Bit Bidirectional Bus



VTII-1755

Figure 2-4. Data-Input-Request Timing

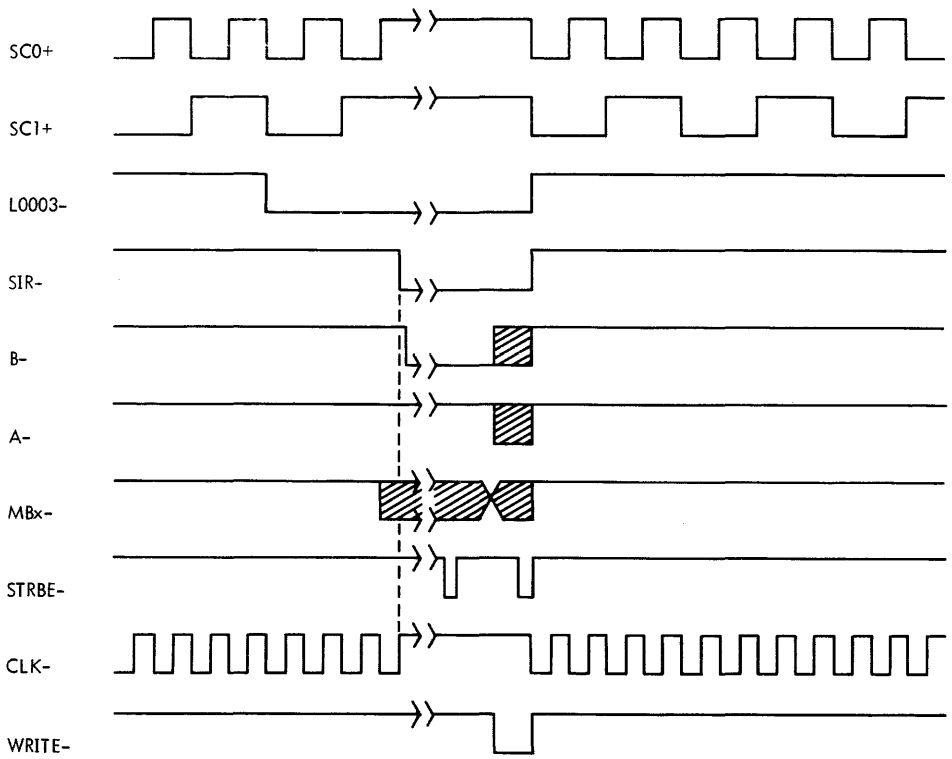


Figure 2-5. Status-Input-Request Timing

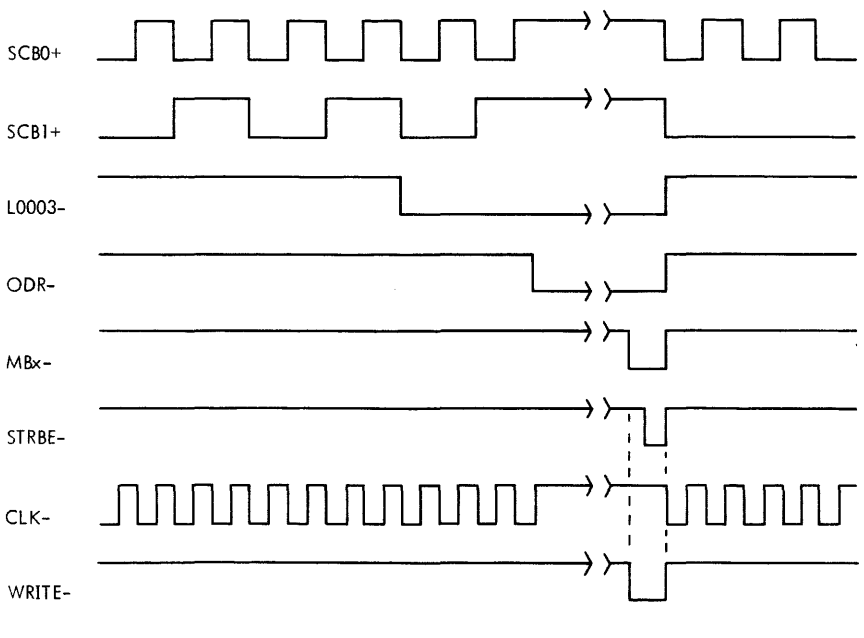


Figure 2-6. Data-Output-Request Timing

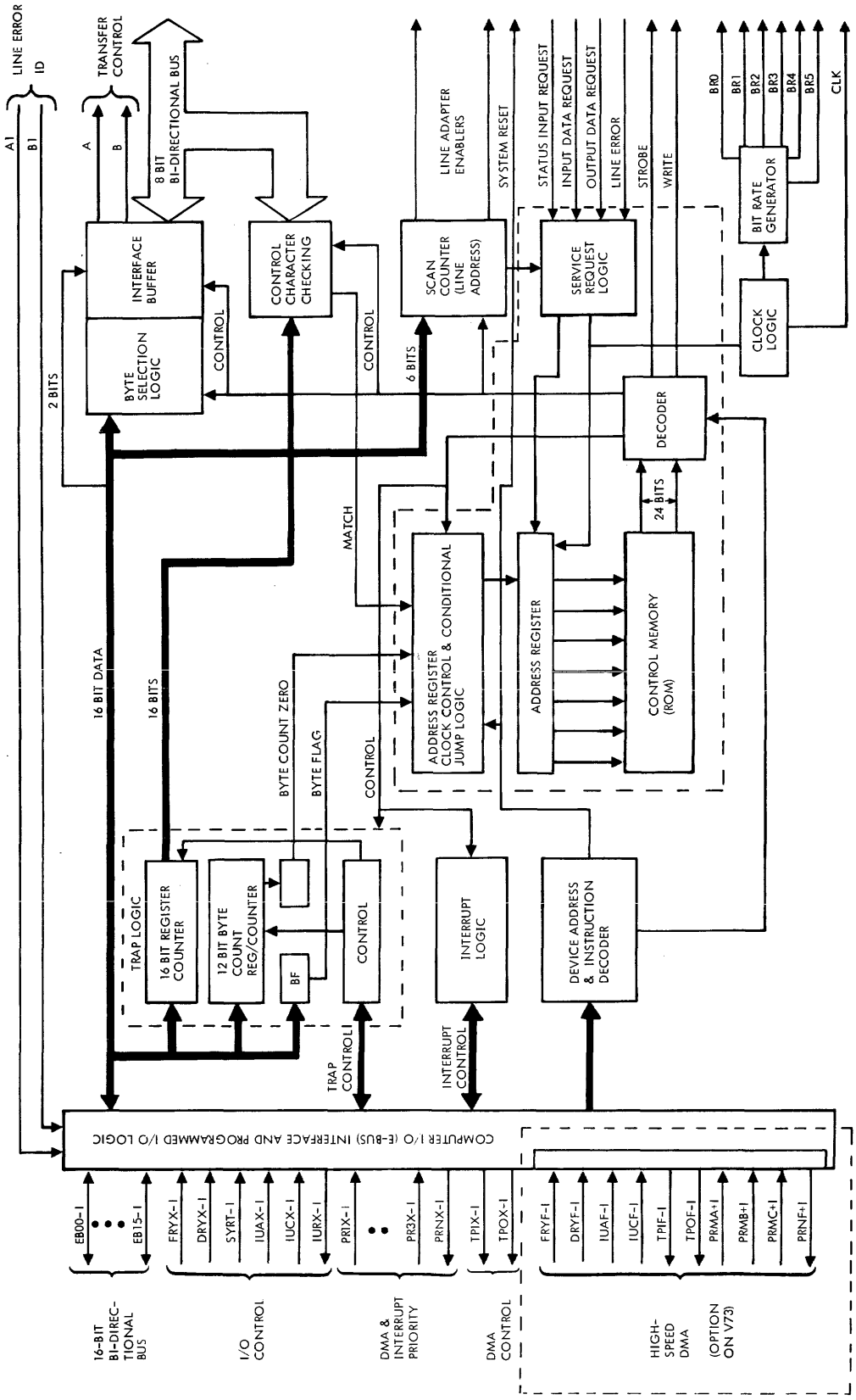


Figure 2-7. MU Detailed Block Diagram

2.2 THEORY OF OPERATION

This section explains the operation of the MU, with each subsection corresponding to a block on the simplified MU block diagram (figure 2-1). Figure 2-7 is a detailed block diagram of the MU. For blocks on MU board 1, refer to logic diagram 91B0412; for board 2, refer to 91B0415.

2.2.1 E-Bus Interface (MU Board 2)

This block provides a standard interface to the 620-type I/O bus. It contains the necessary drivers, receivers, and gating structures needed for the DCM to communicate with the processor. Since the DCM generates its own interrupts and trap requests, it must be assigned a position on the I/O priority chain.

E-bus inputs are buffered and sent to both MU boards. There are several sources for the E-bus outputs, and the enabling signals for these are discussed below.

EBCA + and EBC + transfer the contents of the byte-count register/counter, the BF flip-flop, and SMB flip-flop to the E-bus during the data phase (FSTRP +) of the DMA byte-count storage cycle.

EDR + and EDRA + gate the output of the DMA 16-bit register/counter onto the E-bus. EDR + and EDRA + are true during the address phase of a data fetch/store or input-first-byte-fetch DMA cycle (FSTDP +). The two enabling signals are also true during the data phase of buffer-address storage (FSTRP +).

ECTA + and ECTAA + gate the control-table address (LCB) onto the E-bus lines. Jumper points determine bits 9 thru 15 of the address. The third bit (CTA02 +) is enabled by CMB08 + (input-data sequence) or by FB6 (control-byte fetch). The remaining portion of the LCB address depends on the line address (SCB0 + thru SCB5 +), and CMB05 + and CMB06 + .

ELSMB + and ELMBA + enable the multiplexor bus signals (MB0 + through MB7 +) onto the less-significant byte of the E-bus, and the interface buffer outputs (IB0 + E through IB7 + E) onto the more-significant byte. (During an input sequence, the first byte stored at a buffer address is fetched and stored in the interface buffer before transferring the second byte to memory). ELSMB + and ELMBA + are enabled only during a DMA data-storage cycle (IFB4 +) at FSTRP + time if the byte to be stored is the second one for that buffer address (BF–).

EMSMB + enables the multiplexor bus signals (MB0 + through MB7 +) onto the E-bus during DMA data-storage (IFB4 +) at FSTRP + time if the byte to be stored is the first byte for that buffer address (BF +). For this transfer the less-significant portion of the E-bus is kept at zero. EMSMB + is also enabled during a data-transfer-in (DTI–). In this instance, DTIA + enables the line address (SCC0 + through SCB5 +) onto the less-significant portion of the E-bus. If the transfer is in response to a line error interrupt (FA12–), EABO + is enabled and this signal gates BI + and AI + onto bits six (EB06-I) and seven (EB07-I) of the E-bus.

INTA + and INTB + gate the interrupt address onto the E-bus during an interrupt. The interrupt address is controlled by hardware jumpers for bits 4 through 8 and control-memory bits 01 through 03 (CMB01-CMB03). Bit zero is always zero.

2.2.2 Interface Buffer and Byte-Selection Logic (MU Board 1)

This section contains a ten-bit buffer, gating to load the buffer from the trap logic or the E-bus logic to load the interface buffer from the upper 8 bits of the 16-bit transfer or the lower 8 bits, and drivers to interface to the LADs.

The ten-bit buffer provides temporary storage for data or status information transferred to or from the LADs. Eight of the bits are for status or data. The other two indicate the type of transfer (section 2.4).

The buffer is cleared periodically by SCLKE– until the DCM stops scanning (SCLKE– is held high). The buffer can then be loaded by a DTO instruction or by a DMA operation. DTOA– enables the buffer to be loaded on a data-transfer instruction at FSDRY + time. FSDRY + is an OR-function of DRYX-I and DRYF-I. EIBC– enables loading during a DMA transfer. The buffer is loaded on a data fetch/store (FB4–), a first input-byte fetch (FB5–), or line-control-byte fetch (FB6–). Note that the buffer consists of eight bits of data and two control bits, AC + and BC + . These control bits are normally held to zero and are loaded with the values of EB06 + and EB07 + (bits six and seven of the E-bus) when EAB + is true. EAB + is true during a DTO instruction and during a line-control-byte fetch (FB6–). The eight data inputs to the buffer can be selected from either the more significant eight bits of the E-bus or the less-significant eight bits. The selection is controlled by ELSB– . The more significant eight bits are selected except during a data fetch/store (FB4–). During FB4– , the byte selection is determined by byte flag BFA + .

2.2.3 Scan Counter and Decoder (MU Board 1)

The scan counter is a six-bit synchronous counter clocked at 614,400 Hz. The six outputs of the counter scan up to 64 DCM communication lines. The two least-significant bits of the counter are buffered and sent to each LAD. Bits three, four, and five are decoded to produce eight enabling terms available to each LAD. The most significant bit and its complement enable lines above number 32 and below 32, respectively.

Figure 2-8 shows the scan-counter timing.

The six outputs of the scan counter are decoded to produce the LAD enabling terms L0003– through L2831– . These are sent to the LADs with the two least-significant outputs

(SCB0– and SCB1–) and the most-significant output (SCB5+ and SCB5–). Each LAD receives one of the enabling terms (the actual term depends on what slot it occupies) SCB0–, SCB1–, SCB5+ or SCB5–. SCB5+ enables the LADs below line number 32 and SCB5– enables those above 31. SCB0– and SCB1– are decoded on the LAD for the individual line scanning.

The scan counter is clocked at 614,400 Hz (SCCLK+) as long as the control memory is in idle (CMB00– low). Once CMB00– goes high, SCLKE+ is held low, holding the scan counter at the line number being serviced.

The scan counter is loaded by LSC–. In normal operation, when the control memory is in idle, the scan counter is parallel-loaded with zeros each time it reaches a predetermined count. If the priority mode is selected, the scan counter is loaded with zeros after every other service request (SSCN+). During a control sequence the scan counter is also loaded with a line address over the programmed I/O. In this case, DTOA– enables LSC–, DTOA+ gates the six least-significant bits of the E-bus to the counter, and the counter is clocked at FSDRY+ time.

2.2.4 Instruction Decoder (MU Board 2)

This block contains the data-transfer-out (DTO) and data-transfer-in (DTI) flip-flops for data-transfer control, as well as the gating for decoding the DCM external-control instructions.

The less-significant portion of the device address is decoded from EB01+, and EB02+ to obtain DVC0– through DVC7–. One of these signals is then jumpered to DVC05–. The more-significant portion is obtained by decoding EB03+, EB04+, and EB05+. The results of this are DVCE4– (device address 04x) and DVCE7– (device address 07x). Note that the decoded device-address is not enabled during IUAX+ or IUAF+. This prevents the DMA or interrupt address from being interpreted as an instruction. DVCDS– is gated with DVCE7– to form the DCM device address. DVC4– is gated with DVCE4– to form device-address 044 for common interrupt enable/disable instructions.

The programmed I/O data-transfer flip-flops are clocked by DVCFR+ (device address and FRYX+). If EB13+ or

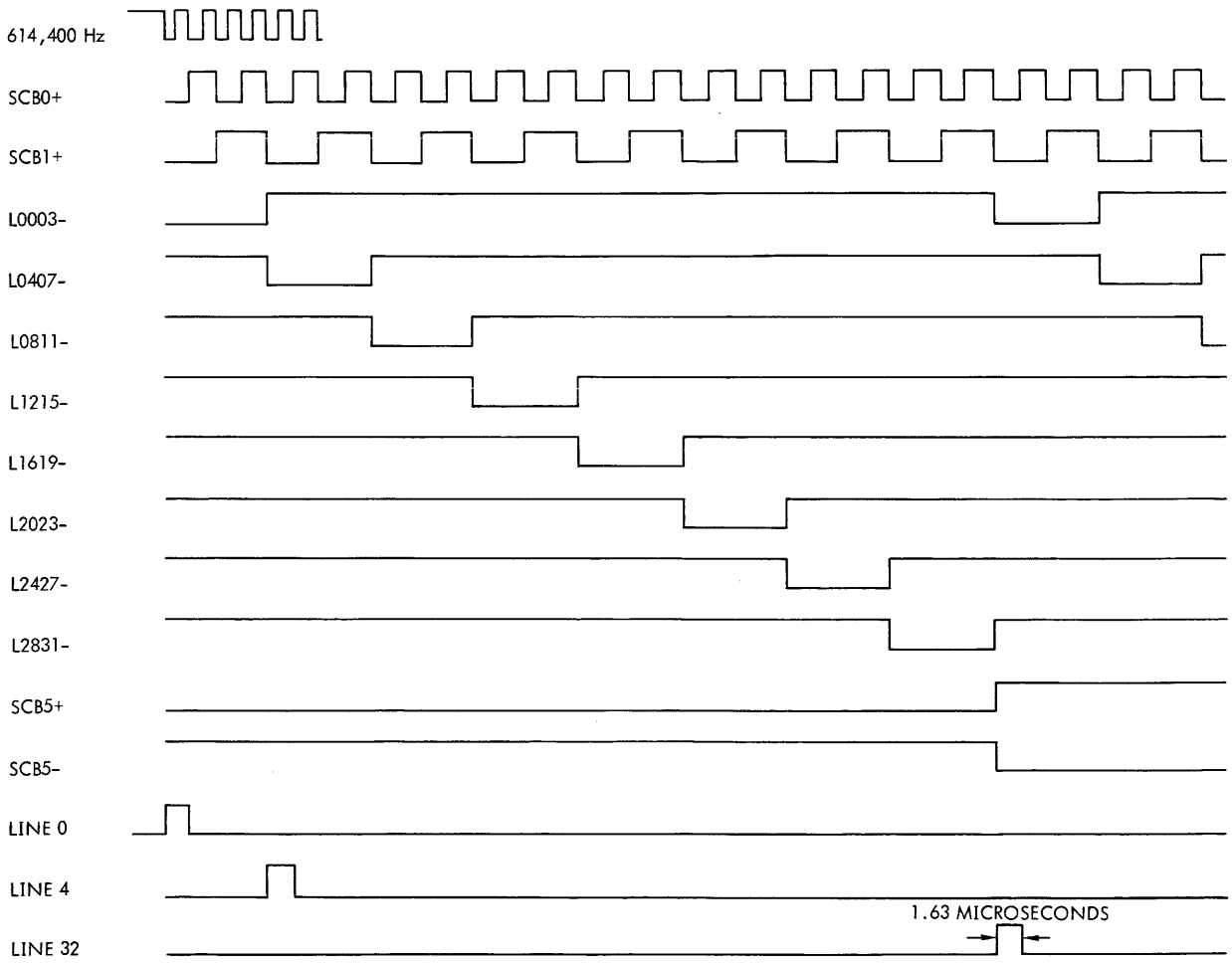


Figure 2-8. MU Scan-Counter Timing

EB14+ is true, DTI+ or DTO+, respectively, is set. The set flip-flop is then reset by DRYX+ at the end of the data-transfer. Function instructions are decoded (EB06+, EB07+, and EB08+) to form X0- thru X6-, which are enabled by EB1-.

X0- sets the system-reset flip-flop. This flip-flop is also set by the manual-reset SYRT-I. SYRT+ remains true for one complete scan cycle and is reset by LSC- at CLKE time.

X1- is the decoded clear-control logic function. It is ORed with SYRTP- to form CCLG+.

X2- enables interrupts by causing the EINT- flip-flop to set. The EINT- flip-flop is reset by X4- or IUJX-I (IUJX-I occurs when a jump-and-mark instruction is located in an interrupt trap-address and the interrupt occurs). Note that the EINT- flip-flop is clocked by FRYX+ for device address 044 as well as for the DCM device address.

X5- causes the processor control write flip-flop to set, and X6- causes read flip-flop to set. These flip-flops are cleared when a control interrupt occurs (FA15 gated with INTA+).

2.2.5 Interrupt Logic (MU Board 2)

This block generates, enables, and disables the six DCM interrupts (section 2.4). Its interface to the 620 I/O bus is similar to that of a priority-interrupt module (PIM, described in manual number 98 A 9902 42x).

2.2.6 Trap Logic (MU Board 2)

This block consists of the necessary logic to generate trap requests to the computer, a 16-bit register/counter for holding and incrementing the data address, a 12-bit register/counter for holding and decrementing the block length count, a flip-flop loaded with the byte flag, an 8-bit register for input byte-packing, and block-length-zero detection logic.

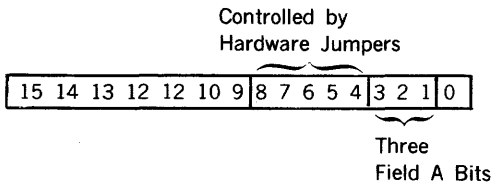
2.2.7 Control Logic (MU Board 1)

This block consists of a control memory (read-only-memory, ROM), ROM-addressing logic and ROM-decoding logic. It controls the transfer of data, control, and status information between the computer and the LAD; the generation of interrupts; control character checking; and all DMA sequences.

The control memory consists of nine 32-by-8-bit read-only memories (ROMs) arranged to produce a memory with ninety-six 24-bit words. The output of the memory is made available at J1 to facilitate testing.

The DCM uses the control memory (ROM) instead of a hardware sequencer for control. The output of the control memory is a 24-bit word that is either decoded or used directly to perform various functions (figure 2-9).

When the most-significant bit of field A is zero, the output of the field is decoded to perform the functions indicated. For interrupt generation, the most-significant bit is set. To generate an interrupt, field A is set to all-ones for one microinstruction and then set to the appropriate interrupt code. When the interrupt is actually generated, the three least-significant bits of field A are gated out on the E-bus to form the interrupt address thus:



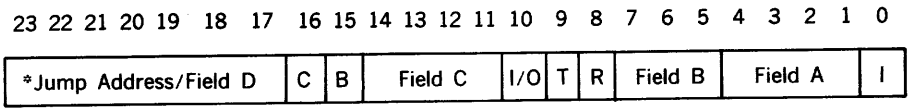
Field B initiates DMA cycles. To start a DMA cycle, field B is set to all-ones for one microinstruction and then set to the appropriate DMA transfer code. Field B is then decoded to complete the selected DMA transfer.

Field C, initiates unconditional and conditional jumpers. Bits 17 thru 23 are used as the jump address, or if field C is all zeros and bit 16 (C) is zero, these bits hold the control memory address-counter at a certain address until a specified condition is met.

Tables 2-12 through 2-17 in section 2.3.3 show the microcodes for each of the DCM sequences. The sequences are started by a request from the LADs or the computer. These requests cause the control memory address-counter to be parallel-loaded with an address corresponding to the appropriate sequence. This action is similar to the interrupt function in the computer. The following is a list of the trap addresses, each containing a jump instruction to the appropriate sequence:

0001	Output data request
0002	Input data request
0003	Status input request
0004	Processor control write
0005	Processor control read
0006	Line-error request

Figure 2-9. DCM Control Memory Word Format



I = Idle bit: 0 = Not idle; 1 = Idle

R = Routine: 0 = Output request routine; 1 = Input request routine (true for entire routine)

T = Trap: 0 = No trap; 1 = Trap (enables field-B decoder)

I/O = Input/Output trap: 1 = Input trap; 0 = Output trap (valid for all trap instructions)

B = Enable B for status-change interrupt

C = Free-running address clock (set to zero when Field D is used)

* If Field C is all zeros, this field is used to control the ROM address counter clock (Field D).

Field A	Description
0 0 0 0	
0 0 0 1	Stop scanner
0 0 1 0	
0 0 1 1	Start strobe
0 1 0 0	Enable writing
0 1 0 1	Disable writing
0 1 1 0	Increment BA, decrement BC
0 1 1 1	
1 0 0 0	Input byte-count-zero interrupt
1 0 0 1	Output byte-count-zero interrupt
1 0 1 0	Line-error interrupt
1 0 1 1	Status-change interrupt
1 1 0 0	Control-character-detected interrupt
1 1 0 1	Control interrupt
1 1 1 0	
1 1 1 1	Start interrupt

Field C	Description
0 0 0 0	
0 0 0 1	Jump
0 0 1 0	Jump if byte count zero
0 1 0 0	Jump if control character detected
1 0 0 0	Jump if byte flag set

Field B	DMA Transfer
0 0 0	Fetch/store byte count
0 0 1	Fetch/store buffer address
0 1 0	Fetch control character
0 1 1	Fetch key bits
1 0 0	Fetch/store data
1 0 1	Fetch first byte (input)
1 1 0	Fetch line-control byte
1 1 1	Start trap

Field D	Clock Address Counter If:
0 0 0 0 0 0 0	-
0 0 0 0 0 0 1	DTO set
0 0 0 0 0 1 0	DTO reset
0 0 0 0 1 0 0	DTI set
0 0 0 1 0 0 0	DTI reset
0 0 1 0 0 0 0	Trap/interrupt complete
0 1 0 0 0 0 0	No strobe (strobe complete)

The synchronization logic synchronizes DTOA+, DTIA+, CPUWA+ (processor control write request), CPURA+ (processor control read request), and TOIC+ (trap or interrupt complete) with CLKB+ before they are sent to the control memory trap logic.

The strobe logic produces a 1.63-microsecond strobe-pulse that is bussed to the LADs. The strobe is initiated by FA03–, decoded from field A of the control memory word.

The control-memory (ROM) address-counter clock consists of CLKB+ gated with an enable/disable logic structure. The clock (CACCL+) is enabled by the flip-flop CAE+. As can be seen from the logic, the status of CAE+ is updated just before the positive transition of CLKB+. This enables the clock to be disabled or enabled (without logic spike problems) by the output of the control memory. Note that the clock is constantly enabled if CMB16+ is true, or if TST+ (test mode) is true.

The control-memory trap logic remains at address 000 when in the idle state (CMB00+ high). In this state, the trap logic is enabled and a service request from the LADs (ODR–, SIR–, or IDR–), or from the processor (CPUWS+ or CPURS+) forces the control-memory address-counter to a specific address.

Processor requests are scanned when CLKE+ is high and LAD requests are scanned when CLKE– is high. This gives priority to processor requests and to give maximum settling time to the LADs after the scan counter is clocked at the positive going edge of CLKE+ (figure 2-10). Note that LERR– (line error) is ANDed with IDR (input data request).

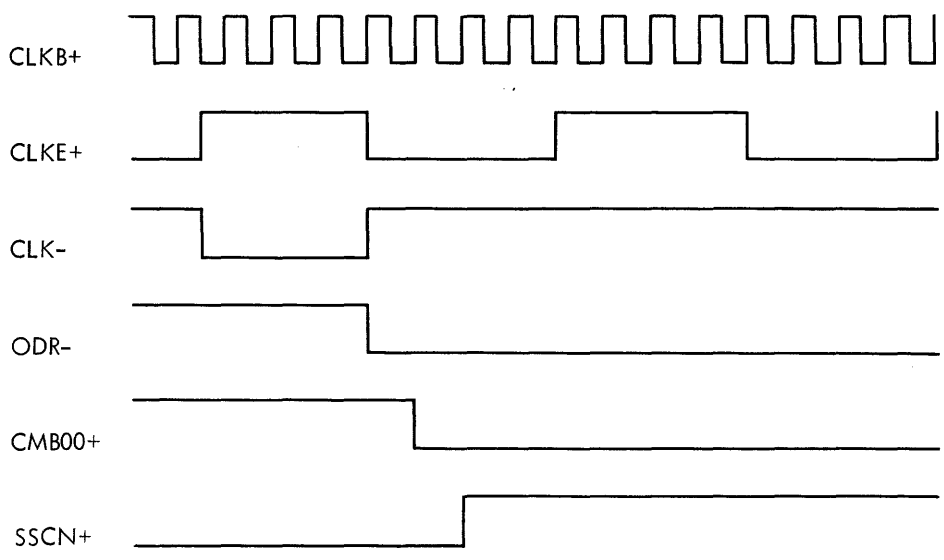
When the control memory is not in IDLE (CMB00+ low) CMB17+, CMB18+, and CMB19+ are gated to the least-significant parallel inputs of the address counters for jump instructions.

During any control memory sequence the flip-flop SSCN+ is set and remains set until the scan counter is clocked (line address is changed). This prevents repetitive servicing of the same line.

The control-memory address-counter is parallel-loaded by CACL, which is enabled when the control memory is in idle (CMB00– low), when in the test mode (TEST– low), or when CACL1– is low. CACL1– is used for jump instructions. CMB11– forces CACL1– low for unconditional jumps, CMB12+ causes a load if the byte count is zero (C2+), CMB13+ causes a load if there is a control character match (CCM+), and CMB14+ causes a load if the byte flag is set (BFA+).

The control-memory address-counter is a seven-bit synchronous counter used to address the control memory. The five least-significant bits are sent directly to the control memory and the two most-significant bits are decoded to produce CME00–, CME32–, or CME64–, which enables memory addresses 000 to 037, 040 to 077, and 0100 to 0137, or 0100 to 0177 respectively. The outputs of the counter also drive the indicators located on the back of MU board 1.

The parallel inputs to the counter are usually gated from the trap logic (CAC10+ through CAC12+) and bits 20 through 23 of the control memory (CMB20+ through CMB23+). By jumpering points X6 and X5 or grounding pin 25 of J1, the DCM is put in the test mode. The test



VTII-1911

Figure 2-10. LAD Request Timing

mode gates the switches TS0 through TS6 or pins 26 through 32 of J1 to the counter. The test mode also enables CACL– and causes CACCL + to run free. The test mode enables the counter to be forced to any control-memory address using the switches or the J1 inputs (Note: all the switches must be opened to use the J1 inputs).

2.2.8 Control-Character Checking Logic (MU Board 1)

This block contains data-comparison logic. The 16-bit register/counter is loaded via DMA with a specific line's two control characters. The two control characters are then compared to the data character in the interface buffer. If the buffered character is one of the control characters, a control-character interrupt is generated.

2.2.9 Clock and Bit-Rate Generator (MU Board 1)

This block contains a crystal controlled 9.8304 MHz clock, a standard bit-rate generator and two variable bit-rate generators. The standard bit-rate generator provides the following rates:

16 x 9600 = 153,600 Hz

16 x 4800 = 76,800 Hz

16 x 2400 = 38,400 Hz

16 x 1200 = 19,200 Hz

16 x 600 = 9,600 Hz

16 x 300 = 4,800 Hz

16 x 150 = 2,400 Hz

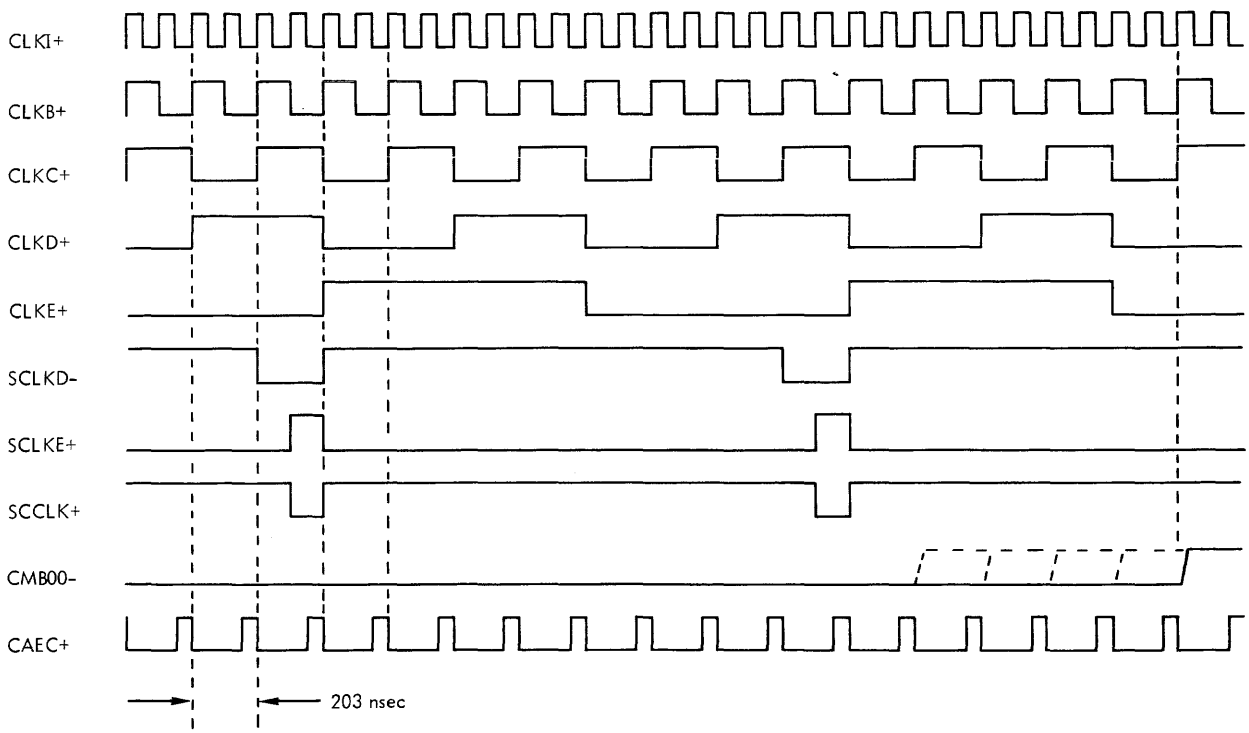
16 x 75 = 1,200 Hz

Any of these rates can be jumpered (section 2.1.3) to any of six buffer gates, which in turn distribute the bit-rates to the LADs. The variable bit-rate generators provide the capability of dividing 9.8304, 4.9152, 2.4576, 1.2288, 0.6144, or 0.3072 MHz by an integer from 1 to 256 to give a rate (150 kHz maximum) that can be jumpered to any of the six buffer gates.

The crystal-controlled 9.8304 MHz oscillator produces a square wave (CLKI +), which is immediately divided down to produce CLKB + , the DCM main clock. It is divided down by a synchronous counter to produce CLKC + , CLKD + , and CLKE + (figure 2-11).

The standard bit rates are generated by simply continuing to count down CLKE + .

The odd-bit-rate counters consist of modulo-four synchronous counters. Counter 1 contains three modules and counter 2 contains two. The counters divide down the input frequency and count until they overflow. Upon overflow, they are loaded with the selected count and continue



VTII-1912

Figure 2-11. MU Clock Timing Diagram

counting. The carry output rate of the last counter module is divided down again with a J-K flip-flop to produce a square wave output.

2.2.10 Multiplexor Bus (MU Board 2)

The eight-bit multiplexor data bus is bidirectional. The direction of the transfer is controlled by the state of the WRITE+ flip-flop, which is controlled by FA04+ and FA05+, the decoded signals from the A field of the control-memory word. WRITE– is also bussed to the LADs. Data transferred from the LADs are stored in an eight-bit register until the transfer to the computer is complete. The register is loaded by SSCN+, and the output of the register is buffered and sent to MU board 2.

AC+ and BC+ from the interface buffer are sent to the LAD as A– and B–. These signals are decoded on the LADs to identify the type of transfer being made. Note that B– is also forced low by CMB15+ for the status-change sequence and causes the LAD to gate the line status onto the data bus.

CLKE+ is sent to the LADs as CLK–. The LADs place service requests on the line on the positive transition of the clock. Once a request is recognized and a control memory sequence begun, CLK– is held high by SSCN–.

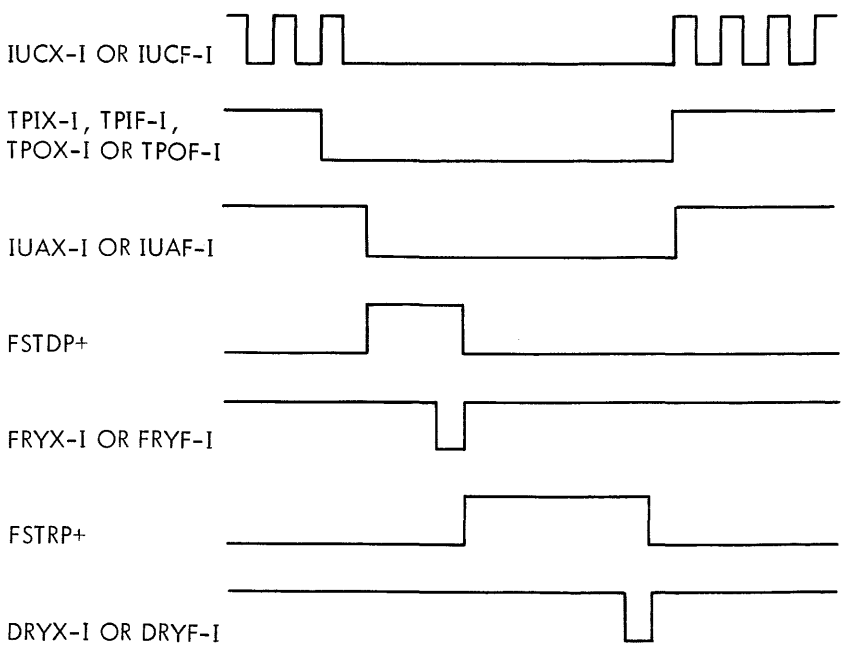
The key bits are fetched from processor memory and stored in the four-bit register. The key bits are then gated to the processor during the address phase (EDR1+) of a data-fetching or store-DMA cycle.

2.2.11 DMA Logic (MU Board 2)

The DCM is capable of operating with the standard DMA or it can use the V70 series high-speed DMA. The operation is identical in either case, so only the standard DMA will be discussed here (figure 2-12).

A trap operation is initiated by setting flip-flop ITRP (FB7–). ITRP+ is then delayed one CLKB+ pulse to allow the trap address to settle (IFTRP+). After this delay, the trap request flip-flop is set (STRQ+) and, if the DCM has priority (PRMX+), a trap-out request (TPOX-I) or a trap-in request (TPIX-I) is made, depending on the state of CMB10–. The trap flip-flop (STRAP+) is set at FRYX+ time when the request has been acknowledged (IUAX+). At the end of the cycle, it is reset by STRQ+.

The byte-count register/counter is a twelve-bit ripple counter with parallel-loading capability. The counter is parallel-loaded by the DMA and incremented by CLKCS+, which is initialized by FA06–. This counter holds the input or output byte-count (buffer length) during an input or output data seqeuence. The byte-count is always decremented before it is stored back in memory. Because it is stored in ones-complement form in the computer memory, the byte-count is decremented by clocking the counter. The gating that produces CZ+ (count zero) is used to check for all-ones in the counter. When the byte-count is fetched from memory, the BF+ bit and the SMB bit are fetched. The BF+ bit (EB15+) indicates whether the data byte to be fetched is in the upper or lower eight-bit portion of the buffer word being accessed. In a storage operation (input sequence), BF+ indicates where to store the data byte. The SMB bit has significance only in an input sequence. If



VTII-1913

Figure 2-12. DMA Timing

the SMB bit is set in memory, the DCM hardware forces the most significant bit to one on all input data. If it is not set, the data are stored as they come from the line. The SMB + flip-flop is held reset unless CMB08 + is true. CMB08 + is true only during an input sequence.

The DMA 16-bit register/counter is a 16-bit ripple counter with parallel-loading capability. It is parallel-loaded during a DMA cycle. It holds the input or output buffer address during an input or output sequence, and also holds the control characters during the control-character check phase of an input sequence. During the control-character check, the output of the counter is compared with the multiplexor bus bits and the signal CCM– goes true if there is a match. The counter also contains logic for incrementing its contents. If the byte flag flip-flop is reset (BF– true), the counter is incremented by CLKCS + . This increments the buffer address before it is transferred back to processor memory if the second byte has been input or output.

An interrupt is initiated by FA17– which sets flip-flop INTR + . Interrupt request (INTR +) is then delayed by CLKB + to allow the interrupt address to settle (INTDL–). Once INTDL– is set, the interrupt flip-flop (INT +) is set on the next IUCX– clock if the interrupts are enabled (EINT–). INT + makes an interrupt request if the DCM has priority (PRMX +), and transfers the interrupt address onto the E-bus (INTA + and INTB +) when the interrupt is acknowledged (IUAX +). INTA + clears the interrupt request (INTR + and INTDL–), and the INT + flip-flop clears on the first IUCX– clock after the interrupt is complete.

2.3 MAINTENANCE

The following equipment is required to test the MU:

a. Varian 620-series or V70 series computer with at least 8K of memory

b. An I/O expansion chassis for the DCM backplane(s)

c. An I/O expansion power supply

d. One card extender (44 D 0015-000 or 44 D 0540-000)

e. Tektronix Type 547 or equivalent oscilloscope

f. At least one asynchronous RS232 interface line adaptor (01 A 1470-000) with two test connectors (burndy edge connectors 57 A 0036-000) wired as follows:

Line 0

1.	Pin 29 to pin 35	Data-terminal-ready to ring-indicator
	Pin 35 to pin 33	Ring-indicator to dataset-ready
	Pin 29 to pin 25	Data-terminal-ready to carrier-on
2.	Pin 27 to pin 37	Request-to-send to clear-to-send
3.	Pin 31 to pin 23	Transmit-data to receive-data
4.	Pin 21 to pin 18	Control-out to control-in

Line 1

5.	Pin 1 to pin 13	Data-terminal-ready to ring-indicator
	Pin 13 to pin 11	Ring-indicator to dataset-ready
	Pin 1 to pin 7	Data-terminal-ready to carrier-on
6.	Pin 3 to pin 19	Request-to-send to clear-to-send
7.	Pin 5 to pin 9	Transmit-data to receive-data
8.	Pin 6 to pin 8	Control-out to control-in

The following additional jumpers can be added so the test connector can also be used on a synchronous RS232 interface LAD.

Pin 38 to pin 39
Pin 39 to pin 41
Pin 38 to pin 17
Pin 17 to pin 15

These jumpers tie the test clock to the receive/transmit clock inputs.

When testing the MU, the asynchronous RS232 interface LAD lines should be set up for eight-bit data, two stop bits, no parity.

g. Software tapes

MAINTAIN II Test Executive	92 U 0107-001
DCM Test Program	92 U 0106-009C

The following system configuration features must be checked before testing:

a. Verify that the following features are jumpered according to special system specifications, if any, or to the standard configuration:

1. Odd-bit-rate counters 1 and 2
2. The source for the six bit-rate (line-speed) signals
3. Scan length
4. Priority mode
5. Device address
6. E-bus priority chain wiring
7. Base interrupt address
8. Base control table address (LCB)
9. High-speed or normal DMA selection

b. If the system term shoe is located in the DCM, verify that + 5V dc is wired to the term shoe slot on pins 118 and 121.

c. Verify that the proper bit rates are jumpered into the line adaptor inputs. These clocks can be changed to higher speeds to facilitate testing, but must be rewired to the standard configuration or system requirements before shipping.

2.3.1 Functional Tests Requiring the DCM Test Program

The following tests use portions of the DCM test program. The specific test required is called out in each section.

Input and output-sequence test: Verify that the multiplexor can successfully transfer a block of data out and transfer a block of data into the computer memory, and that an input or output byte-count-zero interrupt is generated at the end of the transfers. This can be verified by test 1. This test should be run on all lines of the DCM under test. If only one LAD is available, move it from slot to slot, repeating the test after each move.

Control-character detection test: Verify that the multiplexor can detect any possible eight-bit control character and that it only gives a control-character-detected interrupt on the correct control character. This can be checked by test 2.

Framing-error detection test: Verify that the multiplexor can detect and report a framing error to the computer. This can be checked by test 4.

Parity-error detection test: Verify that the multiplexor can detect and report a parity error to the computer. Simultaneously ground LERR– and BI– on the DCM backplane (pins 2 and 11 of any LAD slot) while test 1 is running. The test program will print out the error.

Overrun-error detection test: Verify that the multiplexor can detect and correctly report an overrun error. Use test 5. Data pattern 2 (ascending binary) should be selected.

Status-change sequence test: Verify that the multiplexor can detect a status-change request from a line and properly complete a status-change sequence. Test 5 checks this.

Processor control-read sequence test: Verify that the multiplexor will respond to a computer control-read request and successfully complete the sequence. Use test 6.

Byte-count register/counter test: Use test 7 to verify that a 4095-byte block of data can be transferred in or out on any line.

VDM ASC II Test: Verify that the multiplexor will force the most-significant bit (bit seven) of all data input from a line to a one if the SMB bit (bit fourteen) in the input-byte-count word of the line control table is set (one). Test 17 checks this.

Burn-in: Run test 77 for at least 8 hours consecutively on each of four lines.

2.3.2 Tests Not Covered by the DCM Test Program

Power: Test at slot 9 of the DCM main backplane for the following voltages:

Pin	Voltage
118	+5± 0.2V dc
120	+12± 0.25V dc
119	–12± 0.25V dc

Timing: Verify that the following timing singals are all running within ± 0.01 percent of the correct frequencies. All of the signals are square-waves and are located on MU board 1.

Signal	Frequency	Test Point (MU Board 1)
CLKI +	9.8304 MHz	V19A
CLKB +	4.9152 MHz	P1 pin 50
CLKC +	2.4576 MHz	V15
CLKD +	1.2288 MHz	V16
CLKE +	0.6144 MHz	V17
CLKF +	307,200 Hz	V18
SBR1 +	153,600 Hz	W27
SBR2 +	76,800 Hz	W28
SBR3 +	38,400 Hz	W29
SBR4 +	19,200 Hz	W30
SBR5 +	9,600 Hz	W31
SBR6 +	4,800 Hz	W32
SBR7 +	2,400 Hz	W33
SBR8 +	1,200 Hz	W34

The following signals depend on customer-selected options. The frequencies given are the standard frequencies used if no special requirements are specified. Verify that the signals are square waves and are operating within ± 1 percent of the frequencies listed, or at the customer-selected frequencies. All of the listed test points are on MU board 1.

Signal	Frequency	Test Point (MU Board 1)
OBR1 +	1,760 Hz	X33
OBR2 +	4.9152 MHz	X34
BR0–	153,600 Hz	P1 pin 101
BR1–	76,800 Hz	P1 pin 103
BR2–	19,200 Hz	P1 pin 105
BR3–	4,800 Hz	P1 pin 106
BR4–	2,400 Hz	P1 pin 104
BR5–	1,760 Hz	P1 pin 102

Scan Counter: Place the DCM in idle. The control-memory address-counter indicators will all go out. Verify that the scan-counter is properly scanning the correct number of lines. The scan-counter can be strapped to scan 4, 8, 16, 32, or 64 lines and this is varied per top assembly instructions or system specifications. After establishing the scan-length for the system under test, check the following signals. The pulse width is a square wave for the SCB series, and 6.5 microseconds ± 1 percent for the L series.

Scan Length	Mnemonic	Test Point	Frequency
4,8,16,32,64	SCB0-	P1 pin 97	307,200 Hz
4,8,16,32,64	SCB1-	P1 pin 96	153,600 Hz
4,8,16,32	SCB5-	P1 pin 99	High level
64	SCB5-	P1 pin 99	9,600 Hz
4,8,16,32	SCB5+	P1 pin 100	Low level
64	SCB5+	P1 pin 100	9,600 Hz
4	L0003-	P1 pin 23	153,600 Hz
8	L0003-	P1 pin 23	76,800 Hz
16	L0003-	P1 pin 23	38,400 Hz
32,64	L0003-	P1 pin 23	19,200 Hz
8	L0407-	P1 pin 22	76,800 Hz
16	L0407-	P1 pin 22	38,400 Hz
32,64	L0407-	P1 pin 22	19,200 Hz
16	L0811-	P1 pin 21	38,400 Hz
32,64	L0811-	P1 pin 21	19,200 Hz
16	L1215-	P1 pin 20	38,400 Hz
32,64	L1215-	P1 pin 20	19,200 Hz
32,64	L1619-	P1 pin 27	19,200 Hz
32,64	L2023-	P1 pin 26	19,200 Hz
32,64	L2427-	P1 pin 25	19,200 Hz
32,64	L2831-	P1 pin 24	19,200 Hz

Control-Memory Address Switches and Indicators: Ground J1 pin 25 or jumper pin X6 to X5 on MU board 1 and verify that the control memory address indicators (DS1 through DS7 on MU Board 1) can be turned on with the switches at location D10. When the side of the switch-rocker with the black dot is depressed, the switch is activated.

Control-Memory Output: Verify that the control-memory test-points at J1 (pins 1 through 24) are all high (except pins 24 and 8) when the DCM is in idle (control-memory address 000).

2.3.3 Suggested Methods

DCM test program - Test 0: Test 0 of the test program is a transmit-only test and is helpful in troubleshooting.

Sample program: The following program will be helpful in troubleshooting problems that cannot be found with the DCM test program. The program starts a transfer and at the end of the transfer (input or output byte-count zero), it initializes the DCM and begins again. If sense switch 3 is set, the program jumps back to the test executive at the completion of the transfer. Note that the output byte-count-zero interrupt occurs before the last two bytes have actually been transmitted, so these bytes are lost.

Address (Octal)	Code (Octal)	Instruction	Remarks
000200	002000	JMPM	Call LCB setup
000201	000215		
000202 *	100070	EXC 070	Initialize
000203 *	100270	EXC 270	Enable interrupts
000204 *	100570	EXC 570	Request control write
000205	005000	NOP	
000206	001000	JMP	Wait for interrupt
000207	000205		
		LCB Setup	
000215	000000		
000216	006050	STAI	Store A
000217	000000		
000220	006010	LDAI	Load A
000221 **	107770		Input byte-count word
000222	057500	STA	Store A in LCB
000223	006010	LDAI	Load A
000224 **	000600		Input buffer-address word
000225	057501	STA	Store A in LCB
000226	006010	LDAI	Load A
000227 **	107770		Output byte-count word
000230	057502	STA	Store A in LCB
000231	006010	LDAI	Load A
000232 **	000650		Output buffer-address word
000233	057503	STA	Store A
000234	010217	LDA	Load A (restore)
000235	001000	JMP	Jump back
000236	100215		

* Change these instructions to reflect the correct device address if a nonstandard address is used.

** Enter the byte counts and buffer addresses desired.

Address (Octal)	Code (Octal)	Instruction	Remarks
		Line Setup	
000240	000000		
000241	006050	STAI	Save A
000242	000000		
000243	017504	LDA	Load A with control word
000244 *	103170	OAR	Output to DCM
000245	010242	LDA	Restore A
000246 *	100270	EXC 270	Enable interrupts
000247	0001000	JMP	Return
000250	1000240		
		Input or Output Byte-Count Zero	
000260	000000		
000261	100070	EXC 070	Initialize
000262	001400	J SS3	Jump to
000263	006221		Test executive
000264	001000	JMP	Return to
000265	000200		Start

Interrupt Traps

000060	002000	JMPM
000061	000260	
000062	002000	JMPM
000063	000260	
000064	001000	JMP
000065	000270	
000066	001000	JMP
000067	000271	
000070	001000	JMP
000071	000272	
000072	002000	JMPM
000073	000240	

Address (Octal)	Code (Octal)	Instruction	Remarks

Interrupt Halts

000270	000004	Line error interrupt
000271	000006	Status change interrupt
000272	000001	Control-character-detected interrupt

* Change these instructions to reflect the correct device address if a nonstandard address is used.

LCB Address

The LCB addresses for the line under test are entered here.

Address (Octal)	Code (Octal)	Instruction	Remarks
000500	017000		Input block-length address
000501	017001		Input buffer-address address
000502	017004		Output block-length address
000503	017005		Output buffer-address address
000504	017006		Control-word address

Line Control Block (Line zero shown - base address 017000)

017000	107770	Input block-length
017001	000600	Input buffer-address
017002	177777	Control characters
017003	000000	Spare
017004	107770	Output block-length
017005	000650	Output buffer-address
017006	005500	Line control-word (includes line address)
017007	000000	Spare

Output Buffer

000650	125252
000651	125252
000652	125252
000653	125252

Address (Octal)	Code (Octal	Instruction	Remarks
		Input Buffer	
000600	000000		
000601	000000		
000602	000000		
000603	000000		

Test Features: The multiplexor has a built-in test feature that allows the control memory to be locked at any address. This allows static testing of the interrupt logic, trap logic, etc. The test mode is enabled by jumpering points X6 and X5 on MU board 1. Once test mode is enabled, the control memory address can be controlled by the DIP switch located at location D10 on MU board 1. The seven indicators on the same board indicate the control memory address. These indicators are used in normal operation to indicate what operation the DCM was performing if it hangs up. The sequence tables (tables 2-12 through 2-19) indicate the instructions stored at the various control memory addresses.

The parallel inputs to the control-memory address-counter are also brought out to J1 on MU board 1, along with the outputs of the control memory. This allows attachment of a test device to the DCM and reading of the control-memory contents. This is done by grounding J1 pin 25 (enable test mode), ensuring that all the switches at D10 are off, and entering the desired address at J1 (see logic diagram 91B0412 for the correct pin numbers). Table 2-20 gives the control-memory contents for a standard DCM.

Overloading: If a large number of high-speed lines are in use in the DCM, comparison errors and overrun errors may show up when trying to run the DCM test program. This is especially true if the DCM under test is set up to scan the maximum 64 lines. If this occurs, try testing the lines in question at lower speeds.

Table 2-12. Output Sequence

ADDRESS							REMARKS	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
6	5	4	3	2	1	0		Jump Address/Field D							C	B	Field C				I/O	T	R	Field B			Field A				1
0	0	0	0	0	0	1	Stop scan; start trap; jump	0	0	0	1	0	0	0	1	0	0	0	0	1	0	1	0	1	1	1	0	0	0	1	0
0	0	0	1	0	0	0	Fetch byte count, enable write, wait for trap complete	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0
0	0	0	1	0	0	1	Test byte count, start trap	0	0	1	0	1	1	0	1	0	0	0	1	0	0	1	0	1	1	1	0	0	0	0	0
0	0	0	1	0	1	0	Fetch buffer address, wait for trap complete	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0
0	0	0	1	0	1	1	Start trap	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	1	1	1	0	0	0	0	0
0	0	0	1	1	0	0	Fetch data, wait for trap complete	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
0	0	0	1	1	0	1	Start strobe	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
0	0	0	1	1	1	0	Start trap, test byte flag, inc.	0	0	1	0	0	0	1	1	0	1	0	0	0	1	1	0	1	1	1	0	1	1	0	0
0	0	0	1	1	1	1	Store buffer address, wait for trap complete	0	0	1	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	1	0	0	0	0	0
0	0	1	0	0	0	0	Start trap	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1	0	1	1	1	0	0	0	0	0
0	0	1	0	0	0	1	Store byte count, wait for trap complete	0	0	1	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
0	0	1	0	0	1	0	Test byte count zero	0	0	1	1	0	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	1	1	Wait for strobe complete	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	1	0	0	Back to zero, disable write	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	1	0	1	0
0	0	1	0	1	1	0	Fetch line control byte, wait for trap complete	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0
0	0	1	0	1	1	1	Start strobe, jump	0	0	1	0	0	1	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	1	1	0
0	0	1	1	0	0	1	Gen. int. (output byte-count zero)	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0
0	0	1	1	0	1	0	Wait for DTO set	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0
0	0	1	1	0	1	1	Wait for DTO reset	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	1	1	0	0	Jump	0	0	1	0	0	1	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	1	1	0

Table 2-13. Input Sequence

ADDRESS	REMARKS	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
6 5 4 3 2 1 0		Jump Address/Field D							C	B	Field C				I/O	T	R	FieldB			Field A				1
0 0 0 0 0 1 0	Stop scan, start trap, jump	0	1	0	0	0	0	0	1	0	0	0	0	1	0	1	1	1	1	1	0	0	0	1	0
0 1 0 0 0 0 0	Fetch byte count, wait for trap complete	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
0 1 0 0 0 0 1	Test byte count, strobe LAD	0	1	1	1	1	0	1	1	0	0	0	1	0	0	0	1	0	0	0	0	0	1	1	0
0 1 0 0 0 1 0	Start trap	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0
0 1 0 0 0 1 1	Fetch B.A., unit for trap comp.	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	0
0 1 0 0 1 0 0	Start trap, test B.F.	0	1	0	0	1	1	1	1	0	1	0	0	0	0	1	1	1	1	1	0	0	0	0	0
0 1 0 0 1 0 1	Fetch most significant byte, wait for trap complete	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	1	0	0	0	0	0
0 1 0 0 1 1 0	Start trap	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0
0 1 0 0 1 1 1	Store date, wait for trap complete	0	0	1	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0
0 1 0 1 0 0 0	Start trap, clock B.A. & B.C., jump if B.F. set	0	1	0	1	0	1	1	1	0	1	0	0	0	1	1	1	1	1	1	0	1	1	0	0
0 1 0 1 0 0 1	Store buffer address, wait for trap complete	0	0	1	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	1	0	0	0	0	0
0 1 0 1 0 1 0	Start trap	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0
0 1 0 1 0 1 1	Store byte count, wait for trap complete	0	0	1	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
0 1 0 1 1 0 0	Store trap	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0
0 1 0 1 1 0 1	Fetch control characters, wait for trap complete	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0
0 1 0 1 1 1 0	Test for control character	0	1	1	0	0	1	0	1	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0
0 1 0 1 1 1 1	Test for byte count zero	0	1	1	1	0	0	1	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0
0 1 1 0 0 0 0	Jump back to zero	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0 1 1 0 0 1 0	Generate interrupt	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	1	1	1	1	0
0 1 1 0 0 1 1	Control character detected interrupt; wait for DTO	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	1	1	0	0	0
0 1 1 0 1 0 0	Wait for DTO reset; enable write	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0
0 1 1 0 1 0 1	Start strobe	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1	0
0 1 1 0 1 1 0	Wait for strobe complete; disable write	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
0 1 1 0 1 1 1	Jump to byte-count test	0	1	0	1	1	1	1	1	0	0	0	0	1	0	0	1	0	0	0	0	1	0	1	0
0 1 1 1 0 0 0																									
0 1 1 1 0 0 1	Generate interrupt	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	1	1	1	1	0
0 1 1 1 0 1 0	Input byte count zero interrupt; wait for DTO set	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0
0 1 1 1 0 1 1	Wait for DTO reset; enable write	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0
0 1 1 1 1 0 0	Start strobe	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
0 1 1 1 1 0 1	Wait for strobe complete	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0 1 1 1 1 1 0	Jump to zero; disable write	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	1	0	1	0

Table 2-14. Input Sequence (Memory Map System)

ADDRESS							REMARKS	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
6	5	4	3	2	1	0		Jump Address/ Field D							C	B	Field C				I/O	T	R	Field B			Field A				1
0	0	0	0	0	1	0	Stop scan, start trap, jump	0	1	0	0	0	0	0	1	0	0	0	0	1	0	1	1	1	1	1	0	0	0	1	0
0	1	0	0	0	0	0	Fetch byte count, wait for trap comp.	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
0	1	0	0	0	0	1	Test byte count, strobe LAD	0	1	1	1	1	0	1	1	0	0	0	1	0	0	0	1	0	0	0	0	0	1	1	0
0	1	0	0	0	1	0	Start trap, jump	1	0	1	1	1	0	1	1	0	0	0	0	1	0	1	1	1	1	1	0	0	0	0	0
0	1	0	0	0	1	1	Fetch B.A., unit for trap comp.	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	0
0	1	0	0	1	0	0	Start trap, test B.F.	0	1	0	0	1	1	1	1	0	1	0	0	0	0	1	1	1	1	1	0	0	0	0	0
0	1	0	0	1	0	1	Fetch most significant byte, wait for trap comp.	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	1	0	0	0	0	0
0	1	0	0	1	1	0	Start trap	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0
0	1	0	0	1	1	1	Store data, wait for trap comp.	0	0	1	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0
0	1	0	1	0	0	0	Start trap, clock B.A.&B.C. jump. If B.F. set	0	1	0	1	0	1	1	1	0	1	0	0	0	1	1	1	1	1	1	0	1	1	0	0
0	1	0	1	0	0	1	Store buffer address, wait for trap comp.	0	0	1	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	1	0	0	0	0	0
0	1	0	1	0	1	0	Start trap	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0
0	1	0	1	0	1	1	Store byte count, wait for trap comp.	0	0	1	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
0	1	0	1	1	0	0	Start trap	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0
0	1	0	1	1	0	1	Fetch control characters, wait for trap comp.	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0
0	1	0	1	1	1	0	Test for control character	0	1	1	0	0	1	0	1	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0
0	1	0	1	1	1	1	Test for byte count zero	0	1	1	1	0	0	1	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0
0	1	1	0	0	0	0	Jump back to zero	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	1	1	0	0	1	0	Gen. Int.	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	1	1	1	1	0
0	1	1	0	0	1	1	Control char. detected int. wait for DTO	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	1	1	0	0	0
0	1	1	0	1	0	0	Wait for DTO reset enable write	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0
0	1	1	0	1	0	1	Start strobe	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1	0
0	1	1	0	1	1	0	Wait for strobe complete disable write	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
0	1	1	0	1	1	1	Jump to byte count test	0	1	0	1	1	1	1	1	0	0	0	0	1	0	0	1	0	0	0	0	1	0	1	0
0	1	1	1	0	0	1	Gen. Int.	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	1	1	1	1	0
0	1	1	1	0	1	0	Input byte count zero int., wait for DTO set	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0
0	1	1	1	0	1	1	Wait for DTO reset, enable write	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0
0	1	1	1	1	0	0	Start strobe	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
0	1	1	1	1	0	1	Wait for strobe comp.	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	1	1	1	1	0	Jump to zero disable write	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	1	0	1	0
1	0	1	1	1	0	1	Wait for trap, fetch key bits	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	1	0	0	0	0	0
1	0	1	1	1	1	0	Jump, start trap	0	1	0	0	0	1	1	1	0	0	0	0	1	0	1	1	1	1	1	0	0	0	0	0

Table 2-15. Output Sequence (Memory Map System)

ADDRESS							REMARKS	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
6	5	4	3	2	1	0		Jump Address/Field D							C	B	Field C				I/O	T	R	Field B			Field A				1
0	0	0	0	0	0	1	Stop scan; start trap; jump	0	0	0	1	0	0	0	1	0	0	0	0	1	0	1	0	1	1	1	0	0	0	1	0
0	0	0	1	0	0	0	Fetch byte count, enable write, wait for trap comp.	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0
0	0	0	1	0	0	1	Test byte count, start trap	0	0	1	0	1	1	0	1	0	0	0	1	0	0	1	0	1	1	1	0	0	0	0	0
0	0	0	1	0	1	0	Fetch buffer address, wait for trap comp.	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0
0	0	0	1	0	1	1	Start trap; jump	0	0	1	1	1	0	1	1	0	0	0	0	1	0	1	0	1	1	1	0	0	0	0	0
0	0	0	1	1	0	0	Fetch data, wait for trap comp.	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
0	0	0	1	1	0	1	Start strobe	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
0	0	0	1	1	1	0	Start trap, test byte flag, in.	0	0	1	0	0	0	1	1	0	1	0	0	0	1	1	0	1	1	1	0	1	1	0	0
0	0	0	1	1	1	1	Store buffer address, wait for trap comp.	0	0	1	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	1	0	0	0	0	0
0	0	1	0	0	0	0	Start trap	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1	0	1	1	1	0	0	0	0	0
0	0	1	0	0	0	1	Store byte count, wait for trap comp.	0	0	1	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
0	0	1	0	0	1	0	Test byte count zero	0	0	1	1	0	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	1	1	Wait for strobe comp.	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	1	0	0	Back to zero, disable write	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	1	0	1	0
0	0	1	0	1	1	0	Fetch line control byte, wait for trap comp.	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0
0	0	1	0	1	1	1	Start strobe, jump	0	0	1	0	0	1	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	1	1	0
0	0	1	1	0	0	1	Gen. Int. (output byte count zero)	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0
0	0	1	1	0	1	0	Wait for DTO set	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0
0	0	1	1	0	1	1	Wait for DTO reset	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	1	1	0	0	Jump	0	0	1	0	0	1	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	1	1	0
0	0	1	1	1	0	1	Fetch key bits; wait	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0
0	0	1	1	1	1	0	Start trap; jump	0	0	0	1	1	0	0	1	0	0	0	0	1	0	1	0	1	1	1	0	0	0	0	0

Table 2-16. Status Input Sequence

ADDRESS							REMARKS	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
6	5	4	3	2	1	0		Jump Address/Field D							C	B	Field C				I/O	T	R	Field B			Field A				1
0	0	0	0	0	1	1	Stop scan; jump; enable B	1	0	0	0	0	0	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0
1	0	0	0	0	0	0	Generate interrupt enable	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0
1	0	0	0	0	0	1	Status change interrupt, wait for DTO set	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0
1	0	0	0	0	1	0	Strobe	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
1	0	0	0	0	1	1	Wait for strobe complete	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	Wait for DTO reset; set write	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
1	0	0	0	1	0	1	Strobe	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
1	0	0	0	1	1	0	Wait for strobe complete	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	1	1	1	Jump to zero; disable write	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	1	0	1	0

Table 2-17. Processor Write Sequence

ADDRESS							REMARKS	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
6	5	4	3	2	1	0		Jump Address/Field D							C	-	Field C				I/O	T	R	Field B			Field A				1
0	0	0	0	1	0	0	Stop scan; jump	1	0	0	1	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0
1	0	0	1	0	0	0	Generate interrupt	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0
1	0	0	1	0	0	1	Control int.; wait for DTO set	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0
1	0	0	1	0	1	0	Wait for DTO reset, enable write	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
1	0	0	1	0	1	1	Start strobe	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
1	0	0	1	1	0	0	Wait for strobe complete	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	1	1	0	1	Disable write, jump to zero	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	1	0	1	0

Table 2-18. Processor Read Sequence

ADDRESS							REMARKS	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
6	5	4	3	2	1	0		Jump Address/Field D							C	-	Field C				I/O	T	R	Field B			Field A				1
0	0	0	0	1	0	1	Jump	1	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
1	0	1	0	0	0	0	Generate interrupt	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0
1	0	1	0	0	0	1	Control interrupt; wait for DTI set	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0
1	0	1	0	0	1	0	Wait for DTI reset	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	1	0	0	1	1	Jump to zero; stop scan	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0

Table 2-19. Line Error Sequence

ADDRESS							REMARKS	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
6	5	4	3	2	1	0		Jump Address/Field D							C	—	Field C				I/O	T	R	Field B			Field A				1
0	0	0	0	1	1	0	Stop scan, jump	1	0	1	0	1	1	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0
1	0	1	0	1	1	1	Gen. Int.	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0
1	0	1	1	0	0	0	Line error Int.; wait for DTO set	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
1	0	1	1	0	0	1	Wait for DTO reset setwrite	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
1	0	1	1	0	1	0	Strobe	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
1	0	1	1	0	1	1	Wait for strobe comp.	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	1	1	1	0	0	Jump, disable write, start trap	0	1	0	0	0	0	0	1	0	0	0	0	1	0	1	1	1	1	1	0	1	0	1	0

Table 2-20. Control Memory Contents

Address	ROM Words		
0000000	00000001	00000000	00000001
0000001	00010001	00001010	11100010
0000010	01000001	00001011	11100010
0000011	10000001	10001000	00000010
0000100	10010001	00001000	00000010
0000101	10100001	00001000	00000000
0000110	10101111	00001000	00000010
0000111	00000000	00000000	00000000
0001000	00100000	00000010	00001000
0001001	00101101	00010010	11100000
0001010	00100000	00000010	00100000
0001011*	00000001	00000010	11100000
0001100	00100000	00000010	10000000
0001101	00000001	00000000	00000110
0001110	00100011	01000110	11101100
0001111	00100000	00000110	00100000
0010000	00000001	00000110	11100000
0010001	00100000	00000110	00000000
0010010	00110011	00010000	00000000
0010011	01000000	00000000	00000000
0010100	00000001	00001000	00001010
0010101	00000000	00000000	00000000
0010110	00100000	00000010	11000000
0010111	00100111	00001000	00000110
0011000	00000000	00000000	00000000
0011001	00000001	00000000	00011110
0011010	00000010	00000000	00010010
0011011	00000100	00000000	00000000
0011100	00100111	00001000	00000110
0011101*	00000000	00000000	00000000
0011110*	00000000	00000000	00000000
0011111	00000000	00000000	00000000
	↑ MSB		↑ LSB

Table 2-20. Control Memory Contents *(continued)*

Address	ROM Words		
0100000	00100000	00000011	00000000
0100001	01111011	00010001	00000110
0100010*	00000001	00000011	11100000
0100011	00100000	00000011	00100000
0100100	01001111	01000011	11100000
0100101	00100000	00000011	10100000
0100110	00000001	00000111	11100000
0100111	00100000	00000111	10000000
0101000	01010111	01000111	11101100
0101001	00100000	00000111	00100000
0101010	00000001	00000111	11100000
0101011	00100000	00000111	00000000
0101100	00000001	00000011	11100000
0101101	00100000	00000011	01000000
0101110	01100101	00100001	00000000
0101111	01110011	00010001	00000000
0110000	00000001	00001000	00000000
0110001	00000000	00000000	00000000
0110010	00000001	00000001	00011110
0110011	00000010	00000001	00011000
0110100	00000100	00000001	00001000
0110101	00000001	00000001	00000110
0110110	01000000	00000001	00000000
0110111	01011111	00001001	00001010
0111000	00000000	00000000	00000000
0111001	00000001	00000001	00011110
0111010	00000010	00000001	00010000
0111011	00000100	00000001	00001000
0111100	00000001	00000000	00000110
0111101	01000000	00000000	00000000
0111110	00000001	00001000	00001010
0111111	00000000	00000000	00000000
	↑ MSB		↑ LSB

* For memory map systems the ROM words at the following addresses are changed as indicated:

Address	ROM Words		
0001011	00111011	00001010	11100000
0011101	00100000	00000010	01100000
0011110	00011001	00001010	11100000
0100010	10111011	00001011	11100000
	↑ MSB		↑ LSB

Table 2-20. Control Memory Contents *(continued)*

Address	ROM Words		
1000000	00000001	10000000	00011110
1000001	00000010	00000000	00010110
1000010	00000001	00000000	00000110
1000011	01000000	00000000	00000000
1000100	00000100	00000000	00001000
1000101	00000001	00000000	00000110
1000110	01000000	00000000	00000000
1000111	00000001	00001000	00001010
1001000	00000001	00000000	00011110
1001001	00000010	00000000	00011010
1001010	00000100	00000000	00001000
1001011	00000001	00000000	00000110
1001100	01000000	00000000	00000000
1001101	00000001	00001000	00001010
1001110	00000000	00000000	00000000
1001111	00000000	00000000	00000000
1010000	00000001	00000000	00011110
1010001	00001000	00000000	00011010
1010010	00010000	00000000	00000000
1010011	00000001	00001000	00000010
1010100	00000000	00000000	00000000
1010101	00000000	00000000	00000000
1010110	00000000	00000000	00000000
1010111	00000001	00000000	00011110
1011000	00000010	00000000	00010100
1011001	10000100	00001000	00001000
1011010	00000001	00000000	00000110
1011011	01000000	00000000	00000000

Control Memory Contents *(continued)*

Address	ROM Words		
1011100	01000001	00001011	11101010
1011101*	00000000	00000000	00000000
1011110*	00000000	00000000	00000000
1011111	00000000	00000000	00000000
	↑ MSB		↑ LSB

* For memory map systems, the ROM words at the following addresses are changed as indicated:

Address	ROM Words		
1011101	00100000	00000011	01100000
1011110	01000111	00001011	11100000
	↑ MSB		↑ LSB

2.4 PROGRAMMING

This section discusses general DCM programming. For programming characteristics peculiar to a given LAD, refer to the programming subsection in the section on that particular LAD.

The standard DCM device address is 070, but it can be changed within the range 00 to 077.

2.4.1 DCM Instructions

Table 2-21 explains the DCM instructions, using 070 as the DCM device address.

Table 2-21. DCM Instructions

Octal	Mnemonic	Definition	Decription
100070	EXC 070	Initialize	Programmed system reset that clears the multiplexor and all associated LADs
100170	EXC 0170	Clear control logic	Aborts current sequence and returns multiplexor to scanning mode, but does not disable interrupts
100270	EXC 0270	Enable DCM interrupts	Enables the six DCM interrupts
100470	EXC 0470	Disable DCM interrupts	Disables the six DCM interrupts
100570	EXC 0570	Request control (write)	Permits computer to request use of multiplexor bus for LAD setup; the multiplexor generates a control interrupt upon completion of current operation
100670	EXC 0670	Request control (read)	Permits computer to request use of multiplexor bus for reading LAD status; the multiplexor generates a control interrupt upon completion of current operation
100244	EXC 0244	Enable interrupts (common)	DCM also responds to this general system interrupt-enabling instruction
100444	EXC 0444	Disable interrupts (common)	DCM also responds to this general system interrupt-disabling instruction
102070	IME 070	Input data from DCM to memory	Transfers a 16-bit character from the DCM, where the six least-significant bits come from the scan counter and the remainder from the interface buffer
102170	INA 070	Input data from DCM to A register	
102270	INB 070	Input data from DCM to B register	
102570	CIA 070	Clear A register and input data from DCM	Clears destination register, then transfers a 16-bit character from the DCM, where the six least-significant bits come from the scan counter and the remainder from the interface buffer
102670	CIB 070	Clear B register and input data from DCM	
103170	OAR 070	Output data from A register to DCM	Transfers a 16-bit character to the DCM, where the six least-significant bits go to the scan counter and the remainder to the interface buffer
103270	OBR 070	Output data from B register to DCM	
103370	OME 070	Output data from memory to DCM	

2.4.2 DCM Interrupts

The DCM generates its own interrupt requests to the computer and automatically disables them when the computer executes a jump-and-mark instruction in response to any interrupt request. The interrupts can be enabled again by the DCM enable-interrupt instruction (0100270) or by the common interrupt-enabling instruction used on PIMs (0100244). The interrupts can also be disabled by the DCM disable-interrupt instruction (0100470) or by the PIM common interrupt-disabling instruction (0100444).

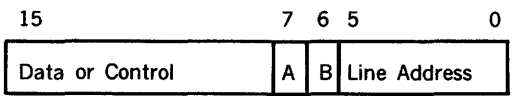
The six DCM interrupts and their standard addresses are:

Input byte-count-zero interrupt	060
Output byte-count zero interrupt	062
Line-error interrupt	064
Status-change interrupt	066
Control-character-detected interrupt	070
Control interrupt	072

This block of addresses can, however, be moved in the range up to 0777.

The program should not keep the DCM interrupts disabled longer than necessary, since the DCM does not scan communications lines during this time. Furthermore, if the software does not provide for the proper responses to DCM interrupts (section 2.4.2.1 through 2.4.2.6), the multiplexor will hang up.

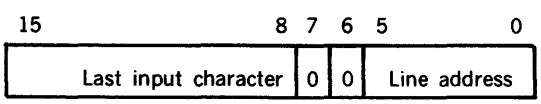
The proper response for all interrupts (except the control interrupt) is a data-transfer-in followed by a data-transfer-out (DTO). Upon receipt of the DTO, scanning resumes. The format for the DTO is:



The following sections explain the character input format for the various interrupts.

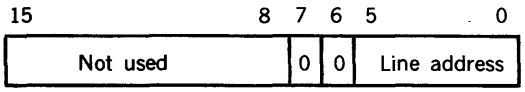
2.4.2.1 Input Byte-Count Zero

This interrupt is generated when the block-length counter decrements to zero due to a DMA data-transfer-in sequence. The program responds with a data-transfer-in instruction. The character format is:



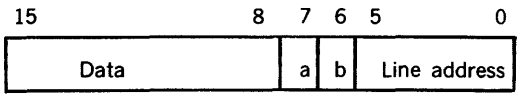
2.4.2.2 Output Byte-Count Zero

This interrupt is generated when the block-length counter decrements to zero due to a DMA data-transfer-out sequence. The program responds with a data-transfer-in instruction. The character format is:



2.4.2.3 Line Error

This interrupt is generated when a line inputs a character with incorrect parity or some other line error. The program responds with a data-transfer-in instruction. The character format is:

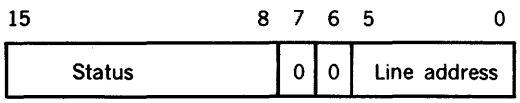


where a and b indicate the type of error. Since the significance of these bits depends on the particular LAD involved, they are discussed in the LAD sections.

Note: After the line error interrupt is serviced, the DCM will transfer the data to the input buffer.

2.4.2.4 Status Change

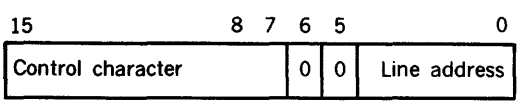
This interrupt is generated when a LAD reports an error or a special event. The program responds with a data-transfer-in instruction. The character format is:



Since the significance of the status bits depends on the particular LAD involved, they are discussed in the LAD sections.

2.4.2.5 Control-Character Detected

The multiplexor checks each input character to see if it is a control character. It does this by accessing (via DMA) the control characters for a specific line from the control table stored in the computer memory. After accessing the control characters (two for each line), the multiplexor checks the input character for a match. If the received character matches one of the control characters, a control-character-detected interrupt is generated. The computer program responds with a data-transfer-in instruction. The character format is:

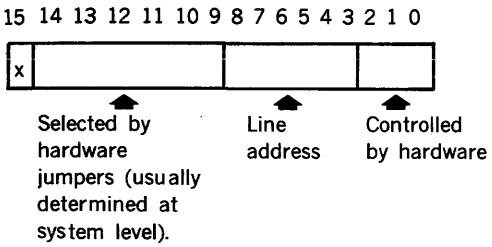


2.4.2.6 Control

This interrupt is explained in section 2.4.4.4 on DCM programming sequences.

2.4.3 DCM DMA Control Table

The DCM works from line control tables (figure 2-13) stored in computer memory for all DMA operations. Each line control table uses eight memory locations. Thus, for a maximum 64-line system, 512 memory locations are allocated for the DCM. The format of the line control table address is:



Note: In memory map systems, the control table is located in key zero (0000).

The byte-transfer flag in the control table is hardware-set and determines which byte is to be transferred. The initial

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
077000	f	s m b	x	x	INPUT BLOCK LENGTH (IN BYTES) *											
077001	x	INPUT BUFFER ADDRESS														
077002	CONTROL CHARACTER 2								CONTROL CHARACTER 1							
077003	y	y	y	y	y	y	y	y	y	y	y	y	MEMORY MAP KEY			
077004	f	x	x	x	OUTPUT BLOCK LENGTH (IN BYTES) *											
077005	x	OUTPUT BUFFER ADDRESS														
077006	LINE-CONTROL BYTE								a	b	y	y	y	y	y	y
077007	y	y	y	y	y	y	y	y	y	y	y	y	MEMORY MAP KEY			

*1's COMPLEMENT OF BLOCK LENGTH. MAXIMUM BLOCK LENGTH = 4095 BYTES (BITS 0 THRU 11 ALL ZEROS). SET INPUT BLOCK LENGTH TO ZERO (ALL ONES) WHEN NOT USING THE LINE IN THE RECEIVE MODE.

DATA BUFFER FORMAT

15	8 7	0
FIRST CHARACTER	SECOND CHARACTER	
THIRD CHARACTER	FOURTH CHARACTER	

WHERE:
- x = UNUSED BITS - THESE BITS WILL BE WRITTEN INTO BY THE DCM
- y = UNUSED BITS - NOT MODIFIED BY DCM
- smb = SET EIGHTH BIT OF INPUT CHARACTER (MOST SIGNIFICANT) TO ONE
- a b = CODE FOR LINE CONTROL BYTE (SEE SEC. 4.8.1.1)
- f = FLAG SET BY HARDWARE TO DETERMINE WHICH BYTE TO TRANSFER (NORMALLY SET TO ONE INITIALLY BY SOFTWARE)
  - 0 = LOWER 8 BITS
  - 1 = UPPER 8 BITS (WHEN THE MOST SIGNIFICANT BYTE IS LOADED, THE LEAST SIGNIFICANT BYTE IS CLEARED)

NOTE: ADDRESSES SHOWN ARE FOR LINE ZERO WITH THE SIX MOST-SIGNIFICANT BITS ALL STRAPPED ON

VTII-1759A

Figure 2-13. Line Control Table

setting is made by software to indicate the first byte. When the more-significant byte is transferred, the less-significant byte is cleared.

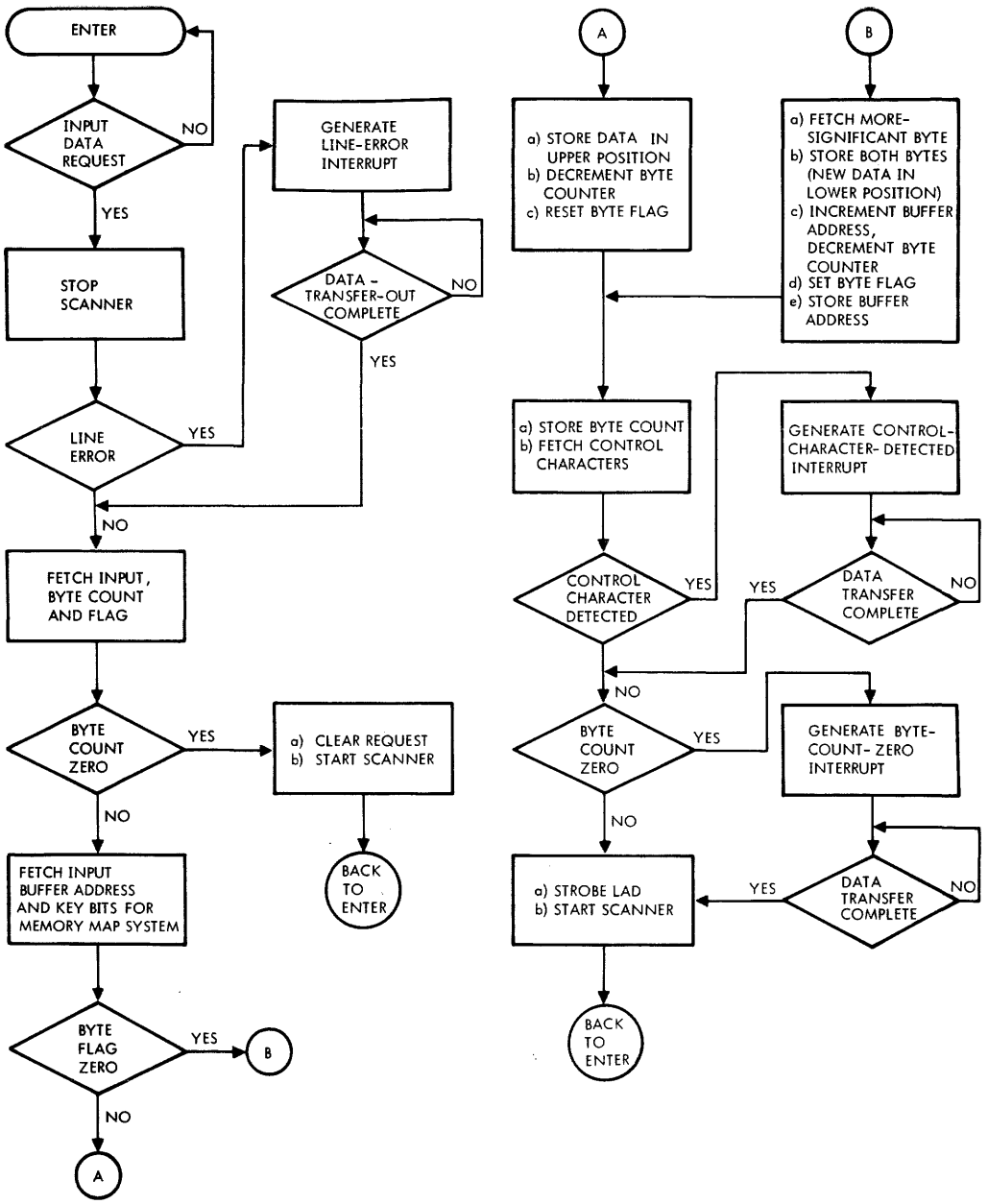
The smb bit, when set, sets the most-significant bit of the input data.

2.4.4 Programming Sequences

This section discusses DCM input, output, status-request, and control sequences.

2.4.4.1 Input Sequence

Figure 2-14 illustrates the sequence the multiplexor follows upon detection of an input data request from a line. If there is no line error, the multiplexor fetches the input block-length count. If the count is not zero, the input buffer address is loaded into the multiplexor where it is used to store the data in the correct location. The byte transfer flag indicates whether a character has already been stored in the less-significant portion of the buffer word. If the flag is set, the input character is loaded in the more-significant byte position the less significant portion is cleared, and the



VTII-1760B

Figure 2-14. DCM Input Sequence

byte flag is reset. If the flag is not set, it is now set and the characters stored in memory with the new character in the less-significant byte position. After the character is stored, the byte count is decremented and restored in memory.

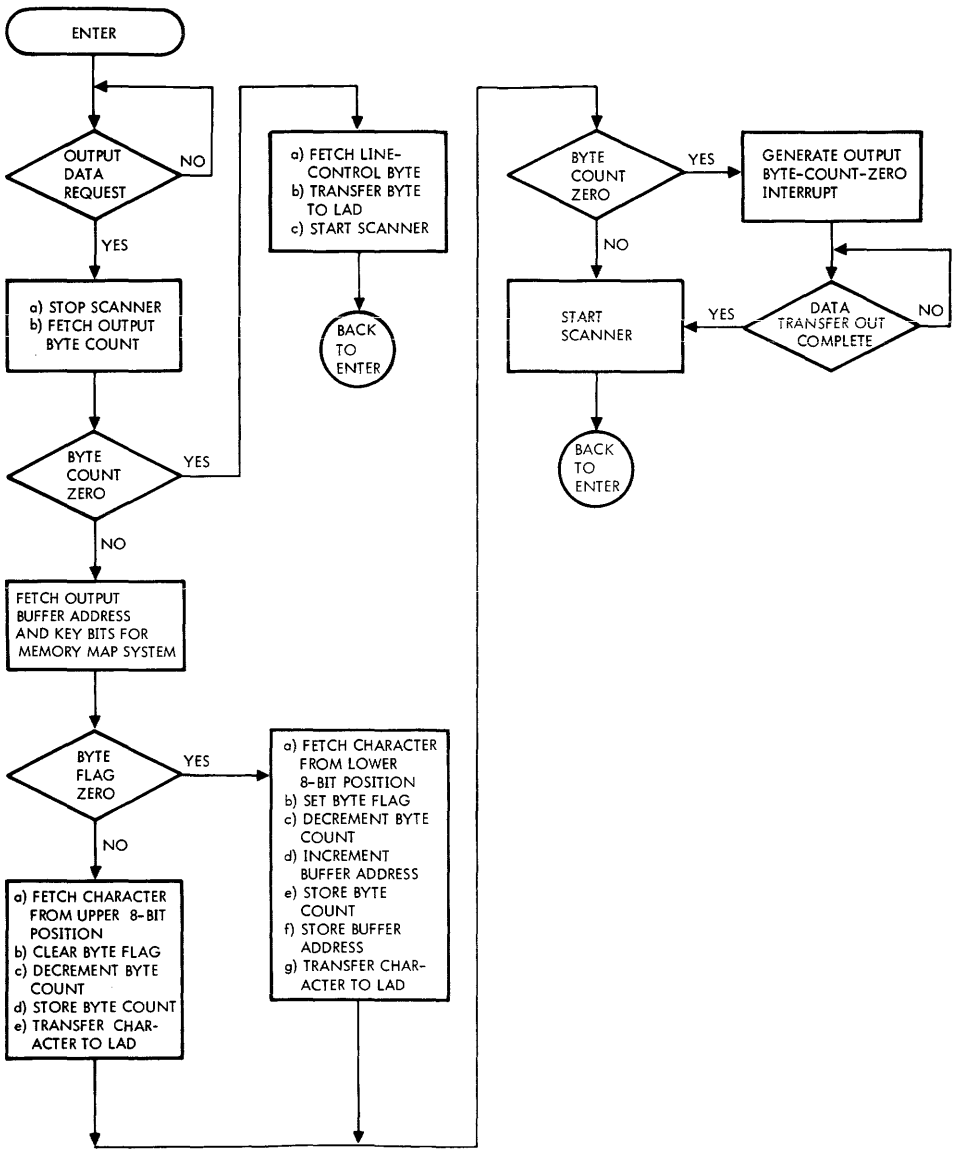
If the byte flag was set, the input buffer address word is incremented and the new address stored in memory. After the input character is stored in the input buffer and all the housekeeping completed, the two control characters are input to the multiplexor and compared to the input character. If there is a match, a control-character-detected interrupt is generated. Upon completion of the control-character check, the multiplexor checks to see if the byte count went to zero during this input sequence. If the byte count is zero, an input byte-count-zero interrupt is generated. The byte-count check completes the input sequence, and scanning is resumed.

If there is an input request from a line whose byte count is already zero, the multiplexor clears the request and continues scanning (unless a line error is present, in which case the line error is reported).

Note: The DCM clears the less-significant byte whenever it loads data in the more-significant byte of a buffer word.

2.4.4.2 Output Sequence

Figure 2-15 illustrates the output data-request sequence. The multiplexor stops scanning when an output data-request is detected and fetches the output byte-count word.



VTII-1761A

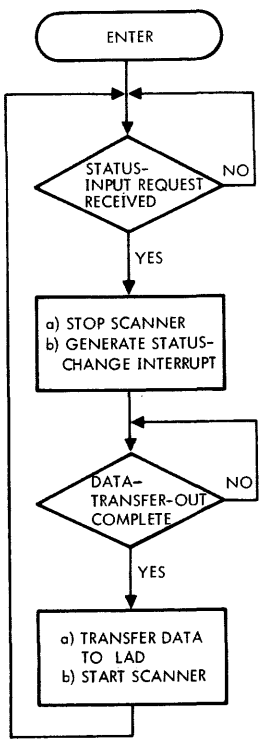
Figure 2-15. DCM Output Sequence

If the byte count is zero, the line control byte is loaded into the multiplexor and transferred to the appropriate line. The a and b bits determine the type of information transferred to the line.

If the byte count is greater than zero, the output buffer address is fetched and the byte-transfer flag tested. If the flag is set, a data character is transferred from the upper eight-bit position of the location indicated by the buffer address, the byte flag reset, the byte count decremented, and the count reloaded in the control table. If the byte flag is not set, the data character is transferred from the lower eight-bit position, the byte flag set, the byte count decremented, the buffer address incremented, and the byte-count and buffer address stored back in the control table.

If the byte count goes to zero during an output sequence, an output byte-count-zero interrupt is generated. After the byte count is tested, the multiplexor continues line scanning.

Note: If a line is left in transmission mode with the byte count at zero, the multiplexor continues to output the control byte each time the line is scanned. This reduces throughput significantly.



VTII-1762

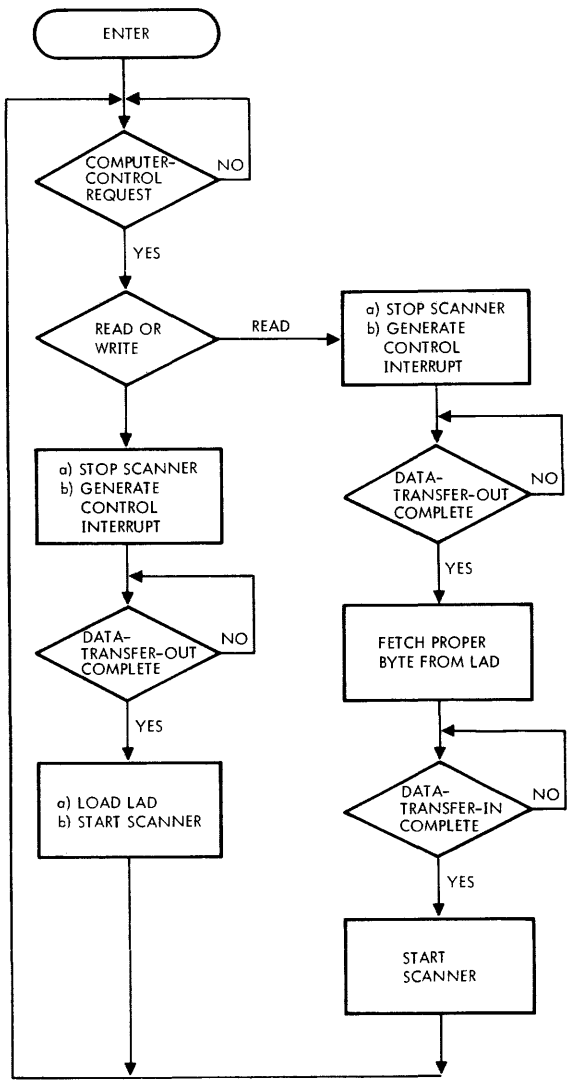
Figure 2-16. DCM Status-Request Sequence

2.4.4.3 Status-Request Sequence

Figure 2-16 shows the sequence the DCM follows when it detects a status request from a communication line. The multiplexor stops line scanning, generates a status-change interrupt, waits for a transfer to the LAD (usually control information), completes the transfer, then resumes scanning.

2.4.4.4 Computer-Control Sequence

Figure 2-17 illustrates the computer-control sequence. The multiplexor always checks for a computer-control request before checking for a service request from a communication line. The computer can thus gain control at the end of any multiplexor sequence before another line is serviced. Whenever a computer-control request (read or write) is detected, the multiplexor stops scanning and generates a control interrupt. If the request is to write, the multiplexor



VTII-1763

Figure 2-17. DCM Computer-Control Sequence

waits for the data-transfer-out and then loads the indicated line and resumes scanning. For reading, the multiplexor waits for the data-transfer-out that identifies the line and the data to be transferred, then fetches the appropriate data and waits for the data-transfer-in. At the end of the sequence, scanning resumes.

Since the multiplexor stops scanning during a control sequence, control operations should be as fast as possible to prevent line overruns and consequent data loss.

Sample control sequences are given in this manual in the programming subsections of each LAD section.

2.5 MNEMONICS

MU Board One

Mnemonic	Definition
A–	Decoded along with B– to indicate the type of information being transferred to the LAD.
AC +	Source for A– from the interface buffer
AC10 + thru AC16 +	Parallel inputs to the control memory address counter
AD +	A input to the interface buffer (EB07 +)
B–	See A–
BC +	Source of B– from the interface buffer
BD +	B input to the interface buffer (EB06 +)
BFA +	Byte flag (buffered) from MU board 2
BR0– thru BR5–	Bit-rate sources for the LADs: Actual frequency determined by jumpers
CACCL +	Clock for the control memory address counter
CAC10 + thru CAC12 +	Three least-significant parallel inputs to the control memory address counter (before normal mode/test mode selection logic)
CACL–	Parallel-load signal for the control memory address counter
CAE +	Control memory address counter clock enabler
CCLG–	Clear-control-logic signal from MU board 2
CCM +	Control-character-match signal from MU board 2
CLK–	Clock for the LADs (nominally 614,400 Hz) (LADs place requests on line on the positive going edge of this clock)
CLKB + CLKBF +	4.9152 MHz clock (square wave)
CLKC +	2.4576 MHz clock (square wave)
CLKD +	1.2288 MHz clock (square wave)
CLKE +	614,400 Hz clock (square wave)
CLKF +	307,200 Hz clock (square wave)
CLKI +	Fundamental clock frequency 9.8304 MHz
CMA00 + thru CMA06 +	Outputs of the control memory address counter
CMB00 + thru CMB23 +	Outputs of the control memory
CME00–	Enable control memory addresses 000 to 037
CME32–	Enable control memory addresses 040 to 077
CME64	Enable control memory addresses 0100 to 0137
CPURA +	CPU read control request from MU board 2
CPURS +	CPUR4 + synchronized with CLKB +
CPUWA +	CPU write control request from MU board 2
CPUWS +	CPUWA + synchronized with CLKB +
CZ +	Byte count zero signal from MU board 2
DTIA +	Data transfer in signal from MU board 2
DTIS +	DTIA + synchronized with CLKB +
DTOA +	Data transfer out signal from MU board 2
DTOS +	DTOA + synchronized with CLKB +

Mnemonic	Definition
EAB +	Enable inputs to the interface buffer for A– and B–
EB00 + thru EB15 +	Internal E-bus signals from MU board 2
ECPR +	Enable CPU request
EDR1–	Enable memory address out for DMA requests for data (from MU board 2)
EIBC–	Enable interface buffer clock
ELAR +	Enable line adapter requests
ELSB–	Enable least-significant byte

The following signals are decoded from the A field of the control memory's 24-bit word:

Mnemonic	Definition
FA01–	Stop scanner
FA03–	Start strobe
FA04–	Enable write
FA05–	Disable write
FA06–	Clock the byte count register and the DMA 16 bit register (sent to MU board 2)

The following signals are decoded from the B field of the control memory's 24-bit word:

FB3–	Fetch key bits
FB4–	Fetch/store data
FB5–	Fetch first byte (input)
FB6–	Fetch line control byte
FSDRY +	DRYX + or DRYF + (from MU board 2)
FSTRPA +	High speed DMA or normal DMA trap (from MU board 2)
IB0 + thru IB7 +	Outputs of the interface buffer
IB0 + E thru IB7 + E	Buffered IBx signals sent to MU board 2
IDR–	Input data request from LADs

Mnemonic	Definition
IOK1-I thru IOK4-I	The 4 key-bit lines (V73-I/O)
LERR–	Line-error from LADS
LSC–	Load scan counter
L0003–	LAD enabler for lines 0-3 and 32-35
L0407–	LAD enabler for lines 4-7 and 36-39
L0811–	LAD enabler for lines 8-11 and 40-43
L1215–	LAD enabler for lines 12-15 and 44-47
L1619–	LAD enabler for lines 16-19 and 48-51
L2023–	LAD enabler for lines 20-23 and 52-55
L2427–	LAD enabler for lines 24-27 and 56-59
L2831–	LAD enabler for lines 28-31 and 60-63
MB0 + thru MB7 +	Buffered and latched multiplexor bus signals (sent to MU board 2)
MB0– thru MB7–	Multiplexor bus lines
OBRIC +	Odd-bit-rate counter 1 clock
OBR2C +	Odd-bit-rate counter 2 clock
OBR1 +	Odd-bit-rate counter 1 output
OBR2 +	Odd-bit-rate counter 2 output
ODR–	Output data request from the LADs
SCB0 + thru SCB5A +	Scan counter outputs
SCKPI +	Preset term for the scan counter-clock flip-flop at the end of each scan-counter clock
SCLKE +	Scan counter clock flip-flop to enable the clock to start and stop
SIR–	Status input request from LADs
SRST–	System reset term sent to LADs
SSCN +	Stop-scan flip-flop to inhibit CPU or LAD requests to the DCM
STR–	LAD strobe complete

Mnemonic	Definition
STRBE	Strobe to LADs
SYRTA–	System-reset from MU board 2
TEST–	Test (when grounded, allows control memory to be addressed by the test switches +S0 through S6 or via J1)
TOI	Trap or interrupt from MU board 2
TOIC +	Trap or interrupt complete
TST +	Test (allows a free running clock to the control memory address counter)
WRITE–	Indicates direction of data transfer on the multiplexor bus

MU Board Two

Mnemonic	Definition
AI–	Decoded with BI– to determine type of line error (from LADs)
BC00 + thru BC11 +	Outputs of the byte-count register/ counter
BF +	Byte flag
BFA +	Byte flag buffered and sent to MU board 1
BI–	See AI–
CCLG–	Clear control logic
CCM–	Match control-character (sent to MU board 1)
CLKB +	4.9152-MHz square-wave from MU board 1
CLKCS +	Clock to decrement the byte count and increment the buffer address
CLKE +	614,400-Hz square-wave from MU board 1
CMB00 thru CMB23	Control memory output from MU board 1
CPURA +	CPU read control request to MU board 1
CPUWA +	CPU write control request to MU board 1

Mnemonic	Definition
CTA09 + thru CTA15 +	Control table (LCB) hardware selectable address bits
CZ +	Byte-count-zero indication to MU board 1
DRYF-I	High-speed DMA data-ready signal
DRYX-I	E-bus data-ready signal
DR00– thru DR15–	Outputs of the DMA 16-bit register/ counter
DTI +	Data transfer in
DTIA +	Buffered DTI to MU board 1
DTO +	Data transfer out
DTOA +	Buffered DTO to MU board 1
DVCD +	Device code (address)
DVCFR +	Device address gated with FRYX +
DVC44 +	Device address 044 for common interrupt enable/disable
D44FR–	Device address 044 gated with FRYX +
EAB0 +	Enable AI and BI onto E-bus
EBC + thru EBCA +	Enable output of byte-count register/ counter onto E-bus
EB00-I thru EB15-I	16-bit bidirectional E-bus signals
ECTA + thru ECTAA +	Enable control-table address onto E-bus
EDR thru EDRA +	Enable output of 16-bit DMA register/ counter onto E-bus
EDRI–	Enable key bits during the address portion of a DMA cycle for data
EINT–	Enable interrupts
ELMBA	Enable multiplexor-bus signals onto less-significant portion of E-bus and interface buffer signals (IBx + E) onto the more-significant portion

Mnemonic	Definition
ELSMB + thru EMSMB +	Enable multiplexor-bus signals onto more-significant portion of E-bus

The following signals are decoded from the A field of the control memory's 24-bit word:

Mnemonic	Definition
FA06–	Clock byte count register and DMA 16-bit register (decoded on MU board 1)
FA12–	Line-error interrupt
FA15–	Control interrupt
FA17–	Start any interrupt

The following signals are decoded from the B field of the control memory's 24-bit word:

FB0–	Fetch/store byte count
FB1–	Fetch/store buffer address
FB2–	Fetch control character
FB4–	Fetch/store data
FB5–	Fetch first byte (input)
FB6–	Fetch line control byte
FB7–	Start any trap
FRYF-I	Function-ready for high-speed DMA
FRYX-I	Function-ready for the E-bus
FSDRY +	OR of DRYF and DRYX
FSTPD +	Address phase of DMA cycle
FSTRP +	Data phase of DMA cycle
FTEN–	Enable fast trap (high-speed DMA)
FTRAP +	Fast trap (data phase)
FTRPD	Fast trap (address phase)
FTRQ +	Fast-trap request
IB0 + E thru IB7 + E	Interface-buffer signals from MU board 1

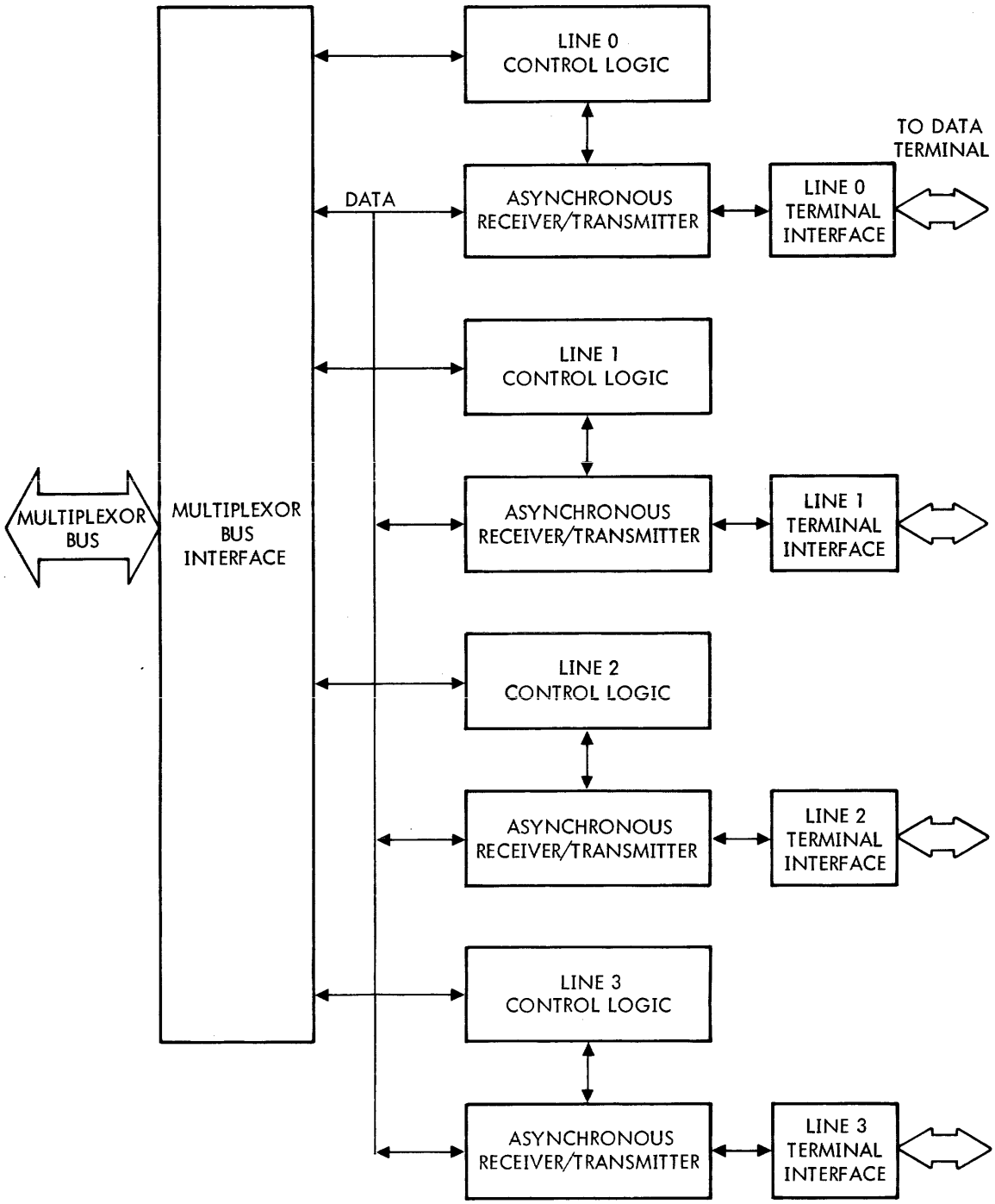
The following signals are decoded from the B field of the control memory's 24-bit word during an input trap sequence:

Mnemonic	Definition
IFB0	Store byte-count
IFB1–	Store buffer-address
IFB4–	Store data
INT +	Interrupt
INTA + thru INTB +	Gate interrupt address onto E-bus
INTR +	Interrupt request
ITA04 + thru ITA08 +	Interrupt address-bits selected by hardware jumpers
IUAF-I	Interrupt acknowledgment for high-speed DMA
IUAX-I	E-bus interrupt-acknowledgment
IUCF-I	High-speed DMA I/O clock
IUCX-I	E-bus I/O clock
IUJX-I	Jump-and-mark (E-bus signal) executed in response to interrupt
IURX-I	E-bus interrupt-request line
LBC–	Load byte-count register/counter
LDR–	Load DMA 16-bit register/counter
LSC–	Load scan-counter (from MU board 1)
MB0 + thru MB7 +	Multiplexor bus signals from MU board 1
PRMA + I PRMB + I PRMC + I	Parallel priority inputs for high-speed DMA
PRMF	High-speed DMA priority
PRMX-I	E-bus priority-chain input
PRNF + I	High-speed DMA priority output
PRNX-I	E-bus priority-chain output

Mnemonic	Definition
SCB0– thru SCB5–	Scan-counter outputs from MU board 1
STRAP +	Normal-DMA trap (data portion)
STRPD +	Normal-DMA trap (address portion)
SYRT-I	E-bus system-reset
SYRTA–	DCM system-reset
TOI–	Trap or interrupt complete (to MU board 1)
TPIF-I	High-speed DMA trap-in request
TPIX-I	Normal-I/O DMA trap-in request
TPOF-I	High-speed DMA trap-out request
TPOX-I	Normal-I/O DMA trap-out request
X0– thru X6–	Function decoded (EB06, EB07, EB08)

SECTION 3
DIRECT-CONNECTION LINE-ADAPTER

This LAD (figure 3-1) provides three direct-connection interfaces for the DCM. It can handle four terminals and its interface circuits can be RS232, current-loop, or relay model (table 3-2).



VTII-1784

Figure 3-1. Direct Connection LAD Simplified Block Diagram

The LAD interfaces with four full-duplex asynchronous data communication lines at a maximum data rate of 9,600 bits per second. It provides character buffering for both input and output, automatically adds start/stop bits and disassembles characters, assembles characters and strips off the start/stop bits, generates output parity (if any), checks incoming data for correct parity (if any), and checks for break characters or framing errors. The parity can be odd, even, or not used; the character length can be 5, 6, 7, or 8 bits; and there can be one or two stop bits. Each of the lines can be assigned any one of six hardwired bit rates.

This LAD is not intended for interfaces to datasets (modem), but is designed for local interfaces with serial devices (data terminals, Teletypes, CRTs, line printers, etc.). It can be operated as a transmitter and/or receiver, or as a transmitter with control-line input.

Table 3-1 gives the general specifications of the direct-connection LAD. Table 3-2 gives a comparison of the three models. Table 3-3 gives cable information.

Table 3-1. Direct-Connection LAD Specifications

Number of lines/board	Four
Transmission type	Serial asynchronous
Line terminal interface	RS232, current-loop, or relay (table 3-2)
Character length	5, 6, 7, or 8 (hardware selectable)
Number of stop bits	1 or 2 (hardware selectable)
Parity generation and checking	Odd, even, or none (hardware selectable)
Buffering	Each line character buffered on input and output
Maximum bit rate	Dependent on terminal interface (table 3-2)
Error reporting	Parity errors, line breaks (or framing errors), and overflow errors
Package	One PC board with approximately 75 ICs
Power requirements	Dependent on terminal interface (table 3-2)
Environment	Temperature 0 to 50 degrees C, relative humidity 0 to 90 percent (without condensation)
Interconnection	Plugs into DCM backplane, interface with up to four terminals via two 44-pin connectors
Cable length	Dependent on terminal interface (table 3-2)

Table 3-2. Direct-Connection LAD Line-Interface Comparison

Interface	RS232	Current-Loop	Relay
Bit Rate	45 to 9600 bps, depending on cable length (table 3-3)	45 to 9600 bps, depending on cable length (table 3-3)	45 to 300 bps, typically 75 or 110 bps (note that relays do not operate reliably above 300 bps)
Power requirements	+5V dc at 1A +12V dc at 100mA −12V dc at 100mA	+5V dc at 1A −12V dc at 50mA (exclusive of user-supplied loop-current battery power)	+5V dc at 1A −12V dc at 50mA (exclusive of user-supplied loop-current battery power)
Cable length	20 feet (6 meters) standard, 50 feet (15 meters) maximum	20 feet (6 meters) standard, more than one mile (1.6 km) maximum, depending on bit rate (table 3-3), e.g., 1,000 feet (300 meters) at 9600 bps	20 feet (6 meters) standard, more than one mile (1.6 km) maximum, depending on bit rate (table 3-3), e.g., 10,000 feet (3,000 meters) at 300 bps

Table 3-3. Cable Length vs. BPS Rate

Cable Length	Maximum Rate in BPS
Up to 1,000 feet (300 meters)	10,000
1,000 to 2,000 feet (300 to 600 meters)	4,800
2,000 to 3,500 feet (600 meters to 1 km.)	1,800
3,500 to 5,000 feet (1 to 1.5 km.)	900
5,000 to 10,000 feet (1.5 to 3 km.)	300

3.1 INSTALLATION

The LAD has been inspected and packed to ensure its arrival in good working order. To prevent damage, take reasonable care during unpacking and handling. Check the shipping list to ensure that all equipment has been received. Immediately after unpacking, inspect the equipment for shipping damage. If any is found:

a. Notify the transportation company

b. Notify Varian Data Machines

c. Save all packing material

3.1.1 Physical Description

The direct-connection LAD is on one printed-circuit board (figure 3-2) mounted in a specially-wired expansion chassis (figures 1-2 and 1-3).

3.1.2 Interfaces

The direct-connection LAD interfaces with the MU over the multiplexor bus (figures 1-1 and 2-1), and with the terminals directly over the line interfaces (section 3.1.2.2).

3.1.2.1 Multiplexor Bus

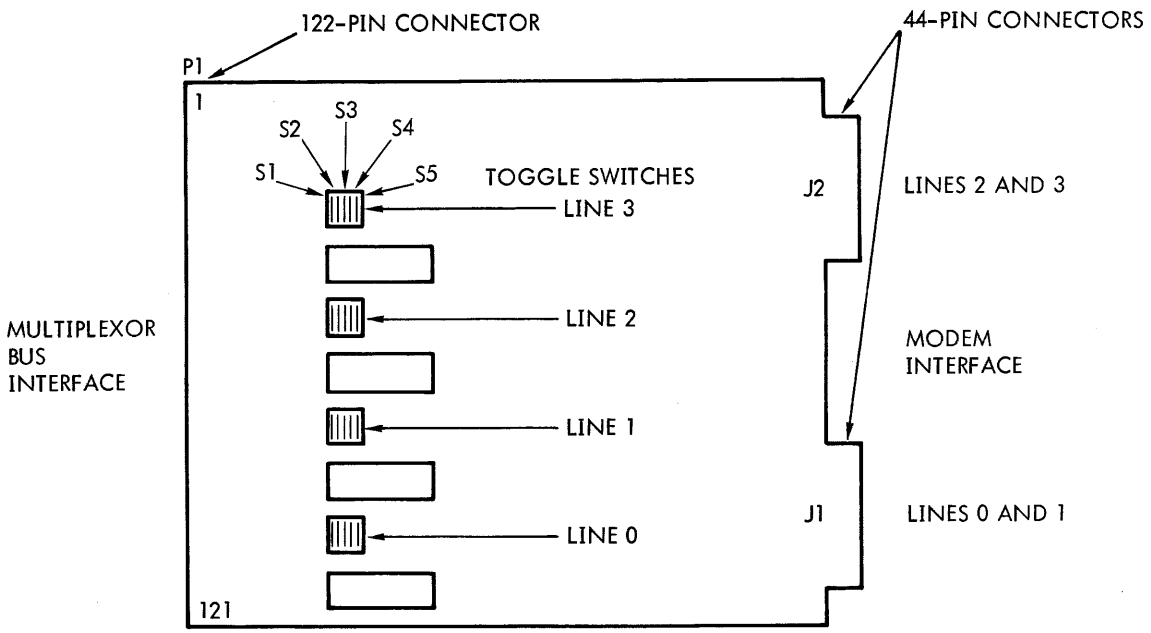
The 38-line multiplexor bus connects the LAD with the MU. It is described in section 2.1.2.2.

3.1.2.2 Line Interface

There is a line interface for each version of the direct-connection LAD:

- RS232
- Current-loop
- Relay

RS232 Interface: This interface drives peripherals with compatible interfaces. It also provides a DTL output that has an open-collector interface capable of sinking 70 mA at



VTII-1785

Figure 3-2. Direct-Connection LAD

0.5V maximum VCEsat and a maximum collector voltage of 7.0V. DTL output data are ground-true (i.e., GND = 1 and +VCC = 0). Table 3-4 shows the pin connections for the RS232 interface, and figure 3-3 illustrates the DTL interface.

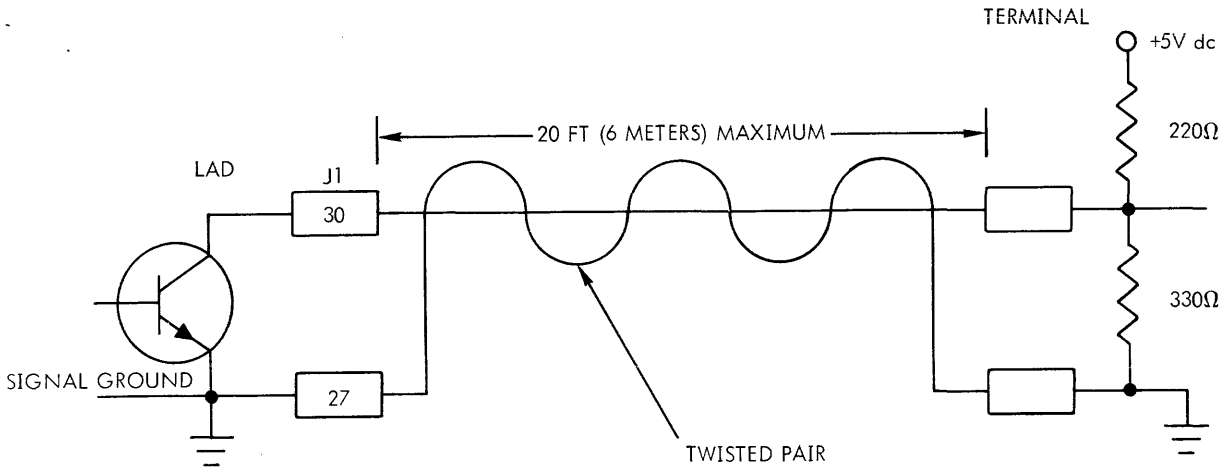
Current-loop interface: This interface provides a solid-state current-loop interface. Table 3-5 contains the specifications of the interface, table 3-6 gives the pin connections, and figure 3-4 shows how to connect the required user-supplied battery. The LAD has a 2K, 2-watt series resistor in the driver, and another in the receiver. These limit the loop current, but can be shorted out if an external current-limiting resistor is desired, or they can be changed to alter the limitation on the loop current so long as the total power dissipation does not exceed 1W per resistor.

Relay interface: This interface uses a relay. Table 3-7 contains the specifications of the interface and table 3-8 gives the pin connections. A line battery must be supplied by the user. The relay driver provides a normally-closed contact output (closed for logical 1), and a normally-open one (open for logical 1) that can be used as illustrated in figure 3-5. Each output, as well as the input coil, has resistors like those in the current-loop interface and subject to the same manipulations and limitations. The input coil also has diodes so placed in the circuitry that the input relay is energized by current in one direction only. If loop currents higher than 10 mA are desired in the LAD receiver circuit, remove the diode and the resistor in parallel with the receiver relay coil and replace them with the appropriate shunt resistance.

Table 3-4. RS232 Interface Pin Connections

Line	Signal	Pin	Definition
0	SDDS0	J1-004	RS232C circuit BA
0	RDDS0	J1-024	RS232C circuit BB
0	GND	J1-027	RS232C circuit AB
0	S- 12V0	J1-043	- 12V through 1K resistor
0	S+12V0	J1-044	+12V through 1K resistor
0	SD0-	J1-030	DTL data output
1	SDDS1	J1-002	RS232C circuit BA
1	RDDS1	J1-022	RS232C circuit BB
1	GND	J1-027	RS232C circuit AB
1	S- 12V1	J1-017	- 12V through 1K resistor
1	S+12V1	J1-020	+12V through 1K resistor
1	SD1-	J1-014	DTL data output
2	SDDS2	J2-004	RS232C circuit BA
2	RDDS2	J2-024	RS232C circuit BB
2	GND	J2-027	RS232C circuit AB
2	S- 12V2	J2-043	- 12V through 1K resistor
2	S+12V2	J2-044	+12V through 1K resistor
2	SD2-	J2-030	DTL data output
3	SDDS3	J2-002	RS232C circuit BA
3	RDDS3	J3-022	RS232C circuit BB
3	GND	J2-027	RS232C circuit AB
3	S- 12V3	J2-017	- 12V through 1K resistor
3	S+12V3	J2-020	+12V through 1K resistor
3	SD3-	J2-014	DTL data output

DIRECT-CONNECTION LINE-ADAPTER



VTII-1920

Figure 3-3. DTL Interface

Table 3-5. Current-Loop Interface Specifications

Driver Specifications	
Collector emitter voltage	150V dc maximum
Collector current - continuous	500 mA maximum
Total dissipation at 25 degrees C	1W maximum
Receiver Specification	
Input forward voltage drop	1.5V dc maximum
Input forward continuous current	60 mA maximum
Isolation	
Common-mode protection	500V dc
Common-mode resistance	10 megohms
Logic 0 = No loop current	
Logic 1 = Loop current (20 mA minimum)	

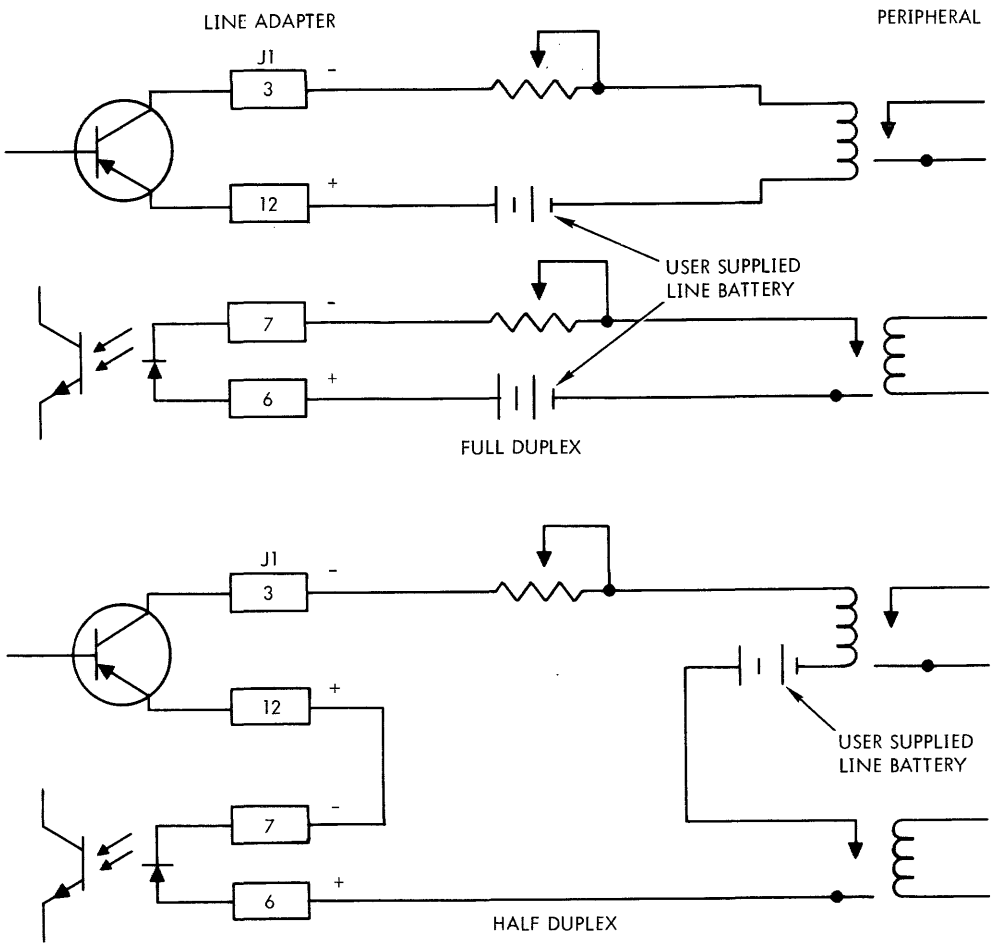
Table 3-6. Current-Loop Interface Pin Connections

Line	Signal	Pin	Definition
0	OUT0-	J1-029	Negative terminal of the output driver
0	OUT0+	J1-040	Positive terminal of the output driver
0	IN0-	J1-037	Negative terminal of the input receiver
0	IN0+	J1-034	Positive terminal of the input receiver
0	GND	J1-027	Ground
0	CS10	J1-039	Negative current source for testing only
0	CS20	J1-041	Negative current source for testing only
1	OUT1-	J1-003	Negative terminal of the output driver
1	OUT1+	J1-012	Positive terminal of the output driver
1	IN1-	J1-007	Negative terminal of the input receiver
1	IN1+	J1-006	Positive terminal of the input receiver

(continued)

Table 3-6. Current-Loop Interface Pin Connections (continued)

Line	Signal	Pin	Definition
1	GND	J1-027	Ground
1	CS11	J1-011	Negative current source for testing only
1	CS21	J1-015	Negative current source for testing only
2	OUT2–	J2-029	Negative terminal of the output driver
2	OUT2 +	J2-040	Positive terminal of the output driver
2	IN2–	J2-037	Negative terminal of the input receiver
2	IN2 +	J2-034	Positive terminal of the input receiver
2	GND	J2-027	Ground
2	CS12	J2-039	Negative current source for testing only
2	CS22	J2-041	Negative current source for testing only
3	OUT3–	J2-003	Negative terminal of the output driver
3	OUT3 +	J2-012	Positive terminal of the output driver
3	IN2–	J2-007	Negative terminal of the input receiver
3	IN2 +	J2-006	Positive terminal of the input receiver
3	GND	J2-027	Ground
3	CS13	J2-011	Negative current source for testing only
3	CS23	J2-015	Negative current source for testing only



VTII-1921

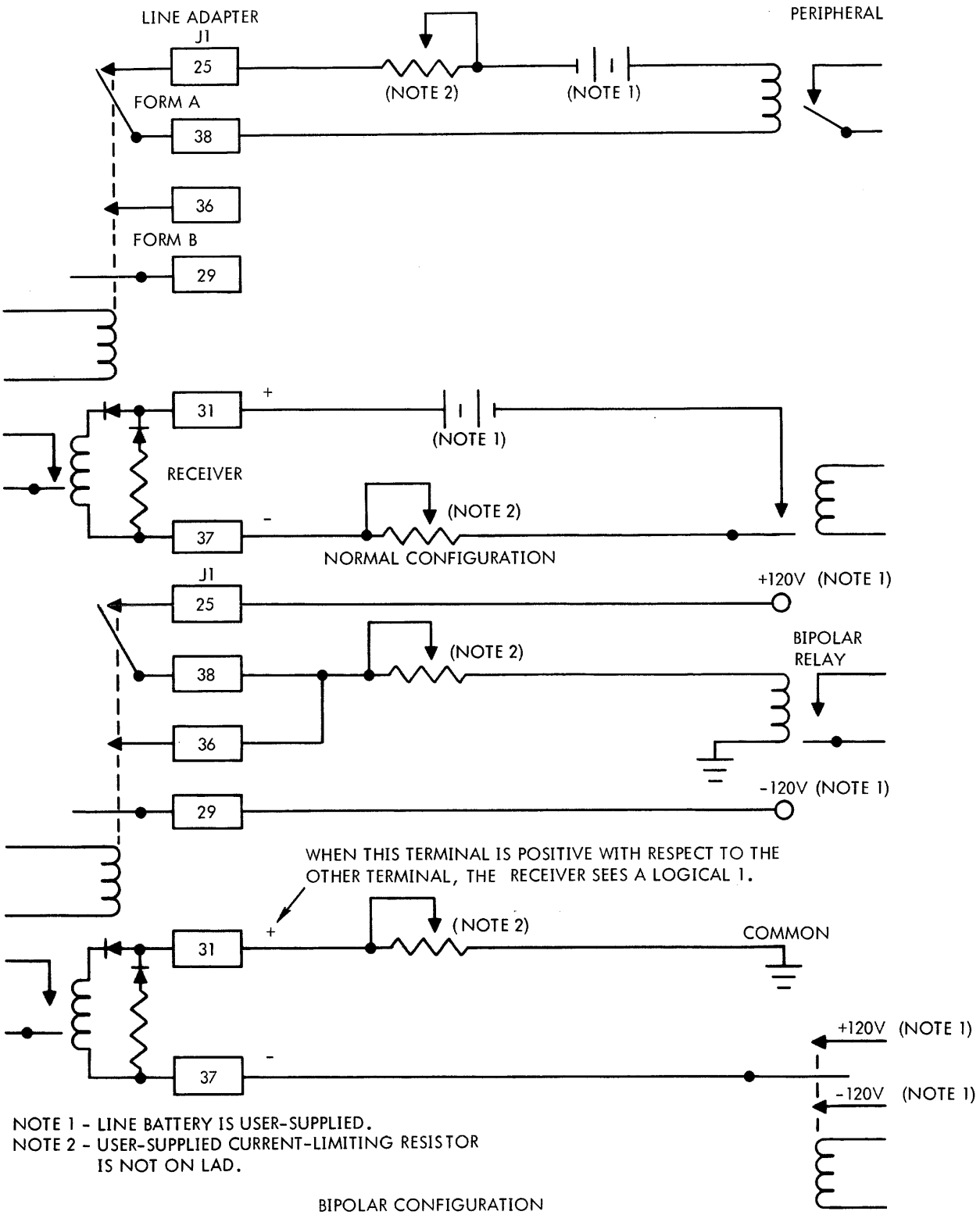
Figure 3-4. Line Battery/Device Hookup

Table 3-7. Relay Interface Specifications

Input	
Relay input-coil rating	12.5V dc maximum
Relay input-coil resistance	1250 ohms
Output	
Reed relay contact closure	Form A, form B, or form C
Maximum contact voltage	120V dc
Isolation	
Common-mode resistance	10 megohms at 400V dc

Table 3-8. Relay Interface Pin Connections

Line	Signal	Pin	Definition
0	NC0	J1-025	Form A contacts (closed for a logical 1)
0	NCS0	J1-038	Form A contacts (closed for a logical 1)
0	NOS0	J1-036	Form B contacts (open for a logical 1)
0	NO0	J1-029	Form B contacts (open for a logical 1)
0	RLY0+	J1-031	Positive receiver input
0	RLY0–	J1-037	Negative receiver input
1	NC1	J1-001	Form A contacts (closed for a logical 1)
1	NCS1	J1-010	Form A contacts (closed for a logical 1)
1	NOS1	J1-008	Form B contacts (opened for a logical 1)
1	NO1	J1-003	Form B contacts (opened for a logical 1)
1	RLY1+	J1-005	Positive receiver input
1	RLY1–	J1-007	Negative receiver input
2	NC2	J2-025	Form A contacts (closed for a logical 1)
2	NCS2	J2-038	Form A contacts (closed for a logical 1)
2	NOS2	J2-036	Form B contacts (opened for a logical 1)
2	NO2	J2-029	Form B contacts (opened for a logical 1)
2	RLY2+	J2-031	Positive receiver input
2	RLY2–	J2-037	Negative receiver input
3	NC3	J2-001	Form A contacts (closed for a logical 1)
3	NCS3	J2-010	Form A contacts (closed for a logical 1)
3	NOS3	J2-008	Form B contacts (opened for a logical 1)
3	NO3	J2-003	Form B contacts (opened for a logical 1)
3	RLY3+	J2-005	Positive receiver input
3	RLY3–	J2-007	Negative receiver input



VTII-1922

Figure 3-5. Relay Interface

3.1.3 Options

Parity (odd, even, or none), character length (5, 6, 7, or 8 bits), and the number of stop bits (1 or 2) are set on a line-by-line basis using the data-format toggle switches (figure 3-2) as follows:

	Switch	1 PS	2 WLS1	3 WLS2	4 SBS	5 PI
Parity	Odd	On				On
	Even	Off				On
	None					Off
Length	5		On	On		
	6		Off	On		
	7		On	Off		
	8		Off	Off		
Stop bits	1				On	
	2				Off	

3.1.3.1 Line Bit-Rate Selection

Each line can be jumpered for one of the six bit-rates provided by the multiplexor bus. The frequencies of these bit-rates are determined by jumpers on the DCM backplane at the individual LAD locations. One clock per line is jumpered into the LAD, and these clocks operate at 16 times the actual desired line bit-rate. Table 3-9 shows the jumper points for the inputs to the LAD and the pin location of the six bit-rates provided by the multiplexor (BR0 through BR5). The frequencies are determined by jumpers on MU board 1. If no bit rates are specified, the speed is 1760 Hz (16 x 110).

Table 3-9. Line Bit-Rate Selection Multiplexor Bus Bit-Rate Locations

Standard Frequencies	Mnemonic	Pin Location at LAD Slot
153,600 Hz (16 x 9600)	BR0	101
38,400 Hz (16 x 2400)	BR1	102
19,200 Hz (16 x 1200)	BR2	103
4,800 Hz (16 x 300)	BR3	104
2,400 Hz (16 x 150)	BR4	105
1,760 Hz (16 x 110)	BR5	106

Line Adapter Clock Inputs (at Backplane)

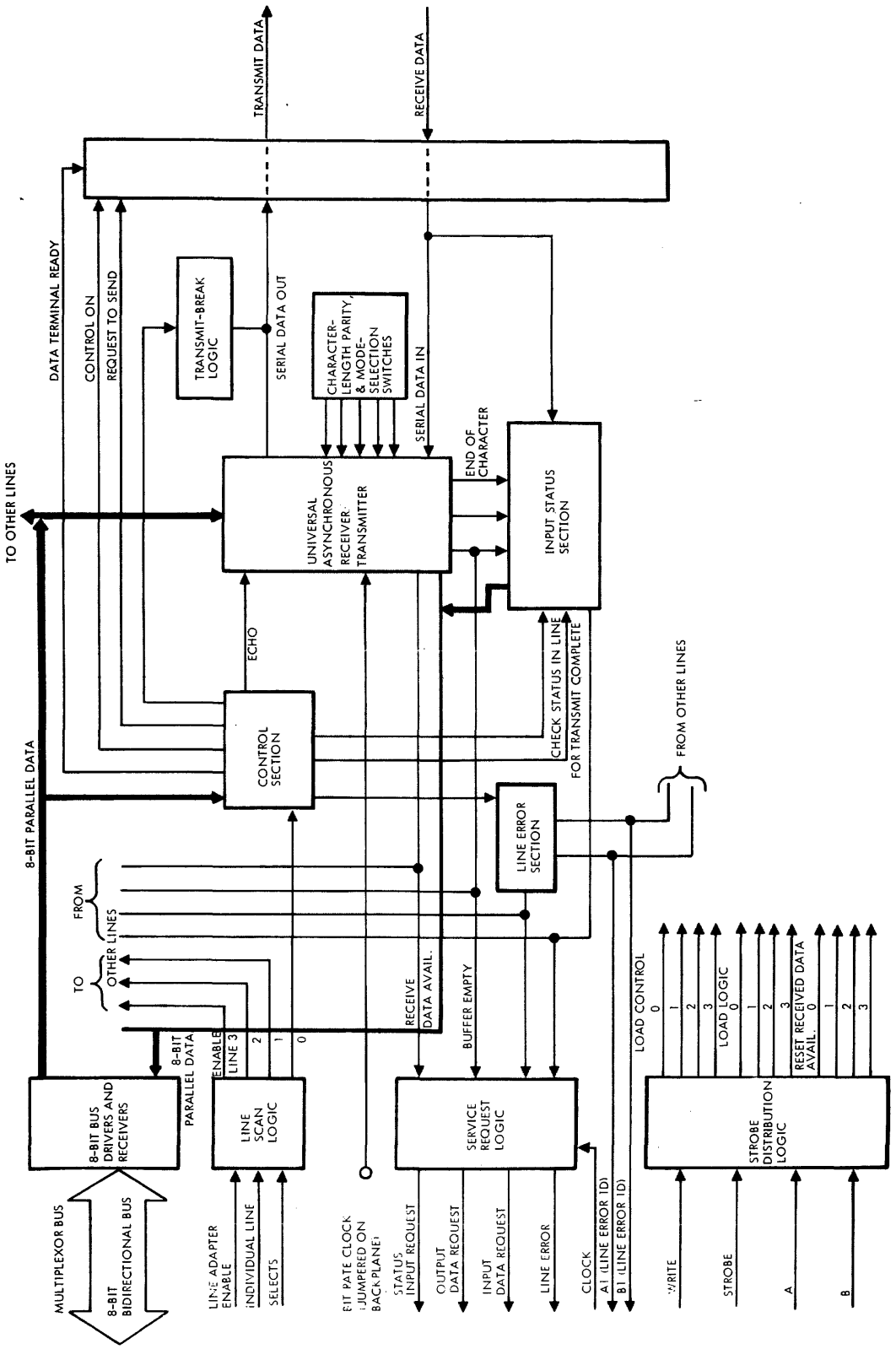
Line	Input Pin
0	92
1	93
2	98
3	97

3.1.3.2 Solid-State Current-Loop Clock

If the LAD is the 44 D 0654 001 version, a clock (CLKB+) is wired in at the DCM backplane. This clock is buffered on the LAD and brought out on P1 pin 42 for the next LAD of this type. The clock is picked up at slot 10 of the DCM main backplane (MU board 1) pin 50 and wired to pin 44 of the first solid-state current-loop direct-connection LAD. If another LAD of this type is used, a wire is added from pin 42 of the first LAD to pin 44 of the next LAD.

3.2 THEORY OF OPERATION

This section explains the operation of the direct-connection LAD, with each subsection corresponding to a block (or set of blocks) on the simplified LAD block diagram (figure 3-1). Figure 3-6 is a detailed block diagram. Refer to logic diagram 91B0414 for understanding this theory.



VTI2-407

Figure 3-6. Direct-Connection Line Adapter Detailed Block Diagram

3.2.1 Multiplexor Bus Interface

This logic provides a common interface for the four lines of the LAD to the multiplexor bus (section 2.1.2.2). This interface includes provision for jumpering a different speed for each line and assigns the LAD an address on the multiplexor bus.

Data/status selection: This LAD output to the multiplexor eight-bit bus has two sources: the eight-bit receiver-data bus (RD0+ through RD7+) from the four asynchronous receiver/transmitter devices, and the two status outputs (SB0 + and SB1 +). Selection of these outputs is accomplished with a quad-two-line-to-one-line data-selector device. The selection term is EDTA−, and the data-selector devices are enabled by EOB−. These terms are formed with half of a dual two-line-to-four-line decoder, which is enabled by LAD enabler (LAE1−) when the multiplexor is reading (WRITE + false). EOB− is true if the data-selection terms A− and B− are both false, or if A− and B− is true. EDTA− is true only when A− and B− are both false. EDTA− true gates the receiver data bus out.

LAD enabling logic: The LAD address input (LADA−) and the LAD address-enabler input (LADE−) are ANDed to form the LAD enabling terms LAE1−, LAE2−, and LAE3−.

3.2.2 Service-Request Logic

The three service-request terms IDRSD+, ODRSD+, and SIRSD+ are clocked into the service-request register on the positive transition of CLK−. The outputs of the register are gated to form a priority structure where the input data-request has the highest priority and the input-status request has the lowest. When an input data-request is made, a line error may be reported (LERR−). If a line error has occurred, the AI− and BI− identify the type of error.

Input data-request (IDRSD+): An input data-request is made if the data ready flag is true (DR+) and the line is in receiving mode, or if the line is in transmission mode and a framing error occurs.

Output data-request (ODRSD+): An output data-request is made if the line is in transmission mode, echo mode is not selected, output-buffer and output-register conditions are not being reported (U− and SB5+), and the output buffer is empty (TBREA+).

Input-status request (SIRSD+): An input-status request is made under the following conditions:

a. The lines ICS+ signal is true and a change of state in the control-line-in is detected.

b. The I+ bit is set in the line-control byte and the transmission buffer and transmission register are both empty.

3.2.3 Control Logic

The control logic for each line (zero through three) is identical. It provides steering logic for data/control transfer to and from the line, storage flip-flops for the control and status bytes, break-transmitting logic, and service-request generation logic (i.e., input or output data requests, etc.).

The control bits are stored in quad-D flip-flop modules with tristate outputs. The tristate outputs allow the corresponding bits of each line to be tied together and only one register at a time is enabled as the lines are sequentially scanned. Each control bit function is described below.

Transmission: T+ indicates that the selected line is in transmission mode. T+ is gated with the LAD enabler to produce SMT+, which enables output and input data requests in the event of a framing error.

Receive: R+ indicates that the line is in receiving mode. R+ enables input data requests as data are received, and enables the data strobe to the asynchronous receiver/transmitter device when a line is in echo mode.

Echo: E+ inhibits output data requests and causes all received data to be transmitted automatically.

Break: TB+ enables the transmit-break logic.

Control Line: ICS+ controls the control-line-in change-of-state detection logic.

Interrupt on underflow: The MT+ output of the control registers are used to enable a status change request when the lines' output buffer and output register are both empty.

3.2.4 Strobe Distribution Logic

A strobe pulse is sent from the multiplexor as a load pulse or a reset pulse, depending on the sequence taking place.

Reset data available (RDA+): This signal clears the data-available flag (DR+) of the four asynchronous transmitter/receiver devices. RDA+ is distributed to the devices via one-half of a dual demultiplexor device. The device provides four outputs, one for each line, that are individually selected by SCB0+ and SCB1+ decoded. The device is enabled by a LAD enabler. RDA+ is a function of the strobe pulse, WRITE+, and input-data-request. Note that DRDA− inhibits the RDA+ strobe if an overrun error occurs (OEB+) after the input data-request sequence has begun. RDA+ is also enabled when a line is first put in receving mode to clear the data ready flag, so that erroneous data are not transferred to the computer.

Reset status input (RSI+): This signal clears the individual control-channel edge-detectors after an input status-request has been made. RSI+ is distributed to the edge detectors in the same manner as RDA+. RSI+ occurs when the ICS bit is first enabled. This ensures that the edge-detectors are cleared when a line is first enabled. RSI+ also occurs when a line has reported an input status-

request and the multiplexor responds with a strobe with WRITE + false. RSI + is inhibited if the underflow signal (TC–) is true.

Control strobe (CST +): This signal loads the control registers with the data on the multiplexor bus. CST + is formed with half of a dual two-to-four decoder device that decodes A– and B– . The strobe pulse appears on CST + if A– is false, B– is true, and WRITE + is true. CST + is distributed to the individual lines in the same manner as the previous strobe signals.

Strobe data (STDD–): This pulse loads data from the multiplexor bus into the individual asynchronous transmitter/receiver device output buffers. STDD– is formed with the same device as CST– , and occurs if A– and B– are both false. The distribution to the individual lines is analogous to that of other strobe signals.

Strobe Break (SBRK–): SBRK– is a result of ANDing STD + with the transmit-break enabling signal. Distribution to the individual lines is via a demultiplexor device.

3.2.5 Individual Line-Scanning Logic

Four line-enabling signals (SCAN0– , SCAN1– , SCAN2– , and SCAN3–) are formed by decoding the two least-significant scan-counter bits SCB0 + and SCB1 + . The line-enabling signals enable the tristate receiver data outputs of the asynchronous transmitter/receiver devices and the tristate outputs of the control register.

3.2.6 Control-Channel Edge Detectors

This logic consists of four separate identical sections (one for each line). Each section contains a flip-flop that remembers the state of the control line, and an exclusive NOR gate that compares the control-channel input state with the flip-flop state. If a difference is detected, a high CCM + (multiplexed CCR0 + , CCR1 + , CCR2 + , and CCR3 +) is sent to the input status-request logic. A strobe causes the flip-flop to go to the state of the control-channel input after the input status-request has been acknowledged.

3.2.7 Asynchronous Receiver/Transmitter

The asynchronous receiver/transmitter sections for each line (zero through three) are identical. They are general purpose, programmable MOS/LSI devices for the transmission and receipt of asynchronous serial data. The transmitter converts parallel data into a serial word that contains the data along with start, parity, and stop bits. The receiver section converts a serial word with start, data, parity, and stop bits, into parallel data and verifies proper code transmission by checking parity and the receipt of a valid stop bit. Both receiver and transmitter are character-buffered. The line toggle-switches can be set up for a word length of 5, 6, 7, or 8 bits; even, odd or no parity; and one or two stop-bits (section 3.1.3). Table 3-10 gives the individual signal descriptions.

Table 3-10. Pin Functions of the Asynchronous Receiver/Transmitter

Pin	Symbol	Name	Function
1	VSS	Power supply	+ 5V supply
2	VDD	Power supply	– 12V supply
3	GND	Ground	Ground
4	RDE	Receiver data enabler	A low-level input enables the outputs (RD8 through RD1) of the receiver buffer-register
5 through 12	RD8 through RD1	Receiver data outputs	These are the 8 tristate data outputs enabled by RDE; unused data output enabled by RDE; unused data output have a low-level output, and received characters are right-justified, i.e., the LSB always appears on the RD1 output
13	PRE	Receiver parity error	This tristate output (enabled by SWE) is high if the received character parity bit does not agree with the selected parity
14	RFE	Receiver framing error	This tristate output (enabled by SWE) is high if the received character has no valid stop-bit

(continued)

Table 3-10. Pin Functions of the Asynchronous Receiver/Transmitter (continued)

Pin	Symbol	Name	Function
15	ROR	Receiver overrun	This tristate output (enabled by SWE) is high if the previously received character is not read (RDA output not reset) before the present character is transferred into the receiver buffer-register
16	SWE	Status word enabler	A low-level input enables the outputs (RPE, RFE, ROF, RDA, and TBMT) of the status-word buffer-register
17	RCP	Receiver clock	This input is a clock whose frequency is 16 times the desired receiver baud rate
18	RDAR	Receiver data-available reset	A low-level input forces the RDA output low
19	RDA	Receiver data available	This tristate output (enabled by SWE) is high when an entire character has been received and transferred into the receiver buffer-register
20	RSI	Receiver serial input	This input accepts the serial-bit input stream; a high-(mark) to-low (space) transition is required to initiate data reception
21	MR	Master reset	This input is pulsed to a high level after power turn-on, setting TSP, TEOC, and TMBT high and resets RDA, RPE, RFE, and ROR low
22	TBMT	Transmitter buffer empty	This tristate output (enabled by SWE) is high when the transmitter buffer-register can be loaded with new data
23	TDS	Transmitter data strobe	A low input strobe enters the data bits into the transmitter buffer-register
24	TEOC	Transmitter end-of-character	This output appears high each time a full character is transmitted, and remains high until the start of transmission of the next character (or for one-half of a TCP period in the case of continuous transmission)
25	TSO	Transmitter serial output	This output provides the entire transmitted character serially, and remains high when no data are being transmitted
26 through 33	TD1 through TD8	Transmitter data inputs	Eight data input lines (strobed by TDS) are available; unused data input lines, as selected by NDB1 and NDB2, may be in either logic state (LSB should always be placed on TD1)
34	CS	Control strobe	A high input enters the control bits (NDB1, NDB2, NSB, POE, and NPB) into

(continued)

Table 3-10. Pin Functions of the Asynchronous Receiver/Transmitter (continued)

Pin	Symbol	Name	Function
			the control-bit holding-register; this line can be strobed or hard-wired high
35	NPB	No parity bit	A high input eliminates the parity bit from being transmitted; stop bit(s) immediately follow the last data bit; in addition, the receiver requires the stop bit(s) to follow last data bit immediately, and RPE output is forced low
36	NSB	Number of stop-bits	This input selects the number of stop-bits, 1 or 2, to be transmitted (low input selects one stop-bit; high selects two)
37 through 38	NDB2, NDB1	Number of data bits/ character	These two inputs are decoded internally to select 5, 6, 7, or 8 data-bits/ character as follows:

NDB2	NDB1	Data bits/character
L	L	5
L	H	6
H	L	7
H	H	8

Pin	Symbol	Name	Function
39	POE	Select odd/even parity	The logic levels on this input and on NPB, determine the parity mode for both the receiver and transmitter, as follows:

NPB	POE	
L	L	Odd parity
L	H	Even parity
H	L	No parity

40	TCP	Transmitter clock	This input is a clock whose frequency is 16 times the desired transmitter baud rate

3.2.8 Line-Terminal Interface

For the direct-connection LAD, there are three types of line-terminal interface (tables 3-2 and 3-3): RS232, current-loop, and relay. In addition the LAD can be used as a TTL/ DTL interface.

The serial output of the transmitter is ORed with the break output to drive SDX+.

Any version can operate in any of the following modes:

a. Full duplex (simultaneously transmitting and receiving)

b. Half duplex (transmitting and receiving, but not simultaneously)

c. Transmitting only

d. Receiving only

3.2.8.1 RS232 Interface

This interface conforms to the RS232C and CCITT V24 standards and provides for the following signals:

AA–	Protective ground
AB–	Signal ground
BA–	Transmitted data
BB–	Received data

The maximum cable length is 50 feet (15 meters).

SDX+ drives an RS232 output drive IC to prodcue the RS232-compatiable serial output (SDDSX). RDDSX (RS232 serial-data input) is applied to an RS232 receiver IC to

generate RDCX+, which is applied to the MOS receiver and the control line logic. In addition, a DTL/TTL-compatible open-collector negative-true serial output (SDX−) is provided.

3.2.8.2 Current-Loop Interface

This version provides an isolated 20-60 mA., maximum 120V dc interface. Isolation exceeds 10 megohms at 50V dc common mode. The interface requires a user-supplied line battery power source.

SDX is ANDed with high-speed clock CLK-B via jumper Y-Z and driven through a current-driver as SDX+D. This serrated serial-data signal is applied to one side of T1 for isolation from the discrete driver Q1. While data are true (marking), Q1 is kept on by the half-wave rectified output of T1. OUTX− and OUTX+ form the driving half of the current loop. The current-loop receiver is an optically isolated IC. 20 to 60 mA of current flowing between INX+ and INX− cause the LED in the receiver to turn on the phototransistor driver. The output of the transistor is buffered to generate ROCX+.

3.2.8.3 Relay Interface

SDX+D (buffered SDX+) is applied to the coil of K1. The outputs available are a normally closed pair (NCX and NCSX), a normally open pair (NOX and NOSX). A form C closure may be emulated by tying NOSX and NCSX together. The receiver (RLYX+ and RLYX−) are the two sides of a relay coil. If R33 is shorted out, 12V dc will activate the relay to make RDCX true.

3.2.8.4 TTL/DTL Interface

The direct-connection LAD can serve as a TTL/DTL interface. There is an open-collector TTL driver with 70 mA sinking capacity, and the RS232 receiver can be used as a DTL receiver presenting a 8.3 mA load to the input line. The maximum cable length is 20 feet (6 meters).

3.3 MAINTENANCE

The following equipment is required to maintain the LAD:

a. Varian 620-series or V70 series computer with at least 8K of memory

b. Data communications multiplexor

c. An I/O expansion chassis for the DCM backplane

d. An I/O expansion power supply

e. One card extender (44 D 0015 or 44 D 0540 000)

f. Two test connectors (burndy edge connectors 57 A 0036 000) wired for the applicable interface:

1. RS232 interface (44 D 0654 002): Install jumpers as follows:

Pin 2 to pin 22
Pin 4 to pin 24

2. Current-loop interface: (44 D 0654 001): Install jumpers as follows:

Pin 3 to pin 11
Pin 29 to pin 39
Pin 37 to pin 40
Pin 27 to pin 34 to pin 6
Pin 12 to pin 7

Note: R27, R30, R39, R42, R51, R54, R63, and R66 must be shorted across when using this test connector.

3. Relay interface (44 D 0654 000): Install jumpers as follows:

Pin 1 to pin 25 to pin 44
Pin 10 to pin 5
Pin 31 to pin 38
Pin 7 to pin 27 to pin 37

Note: R33, R29, R45, R41, R53, R57, R65, and R69 must be shorted across, and jumpers placed where R22 and R46 are mounted in order to use this test connector.

g. Software tapes:

MAINTAIN II Test Executive	92 U 0107 001
DCM Test Program	92 U 0106 009B

h. One oscilloscope, Tektronix 547 or equivalent

The following system configuration features must be checked before testing:

a. Data format (this will be changed during testing and thus must be to the original configuration after testing is completed)

b. If the current-loop LAD is being tested, ensure that CLKB+ is wired to P1 pin 44 of the LAD slot.

c. Line bit-rate selection (ensure that each line has an input clock; reset the rates to system specifications if they have been changed during testing)

3.3.1 Functional Tests

The following tests use portions of the DCM test program. The specific test required is called out in each section.

Input/output test: Verify that each line can transmit and receive a binary data pattern without errors. Run test 1

using all possible data formats (i.e., 5-, 6-, 7-, and 8-bit data, with and without parity, and with one or two stop bits).

Break test: Verify that each line can transmit break characters and detect break characters on input. Test 4 checks these features.

Parity-error detection test: Verify that each line can detect input parity errors. This can be accomplished by switching from the one to two stop-bit data pattern, or by switching the parity enabler on and off while running test 1. The test prints out the error, if detected.

Overrun-error reporting test: Verify that each line reports overruns. Use test 5.

Status-input request test: Verify that each line will make a status-input request if a change of state of the control-line-in is detected, if a ring indication is detected, or if an underflow condition exists (output buffer and register empty). Test 5 checks all of these.

Hardware-echo test: Verify that each line will automatically echo all received data if in an applicable mode and the e bit of the line-control byte is set. If the hardware echo feature has been disabled, do not perform this test. Test 10 verifies hardware-echo operation.

Burn-in test: Test 77 should be run on each LAD for at least 2 passes.

3.3.2 Other Problems

To eliminate problems not solved by the above tests, use the techniques and programs given in section 2.3.3.

3.4 PROGRAMMING

This section describes only those aspects of programming peculiar to the direct-connection LAD. General DCM programming is discussed in section 2.4.

3.4.1 Interrupts

This section explains the use of the line-error interrupt type-of-error bits and the status-change-error interrupt status bits with regard to the direct-connection LAD. The remaining interrupt information is in section 2.4.2.

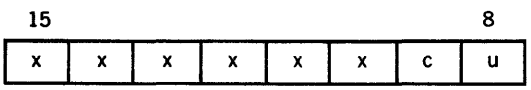
3.4.1.1 Line Error

This interrupt is explained in section 2.4.2.3. For the direct-connection LAD, the type-of-error bits a and b (bits 6 and 7) have the following significance:

ab	Type of Error
00	Framing error
01	Parity error
10	Overflow
11	Parity error *and* overflow

3.4.1.2 Status Change

This interrupt is explained in section 2.4.2.4. For the direct-connection LAD, the status bits have the format:



where the bits, when set, indicate:

u = Both output buffer and output register empty (this bit, when set, causes a status-change interrupt if the i-bit, bit 14, is set in the line control byte)

c = The control-line-in from the terminal is on (this control line is usually tied to the receive-data line and any change of state in this line causes a status-change interrupt to be generated if the c-bit was set in the control word. Note: When this function is used, the receiver is not active to receive data)

x = Not used (zero)

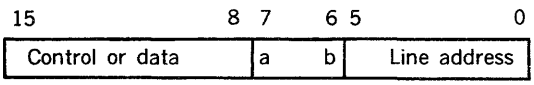
3.4.2 Programming Sequences

This section describes the line setup and status-reading control sequences used with the direct-connection LAD. General DCM programming sequences are given in section 2.4.4.

3.4.2.1 Line Setup Sequence

This sequence begins with an EXC 0570 instruction. The DCM generates a control interrupt as soon as it is free. The program responds with OAR 070. The DCM loads the specified line and resumes scanning.

The character output by OAR 070 has the format:



where a and b indicate the contents of the more-significant byte as follows:

ab Selected byte

00 Data to output buffer

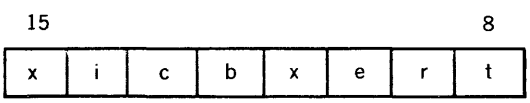
01 Control

10 Not used

11 Not used

DATA BYTE FORMAT (ab = 00): When ab = 00, the more-significant byte contains data, which are right-justified (i.e., bit 8 is the least-significant data bit regardless of character-length, and unused bits, if any, are zeros).

CONTROL BYTE FORMAT (ab = 01): When ab = 01, the more-significant byte contains control information in the format:



where the bits, when set, indicate:

t = Transmission (puts the line in transmission mode; when this bit is enabled and bit e is not, the line begins making output data-requests; when a line is in transmission mode, line breaks are reported as framing errors, all zeros, if they are received and the control line is not enabled)

*r = Receiving (puts the line in receiving mode; the line begins making input data-requests as data are received, and reports line errors as they occur)

e = Echo mode (causes the LAD to echo each word received back as a transmitted character to the terminal as the character is input from the LAD; disables all output data-requests even if the lines is in tramsmission mode)

b = Transmission break (transmits break characters, all spaces,) for each character in the line output buffer)

*c = Control line (enables the control line from the terminal, where the receive-data line doubles as the control-line-in)

i = Interrupt (causes a status-input request to be generated when both the output buffer and the output register are empty. Note: The software should clear this bit upon receipt of the status-change interrupt to prevent repetitive interrupts; it normally will be enabled when the multiplexor detects byte-count-zero and automatically reloads the control byte)

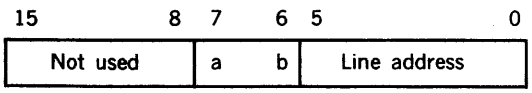
x = Not used

* NOTE: Bits c and r are mutually exclusive. If one bit is set, the other must not be set.

3.4.2.2 Read Line-Status Sequence

This sequence begins with an EXC 0670 instruction. The DCM generates a control interrupt as soon as it is free. The program responds with OAR 070 and CIA 070.

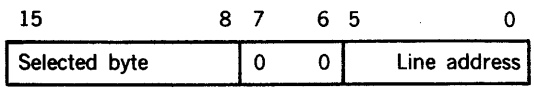
The character output by OAR 070 has the format:



where a and b select one of two possible bytes:

ab	Selected byte
00	Data
01	Line status
10	Not used
11	Not used

The character input by CIA 070 has the format:



DATA BYTE FORMAT (ab = 00): When ab = 00, the selected byte contains data in the same format as in the line-setup sequence (section 3.4.2.1).

LINE-STATUS BYTE FORMAT (ab = 01): When ab = 01, the selected byte contains line-status information in the same format as the status bits in a status-change interrupt (section 3.4.1.2).

3.5 MNEMONICS

Mnemonic	Description
AI through BI	Decoded to indicate type of line error
CCx	Control-line input change
CCM	Multiplexed line from CC0 through CC3
CLK	614,400 Hz clock from MU
CS1x CS2x	20 mA current source from −12V dc for testing
CST	Control strobe
CSTx	Control strobe for line control registers
DR	OR-tied data-ready line from the UARTs
E	Enable echo bit
EDTA	Enable input
ENOUT	Enable output
EOB	Enable status input
FE	Framing error
ICS	Input control-status enabling bit is control register
IDR	Input data-request to MU
IDRS	Stored input data request
IDRSD	Input data request to service request register
INx+ INx−	Current loop input (20-60 mA = 1; 1 mA or less = 0)
LADA	LAD address input
LADE	Enable LAD address input
LAE	Enable LAD
LERR	Line error to MU
LERRS	Stored line-error status
MB0 through MB7	Multiplexor data bus
MT	Underflow
NCx	Form B relay serial output (closed = 1; open = 0)
NCSx	Form B relay serial output (closed = 1; open = 0)
NOx NOSx	Form A relay serial output (open = 1; closed = 0)
OBRKx	Output break
ODR	Output data request to MU
ODRSD	Output data request to service-request register
OE	Overrun error
OUTx+ OUTx−	Current loop serial output (20 mA = 1; 1 mA or less = 0)
PE	Parity error
R	Enable receiving mode
RD0 through RD7	Received data
RDA	Reset data available
RDAx	Reset data available strobes
RDCx	Serial received data lines
RDDSx	RS232 serial input
RDSTRB	Read strobe
RLYx+ RLYx−	Relay serial input
RSI	Reset status input request
RSIx	Reset status input request strobe
SB0	Underflow status bit
SB1	Control-line output (read data) status bit
SBRKx	Strobe break logic
SCANx	Individual line enablers for tristate output devices
SCB0 SCB1	Decoded to address one of four lines
SDx	DTL serial data

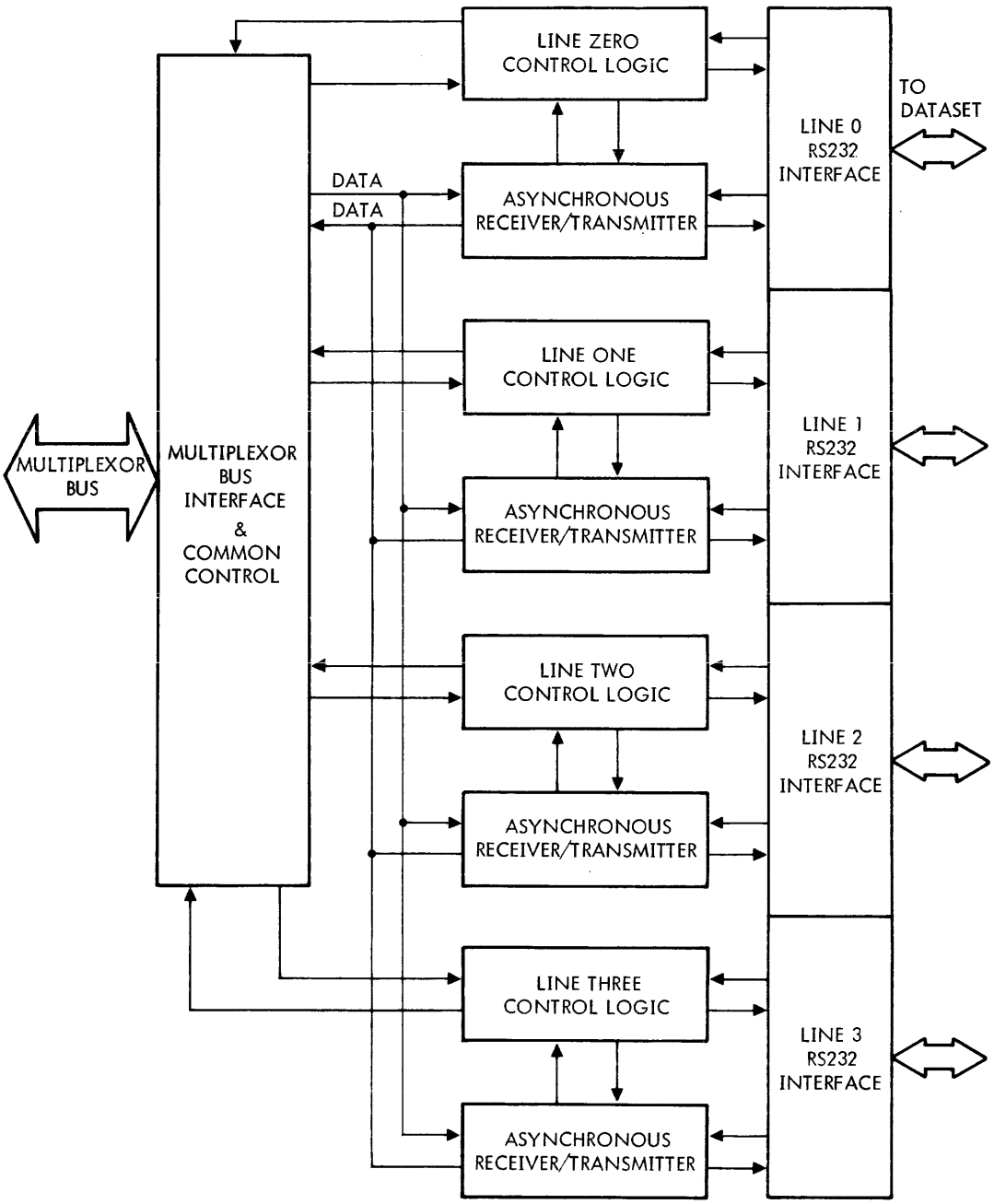
(continued)

Mnemonic	Description
SDDSx	RS232 serial output
SIR	Status input request to MU
SIRSD	Status input request to service-request register
SRST	System reset
STD	Strobe data
STDx	Strobe data signals to individual lines
STRBE	Strobe from MU
T	Enable transmission bit
TB	Enable transmission break
TBRE	Transmission buffer empty
TC	Underflow
TDx	Serial data from asynchronous transmitter/receiver
TREAx	Transmission register empty
TREM	Multiplexed transmission register empty
WRITE	Write line from MU
XMT	Transmission ANDed with LAD enabler

x = line number.

SECTION 4
ASYNCHRONOUS RS232 MODEM LINE-ADAPTER

This LAD (figure 4-1) provides an EIA RS232C interface for the DCM. It can handle four asynchronous modems (datasets), specifically the Bell 103 and 202 datasets or their equivalents. Since its interface circuits are compatible with the CCITT V24 specification, this LAD can also interface with European datasets compatible with the Bell 103 and 202.



VTII-1786

Figure 4-1. Asynchronous RS232 LAD Simplified Block Diagram

The LAD interfaces with four full-duplex asynchronous data communication lines at a maximum data rate of 9,600 bits per second. It provides character buffering for both input and output, automatically adds start and stop bits and disassembles characters, assembles characters and strips off the start and stop bits, generates output parity (if any), checks incoming data for correct parity (if any), and checks for break characters or framing errors. The parity can be odd, even, or not used; the character length can be 5, 6, 7, or 8 bits; and there can be one or two stop-bits. Each of the lines can be assigned any two of six hardwired (section 2.1.3) bit rates, which are then software-selectable. The LAD can also be modified to transmit and receive mirror-image data (most significant bit transmitted first), (section 4.13.5).

Table 4-1 gives the specifications of the asynchronous RS232 Modem LAD.

4.1 INSTALLATION

The LAD has been inspected and packed to ensure its arrival in good working order. To prevent damage, take reasonable care during unpacking and handling. Check the shipping list to ensure that all equipment has been received. Immediately after unpacking, inspect the equipment for shipping damage. If any is found:

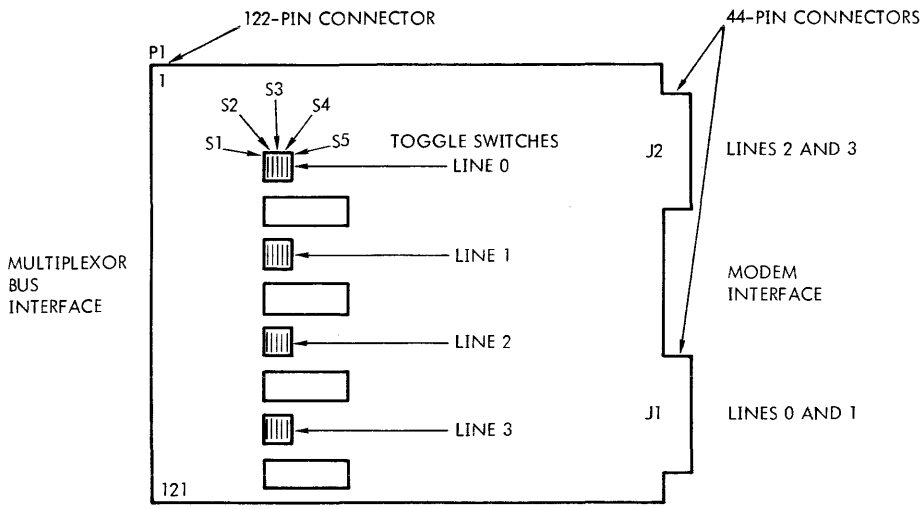
a. Notify the transportation company
b. Notify Varian Data Machines
c. Save all packing material

4.1.1 Physical Description

The asynchronous RS232 modem LAD is on one printed-circuit board (figure 4-2) mounted in a specially-wired expansion chassis (figures 1-2 and 1-3).

Table 4-1 Asynchronous RS232 LAD

Number of lines/board	Four
Maximum cable length	50 feet (15 meters)
Transmission type	Serial asynchronous (normal or mirror-image)
Modem interface	Interface circuits conform to E1A RS232-C
Character length	5, 6, 7, or 8 (hardware selectable)
Number of stop bits	1 or 2 (hardware selectable)
Parity generation and checking	Odd, even, or none (hardware selectable)
Buffering	Each line character-buffered on input and output
Maximum bit rate	9,600 bits per second
Bit rate selection	Each line is assigned two hardware-selectable speeds; these two speeds are then software-selectable
Error reporting	Parity errors, line breaks (or framing errors), and overflow errors
Package	One PC board with approximately 75 ICs
Power requirements	+5V dc at 1.2A +12V dc at 100 mA -12V dc at 150 mA
Environment	Temperature 0 to 50 degrees C, relative humidity 0 to 90 percent (without condensation)
Interconnection	Plugs into DCM backplane, interfaces with up to four modems via two 44-pin connectors



VTII-1787

Figure 4-2. Asynchronous RS232 Modem LAD

4.1.2 Interfaces

The LAD interfaces with the MU over the multiplexor bus (figures 1-1 and 2-1), and with the datasets over the RS232 modem interface (section 4.2.9).

Pin assignments are given on the logic diagrams in Volume 2 of this manual.

4.1.2.1 Multiplexor Bus

The 38-line multiplexor bus connects the LAD with the MU. It is described in section 2.1.2.2.

4.1.2.2 RS232 Modem Interface

This interface is described in section 4.2.9.

4.1.3 Options

Parity (odd, even, or none), character length (5, 6, 7, or 8 bits), and the number of stop bits (1 or 2) are set on a line-by-line basis using the data-format toggle switches (figure 4-2) as follows:

	Switch	1 PS	2 WLS1	3 WLS2	4 SBS	5 PI
Parity	Odd	On				On
	Even	Off				On
	None					Off
Length	5		On	On		
	6		Off	On		
	7		On	Off		
	8		Off	Off		
Stop bits	1				On	
	2				Off	

The standard automatic echo of received characters can be disabled by removal of a jumper.

4.1.3.1 Line Bit-Rate Selection

Each line can be jumpered for two of the six bit-rates provided by the multiplexor bus. Software then selects one of the two speeds on each line by setting or resetting the s bit in the individual line control bytes (s = 0 selects speed A; s = 1 selects speed B). The frequencies of these two bit-rates are determined by jumpers on the DCM backplane at the individual LAD locations. Two clocks per line are jumpered into the LAD, and these clocks operate at 16 times the actual desired line bit-rate. Table 4-2 shows the jumper points for the inputs to the LAD and the pin location of the six bit-rates provided by the multiplexor (BR0 through BR5). The frequencies are determined by jumpers on MU board 1. If no bit rates are specified, speed A is 1760 Hz (16 x 110), and speed B 4800 Hz (16 x 300).

Table 4-2. Line Bit-Rate Selection Multiplexor Bus Bit-Rate Locations

Standard Frequencies	Mnemonic	Pin Location at LAD Slot
153,600 Hz (16 x 9600)	BR0	101
38,400 Hz (16 x 2400)	BR1	102
19,200 Hz (16 x 1200)	BR2	103
4,800 Hz (16 x 300)	BR3	104
2,400 Hz (16 x 150)	BR4	105
1,760 Hz (16 x 110)	BR5	106

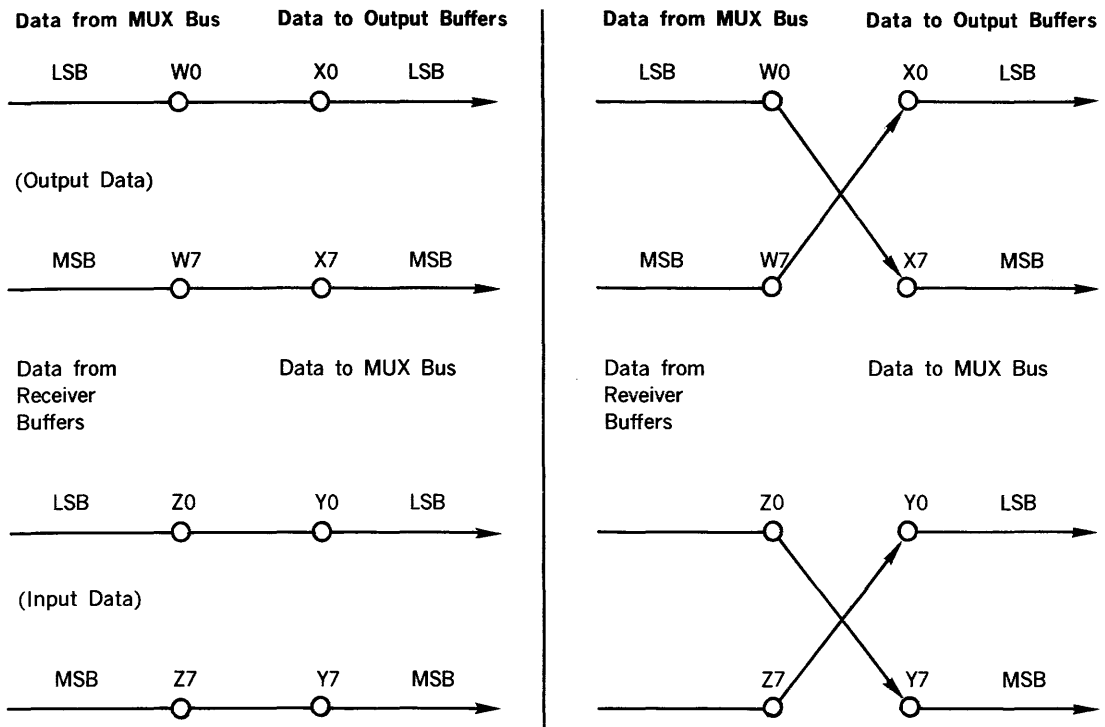
Line Adapter Clock Inputs (at Backplane)

Line	Input Pin	Remarks
0	92	Speed A
	94	Speed B
1	93	Speed A
	91	Speed B
2	98	Speed A
	95	Speed B
3	97	Speed A
	96	Speed B

4.1.3.2 Mirror-Image Data

The LAD can be modified by jumpers and etch-cutting (figure 4-3) so that all four lines accept and transmit *mirror-image data* (most-significant bit first). Note that this mode applies to the entire interface as a unit and is not on a line-by-line basis.

If the LAD is so modified, data are first left-justified when there are fewer than eight data bits within a byte, i.e., bits 15 and 7 are the most-significant data bits and are transmitted or received first. Any low-order bits shifted for this justification are filled with zeros.



Normal Configuration (Standard Etch)

W0 connected to X0
W1 connected to X1
W2 connected to X2
W3 connected to X3
W4 connected to X4
W5 connected to X5
W6 connected to X6
W7 connected to X7

Z0 connected to Y0
Z1 connected to Y1
Z2 connected to Y2
Z3 connected to Y3
Z4 connected to Y4
Z5 connected to Y5
Z6 connected to Y6
Z7 connected to Y7

Mirror-Image (Etch Cut and Jumpers Added)

W0 jumpered to X7
W1 jumpered to X6
W2 jumpered to X5
W3 jumpered to X4
W4 jumpered to X3
W5 jumpered to X2
W6 jumpered to X1
W7 jumpered to X0

Z0 jumpered to X7
Z1 jumpered to X6
Z2 jumpered to X5
Z3 jumpered to X4
Z4 jumpered to X3
Z5 jumpered to X2
Z6 jumpered to X1
Z7 jumpered to X0

Figure 4-3. Normal Mirror-Image Data Transmission

For *normal data*, the data are right-justified and the least-significant data bits (bit 8 and bit 0) handled first.

Figure 4-4 shows the typical format of data in the computer data buffer and in the control-character word. The example illustrates five-bit data characters.

Normal Data or Control-Character (Right-Justified)

BYTE ONE		BYTE TWO	
15	8	7	0
000	Data	000	Data

Least-Significant Data Bits
(Processing starts here)

Mirror-Image Data (Left-Justified)

BYTE ONE		BYTE TWO	
15	8	7	0
Data	000	Data	000

Most-signifcant data bits
(Processing starts here)

Figure 4-4. Data Format (Five-Bit Characters)

4.1.3.3 Protective Ground (RS232C Circuit AA)

The protective ground lines are not normally wired. If these are required, add jumpers as indicated in table 4-3. These jumpers bring the protective ground lines from the modem to free pins on the DCM backplane, where they can be routed to power ground. Do not tie them to signal or DC ground.

Table 4-3. Protective Ground Jumpers

Line	Jumper	Mnemonic	Backplane Pin
0	D to D	P. GND 0	64
1	C to C	P. GND 1	67
2	B to B	P. GND 2	65
3	A to A	P. GND 3	63

4.2 THEORY OF OPERATION

This section explains the operation of the asynchronous RS232 modem LAD, with each subsection corresponding to a block (or set of blocks) on the simplified LAD block diagram (figure 4-1). Figure 4-5 is a detailed block diagram. Refer to logic diagram 91B0414 for understanding this theory.

4.2.1 Multiplexor Bus Interface

This logic provides a common interface for the four lines of the LAD to the multiplexor bus (section 2.1.2.2). This interface includes provision for jumpering two different speeds per line and assigns the LAD an address on the multiplexor bus.

Data/status selection: This LAD output to the multiplexor eight-bit bus has two sources: the eight-bit receiver-data bus (RD0+ through RD7+) from the four asynchronous receiver/transmitter devices, and the six status outputs (SB0+ through SB2+ and SB5 through SB7+). Selection of these outputs is accomplished with two quad-two-line-to-one-line data-selector devices. The selection term is EDTA– and the data-selector devices are enabled by EOB–. These terms are formed with half of a dual two-line-to-four-line decoder, which is enabled by LAD enabler (LAE1–) when the multiplexor is reading (WRITE+ FALSE). EOB– is true if the data-selection terms A– and B– are both false, or if A– and B– are true. EDTA– is true only when A– and B– are both false. EDTA– true gates the receiver data bus out.

LAD Enabling Logic: The LAD address input (LADA–) and the LAD address-enabler input (LADE–) are ANDed to form the LAD enabling terms LAE1–, LAE2–, and LAE3–.

4.2.2 Service-Request Logic

The three service-request terms IDRSD+, ODRSD+, and SIRSD+ are clocked into the service-request register on the positive transition of CLK–. The outputs of the register are gated to form a priority structure where the input data-request has the highest priority and the input-status request has the lowest. When an input data-request is made, a line error may be reported (LERR–). If a line error has occurred, the AI– and BI– identify the type of error.

Input Data-Request (IDRSD+): An input data-request is made if the data ready flag is true (DR+) and the line is in receiving mode, or if the line is in transmission mode and a framing error occurs.

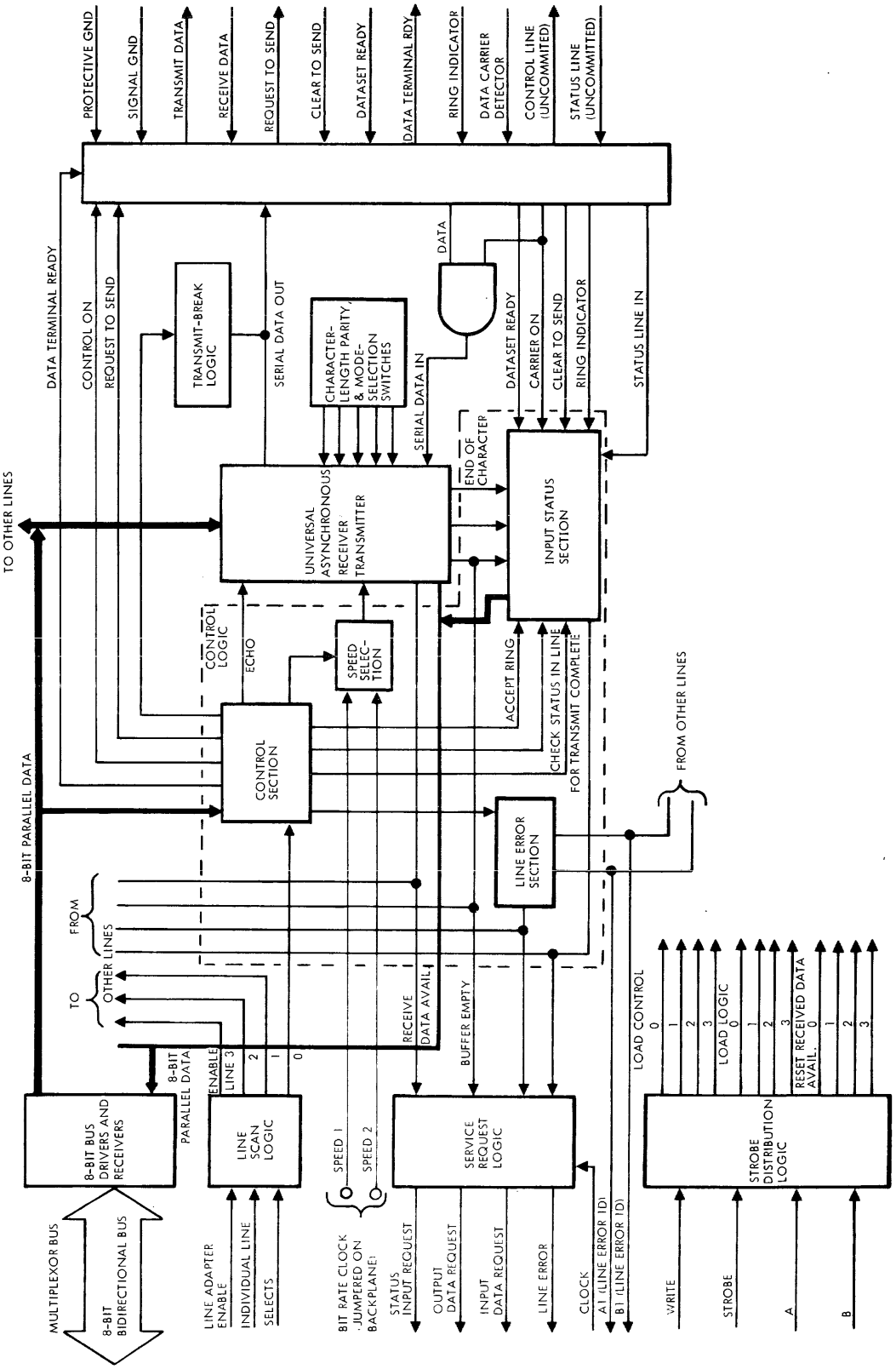
Output Data-Request (ODRSD+): An output data-request is made if the line is in transmission mode, echo mode is not selected, output-buffer and output-register conditions are not being reported (U– and SB5+), and the output buffer is empty (TBREA+).

Input-Status Request (SIRSD+): An input-status request is made under the following conditions:

a. The line is in transmission or receiving mode (TOR+) and a change of state in the control-line-in is detected (CCM+).

b. The accept-ring/break bit is set (AR+), the line is not in transmission mode, and a ring indicator is detected (SB2+).

c. The I+ bit is set in the line-control byte and the transmission buffer and transmission register are both empty.



VTI2-405

Figure 4-5. Asynchronous RS232 Modem Line Adapter Detailed Block Diagram (One Line Shown)

4.2.3 Control Logic

The control logic for each line (zero through three) is identical. It provides steering logic for data/control transfer to and from the line, storage flip-flops for the control and status bytes, break-transmitting logic, and service-request generation logic (i.e., input or output data requests, etc.).

Control of each line is through four eight-bit control registers (one for each line). The register is loaded by software via a control sequence and the outputs enable the various line functions. Each control-bit function is described below.

Speed selection: The S0, S1, S2, and S3 bits select speed A or speed B and gate it to the asynchronous transmitter/recriver devices.

Data-terminal-ready: This bit (DTR0–, DTR1–, DTR2–, and DTR3–) enables the data-terminal-ready signal to the modems.

Control line: C0–, C1–, C2–, and C3– enable and disable the lines' output-control lies.

Transmission: The T0–, T1–, T2–, and T3– bits determines whether the lines are in transmission mode or not. These bits enable the request-to-send line to the modems and the positive signals T0+, T1+, T2+, and T3+ are multiplexed to form TM+, which is then used in the service-request logic. The multiplexor chip that forms TM+ is enabled by LAD LAE2–, and the signals SCB0+ and SCB1+ are decoded to determine which line whose transmission bit is gated onto the TM+ line.

Interrupt on underflow: The I+ bit outputs of the control registers are ORed together and used to enable a status-change request when the output buffer and output register are both empty.

Echo: The echo bits E+ are ORed together, and inhibit output data-requests from a line in transmission mode. E+ also causes all input characters to be transmitted automatically.

Receiver: The receiver bits R+ are ORed together and indicate that a line is in receiving mode.

Accept ring/transmit break: The four AR+ outputs of the control registers are ORed together, and enable the accept-ring function when a line is not in transmission mode. When a line is in transmission mode, AR+ causes break characters to be transmitted.

4.2.4 Strobe Distribution Logic

A strobe pulse is sent from the multiplexor as a load pulse or a reset pulse, depending on the sequence taking place.

Reset Data Available (RDA+): This signal clears the data-available flag (DR+) of the four asynchronous transmitter/receiver devices. RDA+ is distributed to the devices via one-half of a dual demultiplexor device. The device provides four outputs, one for each line, that are individually selected by SCB0+ and SCB1+ decoded. The device is enabled by a LAD enabler. RDA+ is a function of the strobe pulse, WRITE+, and input-data-request. Note that DRDA– inhibits the RDA+ strobe if an overrun error occurs (OEA+) after the input data-request sequence has begun. RDA+ is also enabled by RDAB– when a line is first put in receiving mode to clear the data ready flag, so that erroneous data are not transferred to the computer.

Reset status input (RSI+): This signal clears the individual control-channel edge-detectors after an input status-request has been made. RSI+ is distributed to the edge detectors in the same manner as RDA+. RSI+ occurs for each line every time that any strobe occurs as long as the line is in idle (T0R– true). This ensures that the edge-detectors are cleared when a line is first enabled. RSI+ also occurs when a line has reported an input status-request and the multiplexor responds with a strobe with WRITE+ false. RSI+ is inhibited if the underflow signal (U–) is true.

Control Strobe (CST+): This signal loads the control registers with the data on the multiplexor bus. CST+ is formed with half of a dual two-to-four decoder device that decodes A– and B–. The strobe pulse appears on CST+ if A– is false, B– is true, and WRITE+ is true. CST+ is distributed to the individual lines in the same manner as the previous strobe signals.

Strobe data (STDD–): This pulse loads data from the multiplexor bus into the individual asynchronous transmitter/receiver device output buffers. STDD– is formed with the same device as CST–, and occurs if A– and B– are both false. The distribution to the individual lines is analogous to that of other strobe signals.

Strobe break (SBRK–): SBRK– is a result of ANDing STDD+ with the accept-ring/transmit-break enabling signal AR+. Distribution to the individual lines is via a demultiplexor device.

4.2.5 Individual Line Scan Logic

Four line-enabling signals (SCAN0–, SCAN1–, SCAN2–, and SCAN3–) are formed by decoding the two least-significant scan-counter bits SCB0+ and SCB1+. The line-enabling signals enable the tristate receiver data outputs of the asynchronous transmitter/receiver devices and the four tristate outputs of the control registers: AR+, R+, E+, and I+.

4.2.6 Control-Channel Edge Detectors

This logic consists of four separate identical sections (one for each line). Each section contains a flip-flop that remembers the state of the control line, and an exclusive NOR gate that compares the control-channel input state with the flip-flop state. If a difference is detected, a high

CCM+ (multiplexed CCR0+, CCR1+, CCR2+, and CCR3+) is sent to the input status-request logic. A strobe causes the flip-flop to go to the state of the control-channel input after the input status-request has been acknowledged.

4.2.7 Asynchronous Receiver/Transmitter

The asynchronous receiver/transmitter section for each line (zero through three) are identical. They are general purpose, programmable MOS/LSI devices for the transmission and receipt of asynchronous serial data. The transmitter converts parallel data into a serial word that contains the data along with start, parity, and stop bits. The receiver section converts a serial word with start, data, parity, and stop bits, into parallel data and verifies proper code transmission by checking parity and the receipt of a valid stop bit. Both receiver and transmitter are character-buffered. The line toggle-switches can be set up for a word length of 5, 6, 7, or 8 bits; even, odd, or no parity and one or two stop-bits (section 4.1.3). Table 4-4 gives the individual signal descriptions.

Table 4-4 Pin Functions of the Asynchronous Receiver/ Transmitter

Pin	Symbol	Name	Function
1	VSS	Power supply	+5V supply
2	VDD	Power supply	-12V supply
3	GND	Ground	Ground
4	RDE	Receiver data enabler	A low-level input enables the outputs (RD8 through RD1) of the receiver buffer-register.
5 through 12	RD8 through RD1	Receiver data outputs	These are the 8 tristate data outputs enabled by RDE. Unused data output lines, as selected by NDB1 and NDB2, have a low-level output, and received characters are right-justified, i.e., the LSB always appears on the RD1 output.
13	PRE	Receiver parity error	This tristate output (enabled by SWE) is high if the received character parity bit does not agree with the selected parity.
14	RFE	Receiver framing error	This tristate output (enabled by SWE) is high if the received character has no valid stop-bit.
15	ROR	Receiver overrun	This tristate output (enabled by SWE) is high if the previously received character is not read (RDA output not reset) before the present character is transferred into the receiver buffer-register.
16	SWE	Status word enabler	A low-level input enables the outputs (RPE, RFE, ROF, RDA, and TBMT) of the status-word buffer-register.
17	RCP	Receiver clock	This input is a clock whose frequency is 16 times the desired receiver baud rate.
18	RDAR	Receiver data-available reset	A low-level input forces the RDA output low.

(continued)

Table 4-4 Pin Functions of the Asynchronous Receiver/ Transmitter (continued)

Pin	Symbol	Name	Function
19	RDA	Receiver data available	This tristate output (enabled by SWE) is high when an entire character has been received and transferred into the receiver buffer-register.
20	RSI	Receiver serial input	This input accepts the serial-bit input stream. A high (mark) to-low (space) transition is required to initiate data reception.
21	MR	Master reset	This input is pulsed to a high level after power turn-on, setting TSP, TEOC, and TMBT high and resets RDA, RPE, RFE, and ROR low.
22	TBMT	Transmitter buffer empty	This tristate output (enabled by SWE) is high when the transmitter buffer-register can be loaded with new data.
23	TDS	Transmitter data strobe	A low input strobe enters the data bits into the transmitter buffer-register.
24	TEOC	Transmitter end-of-character	This output appears high each time a full character is transmitted, and remains high until the start of transmission of the next character (or for one-half of a TCP period in the case of continuous transmission).
25	TSO	Transmitter serial output	This output provides the entire transmitted character serially, and remains high when no data are being transmitted.
26 through 33	TD1 through TD8	Transmitter data inputs	Eight data input lines (strobed by TDS) are available. Unused data input lines, as selected by NDB1 and NDB2, may be in either logic state. (The LSB should always be place on TD1.)
34	CS	Control strobe	A high input enters the control bits (NDB1, NDB2, NSB, POE, and NPB) into the control-bit holding-register. This line can be strobed or hard-wired high.
35	NPB	No parity bit	A high input eliminates the parity bit from being transmitted. Stop bit(s) immediately follow the last data bit. In addition, the receiver requires the stop bit(s) to follow last data bit immediately. Also, the RPE output is forced low.
36	NSB	Number of stop-bits	This input selects the number of stop-bits, 1 or 2, to be transmitted. A low input selects one stop-bit; a high selects two.

(continued)

Table 4-4 Pin Functions of the Asynchronous Receiver/ Transmitter (continued)

Pin	Symbol	Name	Function
37 through 38	NDB2, NDB1	Number of data bits/ character	These two inputs are decoded internally to select 5, 6, 7, or 8 data-bits/ character as follows: NDB2 NDB1 Data bits/character L L 5 L H 6 H L 7 H H 8
39	POE	Select odd/even parity	The logic levels on this input and on NPB, determine the parity mode for both the receiver and transmitter, as follows: NPB POE L L Odd parity L H Even parity H L No parity
40	TCP	Transmitter clock	This input is a clock whose frequency is 16 time the desired transmitter baud rate.

4.2.8 Transmit-Break Logic

Each line has logic to transmit break characters. This circuitry consists of two flip-flops and an exclusive-NOR gate. The first flip-flop is set by the data strobe when AR + is true. The second is set when the transmission-register-empty flag goes false (TREO– for line zero) and the first flip-flop cleared. The output of the second flip-flop forces the data transmission line to a space condition for one character. At the end of that character, the transmission-register-empty flag momentarily goes true while a new character is loaded into the transmission register, causing the break flip-flop to be clocked again. If the first flip-flop had been set again while the previous break character was being output, the second flip-flop remains set after the clock. Note that the data being input to the asynchronous receiver/transmitter have no effect; the device is only used while transmitting breaks for timing.

The LAD also has two general-purpose lines that can be used to indicate that the reverse channel is on ("input control line") and to turn on the reverse channel ("output control line"). Note that the LAD does *not* provide data communications capability over the reverse channel.

These general-purpose lines can also be used for input and output speed selection (RS232C circuits CH– and CI–).

4.2.9 RS232 Modem Interface

This interface conforms to the RS232C and CCITT V24 standards and provides for the following signals for each line (pin numbers are indicated in table 4-5):

AA–	Protective ground
AB–	Signal ground
BA–	Transmitted data
BB–	Received data
CA–	Request to send
CB–	Clear to send
CC–	Dataset ready
CD–	Data terminal ready
CE–	Ring indicator
CF–	Data carrier detector

Table 4-5. RS232 Signal Locations

LAD Line Number	RS232 Signal	DCM Mnemonic	Pin
0	AA	PGND0	J1-24
0	BA	TDA0–	J1-31
0	BB	RDS0–	J1-23
0	CA	RTS0 +	J1-27
0	CB	CS0 +	J1-37
0	CC	ILK0 +	J1-33
0	AB	GRD	J1-20,22
0	CF	CO0 +	J1-25
0	CD	DTRY0 +	J1-29
0	CE	RID0 +	J1-35
0	-	CL0 +	J1-21
0	-	CCO0 +	J1-18
0		- 12V	J1-36
0	-	+ 12V	J1-40
1	AA	PGND1	J1-10
1	BA	TDA1–	J1-5
1	BB	RDS1–	J1-9
1	CA	RTS1 +	J1-3
1	CB	CS1 +	J1-19
1	CC	ILK1 +	J1-11
1	AB	GRD	J1-14,12
1	CF	CO1 +	J1-7
1	CD	DTRY1 +	J1-1
1	CE	RID1 +	J1-13
1	-	CL1 +	J1-6
1	-	CCO1 +	J1-8
1	-	- 12V	J1-4
1	-	+ 12V	J1-2
2	AA	PGND2	J2-24
2	BA	TDA2–	J2-31
2	BB	RDS2–	J2-23
2	CA	RTS2 +	J2-27
2	CB	CS2 +	J2-37
2	CC	ILK2 +	J2-33
2	AB	GRD	J2-20,22
2	CF	CO2 +	J2-25
2	CD	DTRY2 +	J2-29
2	CE	RID2 +	J2-35
2	-	CC2 +	J2-21
2	-	CCO2 +	J2-18
2	-	- 12V	J2-36
2	-	+ 12V	J2-40
3	AA	PGND3	J3-10
3	BA	TDA3–	J3-5
3	BB	RDS3–	J3-9
3	CA	RTS3 +	J3-3
3	CB	CS3 +	J3-19
3	CC	ILK3	J3-11
3	AB	GRD	J3-14,12
3	CF	CO3 +	J3-7
3	CD	DTRY3 +	J3-1
3	CE	RID3 +	J3-13
3	-	CL3 +	J3-6
3	-	CCO3 +	J3-8
3	-	- 12V	J3-4
3	-	+ 12V	J3-2

The LAD also has two general-purpose lines that can be used to indicate that the reverse channel is on ("input control line") and to turn on the reverse channel ("output control line"). Note that the LAD does *not* provide data communications capability over the reverse channel.

These general-purpose lines can also be used for input and output speed selection (RS232C circuits CH– and CI–).

The interface provides +12V dc and –12V dc through 1K resistors to hardwire certain leads on for testing or for non-standard datasets. Thus, to use this LAD without a dataset (direct connection): Tie CB–, CC–, and CF– to the +12V dc provided.

4.3 MAINTENANCE

The following equipment is required to maintain the LAD:

a. Varian 620-series or V70 series computer with at least 8K of memory

b. Data communications multiplexor

c. An I/O expansion chassis for the DCM backplane

d. An I/O expansion power supply

e. One card extender (44 D 0015 or 44 D 0540 000)

f. Two test connectors (burndy edge connectors 57 A 0036 000) wired as follows:

Line 0	1.	Pin 29 to pin 35	Data-terminal-ready to ring-indicator
		Pin 35 to pin 33	Ring-indicator to dataset-ready
		Pin 29 to pin 25	Data-terminal-ready to carrier-on
	2.	Pin 27 to pin 37	Request-to-send to clear-to-send
	3.	Pin 31 to pin 23	Transmit-data to receive-data
	4.	Pin 21 to pin 18	Control-out to control-in
Line 1	5.	Pin 1 to pin 13	Data-terminal-ready to ring-indicator
		Pin 13 to pin 11	Ring-indicator to dataset-ready
		Pin 1 to pin 7	Data-terminal-ready to carrier-on
	6.	Pin 3 to pin 19	Request-to-send to clear-to-send
	7.	Pin 5 to pin 9	Transmit-data to receive-data
	8.	Pin 6 to pin 8	Control-out to control-in

The following additional jumpers can be added so the test connector can also be used on a synchronous RS232 interface LAD.

Pin 38 to pin 39
Pin 39 to pin 41
Pin 38 to pin 17
Pin 17 to pin 15

These jumpers tie the test clock to the receive/transmit clock inputs.

g. Software tapes:

MAINTAIN II Test Executive	92 U 0107-001
DCM Test Program	92 U 0106-009B

h. One oscilloscope, Tektronix 547 or equivalent

The following system configuration features must be checked before testing:

a. Data format (this will be changed during testing and thus must be reset to the original configuration after testing)

b. Mirror-image mode or normal-data mode

c. Hardware echo, if required

d. Line bit-rate selection (ensure that each line has two input clocks; reset the rates to system specifications if they have been changed during testing)

4.3.1 Functional Tests

The following tests use portions of the DCM test program. The specific test required is called out in each section.

Input/output test: Verify that each line can transmit and receive a binary data pattern without errors. Run test 1 using all possible data formats (i.e., 5-, 6-, 7-, and 8-bit data, with and without parity, and with one or two stop bits). While running test 1, verify that the serial data out of each line (TDA0– through TDA3–) is in the range +9V to +12V for set bits and –9V to –12V for reset bits.

Speed-selection test: Verify that each line can transmit and receive data with speed B selected. Use test 3.

Break test: Verify that each line can transmit break characters and detect break characters on input. Test 4 checks these features.

Parity-error detection test: Verify that each line can detect input parity errors. This can be accomplished by switching from the one to two stop-bit data pattern, or by switching the parity enabler on and off while running test 1. The test prints out the error, if detected.

Overrun-error reporting test: Verify that each line reports overruns. Use test 5.

Status-input request test: Verify that each line will make a status-input request if a change of state of the control-line-in is detected, if a ring indication is detected, or if an underflow condition exists (output buffer and register empty). Test 5 checks all of these.

Modem status-line test: Verify, using test 6, that the three modem status lines can be read (interlock, carrier-on, and clear-to-send). Ensure that the clear-to-send status bit is set if the line is put in transmission mode and reset otherwise. The interlock status-bit and the carrier-on bit should be controllable by the data-terminal-ready bit in the line-control byte.

Hardware-echo test: Verify that each line will automatically echo all received data if in an applicable mode and the e bit of the line-control byte is set. If the hardware echo feature has been disabled, do not perform this test. Test 10 verifies hardware-echo operation.

Burn-in test: Test 77 should be run on each LAD for at least 2 passes.

4.3.2 Other Problems

To eliminate problems not solved by the above tests, use the techniques and programs given in section 2.3.3.

4.4 PROGRAMMING

This section describes only those aspects of programming peculiar to the asynchronous RS232 modem LAD. General DCM programming is discussed in section 2.4.

4.4.1 Interrupts

This section explains the use of the line-error interrupt type-of-error bits and the status-change-error interrupt status bits with regard to the asynchronous RS232 modem LAD. The remaining interrupt information is in section 2.4.2.

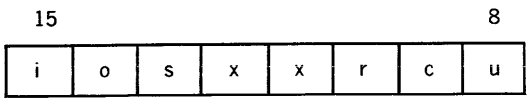
4.4.1.1 Line Error

This interrupt is explained in section 2.4.2.3. For the asynchronous RS232 modem LAD, the type-of-error bits a and b (bits 6 and 7) have the following significance.

ab	Type of Error
00	Framing error
01	Parity error
10	Overflow
11	Parity error *and* overflow

4.4.1.2 Status Change

This interrupt is explained in section 2.4.2.4. For the asynchronous RS232 modem LAD, the status bits have the format:



where the bits, when set, indicate:

u = Both output buffer and output register empty (this bit, when set, causes a status-change interrupt if the i-bit, bit 14, is set in the line control byte)

c = The control-line-in from the modem is on (this control line is usually tied to the modem secondary receive-data line, i.e., reverse channel, and any change of state in this line causes a status change interrupt to be generated if the line is in transmit or receive. Note: no matter what state this line is in when a line is first activitated, i.e., put in transmit and/or receive, no interrupt will be generated)

r = Ring indicator: indicates that the ring indicator line from the modem is on (if the control byte is set up correctly, this bit causes a status-change interrupt). Note: an R true and T false in the control byte.

x = Not used (zeros)

s = The clear-to-send line from the modem is on (does not initiate an interrupt)

o = The carrier-on line from the modem is on (does not initiate an interrupt)

i = The interlock line from the modem is on (does not initiate an interrupt)

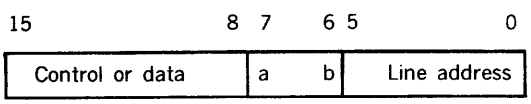
4.4.2 Programming Sequences

This section describes the line-setup and status-reading control sequences used with the asynchronous RS232 modem LAD. General DCM programming sequences are given in section 2.4.4.

4.4.2.1 Line Setup Sequence

This sequence begins with an EXC 0570 instruction. The DCM generates a control interrupt as soon as it is free. The program responds with OAR 070. The DCM loads the specified line and resumes scanning.

The character output by OAR 070 has the format:



where a and b indicate the contents of the more-significant byte as follows:

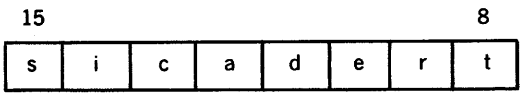
ab	Selected byte
00	Data to output buffer
01	Control
10	Not used
11	Not used

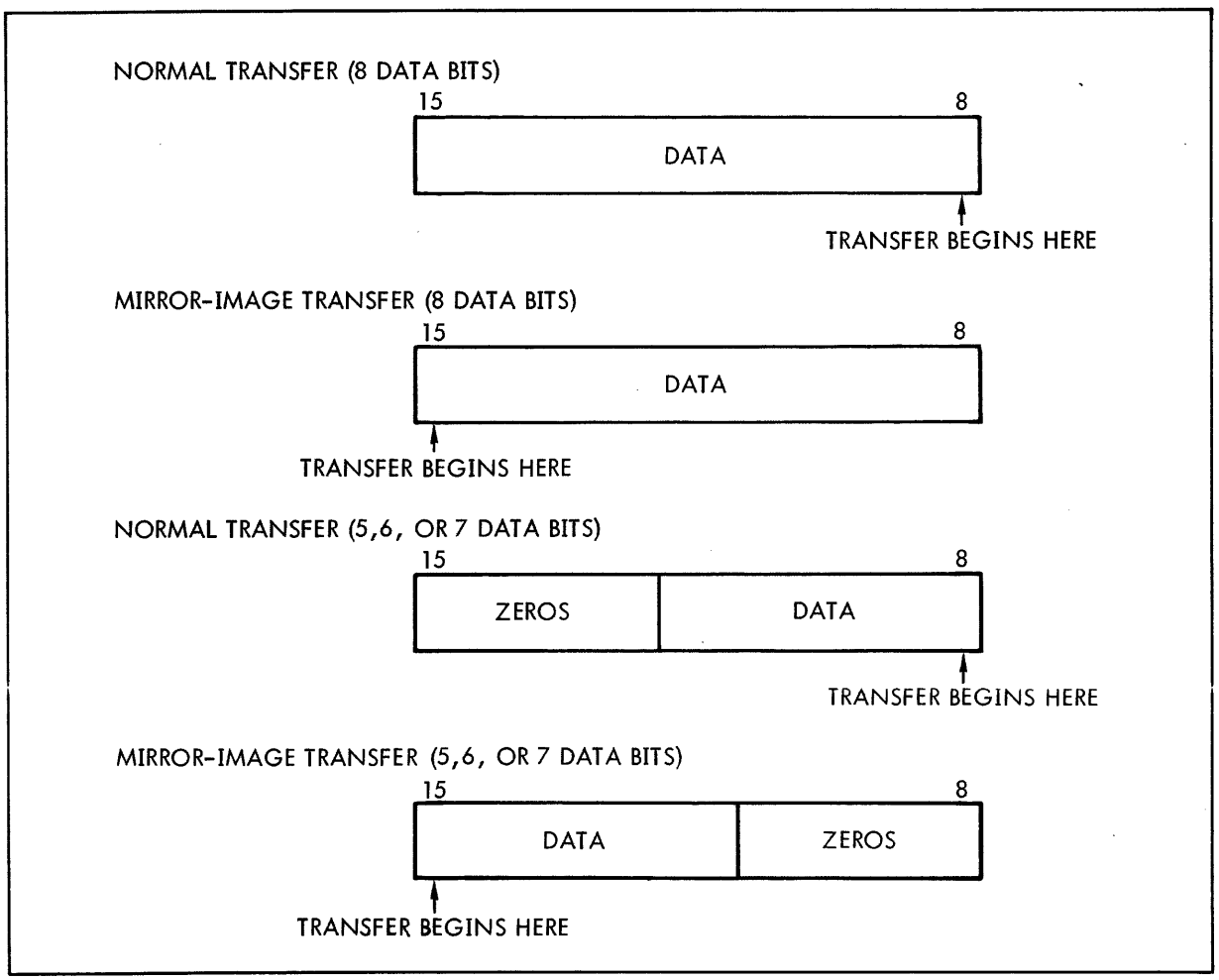
DATA BYTE FORMAT (ab = 00): when ab = 00, the more-significant byte contains data, which can be normal or mirror-image (figure 4-6).

For normal data, the *least-significant* data bit (bit 8) is processed first. For data characters containing fewer than eight bits, the data are *right* justified for transfer.

For mirror-image data (section 4.1.3.5), the *most-significant* data bit (bit 15) is processed first. For data characters containing fewer than eight bits, the data are *left* justified for transfer.

CONTROL BYTE FORMAT (ab = 01): when ab = 01, the more-significant byte contains control information in the format:





VTII-1788

Figure 4-6. Data Byte Formats

where the bits, when set, indicate:

t = Transmission - (puts the line in transmission mode and enables the modem request-to-send line, the line begins making output-data requests as soon as clear-to-send is enabled by the modem, reports line breaks if the modem enables carrier-on. Note: A line must be kept in transmission mode until the last character to be transmitted is in the output register to prevent data loss).

r = Receiving - (puts the line in receiving mode; when carrier-on is enabled by the dataset, the line begins making input-data requests as data are received and reports line errors as they occur)

e = Echo mode - (allows the line to remain in the transmission mode and consequently leaves the modem request-to-send line enabled, but suppresses output-data requests, hardware automatically echos data as received unless this feature is disabled)

d = Data terminal ready - (enables the modem data-terminal-ready line, which allows automatic answer-back and automatic call-termination)

a = Accept ring/transmit break - (when the transmission bit is not set this bit causes a status-input-request interrupt when the ring-indicator line from the modem is enabled. When the transmission bit is set, this bit causes break characters to be transmitted. Note: this bit should be cleared by software upon receipt of the status-change-interrupt ring indicator mode to prevent repetitive interrupts)

c = Control line - (Enables the control line to the modem. Note: This line will usually be tied to the modem's secondary transmitted data line, i.e., the reverse channel)

i = Interrupt - (Causes a status-input request to be generated when both the output buffer and the output register are empty. Note: The software should clear this bit upon receipt of the status-change interrupt to prevent repetitive interrupts. It normally will be enabled when the multiplexor detects the byte count has gone to zero and automatically reloads the control byte). The line must not be in transmission mode unless the echo bit is set. The sequence to set the interrupt bit is:

a. Set up the control table with the buffer size; the control byte stored in the control table resets the transmission bit and sets the interrupt bit set

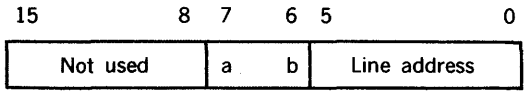
b. Put the line in transmission mode via a control sequence; the line sequentially outputs the data buffer, the byte count goes to zero, the line makes another output-data-request, the multiplexor transfers the control byte to the line, the line goes out of transmission mode, but keeps request-to-send high until the character in the output register is completely transmitted, then a status input request is made indicating the output buffer and the output register are empty

s = Speed selection - (Each line can operate at one of two hardwire selectable speeds; this bit selects speed B, which should be the higher of the two software-selectable speeds. Note: The bit should not be manipulated while a line is transmitting or receiving)

4.4.2.2 Read Line-Status Sequence

This sequence begins with an EXC 0670 instruction. The DCM generates a control interrupt as soon as it is free. The program responds with OAR 070 and CIA 070.

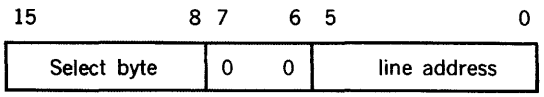
The character output by OAR 070 has the format:



where a and b select one of two possible bytes:

ab	Selected byte
00	Data
01	Line status
10	Not used
11	Not used

The character input by CIA 070 has the format:



DATA BYTE FORMAT (ab = 00): when ab = 00, the selected byte contains data in the same format as in the line-setup sequence (section 4.4.2.1).

LINE-STATUS BYTE FORMAT (ab = 01): when ab = 01, the selected byte contains line-status information in the same format as the status bits in a status-change interrupt (section 4.4.1.2).

4.5 MNEMONICS

Mnemonic	Description
A–	Decoded along with B– to indicate the type of information being transferred to the LAD
AI–	Decoded along with BI– to software to determine the type of line error
AR +	Accept ring
B–	See A–
BI–	See AI–
CCHx–	Control-channel inputs from the modem (can be used for reverse channel)
CCM–	Control channel change-request multiplexed
CCOx +	Control channel inputs from the modem
CCRx +	Control channel change request
CLK–	614,400 Hz clock from the multiplexor
CLx +	Control-line-out to the modem
COx +	Carrier-on from the modem
CST–	Control strobe
CSTx–	Control strobes to the individual control registers
CSx +	Clear-to-send from modem
Cx–	Control channel from control registers
DR +	Data-ready from asynchronous receiver/transmitter
DTRYx +	Data-terminal-ready to modem
E +	Echo bit from control registers
EDTA–	Enable data onto MUX bus
EOB–	Enable output to MUX bus
FE +	Framing error from asynchronous receiver/transmitter
I +	Interrupt on underflow bit from the control registers
IDRS +	Input data-request synchronized with CLK–
IDRSD +	Input data-request into the service request register
ILKx +	Interlock or data-set-ready from modem
IDR–	Input data request to MU
LADA–	LAD address input
LADAE–	Enable LAD address input
LAE1– LAE2– LAE3–	Enable LAD
LERR–	Line error to MU

(continued)

Mnemonic	Description
MB0– MB1–	8-bit multiplexor data bus
OBRKx–	Output break
ODR–	Output data request to MU
OE +	Overrun error from asynchronous receiver/transmitter
PE +	Parity error from asynchronous receiver/transmitter
P.GNDx	Power ground or chassis ground from modem
R +	Receive bit from the control registers
RDA +	Reset data available (data ready)
RDAx–	Reset data available to four asynchronous receiver/transmitters
RDCx +	Receive-data gated with carrier-on
RDSx–	Receive-data from modem
RD0 + through RD7 +	Parallel receive-data bus from asynchronous receiver transmitter
RIDx +	Ring indicator from modem
RIx–	Internal ring indicators
RSIx–	Reset status-input request to control channel edge-detector logic
RSI +	Reset status-input request
RTSx +	Request-to-send to modem
SBRK–	Start break
SBRKx–	Start break signals for the individual lines
SB0 + SB1 + SB2 + SB5 + SB6 + SB7 +	Status bits

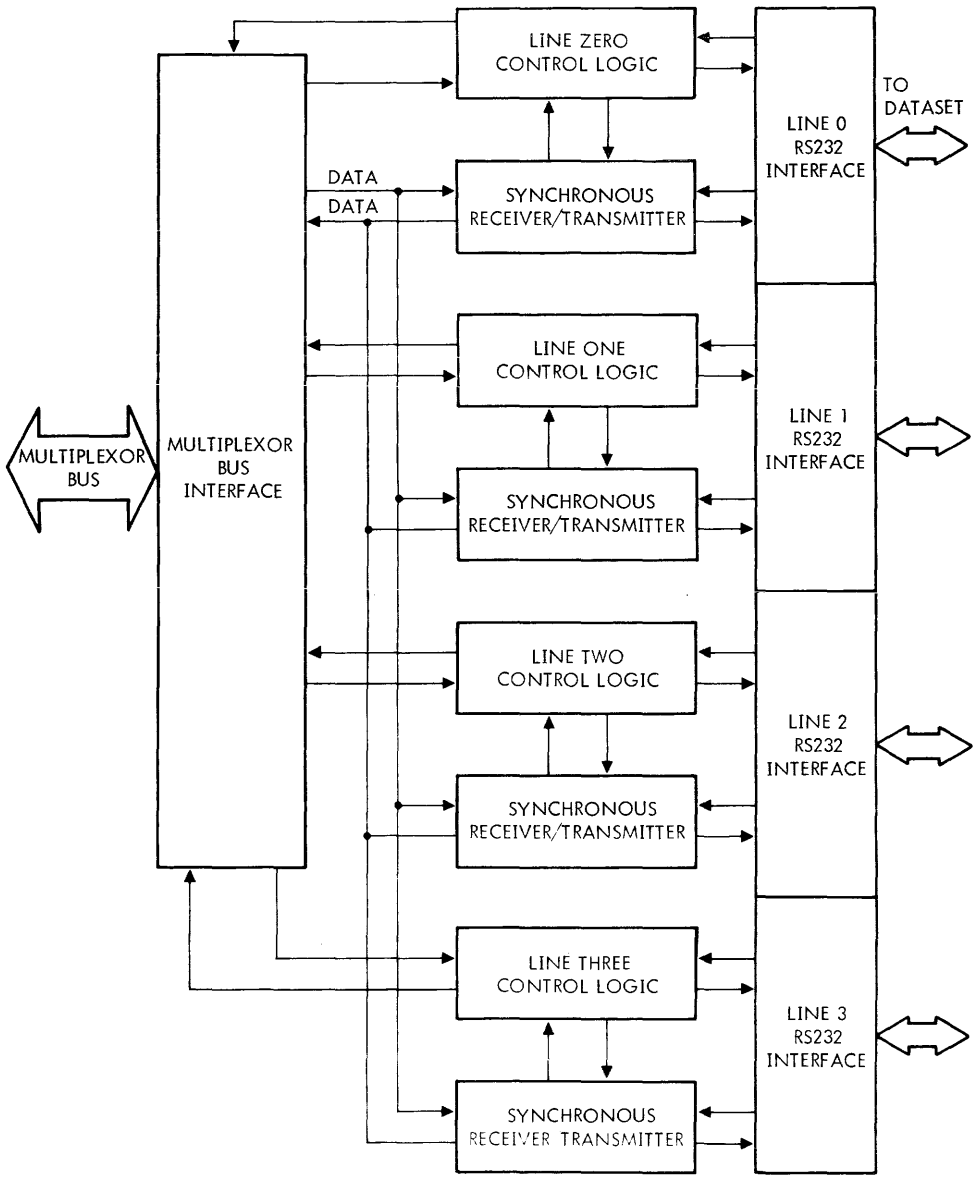
Mnemonic	Description
SCANx–	Enables lines
SCB0 + SCB1 +	2 least-significant bits of the MU scan counter
SIR–	Status input request to MU
SIRSD +	Status input request to service request register
SPDAx–	Clock inputs from the multiplexor bus
SPDx +	Clocks to asynchronous receiver/transmitters
SRST–	System reset
STDD–	Strobe data
STDx–	Strobe data to asynchronous receiver/transmitters
STRBE–	Strobe from MU
Sx +	Speed-selection bit from control register
TBRE +	Transmission buffer empty from asynchronous receiver/transmitter
TDAx–	Transmit data to modem
TDx +	Transmit data from asynchronous receiver/transmitters
TM +	Multiplexed transmission
TOR–	Transmit or receive
TREAx +	Transmission register empty from asynchronous receiver/transmitters
TREM +	Multiplexed transmission-register-empty
Tx +	Transmission bits from control registers
WRITE–	Write line from MU

x = line number.

SECTION 5
SYNCHRONOUS RS232 MODEM LINE-ADAPTER

The LAD (figure 5-1) provides an EIA RS232C interface for the DCM. It can handle four synchronous modems (datasets), specifically the Bell 201, and 208 data sets or their equivalents. Since its interface circuits are compatible with the CCITT V24 specification, the LAD can also interface with European datasets compatible with the Bell 201, and 208.

The LAD interfaces with four full-duplex synchronous data communication lines at a maximum data rate of 20,000 bits per second. It provides character buffering for both input and output, under program control can automatically delete synchronization characters from the incoming data stream and add them to the output data stream when output data is not available to maintain line-synchronization, generates output parity (if any), and checks incoming data for correct parity (if any). The parity can be odd, even, or not used; the character length can be 5, 6, 7, or 8 bits; and separate synchronization character storage is provided for transmission and receipt.



VTII-1789

Figure 5-1. Synchronous RS232 LAD Simplified Block Diagram

Table 5-1 gives the specifications of the synchronous RS232 modem LAD.

Table 5-1. Synchronous RS232 LAD Specifications

Number of lines/board	Four
Transmission type	Serial synchronous
Modem interface	Interface circuits conform to EIA RS232C (Maximum cable length = 50 feet or 15 meters)
Character length	5, 6, 7, or 8 (hardware selectable)
Parity generation and checking	Odd, even, or none (hardware selectable)
Buffering	Each line character buffered on input and output, and there are buffers for an input and an output synchronization character
Maximum bit rate	20,000 bits per second
Bit rate selection	Transmission and receipt clocks provided by the modem
Error reporting	Parity errors, overflow errors, and overrun errors.
Package	One PC board with approximately 75 ICs
Power requirements	+5V dc at 1.2A +12V dc at 100 mA −12V dc at 150 mA
Environment	Temperature 0 to 50 degrees C, relative humidity 0 to 90 percent (without condensation)
Interconnection	Plugs into DCM backplane, interfaces with up to four modems via two 44-pin connectors

5.1 INSTALLATION

The LAD has been inspected and packed to ensure its arrival in good working order. To prevent damage, take reasonable care during unpacking and handling. Check the shipping list to ensure that all equipment has been received. Immediately after unpacking, inspect the equipment for shipping damage. If any is found:

a. Notify the transportation company

b. Notify Varian Data Machines

c. Save all packing material

5.1.1 Physical Description

The synchronous RS232 modem LAD is on one printed-circuit board (figure 5-2) mounted in a specially-wired expansion chassis (figures 1-2 and 1-3).

5.1.2 Interfaces

The LAD interfaces with the MU over the multiplexor bus (figures 1-1 and 2-1), and with the datasets over the RS232 modem interface (section 5.2.7).

5.1.2.1 Multiplexor Bus

The 38-line multiplexor bus connects the LAD with the MU. It is described in section 2.1.1.1.

5.1.2.2 RS232 Modem Interface

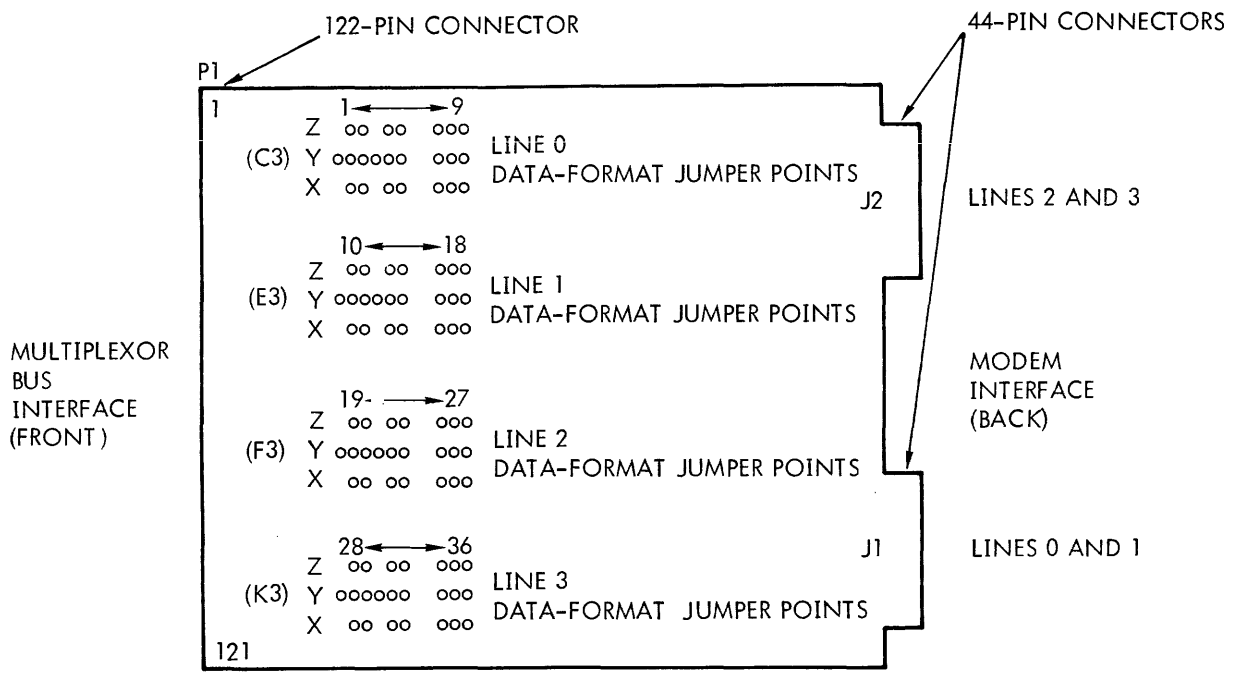
This interface is described in section 5.2.7.

5.1.3 Options

Parity (odd, even, or none), and character length (5, 6, 7, or 8 bits), are set on a line-by-line basis with jumpers (figure 5-2) set according to tables 5-2 and 5-3.

5.1.4 Test Clock Selection

This LAD provides a test clock input from the DCM backplane. This test clock is used when testing the LAD without a modem. The frequency of the clock should be at least 4800 bits per second. Table 5-4 shows the multiplexor bit-rate locations and the standard frequencies. The input pin number for the test clock is 92. (Note: The test clock is used "as is" and not divided down by 16 as in asynchronous LADS.)



VTII-1790

Figure 5-2. Synchronous RS232 Modem LAD

Table 5-2. Data Format Jumpers

Format	Jumpers						Line
8 bits + parity	Y5-Z5	Y3-X6	Y6-X5	Z7-Y7	Z8-Y8	Z9-Y9	
*8 bits no parity	Y5-Y1	Y3-Y4	Y6-X5	Z7-Y7	Z8-Y8	Z9-Y9	
7 bits + parity	Y5-Z5	Y3-Z3	Y6-X5	Z7-Y7	Z8-Y8	X7-Y9	
7 bits no parity	Y5-X2	Y3-Z3	Y6-X5	Z7-Y7	Z8-Y8	X7-Y9	0
6 bits + parity	Y5-Z5	Y3-X6	Y6-Z6	Z7-Y7	X8-Y8	X7-Y9	
6 bits no parity	Y5-X2	Y3-X6	Y6-Z6	Z7-Y7	X8-Y8	X7-Y9	
5 bits + parity	Y5-Z5	Y3-Z3	Y6-Z6	X9-Y7	X8-Y8	X7-Y9	
5 bits no parity	Y5-X2	Y3-Z3	Y6-Z6	X9-Y7	X8-Y8	X7-Y9	
8 bits + parity	Y14-Z14	Y12-X15	Y15-X14	Z16-Y16	Z17-Y17	Z18-Y18	
*8 bits no parity	Y14-Y10	Y12-Y13	Y15-X14	Z16-Y16	Z17-Y17	Z18-Y18	
7 bits + parity	Y14-Z14	Y12-Z12	Y15-X14	Z16-Y16	Z17-Y17	X16-Y18	
7 bits no parity	Y14-X11	Y12-Z12	Y15-X14	Z16-Y16	Z17-Y17	X16-Y18	
6 bits + parity	Y14-Z14	Y12-X15	Y15-Z15	Z16-Y16	X17-Y17	X16-Y18	1
6 bits no parity	Y14-X11	Y12-X15	Y15-Z15	Z16-Y16	X17-Y17	X16-Y18	
5 bits + parity	Y14-Z14	Y12-Z12	Y15-Z15	Y18-Y16	X17-Y17	X16-Y18	
5 bits no parity	Y14-X11	Y12-Z12	Y15-Z15	X18-Y16	X17-Y17	X16-Y18	
8 bits + parity	Y23-Z23	Y21-X24	Y24-X23	Z25-Y25	Z26-Y26	Z27-Y27	
*8 bits no parity	Y23-Y19	Y21-Y22	Y24-X23	Z25-Y25	Z26-Y26	Z27-Y27	
7 bits + parity	Y23-Z23	Y21-Z21	Y24-X23	Z25-Y25	Z26-Y26	X25-Y27	
7 bits no parity	Y23-X20	Y21-Z21	Y24-X23	Z25-Y25	Z26-Y26	X25-Y27	2
6 bits + parity	Y23-Z23	Y21-X24	Y24-Z24	Z25-Y25	X26-Y26	X25-Y27	
6 bits no parity	Y23-X20	Y21-X24	Y24-Z24	Z25-Y25	X26-Y26	X25-Y27	
5 bits + parity	Y23-Z23	Y21-Z21	Y24-Z24	X27-Y25	X26-Y26	X25-Y27	
5 bits no parity	Y23-X20	Y21-Z21	Y24-Z24	X27-Y25	X26-Y26	X25-Y27	
8 bits + parity	Y32-Z32	Y30-X33	Y33-X32	Z23-Y34	Z35-Y35	Z36-Y36	
*8 bits no parity	Y32-Y28	Y30-Y31	Y33-X32	Z34-Y34	Z35-Y35	Z36-Y36	
7 bits + parity	Y32-Z32	Y30-Z30	Y33-X32	Z34-Y34	Z35-Y35	X34-Y36	
7 bits no parity	Y32-X29	Y30-Z30	Y33-Y32	Z34-Y34	Z35-Y35	X34-Y36	
6 bits + parity	Y32-Z32	Y30-X33	Y33-Z33	Z34-Y34	X35-Y35	X34-Y36	
6 bits no parity	Y32-X29	Y30-X33	Y33-Z33	Z34-Y34	X35-Y35	X34-Y36	
5 bits + parity	Y32-Z32	Y30-Z30	Y33-Z33	X36-X34	X35-Y35	X34-Y36	
5 bits no parity	Y32-X29	Y30-Z30	Y33-Z33	X36-Y34	X35-Y35	X34-Y36	

* Standard configuration

Table 5-3. Parity Jumpers

Parity	Jumper	Line
*Odd	Z2-Y2	0
Even	X3-Y2	
*Odd	Y11-Z11	1
Even	X12-Y11	
*Odd	Z20-Y20	2
Even	X21-Y20	
*Odd	Z28-Y29	3
Even	X30-Y29	

* Standard configuration

Table 5-4. Line Bit-Rate Selection

(Test clock input pin on the synchronous LAD is 92)

Multiplexor bus bit-rate locations

Standard Frequencies	Mnemonic	Pin at LAD Slot
153,600 Hz (16 x 9600)	BR0	101
38,400 Hz (16 x 2400)	BR1	102
19,200 Hz (16 x 1200)	BR2	103
*4,800 Hz (16 x 300)	BR3	104
2,400 Hz (16 x 150)	BR4	105
1,760 Hz (16 x 110)	BR5	106

* Normally selected for the synchronous LAD test clock.

5.1.5 Protective Ground (RS232C Circuit AA)

The protective ground lines are not normally wired. If these signals are to be used, jumpers must be added as indicated in table 5-5. These jumpers bring the protective ground lines from the modem out to free pins on the DCM backplane where they should be routed to power ground. They should not be tied to signal or DC ground.

Table 5-5. Protective Ground Jumpers

Line	Jumper	Mnemonic	Backplane Pin
0	M to M	P. GND 0	69
1	N to N	P. GND 1	67
2	R to R	P. GND 2	65
3	S to S	P. GND 3	63

Input data-request (IDRSD+): An input data-request is made if the input-data-request flag from the lines sync-detection logic is true.

Output data-request (ODRSD+): An output data-request is made if the transmission buffer is empty (TBMTM+), the line is in transmission mode (TM+), the accept-ring/stop-output bits is not set (ARM−), and an input status-request is not pending to report an underflow (SIRS3−).

Input status-request (SIRSD+): An input status-request is made under the following conditions:

a. The line is in transmission or receiving mode and a change of state in the control-line-in is detected (SIRS2−).

b. The line is not in transmission mode and a ring occurs (SB2+) and the accept-ring/stop-output bit is true (SIRS1−).

c. The interrupt bit is true (IM+) and a sync-character is transmitted from the sync-character-register (SB0+), (SIRS2−).

5.2 THEORY OF OPERATION

This section explains the operation of the synchronous RS232 modem LAD, with each subsection corresponding to a block (or set of blocks) on the LAD block diagram (figure 5-1). Figure 5-3 is a detailed block diagram. Refer to logic diagram 91B0414 for understanding this theory.

5.2.1 Multiplexor Bus Interface

This logic provides a common interface for the four lines of the LAD to the multiplexor bus (section 2.1.2.2). This interface includes provision for jumpering an internal clock for testing and assigns the LAD an address on the multiplexor bus.

5.2.2 Service Request Logic

The three service-request terms IDRSD+, ODRSD+, and SIRSD+ are clocked into the service-request register on the positive transition of CLK−. The outputs of the register are gated to form a priority structure where the input data-request has the highest priority and the input status-request has the lowest. When an input data-request is made, a line error may be reported (LERR−). If a line error has occurred, the AI−, and BI− identify the type of error.

5.2.3 Control Logic

The control logic for each line (zero through three) is identical. It provides steering logic for data/control transfer to and from the line, storage flip-flops for the control and status bytes, break-transmitting logic, and service-request generation logic (i.e., input or output data requests, etc.).

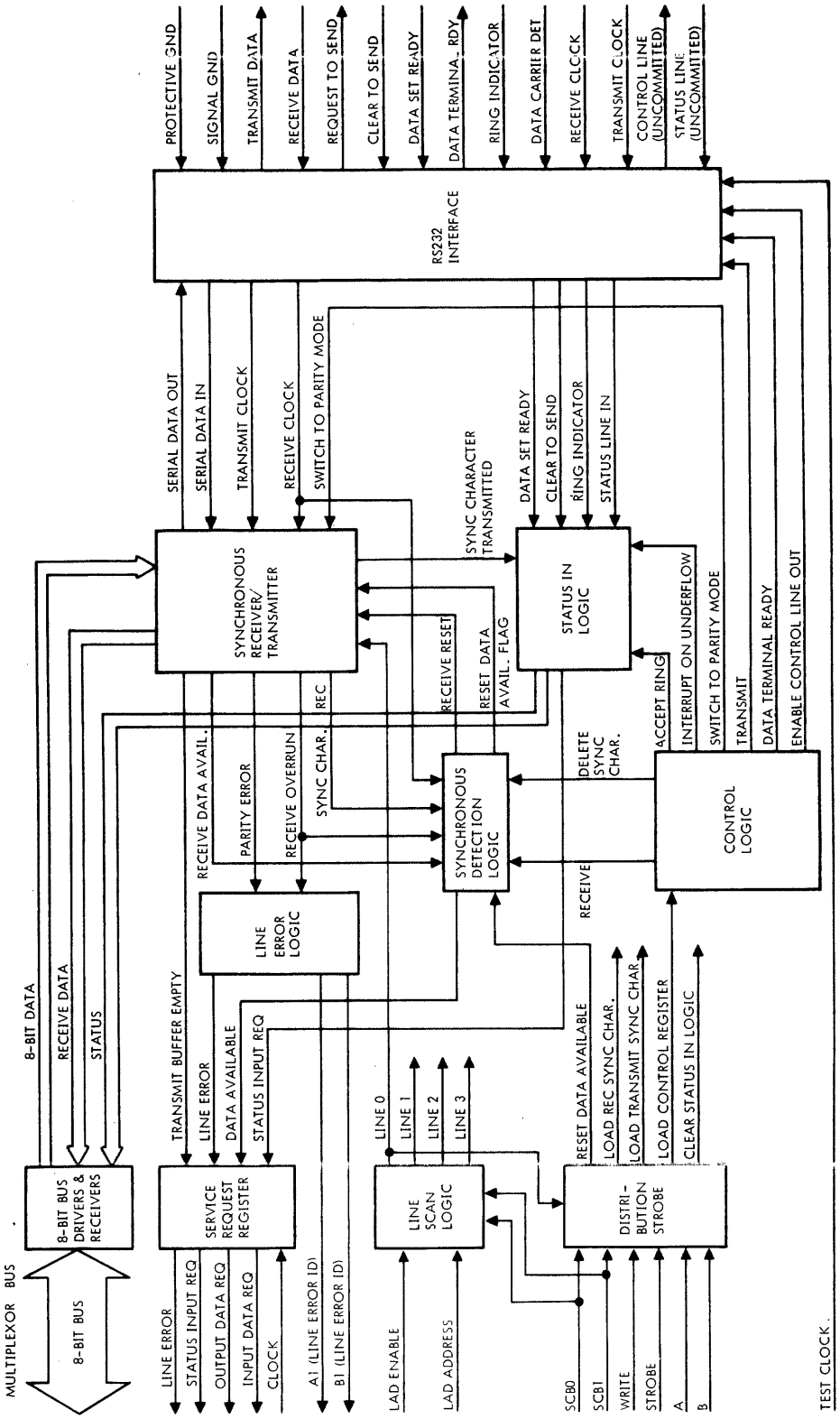
Control of each line is through four eight-bit control registers (one for each line). The register is loaded by software via a control sequence and the outputs enable the various line functions. Each control-bit function is described below.

Delete synchronization: The DS0, DS1, DS2, and DS3 bits are distributed to the appropriate line sync-detection logic. When set, they suppress the input data requests on received characters that match the contents of the receive-sync-character register.

Interrupt on underflow: The I0, I1, I2, and I3 bits are multiplexed together to form IM+. IM+ causes status-input-request whenever an underflow condition exists (sync-character transmitted).

Control-line: C0−, C1−, C2−, and C3− enable and disable individual output control lines.

Accept ring/stop output: The AR0, AR1, AR2, and AR3 bits are multiplexed together to form ARM. ARM enables a



VTI2-408

Figure 5-3. Synchronous RS232 Modem LAD Detailed Block Diagram (One Line Shown)

status-input-request if the line is not in transmission or receiving mode and a ring occurs (SB2+). ARM inhibits output requests if the line is in transmission mode.

Data terminal ready: DTR0–, DTR1–, DTR2–, and DTR3– enable the data-terminal-ready signal to the modems.

Parity enabler: P0, P1, P2, and P3 enable the seven-bit data with parity format. The function is only operational if the line is set up for eight-bit data without parity.

Receiving: R0, R1, R2, and R3 enable receiving mode.

Transmission: T0, T1, T2, and T3 enable transmission mode.

5.2.4 Synchronous Receiver/Transmitter

The synchronous receiver/transmitter is a general purpose, programmable MOS/LSI device for the transmission and receipt of synchronous serial data. The transmitter converts parallel data into serial words that contain both data and parity (if any). The receiver converts serial words into parallel data and verifies proper transmission by checking parity (if any). Both the receiver and the transmitter are character-buffered. The line jumpers can be set up for a word length of 5, 6, 7, or 8 bits; and even, odd, or no parity (section 5.1.3). Table 5-6 describes the pin functions of the device.

The synchronous receiver/transmitter also provides storage for a receiver synchronization-character (RSC) and a transmitter synchronization-character (TSC). The RSC obtains character-synchronization at the beginning of a message and deletes synchronization-characters that are embeded in the incoming message. TSCs are inserted in the outgoing data stream whenever data is not provided fast enough to maintain synchronization on the line.

The input clock frequency for the receiver is set by the modem and the desired receiver sync character (synchronous idle character) is loaded into the receiver sync register. When the receiver-reset input goes from a high to a low, the receiver enters search mode (bit phase). In search mode, the serially received data-bit stream is examined on a bit-by-bit basis until a sync character is found. A sync character is found, by definition, when the contents of the receiver sync-register and the receiver shift-register are identical. When this occurs, the sync-character-received output goes high. This character is then loaded into the receiver buffer-register and the receiver enters character mode. In this mode, each character received is loaded into the receiver buffer-register. The receiver provides flags for receiver-data-available, receiver-overrun, receiver-parity-error, and sync-character-received. Full double buffering eliminates the need for precise external timing by allowing one full character period for received data to be read out.

The input clock frequency for the transmitter is set by the modem and the desired transmitter sync-character is loaded into the transmitter sync-register. Internal logic decides if the character to be transmitted out of the transmitter shift-register is extracted from the transmitter data-register or the transmitter sync-register. The next character transmitted is extracted from the transmitter data-register provided a transmitter-data-strobe pulse occurs during the presently transmitted character. If there is no pulse, the next character transmitted is extracted from the transmitter sync-register and the sync-character-transmitted output is set high. Full double buffering eliminates the need for precise external timing by allowing one full character period to load the next character to be transmitted.

There may be 5, 6, 7, or 8 data bits and odd/even or no parity bit. All inputs and outputs are directly TTL compatible. Tristate receive-data-output levels are provided for the bus structure oriented signals.

Table 5-6. Pin Functions of the Synchronous Receiver/Transmitter

Pin	Symbol	Name	Function
1	VCC	Power supply	+5V supply
2	TBMT	Transmitter buffer empty	This output is at a high-level when the transmitter data buffer register may be loaded with new data
3	TSO	Transmitter serial output	This output serially provides the entire transmitted character. This character is extracted from the transmitter data buffer register provided that a TDS pulse occurs during the presently transmitted character. If TDS is not pulsed, the next transmitted character will be extracted from the transmitter sync register
4	GND	Ground	Ground
5	SCT	Sync character transmitted	This output is high when the character loaded into the transmitted shift register is extracted from the transmitter sync register, indicating that the TDS was not pulsed during the previously transmitted character. This output is low when the character to be transmitted is extracted from the transmitter data buffer register, which can occur only if TDS is pulsed
6	VDD	Power supply	-12V supply
7-14	DB1-DB8	Data bus inputs	This 8-bit bus inputs information into the receiver sync register under control of the RSS strobe, into the transmitter sync register under control of the TSS strobe, and into the transmitter data-buffer register under control of the TDS strobe. The strobes operate independently. Unused bus inputs should be at a high level. The LSB should always be placed on DB1
15	RR	Receiver reset	This input should be pulsed to a high-level after power turn on. This resets the RDA, SCR, ROR, and RPE outputs to a low level. The transition of the RR input from a high to a low puts the receiver in the search mode (bit phase). In the search mode, the serially received data-bit stream is examined on a bit-by-bit basis until a sync character is found. A sync character is found, by definition, when the contents of the receiver sync register and the receiver shift register are identical. When this occurs, the SCR output goes high. This character is then loaded into the receiver buffer register and

Table 5-6. Pin Functions of the Synchronous Receiver/Transmitter *(continued)*

Pin	Symbol	Name	Function
			the receiver is set into the character mode. In this mode, each character received is loaded into the receiver buffer-register
16	RPE	Receiver parity error	This output is high if the received character parity bit does not agree with the selected parity
17	SCR	Sync character received	This output is high whenever a character loaded into the receiver buffer-register is identical to the character in the receiver sync register. This output goes low the next time the receiver buffer-register is loaded with a character that is not a sync character
18	TSS	Transmitter sync strobe	A high-level input strobe loads the character on the DB1-DB8 lines into the transmitter sync register
19	TCP	Transmitter clock	The positive going edge of this clock shifts data out of the transmitter shift register, at a baud rate equal to the TCP clock frequency
20	TDS	Transmitter data-buffer strobe	A high input strobe loads the character on the DB1-DB8 lines into the transmitter data-buffer register
21	RSS	Receiver sync strobe	A high input strobe loads the character on the DB1-DB8 lines into the receiver sync register
22	RSI	Receiver serial input	This input accepts the serial bit-input stream
23	RCP	Receiver clock	The negative-going edge of this clock shifts data into the receiver shift register, at a baud rate equal to the RCP clock frequency
24	RDAR	Receiver data available reset	A high input resets the RDA output to a low
25	RDE	Receiver data enable	A high input enables the outputs (RD8-RD1) of the receiver buffer-register
26	RDA	Receiver data available	This output is high when an entire character has been received and transferred into the receiver buffer-register
27	ROR	Receiver overrun	This output is high if the previously received character is not read (RDA not reset) before the present character is transferred into the receiver buffer register

Table 5-6. Pin Functions of the Synchronous Receiver/Transmitter *(continued)*

Pin	Symbol	Name	Function
28 through 35	RD8 through RD1	Receiver data output	These are 8 tristate data outputs enabled by RDE. Unused data output lines, as selected by NDB1 and NDB2, have a low output, and received characters are right justified, i.e., the LSB always appears on the RD1 output
36,38	NDB2, NDB1	Number of data bits/character	These 2 inputs are internally decoded to select either 5, 6, 7, or 8 data bits/character as follows: NDB2 NDB1 Data bits/character L L 5 L H 6 H L 7 H H 8
37	NPB	No parity bit	A high input prevents the parity bit from being transmitted. In addition, it is necessary that the received character contain no parity bit. Also, the RPE output is forced low
39	CS	Control strobe	A high input enters the control bits (NDB1, NDB2, POE, and NPB) into the control bits register. This line may be strobed or hardwired high
40	POE	Select odd/even parity	The logic levels on this input and on NPB, determine the parity mode for both the receiver and transmitter, as follows: NPB POE MODE L L Odd parity L H Even parity H L No parity

Data/status selection: This LAD output to the multiplexor eight-bit bus has two sources: the eight-bit receiver-data bus (RD0+ through RD7+) from the four asynchronous receiver/transmitter devices, and the six status outputs (SB0+ and SB5 through SB7+). Selection of these outputs is accomplished with two quad-two-line-to-one-line data-selector devices. The selection term is EDTA–, and the data-selector devices are enabled by EOB–. These terms are formed with half of a dual two-line-to-four line decoder, which is enabled by LAD enabler (LAE1–) when the multiplexor is reading (WRITE+ false). EOB– is true if the data-selection terms A– and B– are both false, or if A– and B– is true. EDTA– is true only when A– and B– are both false. EDTA– true gates the receiver data bus out.

LAD Enabling Logic: The LAD address input (LADA–) and the LAD address-enabler input (LADE–) are ANDed to form the LAD enabling terms LAE1–, LAE2–, and LAE3–.

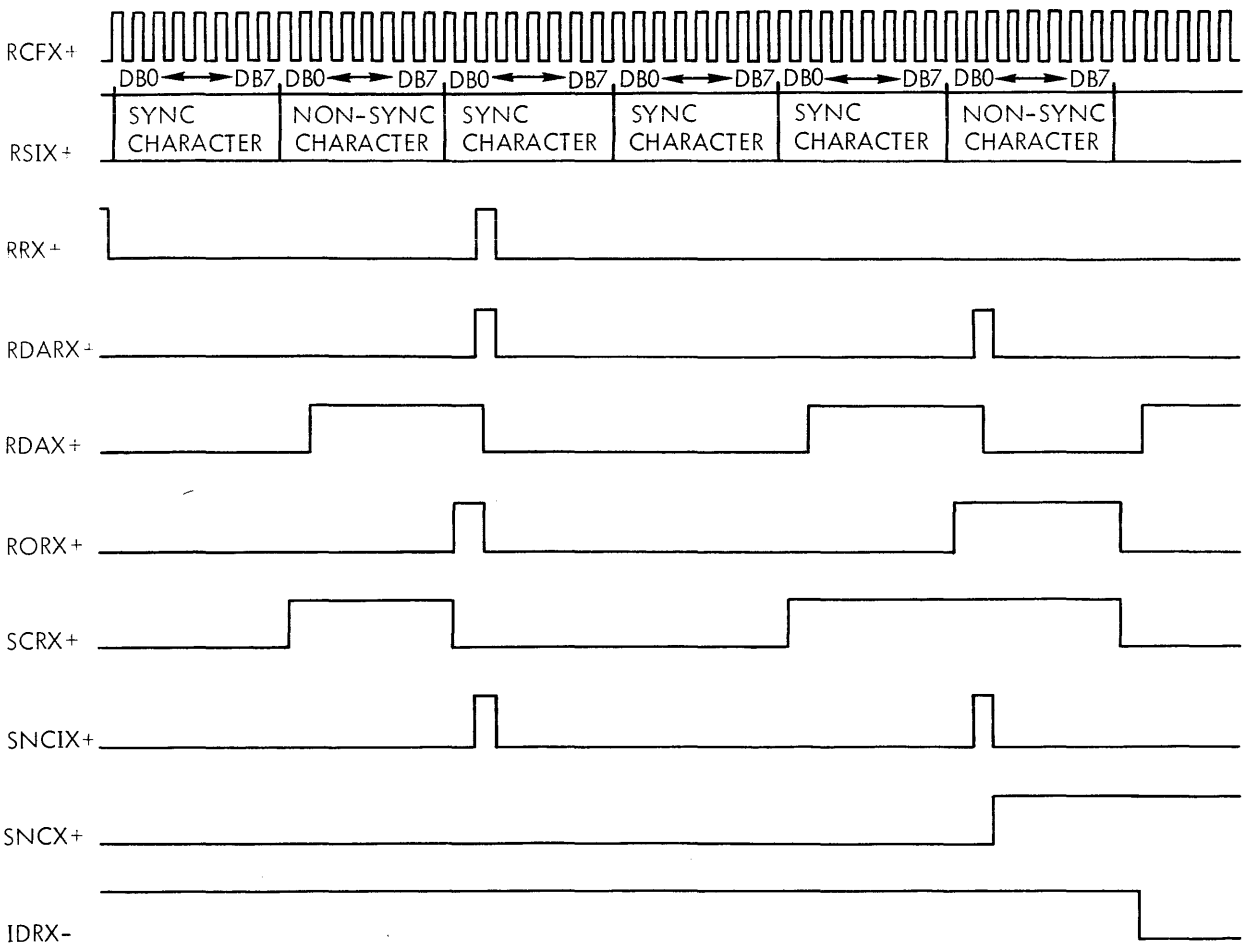
5.2.5 Sync-Detection Logic

The sync-detection logic section for each line is identical. Figure 5-4 illustrates the timing for this logic. The two sync-detection flip-flops (SNC1 and SNC) are clocked by the receive clock (RCP). The detection logic takes no action until the receiver/transmitter device reports an overrun (RORA), which signifies that the device has received one sync character, gone to the character mode, and received

one more character. At the detection of the overrun, the detection logic sets the SNC1 flip-flop on the next negative clock transition. One clock period later, the SNC flip-flop sets if the second character was a sync character (SCRA true) and enables the input data request signal for succeeding characters. If the second character was not a sync character, the receiver is reset (SNF0-) and begins, once again, checking for synchronization in bit mode. Note that the SNC1 flip-flop causes the reset data available signal (RDAR+) to go true so that the first two sync detection characters are not transferred to the computer. If the delete-sync bit is set (DS+), RDAR goes true for every sync-character received and IDR is inhibited. RDS causes receiver-data-available to be reset in normal operation after the character is transferred to the computer.

5.2.6 Control-Channel Edge-Detectors

This logic consists of four separate identical sections (one for each line). Each section contains a flip-flop to store the state of the control line, and an exclusive NOR gate compare the control channel input state with the flip-flop state. If a difference is detected, a high signal (CCM+ - multiplexed term of CCR0+, CCR1+, CCR2+, and CCR3+) is sent to the status-input-request logic. A strobe causes the flip-flop to go to the state of the control-channel input after the status-input request has been acknowledged.



VTII-1914

Figure 5-4. Sync Detection Logic Timing Diagram

5.2.7 RS232 Modem Interface

This interface conforms to the RS232C and CCITT V24 standards and provides for the following signals (pin numbers are indicated in table 5-7):

AA– Protective ground

AB– Signal ground

BA– Transmitted data

BB– Received data

CA– Request to send

CB– Clear to send

CC– Dataset ready

CD– Data terminal ready

CE– Ring indicator

CF– Data carrier detector

DB– Transmitter clock

DD– Receiver clock

Table 5-7. RS232 Signal Locations

LAD Line Number	RS232 Signal	DCM Mnemonic	Pin
0	AA	PGND0	J1-24
0	BA	TDA0–	J1-31
0	BB	RDS0–	J1-23
0	CA	RTS0 +	J1-27
0	CB	CS0 +	J1-37
0	CC	ILK0 +	J1-33
0	AB	GRD	J1-20,22
0	CF	CO0 +	J1-25
0	CD	DTRY0 +	J1-29
0	CE	RID0 +	J1-35
0	-	CL0 +	J1-21
0	-	CCO0 +	J1-18
0		– 12V	J1-36
0	-	+ 12V	J1-40
0	DB	TSET0 +	J1-39
0	DD	RSET0 +	J1-41
1	AA	PGND1	J1-10
1	BA	TDA1–	J1-5
1	BB	RDS1–	J1-9
1	CA	RTS1 +	J1-3
1	CB	CS1 +	J1-19
1	CC	ILK1 +	J1-11
1	AB	GRD	J1-14,12
1	CF	CO1 +	J1-7
1	CD	DTRY1 +	J1-1
1	CE	RID1 +	J1-13
1	-	CL1 +	J1-6

Table 5-7. RS232 Signal Locations *(continued)*

LAD Line Number	RS232 Signal	DCM Mnemonic	Pin
1	-	CCO1 +	J1-8
1	-	- 12V	J1-4
1	-	+ 12V	J1-2
1	DB	TSET1 +	J1-15
1	DD	RSET1 +	J1-17
2	AA	PGND2	J2-24
2	BA	TDA2-	J2-31
2	BB	RDS2-	J2-23
2	CA	RTS2 +	J2-27
2	CB	CS2 +	J2-37
2	CC	ILK2 +	J2-33
2	AB	GRD	J2-20,22
2	CF	CO2 +	J2-25
2	CD	DTRY2 +	J2-29
2	CE	RID2 +	J2-35
2	-	CC2 +	J2-21
2	-	CCO2 +	J2-18
2	-	- 12V	J2-36
2	-	+ 12V	J2-40
2	DB	TSET2 +	J2-39
2	DD	RSET2 +	J2-41
3	AA	PGND3	J3-10
3	BA	TDA3-	J3-5
3	BB	RDS3-	J3-9
3	CA	RTS3 +	J3-3
3	CB	CS3 +	J3-19
3	CC	ILK3	J3-11
3	AB	GRD	J3-14,12
3	CF	CO3 +	J3-7
3	CD	DTRY3 +	J3-1
3	CE	RID3 +	J3-13
3	-	CL3 +	J3-6
3	-	CCO3 +	J3-8
3	-	- 12V	J3-4
3	-	+ 12V	J3-2
3	DB	TSET3 +	J2-15
3	DD	RSET3 +	J2-17

The LAD also has two general-purpose lines that can be used to indicate that the reverse channel is on (" input control line") and to turn on the reverse channel (" output control line"). Note that the LAD does *not* provide data communications capability over the reverse channel.

These general-purpose lines can also be used for input and output speed selection (RS232C circuits CH- and CI-).

The interface provides + 12V dc and - 12V dc through 1K resistors that can be used to hardwire certain leads on for testing or for non-standard datasets. Thus, to use this LAD without a dataset (direct connection): tie CB-, CC-, and CF- to DB- and connect the receive-and-transmit clock DB, DD (external source).

5.2.8 Strobe Logic

There are four strobes to transfer data to the LAD:

A	B	Mnemonic	Description
0	0	TDS-	Transfer data to the transmitter
0	1	CST-	Control strobe (loads control register)
1	0	TSS-	Load transmit sync-character register
1	1	RSS-	Load receive sync-character register

The strobes are formed with half of a dual two-to-four decoder. The device is enabled when WRITE is true and gates the strobe to the appropriate output as determined by the states of A and B.

There are two strobes used to clear functions in the line adapter: RSI, reset input status-requests, clears the control-channel edge-detector logic. RSI occurs for every strobe when a line is not in transmission or receiving mode to ensure that the edge detectors are clear when a line is enabled. RSI is enabled during a input status-request if a control-channel change is being reported. RDS, reset data-available strobe, clears the receiver data-available flag. RDS is enabled during an input data-request. Note that it is disabled if an overrun occurs during the sequence (DRDA–).

The strobe signals are distributed to the individual lines via dual two-to-four line decoders enabled by the LAD enabler. The inputs are distributed to the line specified by SCB0 and SCB1.

5.3 MAINTENANCE

The following equipment is required to maintain the LAD:

a. Varian 620-series or V70 series computer with at least 8K of memory

b. Data communications multiplexor

c. An I/O expansion chassis for the DCM backplane

d. An I/O expansion power supply

e. One card extender (44 D 0015 or 44 D 0540 000)

f. Two test connectors (burndy edge connectors 57 A 0036 000) wired as follows:

Line 0	1.	Pin 29 to pin 35	Data-terminal-ready to ring-indicator
		Pin 35 to pin 33	Ring-indicator to dataset-ready
		Pin 29 to pin 25	Data-terminal-ready to carrier-on
	2.	Pin 27 to pin 37	Request-to-send to clear-to-send
	3.	Pin 31 to pin 23	Transmit-data to receive-data
	4.	Pin 21 to pin 18	Control-out to control-in
Line 1	5.	Pin 1 to pin 13	Data-terminal-ready to ring-indicator
		Pin 13 to pin 11	Ring-indicator to dataset-ready
		Pin 1 to pin 7	Data-terminal-ready to carrier-on
	6.	Pin 3 to pin 19	Request-to-send to clear-to-send
	7.	Pin 5 to pin 9	Transmit-data to receive-data
	8.	Pin 6 to pin 8	Control-out to control-in

The following jumpers tie the test clock to the receive/transmit clock inputs.

Pin 38 to pin 39
Pin 39 to pin 41
Pin 38 to pin 17
Pin 17 to pin 15

g. Software tapes:

MAINTAIN II Test Executive	92 U 0107-001
DCM Test Program	92 U 0106-009B

h. One oscilloscope, Tektronix 547 or equivalent

The following system configuration features must be checked before testing:

a. Data format (the LAD is set up in the standard configuration for initial testing and after these tests are complete, reset to shipping configuration, and all applicable tests rerun).

b. Test clock (verify that a test clock of at least 4800 bit/sec is wired to the LAD slot).

5.3.1 Functional Tests

The following tests use portions of the DCM test program. The specific test required is called out in each section. Data pattern 2 should be used for all tests unless otherwise specified.

Input/output test: Verify that each line can transmit and receive a binary data pattern without errors, using test 1. While running test 1, verify that the serial data out of each line (TDA0– through TDA3–) are oscillating in the range +9 to +12V and –9 to –12V.

Parity-error detection test: Set up the LAD for 8 bit data - no parity - parity mode odd for this test (standard configuration). This test is performed on one line at at a time. Run test 14 continuously and momentarily ground the serial-data output pin. This should cause a parity error that will be reported by the test program.

Overrun-error reporting test: Verify that each line can report an overrun condition by running test 5.

Status-input request test: Verify that each line will make a status input request if a change-of-state of the control-line-in, a ring indication, or an underflow condition (output buffer and register empty) is detected. Test 5 checks all of these.

Modem status lines: Use test 6 to verify that the three modem status lines can be read (interlock, carrier-on, and clear-to-send). Ensure that the clear-to-send status-bit is set if the line is put in transmission mode and reset if the line is not. The interlock status-bit and the carrier-on status-bit should be under control of the data-terminal-ready bit in the line-control byte.

Synchronization test: Verify that each line can obtain synchronization with any valid synchronization character. Test 11 accomplishes this by sequentially transmitting a group of sync-characters that tests all bits. (Test 11 will fail on a line set for five-bit data with odd parity and should be deleted for that configuration.)

Resynchronization test: Verify that each line can be resynchronized in the middle of a message by loading the line receive-sync-character register with a new sync-character. Test 12 accomplishes this.

Transmission sync test: Verify that each line can transmit any bit pattern from its transmit-sync-character register by running test number 13.

Auto parity test: Verify that each line set up for 8-bit data and no parity can be switched under software control to 7-bit data plus parity. Test 14 verifies this.

Burn-in test: Run test 77 for at least 2 passes. (Bypass this test for lines not set up for 8-bit data and no parity.)

5.3.2 Other Problems

To eliminate problems not solved by the above tests, use the program below:

Address (Octal)	Code (Octal)	Instruction	Remarks
000200	002000	JMPM	Call LCB setup
000201	000215		
000202 *	100070	EXC 070	Initialize
000203 *	100270	EXC 270	Enable interrupts
000204 *	100570	EXC 570	Request control write
000205	005000	NOP	
000206	001000	JMP	Wait for interrupt
000207	000205		
		LCB Setup	
000215	000000		
000216	006050	STAI	Store A
000217	000000		
000220	006010	LDAI	Load A
000221 **	107770		Input byte-count word
000222	057500	STA	Store A in LCB
000223	006010	LDAI	Load A
000224 **	000600		Input buffer-address word
000225	057501	STA	Store A in LCB
000226	006010	LDAI	Load A
000227 **	107770		Output byte-count word
000230	057502	STA	Store A in LCB
000231	006010	LDAI	Load A
000232 **	000650		Output buffer-address word
000233	057503	STA	Store A
000234	010217	LDA	Load A (restore)
000235	001000	JMP	Jump back
000236	100215		

* Change these instructions to reflect the correct device address if a nonstandard address is used.

** Enter the byte counts and buffer addresses desired.

Address (Octal)	Code (Octal)	Instruction	Remarks
		Line Setup	
000240	000000		
000241	006050	STAI	Save A
000242	000000		
000243	010320	LDA	Load A with Control Word
000244*	103170	OAR	Output to DCM
000245	006010	LDAI	
000246	000001		
000247	001010	JAZ	
000250	000300		
000251	006010	LDAI	
000252	017504		
000253	050243	STA	
000254	005001	TZA	
000255	050246	STA	
000256	001000	JMP	
000257	000203		
000300	006010	LDAI	
000301	010320		
000302	050243	STA	
000303	006010	LDAI	
000304	000001		
000305	050246	STA	
000306	010242	LDA	Restore A
000307	100270	EXC	Enable Interrupts
000310	001000	JMP	
000311	100240		Return
000320**	113300		Receive Synchronous Character Word
00650	113226		Synchronous Characters

* Change to reflect correct Device Address

** Change to reflect correct Synchronous Character and line number.

		Input or Output Byte-Count Zero	
000260	000000		
000261	100070	EXC 070	Initialize
000262	001400	J SS3	Jump to
000263	006221		Test executive
000264	001000	JMP	Return to
000265	000200		Start
		Interrupt Traps	
000060	002000	JMPM	
000061	000260		
000062	002000	JMPM	
000063	000260		
000064	001000	JMP	
000065	000270		
000066	001000	JMP	
000067	000271		

Address (Octal)	Code (Octal)	Instruction	Remarks
000070	001000	JMP	
000071	000272		
000072	002000	JMPM	
000073	000240		
		Interrupt Halts	
000270	000004		Line error interrupt
000271	000006		Status change interrupt
000272	000001		Control-character-detected interrupt

* Change these instructions to reflect the correct device address if a nonstandard address is used.

LCB Address

The LCB addresses for the line under test are entered here.

Address (Octal)	Code (Octal	Instruction	Remarks
000500	017000		Input block-length address
000501	017001		Input buffer-address address
000502	017004		Output block-length address
000503	017005		Output buffer-address address
000504	017006		Control-word address
		Line Control Block (Line zero shown - base address 017000)	
017000	107770		Input block-length
017001	000600		input buffer-address
017002	177777		Control characters
017003	000000		Spare
017004	107770		Output block-length
017005	000650		Output buffer-address
017006	005500		Line control-word (includes line address)
017007	000000		Spare
		Output Buffer	
000650	125252		
000651	125252		
000652	125252		
000653	125252		
		Input Buffer	
000600	000000		
000601	000000		
000602	000000		
000603	000000		

5.4 PROGRAMMING

This section describes only those aspects of programming peculiar to the synchronous RS232 modem LAD. General DCM programming is discussed in section 2.4.

5.4.1 Interrupts

This section explains the use of the line-error interrupt type-of-error bits and the status-change-error interrupt status bits with regard to the synchronous RS232 modem LAD. The remaining interrupt information is in section 2.4.2.

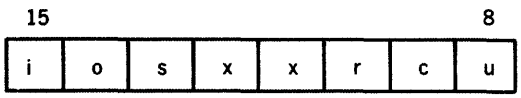
5.4.1.1 Line Error

This interrupt is explained in section 2.4.2.3. For synchronous RS232 modem LAD, the type-of-error bits a and b (bit 6 and 7) have the following significance.

ab	Type of Error
00	Not used
01	Parity error
10	Overflow
11	Parity error *and* overflow

5.4.1.2 Status Change

This interrupt is explained in section 2.4.2.4. For the synchronous RS232 modem LAD, the status bits have the format:



where the bits, when set, indicate:

u = Underflow: synchronization character transmitted (this bit, when set, causes a status-change interrupt if the i-bit bit 14, is set in the line control byte)

c = The control-line-in from the modem is on (this control line is usually tied to the modem secondary receive-data line, i.e. reverse channel. Any change of state in this line causes a status change interrupt to be generated if the line is in transmit or receive)

r = Ring indicator: indicates that the ring indicator line from the modem is on (if the control byte is set up correctly, this bit causes a status-change interrupt).

x = Not used (zeros)

s = The clear-to-send line from the modem is on (does not initiate an interrupt)

o = The carrier-on line from the modem is on (does not initiate an interrupt)

i = The interlock line from the modem is on (does not initiate an interrupt)

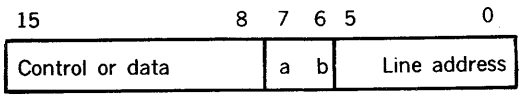
5.4.2 Programming Sequences

This section describes the line-setup and status-reading control sequences used with the synchronous RS232 modem LAD. General DCM programming sequences are given in section 2.4.4.

5.4.2.1 Line Setup Sequence

This sequence begins with an EXC 0570 instruction. The DCM generates a control interrupt as soon as it is free. The program responds with OAR 070. The DCM loads the specified line and resumes scanning.

The character output by OAR 070 has the format:



where a and b indicate the contents of the more-significant byte as follows:

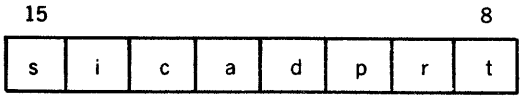
ab	
00	Data to output buffer
01	Control
10	Transmitter synchronization character (TSC)
11	Receiver synchronization character (RSC)

DATA BYTE FORMAT (ab = 00): when ab = 00, the more-significant byte contains data. For data characters containing fewer than eight bits, the data are right-justified (bit 8 is the least-significant data bit) and the unused bits are zeros.

TSC FORMAT (ab = 10): when ab = 10, the more-significant byte contains the TSC that is transmitted by the LAD whenever a line is about to go into an underflow condition to maintain line synchronization. The TSC is right-justified in the control-byte field (unused bits = 0).

RSC FORMAT (ab = 11): when ab = 11, the more-significant byte contains the RSC used to obtain initial line synchronization in receiving mode (the LAD requires at least two consecutive RSCs to obtain line synchronization). If the s bit is set in the line control-byte, every input character that matches the RSC is deleted from the input data stream. The RSC is right-justified within the control-byte field (unused bits = 0). To drop synchronization or to resynchronize a line, the program transfers a new RSC to the line.

CONTROL BYTE FORMAT (ab = 01): when ab = 01, the more-significant byte contains control information in the format:



where the bits, when set, indicate

t = Transmission - (puts the line in transmission mode and enables the modem request-to-send line, the line begins making output-data requests as soon as clear-to-send is enabled by the modem. Note: A line must be kept in transmission mode until the last character to be transmitted has been shifted out of the output register)

r = Receiving - (puts the line in receiving mode; when carrier-on is enabled by the dataset, the line begins comparing input data with the RSC and goes into character mode (i.e., begins transferring data to the computer) as soon as two consecutive synchronization characters are detected)

p = Parity mode - (enables software to switch from 8-data bits and no parity to 7-data bits - plus parity. Note: If the line is not set up for an 8-bit word with no parity, this bit is zero)

d = Data terminal ready - (enables the modem data-terminal-ready line, which allows automatic answer-back and automatic call-termination)

a = Accept-ring/stop-output - (generates a status-input-request interrupt interrupt when the ring-indicator line from the modem is enabled. Note: This bit should be cleared by software upon receipt of the status-change interrupt to prevent repetitive interrupts. Normally, the t, r, and d bits are not when the a bit is set. Upon receipt of the status change interrupt, the r bit and the d bit are set, and the a bit is reset. If a bit is set when the line is in transmission mode, all output data requests are suppressed and the line outputs the contents of the sync character register continuously).

c = Control line - (enables the control line to the modem. Note: This line will usually be tied to the modem's secondary transmitted data line, i.e., the reverse channel)

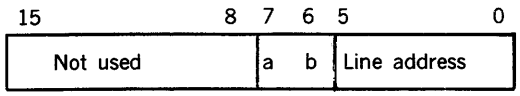
i = Interrupt on underflow - (generates a status-change interrupt when a sync character is transmitted to maintain character synchronization because data are not provided at a high enough rate)

s = Delete sync characters - (no character received that matches the the contents of the RCS. Storage register is transferred to the computer)

5.4.2.2 Read Line-Status Sequence

This sequence begins with an EXC 0670 instruction. The DCM generates a control interrupt as soon as it is free. The program responds with OAR 070 and CIA 070.

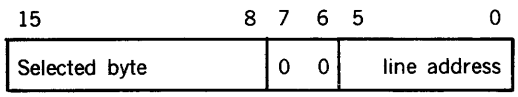
The character output by OAR 070 has the format:



where a and b select one of two possible bytes;

ab	Selected byte
00	Data
01	Line status
10	Not used
11	Not used

The character input by CIA 070 has the format:



DATA BYTE FORMAT (ab = 00): when ab = 00, the selected byte contains data in the same format as in the line-setup sequence (section 5.4.2.1).

LINE-STATUS BYTE FORMAT (ab = 01): when ab = 01, the selected byte contains line-status information in the same format as the status bits in a status-change interrupt (section 5.4.1.2).

5.5 MNEMONICS

Mnemonic	Description
A–	Decoded with B– to indicate the type of information being transferred to the LAD
A1–	Decoded with B1– by software to determine the type of line error
ARx +	Accept-ring/stop output bits from the control registers
ARM +	Multiplexed term of the four accept-ring/stop output bits
B–	See A–
B1–	See A1–
Cx–	Control channel from the control registers
CCHx–	Control channel inputs from the modem (can be used for reverse channel)
CCM +	Control channel change request multiplexed
CCRx–	Control channel change request
CLK–	614,400 Hz clock from MU
COx +	Carrier-on from modems
CSx +	Clear-to-send from modems
CST–	Control strobe
CSTx–	Control strobes to individual control registers
DSx +	Delete sync bits from control registers
DTRx–	Data-terminal-ready bits from control registers
EDTA–	Enable-data onto MUX bus
EOB–	Enable output to MUX bus
Ix +	Enable interrupt-on-underflow bit from control registers
IDR–	Input data request to the multiplexor
IDRx–	Line input-data requests
IDRS +	Input-data request synchronized with CLK–
IDRSD +	Input-data request into the service request register
ILKx +	Interlock or data-set-ready from modem
IM +	Multiplexed term of the four interrupt bits
LADA–	LAD address
LADAE–	Enable LAD address
LAE1– LAE2– LAE3–	Enable LADs
LERR–	Line error signal to the multiplexor
LERRS–	Line error synchronized with CLK–

SYNCHRONOUS RS232 MODEM LINE-ADAPTER

Mnemonic	Description
MB0– through MB7–	8-bit multiplexor data bus
ODR–	Output data request to the multiplexor
ODRSD +	Output data request D input to the service request register
Px +	Parity-mode enabler bits from the control registers
Rx +	Receiver bits from the control registers
RCIx–	Reset control channel edge detectors
RCPx +	Receiver clocks
RD0 + through RD7 +	Parallel receive data bus from asynchronous receiver/transmitter devices
RDAx +	Receiver data-available flags
RDARx +	Reset data-available
RDEx +	Tristate receiver-data output enabler
RDS +	Receiver data strobe
RDSx–	Receiver data strobes
RIx–	Internal ring-indicators
RORx + through RORxA +	Receiver overrun
RPEx +	Receiver parity-error flags
RRx +	Reset receive
RSI +	Reset input status
RSIx +	Receiver serial inputs
RSS–	Receiver sync-character strobe
RSSx +	Receiver sync-character strobes
SB0 + SB1 + SB2 + SB5 + SB6 + SB7 +	Status bits
SCB0 + SCB1 +	Two least-significant bits of MU
SCRx +	Sync character received
SCTAx +	Sync-character-transmitted flag from synchronous transmitter/ receiver devices
SIR–	Status input request to the MU
SIRS3–	Underflow
SIRSD +	Status input request to the service request register
SRST–	System reset
STRBE–	Strobe from MU
Tx +	Transmission bits from control registers
TBMTx +	Transmit buffer-empty flag from synchronous transmitter/receiver devices
TBMTM +	Transmit buffer empty multiplexed signal
TCLK +	Test clock
TCPx +	Transmitter clocks
TDS–	Transmitter data strobe
TDSx +	Transmitter data strobes
TM +	Transmission multiplexed
TOR–	Transmit or receive
TSOx +	Transmitter serial output
TSS–	Transmitter sync-character strobe
TSSx +	Transmitter sync-character strobes
WRITE–	Write line from MU

x = line number.

SECTION 6
BINARY SYNCHRONOUS COMMUNICATION LINE-ADAPTER

The LAD (figure 6-1) provides binary synchronous communications (BSC) capability for the Varian Data Machines 620 and V70 software. It can handle one BSC line at data rates up to 50,000 bits per second. The LAD has a standard EIA RS232C interface that is capable of handling data rates up to 20,000 bits per second and an optional high-speed interface for line rates up to 50,000 bits per second. It is designed to handle Bell 201 and 208 type modems. The optional high-speed interface is designed to interface to the Bell 300 series modems. The LAD does not support 6-bit transcode and related functions (as per IBM document A27-3004-1).

The LAD generates the BSC block-check-character (BCC) on output and detects BCC errors on input. In addition, the LAD detects all BSC data-link control characters and sequences.

The BSC LAD operates in three modes: EBCDIC, USASCII without transparent capability, and USASCII with transparent capability. In the EBCDIC mode, the LAD uses the EBCDIC code for control character detection, uses IBM CRC-16 for the block check character, and is capable of transparent and non-transparent operation. In the USASCII without transparency mode, the LAD uses the USASCII code, generates and checks VRC on each character, uses LRC-8 for the block check character, and operates only in the non-transparent mode. The third mode differs from the second in that it is capable of transparent operation, the VRC generation is suspended in transparent mode, VRC checking is not done on input and IBM CRC-16 is used for the block check character.

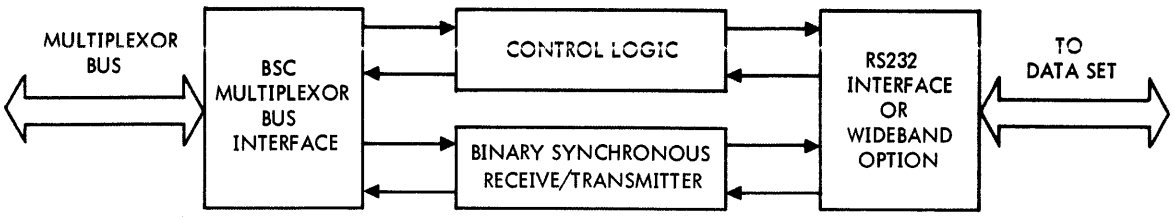
NOTE

The BSC multiplexor is not described in this section since it is functionally the same as the multiplexor unit described in section 2.

Table 6-1 gives the specifications of the LAD.

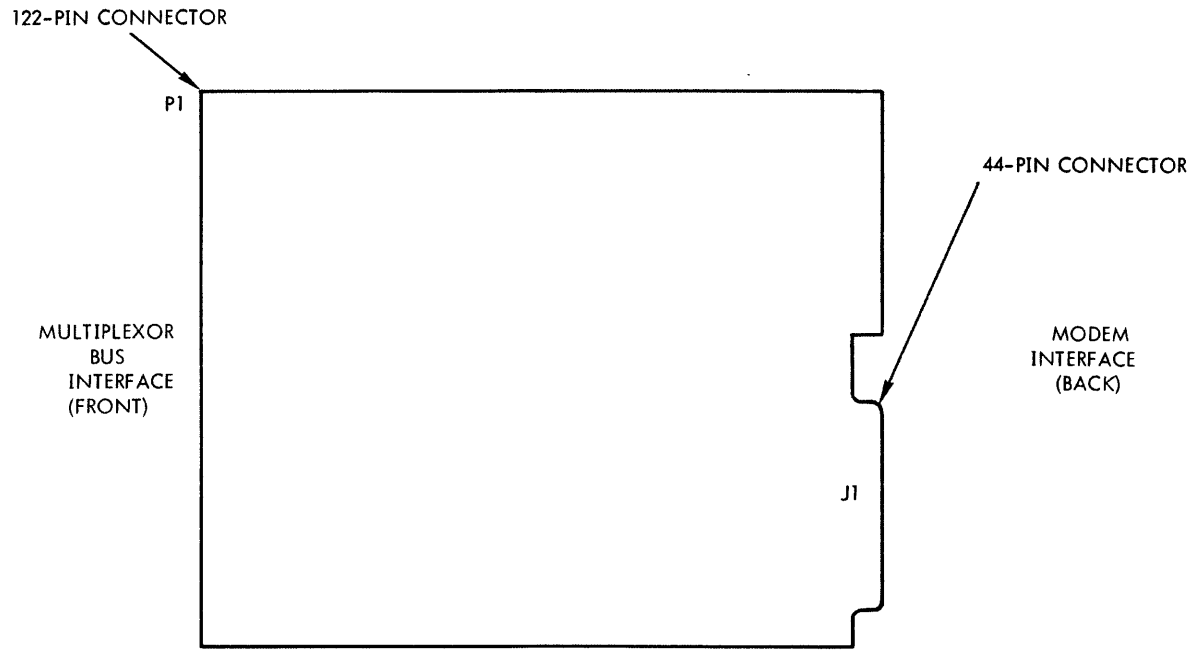
Table 6-1. Binary Synchronous Communications LAD Specifications

Number of Lines/Board	One
Transmission Type	Serial synchronous
Modem Interface	EIA RS232C
Modes of Operation	a. BSC EBCDIC (this mode includes transparent capability) b. BSC USASCII non-transparent c. BSC USASCII with transparency d. Transmit test mode e. Receive test mode
Maximum Bit Rate	20,000 bits per second (RS232C) 50,000 bits per second with optional modem interface
Options	Optional Bell 303 series modem interface
Power Requirements	+5V at 2.8 A +12V at 50 mA -12V at 50 mA
Environment	Temperature 0 to 50 degrees C, relative humidity 0 to 90 percent (without condensation).
Interconnection	Plugs into DCM backplane. Interfaces to modem via a 44-pin connector.



VTII-3182

Figure 6-1. Binary Synchronous Communications LAD Block Diagram



VTII-3183

Figure 6-2. Binary Synchronous Communications LAD

6.1 INSTALLATION

The LAD has been inspected and packed to ensure its arrival in good working order. To prevent damage, take reasonable care during unpacking and handling. Check the shipping list to ensure that all equipment has been received. Immediately after unpacking, inspect the equipment for shipping damage. If any is found:

a. Notify the transportation company.

b. Notify Varian Data Machines.

c. Save all packing material.

6.1.1 Physical Description

The binary synchronous communications LAD is on one printed-circuit board (figure 6-2) mounted in a specially wired expansion chassis (figures 1-2 and 1-3).

6.1.2 Interfaces

The LAD interfaces with the MU over the multiplexor bus (figures 1-1 and 2-1), and with the data sets over the RS232 modem interface (section 6.2.7).

6.1.2.1 Multiplexor Bus

The 38-line multiplexor bus connects the LAD with the MU. It is described in section 2.1.1.1.

6.1.2.2 RS232 Modem Interface

The interface is described in section 6.2.7.

6.1.3 Options

A high-speed interface is available to interface with Bell 300 series modems.

6.2 THEORY OF OPERATION

This section explains the operation of the binary synchronous communication LAD. Refer to the detailed block diagram in figure 6-3 and logic diagram 91B0493 for understanding this theory.

6.2.1 Multiplexor Bus Interface

This logic provides a common interface for the lines of the LAD to the multiplexor bus (section 2.1.2.2). This interface assigns the LAD an address on the multiplexor bus.

6.2.2 Line Address Decode

The line adapter address (LADA-) and line adapter enabling signal (LADE-) are ANDed together to enable 4 of 64 lines. SCB0- and SCB1- are then decoded to provide the address decoder outputs L00 through L03. One of the signals is then picked up on the DCM backplane to provide LADE- for one line address.

6.2.3 Strobe Distribution Logic

The strobe distribution logic is enabled by LAD enabling signal (LADE-). Write (WRITE+) and strobe (STRBE+) signals are ANDed together to provide a write strobe signal (WSTRBE+).

WRITE- and STRBE+ are ANDed together to provide a read strobe (RSTRBE+). These signals, along with STROBE-, are sent to the transmitting test multiplexor and the receiving test multiplexor so they can be sensed by the control memories. A- and B- are decoded by a dual two-four line decoder. The upper half of the decoder is enabled by WRITE- and STRBE+ to provide the three load strobes.

MSLD- is the load clock for the mode selection register. CRLD- loads the control register and OLDCK- is the output-load clock for the output shift register. Note that OLDCK- is only enabled if the line is making an output data request (ODRA). The lower half of the decoder is enabled by WRITE+ and is used to enable status or data onto the multiplexor bus.

6.2.4 Service Request Logic

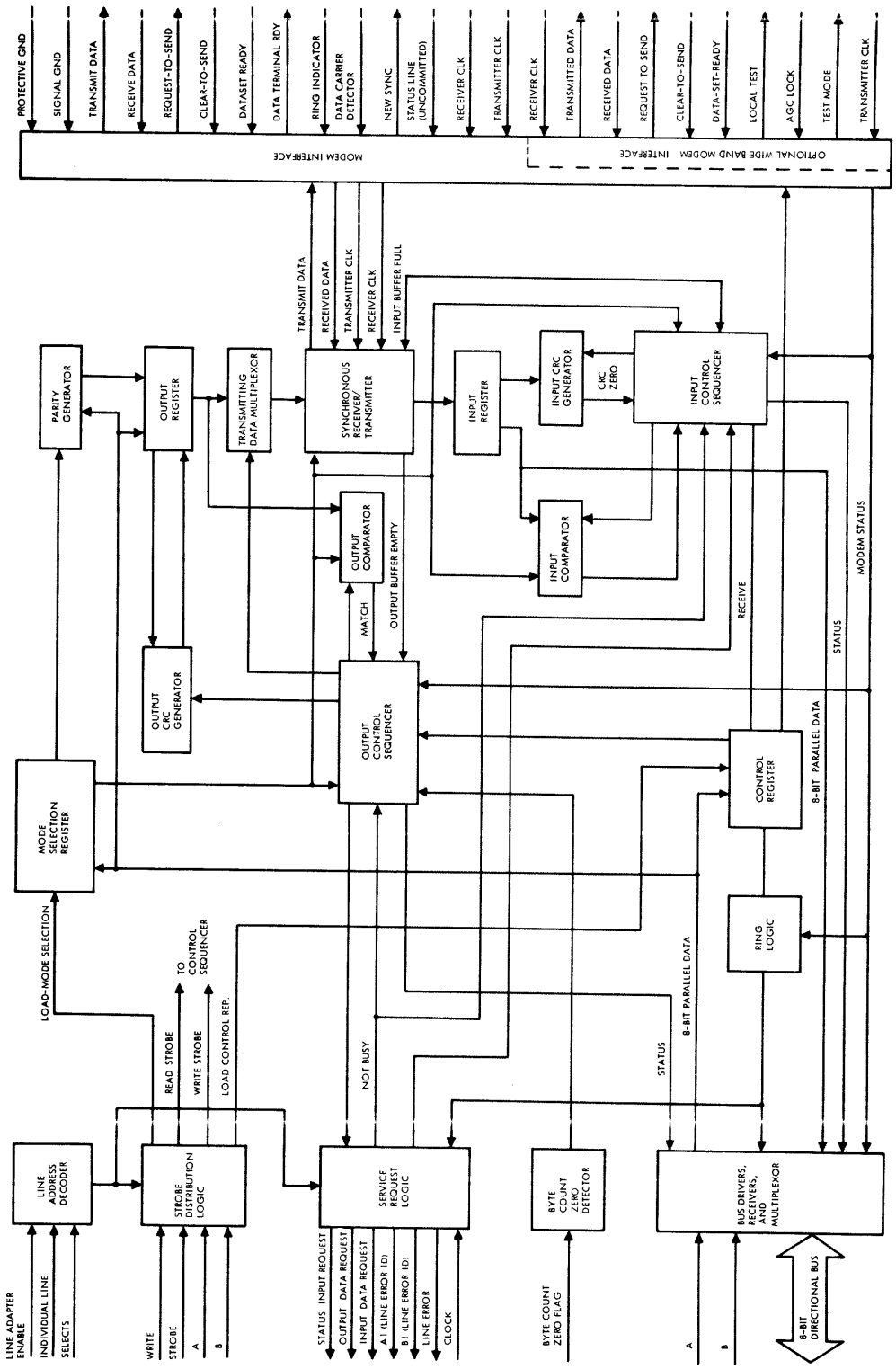
The service requests are clocked into the service request register on the positive going edge of the clock (CLK-) when LAD enabling signal (LADEN+) is true. The status input request (SIR+) is enabled by IFUNC5-, IFUNC6-, or RING-. Line errors are held in the line error register and reported by making an input-data request. The line error clock (LLERR-) is disabled as soon as line error (LERR+) goes true. This allows only the initial error to be reported in any receiving sequence. The line error register is cleared when the R bit is cleared in the control register. Output-service busy (OSBZY-) and input-service busy (ISBZY-) prevent the input and output control sections from making requests at the same time. OSBZY- can be enabled by IFUNC7-. This allows the input control section to present a busy state to the output control section while it waits for the service request logic to become free. This ensures that it will be serviced next.

6.2.5 Bus Drivers, Receivers and Multiplexor

Either data (ISR0+ through ISR7+) or status is output depending on the states of the signals EDATA+ and ESTAT+. The most significant data bit (ISR7+) is only enabled if the input section is in the transparent mode (ITRANS-) or the ASCII mode bit (UNT+) of the mode selection register is reset. IFUNC5- enables the control-character-defected status bit. OFUNC5- gates the output control memory bits. OCM03+ and OCM04+ enable the underflow and/or the output-completion status bits.

6.2.6 Control Register

The control register is loaded from the multiplexor bus to determine the operation of the line adapter. The transmitted and received bits are stored by individual flip-flops so they can be cleared individually by the control programs.



VTI2-436

Figure 6-3. BSC LAD Detailed Block Diagram

6.2.7 Mode-Selection Register and Transparent Mode Flip-Flops

The mode-selection register stores the current line-adapter mode. The signals, TT−, UWT+, and OTRANS− are ANDed together to form signal UBES+. UBES determines whether the output of the parity generator or MB7+ is transferred to the output-shift register.

The two transparent-mode flip-flops (OTRANS+ and ITRANS+) store the fact that the input and output sections are, or are not, in the transparent mode. The input transparent mode signal (ITRANS+) is enabled in the test receive mode (TR+) so that the most significant input data bit is transferred to the processor.

6.2.8 Byte Count Zero Detector

The byte-count zero detector samples the byte count flag (CZ+) from the multiplexor every time data is transferred to the line adapter. The state of the output byte-count zero flip-flop (OBCZ+) is sampled by the output control section to determine when an ending control character is being output in the transparent text mode.

6.2.9 Input/Output Shift Registers and Counters

The input/output shift registers provide buffering and store the data during checks for control characters. The shift-register contents are shifted into the CRC generators for block check character generation and checking.

The input/output counters are used to count the number of clock pulses required to shift the shift-register contents into the CRC generator. Figure 6-4 illustrates the timing relationship for the output section. The input timing is the same.

6.2.10 Input/Output CRC Generator

The input/output CRC generator uses a universal polynominal generator which is capable of handling IBM's CRC-16 polynomial ($X^{16} + X^{15} + X^2 + 1$) and the LRC-8 polynominal ($X^8 + 1$). When the UNT+ bit is set, in the mode-selection register, the LRC-8 polynominal is selected, otherwise CRC-16 is used. The input CRC generator is used only for checking the incoming data stream for the correct block check character. This is accomplished by accumulating a block check character by shifting the data through the generator and checking that the CRC generator contents are completely cleared when the received block check character is shifted in. On output operations, after accumulating the block check character, the shift-right input is enabled and data is transferred out of the CRC generator. See figure 6-4 for the timing relationships.

6.2.11 Input and Output Comparators

The 8-bit input and output comparators are used to compare the shift register contents with the control memory's literal field for control character detection. In the ASCII mode, the eight bit is not included in the comparison.

6.2.12 Clock

The clock logic divides the 4.9152 MHz square wave (CLKB+) from the multiplexor to provide a 2.4576 MHz square wave (BCLB2−) and a 1.2288 MHz square wave (BSCLK+).

6.2.13 Input and Output Control Sequencers

The input and output control sequencers contain the control memories and associated logic to control the LAD input and output operations.

6.2.13.1 Input and Output Control Memories

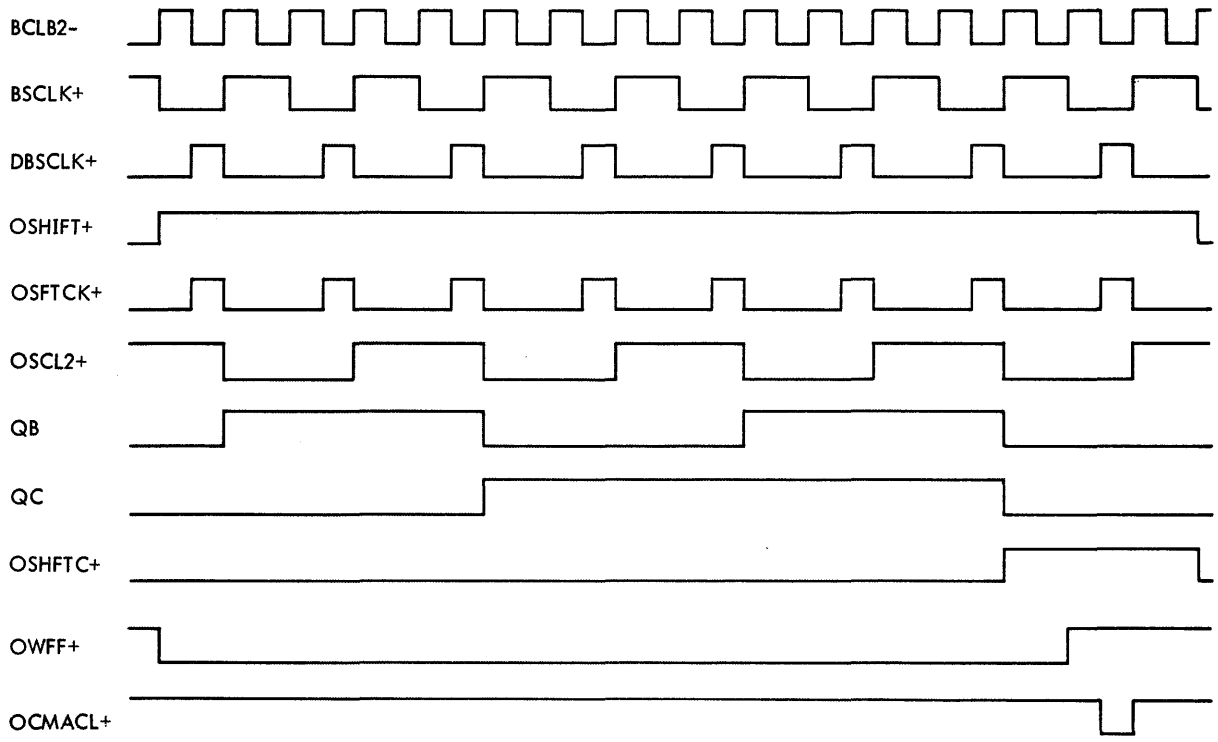
The two identical memories each contain 256 24-bit words. Addresses 0 through 127 are for EBCDIC operation and addresses 127 through 255 are for ASCII operation. The state of the ASCII signal (ASCII+) determines which section of the control memory is accessed.

6.2.13.2 Input and Output Control Memory Address Counters

The two counters are identical except in the way they are cleared. The input counter is cleared whenever the receiving (R+) bit of the control register is low. The output counter is cleared whenever the transmit (T+) bit of the control register is loaded with a zero or by a system reset (SRST+). The counters provide the address for the control memory and count up sequentially as long as the load input is kept high. The counters are loaded from the control-memory jump-address field or from the return register.

6.2.13.3 Input and Output Return Registers

These two registers are identical and are loaded from the literal field of the control memory. The return registers provide subroutine capabilities.



VTII-3183

Figure 6-4. LAD Timing Diagram

6.2.13.4 Input and Output Test Multiplexor

The two test multiplexors allow testing of 16 sources. Selection is provided from the control-memory test field. The output from the test multiplexor is synchronized with the clock (BCLKB +) to provide the capability of doing a wait-and-jump operation on asynchronous signals. The input multiplexor does not have this capability.

6.2.13.5 Input and Output Jump, Wait, and Shift Logic

This logic enables the control sections to do either a jump operation depending on the state of the test multiplexor or to wait until the test multiplexor output goes true. This logic also clocks the shift functions true (figure 6-4 shows the timing relationships). The wait flip-flop (OWFF or IWFF) is normally true and goes false if bit 12 of the control memory is true and the test multiplexor output is false. This disables the address counter clock until the test multiplexor output goes true. If bit 11 is true, the jump flip-flop will go true (OJFF and IJFF) if the output of the test multiplexor is true. This enables the load line (LOCMAC– or LICMAC–) which causes the address counter to be loaded with the contents of the jump-address field. The load line can also be enabled by the function one signal which loads the address counter with the return register contents.

6.2.14 Transmitting Data Multiplexor

The multiplexor selects the input to the transmitting section of the synchronous receiver/transmitter from either the literal field of the output control memory or the contents of the output shift register. Bit 13 of the output control memory (OCM13 +) determines the source.

6.2.15 Synchronous Receiver/Transmitter

The synchronous receiver/transmitter performs all the receiving and transmitting functions associated with synchronous data communications. Table 6-2 gives a description of the input and output signals.

The input clock frequency for the receiver is set by the modem and the desired receiver sync character (synchronous idle character) is loaded into the receiver sync register. When the receiver-reset input changes from a high

level to a low level, the receiver enters the search mode (bit phase). In the search mode, the serially received data bit stream is examined on a bit-by-bit basis until a sync character is found. A sync character is found, by definition, when the contents of the sync register and the receiver shift register are identical. When this occurs, the sync-character-received output is set high. This character is then loaded into the receiver buffer register and the receiver enters the character mode. In this mode, each character received is loaded into the receiver buffer register. The receiver provides flags for receiver-data-available, receiver-overrun, receiver-parity-error, and sync-character-received. Full double buffering eliminates the need for precise external timing by allowing one full character time for received data to be read out.

The input clock frequency for the transmitter is set by the modem and the desired transmitter sync character is loaded into the transmitter sync register. Internal logic determines if the character to be transmitted out of the transmitter shift register is extracted from the transmitter data register or the transmitter sync register. The next character transmitted is extracted from the transmitter data register provided that a transmitter data strobe pulse occurs during the presently transmitted character. If the transmitter data strobe is not pulsed, the next transmitted character is extracted from the transmitter sync register and the sync character transmitted output is set to a high level. Full double buffering eliminates the need for precise external timing by allowing one full character time to load the next character to be transmitted.

There may be 5, 6, 7, or 8 data bits, and odd/even or no parity bit. All inputs and outputs are directly TTL compatible. Tri-state receive-data-output levels are provided for the bus structure oriented signals.

Table 6-2. Pin Function of the Synchronous Receiver/ Transmitter

Pin	Symbol	Name	Function
1	VCC	Power supply	+5V supply
2	TBMT	Transmitter buffer empty	This output is at a high level when the transmitter data-buffer register is loaded with new data
3	TSO	Transmitter serial output	This output serially provides the entire transmitted character. This character is extracted from the transmitter data buffer register provided that a TDS pulse occurs during the presently transmitted character. If TDS does not occur, the next transmitted character will be extracted from the transmitter sync register.
4	GND	Ground	Ground
5	SCT	Sync character transmitted	This output is high when the character loaded into the transmitted shift register is extracted from the transmitter sync register, indicating that the TDS was not pulsed during the previously transmitted character. This output is low when the character to be transmitted is extracted from the transmitter data buffer register, which can occur only if TDS is pulsed
6	VDD	Power supply	-12V supply

Table 6-2. Pin Function of the Synchronous Receiver/ Transmitter *(continued)*

Pin	Symbol	Name	Function
7-14	DB1-DB8	Data bus inputs	This 8-bit bus inputs information into the receiver sync register under control of the RSS strobe, into the transmitter sync register under control of the TSS strobe, and into the transmitter data-buffer register under control of the TDS strobe. The strobes operate independently. Unused bus inputs should be at a high level. The LSB should always be placed on DB1
15	RR	Receiver reset	This input should be pulsed to a high-level after power turn on. This resets the RDA, SCR, ROR, and RPE outputs to a low level. The transition of the RR input from a high to a low puts the receiver in the search mode (bit phase). In the search mode, the serially received data-bit stream is examined on a bit-by-bit basis until a sync character is found. A sync character is found, by definition, when the contents of the receiver sync register and the receiver shift register are identical. When this occurs, the SCR output goes high. This character is then loaded into the receiver buffer register and the receiver is set into the character mode. In this mode, each character received is loaded into the receiver buffer-register
16	RPE	Receiver parity error	This output is high if the received character parity bit does not agree with the selected parity
17	SCR	Sync character received	This output is high whenever a character loaded into the receiver buffer-register is identical to the character in the receiver sync register. This output goes low the next time the receiver buffer-register is loaded with a character that is not a sync character
18	TSS	Transmitter sync strobe	A high-level input strobe loads the character on the DB1-DB8 lines into the transmitter sync register
19	TCP	Transmitter clock	The positive going edge of this clock shifts data out of the transmitter shift register, at a baud rate equal to the TCP clock frequency

Table 6-2. Pin Function of the Synchronous Receiver/ Transmitter *(continued)*

Pin	Symbol	Name	Function
20	TDS	Transmitter data-buffer strobe	A high input strobe loads the character on the DB1-DB8 lines into the transmitter data-buffer register
21	RSS	Receiver sync strobe	A high input strobe loads the character on the DB1-DB8 lines into the receiver sync register
22	RSI	Receiver serial input	This input accepts the serial bit-input stream
23	RCP	Receiver clock	The negative-going edge of this clock shifts data into the receiver shift register, at a baud rate equal to the RCP clock frequency
24	RDAR	Receiver data available reset	A high input resets the RDA output to a low
25	RDE	Receiver data enable	A high input enables the outputs (RD8-RD1) of the receiver buffer-register
26	RDA	Receiver data available	This output is high when an entire character has been received and transferred into the receiver buffer-register
27	ROR	Receiver overrun	This output is high if the previously received character is not read (RDA not reset) before the present character is transferred into the receiver buffer register
28 thru 35	RD8 thru RD1	Receiver data output	These are eight tri-state data outputs enabled by RDE. Unused data output lines, as selected by NDB1 and NDB2, have a low output, and received characters are right justified, i.e., the LSB always appears on the RD1 output
36, 38	NDB2, NDB1	Number of data	These two inputs are internally decoded to select either 5, 6, 7, or 8 data bits/character as follows: NDB2 NDB1 Data bits/character L L 5 L H 6 H L 7 H H 8
37	NPB	No parity bit	A high input prevents the parity bit from being transmitted. In addition, it is necessary that the received character contain no parity bit. Also, the RPE output is forced low

Table 6-2. Pin Function of the Synchronous Receiver/ Transmitter *(continued)*

Pin	Symbol	Name	Function
39	CS	Control strobe	A high input enters the control bits (NDB1, NDB2, POE, and NPB) into the control bits register. This line may be strobed or hardwired high
40	POE	Select odd/even parity	The logic levels on this input and on NPB, determine the parity mode for both the receiver and transmitter, as follows:

NPB	POE	MODE
L	L	Odd parity
L	H	Even parity
H	X	No parity (POE has no affect)

6.2.16 Loopback Multiplexor

The loopback multiplexor provides the capability of testing the LAD without a test shoe or modem. Normally, the transmitter and receiver clocks come from the modem interface, the received data comes from the RD+ line of the modem interface, and output-data-request (OUTDR+) is enabled by the term DTD−. When the LB+ bit is set in the mode-selection register, the test clock (TESTCK+) is gated to the transmitter and receiver clock outputs of the multiplexor, transmitter data (TSOA+) is gated to the receiver data line, and transmission signal (T+) enables the OUTDR+ gate.

6.2.17 RS232C Modem Interface

This interface conforms to the RS232C and CCITT V24 standards and provides for the following signals (pin numbers are indicated in table 6-3):

AA−	Protective ground
AB−	Signal ground
BA−	Transmitted data
BB−	Received data
CA−	Request to send
CB−	Clear to send
CC−	Data ready
CD−	Data terminal ready
CE−	Ring indicator
CF−	Data carrier detector
DB−	Transmitter clock
DD−	Receiver clock

6.2.18 Optional Wide Band Modem Interface

This interface is designed to specifically to the Bell 303 type modem. The interface provides for the following signals (pin numbers are indicated in table 6-3):

CS−	Clear to send
SR−	Send request
SD−	Send data
DSR−	Data set ready
RI−	Ring indicator
LT−	Local test
SCT−	Serial clock transmit
RD−	Received data
SCR−	Serial clock receive
AGC−	AGC lock
DTR−	Data terminal ready

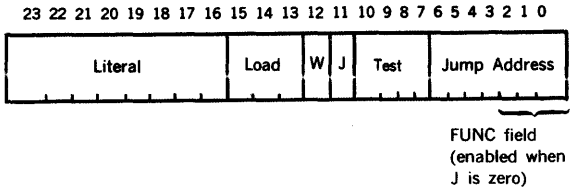
Table 6-3. RS232 Signal Locations

RS232 Signal	Wide Band Option	DCM Mnemonic	Pin
AA		PGND	J2-24
BA		TDA–	J2-31
BB	RD	RDS–	J2-23
CA		RTS +	J2-27
CB	CS	CS +	J2-37
CC	DSR	ILK +	J2-33
AB		GRD	J2-20,22
CF	AGC	CO +	J2-25
CD	DTR	DTRY +	J2-29
CE	RI	RID +	J2-35
		CL1 +	J2-21
		CC02 +	J2-17
		– 12V	J2-14
		+ 12V	J2-15
DB	SCT	TSET +	J2-39
DD	SCR	RSET +	J2-41
	SR		J2-3
	SD		J2-5
	LT		J2-7

6.2.19 BSC LAD Output-Control Program

Data and control information are transmitted by the LAD in the form of an output-control word. The output-control word is described in figure 6-5. Figure 6-6 shows the sequences the LAD follows during the operation of the output-control program. Table 6-4 lists the sequences stored at the various addresses of the output control memory.

Table 6-5 lists the contents of the output control memory. The memory consists of 6 integrated circuits (ICs) with each IC containing 256 4-bit words. Location designations are given in table 6-5 as an aid in locating the memory ICs on the PC board.



Fields are described below:

LOAD	Description
000	No operation
001	Trigger the transmitter 1-second timer
010	Transfer contents of literal field to the output buffer of the synchronous transmitter/receiver
011	Transfer contents of literal field to the return register (literal must be six bits or less)
100	Transfer contents of literal field to the transmitter-sync-character register of the synchronous transmitter/receiver
101	Transfer output-shift register contents to the output buffer of the synchronous transmitter/receiver output buffer
110	Transfer contents of literal field to the receiver-sync-character register of the synchronous transmitter/receiver
111	Clear the transmitter bit in the control register

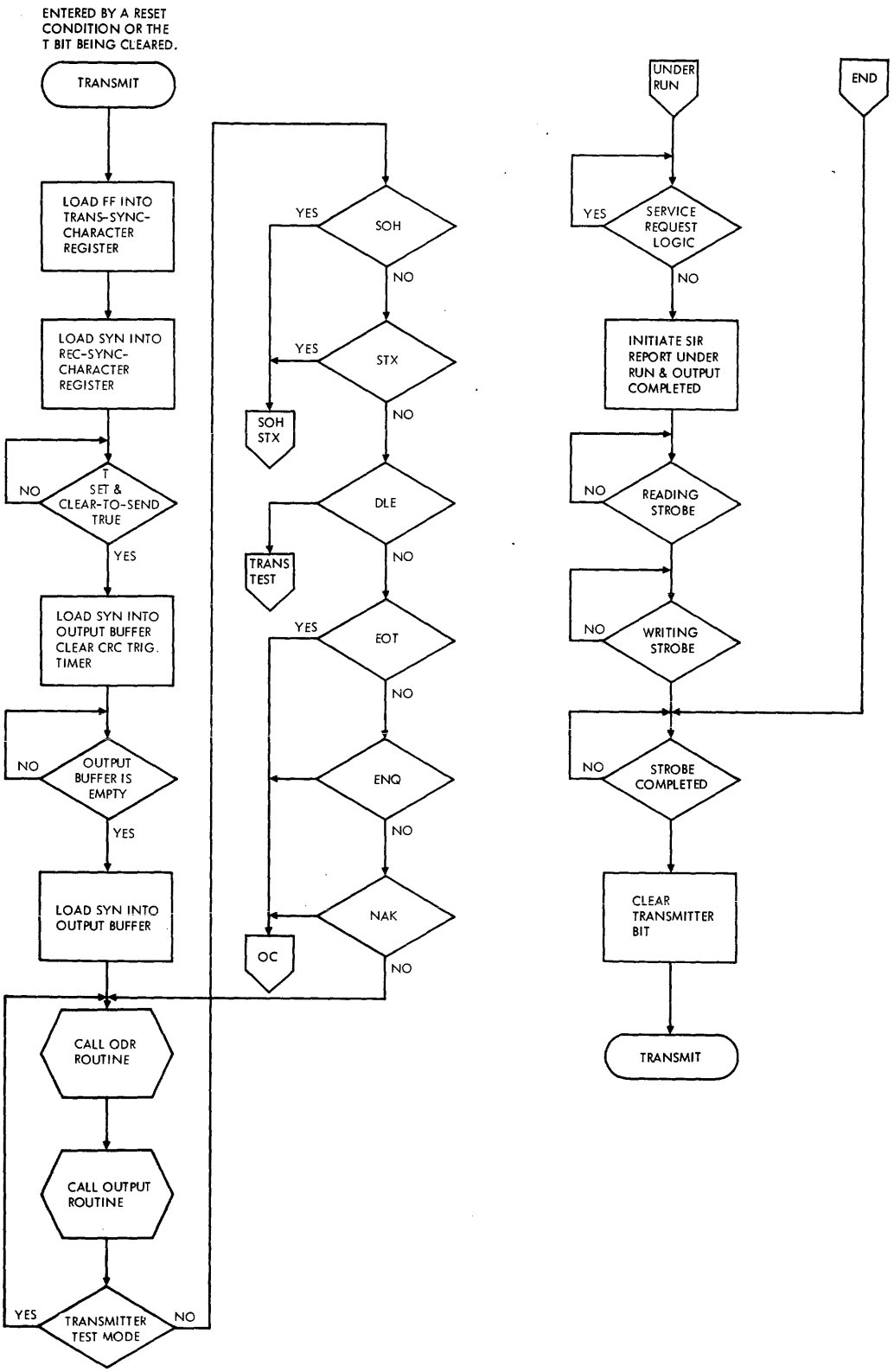
FUNC	Description
(Note: FUNC field is decoded only if J is zero)	
000	Transfer an output data request to the service request logic
001	Transfer the contents of the return register to the control memory address counter
010	Enable the shift-right mode of the CRC generator
011	Enable the CRC accumulation
100	Set the output-transparent-mode flip-flop
101	Transfer a status input request to the service request logic. Bit 3 of the control memory enables underflow status; bit 4 enables output completed status (negative true)
110	Reset the CRC generator and the transparent-mode flip-flop
111	No operation

Test	Description*
0000	Transparent mode flip-flop is set
0001	Transmit bit of the control register is set
0010	Contents of the literal field match the contents of the output-shift register
0011	Transmitter 1-second timer has timed out
0100	NBC bit of the control register is set
0101	TT bit of the mode-selection register is set (transmit test mode)
0110	Output byte count is not zero
0111	UNT bit of the mode-selection register is set
1000	Underflow (sync-character-transmitted flag of the synchronous transmitter receiver is true)
1001	Output buffer of the synchronous transmitter/receiver is empty
1010	Loader clock of output-shift register is true.
1011	Service-request logic is not busy
1100	Read strobe is true
1101	Write strobe is true
1110	CRC shift completion
1111	Strobe completion

* When W is one, a wait condition occurs.

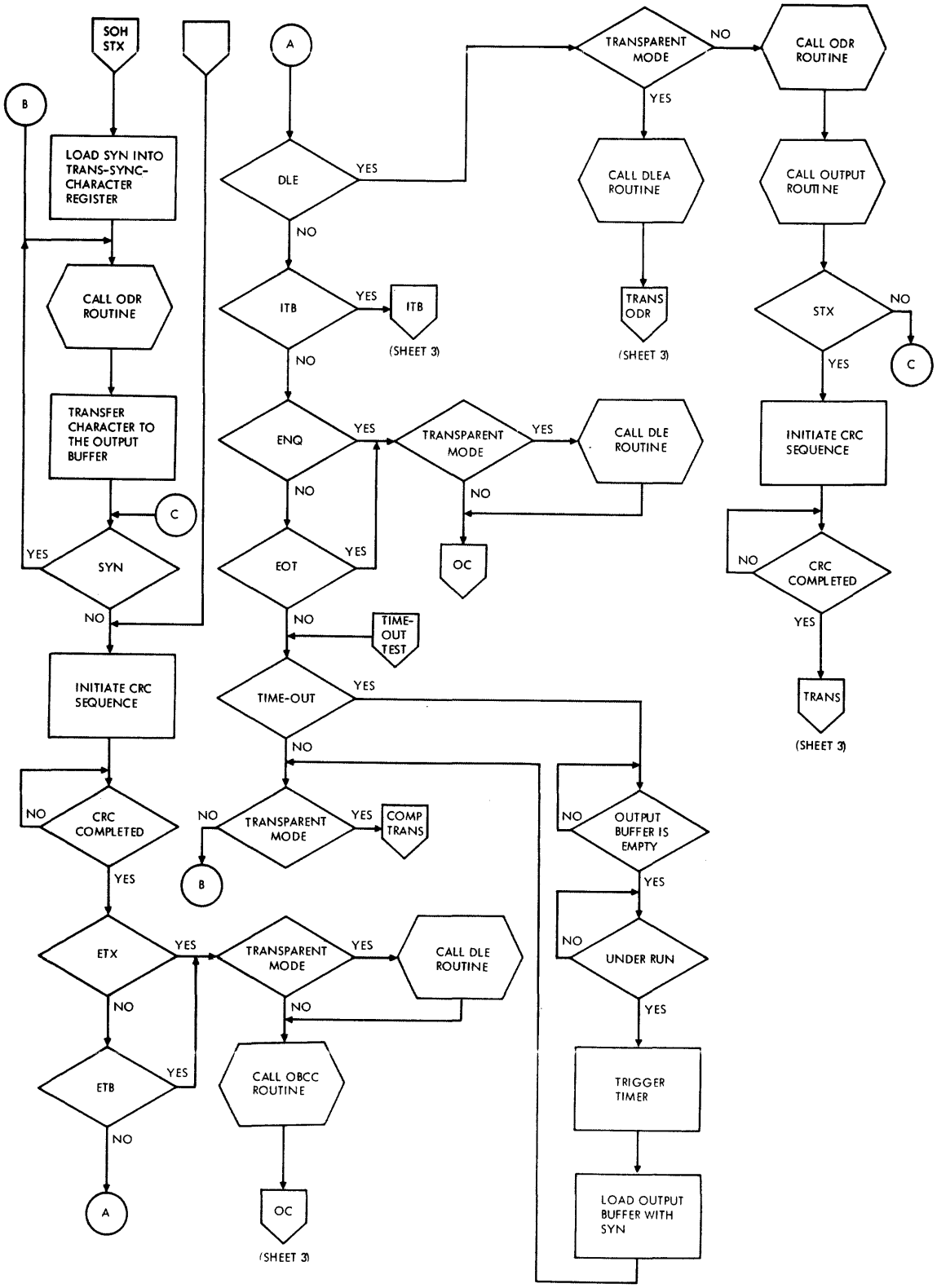
* When J is one, a jump condition occurs.

Figure 6-5. Output-Control Word Format



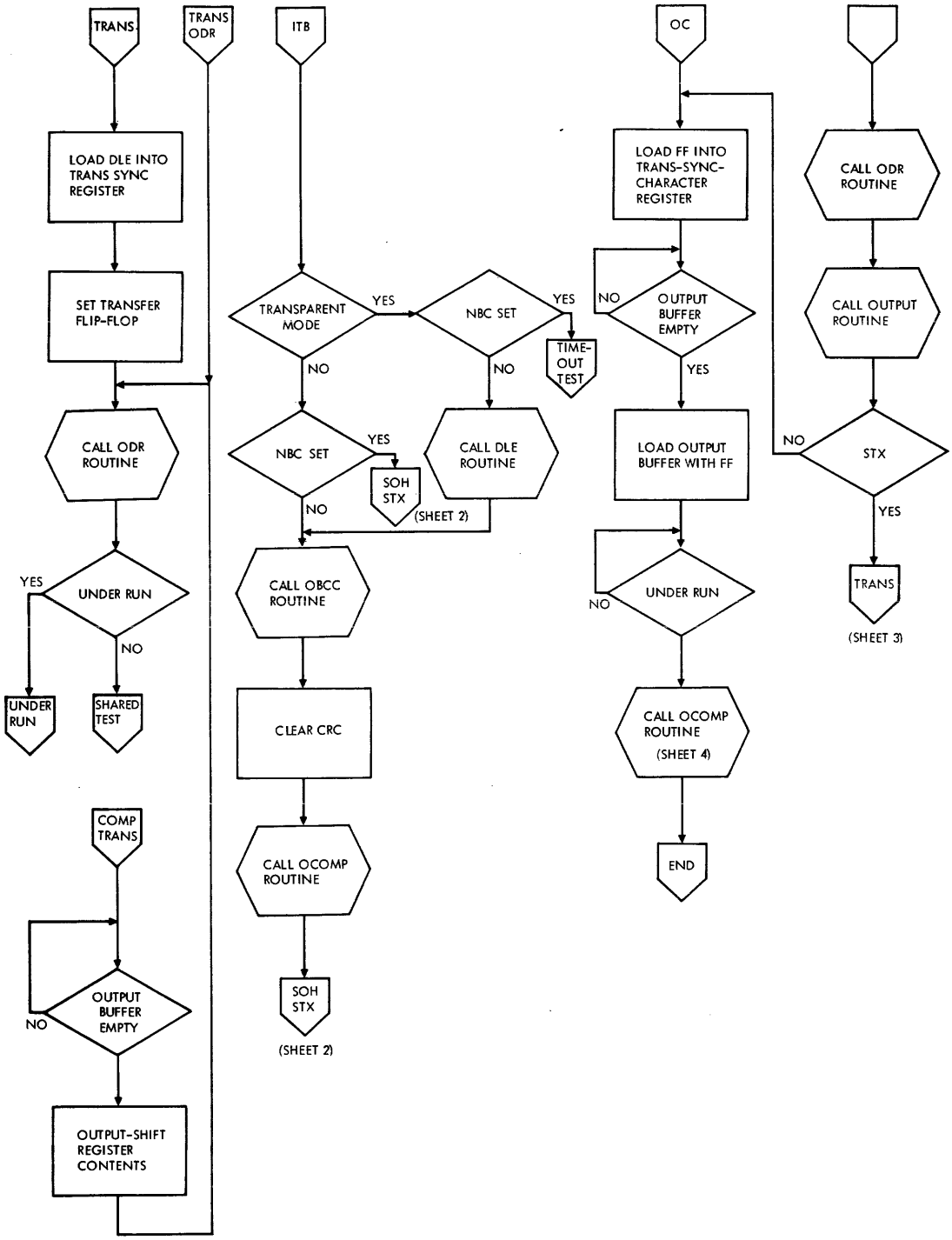
VTI2-427

Figure 6-6. LAD Output-Control Sequences *(sheet 1 of 4)*



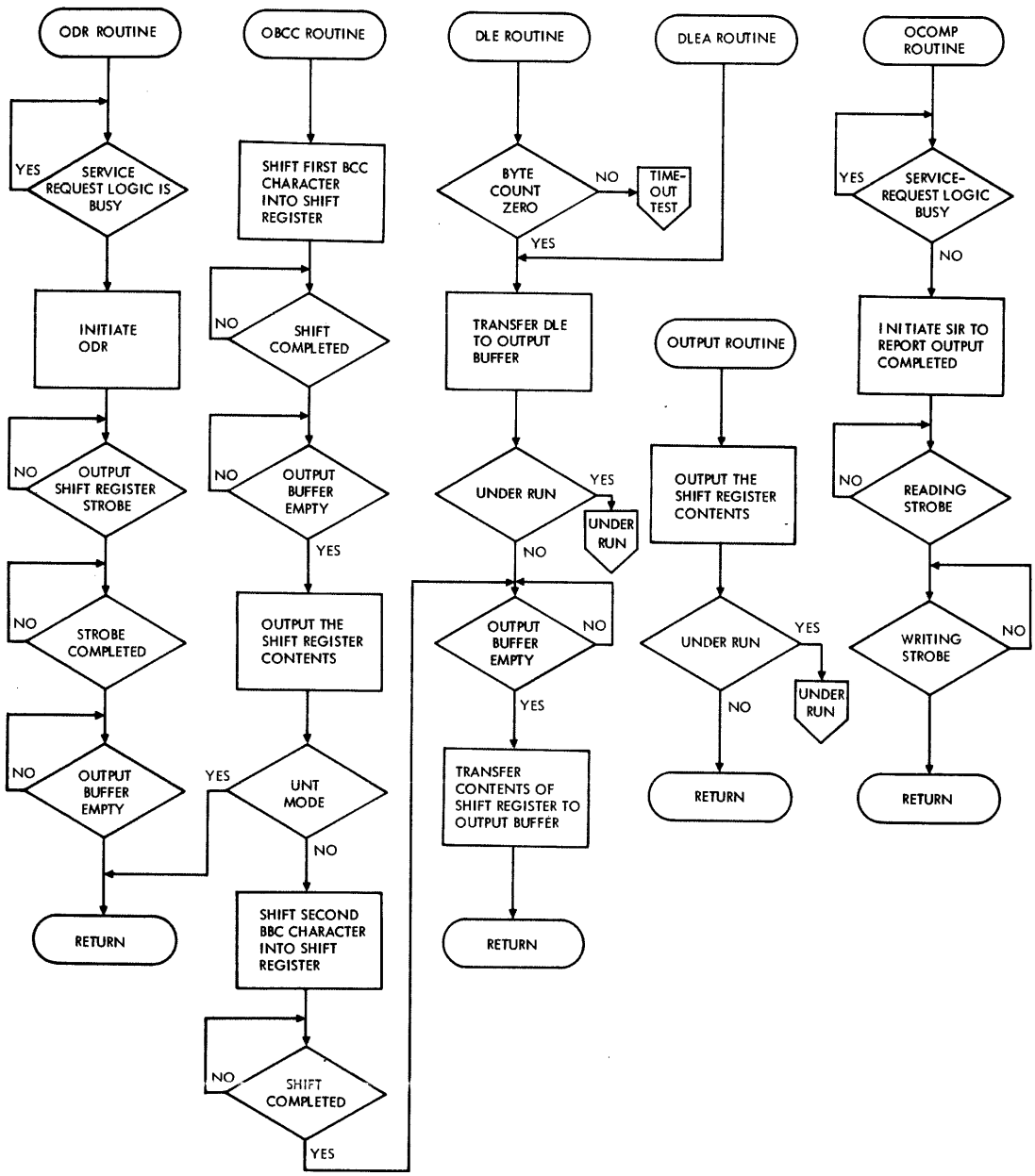
VT12-428

Figure 6-6. LAD Output-Control Sequences *(sheet 2 of 4)*



VTI2-429

Figure 6-6. LAD Output-Control Sequences *(sheet 3 of 4)*



VTI2-430

Figure 6-6. LAD Output-Control Sequences *(sheet 4 of 4)*

Table 6-4. Output-Control Program

7-Bit Address	Remarks	Literal	Load	W	J	Test	Jump Address
00	Load FF into transmitter sync-character register	FF	4	0	0	0	07
01	Load SYN into receiver sync-character register, wait for buffer empty and clear-to-send	SYN	6	1	0	9	07
02	Load SYN into output buffer; clear CRC	SYN	2	0	0	0	06
03	Trigger time, wait until buffer is empty	0 0	1	1	1	9	5F
04	Call ODR; return	0 5	3	1	1	F	47
05	Call output routine; return	0 6	3	1	1	F	55
06	Test for transmitter test mode	0 0	0	0	1	5	04
07	Test for SOH; jump to SOH; STX	SOH	0	0	1	2	0E
08	Test for STX; jump to SOH; STX	STX	0	0	1	2	0E
09	Test for DLE; jump to transmitter test	DLE	0	0	1	2	34
0A	Test for EOT; jump to OC	EOT	0	0	1	2	29
0B	Test for ENQ; jump to OC	ENQ	0	0	1	2	29
0C	Test for NAK; jump to OC	NAK	0	0	1	2	29
0D	Jump back to 04	0 0	0	1	1	F	04
40	Wait until service-request logic is not busy	0 0	0	1	0	B	07
41	Initiate SIR to report under run and output completed; wait for reading strobe	0 0	0	1	0	C	05
42	Wait for writing strobe	0 0	0	1	0	D	07
43	Wait for strobe completion	0 0	0	1	0	F	07
44	Clear transmitter bit; jump to 0	0 0	7	1	1	F	00
5F	Output SYN, jump back to 04	SYN	2	1	1	F	04

Note: Codes are given in hexadecimal within each field.

Table 6-4. Output-Control Program *(continued)*

7-Bit Address	Remarks	Literal	Load	W	J	Test	Jump Address
0E	Load SYN into transmitter sync-character register	SYN	4	0	0	0	07
0F	Call ODR, return	1 0	3	1	1	F	47
10	Transfer character to output buffer; test for SYN	SYN	5	0	1	2	0F
11	Initiate CRC sequence, wait for completion	0 0	0	1	0	E	03
12	Test for ETX; jump to 1B	ETX	0	0	1	2	1B
13	Test for ETB; jump to 1B	ETB	0	0	1	2	1B
14	Test for DLE; jump to 1B	DLE	0	0	1	2	1E
15	Test for ITB; jump to ITB	ITB	0	0	1	2	2C
16	Test for ENQ; jump to 28	ENQ	0	0	1	2	28
17	Test for EOT; jump to 28	EOT	0	0	1	2	28
18	Test for time-out; jump to 23	0 0	0	0	1	3	23
19	Test for transparent mode; jump to transparent-mode completion	0 0	0	0	1	0	38
1A	Jump to (B)	0 0	0	1	1	F	0F
1B	Test for transparent mode; jump to 1D	0 0	0	0	1	0	1D
1C	Call OBCC; return to OC	2 9	3	1	1	F	4B
1D	Call DLE; return to 1C	1 C	3	1	1	F	50
1E	Test for transparent mode; jump to 27	0 0	0	0	1	0	27
1F	Call ODR; return	2 0	3	1	1	F	47
20	Call output routine; return	2 1	3	1	1	F	55
21	Test for STX; jump to 5B	STX	0	0	1	2	5B
22	Jump to C	0 0	0	1	1	F	10
5B	Initiate CRC sequence; wait for completion	0 0	0	1	0	E	03
27	Call DLEA; return to TRANS ODR	3 D	3	1	1	F	51
28	Test for transparent mode; jump to 33	0 0	0	0	1	0	33
33	Call DLE, return to OC	2 9	3	1	1	F	50
23	Wait until output buffer is empty	0 0	0	1	0	9	07
24	Wait for under run, trigger timer	0 0	1	1	0	8	07
25	Load output buffer with SYN; jump back to 14	SYN	2	1	1	F	19

Note: Codes are given in hexadecimal within each field.

Table 6-4. Output-Control Program *(continued)*

7-Bit Address	Remarks	Literal	Load	W	J	Test	Jump Address
5C	Load DLE into transmitter sync-register/ set transparent mode FF	DLE	4	0	0	0	04
5D	Call ODR; return to under run test	3 9	3	1	1	F	47
2C	Test for transparent mode; jump to 31	0 0	0	0	1	0	31
2D	Test NBC; jump to SOH/ STX	0 0	0	0	1	4	0E
2E	Call OBCC; return	2 F	3	1	1	F	4B
2F	Clear CRC	0 0	0	0	0	0	06
30	Call O COMP; return to SOH/STX	0 E	3	1	1	F	58
31	Test NBC; jump to time-out test	0 0	0	0	1	4	18
32	Call DLE; return to 2E	2 E	3	1	1	F	50
29	Load FF into transmitter sync-character register wait until buffer is empty	F F	4	1	0	9	07
2A	Load output buffer with FF	F F	2	0	0	0	07
2B	Wait for under run; call O COMP, return to END	3 B	3	1	1	8	58
34	Call ODR; return	3 5	3	1	1	F	47
35	Call output routine, return	3 6	3	1	1	F	55
36	Test for STX; jump to transparent mode	STX	0	0	1	2	5C
37	Jump to OC	0 0	0	1	1	F	29
38	Wait for output buffer empty	0 0	0	1	1	9	3D
3B	Jump to END	0 0	0	1	1	F	43
3D	Output shift-register contents, jump to transparent mode ODR	0 0	5	1	1	F	5D
39	Test for under run	0 0	0	0	1	8	40
3A	Jump to shared test	0 0	0	1	1	F	11

Note: Codes are given in hexadecimal within each field.

Table 6-4. Output-Control Program *(continued)*

7-Bit Address	Remarks	Literal	Load	W	J	Test	Jump Address
47	Wait for service request logic not busy	0 0	0	1	0	B	07
48	Initiate ODR, wait for output shift-register strobe	0 0	0	1	0	A	00
49	Wait for strobe complete	0 0	0	1	0	F	07
4A	Wait for output buffer empty, return	0 0	0	1	0	9	01
4B	Shift first BCC character into shift-register, wait	0 0	0	1	0	E	02
4C	Wait for output buffer empty	0 0	0	1	0	9	07
4D	Output first BCC character and test for UNT; jump to 57	0 0	5	0	1	7	57
4E	Shift second BCC into shift-register and wait	0 0	0	1	0	E	02
4F	Jump to 53	0 0	0	1	1	F	53
50	Test byte count not zero; jump to time-out test	0 0	0	0	1	6	18
51	Transfer DLE to output buffer	DLE	2	0	0	0	07
52	Test for under run; jump to under run	0 0	0	0	1	8	40
53	Wait until output buffer is empty	0 0	0	1	0	9	07
54	Output shift-register contents; return	0 0	5	0	0	0	01
55	Output shift-register contents	0 0	5	0	0	0	07
56	Test under run; jump to under run	0 0	0	0	1	8	40
57	Return	0 0	0	0	0	0	01
58	Wait until service-request logic is not busy	0 0	0	1	0	B	07
59	Initiate SIR (output completed), wait for reading strobe	0 0	0	1	0	C	0D
5A	Wait for writing strobe: return	0 0	0	1	0	0	01

Note: Codes are given in hexadecimal within each field.

Table 6-5. Contents of Output Control Memory

Address		Memory Contents					
Binary	Decimal	Location K5	Location H5	Location F5	Location D5	Location C5	Location A5
00000000	0	1111	1111	1000	0000	0000	0111
00000001	1	0011	0010	1101	0100	1000	0111
00000010	2	0011	0010	0100	0000	0000	0110
00000011	3	0000	0000	0011	1100	1101	1111
00000100	4	0000	0101	0111	1111	1100	0111
00000101	5	0000	0110	0111	1111	1101	0101
00000110	6	0000	0000	0000	1010	1000	0100
00000111	7	0000	0001	0000	1001	0000	1110
00001000	8	0000	0010	0000	1001	0000	1110
00001001	9	0001	0000	0000	1001	0011	0100
00001010	10	0011	0111	0000	1001	0010	1001
00001011	11	0010	1101	0000	1001	0010	1001
00001100	12	0011	1101	0000	1001	0010	1001
00001101	13	0000	0000	0001	1111	1000	0100
00001110	14	0011	0010	1000	0000	0000	0111
00001111	15	0001	0000	0111	1111	1100	0111
00010000	16	0011	0010	1010	1001	0000	1111
00010001	17	0000	0000	0001	0111	0000	0011
00010010	18	0000	0011	0000	1001	0001	1011
00010011	19	0010	0110	0000	1001	0001	1011
00010100	20	0001	0000	0000	1001	0001	1110
00010101	21	0001	1111	0000	1001	0010	1100
00010110	22	0010	1101	0000	1001	0010	1000
00010111	23	0011	0111	0000	1001	0010	1000
00011000	24	0000	0000	0000	1001	1010	0011
00011001	25	0000	0000	0000	1000	0011	1000
00011010	26	0000	0000	0001	1111	1000	1111
00011011	27	0000	0000	0000	1000	0001	1101
00011100	28	0010	1001	0111	1111	1100	1011
00011101	29	0001	1100	0111	1111	1101	0000
00011110	30	0000	0000	0000	1000	0010	0111
00011111	31	0010	0000	0111	1111	1100	0111
00100000	32	0010	0001	0111	1111	1101	0101
00100001	33	0000	0010	0000	1001	0101	1011
00100010	34	0000	0000	0001	1111	1001	0000
00100011	35	0000	0000	0001	0100	1000	0111
00100100	36	0000	0000	0011	0100	0000	0111
00100101	37	0011	0010	0101	1111	1001	1001
00100110	38	0000	0000	0000	0000	0000	0000
00100111	39	0011	1101	0111	1111	1101	0001
00101000	40	0000	0000	0000	1000	0011	0011
00101001	41	1111	1111	1001	0100	1000	0111
00101010	42	1111	1111	0100	0000	0000	0111
00101011	43	0011	1011	0111	1100	0101	1000
00101100	44	0000	0000	0000	1000	0011	0001
00101101	45	0000	0000	0000	1010	0000	1110
00101110	46	0010	1111	0111	1111	1100	1011
00101111	47	0000	0000	0000	0000	0000	0110
00110000	48	0000	1110	0111	1111	1101	1000
00110001	49	0000	0000	0000	1010	0001	1000
00110010	50	0010	1110	0111	1111	1101	0000
00110011	51	0010	1001	0111	1111	1101	0000
00110100	52	0011	0101	0111	1111	1100	0111

Table 6-5. Contents of Output Control Memory *(continued)*

Address		Memory Contents					
Binary	Decimal	Location K5	Location H5	Location F5	Location D5	Location C5	Location A5
00110101	53	0011	0110	0111	1111	1101	0101
00110110	54	0000	0010	0000	1001	0101	1100
00110111	55	0000	0000	0001	1111	1010	1001
00111000	56	0000	0000	0001	1100	1011	1101
00111001	57	0000	0000	0000	1100	0100	0000
00111010	58	0000	0000	0001	1111	1001	0001
00111011	59	0000	0000	0001	1111	1100	0011
00111100	60	0000	0000	0000	0000	0000	0000
00111101	61	0000	0000	1011	1111	1101	1101
00111110	62	0000	0000	0000	0000	0000	0000
00111111	63	0000	0000	0000	0000	0000	0000
01000000	64	0000	0000	0001	0101	1000	0111
01000001	65	0000	0000	0001	0110	0000	0101
01000010	66	0000	0000	0001	0110	1000	0111
01000011	67	0000	0000	0001	0111	1000	0111
01000100	68	0000	0000	1111	1111	1000	0000
01000101	69	0000	0000	0000	0000	0000	0000
01000110	70	0000	0000	0000	0000	0000	0000
01000111	71	0000	0000	0001	0101	1000	0111
01001000	72	0000	0000	0001	0101	0000	0000
01001001	73	0000	0000	0001	0111	1000	0111
01001010	74	0000	0000	0001	0100	1000	0001
01001011	75	0000	0000	0001	0111	0000	0010
01001100	76	0000	0000	0001	0100	1000	0111
01001101	77	0000	0000	1010	1011	1101	0111
01001110	78	0000	0000	0001	0111	0000	0010
01001111	79	0000	0000	0001	1111	1101	0011
01010000	80	0000	0000	0000	1011	0001	1000
01010001	81	0001	0000	0100	0000	0000	0111
01010010	82	0000	0000	0000	1100	0100	0000
01010011	83	0000	0000	0001	0100	1000	0111
01010100	84	0000	0000	1010	0000	0000	0001
01010101	85	0000	0000	1010	0000	0000	0111
01010110	86	0000	0000	0000	1100	0100	0000
01010111	87	0000	0000	0000	0000	0000	0001
01011000	88	0000	0000	0001	0101	1000	0111
01011001	89	0000	0000	0001	0110	0000	1101
01011010	90	0000	0000	0001	0110	1000	0001
01011011	91	0000	0000	0001	0111	0000	0011
01011100	92	0001	0000	1000	0000	0000	0100
01011101	93	0011	1001	0111	1111	1100	0111
01011110	94	0000	0000	0000	0000	0000	0000
01011111	95	0011	0010	0101	1111	1000	0100

Table 6-5. Contents of Output Control Memory *(continued)*

Address		Memory Contents					
Binary	Decimal	Location K5	Location H5	Location F5	Location D5	Location C5	Location A5
01100000	96	0000	0000	0000	0000	0000	0000
01100001	97						
01100010	98						
01100011	99						
01100100	100						
01100101	101						
01100110	102						
01100111	103						
01101000	104						
01101001	105						
01101010	106						
01101011	107						
01101100	108						
01101101	109						
01101110	110						
01101111	111						
01110000	112						
01110001	113						
01110010	114						
01110011	115						
01110100	116						
01110101	117						
01110110	118						
01110111	119						
01111000	120						
01111001	121						
01111010	122						
01111011	123						
01111100	124						
01111101	125						
01111110	126						
01111111	127	0000	0000	0000	0000	0000	0000
10000000	128	1111	1111	1000	0000	0000	0111
10000001	129	0001	0110	1101	0100	1000	0111
10000010	130	0001	0110	0100	0000	0000	0110
10000011	131	0000	0000	0011	1100	1101	1111
10000100	132	0000	0101	0111	1111	1100	0111
10000101	133	0000	0110	0111	1111	1101	0101
10000110	134	0000	0000	0000	1010	1000	0100
10000111	135	0000	0001	0000	1001	0000	1110
10001000	136	0000	0010	0000	1001	0000	1110
10001001	137	0001	0000	0000	1001	0011	0100
10001010	138	0000	0100	0000	1001	0010	1001
10001011	139	0000	0101	0000	1001	0010	1001
10001100	140	0001	0101	0000	1001	0010	1001
10001101	141	0000	0000	0001	1111	1000	0100
10001110	142	0001	0110	1000	0000	0000	0111
10001111	143	0001	0000	0111	1111	1100	0111
10010000	144	0001	0110	1010	1001	0000	1111
10010001	145	0000	0000	0001	0111	0000	0011
10010010	146	0000	0011	0000	1001	0001	1011
10010011	147	0001	0111	0000	1001	0001	1011

Table 6-5. Contents of Output Control Memory *(continued)*

Address		Memory Contents					
Binary	Decimal	Location K5	Location H5	Location F5	Location D5	Location C5	Location A5
10010100	148	0001	0000	0000	1001	0001	1110
10010101	149	0001	1111	0000	1001	0010	1100
10010110	150	0000	0101	0000	1001	0010	1000
10010111	151	0000	0100	0000	1001	0010	1000
10011000	152	0000	0000	0000	1001	1010	0011
10011001	153	0000	0000	0000	1000	0011	1000
10011010	154	0000	0000	0001	1111	1000	1111
10011011	155	0000	0000	0000	1000	0001	1101
10011100	156	0010	1001	0111	1111	1100	1011
10011101	157	0001	1100	0111	1111	1101	0000
10011110	158	0000	0000	0000	1000	0010	0111
10011111	159	0010	0000	0111	1111	1100	0111
10100000	160	0010	0001	0111	1111	1101	0101
10100001	161	0000	0010	0000	1001	0101	1011
10100010	162	0000	0000	0001	1111	1001	0000
10100011	163	0000	0000	0001	0100	1000	0111
10100100	164	0000	0000	0011	0100	0000	0111
10100101	165	0001	0110	0101	1111	1001	1001
10100110	166	0000	0000	0000	0000	0000	0000
10100111	167	0011	1101	0111	1111	1101	0001
10101000	168	0000	0000	0000	1000	0011	0011
10101001	169	1111	1111	1001	0100	1000	0111
10101010	170	1111	1111	0100	0000	0000	0111
10101011	171	0011	1011	0111	1100	0101	1000
10101100	172	0000	0000	0000	1000	0011	0001
10101101	173	0000	0000	0000	1010	0000	1110
10101110	174	0010	1111	0111	1111	1100	1011
10101111	175	0000	0000	0000	0000	0000	0110
10110000	176	0000	1110	0111	1111	1101	1000
10110001	177	0000	0000	0000	1010	0001	1000
10110010	178	0010	1110	0111	1111	1101	0000
10110011	179	0010	1001	0111	1111	1101	0000
10110100	180	0011	0101	0111	1111	1100	0111
10110101	181	0011	0110	0111	1111	1101	0101
10110110	182	0000	0010	0000	1001	0101	1100
10110111	183	0000	0000	0001	1111	1010	1001
10111000	184	0000	0000	0001	1100	1011	1101
10111001	185	0000	0000	0000	1100	0100	0000
10111010	186	0000	0000	0001	1111	1001	0001
10111011	187	0000	0000	0001	1111	1100	0011
10111100	188	0000	0000	0000	0000	0000	0000
10111101	189	0000	0000	1011	1111	1101	1101
10111110	190	0000	0000	0000	0000	0000	0000
10111111	191	0000	0000	0000	0000	0000	0000
11000000	192	0000	0000	0001	0101	1000	0111
11000001	193	0000	0000	0001	0110	0000	0101
11000010	194	0000	0000	0001	0110	1000	0111
11000011	195	0000	0000	0001	0111	1000	0111
11000100	196	0000	0000	1111	1111	1000	0000

Table 6-5. Contents of Output Control Memory *(continued)*

Address		Memory Contents					
Binary	Decimal	Location K5	Location H5	Location F5	Location D5	Location C5	Location A5
11000101	197	0000	0000	0000	0000	0000	0000
11000110	198	0000	0000	0000	0000	0000	0000
11000111	199	0000	0000	0001	0101	1000	0111
11001000	200	0000	0000	0001	0101	0000	0000
11001001	201	0000	0000	0001	0111	1000	0111
11001010	202	0000	0000	0001	0100	1000	0001
11001011	203	0000	0000	0001	0111	0000	0010
11001100	204	0000	0000	0001	0100	1000	0111
11001101	205	0000	0000	1010	1011	1101	0111
11001110	206	0000	0000	0001	0111	0000	0010
11001111	207	0000	0000	0001	1111	1101	0011
11010000	208	0000	0000	0000	1011	0001	1000
11010001	209	0001	0000	0100	0000	0000	0111
11010010	210	0000	0000	0000	1100	0100	0000
11010011	211	0000	0000	0001	0100	1000	0111
11010100	212	0000	0000	1010	0000	0000	0001
11010101	213	0000	0000	1010	0000	0000	0111
11010110	214	0000	0000	0000	1100	0100	0000
11010111	215	0000	0000	0000	0000	0000	0001
11011000	216	0000	0000	0001	0101	1000	0111
11011001	217	0000	0000	0001	0110	0000	1101
11011010	218	0000	0000	0001	0110	1000	0001
11011011	219	0000	0000	0001	0111	0000	0011
11011100	220	0001	0000	1000	0000	0000	0100
11011101	221	0011	1001	0111	1111	1100	0111
11011110	222	0000	0000	0000	0000	0000	0000
11011111	223	0001	0110	0101	1111	1000	0100

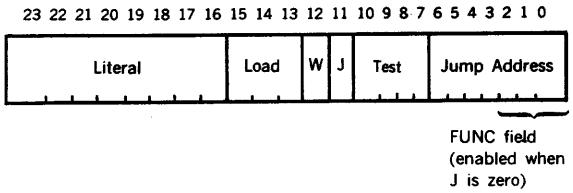
Table 6-5. Contents of Output Control Memory *(continued)*

Address		Memory Contents					
Binary	Decimal	Location K5	Location H5	Location F5	Location D5	Location C5	Location A5
11100000	224	0000	0000	0000	0000	0000	0000
11100001	225						
11100010	226						
11100011	227						
11100100	228						
11100101	229						
11100110	230						
11100111	231						
11101000	232						
11101001	233						
11101010	234						
11101011	235						
11101100	236						
11101101	237						
11101110	238						
11101111	239						
11110000	240						
11110001	241						
11110010	242						
11110011	243						
11110100	244						
11110101	245						
11110110	246						
11110111	247						
11111000	248						
11111001	249						
11111010	250						
11111011	251						
11111100	252						
11111101	253						
11111110	254						
11111111	255	0000	0000	0000	0000	0000	0000

6.2.20 BSC LAD Input-Control Program

Data and control information are received by the LAD in the form of an input-control word. The input control word is described in figure 6-7. Figure 6-8 shows the sequences the LAD follows during the operation of the input-control program. Table 6-6 lists the sequences stored at the various addresses of the input control memory.

Table 6-7 lists the contents of the input control memory. As in the output control memory, this memory consists of 6 ICs each containing 256 4-bit words. Location designations are given in table 6-7 as an aid in locating the memory ICs on the PC board.



Fields are described below:

Load	Description
000	No operation
001	Trigger the receiver 3-second timer
010	Load the line error register with the literal contents (bit 16 enables LERR+, bit 18 enables AI, and bit 17 enables BI)
011	Transfer the contents of the literal field to the return register (Literal must be six bits or less)
100	Clear the CRC generator and the input-transparent-mode flip-flop
101	Transfer received data to the input-shift register and clear the received-data-available flag of the synchronous receiver/transmitter.
110	Reset receiver section of the synchronous receiver/transmitter
111	Clear the receiver bit of the control register

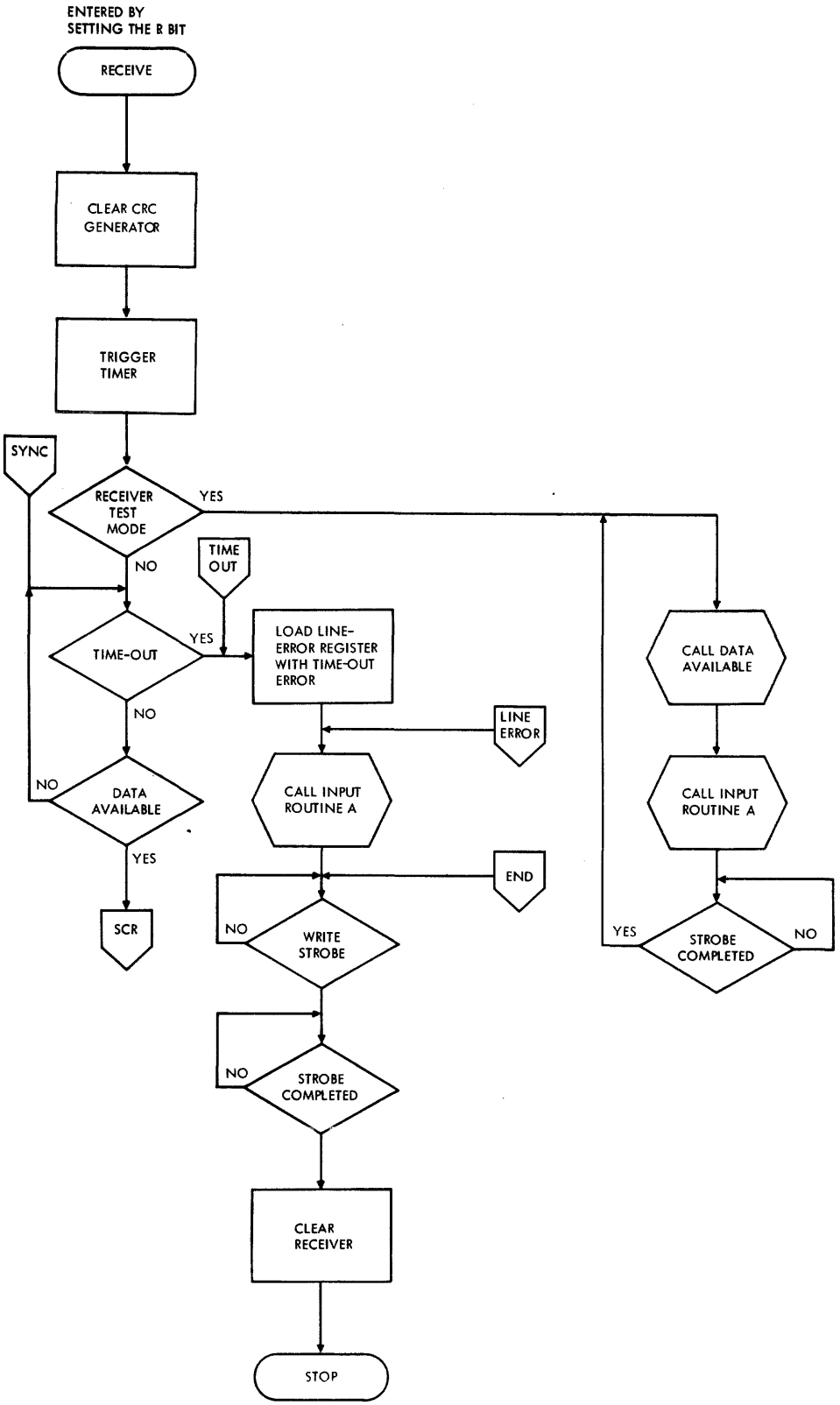
FUNC	Description
	(Note: FUNC field is decoded only if J is zero)
000	Transfer an input data request to the service request logic
001	Transfer the contents of the return register to the control memory address counter
010	Enable the shift-right mode of the CRC generator
011	Enable the CRC accumulation
100	Set the input-transparent-mode flip-flop
101	Transfer a status input request to the service request logic
110	No operation
111	Transfers the busy flag of the service-request to the output control section

Test	Description*
0000	Contents of input CRC generator equal zero
0001	Receiver parity error
0010	Contents of the literal field match the contents of the input-shift register
0011	Receiver timer has timed out or LAD is in receiver test mode
0100	NBC bit of the control register is set
0101	IIB bit of the control register is set
0110	Receiver section of the synchronous transmitter/receiver has overrun
0111	UNT bit of the mode-selection register is set
1000	A line error has been loaded in the line-error register
1001	Received data is available
1010	Input transparent flip-flop is reset
1011	Service-request ligic is not busy
1100	Read strobe is true
1101	Write strobe is true
1110	CRC shift completion
1111	Strobe completion

* When W is one, a wait condition occurs

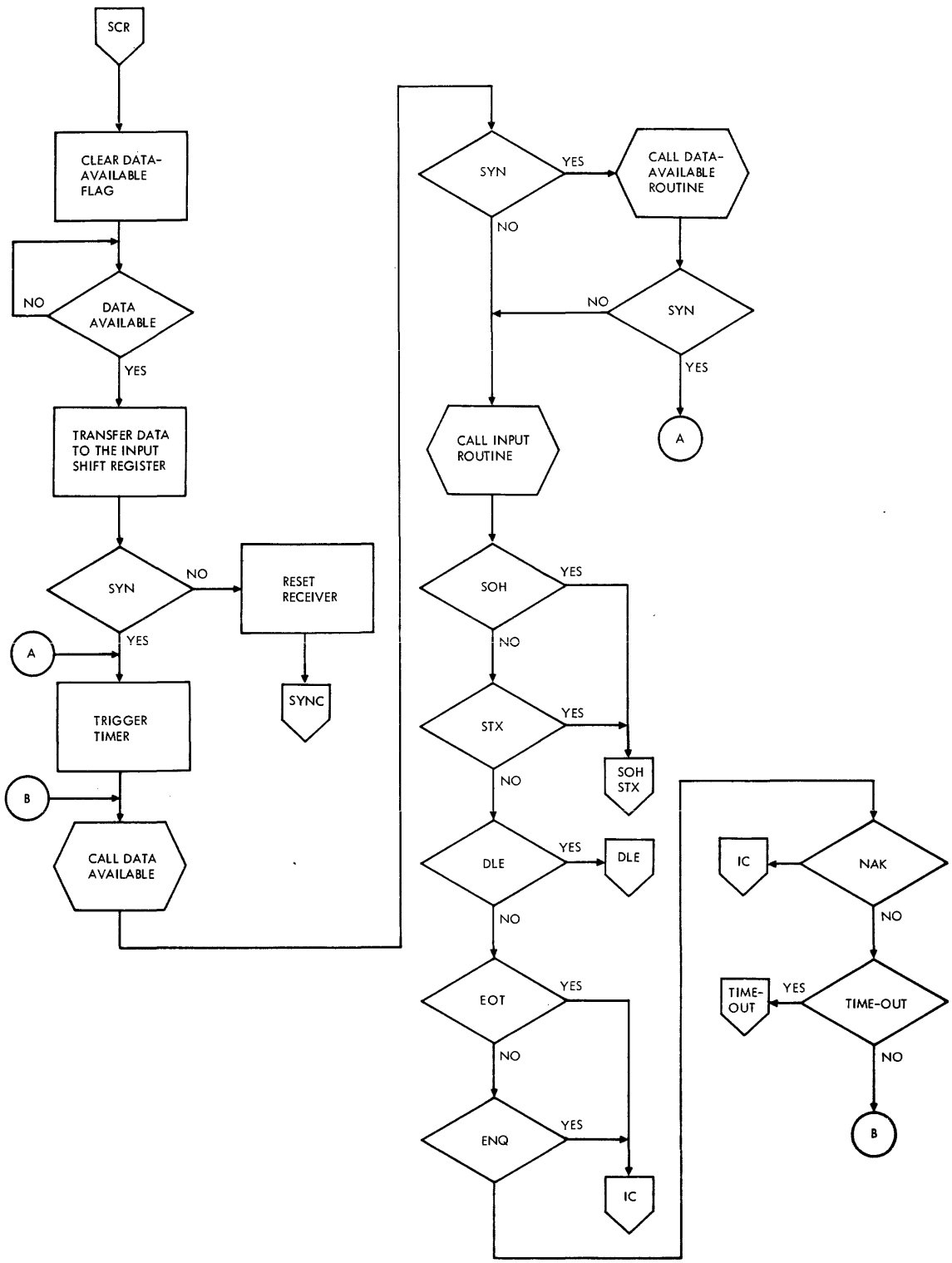
* When J is one, a jump condition occurs

Figure 6-7. Input-Control Word Format



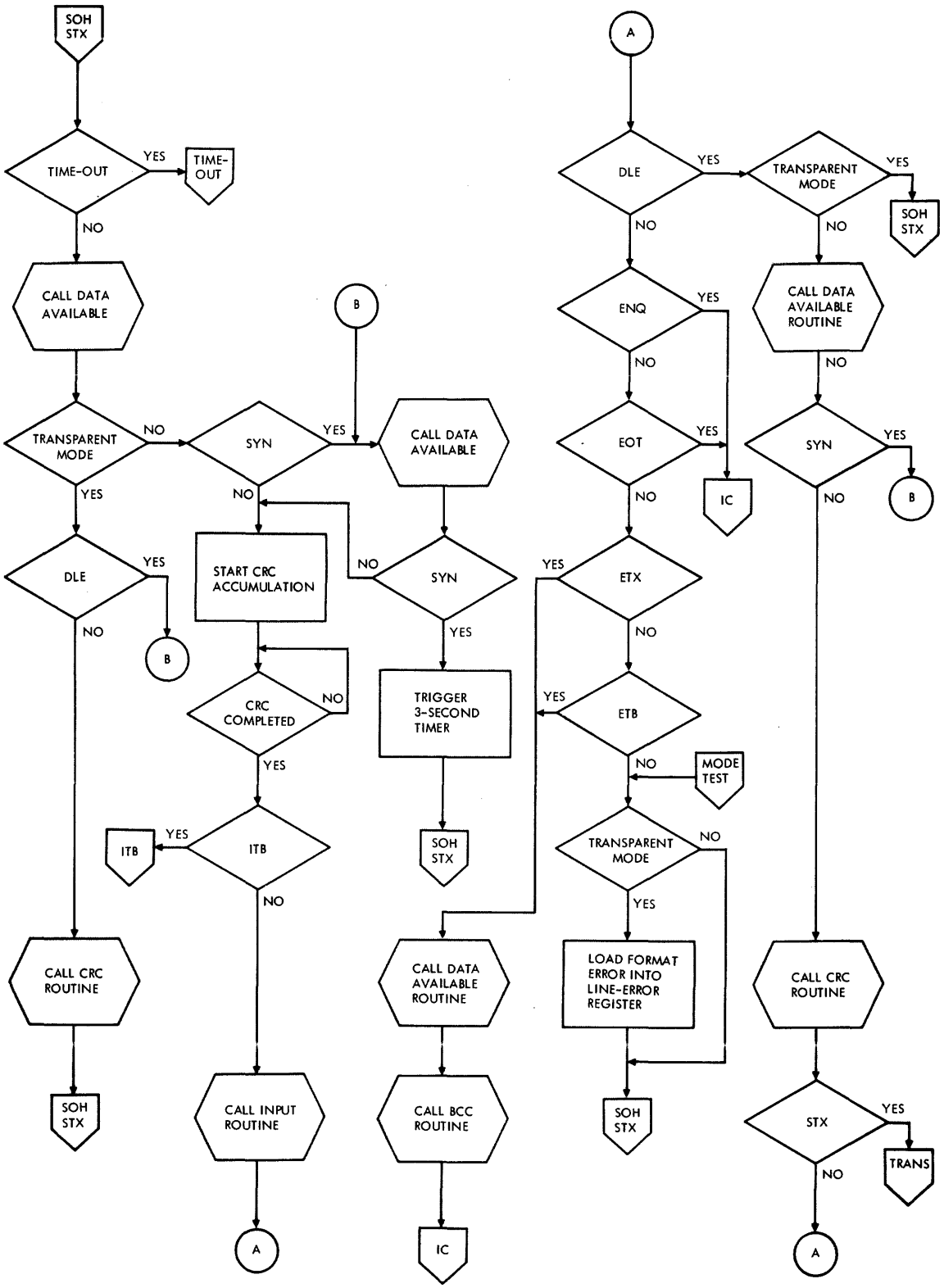
VTI2-431

Figure 6-8. Input-Control Sequences *(sheet 1 of 5)*



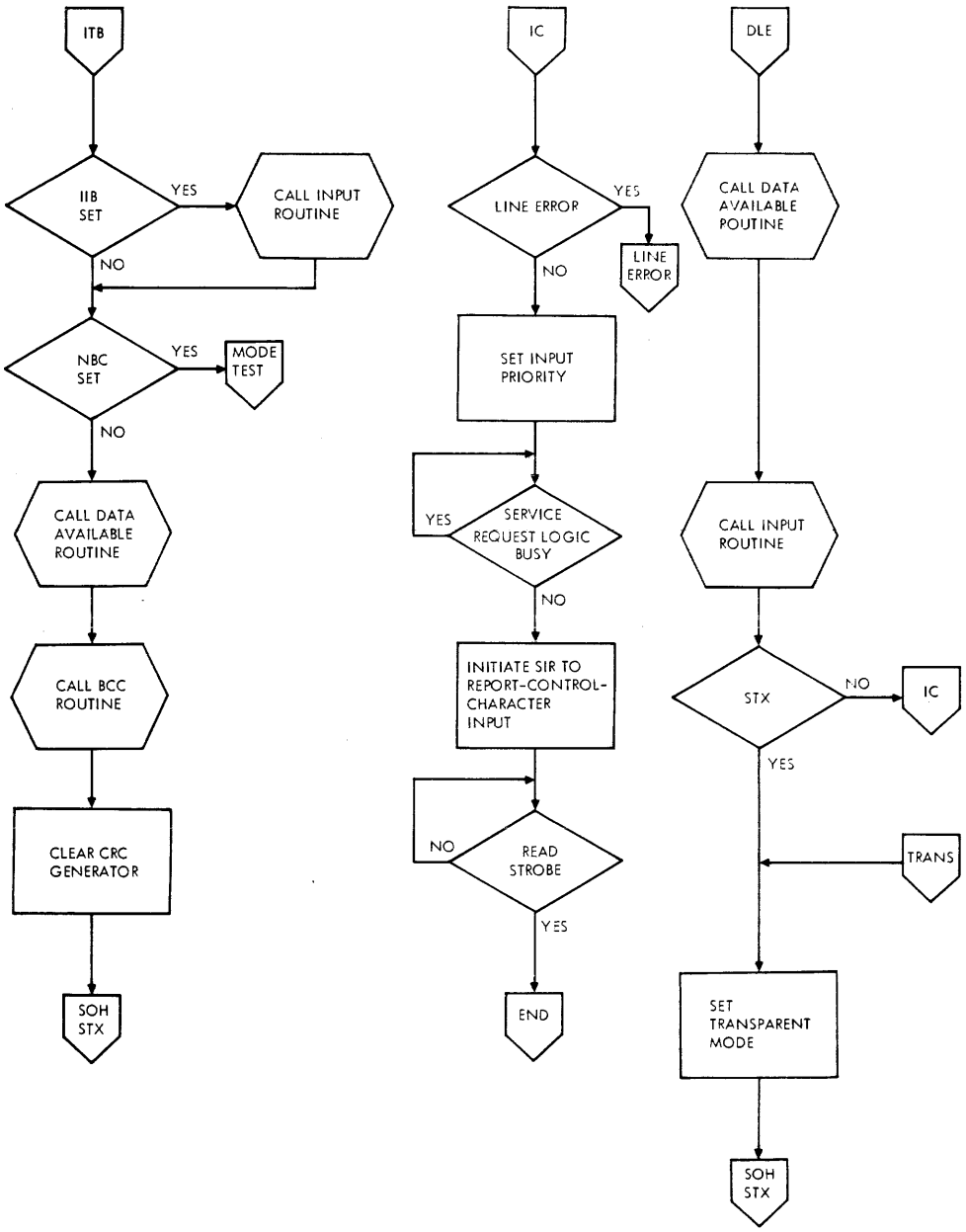
VTI2-432

Figure 6-8. Input-Control Sequences *(sheet 2 of 5)*



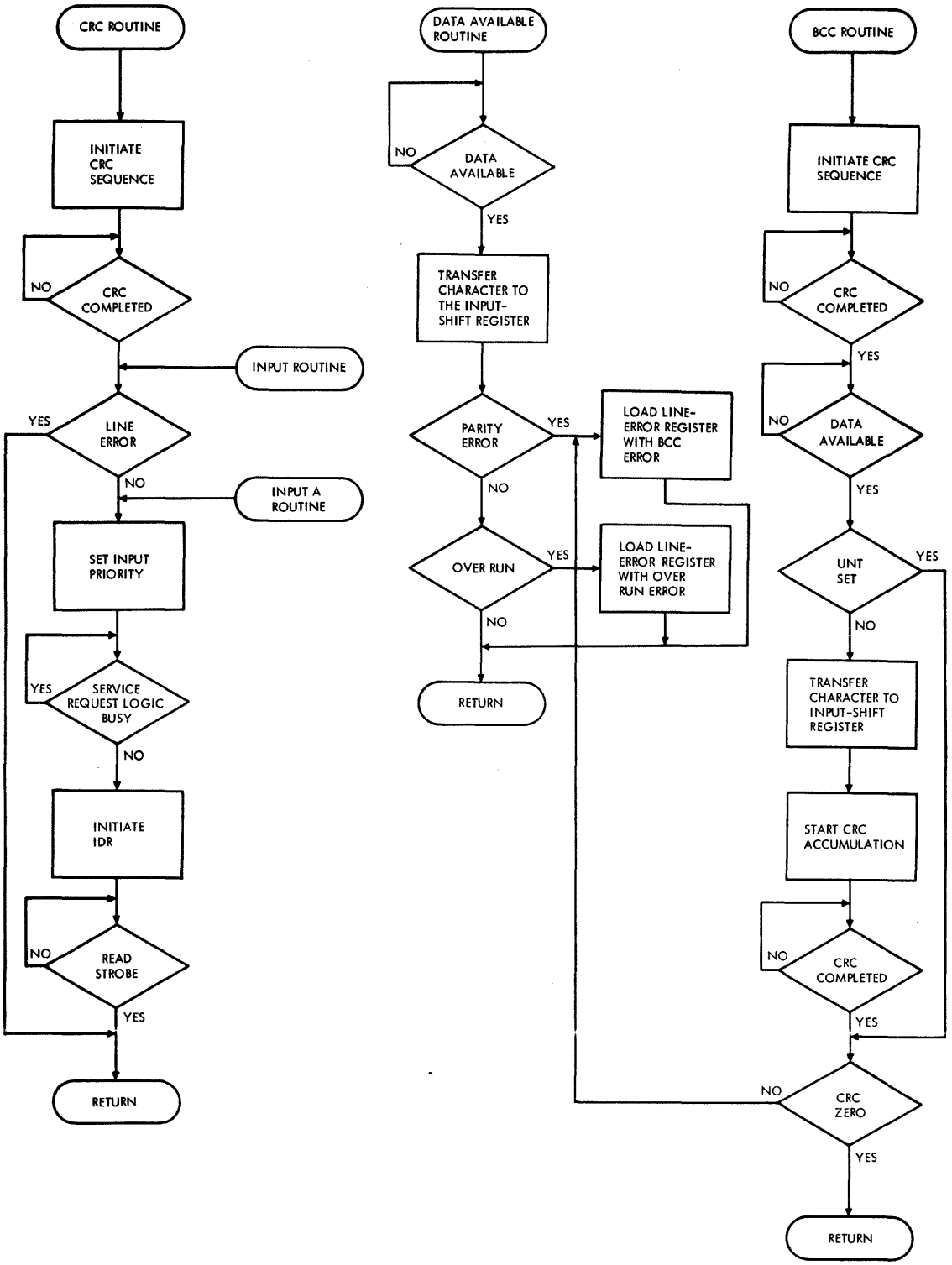
VTI2-433

Figure 6-8. Input-Control Sequences *(sheet 3 of 5)*



VTI2-434

Figure 6-8. Input-Control Sequences *(sheet 4 of 5)*



VTI2-435

Figure 6-8. Input-Control Sequences *(sheet 5 of 5)*

Table 6-6. Input-Control Program

7-Bit Address	Remarks	Literal	Load	W	J	Test	Jump Address
00	Clear CRC generator	0 0	4	1	1	F	40
40	Trigger timer	0 0	1	0	0	0	06
41	Test for receiver test mode; reset receiver	0 0	6	0	1	3	01
42	Test for a time-out	0 0	0	0	1	3	04
43	Test data available	0 0	0	0	1	9	48
44	Jump back to 42	0 0	0	1	1	F	42
01	Call data available (load return register)	0 2	3	1	1	F	55
02	Call input routine A (load return register)	0 3	3	1	1	F	52
03	Wait for strobe completion	0 0	0	1	1	F	01
04	Load time-out in line-error register	0 7	2	0	0	0	06
05	Call input routine A (load return register)	3 D	3	1	1	F	52
3D	Wait for writing strobe	0 0	0	1	1	D	46
46	Wait for strobe completion	0 0	0	1	0	F	06
47	Clear receiver; jump to zero	0 0	7	0	1	0	00

Note: Codes are given in hexadecimal within each field.

Table 6-6. Input-Control Program *(continued)*

7-Bit Address	Remarks	Literal	Load	W	J	Test	Jump Address
48	Clear data-available flag	0 0	5	0	0	0	06
49	Wait for data available	0 0	0	1	0	9	06
4A	Transfer data to shift register	0 0	5	0	0	0	06
4B	Test for SYN; jump to A	SYN	0	0	1	2	06
4C	Reset receiver; jump to SYNC	0 0	6	1	1	F	42
06	Trigger timer	0 0	1	0	0	0	06
07	Call data available (load-return register)	0 8	3	1	1	F	55
08	Test for SYN	SYN	0	0	1	2	12
09	Call input routine (load-return register)	0 A	3	1	1	F	51
0A	Test for SOH; jump to SOH/STX	SOH	0	0	1	2	16
0B	Test for STX; jump to SOH/STX	STX	0	0	1	2	16
0C	Test for DLE; jump to DLE	DLE	0	0	1	2	30
0D	Test for EOT; jump to IC	EOT	0	0	1	2	33
0E	Test for ENQ; jump to IC	ENQ	0	0	1	2	33
0F	Test for NAK; jump to IC	NAK	0	0	1	2	33
10	Test for time-out; jump to time-out	0 0	0	0	1	3	04
11	Jump to B	0 0	0	1	1	F	07
12	Call data available (load return register)	1 3	3	1	1	F	55
13	Test for SYN	SYN	0	0	1	2	06
14	Jump back to 04	0 0	0	1	1	F	09

Note: Codes are given in hexadecimal within each field.

Table 6-6. Input-Control Program *(continued)*

7-Bit Address	Remarks	Literal	Load	W	J	Test	Jump Address
15	Test for transparent mode not	0 0	0	0	1	A	23
16	Test for time-out	0 0	0	0	1	3	04
17	Call data available	1 8	3	1	1	F	55
18	Test for transparent mode not	0 0	0	0	1	A	1B
19	Test for DLE; jump to B	DLE	0	0	1	2	1F
1A	Call CRC routine; return to SOH/STX	1 6	3	1	1	F	50
1B	Test for SYN; jump to B	SYN	0	0	1	2	1F
1C	Start CRC accumulation; wait for completion	0 0	0	1	0	E	03
1D	Test for ITB; jump to ITB	ITB	0	0	1	2	37
1E	Call input routine; return to A	2 7	3	1	1	F	51
1F	Call data available; return	2 0	3	1	1	F	55
20	Test for SYN	SYN	0	0	1	2	22
21	Jump back to IC	0 0	0	1	1	F	1C
22	Trigger timer; jump to SOH/STX	0 0	1	1	1	F	16
23	Call data available; return	2 4	3	1	1	F	55
24	Test for SYN; jump to B	SYN	0	0	1	2	1F
25	Call CRC routine; return	2 6	3	1	1	F	50
26	Test for STX; jump to TRANS	STX	0	0	1	2	4D
27	Test for DLE; jump to 15	DLE	0	0	1	2	15
28	Test for ENQ; jump to IC	ENQ	0	0	1	2	33
29	Test for EOT; jump to IC	EOT	0	0	1	2	33
2A	Test for ETX; jump to 2E	ETX	0	0	1	2	2E
2B	Test for ETB; jump to 2E	ETB	0	0	1	2	2E
2C	Test for transparent mode not; jump to SOH/STX	0 0	0	0	1	A	16
2D	Format error; load line error register; jump to SOH/STX	0 1	2	1	1	F	16
2E	Call data available; return	2 F	3	1	1	F	55
2F	Call BCC routine; return to IC	3 3	3	1	1	F	5A

Note: Codes are given in hexadecimal within each field.

Table 6-6. Input-Control Program *(continued)*

7-Bit Address	Remarks	Literal	Load	W	J	Test	Jump Address
4D	Set transparent mode	0 0	0	0	0	0	04
4E	Jump to SOH/STX	0 0	0	1	1	F	16
30	Call data available, return	3 1	3	1	1	F	55
31	Call input routine; return	3 2	3	1	1	F	51
32	Test for STX; jump to TRANS	STX	0	0	1	2	4D
33	Test for line error; jump to line error	0 0	0	0	1	8	05
34	Set input priority; wait	0 0	0	1	0	B	07
35	Initiate SIR to report-control character input	0 0	0	1	0	C	05
36	Jump to END	0 0	0	1	1	F	3D
37	Test ITB bit; jump to 3C	0 0	0	0	1	5	3C
38	Test NBC bit; jump to mode test	0 0	0	0	1	4	2C
39	Call data available; return	3 A	3	1	1	F	55
3A	Call BCC routine; return	3 B	3	1	1	F	5A
3B	Clear CRC generator; jump to SOH/STX	0 0	4	1	1	F	16
3C	Call input routine; return	3 8	3	1	1	F	51
50	Start CRC sequence; wait for completion	0 0	0	1	0	E	03
51	Test for line error; jump to 54	0 0	0	0	1	8	54
52	Set input priority; wait for SRL not busy	0 0	0	1	0	B	07
53	Initate IDR; wait for read strobe	0 0	0	1	0	C	00
54	Return	0 0	0	0	0	0	01
55	Wait for data available	0 0	0	1	0	9	06
56	Transfer character to SR and test for parity error	0 0	5	0	1	1	5F
57	Test for overrun; jump to 59	0 0	0	0	1	6	59
58	Return	0 0	0	0	0	0	01
5F	Load line-error register with BCC error; return	0 3	2	0	0	0	01
59	Load line-error register with overrun; return	0 5	2	0	0	0	01
5A	Start CRC accumulation; wait for completion	0 0	0	1	0	E	03
5B	Wait for data available	0 0	0	1	0	9	06
5C	Transfer character to shift register; test UNT	0 0	5	0	1	7	5E
5D	Start CRC accumulation; wait	0 0	0	1	0	E	03
5E	Test CRC zero; jump to 58	0 0	0	0	1	0	58

Note: Codes are given in hexadecimal within each field.

Table 6-7. Contents of Input Control Memory

Address		Memory Contents					
Binary	Decimal	Location K4	Location H4	Location E4	Location D4	Location C4	Location A4
00000000	0	0000	0000	1001	1111	1100	0000
00000001	1	0000	0010	0111	1111	1101	0101
00000010	2	0000	0011	0111	1111	1101	0010
00000011	3	0000	0000	0001	1111	1000	0001
00000100	4	0000	0111	0100	0000	0000	0110
00000101	5	0011	1101	0111	1111	1101	0010
00000110	6	0000	0000	0010	0000	0000	0110
00000111	7	0000	1000	0111	1111	1101	0101
00001000	8	0011	0010	0000	1001	0001	0010
00001001	9	0000	1010	0111	1111	1101	0001
00001010	10	0000	0001	0000	1001	0001	0110
00001011	11	0000	0010	0000	1001	0001	0110
00001100	12	0001	0000	0000	1001	0011	0000
00001101	13	0011	0111	0000	1001	0011	0011
00001110	14	0010	1101	0000	1001	0011	0011
00001111	15	0011	1101	0000	1001	0011	0011
00010000	16	0000	0000	0000	1001	1000	0100
00010001	17	0000	0000	0001	1111	1000	0111
00010010	18	0001	0011	0111	1111	1101	0101
00010011	19	0011	0010	0000	1001	0000	0110
00010100	20	0000	0000	0001	1111	1000	1001
00010101	21	0000	0000	0000	1101	0010	0011
00010110	22	0000	0000	0000	1001	1000	0100
00010111	23	0001	1000	0111	1111	1101	0101
00011000	24	0000	0000	0000	1101	0001	1011
00011001	25	0001	0000	0000	1001	0001	1111
00011010	26	0001	0110	0111	1111	1101	0000
00011011	27	0011	0010	0000	1001	0001	1111
00011100	28	0000	0000	0001	0111	0000	0011
00011101	29	0001	1111	0000	1001	0011	0111
00011110	30	0010	0111	0111	1111	1101	0001
00011111	31	0010	0000	0111	1111	1101	0101
00100000	32	0011	0010	0000	1001	0010	0010
00100001	33	0000	0000	0001	1111	1001	1100
00100010	34	0000	0000	0011	1111	1001	0110
00100011	35	0010	0100	0111	1111	1101	0101
00100100	36	0011	0010	0000	1001	0001	1111
00100101	37	0010	0110	0111	1111	1101	0000
00100110	38	0000	0010	0000	1001	0100	1101
00100111	39	0001	0000	0000	1001	0001	0101
00101000	40	0010	1101	0000	1001	0011	0011
00101001	41	0011	0111	0000	1001	0011	0011
00101010	42	0000	0011	0000	1001	0010	1110
00101011	43	0010	0110	0000	1001	0010	1110
00101100	44	0000	0000	0000	1101	0001	0110
00101101	45	0000	0001	0101	1111	1001	0110
00101110	46	0010	1111	0111	1111	1101	0101
00101111	47	0011	0011	0111	1111	1101	1010
00100000	48	0011	0001	0111	1111	1101	0101
00110001	49	0011	0010	0111	1111	1101	0001
00110010	50	0000	0010	0000	1001	0100	1101
00110011	51	0000	0000	0000	1100	0000	0101
00110100	52	0000	0000	0001	0101	1000	0111

Table 6-7. Contents of Input Control Memory *(continued)*

Address		Memory Contents					
Binary	Decimal	Location K4	Location H4	Location E4	Location D4	Location C4	Location A4
00110101	53	0000	0000	0001	0110	0000	0101
00110110	54	0000	0000	0001	1111	1011	1101
00110111	55	0000	0000	0000	1010	1011	1100
00111000	56	0000	0000	0000	1010	0010	1100
00111001	57	0011	1010	0111	1111	1101	0101
00111010	58	0011	1011	0111	1111	1101	1010
00111011	59	0000	0000	1001	1111	1001	0110
00111100	60	0011	1000	0111	1111	1101	0001
00111101	61	0000	0000	0001	1110	1100	0110
00111110	62	0000	0000	0000	0000	0000	0000
00111111	63	0000	0000	0000	0000	0000	0000
01000000	64	0000	0000	0010	0000	0000	0110
01000001	65	0000	0000	1100	1001	1000	0001
01000010	66	0000	0000	0000	1001	1000	0100
01000011	67	0000	0000	0000	1100	1100	1000
01000100	68	0000	0000	0001	1111	1100	0010
01000101	69	0000	0000	0000	0000	0000	0000
01000110	70	0000	0000	0001	0111	1000	0110
01000111	71	0000	0000	1110	1000	0000	0000
01001000	72	0000	0000	1010	0000	0000	0110
01001001	73	0000	0000	0001	0100	1000	0110
01001010	74	0000	0000	1010	0000	0000	0110
01001011	75	0011	0010	0000	1001	0000	0110
01001100	76	0000	0000	1101	1111	1100	0010
01001101	77	0000	0000	0000	0000	0000	0100
01001110	78	0000	0000	0001	1111	1001	0110
01001111	79	0000	0000	0000	0000	0000	0000
01010000	80	0000	0000	0001	0111	0000	0011
01010001	81	0000	0000	0000	1100	0101	0100
01010010	82	0000	0000	0001	0101	1000	0111
01010011	83	0000	0000	0001	0110	0000	0000
01010100	84	0000	0000	0000	0000	0000	0001
01010101	85	0000	0000	0001	0100	1000	0110
01010110	86	0000	0000	1010	1000	1101	1111
01010111	87	0000	0000	0000	1011	0101	1001
01011000	88	0000	0000	0000	0000	0000	0001
01011001	89	0000	0101	0100	0000	0000	0001
01011010	90	0000	0000	0001	0111	0000	0011
01011011	91	0000	0000	0001	0100	1000	0110
01011100	92	0000	0000	1010	1011	1101	1110
01011101	93	0000	0000	0001	0111	0000	0011
01011110	94	0000	0000	0000	1000	0101	1000
01011111	95	0000	0011	0100	0000	0000	0001

Table 6-7. Contents of Input Control Memory *(continued)*

Address		Memory Contents					
Binary	Decimal	Location K4	Location H4	Location E4	Location D4	Location C4	Location A4
01100000	96	0000	0000	0000	0000	0000	0000
01100001	97						
01100010	98						
01100011	99						
01100100	100						
01100101	101						
01100110	102						
01100111	103						
01101000	104						
01101001	105						
01101010	106						
01101011	107						
01101100	108						
01101101	109						
01101110	110						
01101111	111						
01110000	112						
01110001	113						
01110010	114						
01110011	115						
01110100	116						
01110101	117						
01110110	118						
01110111	119						
01111000	120						
01111001	121						
01111010	122						
01111011	123						
01111100	124						
01111101	125						
01111110	126						
01111111	127	0000	0000	0000	0000	0000	0000
10000000	128	0000	0000	1001	1111	1100	0000
10000001	129	0000	0010	0111	1111	1101	0101
10000010	130	0000	0011	0111	1111	1101	0010
10000011	131	0000	0000	0001	1111	1000	0001
10000100	132	0000	0111	0100	0000	0000	0110
10000101	133	0011	1101	0111	1111	1101	0010
10000110	134	0000	0000	0010	0000	0000	0110
10000111	135	0000	1000	0111	1111	1101	0101
10001000	136	0001	0110	0000	1001	0001	0010
10001001	137	0000	1010	0111	1111	1101	0001
10001010	138	0000	0001	0000	1001	0001	0110
10001011	139	0000	0010	0000	1001	0001	0110
10001100	140	0001	0000	0000	1001	0011	0000
10001101	141	0000	0100	0000	1001	0011	0011
10001110	142	0000	0101	0000	1001	0011	0011
10001111	143	0001	0101	0000	1001	0011	0011
10010000	144	0000	0000	0000	1001	1000	0100
10010001	145	0000	0000	0001	1111	1000	0111
10010010	146	0001	0011	0111	1111	1101	0101
10010011	147	0001	0110	0000	1001	0000	0110
10010100	148	0000	0000	0001	1111	1000	1001

Table 6-7. Contents of Input Control Memory *(continued)*

Address		Memory Contents					
Binary	Decimal	Location K4	Location H4	Location E4	Location D4	Location C4	Location A4
10010101	149	0000	0000	0000	1101	0010	0011
10010110	150	0000	0000	0000	1001	1000	0100
10010111	151	0001	1000	0111	1111	1101	0101
10011000	152	0000	0000	0000	1101	0001	1011
10011001	153	0001	0000	0000	1001	0001	1111
10011010	154	0001	0110	0111	1111	1101	0000
10011011	155	0001	0110	0000	1001	0001	1111
10011100	156	0000	0000	0001	0111	0000	0011
10011101	157	0001	1111	0000	1001	0011	0111
10011110	158	0010	0111	0111	1111	1101	0001
10011111	159	0010	0000	0111	1111	1101	0101
10100000	160	0001	0110	0000	1001	0010	0010
10100001	161	0000	0000	0001	1111	1001	1100
10100010	162	0000	0000	0011	1111	1001	0110
10100011	163	0010	0100	0111	1111	1101	0101
10100100	164	0001	0110	0000	1001	0001	1111
10100101	165	0010	0110	0111	1111	1101	0000
10100110	166	0000	0010	0000	1001	0100	1101
10100111	167	0001	0000	0000	1001	0001	0101
10101000	168	0000	0101	0000	1001	0011	0011
10101001	169	0000	0100	0000	1001	0011	0011
10101010	170	0000	0011	0000	1001	0010	1110
10101011	171	0001	0111	0000	1001	0010	1110
10101100	172	0000	0000	0000	1101	0001	0110
10101101	173	0000	0001	0101	1111	1001	0110
10101110	174	0010	1111	0111	1111	1101	0101
10101111	175	0011	0011	0111	1111	1101	1010
10110000	176	0011	0001	0111	1111	1101	0101
10110001	177	0011	0010	0111	1111	1101	0001
10110010	178	0000	0010	0000	1001	0100	1101
10110011	179	0000	0000	0000	1100	0000	0101
10110100	180	0000	0000	0001	0101	1000	0111
10110101	181	0000	0000	0001	0110	0000	0101
10110110	182	0000	0000	0001	1111	1011	1101
10110111	183	0000	0000	0000	1010	1011	1100
10111000	184	0000	0000	0000	1010	0010	1100
10111001	185	0011	1010	0111	1111	1101	0101
10111010	186	0011	1011	0111	1111	1101	1010
10111011	187	0000	0000	1001	1111	1001	0110
10111100	188	0011	1000	0111	1111	1101	0001
10111101	189	0000	0000	0001	1110	1100	0110
10111110	190	0000	0000	0000	0000	0000	0000
10111111	191	0000	0000	0000	0000	0000	0000

Table 6-7. Contents of Input Control Memory *(continued)*

Address		Memory Contents					
Binary	Decimal	Location K4	Location H4	Location E4	Location D4	Location C4	Location A4
11000000	192	0000	0000	0010	0000	0000	0110
11000001	193	0000	0000	1100	1001	1000	0001
11000010	194	0000	0000	0000	1001	1000	0100
11000011	195	0000	0000	0000	1100	1100	1000
11000100	196	0000	0000	0001	1111	1100	0010
11000101	197	0000	0000	0000	0000	0000	0000
11000110	198	0000	0000	0001	0111	1000	0110
11000111	199	0000	0000	1110	1000	0000	0000
11001000	200	0000	0000	1010	0000	0000	0110
11001001	201	0000	0000	0001	0100	1000	0110
11001010	202	0000	0000	1010	0000	0000	0110
11001011	203	0001	0110	0000	1001	0000	0110
11001100	204	0000	0000	1101	1111	1100	0010
11001101	205	0000	0000	0000	0000	0000	0100
11001110	206	0000	0000	0001	1111	1001	0110
11001111	207	0000	0000	0000	0000	0000	0000
11010000	208	0000	0000	0001	0111	0000	0011
11010001	209	0000	0000	0000	1100	0101	0100
11010010	210	0000	0000	0001	0101	1000	0111
11010011	211	0000	0000	0001	0110	0000	0000
11010100	212	0000	0000	0000	0000	0000	0001
11010101	213	0000	0000	0001	0100	1000	0110
11010110	214	0000	0000	1010	1000	1101	1111
11010111	215	0000	0000	0000	1011	0101	1001
11011000	216	0000	0000	0000	0000	0000	0001
11011001	217	0000	0101	0100	0000	0000	0001
11011010	218	0000	0000	0001	0111	0000	0011
11011011	219	0000	0000	0001	0100	1000	0110
11011100	220	0000	0000	1010	1011	1101	1110
11011101	221	0000	0000	0001	0111	0000	0011
11011110	222	0000	0000	0000	1000	0101	1000
11011111	223	0000	0011	0100	0000	0000	0001

Table 6-7. Contents of Input Control Memory *(continued)*

Address		Memory Contents					
Binary	Decimal	Location K4	Location H4	Location E4	Location D4	Location C4	Location A4
11100000	224	0000	0000	0000	0000	0000	0000
11100001	225						
11100010	226						
11100011	227						
11100100	228						
11100101	229						
11100110	230						
11100111	231						
11101000	232						
11101001	233						
11101010	234						
11101011	235						
11101100	236						
11101101	237						
11101110	238						
11101111	239						
11110000	240						
11110001	241						
11110010	242						
11110011	243						
11110100	244						
11110101	245						
11101110	246						
11110111	247						
11111000	248						
11111001	249						
11111010	250						
11111011	251						
11111100	252						
11111101	253						
11111110	254						
11111111	255	0000	0000	0000	0000	0000	0000

6.3 MAINTENANCE

The following equipment is required to maintain the LAD.

a. Varian 620 or V70 series computer with at least 8K of memory.

b. Data communications multiplexor.

c. An I/O expansion chassis.

d. An I/O expansion power supply.

e. One card extender (44D0015-000 or 44D0540-000).

f. Oscilloscope Tektronix type 547 or equivalent.

g. A BSC RS232 test connector or a wide band test connector (VDM connector 57A000036-000) wired as follows:

RS232 Test Connector:

Data Terminal Ready	29	33	Dataset-ready
		25	Carrier on
Request-to-send	27	37	Clear-to-send
Transmitted data	31	23	Received data
Control line #1	20	18	Status line
Control line #2	17	35	Ring indicator
Test clock 1	38	39	Transmitter clock
Test clock 2	10	41	Receiver clock

Wide Band Test Connector:

Transmitted data	(5)	(23)	Received data
Send request	(3)	(37)	Clear-to-send
Control line #1	(7)	(18)	Status line in
(local test)		(33)	Dataset-ready
		(25)	Carrier on
Control line #2	(17)	(35)	Ring indicator
Wide band test clock	(44)	(39)	Transmitter clock
		(41)	Receiver clock

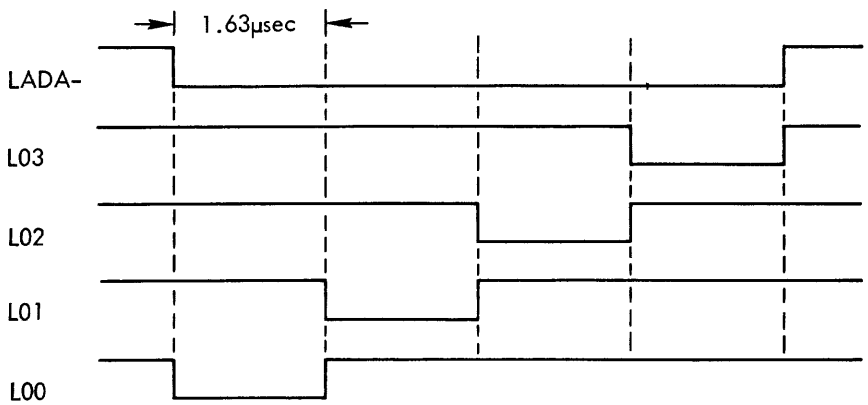
h. Software tapes:

MAINTAIN II Test Executive	92 U 0107-001
DCM Test Program	92 U 0106-009B

6.3.1 Functional Tests

The following tests use portions of the DCM BSC test program. The specific test required is called out in each section. Data pattern 2 should be used for all tests unless otherwise specified.

Line address: Verify that the LAD address decoder logic is functioning properly by verifying that it can respond to any of the four line addresses in the group defined by the line-address input signal LADA– on pin 36. This can be done by observing enabling signals L03, L02, L01, and L00 on pins 111, 112, 113, and 114, respectively, while the multiplexor is scanning and no lines are active (see waveforms below). Press the system reset on the computer control panel to clear the multiplexor.



VTII-3242

1-second timer: Set up the line control block for a zero output byte count and place the LAD in the transmitting mode. Verify that the 1-second timer is triggered and the output remains true for a minimum of 0.8 seconds and not more than 1.1 seconds. Test 1 of the test program can also be used to trigger the timer.

3-second timer: Place the LAD in the receive-only mode and verify that the 3-second timer goes true for a minimum of three seconds, but not more than four. Test 7 of the test program can be used to trigger the timer if the test is set up for only one pass.

New-sync timer: Enable the control line 2 (C2 of the LADs control register) and verify that the new-sync timer (NSYNC+) goes true for a minimum of 1.0 milliseconds, but not more than 3.0 milliseconds. Test 11 of the test program running continuously can be used to trigger the timer.

BSC control character test: Place the LAD in the transmitting mode and receiving mode and verify that the LAD can successfully transmit and receive the messages listed below with no errors. Also verify that an output-completion status is given through the status-change interrupt and that receiver-control character status is conveyed through the status-change interrupt after the message has been received. Verify operation in the EBCDIC mode, and both ASCII modes.

a)	SYN SYN SOH ETX	f)	SYN SYN STX EOT
b)	SYN SYN DLE "00"	g)	SYN SYN NAK
c)	SYN SYN STX ENQ	h)	SYN SYN DLE STX ETX
d)	SYN SYN EOT	i)	SYN SYN STX ETB
e)	SYN SYN ENQ	j)	SYN SYN DLE EOT

Successful running of tests 2, 4, and 5 of the test program will thoroughly test the LADs BSC control-character detection and reporting capabilities.

BSC error: Place the transmitter section in the test mode and the receiver section in the normal mode. Formulate a BSC message with a bad block-check character. Output the message and verify that a BCC line error is reported. Test 11 of the test program tests this operation.

Format error: Place the transmitter section in the test mode and the receiver section in the normal mode. Formulate a transparent text message with an invalid DLE sequence imbedded. Output the message and verify that a format line error is reported. Do this for the EBCDIC mode and the ASCII mode with transparent capability. Test 11 of the test program tests this operation.

Overrun error: Place the line in the transmitter and receiver modes and output two SYN characters. Delay responding to the input-byte count zero interrupt for a minimum of two character times. Verify that an overrun line error is reported after the input-byte count zero interrupt is finally serviced. Test 11 of the test program tests this operation.

Time-out error: Place the line in the receiver mode and verify that a time-out error is reported after approximately three seconds. Test 11 of the test program tests this operation.

Modem control signals: Verify that the corresponding modem control signals are enabled. Use the test connector (previously defined) when the control-register bits, data-terminal ready, transmitter, control-line 1, and control-line 2 are enabled.

NOTE: Since control line 2 enables the ring indicator it is necessary to enable the accept-ring signal (AR) to verify correct operation of these signals. When AR and control line 2 are enabled, verify that a status change interrupt occurs.

Successful running of tests 11 and 12 of the test program verifies the correct operation of the modem control signals.

ITB operation: Verify that ITB is not treated as a control character when the nbc bit is set in the control register. Verify correct intermediate block transmission and receipt when the nbc bit is cleared. When received, verify that the IIB bit controls whether or not ITB characters are transferred to memory. Test this operation using the EBCDIC mode and both ASCII modes. Successful running of test 6 and 7 verifies correct ITB operation.

Transparent transmitter operation: Place the line in normal-transmitter and test-receiver modes and verify that the LAD properly inserts DLE characters in the output-data stream where required. Test the EBCDIC mode and the ASCII mode with transparent capability. Use tests 4, 5, 6, and 7 of the test program to verify this.

Insertion of SYNs in text: The BSC LAD inserts two SYN characters in the output-data stream every second while transmitting the BSC normal test. To verify this operation, monitor the output-data stream with an oscilloscope while running test 0 of the test program (with data pattern 0 selected and an external test clock of 50 Hz). Do this for all three data modes. Note: When using modes 1 and 2, the LAD appends a parity bit to the transmitted character so the inserted SYN character is more difficult to detect.

Insertion of DLE SYN in transparent text: The BSC LAD inserts DLE SYN in the output-data stream every second while transmitting BSC transparent text. To verify this operation, monitor the output-data stream with an oscilloscope while running test 0 of the test program (with an external test clock of 50 Hz). Test pattern 4 must be used with the following input:

Data Pattern 4, 10002, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 3.

Do this for mode 0 and mode 2.

Loop-back test: The loop-back feature of the BSC LAD can be tested by running test 3 of the test program with no test shoe on the LAD.

UWT mode parity generation: Place the LAD in the UWT mode and the test-receiver mode. Output an ASCII BSC test message and verify on the received data that the transmitter inserted an odd parity bit in the most-significant-bit position of each character in the message (not in the block-check character). Test 4 of the test program verifies this and test 1 with mode 2 selected provides a visual check.

Data path check: Place the LAD in both the test-receiver and test-transmitter mode, or use normal mode with a transparent text message. Output a message consisting of the full 8-bit ascending or descending binary-data pattern and verify the correct reception of the message. Use test 10 of the test program.

Universal synchronous receiver/transmitter reset pulse: Verify that the reset pulse RR+ is a minimum of 1 microsecond long, but not more than 3.5 microseconds. Use test 4 of the test program to trigger the one shot.

Burn-in test: Run test 77 for at least two passes.

6.3.2 Other Problems

To eliminate problems not solved by the previous tests, the following program can be used as an aid in troubleshooting:

Address (Octal)	Code (Octal)	Instruction	Remarks
000200	002000	JMPM	Call LCB setup
000201	000215		
000202 *	100070	EXC 070	Initialize
000203 *	100270	EXC 270	Enable interrupts
000204 *	100570	EXC 570	Request control write
000205	005000	NOP	
000206	001000	JMP	Wait for interrupt
000207	000205		
	LCB Setup		
000215	000000		
000216	006050	STAI	Store A
000217	000000		
000220	006010	LDAI	Load A
000221 **	107770		Input byte-count word
000222	057500	STA	Store A in LCB
000223	006010	LDAI	Load A
000224 **	000600		Input buffer-address word
000225	057501	STA	Store A in LCB
000226	006010	LDAI	Load A
000227 **	107770		Output byte-count word
000230	057502	STA	Store A in LCB
000231	006010	LDAI	Load A
000232 **	000650		Output buffer-address word
000233	057503	STA	Store A
000234	010217	LDA	Load A (restore)
000235	001000	JMP	Jump back
000236	100215		

* Change these instructions to reflect the correct device address if a nonstandard address is used.

** Enter the byte counts and buffer addresses desired.

Address (Octal)	Code (Octal)	Instruction	Remarks
		Line Setup	
000240	000000		
000241	006050	STAI	Save A
000242	000000		
000243	017504	LDA	Load A with control word
000244 *	103170	OAR	Output to DCM
000245	010242	LDA	Restore A
000246 *	100270	EXC 270	Enable interrupts
000247	0001000	JMP	Return
000250	1000240		
		Input or Output Byte-Count Zero	
000260	000000		
000261	100070	EXC 070	Initialize
000262	001400	J SS3	Jump to
000263	006221		Test executive
000264	001000	JMP	Return to
000265	000200		Start
		Interrupt Traps	
000060	002000	JMPM	
000061	000260		
000062	002000	JMPM	
000063	000260		
000064	001000	JMP	
000065	000270		
000066	001000	JMP	
000067	000271		
000070	001000	JMP	
000071	000272		
000072	002000	JMPM	
000073	000240		

Address (Octal)	Code (Octal)	Instruction	Remarks
		Interrupt Halts	
000270	000004		Line error interrupt
000272	000001		Control-character interrupt

* Change these instructions to reflect the correct device address if a nonstandard address is used.

LCB Address

The LCB addresses for the line under test are entered here.

Address (Octal)	Code (Octal)	Instruction	Remarks
000500	017000		Input block-length address
000501	017001		Input buffer-address address
000502	017004		Output block-length address
000503	017005		Output buffer-address address
000504	017006		Control-word address
	Line Control Block (Line zero shown - base address 017000)		
017000	107770		Input block-length
017001	000600		Input buffer-address
017002	177777		Control characters
017003	000000		Spare
017004	107770		Output block-length
017005	000650		Output buffer-address
017006	005500		Line control-word (includes line address)
017007	000000		Spare
	Output Buffer		
000650	031062		
000651	000400		
000652	000000		
000653	000003		
	Input Buffer		
000600	000000		
000601	000000		
000602	000000		
000603	000000		

6.4 PROGRAMMING

This section describes only those aspects of programming peculiar to the binary synchronous communications LAD. General DCM programming is discussed in section 2.4.

6.4.1 Interrupts

This section explains the use of the line-error interrupt type-of-error bits and the status-change-error interrupt status bits with regard to the binary synchronous communications LAD. The remaining interrupt information is in section 2.4.2.

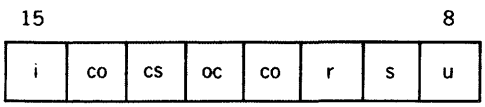
6.4.1.1 Line Error

This interrupt is explained in section 2.4.2.3. For binary synchronous communications LAD, the type-of-error bits a and b (bit 6 and 7) have the following significance.

ab	Type of Error
00	Format error (incorrect character received after DLE in the transparent text mode)
01	BSC or parity error
10	Overrun
11	3-second time-out. The receiver times out if two contiguous SYN characters are not received within three seconds after the LAD enters the receiver mode, or if two contiguous SYN characters (DLE SYN in transparent text) are not received at least every three seconds during the input message.

6.4.1.2 Status Change

This interrupt is explained in section 2.4.2.4. For the binary synchronous communications LAD, the status bits have the format:



where the bits, when set, indicate:

u = Underflow: not normally set, but will be set when the LAD is in the transmitter mode and the transmitter underflows (i.e., it is unable to maintain character sync on the line). This condition is reported via a status-change interrupt. When not in the test mode, the LAD will not report underflow, if it can maintain character sync by outputting SYN or DLE SYN characters. Whenever underflow is reported, the OC bit is also set.

s = Status-line-in: conveys the status of the uncommitted input line from the modem. This bit cannot cause an interrupt to be generated.

r = Ring indicator: indicates that the ring-indicator line from the modem is on. If the control byte is set up correctly, this bit causes a status-change interrupt.

cd = Control character detected: a status-change interrupt occurs to convey that this bit has been set. Paragraph 6.6 explains which control characters cause this bit to set. The receiver section uses this status to convey that an ending-control character has been detected and that the receiver sequence has been terminated (i.e., the line has removed itself from the receiver mode). This status is not reported in the receiver-test mode.

oc = Output completion: the LAD sets this bit and initiates a status-change interrupt when the output sequence is complete. Output completion is not reported in the transmitter-test mode unless an underflow occurs.

cs = Clear to send: indicates the status of the clear-to-send line from the modem (it does not initiate an interrupt).

co = Carrier on: indicates the status of the carrier-on from the modem (it does not initiate an interrupt).

i = Interlock (dataset ready): indicates the status of the interlock line from the modem (it does not initiate an interrupt).

6.4.2 Programming Sequences

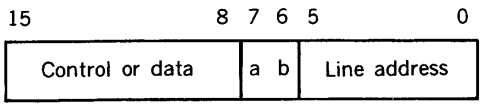
This section describes the line-setup and status-reading control sequences used with the binary synchronous

communications LAD. General DCM programming sequences are given in section 2.4.4.

6.4.2.1 Line Setup Sequence

This sequence begins with an EXC 0570 instruction. The DCM generates a control interrupt as soon as it is free. The program responds with OAR 070. The DCM loads the specified line and resumes scanning.

The character output OAR 070 has the format:



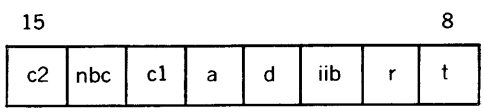
where a and b indicate the contents of the more-significant byte as follows:

ab	Selected Byte
00	Data to output buffer
01	Control
10	Mode selected*
11	Not used

* The LAD must not be in receiving or transmitting mode when mode selection is loaded.

DATA BYTE FORMAT (ab = 00): when ab = 00, the more-significant byte contains data. For data characters containing fewer than eight bits, the data is right justified (bit 9 is the least-significant data bit) and the unused bits are zeros.

CONTROL BYTE FORMAT (ab = 01): when ab = 01, the more-significant byte contains control information in the following format:



where the bits, when set, indicate

t = Transmission: puts the line in transmission mode and enables the modem request-to-send line, the line begins making output-data requests as soon as clear-to-send line is enabled by the modem. The output data is scanned by the transmitter for control characters so that proper output sequences are completed.

r = Receiving: puts the line in receiving mode. When carrier-on is enabled by the dataset, the line begins comparing input data with the RSC and goes into character mode (i.e., begins transferring data to the computer) as soon as two consecutive synchronization characters are detected. Data is scanned as it is received for control characters so that proper input sequences are completed.

iib = Input ITB characters: allows ITB character to be transferred to memory.

d = Data terminal ready: enables the modem data-terminal-ready line, which allows automatic call-termination.

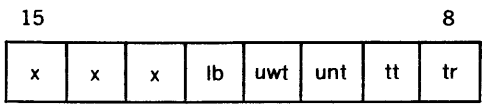
a = Accept-ring: when the t and r bits are not set this bit allows a status change interrupt to occur when the ring indicator line from the modem goes true. This bit should be cleared by software upon receipt of the status-change interrupt to prevent repetitive interrupts.

c1 = Control line 1: enables an uncommitted line to the modem. One possible use of the this line would be to enable the test mode on the Bell 303 modem.

nbc = No block check after ITB or DLE ITB: ITB is not treated as a control character and the receipt of a DLE ITB in transparent text will result in a format error.

c2 = Control line output 2: causes a 1-millisecond pulse to be sent out on this line to the modem. It is used to enable the new-sync line that is present on some Bell modems.

MODE SELECTION FORMAT (ab = 10): when ab = 10, the most-significant byte contains mode selection information in the following format:



tr = Receiver test mode: in this mode, all data received including all SYN characters are input to the processor. This includes the BCC and all control characters. In the receiver test mode, the LAD requires one SYN character to obtain synchronization (the LADs transmit section always sends two SYN characters when first enabled). If the unt bit is also set, the most-significant bit of each character transferred to the processor will be zero (the receiver section checks each input character for odd parity and removes the parity bit in this mode). The line will remain in the receiver mode until cleared by a control byte with the r bit set to zero.

tt = Transmitter test mode: in this mode, the LAD sends the data as it appears in the output buffer. The only exceptions are; when the transmitter is first enabled, two SYN characters are sent, and if the unt bit is set, parity is generated and set in the eighth bit of each character. If the uwt bit is set, no parity is generated the line remains in the transmitter mode until cleared by the program.

unt and uwt = these two bits determine the type of error checking and the character format that is used. The following table shows the character format and error checking for the different settings of these bits.

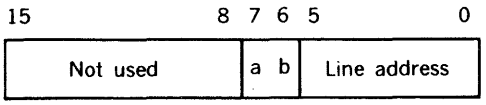
uwt	unt	Character Format	Error Checking
0	0	EBCDIC (with transparent text capability)	CRC 16
0	1	ASCII (without transparent text capability)	CRC 8 (CRC) and VRC (odd parity)
1	0	ASCII (with transparent text capability)	CRC 16 and VRC (odd parity) VRC generation is suspended during transparent text.
1	1	Illegal configuration	

lb = Loopback: the LADs transmitted data is sent back to the received data (data is not passed through the modem drivers and receivers) and an internal test clock is provided for the receiver and transmitter. This mode is meant for testing without a test shoe or modem. It does not disable the modem interface except for the received data line. The LAD will ignore the modem clear-to-send line.

6.4.2.2 Read Line-Status Sequence

This sequence begins with an EXC 0670 instruction. The DCM generates a control interrupt as soon as it is free. The program responds with OAR 070 and CIA 070.

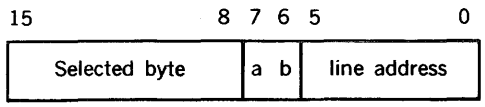
The character output by OAR 070 has the format:



where a and b select one of two possible bytes:

ab	Selected Byte
00	Data
01	Line status
10	Not used
11	Not used

The character input by CIA 70 has the format:



DATA BYTE FORMAT (ab = 00): when ab = 00, the selected byte contains data in the same format as in the line-setup sequence (section 6.4.2.1).

LINE-STATUS BYTE FORMAT (ab = 01): when ab = 01, the selected byte contains line-status information in the same format as the status bits in a status-change interrupt (section 6.4.1.2).

6.5 BSC LAD TRANSMITTING OPERATION

When the BSC LAD is in a transmitting operation and the transmitter test mode is not selected, it can insert the beginning and ending PAD characters, generate and transmit the BCC, detect and enter the transparent text mode, detect and report underflow conditions, and insert SYN SYN or DLE SYN every second while transmitting text or heading data.

The following paragraphs explain how the LAD handles each control character during transmission. Note: The ending character (ENQ, ETB, ETX, or ITB) of a block transparent text must have the most significant bit (the eighth bit) set to the correct parity when ASCII mode is selected (uwt bit set in the mode-selection byte).

6.5.1 Normal Transmitting Operation

The format for the normal transmitting operation is illustrated in figure 6-9. Actions taken by the LAD during normal receiving operations are described in the following paragraphs.

6.5.1.1 Synchronous Idle (SYN)

This character is used to establish and maintain synchronization. The computer program must insert the beginning SYN characters for an output message. A minimum of two contiguous SYN characters must precede each output block, including intermediate blocks. The BSC LAD transmits the SYNs as they are transferred from the computer, but does not include them in the BCC accumulation. The LAD inserts two SYN characters in the output data stream every second while transmitting text or heading data. The LAD inserts two SYN characters as a pad, when first put in the transmitting mode after clear-to-send goes true.

6.5.1.2 Start of Heading (SOH) and Start of Text (STX)

The first SOH or STX character of a message starts the block check character accumulation. The control character that begins the accumulation is not included in the BCC accumulation; however, all succeeding SOHs or STXs, including any that begin succeeding intermediate blocks, are included.

6.5.1.3 End of Transmission Block (ETB) End of Text (ETX)

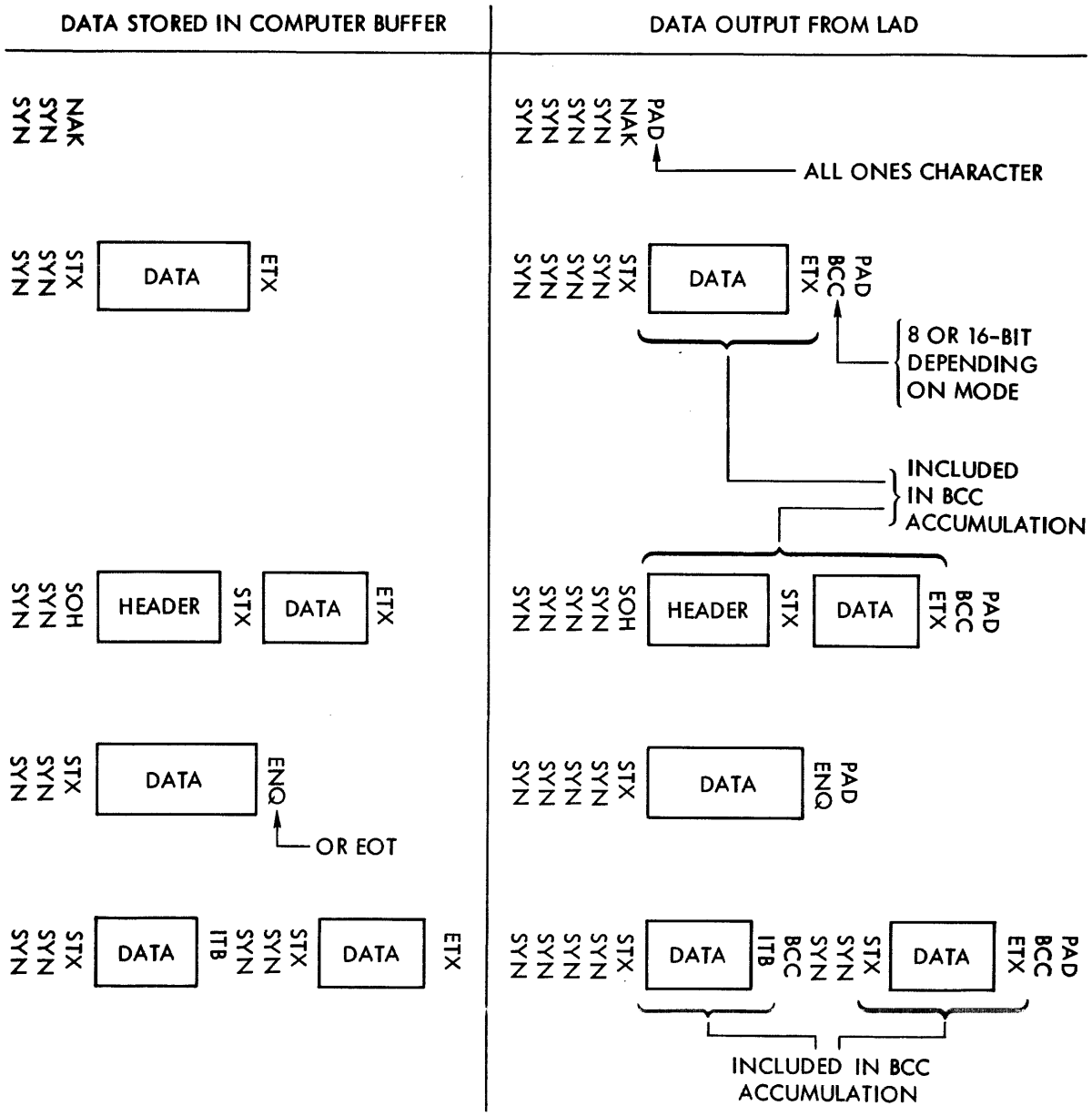
When outputting text or heading data, the detection of an ETB or ETX character in the output data stream causes the LAD to: send a BCC, send a PAD, reset the transmitter bit in the LADs control register after transmission of the PAD, and issue a status change interrupt to report output completion.

6.5.1.4 End of Intermediate Transmission Block (ITB)

If an ITB character ends a block started by SOH or STX, the LAD sends the ITB, sends the BCC, and issues a status-change interrupt to report output completion. The LAD will remain in the transmitter mode and begin requesting the next block of data. If a SOH or STX begins the next intermediate block, it is included in the BCC accumulation. If the nbc bit is set in the LADs control register, ITBs are treated as a non-control character and do not cause a BCC to be sent or an interrupt to be generated.

6.5.1.5 End of Transmission (EOT)

This character will cause the LAD to send a PAD character after sending the EOT, issue a status change interrupt reporting output completion after transmission of the PAD, and resets the transmitter bit in the LADs control register.



VTII-3243

Figure 6-9. BSC LAD Normal Transmitting Operation

6.5.1.6 Enquiry (ENQ)

After sending an ENQ, the LAD sends a PAD character, issues a status change interrupt reporting output completion after transmission of the PAD, and resets the transmitter bit in the LADs control register.

6.5.1.7 Affirmative Acknowledgment (ACK0/ACK1)

ACK0 and ACK1 are treated as normal data if imbedded in heading or text data (i.e., if detected after a SOH or STX), but otherwise are handled the same as ENQ.

6.5.1.8 Wait-Before-Transmit-Positive Acknowledgment (WACK)

WACK is handled the same as ACK0 and ACK1.

6.5.1.9 Negative Acknowledgment (NAK)

NAK is handled the same as ACK0 and ACK1.

6.5.1.10 Reverse Interrupt (RVI)

RVI is handled the same as ACK0 and ACK1.

6.5.1.11 Data Link Escape (DLE)

DLE is used to provide supplementary line control characters, such as WACK, ACK0, ACK1, and RVI. If a DLE without the proper second character is sent and the LAD is not sending text or heading data, the LAD handles the sequence as though an ACK0 or ACK1 was sent. If this erroneous DLE sequence is sent during text or heading transmission, it is treated as data. See section 6.5.2 for futher DLE sequences.

6.5.2 Transparent Transmitting Operation

The format for the transparent transmitting operation is illustrated in figure 6-10.

6.5.2.1 DLE STX

This sequence initiates the transparent mode. The LAD responds to this sequence even if it is imbedded in normal heading or text data. The LAD begins accumulating the BCC on the character following the STX, unless it follows an SOH character or begins an intermediate block following a previous intermediate block. In this case, the DLE STX is included in the BCC accumulation.

6.5.2.2 DLE ETB and DLE ETX

Once the transparent mode is entered and an ETB or ETX character is detected in the output data, the LAD tests to see if the byte count went to zero on this character. If it did not, the LAD assumes the character is data; if it did, the LAD inserts a DLE in the output data stream, sends the ETB or ETX, sends the BCC, sends a PAD, reports output completed via a status change interrupt (after the PAD has been transmitted), and resets the transmitter bit in the LADs control register. Note: The ETB and ETX must contain the correct parity in the eighth bit position as it is stored in memory when using the ASCII with the transparent mode.

6.5.2.3 DLE SYN

This sequence is used to maintain synchronization or as a time-filler in transparent transmission. This sequence cannot be inserted in the output buffer by the computer software because the LAD treats it as binary data and the sequence DLE DLE SYN is sent by the LAD. In the transparent text mode, the LAD inserts a DLE SYN in the output data stream every second. DLE SYN sequences are not included in the BCC accumulation.

6.5.2.4 DLE ENQ and DLE EOT

This sequence is used to prematurely end a BSC transparent text block. If this sequence is sent, software needs only to insert the ENQ or EOT as the last output buffer character; the DLE is inserted as it is for the DLE ETB or DLE ETX. When the LAD detects output transparent text that looks like an ENQ, or EOT, it tests to determine if the byte count went to zero on this character. If it did, a DLE is sent then ENQ followed by a PAD. After the PAD clears the LAD, the transmitter bit in the LADs control register is cleared and an output completion is conveyed via a status-change interrupt.

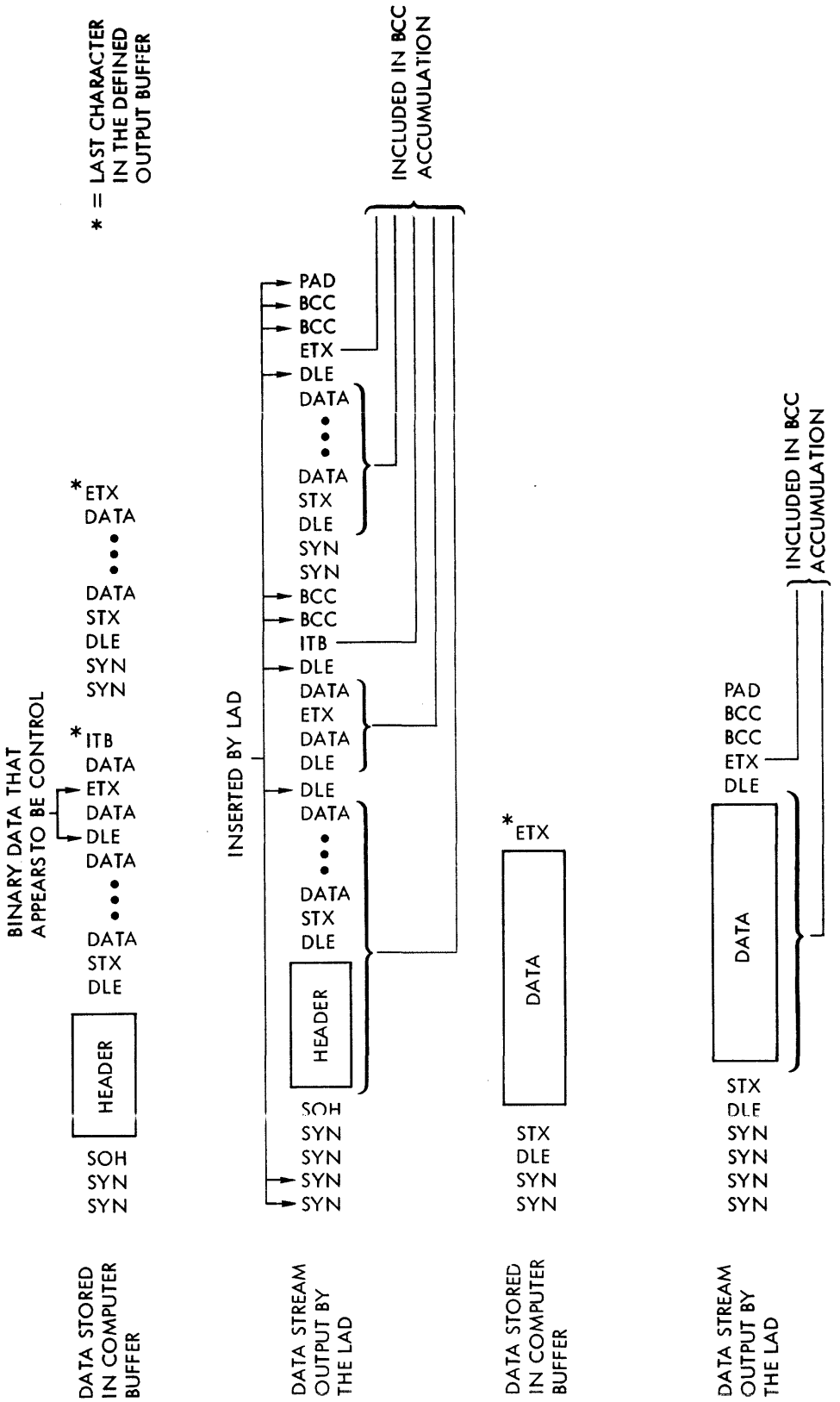
6.5.2.5 DLE DLE

The BSC LAD inserts a second DLE character whenever it detects transparent data that appears to be a DLE control character. This second DLE is not included in the BCC accumulation.

6.5.2.6 DLE ITB

This sequence is handled the same as the DLE ETB or DLE ETX except that the LAD remains in the transmitting mode and begins fetching the next intermediate block after the BCC is sent. Successive intermediate blocks include the starting control character or sequence, if any, in the BCC accumulation (i.e., DLE STX, STX, or SOH).

If the nbc bit is set in the LADs control register, the ITB does not initiate any action and is treated as transparent



VTII-3244

Figure 6-10. BSC LAD Transparent Transmitting Operation

text. Note: Since it is necessary to place the ITB at the end of the defined output buffer and since the LAD remains in the transmitter mode between intermediate blocks, the output buffer address must be updated before the output byte count when updating the line control block between intermediate blocks.

6.6 BSC LAD RECEIVING OPERATION

When in a receiving operation with the receiver test mode not selected, the BSC LAD performs the following:

a. Obtains character synchronization.

b. Removes SYN characters (DLE SYN in transparent text) from incoming data.

c. Detects and enters the transparent mode.

d. Detects and reports overrun conditions, parity errors ASCII mode without transparency), and format errors (transparent mode only).

e. Accumulates BCC and check for BCC errors.

f. Reports the receipt of BSC data link control characters via the status change interrupt (excluding SOH and STX in DLE STX).

All line errors that occur (overrun, BCC, parity, or format) are stored and reported after a valid ending character (ETX, ETB, EOT, ENQ, DLE, ETX, DLE ETB, DLE EOT, or DLE ENQ) or the receiver times out. If more than one error occurs in a message, only the first error is reported. The detection of a line error ends the transfer of data to the computer.

6.6.1 Normal Receiving Operation

The format for the normal receiving operation is illustrated in figure 6-11. Actions taken by the LAD during normal receiving operations are described in the following paragraphs.

6.6.1.1 Synchronous Idle (SYN)

The BSC LAD requires a minimum of two contiguous SYN characters to obtain character synchronization. After obtaining character synchronization, the LAD removes all successive SYN characters from the incoming data. SYN characters are not included in the BCC accumulation and are not transferred to the processor except in the test receiving mode.

6.6.1.2 Start of Heading (SOH) and Start of Text (STX)

Once synchronization is obtained, the BSC LAD transfers all incoming data to the processor except SYN characters. The first SOH or STX character starts the BCC accumulation but is not included in the accumulation. Succeeding SOH or STX characters are included in the BCC accumulation, including the ones that begin successive intermediate blocks.

6.6.1.3 End of Transmission (ETB) and End of Text (ETX)

Once an SOH or STX character is received, and an ETB or ETX character is detected, the LAD transfers the character to the processor, accepts the BCC, and checks for a BCC error. If there is no error, the LAD reports the receipt of the character via the status-change interrupt, and resets the receiver bit in the control register. If there is a BCC error, it is reported as a line error before the LAD resets the receiver bit, and no status change interrupt occurs.

6.6.1.4 End of Intermediate Transmission (ITB)

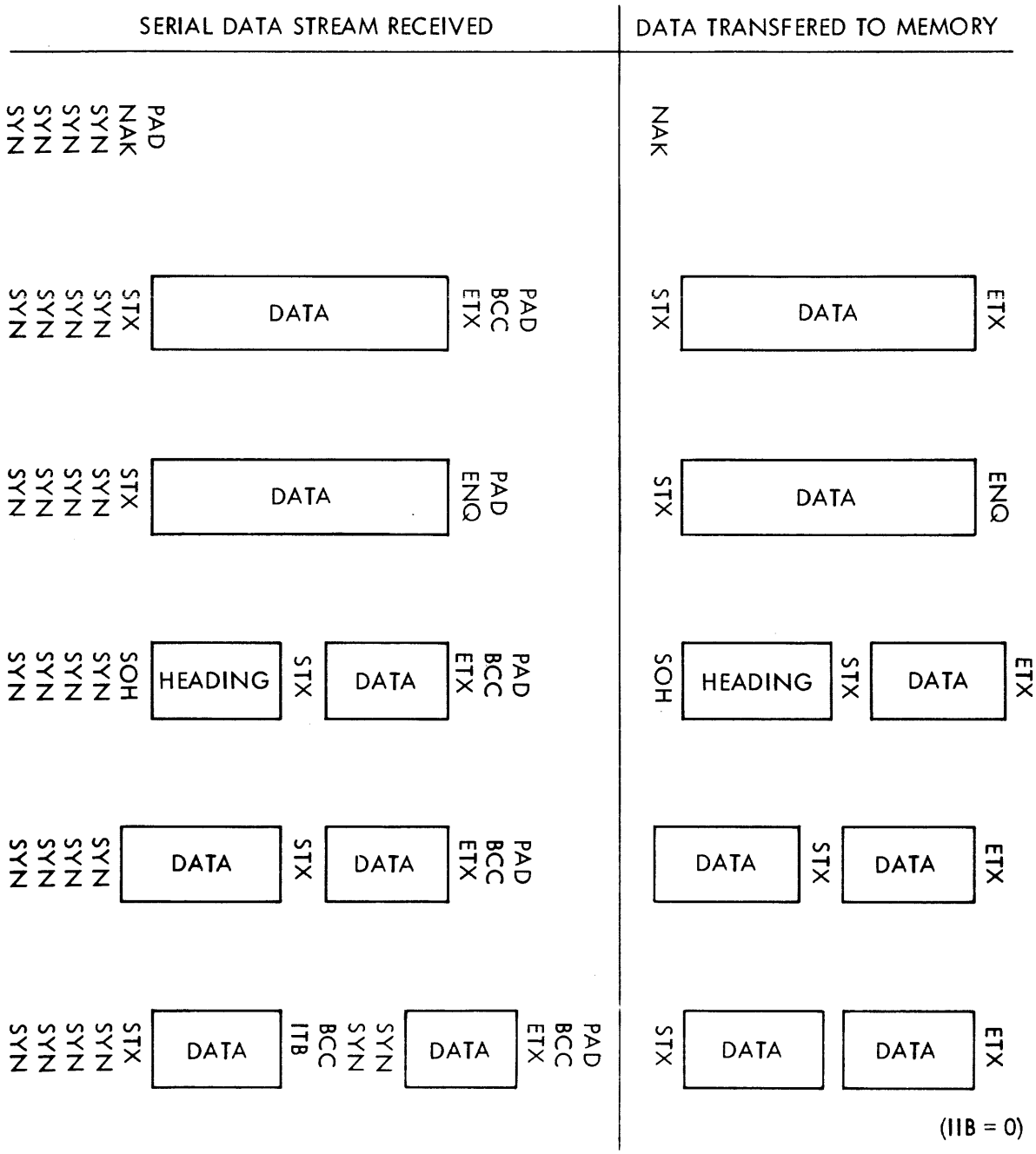
After a SOH or STX character is received, the detection of an ITB causes the LAD to accept the BCC and check for a BCC error. The ITB may or may not be transferred to the processor, depending on the state of the iib bit in the LADs control register. No status-change interrupt is generated between intermediate blocks and, if a line error occurs in an intermediate block, no more data is transferred to memory. The line error is reported when an ending character is detected or a receiver time-out occurs. If the nbc bit is set in the LADs control register the ITB character does not initiate a block check sequence and is treated as a non-control character. However, the state of the iib bit still determines if the ITB is to be transferred to the processor.

6.6.1.5 End of Transmission (EOT)

This character is sent to the processor and then reported via the status-change interrupt. After receiving and reporting a valid EOT, the LAD resets the receiver bit in the LADs control register.

6.6.1.6 Enquiry (ENQ)

If this character is received, a status change interrupt is generated after the ENQ is transferred to the processor. After receiving the ENQ, the LAD resets the receiver bit in the LADs control register.



VTII-3245

Figure 6-11. BSC LAD Normal Receiving Operation

6.6.1.7 Affirmative Acknowledgment (ACK0, ACK1)

ACK0 and ACK1 is ignored if imbedded in heading or text data, but otherwise is handled the same as ENQ.

6.6.1.8 Wait-Before-Transmit Positive Acknowledgment (WACK)

WACK is handled the same as ACK0 and ACK1.

6.6.1.9 Negative Acknowledgment (NAK)

NAK is handled the same as ACK0 and ACK1.

6.6.1.10 Reverse Interrupt (RVI)

RVI is handled the same as ACK0 and ACK1.

6.6.1.11 Data Link Escape (DLE)

DLE is used to provide supplementary line control characters, such as WACK, ACK0, ACK1, and RVI. If a DLE is received without the proper second character and the LAD is not receiving text or heading data (i.e., a SOH or STX has not been detected), the LAD handles the sequence as though an ACK0 or ACK1 was received. If this erroneous DLE sequence is received during text or heading data, it is not treated as a control sequence. See section 6.6.2 for further DLE sequences.

6.6.2 Transparent Receiving Operations

The format for the normal receiving operation is illustrated in figure 6-12. Actions taken by the LAD during transparent receiving operations are described in the following paragraphs.

6.6.2.1 DLE STX

The receipt of this sequence initiates the transparent mode. The LAD responds to this sequence even if it is imbedded in normal heading or text data. If BCC accumulation has not been started, the DLE STX begins the accumulation but is not included in it. If the BCC accumulation has already started or if the DLE STX begins a successive intermediate block, the DLE STX is included in the accumulation. The DLE STX is then transferred to memory.

6.6.2.2 DLE ETB and DLE ETX

Once the transparent mode has been entered the receipt of either of these two sequences, generates a status-change interrupt. The interrupt occurs after the sequence is transferred to the processor. After detecting the DLE ETB or DLE ETX, the LAD accepts the BCC and checks for an error. If an error occurs, a line-error interrupt occurs instead of the status-change interrupt. The LAD resets the receiver bit of the control register after examining the BCC. Only the ETB or ETX is transferred to memory.

6.6.2.3 DLE SYN

If a DLE SYN sequence is received while in the transparent mode, the receiver timer is retriggered. The DLE SYN is not transferred to memory and is not included in the BCC accumulation.

6.6.2.4 DLE ENQ

The receipt of this sequence results in a status-change interrupt indicating that the ENQ has been transferred to the processor. The LAD then resets the receiver bit in the LADs control register. The DLE is not transferred to memory if transparent text is being received.

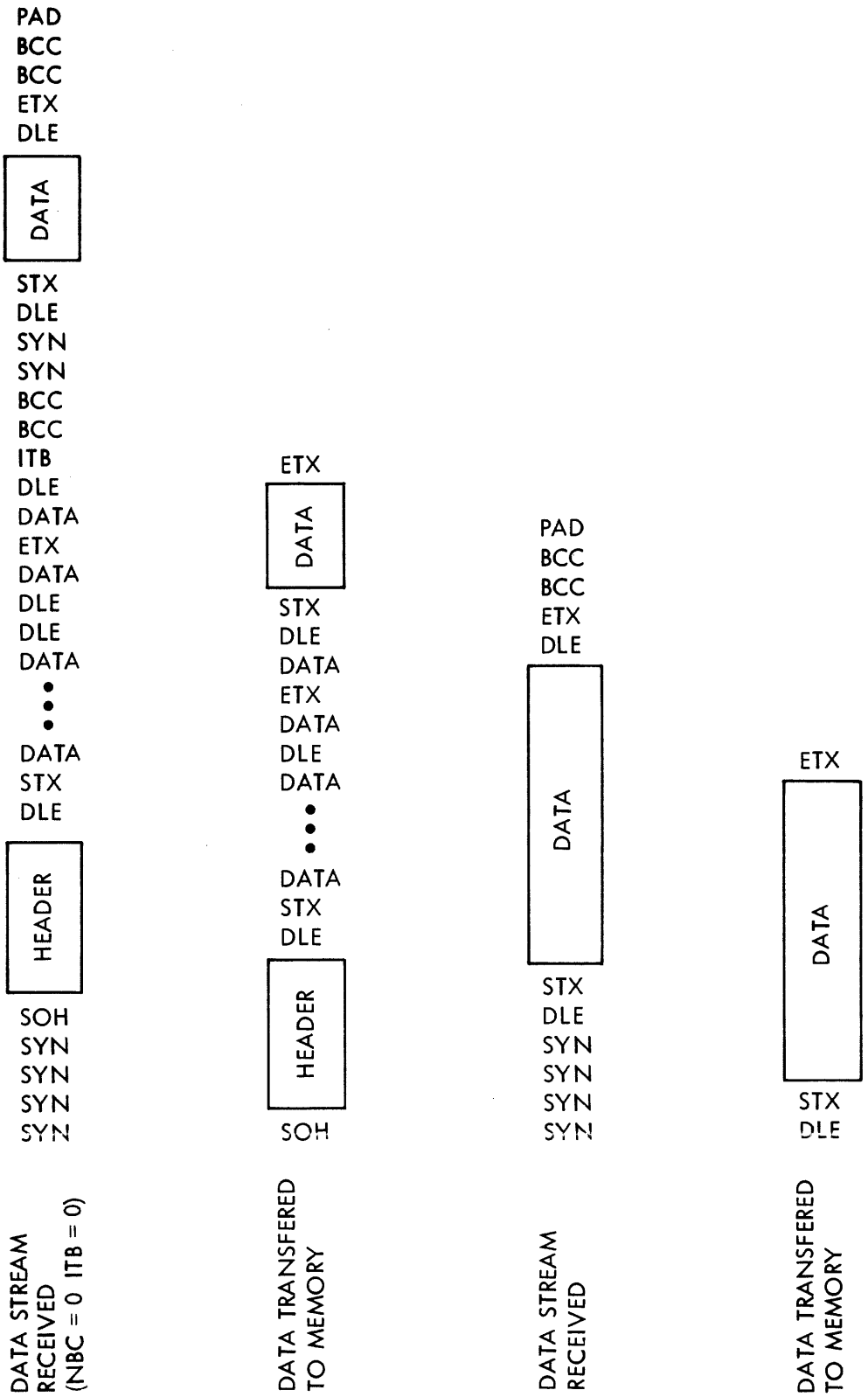
6.6.2.5 DLE DLE

The first DLE of a DLE DLE sequence is removed from the incoming data and is not included in the BCC accumulation.

6.6.2.6 DLE ITB

DLE ITB is used to end intermediate transparent blocks. At the end of each transparent intermediate block, the LAD remains in the receiver mode, but is removed from the transparent text mode. DLE ITB does not generate a status-change interrupt. The nbc bit and iib bit control DLE ITB as shown below:

nbc	iib	Action Upon Receipt of DLE ITB
0	0	A block check sequence is performed; DLE ITB is not transferred to the processor.
0	1	A block check sequence is performed; DLE is not transferred to the processor. ITB is transferred to the processor.
1	0	DLE ITB is treated as a format error. DLE ITB is not transferred to the processor.
1	1	DLE ITB is treated as a format error. DLE is not transferred to the processor. ITB is transferred to the processor.



VTII-3246

Figure 6-12. BSC LAD Transparent Receiving Operation

6.6.2.7 DLE EOT

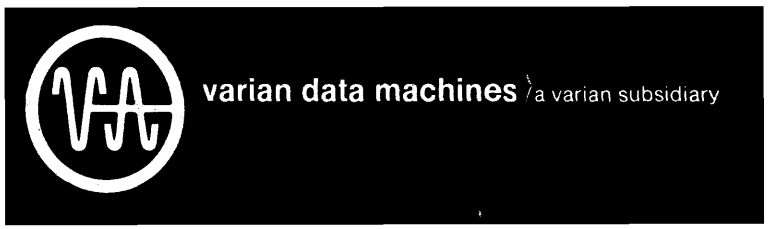
DLE EOT is handled the same as DLE ENQ.

6.7 MNEMONICS

AI–	Decoded with BI– to determine type of line error
AR +	Accept ring
ASCII +	ASCII mode indicates that UNT or UWT is set
A–	Multiplexor bus signal
BCLB2–	2.457C MHz square wave derived from CLKB +
BCLKB +	Repowered CLKB +
BCZ +	Repowered CZ +
BI–	See AI–
BSCLK +	1.2288 MHz square wave derived from CLKB +
B–	Multiplexor bus signal
CL1 +	Control line 1
CL2 +	Control line 2
CLK–	4.9152 MHz square wave from multiplexor board 1
CLKB +	Multiplexor bus signal
CO +	Carrier on
CRLD +	Control register loading clock
CRON +	Carrier on
CS +	Clear-to-send
CZ +	Byte count zero flag from multiplexor board 2
C1 +	Control line 1
C2 +	Control line 2
DATR +	Data-terminal-ready
DBSCLK +	Delayed BSCLK +
DSRDY +	Dataset ready
DSR +	Dataset ready
DTD–	Disable transmitter data
DTLD–	Load data
DTR +	Data-terminal ready
EDATA +	Enable data to the multiplexor bus
EOCRC +	Enable CRC generator output to the output-shift register
EOSR0 +	Enable wrap-around of the output-shift register
ESTAT +	Enable status to the multiplexor bus
ICMACL +	Clock for input-control memory-address counter
ICMA00 + thru ICMA06 +	Input-control memory-address lines
ICM00 + thru ICM23 +	Input-control memory output
ICRCZ +	Input CRC generator contents are zero
IDR +	Input-data request
IEA00–	Enabling signal for the input control memory words 0 through 31 (ASCII)
IEA32–	Enabling signal for the input-control memory words 32 through 63 (ASCII)
IEA64–	Enabling signal for the input-control memory words 64 through 95 (ASCII)
IEE00– IEE32– IEE64–	Enabling signals for EBCIDIC ROMs
IFUNC0– thru IFUNC7–	Decoded outputs of the FUNC field of the input-control memory
IJFF +	Input-jump flip-flop
ILOAD1– thru ILOAD7–	Decoded output of the LOAD field of the input control memory
IMCH +	Input match
IRR0 + thru IRR5 +	Outputs of the input-return register
ISBZY–	Service request logic busy (input)
ISFTCK +	Input-shift clock

ISHFTC+	Input-shift complete
ISHIFT+	Input shift enable
ISR0+ thru ISR7+	Parallel outputs of the input-shift register
ITRANS–	Input transparent-data mode
ITTRUE+	Input test true
IWFF+	Input wait flip-flop
LADA–	Line adapter address
LADE–	Line adapter enabler
LERR–	Multiplexor bus signal
LICMAC–	Load input-control memory-address counter
LOCMAC–	Load output-control memory
LT+	Local test (wideband interface)
L00– thru L03–	Line-adapter address-decoder outputs
MB0– thru MB7–	Multiplexor 8-bit bus
MSLD–	Loading strobe for the mode selecting register
NBC–	No block checks initiated by ITB
NSYNC+	New sync
OCMACL+	Clock signal for the output-control memory-address counter
OCMA00+ thru OCMA06+	Outputs of the output-control memory-address counter
OCM00+ thru OCM23+	Output-control memory-output signals
ODR+	Output-data request
OEA00–	Enabling signal for the output-control memory words 0 through 31 (ASCII)
OEA32–	Enabling signal for the output-control memory words 32 through 63 (ASCII)
OEA64–	Enabling signal for the output-control memory words 64 through 95 (ASCII)
OEE00– OEE32– OEE64–	Enabling signals for EBCIDIC ROMs
OFUNC0– thru OFUNC6–	Decoded outputs of the FUNC field of the output-control memory
OJFF+	Output jump flip-flop
OLDCK+	Output data strobe
OLOAD1– thru OLOAD7–	Deocded outputs of the LOAD field of the output-control memory
OMCH+	Output match
ORR0+ thru ORR5+	Outputs of the output-return register
ORST–	Output reset
OSBZY+	Service-request logic is busy (output)
OSFTCK+	Output shift clock
OSHFTC+	Output shift completed
OSHIFT+	Output shift enabler
OSR0+ thru OSR7+	Parallel outputs of the output-shift register
OTRANS+	Output transparent text mode
OTTRUE+	Output text true
OWFF+	Output wait flip-flop
RCP+	Receiver clock
RDA+	Receiver data available
RD+	Receiver data
RD0+ thru RD7+	Parallel receiver data outputs of the synchronous receiver/transmitter
ROR+	Receiver overrun
RPE+	Receiver parity error
RSI+	Receiver serial input
RSTRBE+	Receiver strobe
RTS+	Request-to-send

SCB0– SCB1–	Multiplexor bus signals
SCTA +	Sync. character transmitted (underflow)
SIR +	Status input request
TD +	Transmit data
TD0 + thru TD7 +	Parallel transmit data inputs to the synchronous receiver/transmitter
TSET +	Transmitter clock
UNT +	USASCII mode without transparent capability
UWT +	USASCII mode with transparent capability
WSTRBE +	Write strobe


varian data machines
a varian subsidiary