SYSTEMS
ENGINEERING LABORATORIES

TECHNICAL MANUAL

SYSTEMS 32 SERIES

TLC Controller Model 9005

June 1979

• Publication Number 303-329005-000

REVISION INSTRUCTIONS AND MANUAL HISTORY

IUIPMENT: TLC Controller Model 9005 **PUBLICATION NO.** 303-329005-000

IRPOSE: To provide a general description, operation and programming information, and theory of operation for the Teletypewriter, Line Printer, and Card Reader Controller (TLC), Model 9005.

EVISION INSTRUCTIONS: Delete and add pages as shown on the following table.

DELETE	ADD
NA	NA

MANUAL HISTORY

REV. TYPE	REV. NO.	DATE ISSUED	CONTROL DOC. NO.	ECL
1st Ptg	Original	6-79	180-103160	4

REV. TYPE	REV. NO.	DATE ISSUED	CONTROL DOC. NO.	ECL

R= INTERIM REVISION
F= FORMAL REVISION

LIST OF EFFECTIVE PAGES

The total number of pages in this book is 81, consisting of the following:

Page	Issue
Title	Original
iii through ix/x	Original
1-1 through 1-4	Original
2-1 through 2-22	Original
3-1 through 3-38	Original
A-1 through A-2	Original
B-1 through B-2	Original
C-1 through C-2	Original
D-1 through D-2	Original

TABLE OF CONTENTS

SECTION I GENERAL DESCRIPTION

SECTION II OPERATION AND PROGRAMMING

TABLE OF CONTENTS (Cont'd)

SECTION III THEORY OF OPERATION

APPENDIXES

LIST OF ILLUSTRATIONS

LIST OF TABLES

LIST OF RELATED PUBLICATIONS

Title	Publication Number
Drawings Manual - SYSTEMS 32 SERIES TLC Controller Model 9005	304-329005
Firmware Manual - SYSTEMS 32 SERIES TLC Controller Model 9005	305-329005
Technical Manual - SYSTEM 32 SERIES Input/Output Microprogrammable Processor	325-329000
SYSTEMS 32 SERIES Circuit Registration Manual	313-325000

SECTION I

GENERAL DESCRIPTION

1-1 INTRODUCTION

This manual contains or references information concerning the maintenance and installation of the SYSTEMS 32 SERIES Computer TLC Controller, Model 9005, shown in Figure 1-1. The TLC Controller is designed and manufactured by Systems Engineering Laboratories, Fort Lauderdale, Florida.

The information contained in this manual is presented in the following order:

Section I	-	General Description
Section II	-	Programming
Section III	-	Theory of Operation

The Drawings Manual (Publication Number 304-329005) provides the assembly drawings, circuit card drawings, and logic diagrams used with the SYSTEMS 32 SERIES Computer. The Firmware Manual (Publication Number 305-329005) contains the microprogram listings for the Input/Output Microprogrammable Processor (IOM) used for control of the SYSTEMS 32 SERIES Computer peripheral devices.

The TLC Controller is an optional feature for SYSTEMS 32 SERIES Computers and provides the capability to output data to a teletypewriter or line printer and input data from a teletypewriter or card reader. The TLC Controller also provides the capability to input data from the teletypewriter and output it to the line printer.

1-2 MODEL NUMBERS

SYSTEMS Model 9005 TLC Controller includes an IOM, Teletypewriter, Line Printer, and Card Reader Device Dependent Interface logic. The The IOM consists of a SelBUS interface and a Microprogrammable Processor (MP).

The SYSTEMS Model 9005 TLC Controller is designed to operate with the following compatible input/output peripheral devices:

1. Teletypewriter (TTY) Interface

   Model 9201 TTY (10 cps)
   Model 9202 TTY (30 cps)
   Model 9203 A/N CRT (95 char)
   Model 9204 A/N CRT (64 char)

2. Line Printer (LP) Interface

   Model 9223 Matrix Printer
   Model 9225 LP (300 lpm)
   Model 9245 LP (260 lpm)
   Model 9226 LP (600 lpm)
   Model 9246 LP (436 lpm)
   Model 9237 LP (900 lpm)
   Model 9247 LP (660 lpm)

3. Card Reader (CR) Interface

   Model 9209 CR (200 cpm)
   Model 9210 CR (300 cpm)
   Model 9211 CR (1000 cpm)

1-3 PHYSICAL DESCRIPTION

The TLC Controller, Model 9005, consists of an IOM, teletypewriter, line printer, and card reader device plug-in module. The Model 9005 plugs directly into the SelBUS like all other standard IOMs for the SYSTEMS 32 SERIES Computers. The TLC Controller circuit card is 15 inches x 17.90 inches, installs in the CPU logic chassis or an I/O extender logic chassis, and requires one SelBUS slot.

1-4 FUNCTIONAL DESCRIPTION

The Model 9005 TLC Controller is a multi-device IOM for the SYSTEMS 32 SERIES Computers. The TLC, contained on a single plug-in module, provides I/O control for a teletypewriter, line printer, and card reader.

The TLC Controller consists of the five functional parts listed below:

1. SelBUS Interface
2. Microprogrammable Processor (MP)
3. Teletypewriter Interface
4. Line Printer Interface
5. Card Reader Interface

The SelBUS interface includes drivers and receivers for data and control lines on the SelBUS, a 32-bit buffer register, and a toggle switch for logically disconnecting the TLC Controller from the SelBUS.

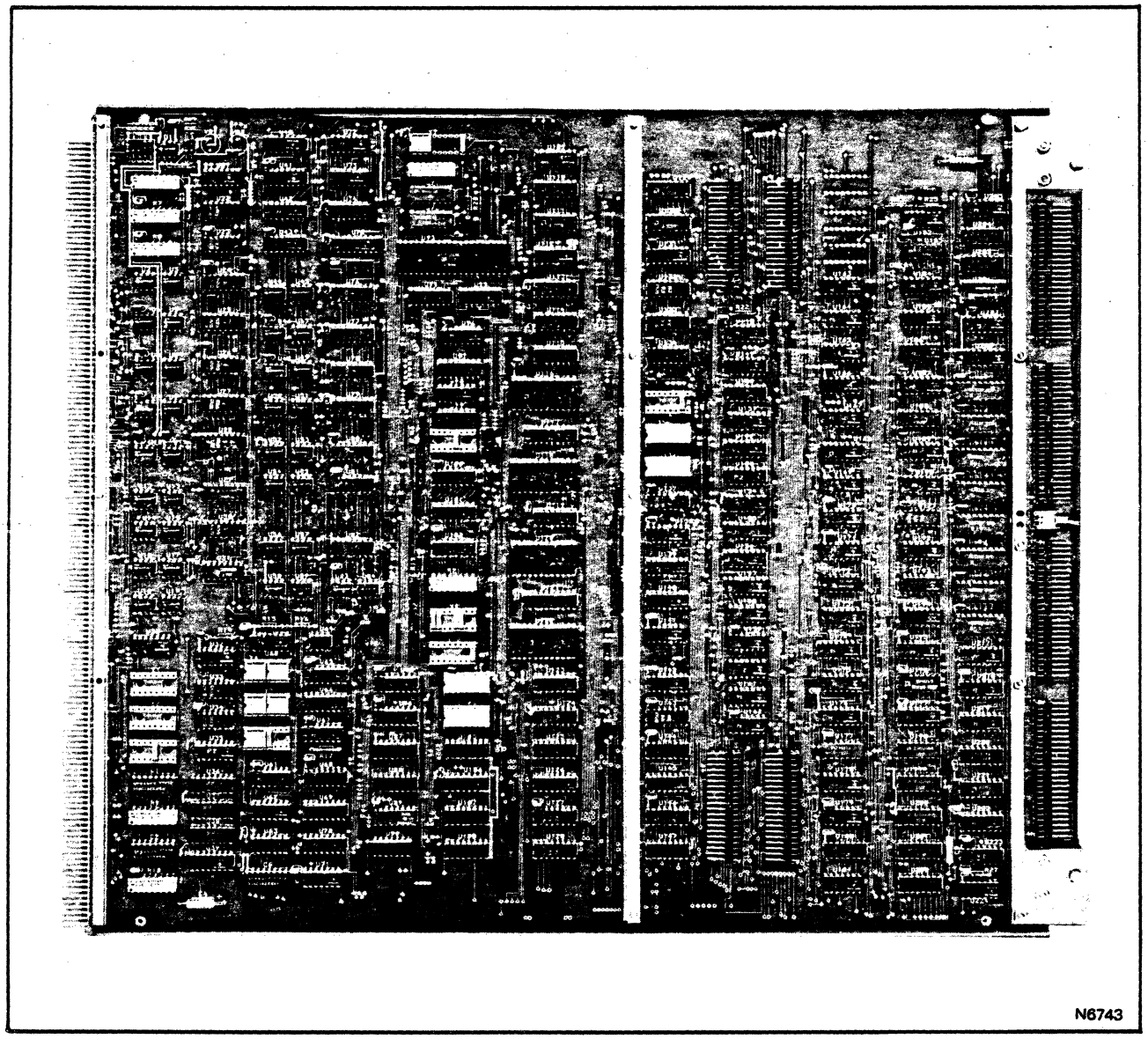


Figure 1-1. Copper TLC Controller

The firmware in the microprocessor has been designed to respond to the Command Device (CD), Test Device (TD), and other SYSTEMS 32 SERIES I/O instructions. Any of the three interfaces in the TLC Controller can be conditioned for a block transfer by the CPU executing a single CD instruction. Once the block transfer has been initialized, the TLC Controller takes full control over the operation, and the CPU is free to perform other tasks.

When the block transfer is completed, the TLC Controller signals the CPU with an interrupt. The CPU then executes a TD instruction to determine whether the transfer was successful and, if not, what errors were encountered.

Concurrent operation of the Teletypewriter, Line Printer, and Card Reader interfaces in the TLC Controller is accomplished by using individual buffers and I/O interrupts for each device.

Figure 1-2 is a block diagram for the TLC Controller and related input/output devices.

1-5 SPECIFICATIONS AND LEADING PARTICULARS

Table 1-1 lists the specifications and leading particulars of the TLC Controller.

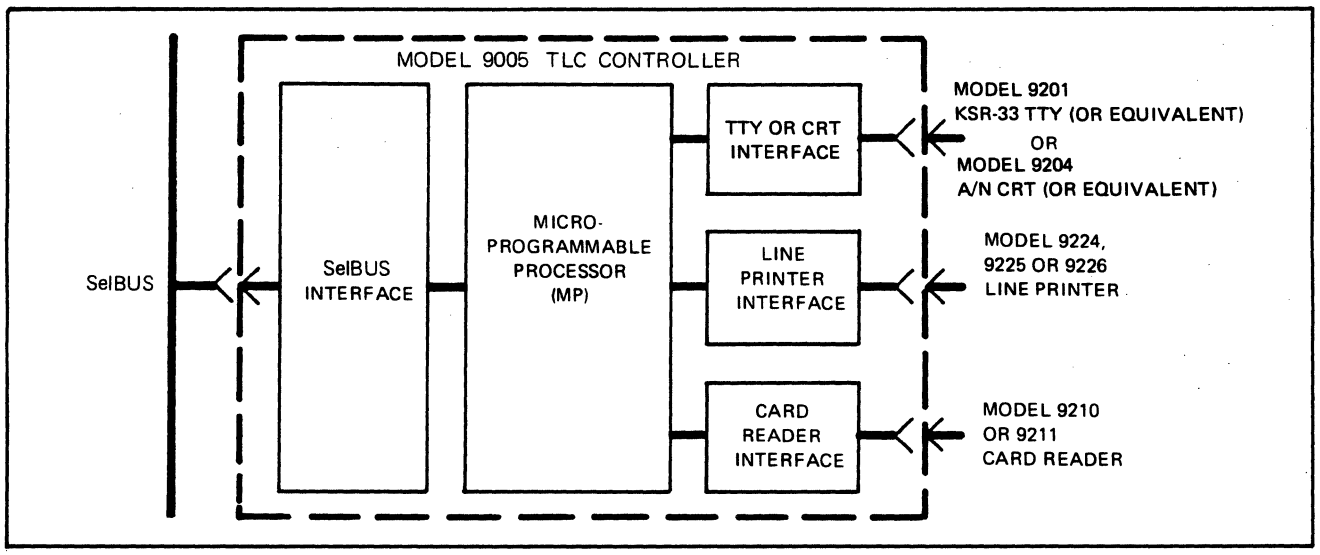


Figure 1-2. Block Diagram - TLC Controller

Table 1-1. Specifications and Leading Particulars

Characteristic	Specification
IOM Thruput (Max)	1.2M bytes/second
Number of Devices	1 each; TTY, LP, and CR
Compatible Devices	
TTY Channel	Model 9201 KSR-33, or Model 9204 A/N CRT or equivalent devices
LP Channel	Model 9224 LP (125 lpm) Model 9225 LP (300 lpm) Model 9226 LP (600 lpm)
CR Channel	Model 9210 CR (285 cpm) Model 9211 CR (1000 cpm)
Dimensions	15 in. x 17.90 in. (plug-in board)
Weight	3 lb
SelBUS Slot Requirement	2 slots
SelBUS Electrical Load	1 load
Power Requirements	Provided by SelBUS
Temp/Humidity	Same as CPU
Maximum Cable Length To:	
TTY or CRT	30 ft
LP	20 ft
CR	60 ft
Prerequisite	SYSTEMS 32 SERIES Computer

SECTION II

OPERATION AND PROGRAMMING

2-1 INTRODUCTION

This section of the manual contains the operating and programming instructions for the TLC Controller; however, specific operating instructions for the related peripheral devices not contained in this manual may be found in the vendor's manual associated with the device.

2-2 CONTROLS AND INDICATORS

2-3 OFF-LINE SWITCH

The TLC Controller circuit card is provided with an Off-Line switch on the front edge of the circuit card. This switch must be placed in the On-Line (right-hand) position to enable the TLC controller to communicate with the SelBUS and the CPU.

2-4 TLC PHYSICAL ADDRESS JUMPERS

The TLC Controller circuit card has a set of jumpers that selects the physical address (SelBUS address) of the TLC. The address selected by these jumpers must correspond to the TLC physical address associated with the TLC I/O instruction address (CD, Device Address field) during the SYSTEMS 32 SERIES CPU Initial Program Load (IPL) of the Initial Configuration List (ICL).

The Physical Address jumpers are shown on Logic Drawing 130-103175 (sheet 6) contained in the Drawings Manual. The Physical Address jumpers are referenced by logic callouts X7-1 through X7-7 and must be set to reflect the Low true physical address of the TLC. Switch X7-1 selects the most significant address bit, and X7-7 selects the least significant address bit.

2-5 MULTIPLE CONTROLLER CONTROLLER (MCC)

In an MCC configuration, the IOM acts as several types of controllers, such as the TLC Controller, which controls a teletype, line printer, and card reader which are contained on one logic card.

2-6 SelBUS PRIORITY RECOGNITION JUMPERS

The TLC circuit card has a set of 21 Priority Recognition jumpers used to assign priorities to all system modules that have a SelBUS transfer priority higher than this TLC. The TLC's request for transfer is inhibited if any controller with a higher priority wishes to transfer data at the same time. The Priority Recognition jumpers are shown on Logic Drawing 130-103175 (sheet 10) and are referenced by logic callouts X4-2 through X4-8, X5-1 through X5-8, and X6-1 through X6-6. To assign higher priority transfer levels, the jumpers corresponding to the specific higher priority levels must be placed in the CLOSED (ON) position, and the jumpers corresponding to the priority level assigned to this TLC and all lower priorities must be placed in the OPEN (OFF) position.

2-7 SelBUS PRIORITY GENERATION JUMPERS

The TLC circuit card has a set of 22 Priority Generation jumpers used to assign the bus transfer priority of this TLC. The Priority Generation jumpers are shown on Logic Drawing 130-103175 (sheet 10) and are referenced by logic callouts X1-1 through X1-8, X2-1 through X2-8, and X3-1 through X3-6. To assign a bus transfer priority level to this TLC, the jumper controlling the priority level must be placed in the CLOSED (ON) position, and all remaining jumpers must be placed in the OPEN (OFF) position. The priority level chosen by this procedure must correspond to the priority level used for this TLC in the Priority Recognition jumpers.

2-8 SelBUS PRIORITY ENABLE SWITCHES

The SYSTEMS 32 SERIES CPU logic chassis backplane has a SelBUS terminator circuit card that contains a set of 22 Priority Enable switches. Those switches associated with priority levels assigned to modules on the SelBUS must be in the OPEN (OFF) position, and switches associated with the unassigned priority levels must be in the

CLOSED (ON) position. The SYSTEMS 32 SERIES Computer Technical Manual applicable to the system being used provides a system level discussion of all switches that must be set to communicate with the SelBUS.

2-9 BAUD RATE JUMPERS

The TLC circuit card has a set of 21 jumpers that selects the Baud Rate for the desired operation when the Teletypewriter Device Dependent interface is used to interface with a CRT unit. Under normal operating conditions, the Baud Rate jumpers are configured for 110 Baud.

Table 2-1 shows the settings of the Baud Rate jumpers for the 12 different Baud Rates available in the TLC Controller.

The Baud Rate jumpers are shown on Logic Diagram 130-103175 (sheet 24), contained in the drawings manual. The Baud Rate switches are referenced by logic callouts X8-1 through X8-7, X9-1 through X9-8, and X10-2 through X10-6.

The TLC circuit card has no indicators that monitor TLC or system operation.

2-10 POWER ON/OFF PROCEDURES

DC power is supplied to the TLC circuit card by the logic chassis in which the TLC circuit card is installed. The Power On/Off procedures for the logic chassis are normally covered by the CPU Power On/Off procedures. These procedures are described in the applicable SYSTEMS 32 SERIES Computer Technical Manual.

2-11 OPERATING PROCEDURES

There are no special operating procedures for the TLC; however, the technical manual for the I/O devices attached to the TLC should be consulted before the TLC is operated.

2-12 PROGRAMMING

The following discussions are intended to provide a brief description of the software instructions used to control and obtain

Table 2-1. Baud Rate Switch Settings

Baud Rate	Hex Code X8	X9	X10	X8 1 2 3 4	X8 5 6 7 8	X9 1 2 3 4	X9 5 6 7 8	X10 1 2 3 4	X10 5 6 7 8
110	10	17	2C	0 0 0 1	0 0 0 X	0 0 0 1	0 1 1 1	X 0 1 X	1 1 X X
150	10	67	24	0 0 0 1	0 0 0 X	0 1 1 0	0 1 1 1	X 0 1 X	0 1 X X
300	20	9E	48	0 0 1 0	0 0 0 X	1 0 0 1	1 1 1 0	X 1 0 X	1 0 X X
600	40	79	20	0 1 0 0	0 0 0 X	0 1 1 1	1 0 0 1	X 0 1 X	0 0 X X
900	80	D7	40	1 0 0 0	0 0 0 X	1 1 0 1	0 1 1 1	X 1 0 X	0 0 X X
1200	80	E6	40	1 0 0 0	0 0 0 X	1 1 1 0	0 1 1 0	X 1 0 X	0 0 X X
1800	02	DB	00	0 0 0 0	0 0 1 X	1 1 0 1	1 0 1 1	X 0 0 X	0 0 X X
2400	02	5D	00	0 0 0 0	0 0 1 X	0 1 0 1	1 1 0 1	X 0 0 X	0 0 X X
3600	04	3A	00	0 0 0 0	0 1 0 X	0 0 1 1	1 0 1 0	X 0 0 X	0 0 X X
4800	04	F2	00	0 0 0 0	0 1 0 X	1 1 1 1	0 0 1 0	X 0 0 X	0 0 X X
7200	08	6C	00	0 0 0 0	1 0 0 X	0 1 1 0	1 1 0 0	X 0 0 X	0 0 X X
9600	08	CC	00	0 0 0 0	1 0 0 X	1 1 0 0	1 1 0 0	X 0 0 X	0 0 X X

	X7 . 10
EIA	0
Current	1

status from the TLC. A more complete description of the software I/O instructions is provided in the SYSTEMS 32 SERIES Computer Technical Manual applicable to the user's particular system.

2-13 I/O INSTRUCTIONS

2-14 COMMAND DEVICE (CD) INSTRUCTION

The TLC is controlled by the CD software instruction. Figure 2-1 shows the basic format of the CD instruction and lists the basic decodes of the instruction. The specific operations that can be controlled by the CD instruction are defined in subsequent discussions under the individual peripheral devices.

2-15 Transfer Control Word (TCW)

The TCW is a 32-bit word used with the CD Initialize Data Transfer instruction. The TCW provides the memory data transfer address and the data transfer count as shown in Figure 2-2. The TCW is stored in a dedicated memory location for the TLC Controller. Table 2-2 lists the dedicated memory locations (transfer interrupt addresses) used by the CPU.

The controlling software system must program the TCW with the memory data address, data transfer count, and the F- and C-bits before the CD Initialize Data Transfer instruction is initiated.

2-16 Input/Output Command Doubleword (IOCD)

The CPU firmware formats the CD instruction and either the TCW address or the contents of the TCW into a 64-bit word called the Input/Output Command Doubleword (IOCD). This doubleword is stored in memory locations dedicated to the TLC Controller being operated by the CD instruction. Remember that the IOCD is a firmware format and has no relationship to the system software.

The specific IOCD format is a function of the class of device or controller being operated by the CD instruction and the type of I/O operation being initiated. Figure 2-3 shows the format for the IOCD used with Class 0, 1, and 2 devices controlled by the TLC Controller. The Class 0 device is the line printer, the Class 1 device is the card reader, and the Class 2 device is the teletypewriter. In this format, the CD instruction function code is translated into the Order byte in the first IOCD word, and the contents of the TCW are formatted into the remaining portions of the IOCD as shown in Figure 2-3.

Figure 2-4 shows the IOCD format of an IPL initiated by the IPL switch on the CPU Turnkey Panel. This specific IOCD format is only used for the first Read from the IPL device; however, the basic IOCD format remains true for all subsequent reads from the IPL device during the CPU IPL firmware sequence.

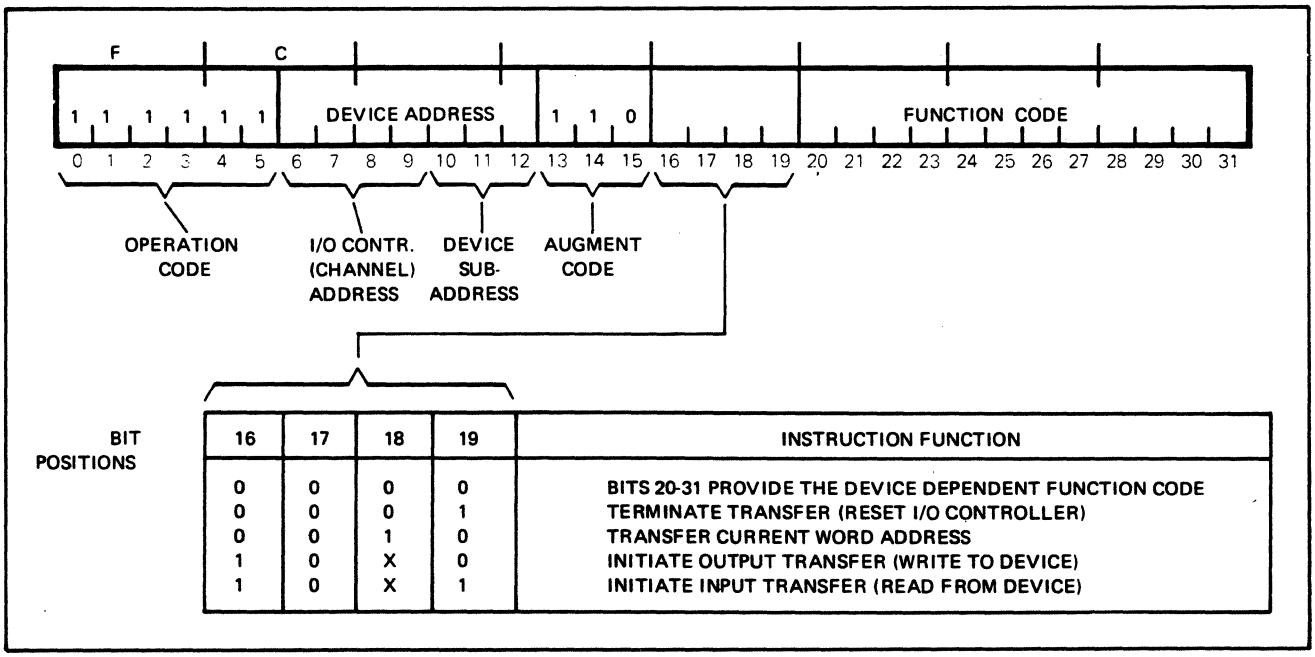


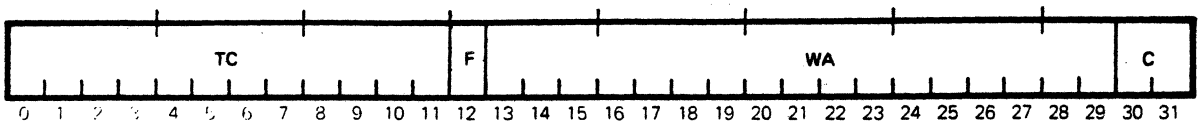
Figure 2-1. Command Device (CD) Instruction Format

When the CPU generates the IOCD, it stores the IOCD at a dedicated memory address for the TLC Controller being operated. Table 2-3 lists the dedicated memory locations used with the 16 I/O controllers that can be controlled by the CPU.

2-17 TEST DEVICE (TD) INSTRUCTION

The TD instruction is used by system software to obtain status from the TLC. The system software can program the TD instruction at the 8000, 4000, or 2000 levels. The 8000 level obtains basic I/O controlled (channel) status, which is returned to the CPU and software as four software testable condition codes. The 4000 level obtains TLC status, which is returned as four software testable condition codes. The 2000 level obtains the specific status of the TLC device dependent interface and the I/O device. The status is returned as a 16-bit status halfword and/or four condition codes. The bit definitions of the status halfword and three of the four condition codes are device dependent, and the fourth condition code indicates that the status halfword transfer was not performed. When the CPU receives the status halfword from the TLC, it stores the halfword at the memory location addressed by the contents of the TCW for TLC Controller addressed by the TD instruction.

Figure 2-5 shows the format of the TD instruction and lists the condition code response for each of the three levels of the TD instruction.



BITS 0-11 DESIGNATE THE NUMBER OF TRANSFERS TO BE MADE BETWEEN MEMORY AND THE DEVICE CONTROLLER CHANNEL. THE TRANSFER COUNT IS WORDS, HALFWORDS, OR BYTES AS SPECIFIED BY THE 'F' AND 'C' BITS.

BITS 13-29 DESIGNATE THE MEMORY LOCATION FOR EACH TRANSFER. THE MEMORY ADDRESS IS EITHER A WORD, HALFWORD, OR BYTE ADDRESS AS SPECIFIED BY THE 'F' AND 'C' BITS.

BITS 12, 30,31 ('F' AND 'C' BITS) SPECIFY THE FORMAT CODE FOR THE TRANSFER AS FOLLOWS:

FORMAT BITS			TRANSFER TYPE
F	C		
BIT 12	BIT 30	BIT 31	
0	0	0	WORD TRANSFER
0	Y	1	HALFWORD TRANSFER
1	X	X	BYTE TRANSFER

NOTES:

Y=0 SPECIFIES LEFT HALFWORD
Y=1 SPECIFIES RIGHT HALFWORD
XX = BYTE NUMBER AS FOLLOWS:

00 = BYTE 0
01 = BYTE 1
10 = BYTE 2
11 = BYTE 3

NOTES:
1. FOR TEST DEVICE LEVEL 2000 INSTRUCTIONS, THE TCW MUST SPECIFY A HALFWORD TRANSFER TO EITHER THE LEFT OR RIGHT HALFWORD.
2. REFER TO TABLE 2-2 FOR TCW DEDICATED MEMORY LOCATIONS.

Figure 2-2. Transfer Control Word (TCW) Format

Table 2-2. Transfer Interrupt (TCW) Dedicated Memory Locations

Memory Dedicated Address (H)	Typical CD or TD Address (H) (See Note 1)	Function	I/O Controller Service Interrupt Level (H)
100	00	Input/Output Controller 0	14
104	04	1	15
108	08	2	16
10C	0C	3	17
110	10	4	18
114	18	5	19
118	20	6	1A
11C	30	7	1B
120	40	8	1C
124	50	9	1D
128	60	10	1E
12C	70	11	1F
130	78	12	20
134	7A	13	21
138	7C	14	22
13C	7E	Input/Output Controller 15	23

Notes: 1. Typical CD and TD Device addresses refer to the address configurations most commonly used to address the respective I/O controller. The Typical CD and TD addresses are CD or TD instructions bits 06-12 which are configured as follows:

a. Hex digit 1 represents instruction bits 06-08.

b. Hex digit 2 represents instruction bits 09-12.

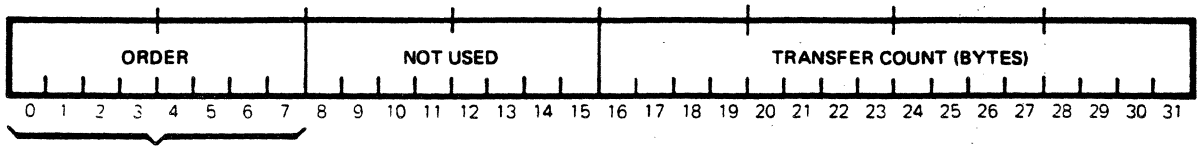
2. The TD memory location is used to hold the Transfer Control Word (TCW) for the corresponding I/O controller.

2-18 INTERRUPT CONTROL INSTRUCTIONS

The software Interrupt Control instructions control the interrupt level provided by the TLC. Figure 2-6 shows the format of the Interrupt Control instructions and lists the Interrupt Control functions that can be commanded by the instruction. Each of the 128 priority interrupts in the CPU has a dedicated memory location that contains the address of a software program that handles the interrupt. This memory location is accessed whenever an interrupt occurs from the corresponding interrupt level. If the interrupt level is enabled and the level has the highest priority, the program addressed by the dedicated memory location is initiated. Table 2-4 lists the priority levels available to the CPU and the dedicated memory location (Service Interrupt address) for each level. Note that only levels 14_H through 23_H are applicable to the I/O controllers. Interrupt levels 02 through 11 are dedicated to transfer interrupts, and the contents of these locations is the TCW for the corresponding I/O controller.

FORMAT A

DEVICE CLASS 0, 1, OR 2
INPUT/OUTPUT COMMAND DOUBLEWORD - WORD 0



ORDER

	IOCD WORD 0 BITS								FUNCTION
	0	1	2	3	4	5	6	7	
	M	M	M	M	M	0	0	1	BASIC WRITE - CD INITIALIZE DATA OUTPUT
	M	M	M	M	M	0	1	0	BASIC READ - CD INITIALIZE DATA INPUT
	M	M	M	M	M	1	1	0	CD CONTROL (NON-DATA TRANSFER)
DEVICE CLASS 0 (LINE PRINTER)	20	NU	21	22	23	X	X	X	CD INSTRUCTION BITS TRANSLATED INTO IOCD
	0	0	0	0	0	0	0	1	WRITE (PRINT DATA WITH NO PAPER ADVANCE)
	0	0	21	22	23	0	0	1	ADVANCE PAPER THE NUMBER OF LINES SPECIFIED BY CD BITS 21, 22, AND 23 THEN PRINT DATA
	1	0	21	22	23	0	0	1	ADVANCE PAPER TO FORMAT LOOP COLUMN SPECIFIED BY CD BITS 21, 22, AND 23 THEN PRINT DATA
	0	0	21	22	23	1	1	0	ADVANCE PAPER THE NUMBER OF LINES SPECIFIED BY CD INSTRUCTION BITS 21, 22, AND 23 (NO-PRINT)
	1	0	21	22	23	1	1	0	ADVANCE PAPER TO THE FORMAT LOOP COLUMN SPECIFIED BY CD INSTRUCTION BITS 21, 22, AND 23 (NO-PRINT)
DEVICE CLASS 1 (CARD READER)	X	NU	$\frac{20}{21}$	NU	21	0	X	0	CD INSTRUCTION BITS TRANSLATED INTO IOCD
	1	0	0	0	0	0	1	0	READ IN FULL ASCII MODE (ONLY USED IN CPU INITIALIZATION-IPL SEQUENCE)
	0	0	1	0	0	0	1	0	READ IN HALF ASCII MODE (TRANSLATE MODE)
	0	0	0	0	1	0	1	0	READ IN AUTOMATIC MODE
	0	0	0	0	0	0	1	0	READ IN BINARY MODE
DEVICE CLASS 2 (TELETYPEWRITER OR CRT)	17	17	17	17	20	0	X	X	CD INSTRUCTION BITS TRANSLATED INTO IOCD
	0	0	0	0	0	0	0	1	WRITE TO PRINTER
	0	0	0	0	0	0	1	0	READ FROM KEYBOARD
	0	0	0	0	1	0	1	0	READ FROM KEYBOARD THEN WRITE TO PRINTER (ECHO MODE)
	1	1	1	1	0	0	0	0	ILLEGAL COMMAND CAUSED BY CD BIT 17

KEY:
- 'M' = MODIFY BIT USED TO MODIFY THE BASIC WRITE, READ, OR CD CONTROL IOCD ORDERS.
- 'X' = BITS AVAILABLE FOR DEFINING THE BASIC ORDER FUNCTION (WRITE, READ OR CD CONTROL)
- 17,20,21,22, & 23 = CD INSTRUCTION BITS THAT ARE TRANSLATED INTO IOCD ORDER 'M' BITS.
- $\overline{20}$ $\overline{21}$ = FALSE CONDITION OF CD INSTRUCTION BITS 20 AND 21 ARE REQUIRED TO GENERATE THIS BIT IN THE IOCD ORDER.
- NU = BIT NOT USED.

DEVICE CLASS 0, 1, OR 2
INPUT/OUTPUT COMMAND DOUBLEWORD - WORD 0

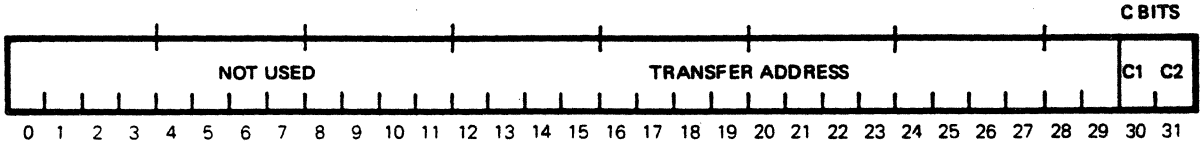


Figure 2-3. Input/Output Command Doubleword (IOCD) Format

The Branch and Reset Interrupt (BRI) instruction is also used by software to clear the active condition of the TLC controller interrupt level, although the BRI is not one of the Interrupt Control instructions. A description of the format and functions of the BRI instruction can be found in any SYSTEMS 32 SERIES Computer Reference Manual.

2-19 CPU INITIAL PROGRAM LOAD (IPL)

The CPU IPL sequence is entirely controlled by the CPU and is used to load software and hardware parameters that describe the system configuration into the CPU firmware control logic. The IPL sequence also loads a bootstrap software program into memory and initiates the execution of the bootstrap. The execution of the bootstrap causes the operating system program to be loaded into memory.

A complete and more detailed description of the IPL sequence can be found in the IOM Technical Manual, Publication Number 325-329000.

2-20 SelBUS TRANSFERS

During the execution of the I/O command, a series of SelBUS transfers occurs. The SelBUS can handle a total of 12 types of transfers; these are:

1. Write Data or Order Transfer (WDOT)
2. Read Data Transfer (RDT)
3. Interrupt Control Transfer (ICT)
4. Read Status Transfer (RSTX)
5. Advance Read Status Transfer (ARSTX)
6. Advance Interrupt Control Transfer (AICT)
7. Data Return Transfer (DRT)
8. Error Transfer (ET)
9. Memory Write Transfer (MWT)
10. Memory Read Transfer (MRT)
11. Memory Read and Lock Transfer (MRLT)
12. Memory Instruction Read Transfer (MIRT)

2-21 I/O MICROCOMMANDS

The TLC firmware, which is implanted in the control memory (PROM), uses a 32-bit microcommand word format (Figure 2-7). As data

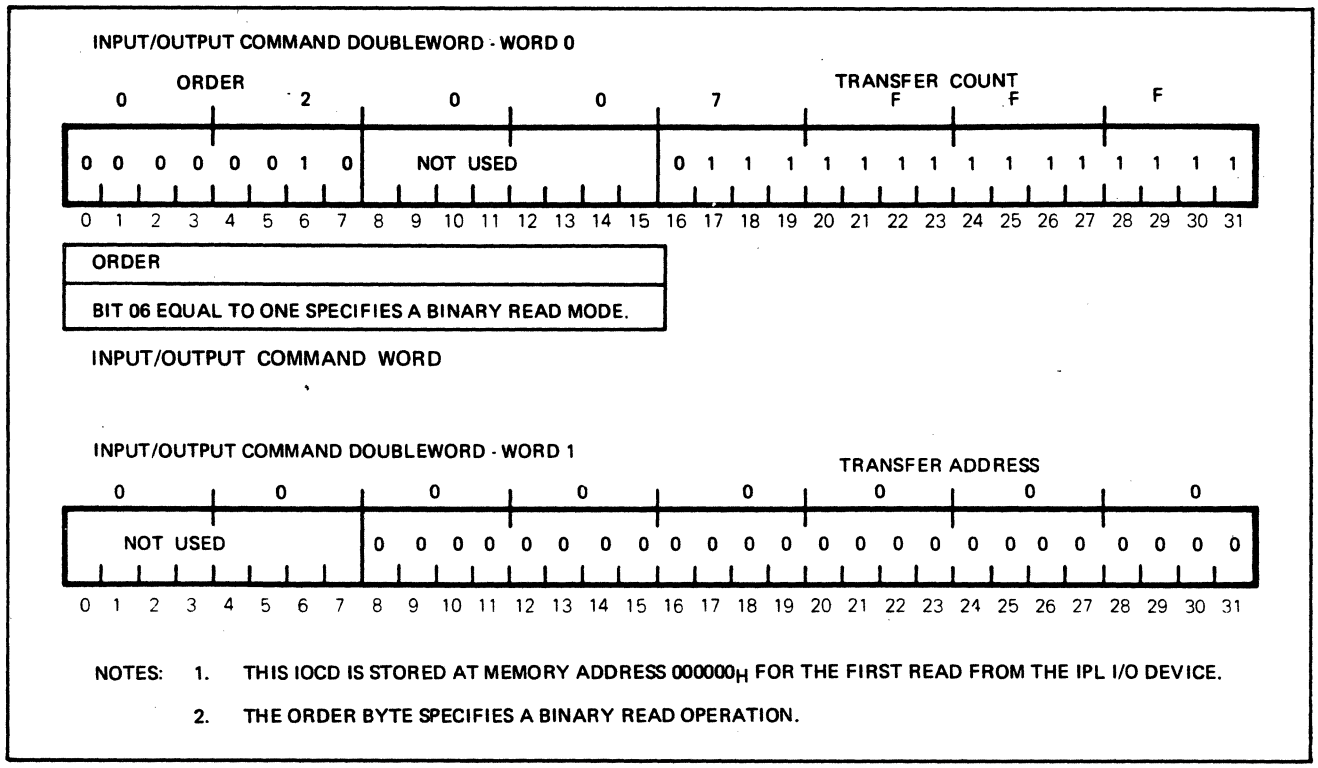


Figure 2-4. IPL Basic Input/Output Command Doubleword (IOCD) Format

and control signals are received by the MP from the SelBUS or the external device, the MP sequences through its program at a rate of 6.66 million microcommands per second. The MP interprets all incoming control signals, generates its own control signals, and passes data to and from the external device and the SelBUS.

Since the machine language instructions and their macro instruction components are processed by the CPU, they are not discussed here. The remainder of this discussion describes in more detail the operation of the SelBUS interface which responds to SelBUS transfers, and the MP which processes the 32-bit microcommands.

2-22 TELETYPEWRITER (TTY)

2-23 CD INSTRUCTION

The TTY system can operate on eight command functions. These function codes are unique to the TTY system and should not be confused with those for other devices. Figure 2-8 shows the CD instruction, and Table 2-5 defines the CD function codes in hexadecimal format.

2-24 TD INSTRUCTION

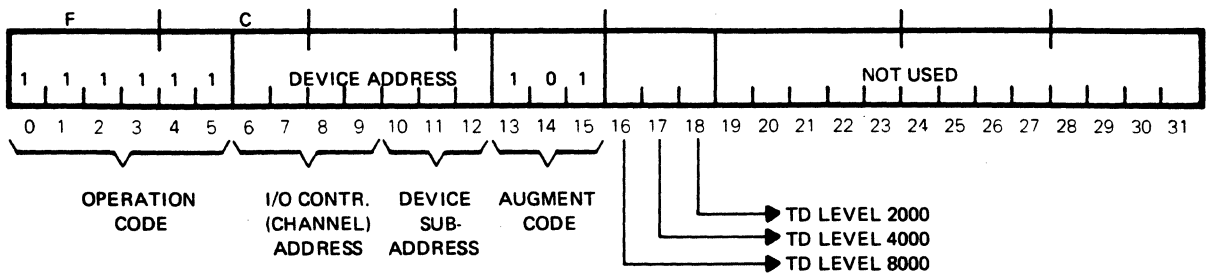
The initial status level of the TTY is obtained by executing a TD 8000_H instruction. This instruction tests for Channel Active, Device Controller Channel Error, and Device Abnormal. If any of these conditions are found, an associated bit is returned to the condition code field of the Program Status Word Register (PSWR).

If testing at this level returns a Device Controller Channel Error, further testing by execution of a TD 4000_H instruction determines the specific error. Execution of a TD at this level returns the following information: Invalid Memory Access, Memory Parity Error, Program Violation, Overflow, or Underflow. Figure 2-9 and Tables 2-6 and 2-7 support the preceding discussion on TD instructions.

Table 2-3. Input/Output Command Doubleword (IOCD) Dedicated Memory Locations

I/O Controller Service Interrupt Level (H)	IOCD Memory Dedicated Address (H)	Used With
14	700	Input/Output Controller 0
15	708	1
16	710	2
17	718	3
18	720	4
19	728	5
1A	730	6
1B	738	7
1C	740	8
1D	748	9
1E	750	10
1F	758	11
20	760	12
21	768	13
22	770	14
23	778	Input/Output Controller 15

Note: During the CPU Firmware IPL sequence all IOCD's for the IPL I/O device are stores at memory location 000000_H.

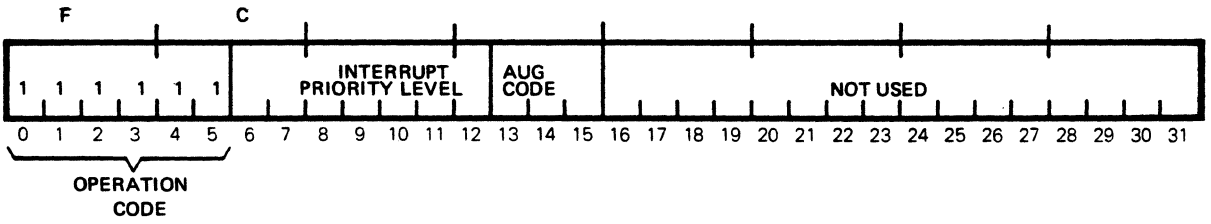


TD LEVEL	CONDITION CODE RESPONSE			
	CC1	CC2	CC3	CC4
TD 8000	DELAY	I/O CHANNEL ACTIVE (BUSY)	TD 4000 STATUS PRESENT	TD 2000 STATUS PRESENT
TD 4000	NON-PRESENT MEMORY ACCESS	MEMORY PARITY ERROR	PROGRAM VIOLATION	DATA UNDERFLOW OR OVERFLOW
TD 2000	**	STATUS TRANSFER NOT PERFORMED	**	**

NOTE: 1. ** INDICATES I/O CONTROLLER DEPENDENT CONDITION CODE

2. IF ALL CONDITION CODES AT ANY TD LEVEL ARE TRUE, THE I/O CONTROLLER IS NOT PRESENT OR TOTALLY INOPERABLE.

Figure 2-5. Test Device (TD) Instruction Format



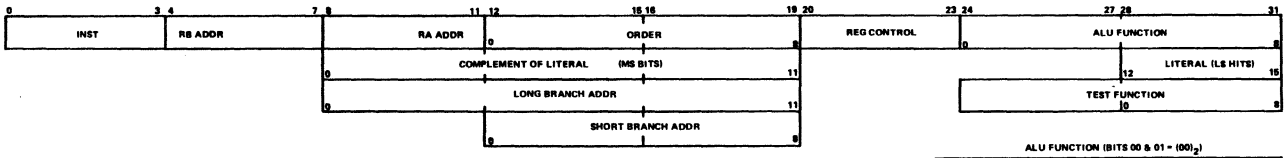
AUGMENT CODE			
BITS			INSTRUCTION FUNCTION
13	14	15	
0	0	0	ENABLE INTERRUPT
0	0	1	DISABLE INTERRUPT
0	1	0	REQUEST INTERRUPT
0	1	1	ACTIVATE INTERRUPT
1	0	0	DEACTIVATE INTERRUPT

Figure 2-6. Interrupt Control Instruction Format

Table 2-4. Priority Interrupt Dedicated Memory Locations

Priority Level (H)	Memory Dedicated Address (H)	Function
00*	0F0	Power Fail Safe - Auto Start Interrupt
00*	0F4	Power Fail Safe - Auto Start Trap
01*	0F8	System Override Interrupt
01*	0FC	System Override Trap
02**	100	Input/Output Controller 0 Transfer Interrupt
03**	104	Input/Output Controller 1 Transfer Interrupt
04**	108	Input/Output Controller 2 Transfer Interrupt
05**	10C	Input/Output Controller 3 Transfer Interrupt
06**	110	Input/Output Controller 4 Transfer Interrupt
07**	114	Input/Output Controller 5 Transfer Interrupt
08**	118	Input/Output Controller 6 Transfer Interrupt
09**	11C	Input/Output Controller 7 Transfer Interrupt
0A**	120	Input/Output Controller 8 Transfer Interrupt
0B**	124	Input/Output Controller 9 Transfer Interrupt
0C**	128	Input/Output Controller 10 Transfer Interrupt
0D**	12C	Input/Output Controller 11 Transfer Interrupt
0E**	130	Input/Output Controller 12 Transfer Interrupt
0F**	134	Input/Output Controller 13 Transfer Interrupt
10**	138	Input/Output Controller 14 Transfer Interrupt
11**	13C	Input/Output Controller 15 Transfer Interrupt
12*	0E8	Memory Parity Trap
13*	0EC	Console Interrupt (Turnkey Panel Attention)
14	140	Input/Output Controller 0 Service Interrupt
15	144	Input/Output Controller 1 Service Interrupt
16	148	Input/Output Controller 2 Service Interrupt
17	14C	Input/Output Controller 3 Service Interrupt
18	150	Input/Output Controller 4 Service Interrupt
19	154	Input/Output Controller 5 Service Interrupt
1A	158	Input/Output Controller 6 Service Interrupt
1B	15C	Input/Output Controller 7 Service Interrupt
1C	160	Input/Output Controller 8 Service Interrupt
1D	164	Input/Output Controller 9 Service Interrupt
1E	168	Input/Output Controller 10 Service Interrupt
1F	16C	Input/Output Controller 11 Service Interrupt
20	170	Input/Output Controller 12 Service Interrupt
21	174	Input/Output Controller 13 Service Interrupt
22	178	Input/Output Controller 14 Service Interrupt
23	17C	Input/Output Controller 15 Service Interrupt
24*	190	Nonpresent Memory Trap
25*	194	Undefined Instruction Trap
26*	198	Privilege Violation Trap
27*	19C	Call Monitor Interrupt
28*	1A0	Real-Time Clock Interrupt
29*	1A4	Arithmetic Exception Interrupt
2A*	1A8	External Interrupt
2B*	1AC	External Interrupt
2C*	1B0	External Interrupt
2D*	1B4	External Interrupt
2E*	1B8	External Interrupt
2F*	1BC	External Interrupt (Last PI in standard RTOM)
30	1C0	External Interrupt
↕	↕	↕
7F	2FC	External Interrupt

* Present in first RTOM.
** These dedicated addresses are reserved for Transfer Control Words (TCW) and cannot be used by Priority Interrupt software.



INST
0 NORMAL
1 NORMAL ENBINT
2 NORMAL ENB ORDER
3 NORMAL ENB INT & ORDER
4 WAIT
5 WAIT ENBINT
6 WAIT ENB ORDER
7 WAIT ENB INT & ORDER
8 SHORT BRANCH
9 SHORT BRANCH ENB INT
A INDIRECT SHORT BRANCH
B
C LONG BRANCH
D LONG BRANCH ENB INT
E INDIRECT LONG BRANCH
F

ORDER BIT 02 = 1			
L/H LEVEL		PULSE	
MICRORETRY 01/B1	ACTIVE 02/B2	CLEARS PC16 04	CLR.HALTIO 08
TEST 3 11/B1	MICROBUSY 12/B2	STBLPDATA 14	LBRANCH-SEL 18
TEST1 21/A1	DPLPCLEAR 22/A2	LCLRINT 24	RDCARD 28
TTYSTBINDAT 31/B1	DPPAPERINST 32/B2	RSTCRDTHERE 34	RSTCRDLST 38
RSTDATAVAL 41/C1	STOPRD 42/C2	* 44	LREQINT 48
MICRORST 51/D1	* 52/D2	LMICROACK 54	LMICRODATALD 58
ENINBUSA 61/E1	DPLPSTROBE 62/E2	LMICROTRANS 64	LMICRODESTLD 68
ENINBUSB 71/F1	ENINBUSC 72/F2	* 74	LMICROREADY 78

TEST FUNCTION BITS 00 & 01=(10, 11, OR 01)$_2$		REG CONTROL
SIN08 20/A0	SIN00 40/C0	0
PEOIN 21/A1	SIN01 41/C1	PC TO A FILE 1
SIN10 22/A2	SIN02 42/C2	EXT DEV TO B FILE 2
SIN11 23/A3	SIN03 43/C3	3
SIN12 24/A4	SIN04 44/C4	SEL DATA TO A & B FILE 4
SIN13 25/A5	SIN05 45/C5	5
SIN14 26/A6	SIN06 46/C6	6
SIN15 27/A7	SIN07 47/C7	7
AMSB-FF 28/A8	AEQB-FF 48/C8	8
MEMDATAHERE 29/A9	NONPRSMEM 49/C9	DEST TO A FILE 9
RDIN 2A/AA	P3EIN 4A/CA	LITERAL TO B FILE A
CNTOIN 2B/AB	MICROINPUT 4B/CB	B
CNT1IN 2C/AC	P2EIN 4C/CC	ALU TO A & B FILE C
P1EIN 2D/AD	SYNC 4D/CD	ALU TO A FILE D
CTRL2 2E/AE	UNCONDITIONAL 4E/CE BRANCH (+V)	ALU TO B FILE E
ERRORIN 2F/AF	CPU DATA HERE 4F/CF	CHECK ALU OUTPUT F
RA00 30/B0	RA06 50/D0	
RA07 31/B1	RA05 51/D1	
RA08 32/B2	CRBUSY 52/D2	
CRDATALST 33/B3	CTRL4 53/D3	
CRDATAHERE 34/B4	CTRL3 54/D4	
DPLPREADY 35/B5	* 55/D5	
TTYDATAREQ 36/B6	RA04 56/D6	
TTYDATAHERE 37/B7	RA03 57/D7	

ALU FUNCTION (BITS 00 & 01 = $(00)_2$)					
DO NOT SAVE ALU STATUS (BIT 31=0)			SAVE ALU STATUS (BIT 31=1)		
CODE	FUNCTION	CARRY	CODE	FUNCTION	CARRY
00	A-1	F	01	A-1	F
08	A		09	A	
30	−1	F	31	−1	F
32	ZERO	T	32	ZERO	T
58	B		59	B	
60	A-B-1	F	61	A-B-1	F
62	A-B	T	63	A-B	T
90	A+B	F	91	A+B	F
92	A+B+1	T	93	A+B+1	T
94	A+B+CARRY	LAST	95	A+B+CARRY	LAST
98	A!B		99	A!B	
A8	B		A9	B	
B8	A:B		B9	A:B	
C0	A+A (SHIFT LEFT)	F	C1	A+A (SHIFT LEFT)	F
C6	A+A+MSB (SHIFT CIRCULAR)	MSB	C7	A+A+MSB (SHIFT CIRCULAR)	MSB
C8	ZERO		C9	ZERO	
E8	A & B		E9	A & B	
F8	A		F9	A	

*AVAILABLE FOR THE DEVICE DEPENDENT INTERFACE

NOTE: THIS FORMAT DOES NOT PROVIDE THE 'LOAD LITERAL INTO A FILE' FUNCTION.

LEGEND	
ALU	CARRY
+ = ADD	F = FALSE
− = SUBTRACT	T = TRUE
& = LOGICAL AND	MSB= GENERATE FROM MOST SIGNIFICANT INPUT BIT
! = LOGICAL OR	LAST = GENERATE FROM CARRY STATUS THAT WAS SAVED
⊻ = EXCLUSIVE OR	
A = ONES COMPLEMENT	

Figure 2-7. TLC Microcommand Format

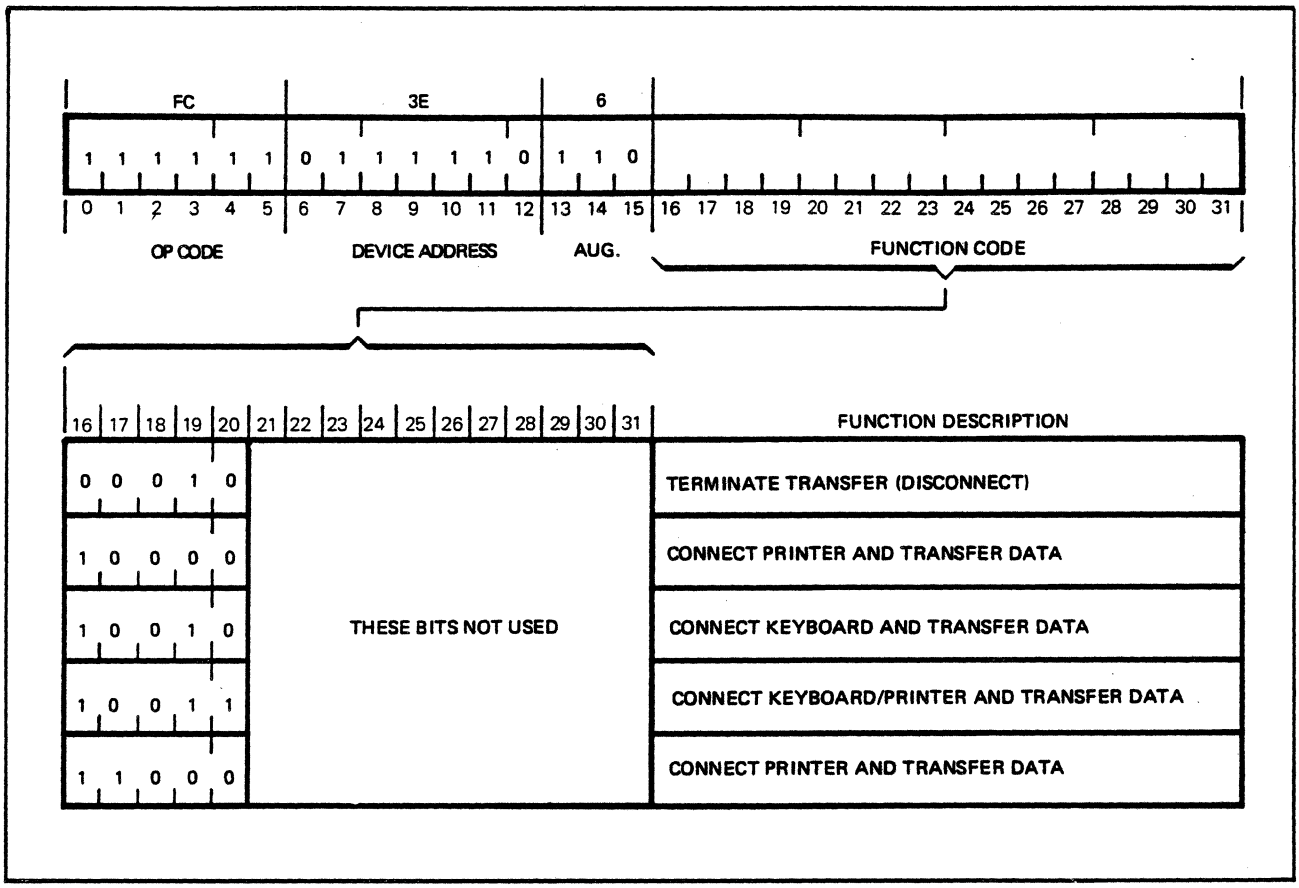


Figure 2-8. Command Device Format TTY System

Table 2-5. Command Device Function Codes (TTY)

16-Bit Function Code (H)	Function	Definition
1000	Control	Execution of a Command Device with this function code causes a Terminate Transfer (disconnect). This aborts the data transfer and shuts down the device in an orderly manner.
8000	Data Transfer	Execution of a Command Device with this function code connects the Page Printer to the Device Controller Channel and transfers data.
9000	Data Transfer	Execution of a Command Device with this function code connects the Keyboard to the Device Controller Channel and transfers keyed data.
9800	Data Transfer	Execution of a Command Device with this function code connects the Keyboard and Page Printer to the Device Controller Channel for transfer of keyed data with Character Turn-Around.
C000	Data Transfer	Execution of a Command Device with this function code connects the Page Printer to the Device Controller Channel and transfers data.

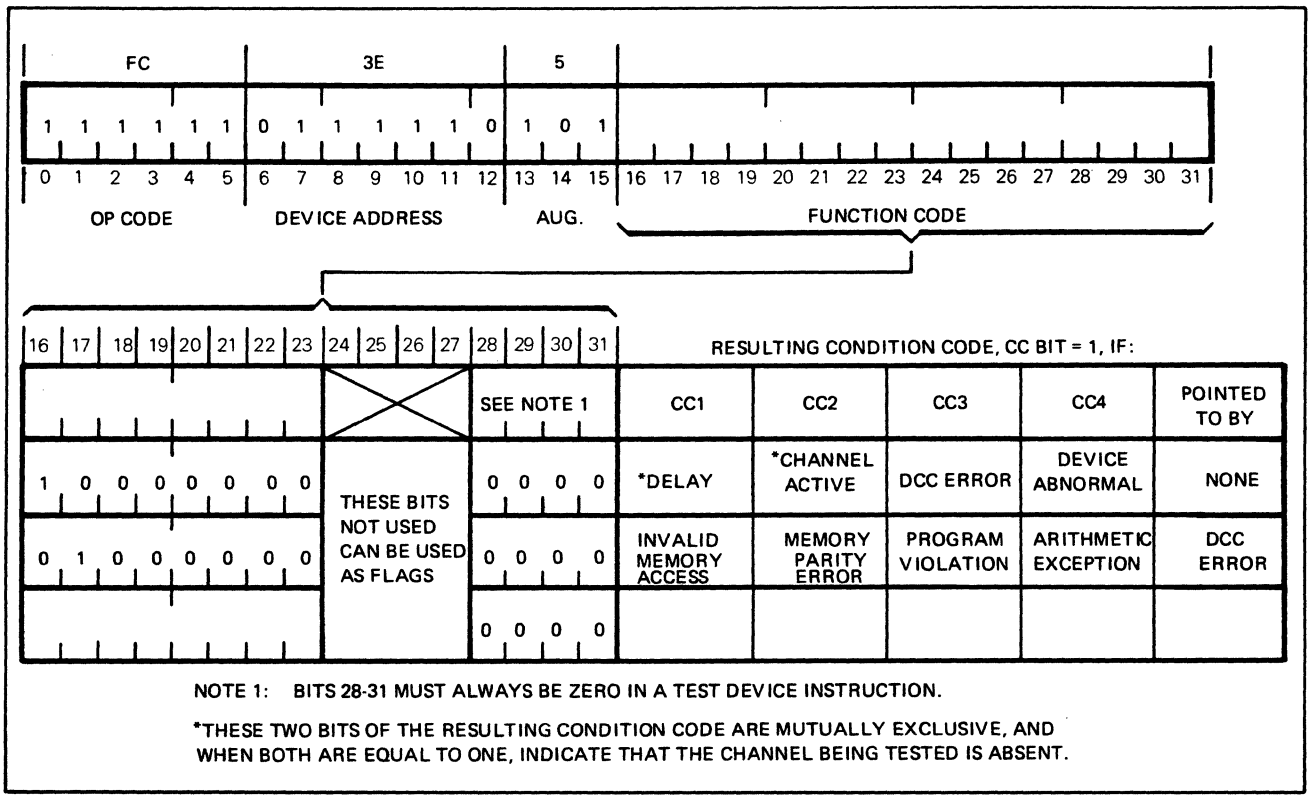


Figure 2-9. Test Device Format and Function Code TTY System

Table 2-6. Test Device Results, Level $8000_{(H)}$

Function Code$_{(H)}$	Condition Code	Designation	Definition
8000	CC1 = 1	Delay	Not applicable to this device.
8000	CC2 = 1	Channel Active	Indicates that the Device Controller Channel is performing a Data Transfer.
8000	CC3 = 1	Device Controller Channel Error	Indicates that an error condition was detected within the Device Controller Channel during the previous operation. The specific error can be determined by executing a Test Device at level $4000_{(H)}$.
8000	CC4 = 1	Device Abnormal	Indicates that an abnormal condition has been detected. In the keyboard/printer the presence of this bit indicates that the device is Inoperable: Power Off, in Local Modes, etc.
8000	CC1 = 1 CC2 = 1 CC3 = 1 CC4 = 1	Nonexistent	Indicates that the channel to which the Test was directed is nonexistent.
8000	CC1 = 1 CC2 = 1 CC3 = 0 CC4 = 0	Device Not Available	Not applicable to this device.

Table 2-7. Test Device Results, Level $4000_{(H)}$

Function Code$_{(H)}$	Condition Code	Designation	Definition
4000	CC1 = 1	Invalid Memory Access	Indicates that the Device Controller Channel has addressed a memory module that is either not present in the CPU or, in a Multiprocessor Configuration, to which the Device Controller Channel is denied access.
4000	CC2 = 1	Memory Parity Error	Indicates that a Memory Parity Error was detected while the Device Controller Channel was fetching data or a Transfer Control Word from memory.
4000	CC3 = 1	Program Violation	Indicates that a Command Device was executed that could not be performed by the device. This can occur when a command is issued before the Device Controller Channel has completed a current operation. The Service Interrupt is generated when the current operation is completed, and this bit is provided to indicate the Program Violation.
4000	CC4 = 1	Overflow or Underflow	Indicates that data was lost while being transferred from the Device Controller Channel or memory to the peripheral device (Overflow) or from the Peripheral Device to the Device Controller Channel or memory (Underflow).

2-25 LINE PRINTER

2-26 CD INSTRUCTION

The CD instruction causes the transfer of an 8-bit instruction command code to the TLC Controller. The CD instruction command codes are shown in Figure 2-10. The bit assignments for the CD command codes are as follows:

1. Bit 16 equals One - Initialize Output Data Transfer
2. Bit 17 equals One - Reserved for future expansion.
3. Bit 19 equals One - Terminate Operation
4. Bits 20 through 23 - Paper Line Advance Control bits
5. Bits 24 through 31 - These bits are not used with the line printer

2-27 TD INSTRUCTION

The TD instruction tests the status information to determine the condition of the line printer Device Controller Channel (DCC) and the line printer device. The TD instruction format is shown in Figure 2-11.

1. Bit 16 equals One - Select Test Level 8000 (Initial Conditions)
2. Bit 17 equals One - Select Test Level 4000 (DCC Errors)
3. Bit 18 equals One - Select Test Level 2000 (Device Status)

The initial condition of the line printer is obtained by executing a TD 8000 instruction. This instruction tests for Channel Active, DCC Error, and Device Abnormal. If any of these conditions are found, an associated bit is returned to the condition code field of the PSWR.

If an 8000 level test shows an error in the DCC, a TD 4000 instruction is used to determine the specific error.

If an 8000 level test shows an abnormal status in the device, a TD 2000 instruction is used to determine the fault within the device. Tables 2-8 through 2-12 support the previous discussion.

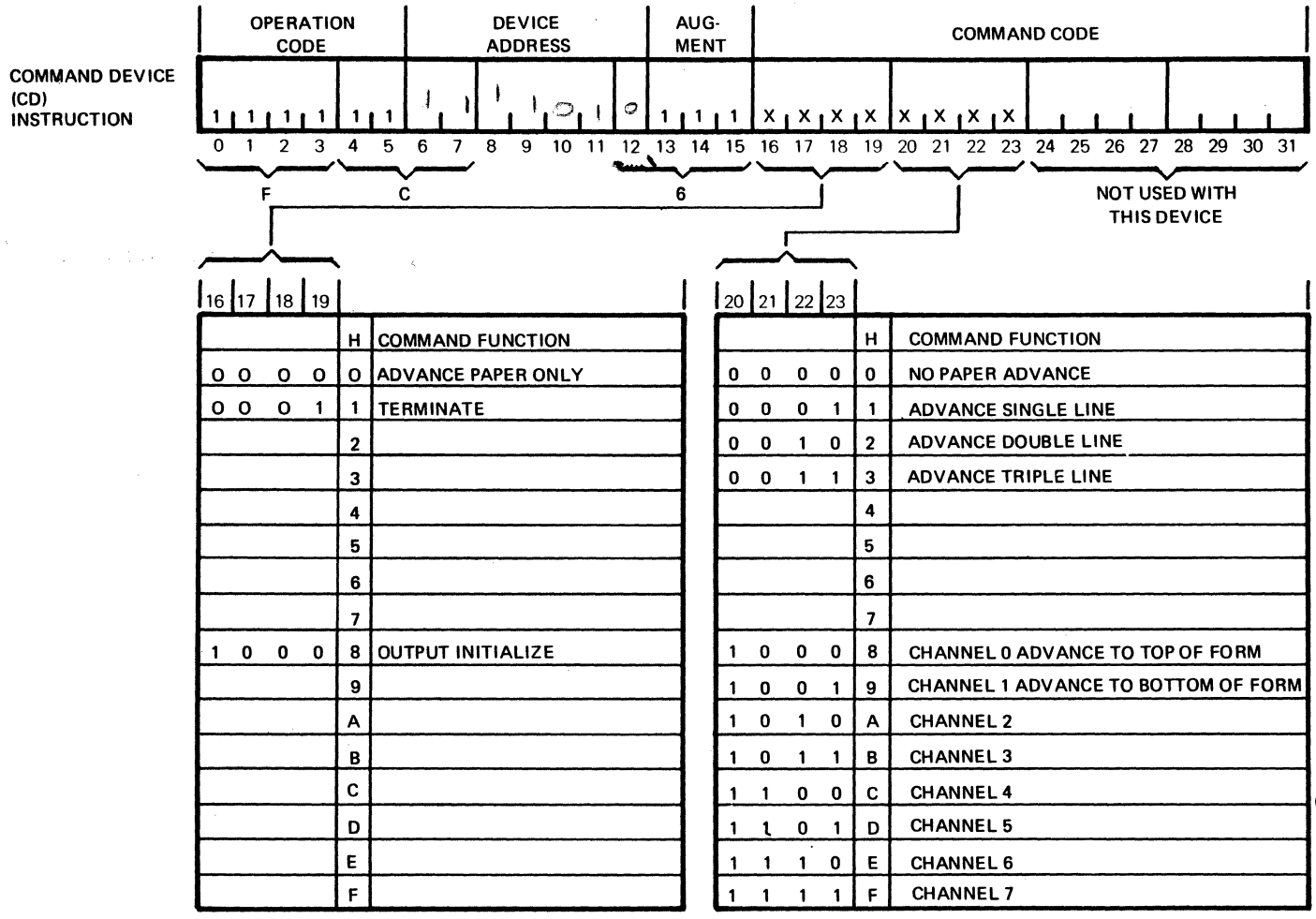


Figure 2-10. Line Printer Command Device Instruction Format

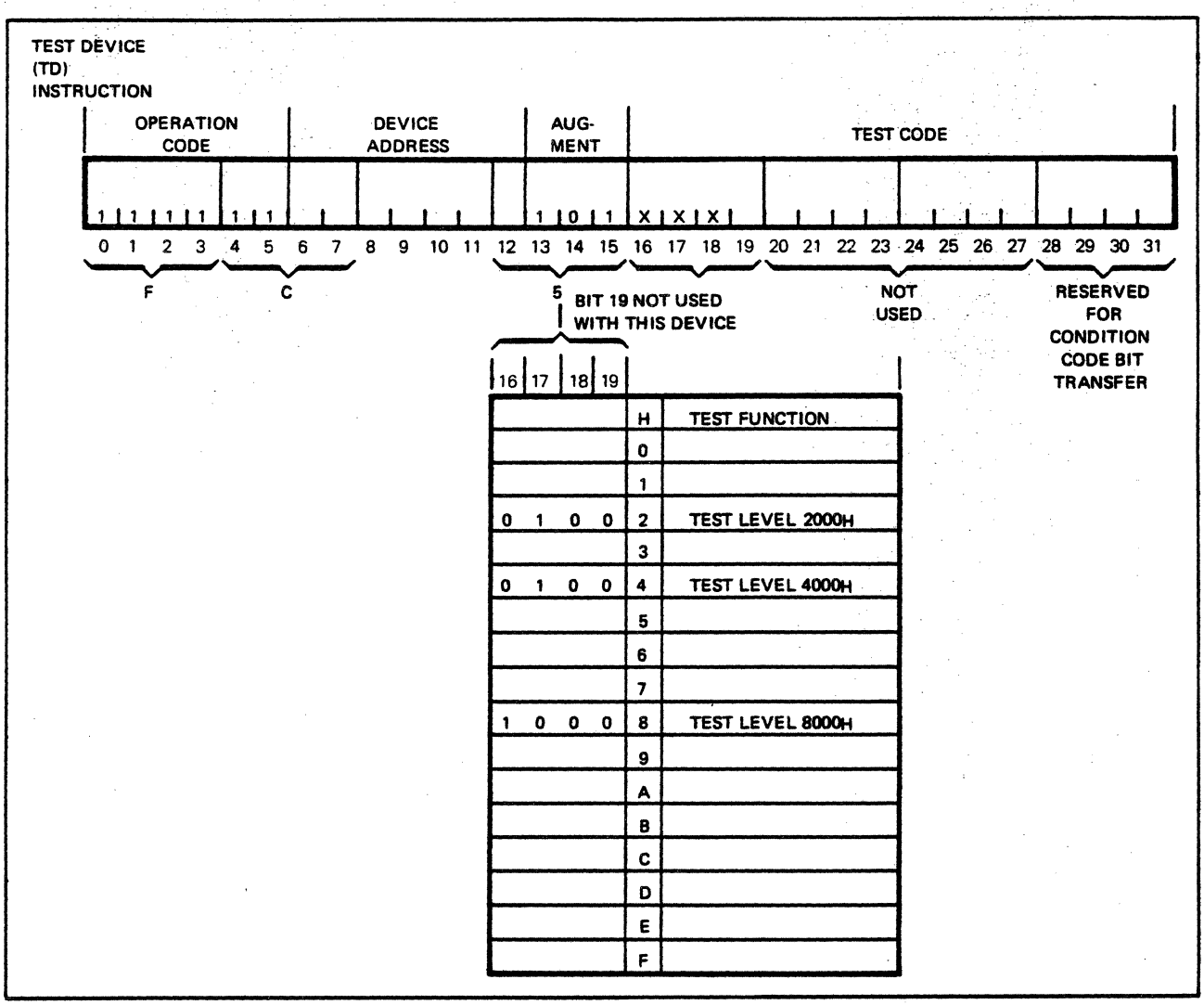


Figure 2-11. Line Printer Test Device Instruction Format

Table 2-8. 8000 Level Condition Code Definitions

Condition	Circumstances
(8000_H, CC_1)	Not used in Printer.
Channel Active (8000_H, CC_2)	The Device Controller Channel (DCC) is presently performing data or non-data commands.
DCC Error (8000_H, CC_3)	An error condition was detected within the DCC during the previous operation. The specific condition may be determined by executing a Test Device instruction with a test code of 4000_H.
Device Status (8000_H, CC_4)	An abnormal condition is present in the device. The specific condition may be determined by executing a Test Device instruction with a test code of 2000_H.

Table 2-9. 4000 Level Condition Code Definitions

Condition	Circumstances
Invalid Memory Access (4000_H, CC_1)	The Device Controller Channel addressed a memory module that is either not present in the computer system or is in a multiprocessor configuration to which it is denied access.
Memory Parity Error (4000_H, CC_2)	A Memory Parity Error was detected in the previous operation while the Device Controller Channel was obtaining data or a Transfer Control Word from memory.
Program Violation (4000_H, CC3) (DCC or Device)	A Command Device instruction has been executed that could not be performed by the specified device. This can occur when a command is issued before the the Device Controller Channel has completed the current operation. The Service Interrupt is generated at the end of the current operation, and this indicator is provided.
Overflow (4000_H, CC_4)	Data has been lost while being transferred from the computer system memory to the device (Overflow). There is no Underflow circumstance with the Line Printer Device Controller Channel, as all transfers are in the output direction.

Table 2-10. 2000 Level Status Bit Definitions

Condition	Circumstances
Program Violation Halfword Bit 1/17	A Command Device instruction has been executed that cannot be performed by the specified device. The Service Interrupt is generated and the instruction is terminated.
Device Inoperable Halfword Bit 2/18	The Line Printer cannot be operated in its present state due to one (or more) of the following: 1. Drum Arm Open 2. VFU Reader Open 3. Hammer Fuse Open 4. Printer Voltage Incorrect 5. Out of Paper 6. Device Absent 7. Power Off
Bottom-of-Form Halfword Bit 9/25	The Line Printer page is at the Bottom-of-Form (Channel 1) position.
Device Active (Busy) Halfword Bit 13/29	The Line Printer is Busy. The Busy time is defined as the interval from the acceptance of a Command Device instruction to the generation of the Line Printer Service Interrupt request, indicating that the operation has been completed.

Table 2-11. Line Printer DCC/Device Condition Code Bit Assignment

TD Test Code	CC_1	CC_2	CC_3	CC_4
Test Level 8000_H Initial Conditions	Not Used	Channel Active	DCC Error	Device Status
Test Level 4000_H DCC Errors	Invalid Memory Access	Memory Parity Error	Program Violation (DCC)	Overflow

Table 2-12. Line Printer Device Status Bit Assignment

Left/Right Halfword Bit	1/17	2/18	13/29	9/25
Test Level 2000_H Device Status	Device Program Violation	Device Inoperable	Device Active (Busy)	Bottom-of-Form

2-28 CARD READER

2-29 CD INSTRUCTION

The TLC Controller Channel for both modules of the card reader have in common the ability to execute eight CD instruction function codes. Two of these function codes specify control operations; the other six initialize data transfers from the card reader. In addition, the TLC Controller Channel can execute a special control function code causing the card reader to offset a card in the stacker.

Figure 2-12 shows a complete breakdown of the CD function codes. The operations specified by these codes are further defined in Table 2-13.

2-30 TD INSTRUCTION

The TLC Controller Channel for the card reader can execute two levels of TD instructions. Response to either level is condition codes, which represent error conditions existing in either the TLC Controller Channel or the device itself.

Figure 2-13 shows the TD format, the function code, and the resulting condition codes for the card reader.

2-31 TD Level 8000_H

The initial level of status information is obtained by executing a TD 8000_H instruction. The condition codes defined in Table 2-14 are returned to the Condition Code (CC) field of the PSWR in the central processor.

2-32 TD Level 4000_H

If the condition code returned in response to the TD 8000_H instruction indicates a DCC error (CC3), the specific error can be identified by executing a TD 4000_H instruction. The condition codes resulting from this level are defined in Table 2-15.

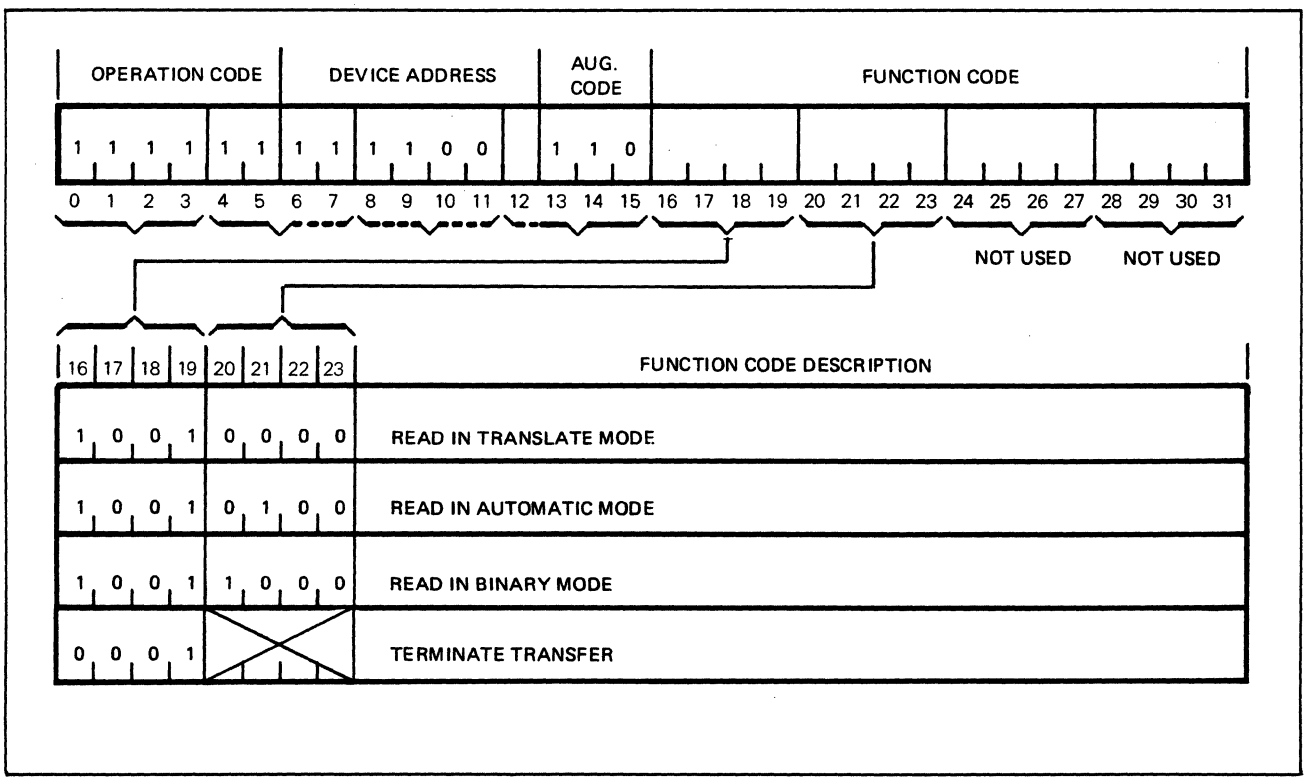


Figure 2-12. Card Reader Command Device Instruction Format

Table 2-13. Command Device Function Codes (CR)

Function Code (H)	Function	Description
1000	Control	Terminate transfer. This command inhibits further data transfers from the card reader Device Controller Channel.
9000	Data Transfer	This command causes a reading of data in the Translate Mode.
9400	Data Transfer	This command causes a reading of data in the Automatic Mode.
9800	Data Transfer	This command causes a reading of data in the Binary Mode.

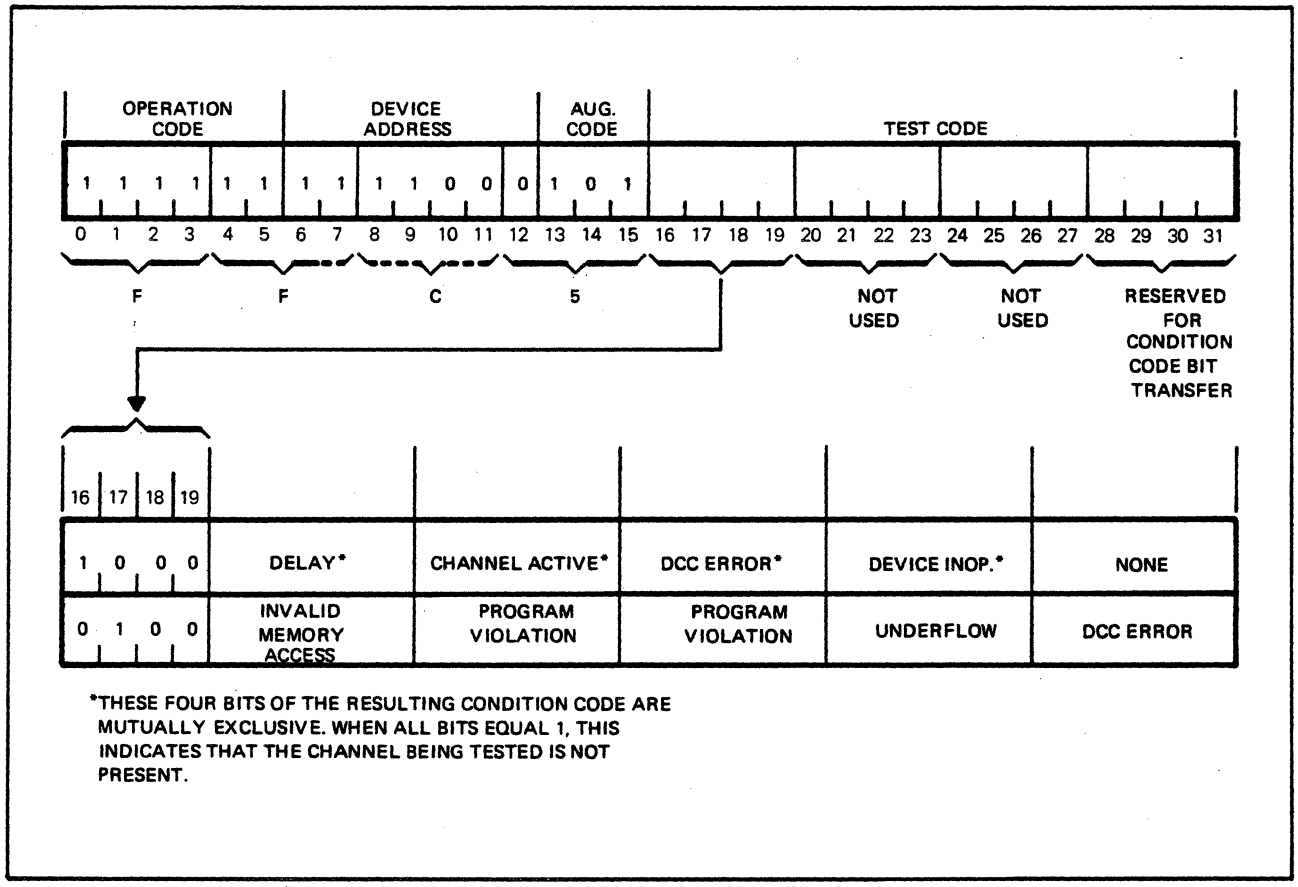


Figure 2-13. Card Reader Test Device Instruction Format

Table 2-14. Test Device Condition Codes - Level $8000_{(H)}$

Condition Code	Designation	Definition
CC1	Delay	Not applicable for this device; always equal to zero if device is present.
CC2	Channel Active	Indicates that the Device Controller Channel is performing a data transfer.
CC3	DCC Error	Indicates that an error condition exists in the Device Controller Channel. The specific error may be detected by executing a Test Device level $4000_{(H)}$.
CC4	Device Inoperable	Indicates that the Card Reader is inoperable.

Note: These four Condition Codes are mutually exclusive. When all Condition Codes equal 1, this indicates that the channel being addressed is not present.

Table 2-15. Test Device Condition Codes - Level $4000_{(H)}$

Condition Code	Designation	Definition
CC1	Invalid Memory Access	Indicates that the Device Controller Channel has addressed a memory module that is either not present in the CPU, or to which the Device Controller Channel is denied access.
CC2	Memory Parity Error	Indicates the detection of a Memory Parity Error during the fetching of a Transfer Control Word from memory in the previous operation.
CC3	Program Violation	Indicates the issue of a Command Device instruction that cannot be performed by this device. This can occur when a command is issued before the Device Controller Channel has completed the current operation. A Service Interrupt is Generated at the end of the current operation, and this indicator is provided.
CC4	Underflow	Indicates that data has been lost during transfer from the Card Reader to memory.

SECTION III

THEORY OF OPERATION

3-1 INTRODUCTION

The Theory of Operation for the SYSTEMS TLC Controller is divided into two levels of discussion. The first level is the general theory of operation which describes the purpose, basic organization, and overall operation of the TLC Controller. The second level is the detailed theory of operation which describes the operational characteristics of the logic components of the TLC Controller.

3-2 GENERAL THEORY

3-3 PURPOSE

The primary function of the TLC Controller is to execute input or output data transfers to the addressed teletypewriter, line printer, or card reader after the input/output (I/O) operation has been initiated by the controller computer software. Once the controlling computer has initiated the operation, the TLC Controller executes the data transfers between the computer's memory and the addressed peripheral device independently of the computer's operation. A data transfer operation may consist of a single transfer or a group of transfers. When the data transfer operation is complete, the TLC Controller generates an interrupt to the controlling computer, indicating that the transfer operation is complete. Before any I/O operation is initiated and at the termination of any I/O operation, the TLC Controller assembles its status and the status of the addressed peripheral device, and makes that status available to the controlling computer upon the computer's request.

The TLC Controller is designed to operate with the SYSTEMS 32 SERIES Computer and is intended to provide the basic interface with the computer for the teletypewriter, line printer, and card reader, thus providing a complete TLC controller on one circuit card. Note that in IOM applications, the bulk of control logic for the peripheral device is performed by the firmware microprogram in the MP of the TLC Controller.

3-4 TLC BASIC ORGANIZATION

The TLC Controller can be divided into three sections for discussion. The sections are the SelBUS interface, the MP, and the Device Dependent Interface logic. The SelBUS interface contains the logic necessary to receive or transmit data, status, or commands on the SelBUS. All modules of the computer system must interface with the SelBUS to communicate between the modules.

Figure 3-1 is a block diagram of a basic SYSTEMS 32 SERIES Computer system and shows the modules of the system that communicate using the SelBUS. Also shown is the SelBUS interface and MP sections of the IOM which, with the peripheral device dependent control electronics and custom firmware (microprogram) for the MP, provide a total TLC Controller that interfaces directly to the SelBUS.

The MP section of the TLC Controller provides the control of both the SelBUS interface and the peripheral device dependent controller under the direction of the microprogram (firmware). The firmware is in a Programmable Read-Only Memory (PROM) in the MP; however, the firmware is organized with respect to the TLC devices, since most of the functions performed by the control logic are performed by the firmware.

The peripheral Device Dependent interface contains the logic necessary to transmit/receive data, status, or commands to/from the peripheral device.

3-5 OVERALL OPERATION

The primary function of the TLC Controller is to control the execution of an input or output operation between the computer and the peripheral device. The input/output operation is executed independently of computer operations once the operation has been initiated by the computer software.

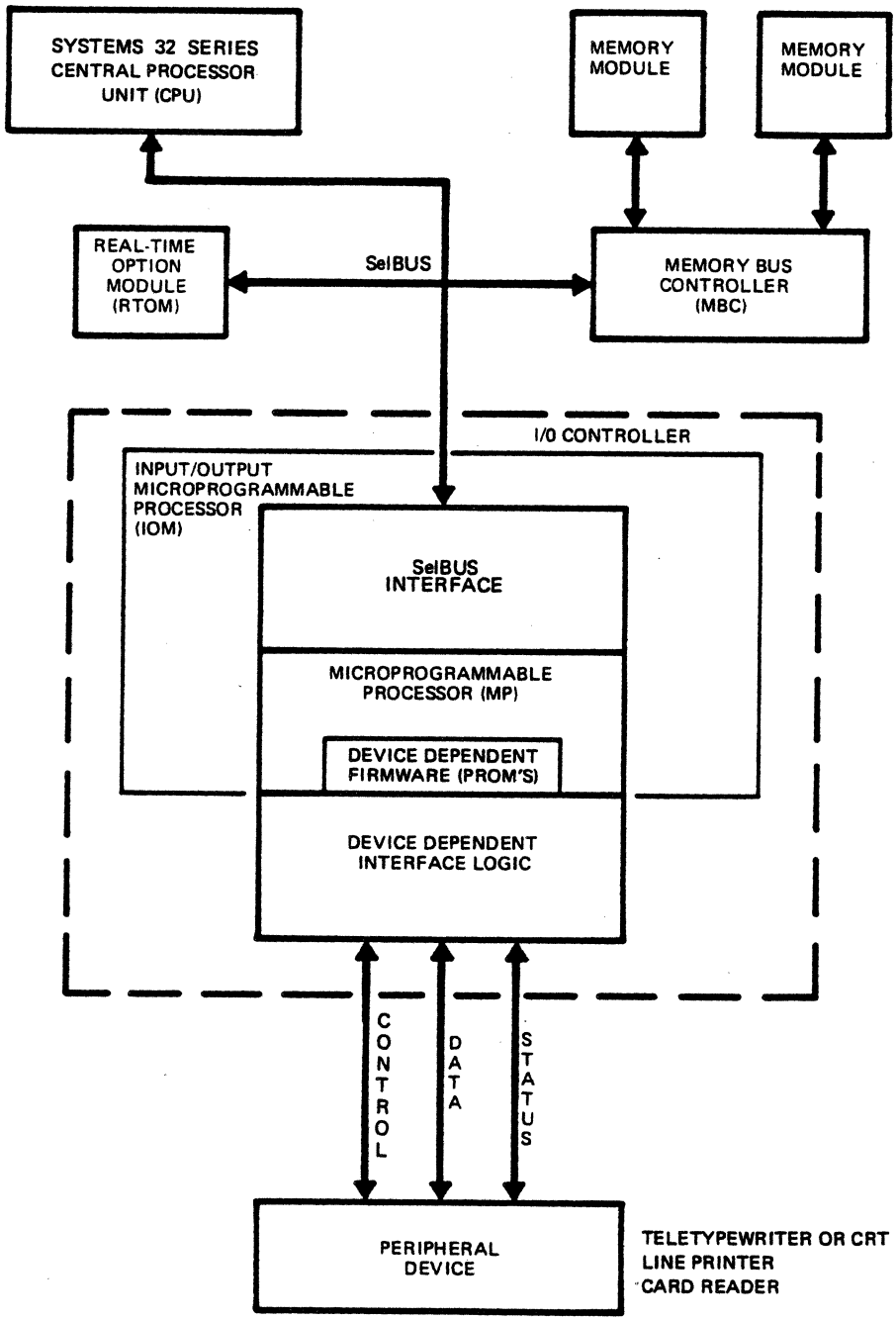


Figure 3-1. Block Diagram - SYSTEMS 32 SERIES Computer

The SYSTEM 32 SERIES Computer uses the following software instructions to control or initiate I/O operations in the IOM and peripheral device:

1. The TD instruction to obtain the status of the IOM and peripheral device.

2. The CD instruction to initiate data transfers or control operations within the IOM and peripheral device.

3. The Interrupt Control instructions to condition the IOM interrupt control logic for the following conditions:

   a. Request the Interrupt

   b. Enable the Interrupt

   c. Disable the Interrupt

   d. Activate the Interrupt

   e. Deactivate the Interrupt

The SYSTEMS 32 SERIES Computer firmware converts the software instructions into a series of SelBUS transfers to the IOM and peripheral device addressed by the software instruction. The SelBUS interface logic of the IOM detects the SelBUS transfer. If the address on the SelBUS matches the address of the IOM, the SelBUS interface logic stores the transfer for examination by the MP. After the SelBUS transfer has been stored, the SelBUS interface logic generates a microprogram interrupt to the MP. This microprogram interrupt causes the microprogram to branch to a subroutine that tests the SelBUS transfer information stored in the SelBUS interface. When the microprogram has identified the type of SelBUS transfer received, it takes the appropriate actions required to execute the operations directed by the SelBUS transfer. For example, if the SelBUS transfer requests status from the IOM and peripheral device, the microprogram assembles the addressed peripheral device's status and the IOM status. After the status is assembled, the microprogram signals the CPU firmware that it is ready to transfer the requested status. The CPU firmware responds to the Ready indication with a SelBUS transfer that tells the IOM microprogram to transfer the assembled status. The microprogram responds to this CPU firmware request by generating a SelBUS Data Return transfer which shifts the requested status from the IOM to the CPU completing the status transfer operation.

CD instructions are initiated by the CPU firmware which formats the command code function bits from the software CD instruction and the transfer count and memory address from the software TCW into an IOCD. After the IOCD has been assembled, the CPU firmware stores the IOCD in memory and requests status from the IOM and peripheral device addressed by the CD instruction. If the CPU firmware decides that the IOM and peripheral device status is satisfactory, the CPU firmware issues a SelBUS Start transfer to the IOM. This instruction specifies the memory address of the IOCD and indicates that the IOM is ready to start an I/O operation. The IOM microprogram uses the SelBUS Start transfer instruction to generate a Memory Read Transfer (MRT) to the Memory Bus Controller (MBC) using the SelBUS. The MRT specifies a Memory Read operation at the IOCD memory address. When the MBC completes the Memory Read, it generates a DRT to the IOM using the SelBUS. The IOM microprogram stores the first word of the IOCD in its internal registers and generates another MRT to the MBC to obtain the second word of the IOCD. When the MBC generates the DRT to the IOM, the IOM stores the second word of the IOCD in internal registers.

After the IOM fetches the IOCD from memory, it uses the command code function (order) bits from the IOCD to initiate the I/O operation requested by the CD instruction. If the I/O operation is a data transfer operation, the IOM microprogram performs the actual data transfers independently of the CPU. The IOM transfers data directly to/from the MBC using the SelBUS. The memory address for the transfers is provided by the memory address portion of the IOCD; the microprogram updates (increments) this address after every memory transfer. The transfer count of the IOCD is decremented after every memory transfer. When the transfer count is zero, the I/O operation is complete. The microprogram then causes the SelBUS interface of the IOM to generate a Service Interrupt (SI) to the CPU, indicating that the I/O operation is complete.

The SYSTEMS 32 SERIES Interrupt Control instructions are executed by the CPU firmware, which translates the software instruction into a SelBUS transfer to the IOM addressed by the Interrupt Control instruction. The SelBUS Interface logic of the IOM receives and stores the SelBUS transfer until the microprogram can test the SelBUS transfer information. When the microprogram determines the type of SelBUS transfer that has occurred, it conditions the Service Interrupt Control logic of the SelBUS interface section of the IOM as directed by the software Interrupt Control instruction.

The software formats of the TD, CD, and Interrupt Control instructions are illustrated in the SYSTEMS 32 SERIES Computer Reference Manual applicable to the system being used. The formats for the TCW and its dedicated memory locations are also illustrated in the same publication. Specific formats for the CD function code bits and the TD condition codes and status bits are a function of the peripheral device and controller used with the IOM, and are provided in this technical manual for the peripheral device and controller. Note that the above formats are all CPU software level formats and are not sent to the IOM. The CPU firmware reformats the instruction into SelBUS transfers to the IOM and the IOCD that is stored in memory. The actual formats of all transfers received from the SelBUS are provided in the SelBUS Interface - Detailed Theory discussion of the IOM Technical Manual, Publication Number 325-329000, and in the SYSTEMS 32 SERIES Computer Technical Manual applicable to the system being used.

3-6 PERIPHERAL DEVICE CONTROL

The MP section of the TLC Controller provides the control for the peripheral device controller and peripheral device used with an IOM. The MP firmware (microprogram) provides the main elements of control over the peripheral device controller using the Order Structure and the Test Structure logic of the MP. The Order Structure provides the microprogram with the capability to generate individual pulsed signals or level signals at the direction of a decode of a microinstruction. Since these orders are generated by the firmware, they can be used in the peripheral device controller to initiate or regulate device controller operations. Some order signals generated are used internally to furnish control for the MP and SelBUS Interface logic. The Test Structure provides the microprogram with the capability to test preselected individual signals for either a logic High or Low level at the direction of a decode of a microinstruction. Since these tests are requested by the firmware, they can be used to determine the current condition of the peripheral device controller, SelBUS interface, and MP. The Test Structure and the Order Structure provide the firmware with the ability to make a decision based on current conditions within the IOM and peripheral device controller and to take appropriate action based on the decision.

The MP outputs data to the peripheral device controller in either 16- or 32-bit operands. This data is obtained from the MP's 16-bit A-register file and/or the 16-bit B-register file. The outputted data may represent either data to be recorded by the peripheral device or control information (such as device address information) for the device controller. The microprogram has the ability to define the type of data being outputted by using the Order Structure logic to generate output control signals to the device controller in conjuction with the output transfer. These order signals must be predefined when the firmware is generated and the device controller is designed.

The microprogram inputs data from the peripheral device controller in 16-bit operands. The input data is stored in the 16-bit B-register file of the MP. The input data may represent data read from the peripheral device or status information. The microprogram has the ability to request the type of data input to the MP using the Order Structure logic to generate input control signals to the device controller in conjunction with the input transfer. These orders must be predefined at the same time the firmware is generated and the device controller is designed.

3-7 DETAILED THEORY

The detailed theory of the TLC Controller is divided into three topics: the SelBUS interface, the MP, and the Device Controller interface. The detailed discussion describes the logic components of each topic, the primary function of the logic components, and the primary control signals that control a logic component and link one logic component to another.

The detailed discussions provide logic drawing references to locate the logic components on a specific logic drawing sheet. The logic drawing references are intended to orient the user to the overall organization of the logic drawings. The logic drawings for the TLC Controller are shown on Logic Diagram 130-103175. The physical location of logic components on the TLC Controller circuit card and the List of Materials (LM) for the circuit card are shown on Drawing 161-103175. The Microprogram firmware and Device Controller circuits are installed on the circuit card and documented on the 160-103175 drawings.

The drawings for the TLC Controller are provided in the drawings manual and may constitute a part of the Peripheral Device Documentation package. The SYSTEM 32 SERIES Circuit Registration Manual, Publication Number 313-325000, provides all conventions necessary to read the logic drawings contained in the drawings manual.

The logic diagrams for the TLC Controller contain many mnemonics. These mnemonics are defined in this manual in Appendix A for the teletypewriter, Appendix B for the line printer, and Appendix C for the card reader.

3-8 SelBUS INTERFACE

The SelBUS interface of the TLC Controller provides the communication path from the MP to the SelBUS. The SelBUS connects the various modules of the SYSTEMS 32 SERIES Computer as shown in Figure 3-1. In general terms, the SelBUS provides a 32-bit bidirectional data bus and a 24-bit bidirectional address bus for communications between the modules of the computer system. In more specific terms, the SelBUS consists of a number of Signal lines.

1. 32-bit bidirectional data lines that carry data, commands, status, device addresses, or memory addresses.

2. 24-bit bidirectional destination bus (address bus) that carries memory addresses, IOM addresses, peripheral device addresses, or priority interrupt levels.

3. Six Tag lines which define the type of data and addresses carried on the data and destination buses. The Tag lines include the Transfer, Memory, Control 0, Control 1, Read, and Error lines. The Tag lines define the type of transfer in progress.

4. Five Response lines which indicate whether a SelBUS transfer was accepted. The Response lines include the Busy, Retry, Transfer Acknowledge, Ready, and Unsuccessful Memory Transfer lines.

5. Twenty-two Priority lines which control the SelBUS transfer priority scheme.

6. Two Memory Address Echo lines and four Inhibit lines that determine which memory module(s) are busy.

7. Three Priority Interrupt lines that provide the IOM with priority interrupt capabilities.

8. Four miscellaneous lines that include the Master Clock, Stop Clock, I/O Interrupt Inhibit, and I/O Reset signals.

The primary functions of the SelBUS Interface logic can be divided into three main functions as described in the following paragraphs.

The first function of the SelBUS Interface logic is to receive and store bus transfers from the SelBUS if the Interface logic is not busy. An interface condition occurs when a SelBUS transfer is being stored in the interface staging register and a new transfer would overlay that transfer. Under these conditions, the Interface logic rejects the new transfer with a Busy or Retry response signal to the SelBUS and module that initiated the transfer. If the SelBUS interface is not busy, the transfer is stored in the staging register. The Interface logic indicates to the microprocessor that a transfer has been received and generates a Transfer Acknowledge (TA) to the SelBUS. The Interface logic can indicate the reception of a transfer by using the microprocessor Test Structure (for transfers that were expected by the microprogram) or the Microinterrupt logic (for transfers that are unexpected). In either case, the microprogram must examine the transfer and take the appropriate action. When the microprogram is finished with the transfer, it must clear the Busy condition in the SelBUS Interface logic.

The second function of the SelBUS Interface logic is to store output transfers to the SelBUS when the microprogram is assembling the transfer. The Interface logic stores an output transfer in the staging register during assembly. Any input transfers received from the SelBUS must be rejected with either a Busy or Retry signal to prevent the transfer being assembled from being overlaid by the input transfer. Once an output transfer has been assembled in the staging registers, the interface must obtain SelBUS transfer priority and complete the output transfer to the SelBUS.

The third function of the SelBUS Interface logic is to generate priority interrupts (Service Interrupts) at the request of the microprogram. Once an interrupt has been requested by the microprogram, the Interface logic must generate that interrupt if no higher priority interrupt is being generated by another IOM. The microprogram furnishes the priority of the interrupt to be generated when the interrupt is requested. It is the SelBUS Interface logic's responsibility to generate the interrupt during a time frame defined by the CPU through the SelBUS if no higher priority interrupt is in progress.

A SelBUS transfer requires 150 nanoseconds. Four SelBUS transfer cycles occur during every 600-nanosecond memory cycle. The leading and trailing edge of a SelBUS transfer cycle is marked by one full cycle of the Master Clock (LCLK) signal, which is used to

define the transfer cycle. The LCLK signal is also used for timing in the microprocessor so that both microprocessor instruction cycles and SelBUS transfer cycles require 150 nanoseconds and both cycles are always in sync. During output transfers to the SelBUS, the Interface logic drives the output transfer to the bus at the start of the transfer cycle and holds the information on the bus for the duration of the cycle. During an input transfer, the Interface logic receives the transfer for the duration of the transfer cycle; however, the information is not strobed into the staging register until the trailing edge of the transfer cycle. The TA function is always performed in the bus transfer cycle following the input transfer.

3-9 SelBUS TRANSFER CLASSIFICATION

All transfers on the SelBUS can be classified as either input or output with respect to the IOM. All input transfers originate in either the SYSTEMS 32 SERIES CPU or MBC. All output transfers have a destination of either the SYSTEMS 32 SERIES CPU or MBC. There are nine categories of SelBUS transfers. Each category describes the type of information contained in the transfer.

Each type of transfer is identified by the configuration of the SelBUS Tag Signal lines, which include the following signals:

1. The Transfer (LTX) signal, which indicates that the transfer on the SelBUS is valid.

2. The Memory (LMEM) signal, which indicates that the transfer on the SelBUS is destined for memory.

3. The Control 0 (LCNT0) signal, which is one of the signals describing the type of transfer.

4. The Control 1 (LCNT1) signal, which is one of the signals describing the type of transfer.

5. The Read (LRD) signal, which (in memory transfers) indicates a memory read operation or describes the type of transfer in nonmemory transfers.

6. The Error (LERROR) signal, which indicates that a parity error has occurred during a memory read access.

The specific tag line configurations that define the types of SelBUS transfers are listed in Table 3-1. The following paragraphs describe each type of transfer and provide a format for the information contained on the SelBUS during each type of transfer.

3-10 WRITE DATA OR ORDER TRANSFER (WDOT)

The WDOT originates in the SYSTEMS 32 SERIES CPU and has the tag line configuration listed in Table 3-1. Figure 3-2 shows the WDOT format. The WDOT is used to initiate or halt operations within the IOM addressed by destination bus bits 09 through 15. These bits provide the physical controller address. The subaddress field of the destination bus specifies the individual peripheral device to be controlled by the IOM.

The information contained in the 32-bit data bus during WDOT transfers depends on the condition of data bus bits 00, 01, 02, and 03. When data bus bit 00 is equal to One, the WDOT is a start order that causes the IOM to initiate an I/O operation. During a start order, data bus bits 08 through 31 contain the memory address of the IOCD. The IOCD contains the TCW address and the actual I/O operation to be performed. Once the microprogram has the IOCD address, it can read the IOCD from memory and initiate the specified I/O operation.

When data bus bit 01 is equal to One, the WDOT is a load RAM order that indicates that data bus bits 25 through 31 contain the IOM's physical address, and destination bus bits 01 through 07 contain the interrupt priority level of the IOM. The microprogram must recognize this transfer and store the IOM address and the priority level in its reserved B-file register. The load RAM version of the WDOT only occurs when the SYSTEMS 32 SERIES CPU is executing its IPL firmware sequence following a System Reset. This action provides each IOM with its physical address and interrupt priority level.

When data bus bit 02 is equal to One, the WDOT is a Halt order that causes the microprogram to halt all I/O operations in progress. During a Halt order, the data bus bits 03 through 31 are not used.

Data bus bit 03 is used with bit 00 to indicate a Start I/O operation for an IPL sequence. The remaining data bus bits supply the IOCD address for a normal (non-IPL) Start I/O transfer. The specific IPL sequence is a function of the device dependent firmware (microprogram) for the IOM. However, the IPL Start I/O usually causes the addressed peripheral device to read a software bootstrap program and store this program in memory at the locations addressed by the IOCD.

Table 3-1. SelBUS Input/Output Transfer Identifications

Tag Bus Signals						Input Transfer Operation
LTX	LMEM	LCNT0	LCNT1	LRD	LERROR	
L	H	H	H	H	H	Write Data or Order Transfer (WDOT)
L	H	H	L	H	H	Interrupt Control Transfer (ICT)
L	H	H	L	L	H	Read Status Transfer (RSTX)
L	H	L	H	H	H	Advance Read Status Transfer (ARSTX)
L	H	L	H	L	H	Advance Interrupt Control Transfer (AICT)
L	H	L	L	H	H	Data Return Transfer (DRT) (Read data from memory)
H	H	H	H	H	L	Error Transfer (Previous DRT from memory contained a Parity Error)
						Output Transfer Operation
L	L	H	H	H	H	Memory Write Transfer (MWT)
L	L	H	H	L	H	Memory Read Transfer (MRT)
L	H	L	L	H	H	Data Return Transfer (DRT) (Status information to the CPU)

3-11 ADVANCE INTERRUPT CONTROL TRANSFER (AICT)

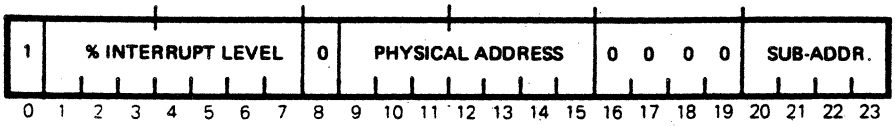
The AICT originates in the SYSTEMS 32 SERIES CPU and has the tag line configuration listed in Table 3-1. Figure 3-3 shows the format of the AICT. The AICT preconditions the microprogram for an interrupt control operation. The AICT can be set by the following actions in the microprogram (as indicated by the data bus bits configurations):

1. Request Interrupt
2. Deactivate Interrupt
3. Activate Interrupt
4. Disable Interrupt
5. Enable Interrupt

The AICT preconditions the microprogram for interrupt control actions and causes the microprogram to perform the setup procedures required before executing the interrupt control action. When the microprogram is ready to execute the interrupt control action, it generates a Ready signal to the computer. The Ready signal is generated by the Order Structure logic of the microprogram and is one of the SelBUS response signals. The Ready signal causes the computer firmware to issue an Interrupt Control Transfer (ICT) to the SelBUS. The ICT causes the microprogram to execute the interrupt control operation. In the CPU, the AICT is issued as a result of one of the software Interrupt Control instructions. Those instructions are listed in the SYSTEMS 32 SERIES Computer Reference Manual applicable to the system being used.

WDOT SelBUS TRANSFER

DESTINATION BUS (FOR ALL FORMATS)



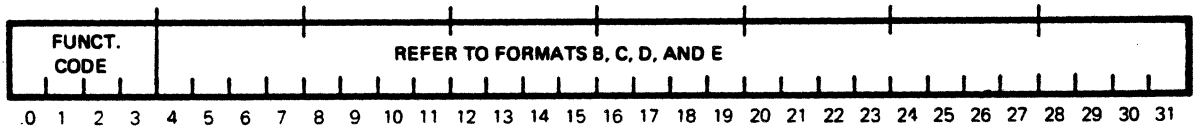
INTERRUPT LEVEL
BITS 01-07 PROVIDE THE ONES COMPLEMENT OF THE I/O CONTROLLER INTERRUPT LEVEL.

PHYSICAL ADDRESS
BITS 09-15 PROVIDE THE PHYSICAL ADDRESS OF THE I/O CONTROLLER.

SUBADDRESS
BITS 20-23 PROVIDE THE I/O CONTROLLER DEVICE SUBADDRESS

FORMAT A

DATA BUS FOR WDOT TRANSFER (GENERAL FORMAT)



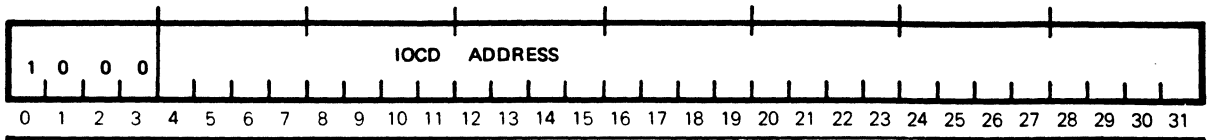
FUNCTION CODE BITS				
00	01	02	03	DEFINITION
1	0	0	0	START I/O. REFER TO FORMAT B FOR THE CONTENTS OF BITS 04-31.
0	1	0	0	LOAD RAM. REFER TO FORMAT C FOR THE CONTENTS OF BITS 04-31.
0	0	1	0	HALT I/O. SEE NOTE 2.
1	0	0	1	INITIAL PROGRAM LOAD AND START I/O. REFER TO FORMAT E FOR THE CONTENTS OF BITS 04-31.

Figure 3-2. Write Data or Order Transfer (WDOT) Format (Sheet 1 of 3)

WDOT SelBUS TRANSFER

FORMAT B

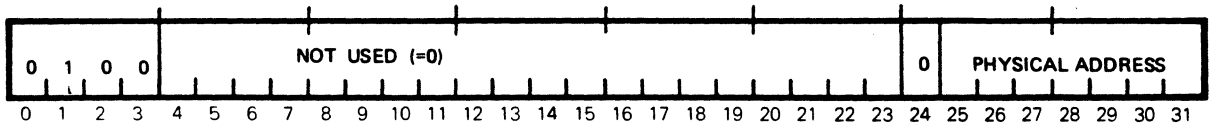
DATA BUS FOR WDOT - START I/O



IOCD ADDRESS
BITS 04-31 PROVIDE THE MEMORY ADDRESS OF THE INPUT/OUTPUT COMMAND DOUBLEWORD (IOCD).

FORMAT C

DATA BUS FOR WDOT - LOAD RAM



PHYSICAL ADDRESS
BITS 25-31 PROVIDE THE PHYSICAL ADDRESS OF THE I/O CONTROLLER.

FORMAT D

DATA BUS FOR WDOT - HALT I/O

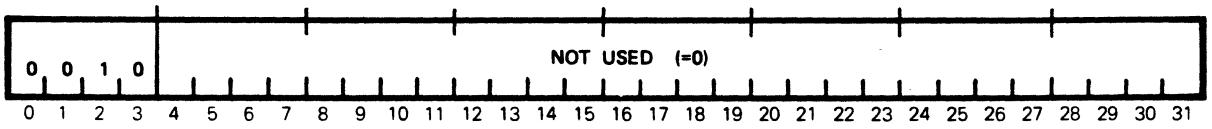


Figure 3-2. Write Data or Order Transfer (WDOT) Format (Sheet 2 of 3)

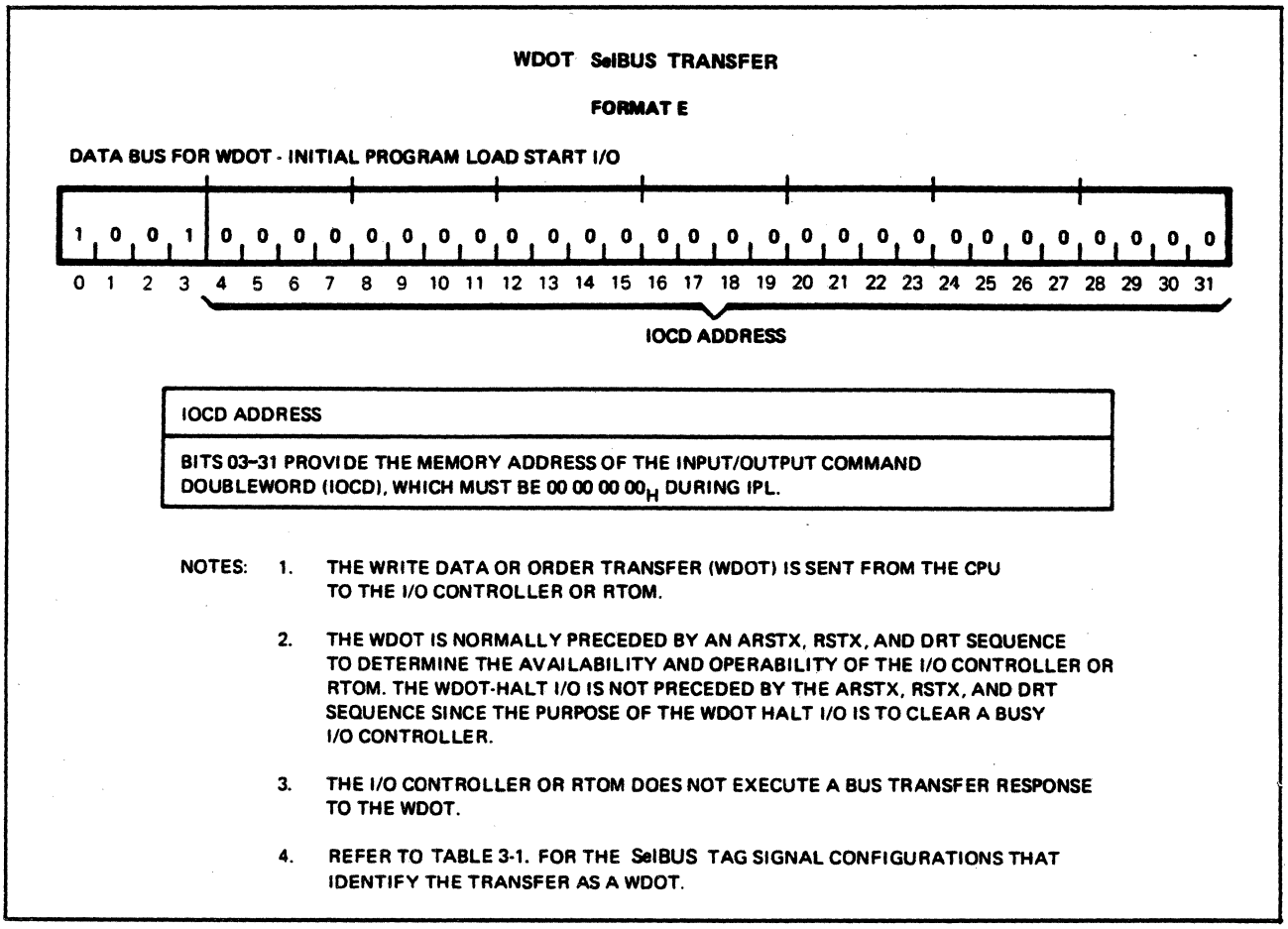


Figure 3-2. Write Data or Order Transfer (WDOT) Format (Sheet 3 of 3)

3-12 INTERRUPT CONTROL TRANSFER (ICT)

The ICT originates in the SYSTEMS 32 SERIES CPU and is used with the AICT. The tag line configuration for the ICT is listed in Table 3-1. The data bus and destination bus formats for the two transfers are identical and are shown in Figure 3-3. The specific functions of the ICT are provided in the AICT discussion, which occurred earlier in this text.

The ICT is the second and last transfer of an AICT and ICT pair that is used to control the Interrupt logic of the SelBUS interface and microprogram. The AICT preconditions the microprogram, which responds with a Ready signal to the CPU when the microprogram is ready to execute the interrupt control. The CPU responds to the Ready signal by generating an ICT to the IOM. This action causes the microprogram to execute the interrupt control operation and generate a DRT to the CPU. This DRT is used to acknowledge the ICT. The contents of the DRT are not significant to the CPU.

In actual operation, the CPU may not immediately respond to the Ready signal with the ICT. If the microprogram detects more than nine SelBUS transfer cycles (microinstruction cycles) following the generation of the Ready signal, and the ICT has not been received from the CPU, the microprogram can assume that the CPU was interrupted by some other device and that the original AICT will be reissued later. If the ICT is received within nine bus cycles following the Ready signal, the microprogram must wait for 18 bus cycles before generating the DRT to the CPU. The 18 cycles must be timed from the point at which execution of the ICT is completed. The 18 bus cycles can also be timed by testing the Interrupt Sync flip-flop, using the microprocessor Test Structure. The microprogram must ensure that the Sync flip-flop is set twice to time the 18 bus cycles. The contents of the DRT sent to the CPU are not used by the CPU and are therefore insignificant.

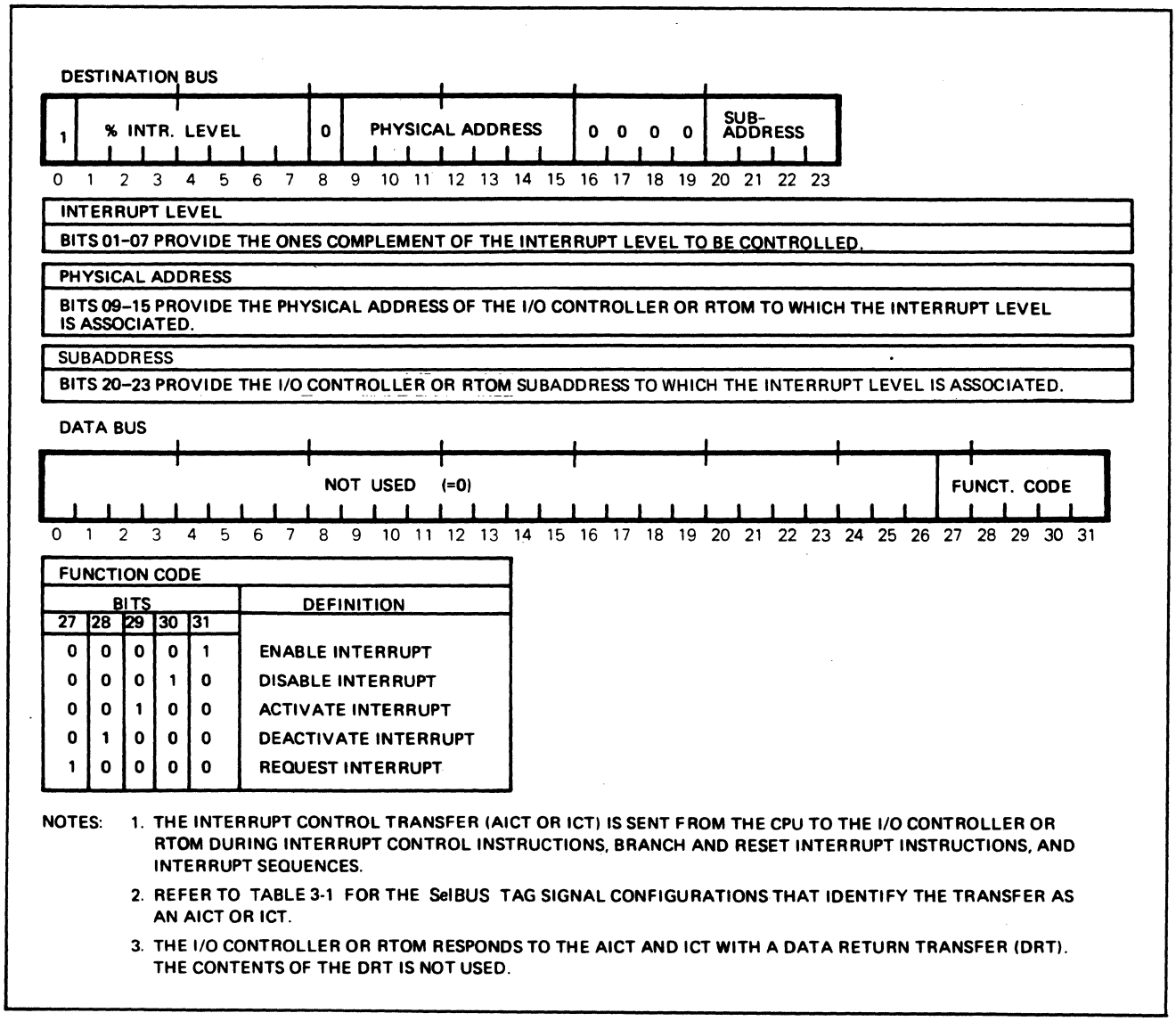


Figure 3-3. Advance Interrupt Control Transfer (AICT) and Interrupt Control Transfer (ICT) Formats

3-13 ADVANCE READ STATUS TRANSFER (ARSTX)

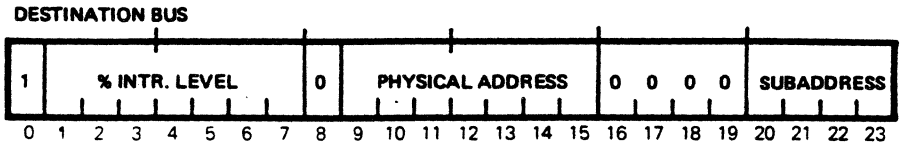
The ARSTX originates in the SYSTEMS 32 SERIES CPU and has the tag line configuration listed in Table 3-1. Figure 3-4 shows the ARSTX format. The ARSTX preconditions the microprogram for a status transfer to the CPU and causes the microprogram to assemble the requested status. The ARSTX can request the following types of status transfers according to the data bus bit configurations:

1. Acknowledge Interrupt
2. Subaddress and Controller Status
3. Device Status
4. Transfer Current Word Address (TCA)

When the microprogram has the requested status assembled and ready for transfer to the CPU, the microprogram issues a Ready signal to the CPU using the SelBUS. The Ready signal is generated by the Microprocessor Order Structure. The CPU responds to the Ready signal by generating an RSTX to the IOM. The RSTX causes the microprogram to transfer the assembled status to the CPU in a DRT on the SelBUS. The RSTX and DRT transfers are discussed later in this text.

The Acknowledge Interrupt function of the ARSTX/RSTX is generated by the CPU firmware in response to an SI that was generated by the IOM and sent to the CPU. The Acknowledge interrupt causes the IOM microprogram to set the IOM interrupt level active. This action

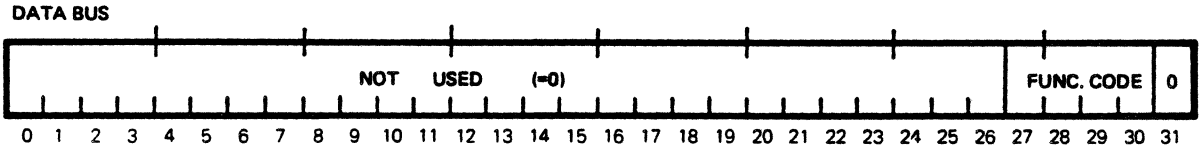
ARSTX/RSTX SelBUS TRANSFER



INTERRUPT LEVEL
BITS 01-07 PROVIDE THE ONES COMPLEMENT OF THE I/O CONTROLLER INTERRUPT LEVEL

PHYSICAL ADDRESS
BITS 09-15 PROVIDE THE PHYSICAL ADDRESS OF THE I/O CONTROLLER

SUBADDRESS
BITS 20-23 PROVIDE THE I/O CONTROLLER DEVICE SUBADDRESS



FUNCTION CODE				
BITS				DEFINITION
27	28	29	30	
0	0	0	1	TRANSFER CURRENT WORD ADDRESS (TCA)
0	0	1	0	DEVICE STATUS (TD2000 STATUS)
0	1	0	0	CONTROLLER STATUS (TD8000 AND 4000 STATUS)
1	0	0	0	ACKNOWLEDGE INTERRUPT (ACTIVATE INTERRUPT LEVEL)

NOTES:
1. THE READ STATUS TRANSFER (ARSTX OR RSTX) IS SENT FROM THE CPU TO THE I/O CONTROLLER DURING ANY CPU-I/O COMMUNICATION SEQUENCE.
2. THE I/O CONTROLLER RESPONDS TO THE ARSTX AND RSTX SEQUENCE WITH A DRT THAT CONTAINS THE REQUESTED STATUS. REFER TO FIGURE 3-5. FOR THE VARIOUS DRT FORMATS.
3. REFER TO TABLE 3-1 FOR THE SelBUS TAG SIGNAL CONFIGURATIONS THAT IDENTIFY THE TRANSFER AS A RSTX OR RSTX.

Figure 3-4. Advance Read Status Transfer (ARSTX) and Read Status Transfer (RSTX) Formats

inhibits all lower priority interrupts in the computer system. The IOM interrupt level remains active until a Deactivate Interrupt AICT/ICT signal is received from the CPU. The CPU firmware generates the Deactivate Interrupt AICT/ICT as a result of the firmware sequence of the software Deactivate Interrupt, Interrupt Control instruction, or a BRI instruction. The contents of the DRT returned to the CPU in response to the Acknowledge Interrupt ARSTX/RSTX are not used and are therefore insignificant.

The Subaddress and Controller Status function of the ARSTX/RSTX are generated by the CPU firmware as part of the sequence for the software TD 8000 and 4000 instructions or as part of the CD instruction. The DRT returned to the CPU, in response to the request for Subaddress and Controller Status, contains the condition codes for both TD 8000 and 4000 TD instructions and a bit to indicate when the microprogram is busy executing a CD (WDOT Start I/O) instruction. The TD 8000 and 4000 condition codes are defined in Section II of this manual.

The Device Status function of the ARSTX/RSTX is generated by CPU firmware as part of the sequence for the software TD 2000 instruction. The DRT returned to the CPU (in response to the Device Status ARSTX/RSTX) contains a 16-bit status halfword, the TD 2000 condition codes, and a bit to indicate when the microprogram is busy executing a CD (WDOT Start I/O) instruction. The 16-bit status halfword is stored in memory by the CPU firmware at the location addressed by the TCW for the IOM. The definitions of the status halfword and the TD 2000 condition codes are device dependent. There are no status halfwords for a teletype or card reader; all zeros are returned to the CPU in response to a TD 2000 to these devices.

The TCA function of the ARSTX/RSTX is generated by CPU firmware as part of the sequence for the software command device Transfer Current Word Address instruction. The DRT returned to the CPU (in response to the TCA ARSTX/RSTX) contains the current contents of the TCW. The TCW is stored in the IOM's TCW dedicated memory location by the CPU firmware at the end of the command device TCA instruction.

3-14 READ STATUS TRANSFER (RSTX)

The RSTX originates in the SYSTEMS 32 SERIES CPU and is used with the ARSTX. The tag line configuration of the RSTX is listed in Table 3-1. The data bus and destination bus for the two types of transfers are identical and are shown in Figure 3-4. The specific functions of the RSTX are provided in the ARSTX discussion in this text.

The RSTX is the second and last transfer of an ARSTX/RSTX pair used to request a status transfer from the IOM. The ARSTX causes the microprogram to assemble the request status. When the assembly is complete, the microprogram generates a Ready signal to the CPU. The CPU responds to the Ready signal by generating an RSTX to the IOM. This action causes the microprogram to transfer the requested status to the CPU by the SelBUS in a DRT.

In actual operation, the CPU may not immediately respond to the Ready signal with the RSTX. If the microprogram detects more than nine SelBUS transfer cycles (microinstruction cycles) following the generation of the Ready signal, and the RSTX has not been received from the CPU, the microprogram assumes that the CPU was interrupted by some other device and that the original ARSTX will be reissued later. If the RSTX is received within nine bus cycles following the Ready signal, the microprogram responds with the DRT containing the requested status.

3-15 DATA RETURN TRANSFER (DRT)

The DRT can be originated by either the IOM or the MBC and has the tag line configuration listed in Table 3-1. Figure 3-5 illustrates the formats that the DRT data bus and destination bus can have.

When the DRT originates in the IOM, its destination is the CPU. The DRT contains status that was requested by a CPU generated AICT/ICT or ARSTX/RSTX transfer pair. When the DRT is generated by the IOM in response to AICT/ICT or an Interrupt Acknowledge ARSTX/RSTX, the data bus bits of the DRT are not used and are insignificant. Here the DRT is only used by the CPU firmware for the IOM to acknowledge the reception of the ICT or RSTX. A DRT from the IOM to the CPU is identified by the true condition of the destination bus bit 08, which specifies the CPU as the destination address of the SelBUS transfer.

When the DRT originates in memory, its destination is the IOM. The DRT contains a data read from memory at the request of an MRT. A DRT that originates in memory can be identified by the IOM physical address of the destination bus bits 09 through 15 and the false condition of CPU address bit 08.

Figure 3-5 shows the data bus formats for the DRT transfers. Note that the DRT format can vary for the variations of the ARSTX/RSTX

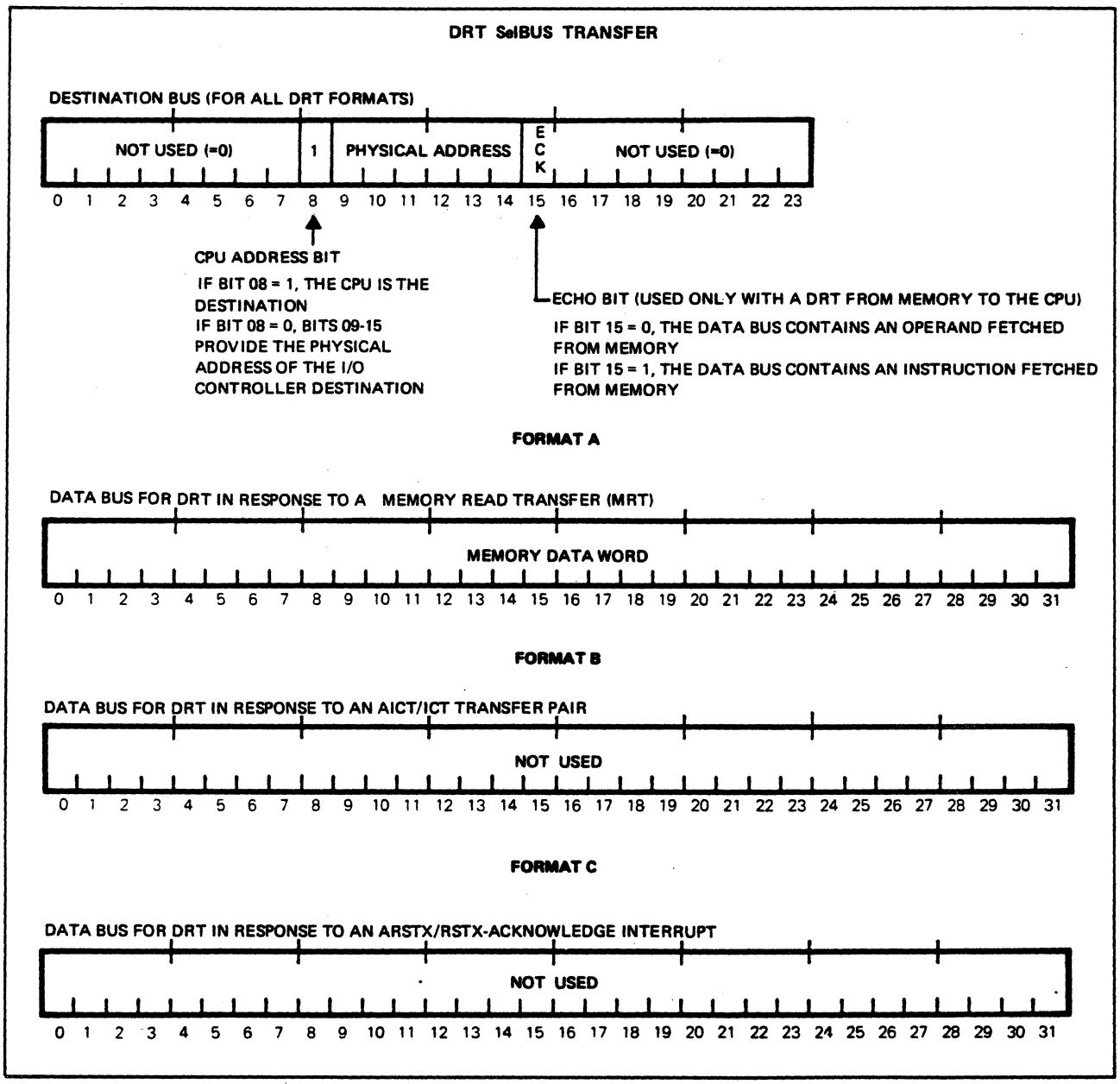


Figure 3-5. Data Return Transfer (DRT) Format (Sheet 1 of 2)

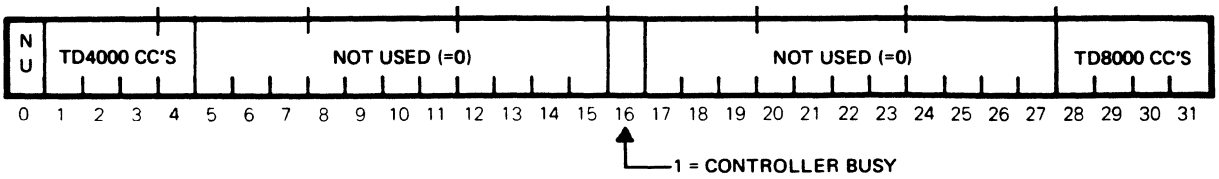
transfer pair. Each format is illustrated. The specific condition code definitions for the 8000 and 4000 instructions (ARSTX/RSTX Subaddress and Controller Status) are provided in Section II of this manual. The condition codes and status halfword definitions for the TD 2000 instruction (ARSTX/RSTX Device Status) are device dependent and are defined in the specific technical manual for the peripheral device used with the IOM. The destination bus format for all types of DRTs remains the same except that the CPU address bit is used by the DRTs generated in the IOM, and the IOM physical address bits are used in the DRTs generated by the memory (MBC).

3-16 ERROR TRANSFER (ET)

The ET originates in memory and consists only of the Error tag line. The SelBUS configuration for the Error transfer is shown in Figure 3-6 with the data bus and destination bus formats. The ET is a single tag line transfer. The remaining parts of the SelBUS are available to any other type of transfer.

DRT SelBUS TRANSFER
FORMAT D

DATA BUS FOR TLC CONTROLLER RESPONSE TO AN ARSTX/RSTX CONTROLLER STATUS REQUEST (TD8000 AND TD4000 STATUS)

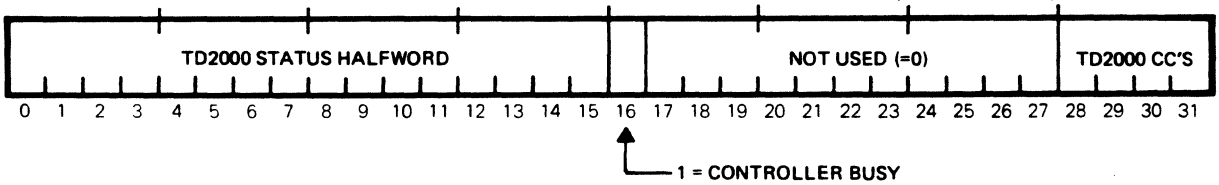


TD4000 CONDITION CODES
BITS 01, 02, 03 AND 04 PROVIDE THE TD4000 CONDITION CODES 1, 2, 3, AND 4, RESPECTIVELY. REFER TO SECTION II FOR CONDITION CODE BIT DEFINITION.

TD8000 CONDITION CODES
BITS 28, 29, 30, AND 31 PROVIDE THE TD8000 CONDITION CODES 1, 2, 3, AND 4, RESPECTIVELY. REFER TO SECTION II FOR CONDITION CODE BIT DEFINITION.

FORMAT E

DATA BUS FOR TLC CONTROLLER RESPONSE TO AN ARSTX/RSTX DEVICE STATUS REQUEST (TD2000 STATUS)

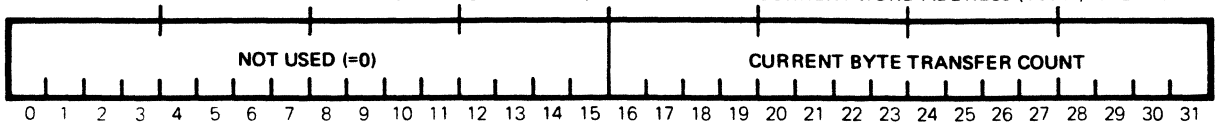


TD2000 STATUS HALFWORD
BITS 00-15 PROVIDE THE TD2000 STATUS HALFWORD. REFER TO SECTION II FOR CONDITION CODE BIT DEFINITION.

TD2000 CONDITION CODES
BITS 28, 29, 30, AND 31 PROVIDE THE TD2000 CONDITION CODES 1, 2, 3, AND 4 RESPECTIVELY. REFER TO SECTION II FOR CONDITION CODE BIT DEFINITION.

DRT SelBUS TRANSFER
FORMAT F

DATA BUS FOR TLC CONTROLLER RESPONSE TO AN ARSTX/RSTX TRANSFER CURRENT WORD ADDRESS (TCWA) REQUEST



CURRENT BYTE TRANSFER COUNT
BITS 16-31 PROVIDE THE CURRENT BYTE TRANSFER COUNT.

Figure 3-5. Data Return Transfer (DRT) Format (Sheet 2 of 2)

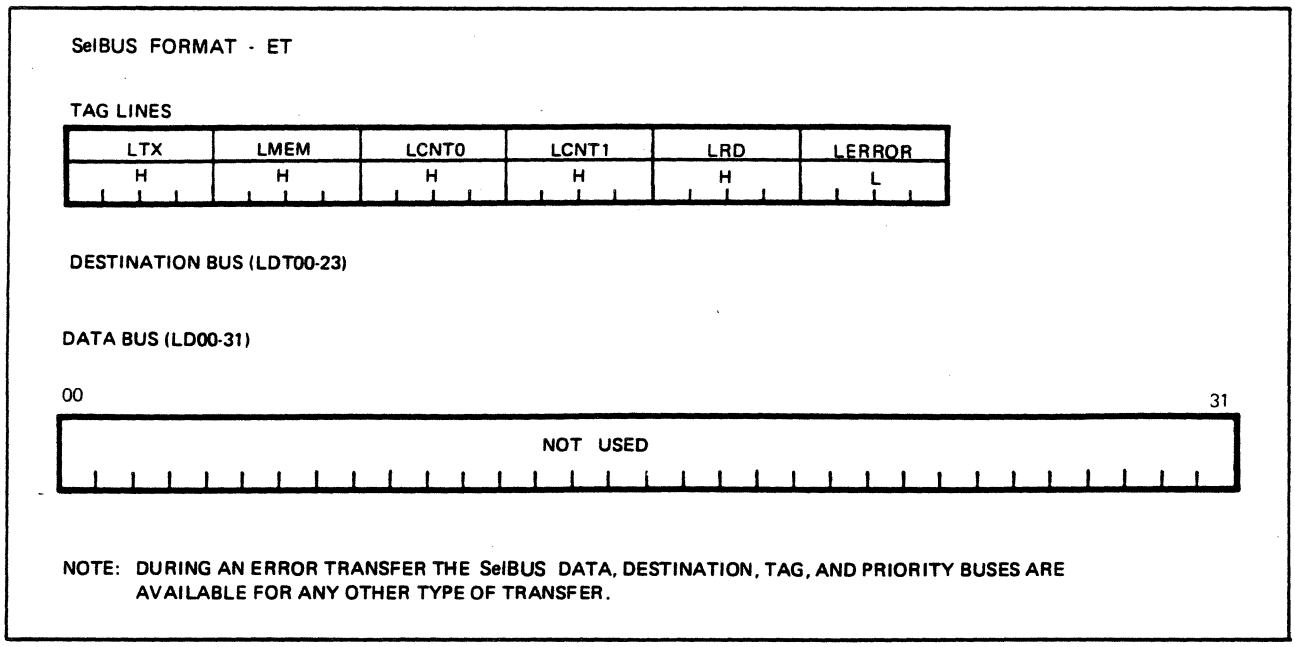


Figure 3-6. Error Transfer (ET) Format

3-17 MEMORY WRITE TRANSFER (MWT)

The MWT originates in the IOM. Its destination is the memory that is connected to the SelBUS. The tag line configuration is listed in Table 3-1. Figure 3-7 shows the MWT format.

The destination of the MWT is indicated by the true condition of the tag line Memory (LMEM) signal. The false condition of the tag line Read (LRD) signal indicates that the MWT is a memory write. In the MWT, the destination bus contains the memory address, and the data bus contains the data (32 bits) to be stored at the memory address. During the SelBUS transfer cycle following an MWT, the SelBUS Interface logic must monitor the SelBUS Transfer Acknowledge (LTA) signal line. If the LTA signal is not received, the MWT is assumed to have addressed nonpresent memory. In a nonpresent memory condition, the SelBUS Interface logic must notify the microprogram of the error condition so that the microprogram can terminate the I/O operation and generate a priority (Service) interrupt to the computer.

3-18 MEMORY READ TRANSFER (MRT)

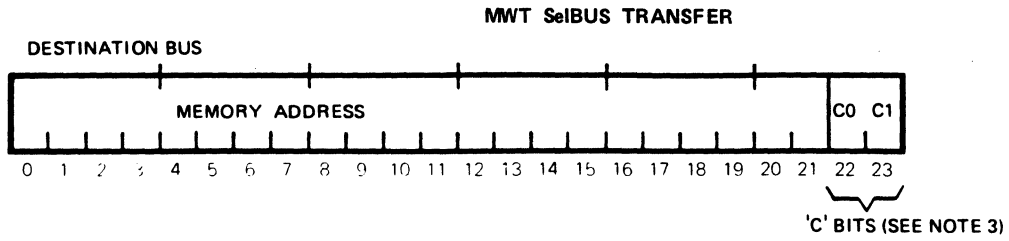
The MRT originates in the IOM. Its destination is the memory which is connected to the SelBUS. The tag line configuration is listed in Table 3-1. Figure 3-8 shows the MRT format.

The destination of the MRT is indicated by a true condition of the tag line Memory (LMEM) signal. A true condition of the tag line Read (LRD) signal indicates that the MRT is a memory read. In the MRT, the destination bus contains the memory address to be read, and the data bus contains the IOM physical address in bits 24 through 31. When the memory has read the location specified by the memory address, it generates a DRT to the IOM specified by the physical address provided in the MRT. The DRT contains the data read from memory. If an error occurs during the memory read operation, the memory generates an ET following the DRT. The ET indicates that a parity error occurred during the read instruction.

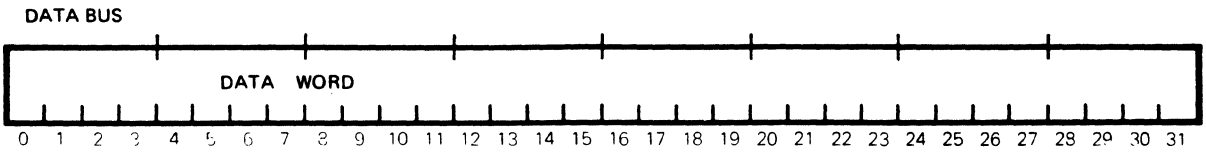
During the SelBUS transfer cycle following the MRT, the SelBUS Interface logic must monitor the SelBUS LTA signal lines. If the LTA signal is not received, the MRT is assumed to have addressed nonpresent memory. In a nonpresent memory condition, the SelBUS interface must notify the microprogram of the error condition so that the microprogram can terminate the I/O operation and generate a priority (Service) interrupt to the computer.

The microprogram and SelBUS interface must always expect a DRT to be returned from memory following an MRT.

A complete detailed analysis of the SelBUS interface may be found in the IOM Technical Manual, Publication Number 325-329000.



MEMORY ADDRESS
BITS 00-23 PROVIDE THE MEMORY ADDRESS FOR THE WRITE MEMORY FUNCTION.



DATA WORD
BITS 00-31 PROVIDE THE DATA WORD TO BE STORED AT THE ADDRESS SPECIFIED BY THE DESTINATION BUS.

NOTES:

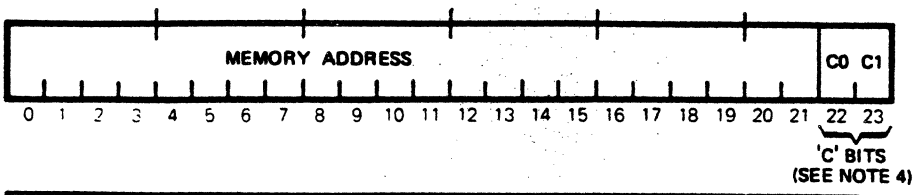
1. THE MEMORY WRITE TRANSFER (MWT) CAN BE ORIGINATED BY EITHER AN I/O CONTROLLER OR THE CPU. THE DESTINATION OF THE TRANSFER IS THE MEMORY BUS CONTROLLER.
2. REFER TO TABLE 3-1 FOR THE SelBUS TAG SIGNAL CONFIGURATIONS THAT IDENTIFY A TRANSFER AS AN MRT.
3. THE MWT CAN BE USED TO SPECIFY A WORD, HALFWORD, OR BYTE WRITE FUNCTION. IN ANY OF THESE CASES, THE DATA TO BE STORED IN MEMORY MUST BE RIGHT JUSTIFIED ON THE DATA BUS, AND THE SelBUS TAG BUS SIGNAL 'LDTF' ('F' BIT) AND THE DESTINATION BUS 'C' BITS ARE USED AS FOLLOWS:

F BITS (LDTF SIGNAL)	C BITS DESTINATION BUS BITS		TRANSFER FUNCTION
	22	23	
0 (high)	0	0	WORD TRANSFER
0 (high)	0	1	HALFWORD TRANSFER (LEFT HALFWORD)
0 (high)	1	1	HALFWORD TRANSFER (RIGHT HALFWORD)
1 (low)	0	0	BYTE TRANSFER TO BYTE 0 (BITS 00-07)
1 (low)	0	1	BYTE TRANSFER TO BYTE 1 (BITS 08-15)
1 (low)	1	0	BYTE TRANSFER TO BYTE 2 (BITS 16-23)
1 (low)	1	1	BYTE TRANSFER TO BYTE 3 (BITS 24-31)

Figure 3-7. Memory Write Transfer (MWT) Format

MRT SelBUS TRANSFER

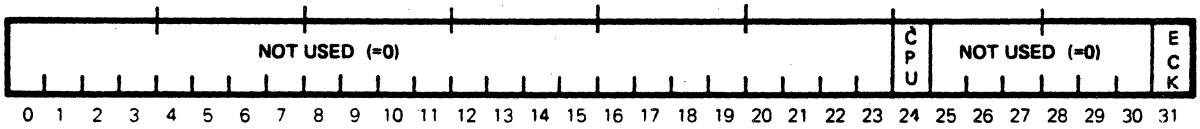
DESTINATION BUS (FOR ALL FORMATS)



MEMORY ADDRESS
BITS 00–23 PROVIDE THE MEMORY ADDRESS OF THE LOCATION TO BE READ.

FORMAT A

DATA BUS FOR A CPU ORIGINATED MRT

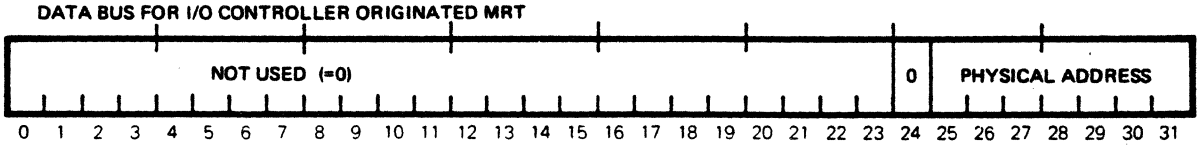


CPU
BIT 24 EQUAL TO ONE SPECIFIES THAT THE CPU IS THE SOURCE OF THE MRT AND THAT THE DATA READ FROM THE ADDRESSED LOCATION IS TO BE RETURNED TO THE CPU.

ECHO
BIT 31 EQUAL TO ZERO, SPECIFIES THAT AN OPERAND IS TO BE READ FROM MEMORY; BIT 31 EQUAL TO ONE SPECIFIES THAT AN INSTRUCTION IS TO BE READ FROM MEMORY.

FORMAT B

DATA BUS FOR I/O CONTROLLER ORIGINATED MRT



PHYSICAL ADDRESS
BITS 25–31 PROVIDE THE PHYSICAL ADDRESS OF THE I/O CONTROLLER THAT ORIGINATED THE MRT AND SPECIFIES THAT THE DATA READ FROM THE ADDRESSED MEMORY LOCATION IS TO BE SENT TO THIS I/O CONTROLLER ADDRESS.

Figure 3-8. Memory Read Transfer (MRT) Format (Sheet 1 of 2)

NOTES:
1. THE MEMORY READ TRANSFER (MRT) CAN BE ORIGINATED BY EITHER AN I/O CONTROLLER OR THE CPU. THE DESTINATION OF THE TRANSFER IS THE MEMORY BUS CONTROLLER (MBC).
2. THE MEMORY BUS CONTROLLER RESPONDS TO THE MRT WITH A DATA RETURN TRANSFER (DRT) CONTAINING THE CONTENTS OF THE MEMORY LOCATION ADDRESSED BY THE MRT. THE DESTINATION OF THE DRT IS THE MODULE THAT ORIGINATED THE MRT.
3. REFER TO TABLE 3-1 FOR THE SelBUS TAG SIGNAL CONFIGURATIONS THAT IDENTIFY A TRANSFER AS A MRT.
4. THE MRT CAN BE USED TO SPECIFY A WORD, HALFWORD, OR BYTE READ FUNCTION. IN ANY OF THESE CASES, THE DATA READ FROM MEMORY IS RETURNED ON THE DATA BUS, RIGHT JUSTIFIED IN A DRT TRANSFER. THE 'F' BIT (THE TAG BUS LDTF SIGNAL) AND THE DESTINATION BUS 'C' BITS ARE USED TO SPECIFY ANY OF THESE MODES AS FOLLOWS:

F BIT (LDTF SIGNAL)	C BITS DESTINATION BUS BITS		TRANSFER FUNCTION
	22	23	
0 (HIGH)	0	0	WORD TRANSFER
0 (HIGH)	0	1	HALFWORD TRANSFER (LEFT HALFWORD)
0 (HIGH)	1	1	HALFWORD TRANSFER (RIGHT HALFWORD)
1 (LOW)	0	0	BYTE TRANSFER TO BYTE 0 (BITS 00-07)
1 (LOW)	0	1	BYTE TRANSFER TO BYTE 1 (BITS 08-15)
1 (LOW)	1	0	BYTE TRANSFER TO BYTE 2 (BITS 16-23)
1 (LOW)	1	1	BYTE TRANSFER TO BYTE 3 (BITS 24-31)

Figure 3-8. Memory Read Transfer (MRT) Format (Sheet 2 of 2)

3-19 MICROPROGRAMMABLE PROCESSOR (MP)

The MP provides the primary control for the SelBUS Interface and device controller logic contained on the TLC Controller circuit card. In actual operation, it is the microprogram firmware that controls the TLC Controller by commanding various portions of functional logic in the MP. The functional logic of the MP then generates the control signals to the SelBUS interface or device controller as commanded by the firmware. The firmware primary control can be broken down into several areas which provide the basis of organization of the firmware microprogram. The specific microinstructions needed to control these areas can vary from application to application; however, the following list includes the areas that must be controlled by the firmware:

1. The firmware must recognize and acknowledge input transfers from the SelBUS to the SelBUS Interface logic of the TLC Controller.
2. The firmware must recognize and decode the input SelBUS transfer that has occurred.
3. If the input transfer requested Controller or Peripheral Device status, the firmware must accumulate the status and format it into an output transfer to the SelBUS and CPU. The firmware provides direct control for the SelBUS output transfer.

4. If the input transfer indicates an interrupt control function, the firmware must condition its own internal logic and the interrupt logic of the SelBUS interface as commanded by the input transfer.

5. If the input transfer specified a Start I/O operation, the firmware must obtain the IOCD from the memory by generating a Memory Read output transfer to the SelBUS. The MRT must contain the memory address of the IOCD, which the firmware obtained from the Start I/O transfer. When the memory performs the requested read operation, it generates a DRT containing the first word of the IOCD to SelBUS Interface logic of the TLC Controller. After the first word of the IOCD has been obtained, the firmware must increment the IOCD memory address to the next sequential memory address and generate a second MRT to obtain the second word of the IOCD.

6. The firmware is responsible for clearing the SelBUS Interface logic after every input transfer from the SelBUS.

7. In a Start I/O operation, the firmware must initiate the I/O operation indicated by the obtained IOCD to the peripheral device indicated by the Start I/O transfer.

8. In a data transfer I/O operation, the firmware must determine the direction of the data transfer from the IOCD. If the data transfer is a Write to the peripheral device, the firmware must obtain the data from memory by a SelBUS MRT. The memory address for the MRT is obtained from the IOCD. After data has been returned from the memory by a SelBUS DRT, the firmware can transmit the data (up to 32 bits per transfer) to the device controller as requested.

9. If the I/O operation is an RDT from the peripheral device, the firmware must receive the data from the peripheral device controller and format the data into a Memory transfer. The firmware generates an MWT to the memory using the SelBUS. The MWT contains the memory address obtained from the IOCD.

10. After a Memory transfer has been made, the firmware must decrement the transfer count of the IOCD and increment the memory address in preparation for the next Memory transfer.

11. When the transfer count of the IOCD has been decremented to zero, the firmware must determine whether the I/O operation is complete by examining the Flag byte of the IOCD. If no flags are set, the I/O operation is completed. The firmware must issue an SI to the computer if the IOM's interrupt level is enabled. If a Continuation flag in the Flag byte is set, the firmware must obtain a new IOCD from memory by incrementing the IOCD memory address received during the first Start I/O transfer. The updated IOCD address is used for an MRT to the memory using the SelBUS. After the new IOCD has been obtained, the I/O operation is continued.

A complete analysis of the MP may be found in the IOM Technical Manual, Publication Number 325-329000.

3-20 PERIPHERAL DEVICE CONTROLLER INTERFACE

The Peripheral Device Controller interface and the IOM are on the same circuit card and make up the TLC Controller. The interface consists of the following signals:

1. The A-Register file output bits that are sent to the peripheral device controller.

2. The B-Register File Output bits that are sent to the peripheral device controller.

3. The Level Order signals and the Pulse Order signals from the Order Structure to the peripheral device controller.

4. The External Input Signal lines from the peripheral device controller to the B-Register file input multiplexer.

5. The Test Input signals from the peripheral device controller to the Test Structure.

6. The Clock signals and Reset signals that are used to synchronize the operation of the peripheral device controller to the microprocessor. These Clock and Reset signals are generated by the microprocessor and are sent to the peripheral device controller.

Tables 3-2 and 3-3 list the pins and signals for the peripheral device connectors P1A and P1C.

3-21 TELETYPEWRITER (TTY) DEVICE DEPENDENT INTERFACE CONTROLLER

The TTY Device Dependent Interface Controller provides the circuitry necessary to transmit

or receive data between the IOM and TTY peripheral device. The Universal Asynchronous Receiver/Transmitter (UART), the heart of the TTY Device Dependent Interface Controller, performs the greatest portion of the operation. Detailed operation of the large scale integrated circuit is discussed in the following paragraph. A block diagram of the TTY Device Dependent Interface Controller is shown in Figure 3-9.

Table 3-2. P1A Connector Pin and Signal List

Pin	Signal	Pin	Signal
1	HDATAIN	26	GND
2	HDATAIN12V	27	GND
3		28	LDPCONNECTED
4		29	HDPDMDLN
5	LDATAOUT	30	GND
6	HDATAOUT12V	31	HDPLPSTRB
7	GND	32	GND
8	LEIADATI	33	HPAPERINSTOP
9	GND	34	GND
10	LEIADATX	35	GND
11		36	GND
12	GND	37	HLPDATA7
13	HLP+5V	38	GND
14	GND	39	HLPDATA6
15	HDPRESET	40	GND
16	GND	41	HLPDATA5
17	HDPPAPERMOTN	42	GND
18	GND	43	HLPDATA4
19	HDPTOPOFFORM	44	GND
20	GND	45	HLPDATA3
21	HDPBOTOFFORM	46	GND
22	GND	47	HLPDATA2
23	HDPREADY	48	GND
24	GND	49	HLPDATA1
25	HDPONLINE	50	GND

Table 3-3. P1C Connector Pin and Signal List

Pin	Signal	Pin	Signal
1	HPC	26	GND
2	GND	27	HBSY
3	HD12	28	GND
4	GND	29	HIM
5	HD11	30	GND
6	GND	31	HRDY
7	HD0	32	GND
8	GND	33	HHCK
9	HD1	34	GND
10	GND	35	HERROR
11	HD2	36	GND
12	GND	37	HMOCK
13	HD3	38	GND
14	GND	39	GND
15	HD4	40	
16	GND	41	
17	HD5	42	
18	GND	43	
19	HD6	44	
20	GND	45	
21	HD7	46	
22	GND	47	
23	HD8	48	
24	GND	49	
25	HD9	50	

3-22 UNIVERSAL ASYNCHRONOUS RECEIVER/ TRANSMITTER (UART)

The UART is a large scale integrated subsystem which accepts binary characters from either a terminal device or a computer and receives/transmits this character with appended control and error detecting bits. All characters contain start bits 5-8, data bits, 1-2 stop bits, and an odd/even or no parity bit. The UART, baud rate, bits per word, parity mode, and number of stop bits are externally selectable.

Figure 3-10 is a block diagram of the UART circuitry. Table 3-4 defines the input/output pins and signals.

3-23 Transmitter Operation

For proper transmitter operation after power is applied, the External Reset must be enabled and a Clock Pulse having a frequency of 16 times the desired baud rate must be applied. The above conditions set the Transmitter Buffer Empty (TBMT), End of Character (EOC), and Serial Output (SO) to a logic One (Line is marking).

After initializing is completed, the user may set the control and data bits. Control bit selection normally occurs before data bit selection; however, both the Data Strobe (DS) and Control Strobe (CS) may be set simultaneously if minimum pulse width specifications are followed. Once the DS is pulsed, the TBMT signal changes from a logic One to a logic Zero indicating that the data bits holding register is filled with a character and is unable to receive new data bits, and the transmitter shift register is transmitting previously loaded data. The TBMT signal then returns to a logic One. When the transmitter shift register is empty, data bits in the holding register are immediately loaded into the transmitter shift register for transmission. The shifting of information from the holding register to the transmitter shift register is followed by the SO and EOC going to a logic Zero. The TBMT also goes to a logic One indicating that the shifting operation is completed and that the data bits holding register is ready to accept new data. Remember that one full character time is now available for loading the next character without loss in transmission speed due to double buffering (separate data bits in the holding register and in the transmitter shift register).

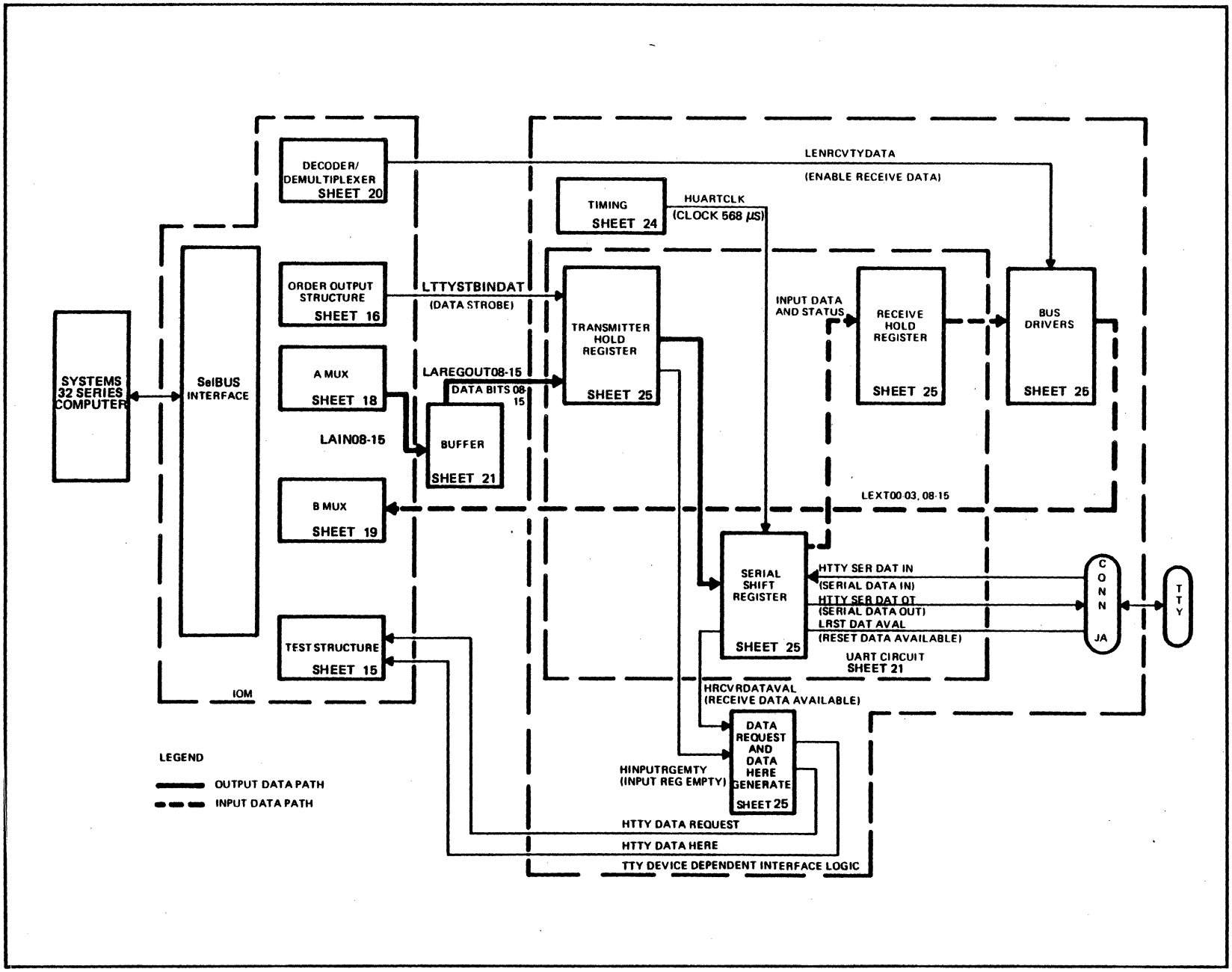


Figure 3-9. TTY Device Dependent Interface Controller

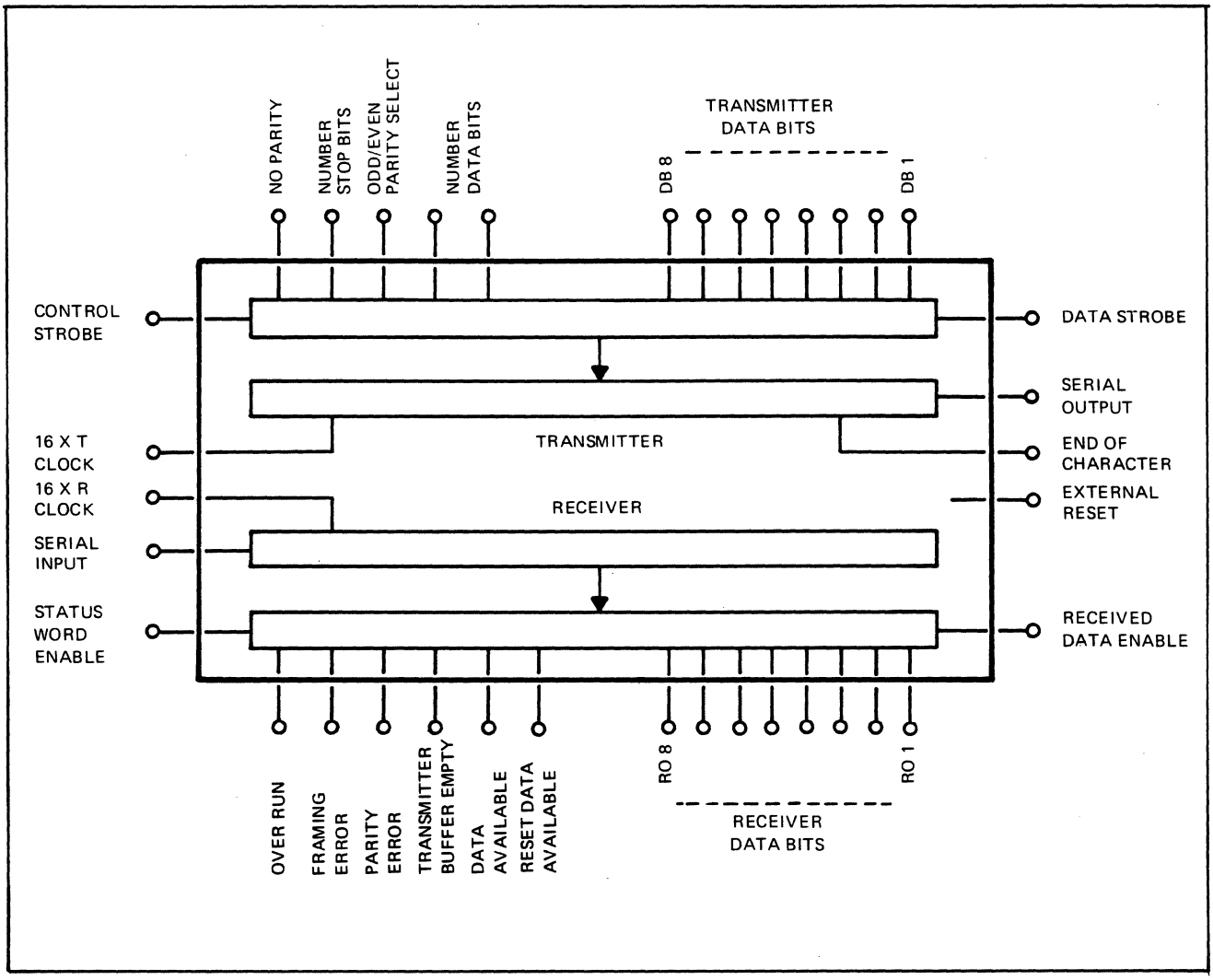


Figure 3-10. Block Diagram - Universal Asynchronous Receiver/Transmitter (UART)

Data transmission is initiated with transmission of a start bit, data bits, parity bit (if desired), and stop bit(s). When the last stop bit has been on-line for one bit time, EOC goes to a logic One indicating that a new character is ready for transmission. This new character is transmitted only if TBMT is a logic Zero as previously discussed (refer to Figure 3-11).

3-24 Receiver Operation

For proper receiver operation after power is applied, the External Reset must be enabled and a Clock Pulse having a frequency of 16 times the desired baud rate must be applied. The above conditions set the data available (DAV) to a logic Zero.

After initializing is completed, the user should note that one set of control bits are used for both receiver and transmitter making individual control bit setting unnecessary. Data reception starts when the serial input signal changes from Marking (logic One) to Spacing (logic Zero). This change initiates the start bit. The start bit is valid if, after transition from logic One to logic Zero, the SI line continues to be at logic Zero when center sampled eight clock pulses later. If the SI line is at a logic One when center sampling occurs, the start bit verification process is reset. If the Serial input line changes from a logic One to a logic Zero (Marking to Spacing) when the 16 x clock is in a logic One state, the bit time for center sampling begins when the clock line changes from a logic One to a logic Zero state. After verification of a genuine start bit, data bit reception, parity bit reception, and stop bit(s) reception proceed in an orderly manner.

Table 3-4. Description of Pin Functions (UART)

Pin No.	Name	Symbol	Function
1	V_{CC} Power Supply	V_{CC}	+5V Supply
2	V_{GG} Power Supply	V_{GG}	-12V Supply
3	Ground	V_{GI}	Ground
4	Received Data Enable	$\overline{RDE}$	A logic ONE on the Receive Enable line places the received data onto the Output lines.
5-12	Received Data Bits	RD8-RD1	These are the 8 Data Output lines. Received characters are right justified: the LSB always appears on RD1. These lines have tri-state outputs; i.e., they have the normal TTL output characteristics when $\overline{RDE}$ is ZERO and a high impedance state when $\overline{RDE}$ is ONE. Thus, the Data Output lines can be Bus Structure oriented
13	Parity Error	PE	This line goes to a logic ONE if the received character parity does not agree with the selected parity. Tri-state.
14	Framing Error	FE	This line goes to a logic ONE if the received character has no valid Stop bit. Tri-state.
15	Over-Run	OR	This line goes to a logic ONE if the previously received character is not read (DAV line not reset) before the present character is transferred to the receiver holding register. Tri-state.
16	Status Word Enable	$\overline{SWE}$	A logic ZERO on this line places the status word bits (PE, FE, OR, DAV, TBMT) onto the output lines. Tri-state.
17	Receiver Clock	RCP	This line will contain a clock whose frequency is 16 times (16X) the desired receiver baud rate.
18	Reset Data Available	$\overline{RDAV}$	A logic ZERO will reset the DAV line. Only the DAV X9 flip-flop is reset.
19	Data Available	DAV	This line goes to a logic ONE when an entire character has been received and transferred to the receiver holding register. Tri-state.
20	Serial Input	SI	This line accepts the serial bit input stream. A Marking (logic ONE) to Spacing (logic ZERO) transition is required for initiation of data reception.
21	External Reset	XR	Resets shift registers. Sets SO, EOC, and TBMT to a logic ONE. Resets DAV and error flags to ZERO. Clears input data buffer. Must be tied to logic ZERO when not in use.
22	Transmitter Buffer Empty	TBMT	The Transmitter Buffer Empty flag goes to a logic ONE when the Data Bits Holding Register may be loaded with another character. Tri-state.

Table 3-4. Description of Pin Functions (UART) (Cont'd)

Pin No.	Name	Symbol	Function
23	Data Strobe	$\overline{DS}$	A strobe on this line will enter the data bits into the Data Bits Holding Register. Initial data transmission is initiated by the rising edge of DS. Data must be stable during entire strobe.
24	End of Character	EOC	This line goes to a logic ONE each time a full character is transmitted. It remains at this level until the start of transmission of the next character.
25	Serial Output	SO	This line will serially, bit by bit, provide the entire transmitted character. It will remain at a logic ONE when no data is being transmitted.
26-33	Data Bit Inputs	DB1-DB8	There are 8 Data Bit Input lines available.
34	Control Strobe	CS	A logic ONE on this lead will enter the control bits (EPS, NB1, NB2, TSB, NP) into the Control Bits Holding Register. This line can be strobed or hard wired to a logic ONE level.
35	No Parity	NP	A logic ONE on this lead will eliminate the Parity bit from the transmitted and received character (no PE indication). The Stop bit(s) will immediately follow the last data bit. If not used, this lead must be tied to a logic ZERO.
36	Number of Stop Bits	TSB	This lead will select the number of Stop bits, 1 or 2, to be appended immediately after the Parity bit. A logic ZERO will insert 1 Stop bit, and a logic ONE will insert 2 Stop bits.
37-38	Number of Bits/ Character	NB2, NB1	These two leads will be internally decoded to select 5, 6, 7, or 8 data bits/character. NB2 NB1 Bits/Character 0 0 5 0 1 6 1 0 7 1 1 8
39	Odd/Even Parity Select	EPS	The logic level on this pin selects the type of parity which will be appended immediately after the data bits. It also determines the parity that will be checked by the receiver. A logic ZERO will insert odd parity, and a logic ONE will insert even parity.
40	Transmitter Clock	TCP	This line will contain a clock whose frequency is 16 times (16X) the desired transmitter baud rate.

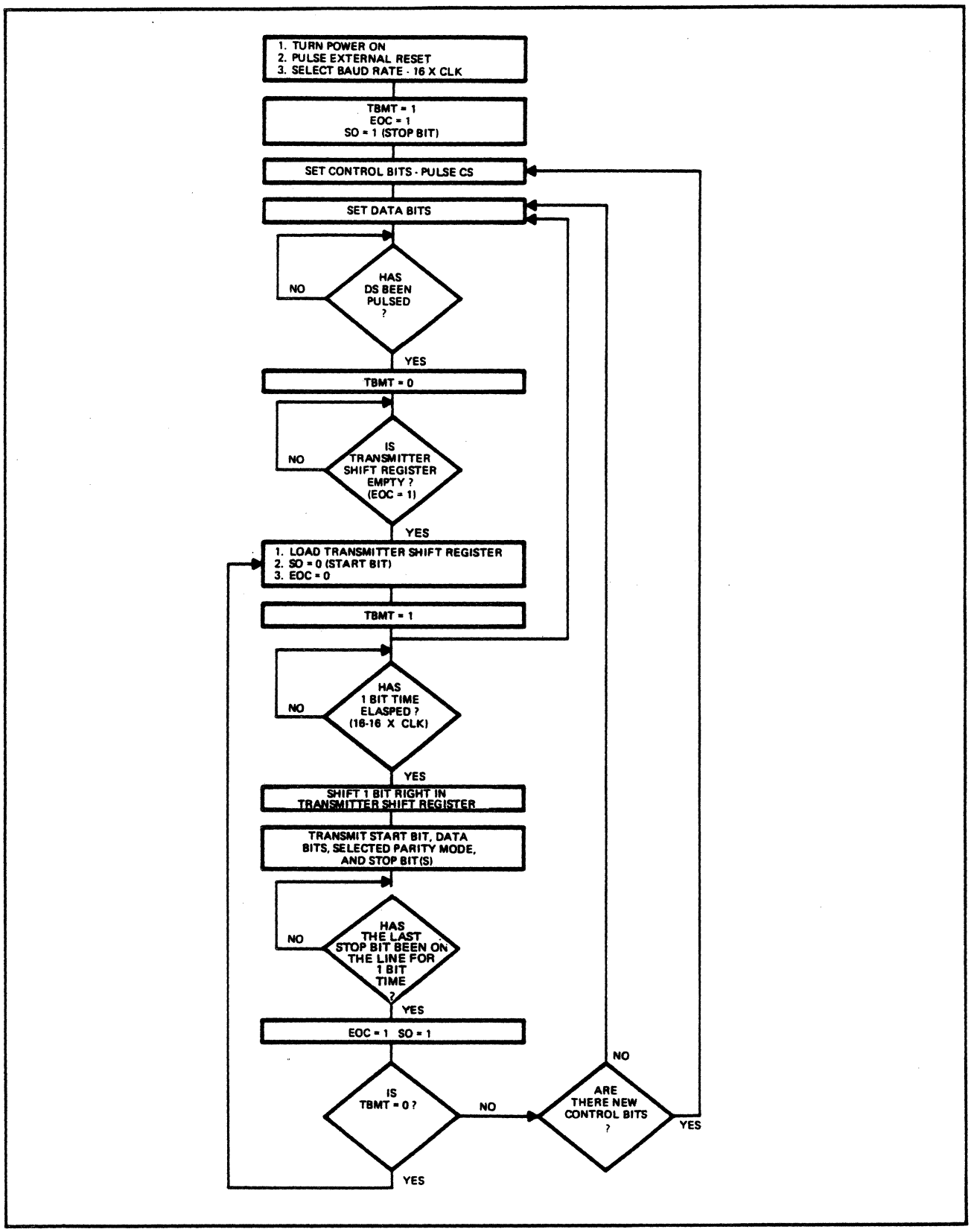


Figure 3-11. UART - Transmitter Flow Diagram

While receiving parity and stop bits, the receiver compares transmitted parity and stop bits with control data bits (parity and number of stop bits) previously set and indicates an error by changing the Parity Error flip-flop and/ or the Framing Error flip-flop to a logic One. Note that if the No Parity mode is selected, the Parity Error (PE) is unconditionally set to a logic Zero.

Once a full character is received, internal logic looks at the DAV signal to determine whether data has been read out. If the DAV signal is at a logic One, the receiver assumes that data has not been read out, and the Overrun flip-flop of the status word holding register is set to a logic One. If the DAV signal is at a logic Zero, the receiver assumes that data has been read out. After DAV goes to a logic One, the receiver shift register is ready to accept the next character, and has one full character time to remove the received character (refer to Figure 3-12).

3-25 OUTPUT DATA TRANSFER (TTY)

The TTY Device Dependent Interface Controller receives the Output Data transfer from the CPU and processes it as previously discussed. The MP places the data to be transferred to the teletypewriter into the A-register for transfer. Figure 3-13 outlines the sequence of events that takes place from the output of the MP to the output of data to the TTY.

The outputs of the MP A-register (LAIN08 - LAIN15), Logic Drawing 130-103175 (sheet 18), are applied to a noninverting buffer to prevent excessive loading of the A-register. The isolation buffer (sheet 21) generates the A-register Output signals (LAREGOUT08 - LAREGOUT15) and applies LAREGOUT08 through LAREGOUT15 to the input of the UART circuitry where it is loaded into the holding register by the Teletypewriter Strobe Input Data (LTTYSTBINDAT) signal. The transmitter circuitry of the UART circuit monitors the transmitter shift register. When it is empty, the transmitter circuitry loads the transmitter shift register with the TTY data and generates the Input Hold Register Empty (HINPUTRGEMTY) signal to the Data Request generator. At H3CLK2 time, the transmitter circuitry of the UART circuit generates the Teletypewriter Data Request (HTTYDATAREQ) signal for Test Structure logic (sheet 15) to obtain the next character.

The UART clock generator (sheet 24), composed of synchronous 4-bit counters and Baud Rate switches, uses the 150-nanosecond H2CLKFREERUN signal to generate the 568-microsecond UART Clock (HUARTCLK) signal. The HUARTCLK signal triggers the shift operation of the transmitter shift register, shifting the serial data which contains a start bit, seven data bits, one blank, and two stop bits (HTTYSERDATOT), onto the peripheral device connector P1A-10 (EIA RS-232-C) or P1A-5 (current loop) to the teletypewriter device.

3-26 INPUT DATA TRANSFER (TTY)

The TTY Device Dependent Interface Controller receives the input data from the TTY at the peripheral device connector P1A-8, (EIA RS-232-C) or P1A (current loop), Logic Drawing 130-103175 (sheet 25) (refer to Figure 3-14). The Data Input (HDATAIN) generates the Teletypewriter Serial Data Input (HTTYSERDATIN) signal and applies it to the UART circuitry (sheet 25), where it is shifted and loaded into the receiver shift register by the HUARTCLK signal from the UARTCLK generator (sheet 24). The TTY data is then transferred from the shift register to the holding register. The UART circuit generates the Received Data Available (HRCVDDATAVAL) signal and applies it to the Teletypewriter Data Here generation circuit to generate the HTTYDATAHERE signal to the IOM Test Structure logic (sheet 15).

The IOM processes the HTTYDATAHERE signal and returns the Enable Receive TTY Data (LENRCVTYDATA) signal (sheet 20) to gate the data and/or status to the IOM (LEXT00-03 and LEXT08-15). The IOM generates the Reset Data Available (LRSTDATAVAL) signal, which is applied to the UART circuit, removing the Received Data Available (HRCVDDATAVAL) signal.

Figure 3-15 shows the DRT from the TTY to the IOM.

3-27 LINE PRINTER (LP) DEVICE DEPENDENT INTERFACE CONTROLLER

The LP Device Dependent Interface Controller provides the circuitry necessary to output data from the IOM to the line printer or to input status from the line printer to the IOM.

The CPU outputs data to the TLC Controller where it is decoded and formatted by the IOM firmware for data or commands to the LP Device Dependent interface. The following paragraphs discuss the sequence of events that takes place from the A-register of the MP to the output of the Line Printer Device Dependent interface. Figure 3-16 shows these steps in a flow diagram for a Print and Paper Advance operation. Figure 3-17 illustrates a signal flow in a functional block diagram; Figure 3-18 shows the WDOT format; and Figure 3-19 shows the DRT format between the IOM and the line printer.

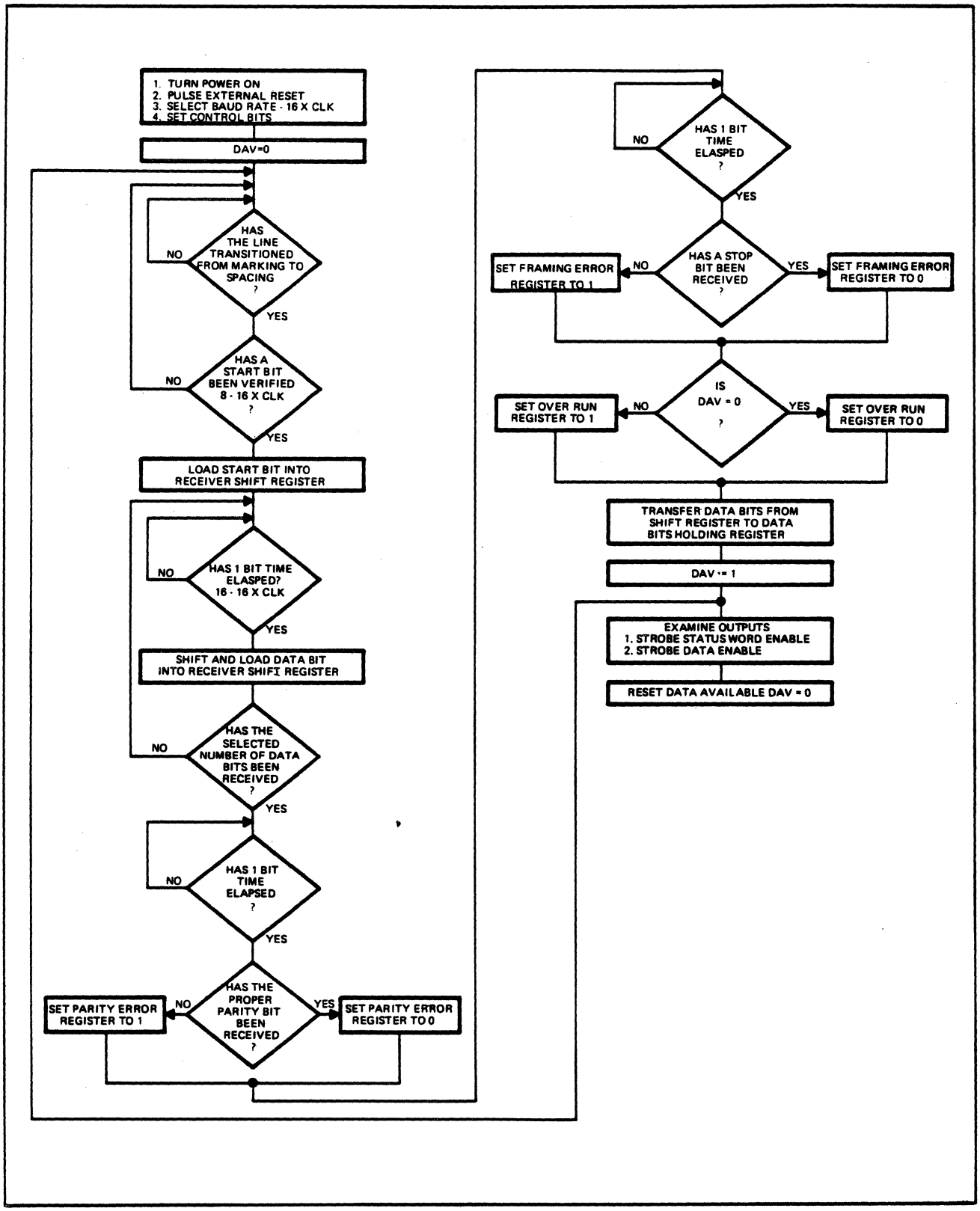


Figure 3-12. UART - Receiver Flow Diagram

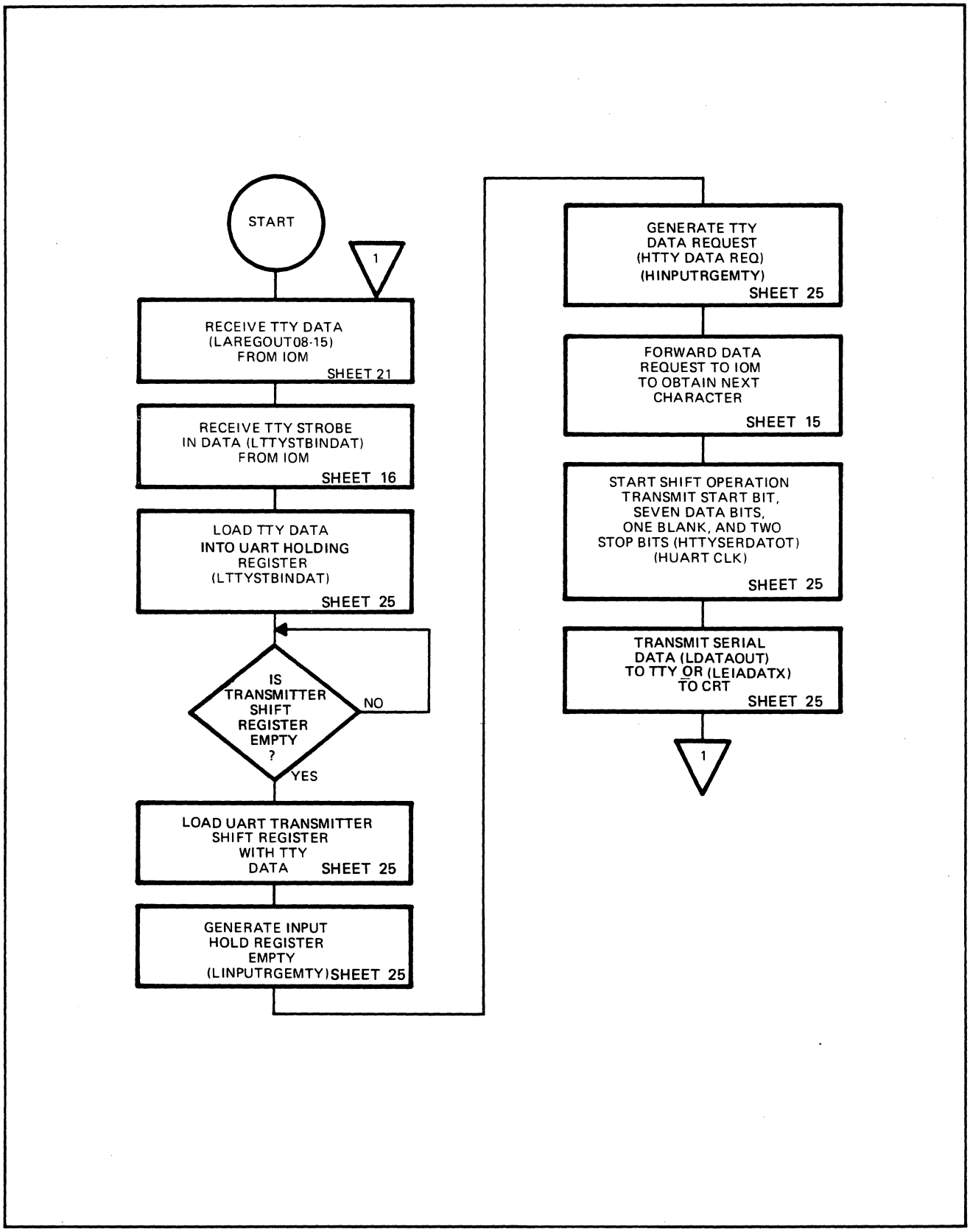


Figure 3-13. Output Data Transfer (TTY)

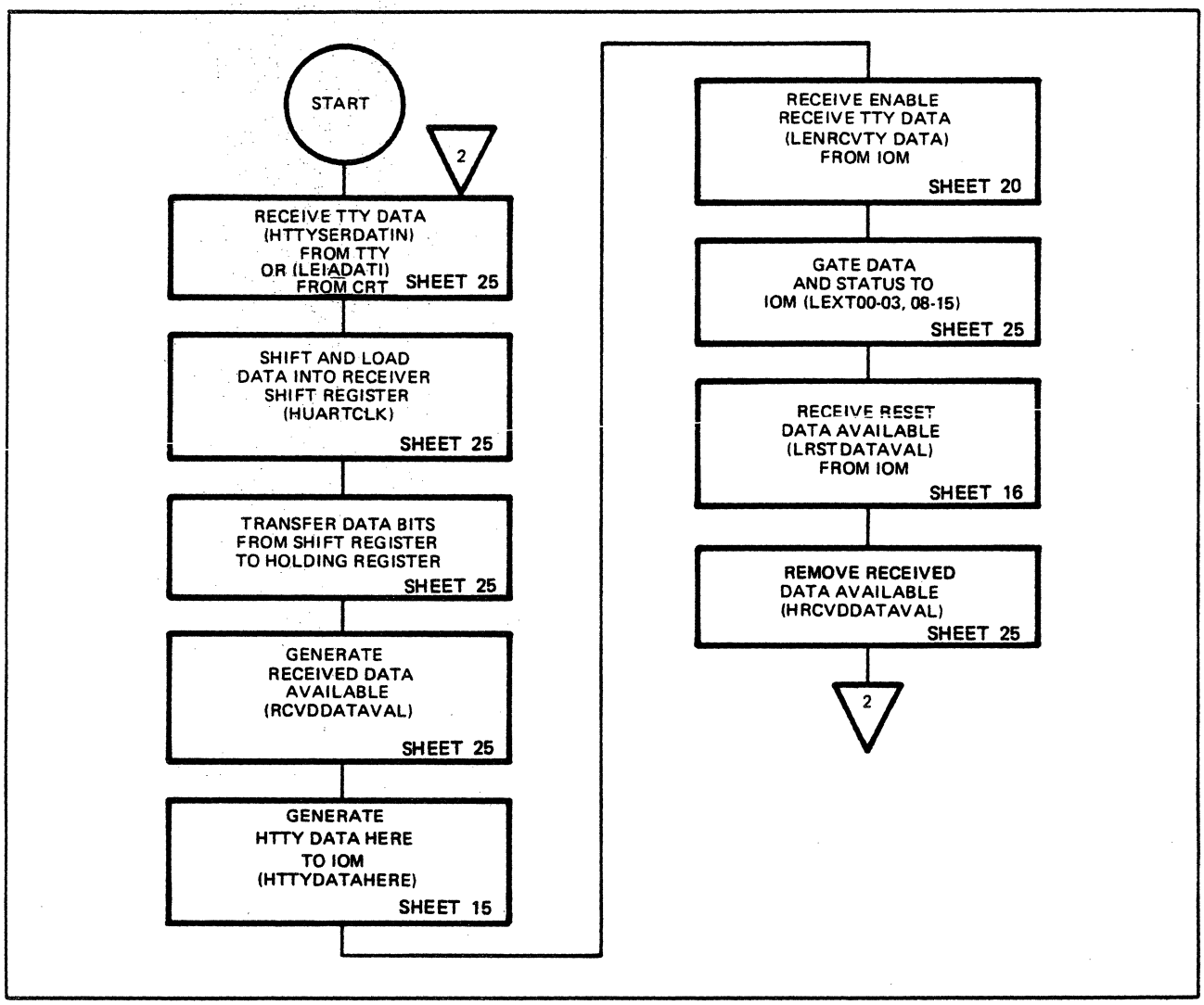


Figure 3-14. Input Data Transfer (TTY)

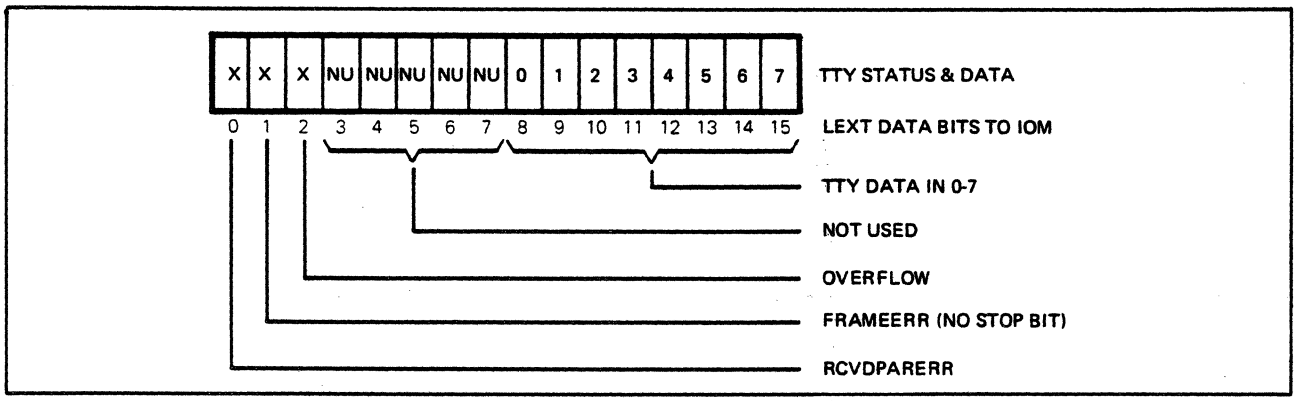


Figure 3-15. TTY DRT Format

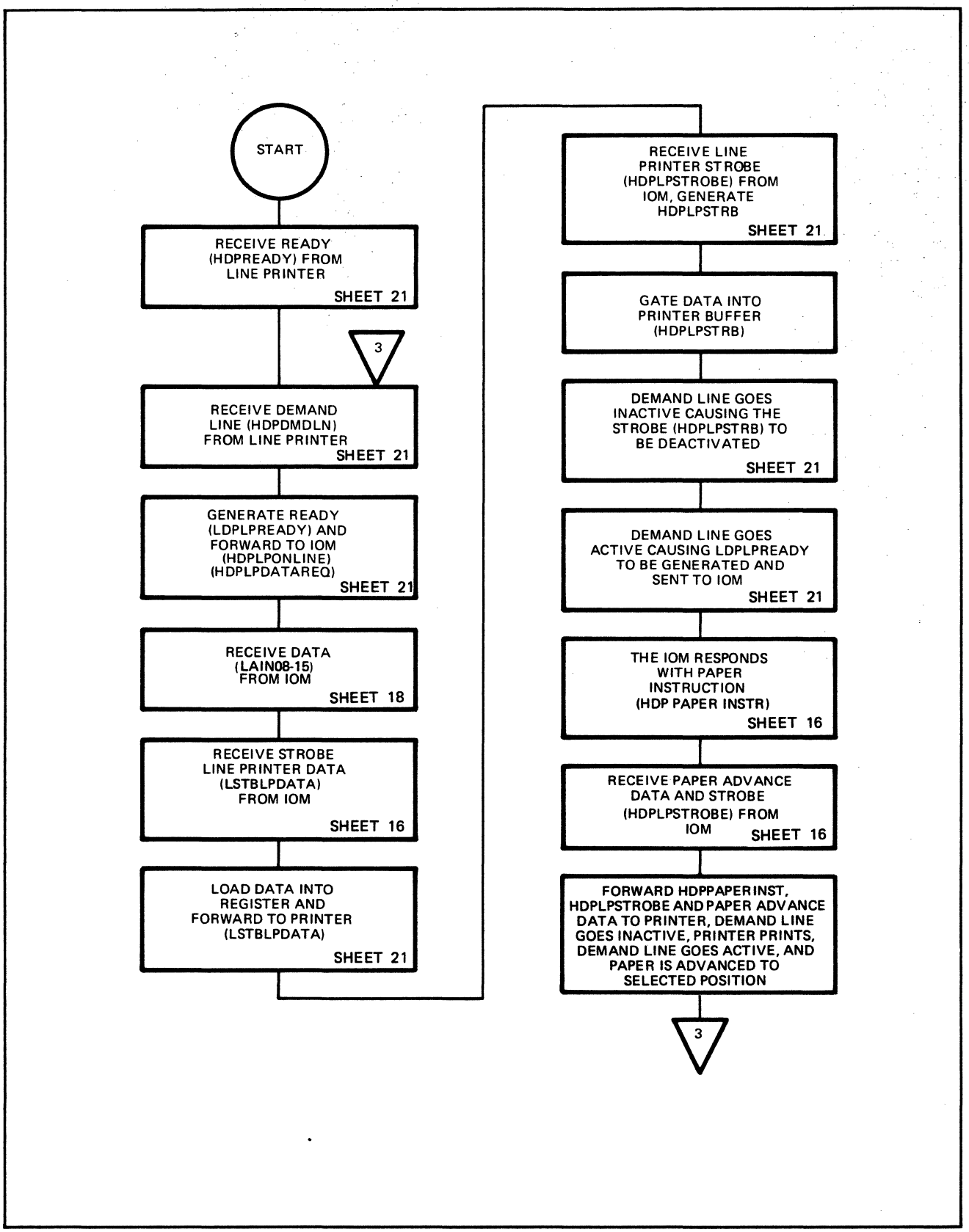


Figure 3-16. Print and Paper Advance Operation

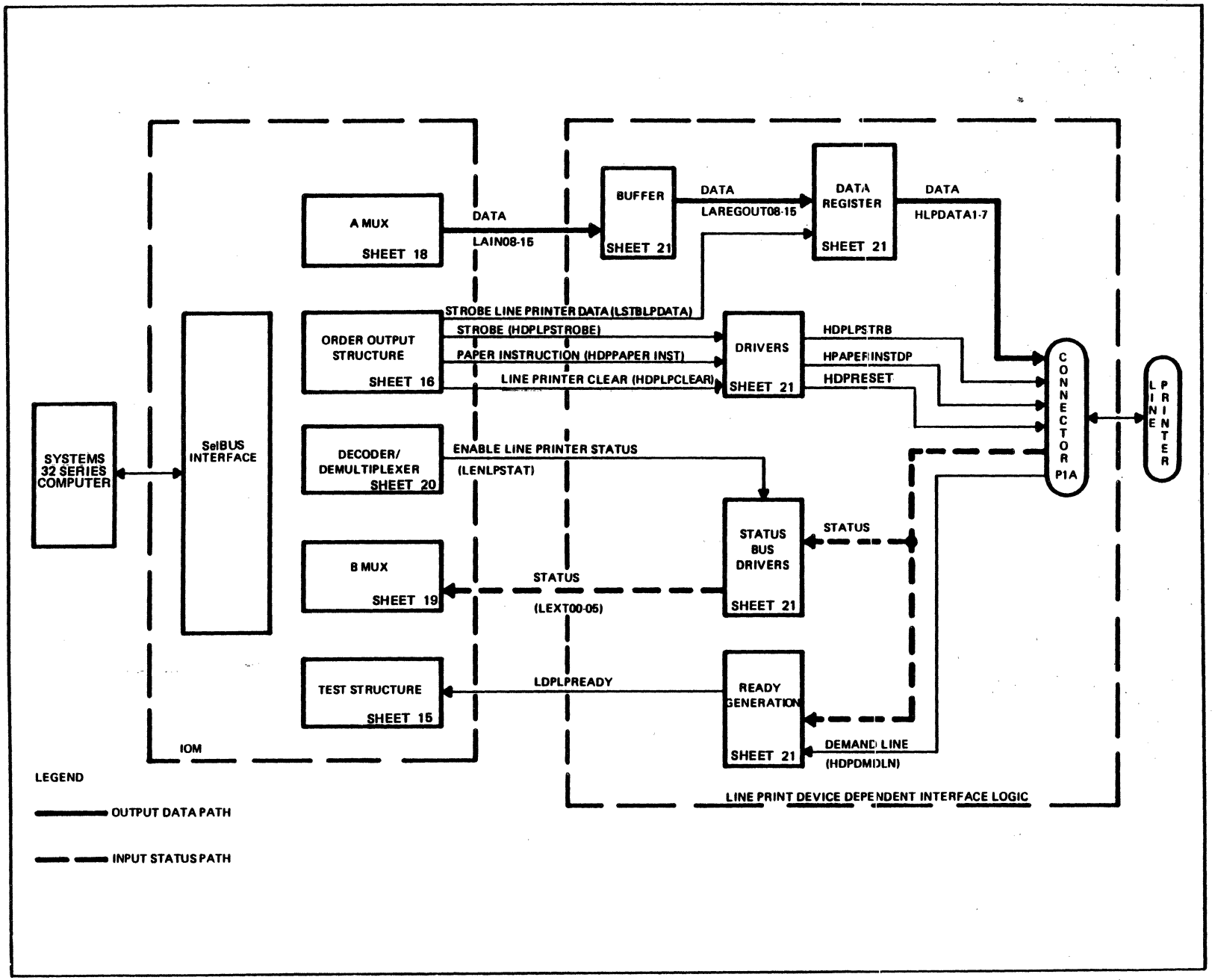


Figure 3-17. Line Pinter Device Dependent Interface Controller

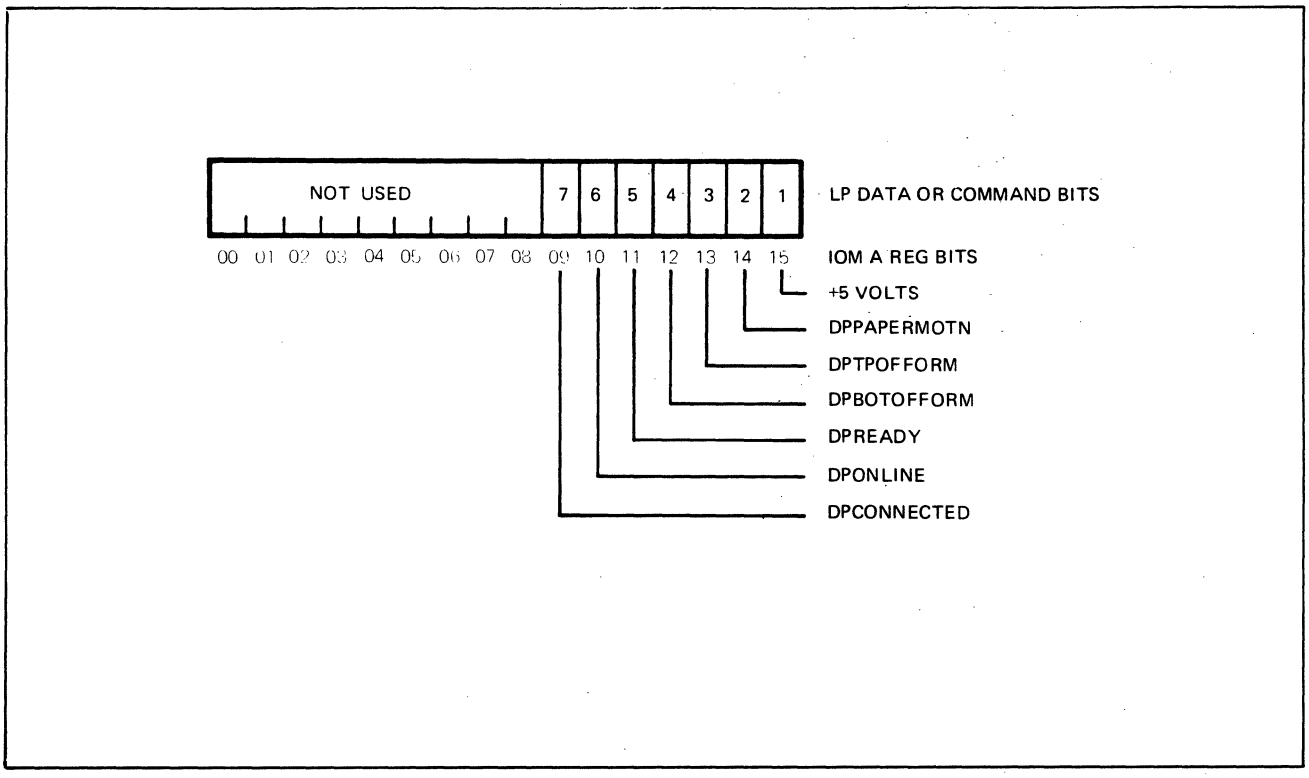


Figure 3-18. IOM to LP WDOT Format

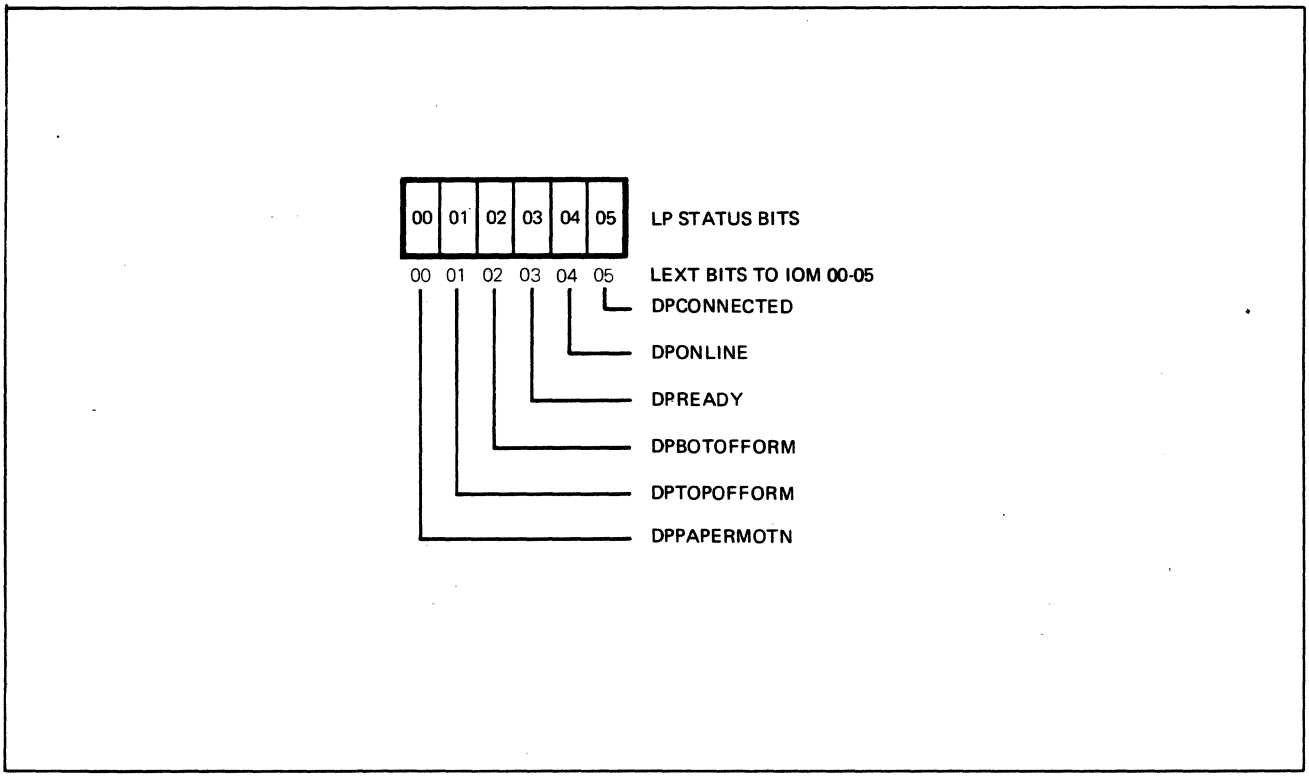


Figure 3-19. LP to IOM DRT (Status) Format

The LP Device Dependent interface receives the Ready (HDPREADY) signal from the line printer at the peripheral device connector P1A-23, and the Demand Line (HDPDMDLN) signal at P1A-29, Logic Drawing 130-103175 (sheet 21). At H3CLK2 time, these signals are clocked into the HDPLPONLINE and HDPLPDATAREQ registers, respectively, to generate the Line Printer Ready signal to the MP.

The output of the MP A-register (LAIN08 - LAIN15) is applied to the isolation buffers (sheet 21) to generate the A-register Output signals (LAREGOUT08 - LAREGOUT15) and apply the 7-bit ASCII data bits (LAREGOUT08 - LAREGOUT15) to the input of the data register. The MP then sends the Strobe Line Printer Data (LSTBLPDATA) signal (sheet 16), strobing the data into the data register where it is forwarded to the line printer (HLPDATA1 - HLPDATA7) at the peripheral device connector P1A-30 with the Line Printer Strobe (HDPLPSTRB) signal (sheet 21). The Demand Line (HDPDMDLN) signal goes Low, removing the Line Printer Strobe (HDPLPGTRB) signal.

The Demand line goes active sending the LDPLDREADY signal to the MP. The MP now generates the Paper Instruction (HDPPAPERINST) signal, the Paper Advance data, and the Line Printer Strobe (HDDLPSTROBE) signal to the printer. The Demand line goes Low, and the Printer prints; the Demand line goes High, and paper is advanced.

Note

If Paper Advance is desired without printing, the DPLPCLEAR signal (sheet 16) is generated and forwarded to the line printer (P1A-15); this signal clears the printer buffer, advancing the paper without printing characters.

Status is returned to the MP when the CPU issues an ARSTX/RSTX transfer request. The MP issues the Enable Line Printer Status (LENLPSTAT) signal. The Line Printer status is constantly present at the peripheral device connector P1A (sheet 21). The LENLPSTAT signal gates the status to the MP which returns the information to the CPU with a DRT transfer.

3-28 CARD READER (CR) DEVICE DEPENDENT INTERFACE

The CR Device Dependent interface contains the required logic circuitry to output commands/orders from the MP to the card reader or to input data or status from the card reader to the MP. Also contained in the logic is the circuitry for Hollerith to ASCII code conversion, allowing operation in either the Binary, Full-ASCII, or Half-ASCII mode.

Figure 3-20 outlines the sequence of events when data is transferred from the card reader to the MP. Figure 3-21 shows the signal flow in a functional block diagram, and Figure 3-22 establishes the Data and/or Status format of a DRT from the card reader to the MP. All logic references called out in the following discussion refer to Logic Drawing 130-103175 unless otherwise noted.

The IOM receives the instruction from the CPU, processes it, generates the 75-nanosecond negative pulse order Read Card (LRDCARD) at the order output structure (Logic Sheet 11), and applies it to the J-input of the Pick Command flip-flop (sheet 23). The H2CLK3 signal, which occurs 65 nanoseconds later, triggers the Pick Command flip-flop set generating the Pick Command (HPC) signal to the peripheral device connector P1C-1. From P1C-1, the HPC signal is applied to the card reader.

The card reader responds to the Pick Command by reading a card and generating the Busy (HBSY) signal at P1C-27 (sheet 23) when the picked card is under the read head. Each column of the picked card generates an Index Mark (HIM) signal P1C-29 (Sheet 23). The HBSY and HIM signals generate the Index Mark (HINDEX) signal.

The HINDEX signal (sheet 23) enables the D1 input of circuit U69. On the next positive-going edge of the H3CLK21 signal, the level at the D1 input is transferred to the Q1 output generating the HINDEX-1 signal. The LINDEX-1 signal is generated from the output. The Q1 output enables the D2 input of circuit U69. On the next postive-going edge of the H3CLK21 signal, the level at the D2 input is transferred to the Q2 output, generating the HINDEX-2 signal. This signal enables the D3 input of circuit U69. On the next postive-going edge of the H3CLK21 signal, the level at the D3 input is transferred to the Q3 output, generating the HINDEX-3 signal. The HINDEX-2 and HINDEX-3 signals generate the HINDEX 23 signal. When the HINDEX signal goes Low at the D1 input, the next positive-going clock transfers a Low level to the Q1 output, deactivating the LINDEX-1 signal. This action generates the HINDEXMRK signal. With the HINDEXMRK signal providing an enable signal to the Card Reader Data flip-flop (U70-1), the next negative-going edge of the L3CLK21 signal sets the Card Reader Data flip-flop generating the HCRDATAHERE signal. This signal is applied to the MP Test Structure.

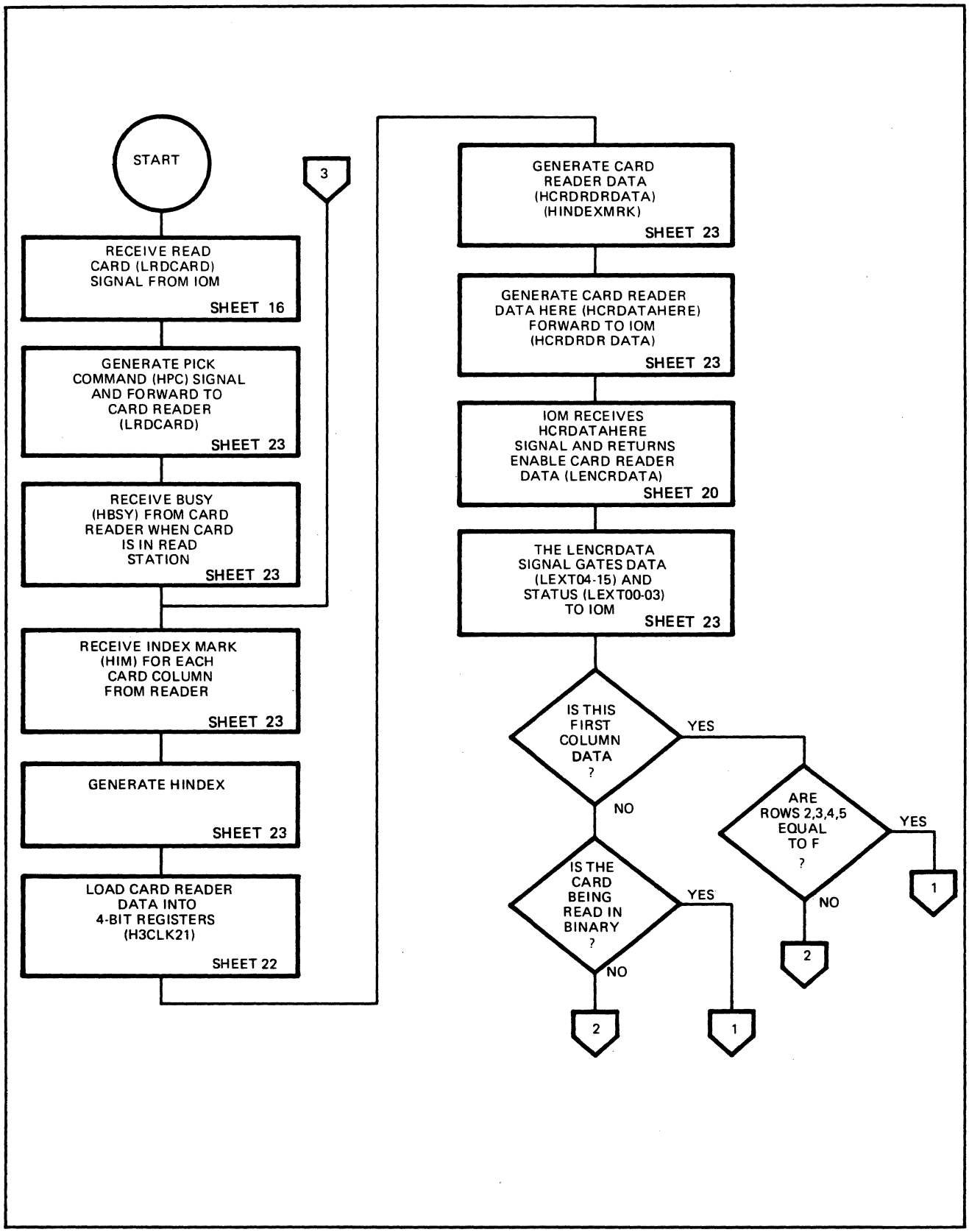


Figure 3-20. Card Reader Operation (Sheet 1 of 2)

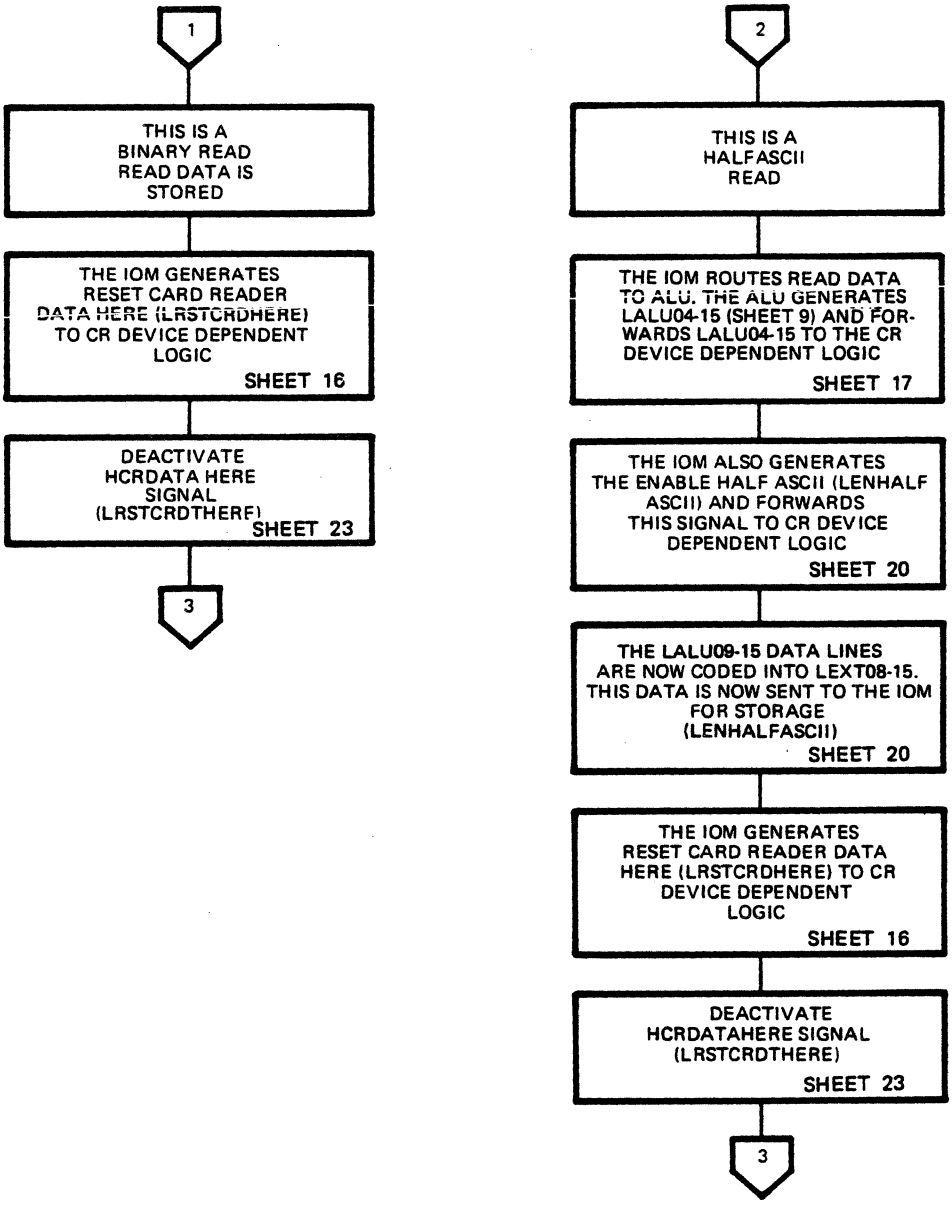


Figure 3-20. Card Reader Operation (Sheet 2 of 2)

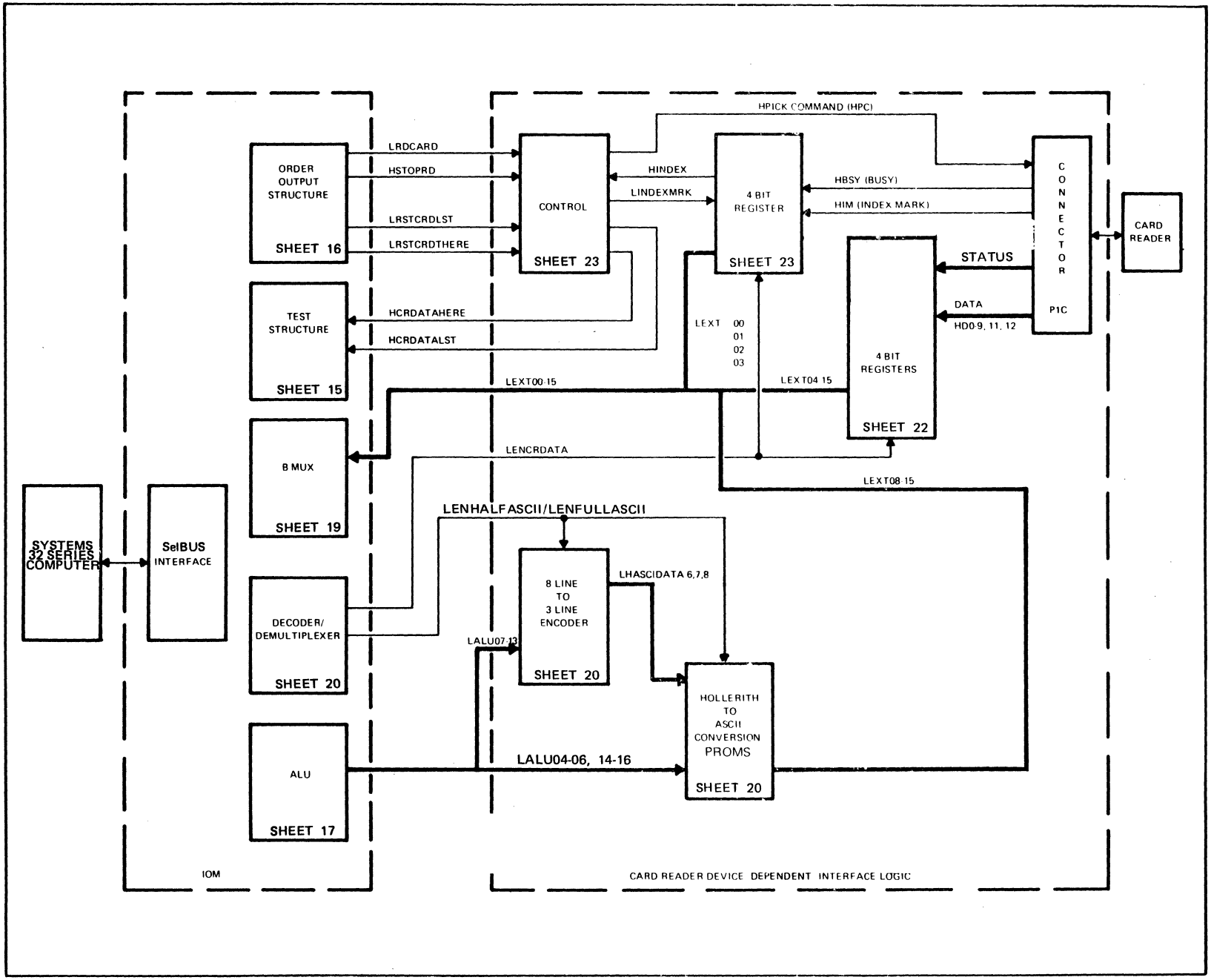


Figure 3-21. Card Reader Device Dependent Interface Controller

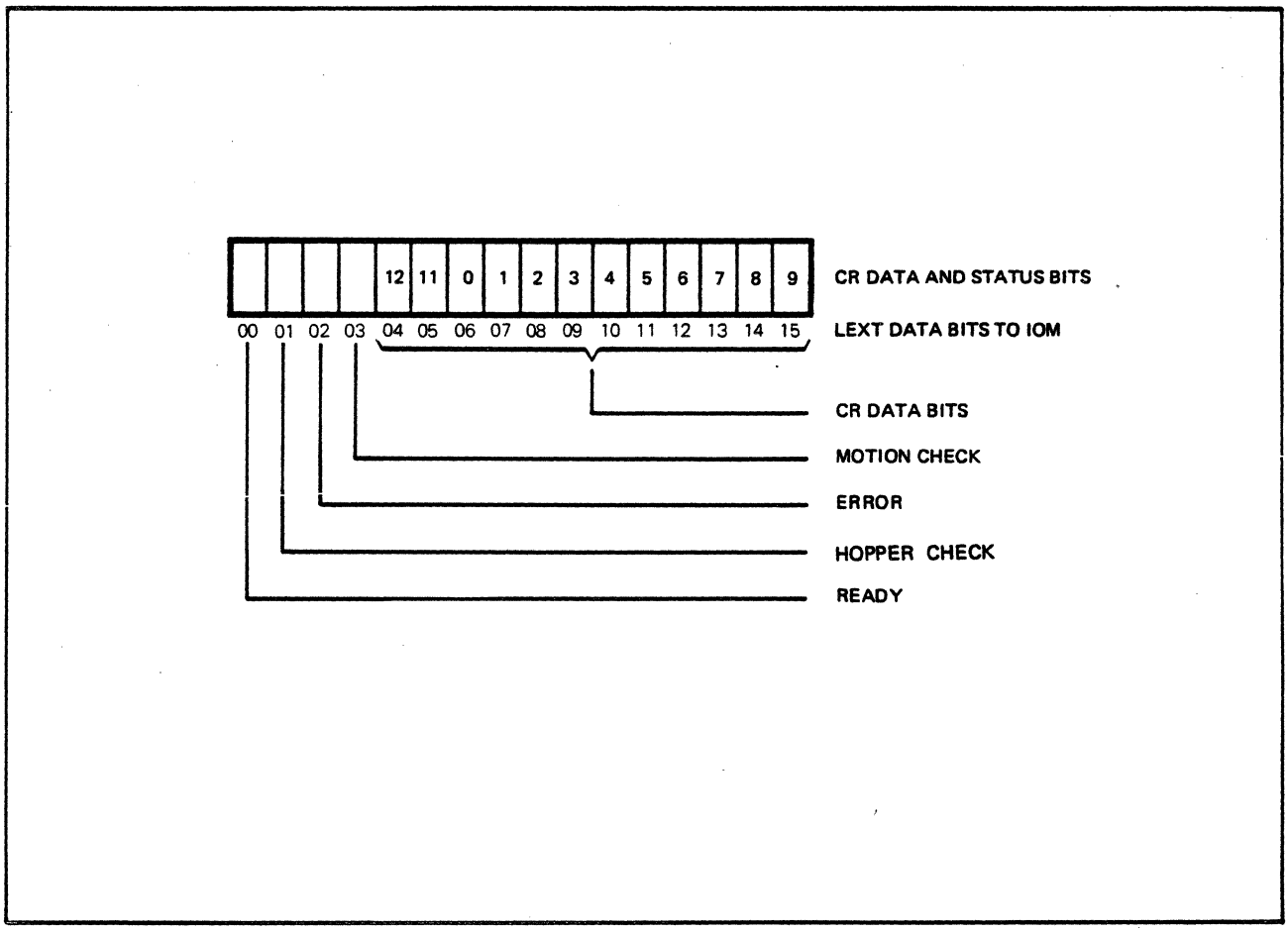


Figure 3-22. CR to IOM DRT (Data/Status) Format

The MP receives the HCRDATAHERE signal (sheet 15), interrogates the test A2 conditions, and generates the Enable Card Reader Data (LENCRDATA) signal (sheet 20) which is applied to the Buffer registers (sheet 22), gating data and status (LEXT00-LEXT15) to the IOM B-register.

Note

If the Card Reader Device Dependent interface receives the next index mark before the MP accepts the data, the HCRDATAHERE and HINDEXMRK signal set the Card Reader Data Lost flip-flop (sheet 23) generating the LCRDATALST signal to the MP Test Structure (sheet 15) where it is examined. The Reset Card Read Data Lost (LRSTCRDLST) signal is generated (sheet 16) to reset the Card Reader Data Lost flip-flop, aborting the operation.

The MP examines the first column data, rows 2, 3, 4, and 5, to determine whether they are set (equals F). If all are set, the operation is in the Binary mode, the data is stored, and the Reset Card Reader Data Here (LRSTCRDHERE) signal (sheet 16) is used to deactivate the Card Reader Data Here (HCRDATAHERE) signal (sheet 23). The process is repeated until all cards have been read.

If the first column data, rows 2, 3, 4, and 5 are not set (equals 0), the operation is in the Half-ASCII mode. The MP routes the read data to the ALU where the LALU00-15 bits are generated (sheet 17) and the LALU04-15 bits are applied to the Data Select circuitry (sheet 20). The MP generates the Enable Half-ASCII (LENHALFASCII) signal (sheet 20) which gates the converted Half-ASCII data bits (LEXT08-15) back to the MP B-register. The MP now generates the Reset Card Reader Data Here (LRSTCRDHERE) signal (sheet 16) deactivating the HCRDATAHERE signal. The process is repeated until all cards have been read.

APPENDIX A

TELETYPEWRITER MNEMONIC LIST

Mnemonic	Definition
HASCIIDATA2-8	ASCII Data
HBSY	Busy
HCRDATAHERE	Card Reader Data Here
HCRDATALST	Card Reader Data Lost
HCRDRDRDATA	Card Reader Data
HDO-9,11,12	Card Reader Data
HDATAIN	Data In
HDATAOUT	Data Out
HDPBOTOFFORM	Printer Bottom of Form
HDPDMDLN	Printer Demand Line
HDPLPCLEAR	Clear Line Printer Buffer
HDPLPDATAREQ	Line Printer Data Request
HDPLPONLINE	Line Printer On-Line
HDPLPREADY	Line Printer Ready
HDPLPSTRB	Line Printer Strobe
HDPLPSTROBE	Line Printer Strobe
HDPONLINE	Printer On-Line
HDPPAPERINST	Printer Paper Instruction
HDPPAPERMOTN	Printer Paper Motion
HDPREADY	Printer Ready
HDPTOPOFFORM	Printer Top of Form
HERROR	Error
HFRAMEERR	Frame Error
HHCK	Hopper Check
HHTYDATAREQ	TTY Data Request
HIM	Index MART
HINDEXMRK	Index Mark
HINPUTRGEMTY	Input Register Empty
HLPDATA1-7	Line Printer Data
HMOCK	Motion Check
HOVERFLOW	Overflow Error
HPAPERINSTOP	Printer Paper Instruction
HPC	Pick Command
HRCVDDATAVAL	Receive Data Available
HRCYDPARERR	Receive Character Parity Error
HRDY	Ready
HSTOPRD	Stop Read
HTTYDATAHERE	TTY Data Here
HTTYDATAINO-7	TTY Data In
HTTYSERDATIN	TTY Serial Data In
HTTYSERDATOT	TTY Serial Data Out
HUARTCLK	UART Clock (568 μs)
LAINO4-15	A-Register Data Bits
LALUO4-15	Arithmetic Logic Unit Bits
LAREGOUTO4-15	A-Register Out Data Bits
LAREGOUTO8-15	A-Register Out Data Bits
LDPCONNECTED	Printer Connected

APPENDIX A (CONT'D)

TELETYPEWRITER MNEMONIC LIST

Mnemonic	Definition
LEIADATI	EIA Data In
LEIADATX	EIA Data Transmit
LENASCII	Enable ASCII
LENCRDATA	Enable Card Reader Data
LENHALFASCII	Enable Half ASCII
LENLPSTAT	Enable Line Printer Status
LENRCVTYDATA	Enable Receive TTY Data
LEXT00-03	Status Bits
LEXT00-05	Status Bits
LEXT04-15	Data Bits
LEXT08-15	Data Bits
LHASCIDATA6-8	Half ASCII Data
LRDCARD	Read Card
LRSTCRDLST	Reset Card Reader Data Lost
LRSTCRDTHERE	Reset Card Reader Data Here
LRSTDATAVAL	Reset Data Available
LSTBLPDATA	Strobe Line Printer Data
LTTYSTBINDAT	TTY Strobe Data In

APPENDIX B

LINE PRINTER MNEMONIC LIST

Mnemonic	Definition
HDPBOTOFFORM	Printer Bottom of Form
HDPDMDLN	Printer Demand Line
HDPLPCLEAR	Clear Line Printer Buffer
HDPLPDATAREQ	Line Printer Data Request
HDPLPONLINE	Line Printer On Line
HDPLPREADY	Line Printer Ready
HDPLPSTRB	Line Printer Strobe
HDPLPSTROBE	Line Printer Strobe
HDPONLINE	Priner On Line
HDPPAPERINST	Printer Paper Instruction
HDPPAPERMOTN	Printer Paper Motion
HDPREADY	Printer Ready
HDPTOPOFFORM	Printer Top of Form
HLPDATA1-7	Line Printer Data
HPAPERINSTDP	Printer Paper Instruction
LAIN04-15	A Register Data Bits
LAREGOUT04-15	A Register Out Data Bits
LDPCONNECTED	Printer Connected
LENLPSTAT	Enable Line Printer Status
LEXT00-05	Status Bits
LSTBLPDATA	Strobe Line Printer Data

APPENDIX C

CARD READER MNEMONIC LIST

Mnemonic	Definition
HASCIIDATA2-8	ASCII Data
HBSY	Busy
HCRDATAHERE	Card Reader Data Here
HCRDATALST	Card Reader Data Lost
HCRDRDRDATA	Card Reader Data
HDO-9,11,12	Card Reader Data
HERROR	Error
HHCK	Hopper Check
HIM	Index Mart
HINDEXMRK	Index Mark
HMOCK	Motion Check
HPC	Pick Command
HRDY	Ready
HSTOPRD	Stop Read
LALU04-15	Arithmetic Logic Unit Bits
LENASCII	Enable ASCII
LENCRDATA	Enable Card Reader Data
LENHALFASCII	Enable Half ASCII
LEXT00-03	Status Bits
LEXT04-15	Data Bits
LHASCIDATA6-8	Half ASCII Data
LRDCARD	Read Card
LRSTCRDLST	Reset Card Reader Data Lost
LRSTCRDTHERE	Reset Card Reader Data Here

APPENDIX D

TLC JUMPER CHART

LOCATION	FUNCTION
X1-1 through X3-6	SelBUS Priority Generation (1-22)
X3-7 through X3-8	Unused
X4-1	Unused
X4-2 through X6-6	SelBUS Priority Recognition (1-21)
X6-7 through X6-8	Unused
X7-1 through X7-7	Physical Address (MSB-LSB)
X7-8	Unused
X7-9	UART Stop Bit Selection (1 or 2)
X7-10	TTY Input Selection (EIA RS-232-C or Current Loop)
X8-1 through X8-7	UART Baud Rate Selection
X8-8	Unlisted
X9-1 through X9-8	UART Baud Rate Selection
X10-1,4,7,8	Unused
X10-2,3,5,6	UART Baud Rate Selection
X11-1 through X11-6	Unused
X11-7	Half-ASCII Mode
X11-8	Full-ASCII Mode

Reader's Comment Form

Date _____

Manual Title: _____

Publication Number _____

- **How do you use this publication?**

  ☐ As an introduction to the subject.
  ☐ Emergency Maintenance.
  ☐ Other _____

- **Is the material:**

	Yes	No
Easy to read?	☐	☐
Well organized?	☐	☐
Complete?	☐	☐
Well illustrated?	☐	☐
Accurate?	☐	☐
Suitable for its intended use?	☐	☐

- **Please check the items that describe your position:**

☐ Customer Personnel
☐ SYSTEMS Personnel
☐ Engineer
☐ Sales Representative
☐ Technician
☐ Field Service
☐ Operator
☐ Programmer
☐ Instructor
☐ Trainee
☐ Other _____

- **Please check specific criticism(s), give page number(s), and explain below:**

☐ Clarification on page(s) _____
☐ Addition on page(s) _____
☐ Deletion on page(s) _____
☐ Error on page(s) _____

Explanation:

Your Name: _____

Your Company: _____

402-02 (4/81)
Printed in USA

May we have your comments?

This publication is one of a series of SYSTEMS Engineering Laboratories technical documents written to serve each of a wide variety of users. Your completion of the attached form will aid SYSTEMS in the continued production of complete, easily referenced material in each of these various technical publications.

Thank You

Fold and Staple for Mailing

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 2356 FORT LAUDERDALE, FLORIDA

POSTAGE WILL BE PAID BY ADDRESSEE

SYSTEMS Engineering Laboratories
6901 West Sunrise Boulevard
Fort Lauderdale, Florida 33313

Fold and Staple for Mailing

Proven COMPUTER Performance

SYSTEMS

A Subsidiary of GOULD INC.

6901 WEST SUNRISE BLVD., FT. LAUDERDALE, FLORIDA 33313