Publication Number C29-691

MODEL 550
VIDEO DISPLAY UNIT (VDU)
INSTALLATION AND PROGRAMMING MANUAL

PERKIN-ELMER

Computer Systems Division
2 Crescent Place
Oceanport, N.J. 07757

PAGE REVISION STATUS SHEET

PUBLICATION NUMBER C29-691

TITLE Model 550 Video Display Unit (VDU) Installation and Programming Manual

REVISION R00 DATE April 1979

PAGE	REV.	DATE	PAGE	REV.	DATE	PAGE	REV.	DATE
i/ii	R00	4/79						
iii thru vi	R00	4/79						
1-1 thru 1-9/ 1-10	R00	4/79						
2-1 thru 2-47/ 2-48	R00	4/79						
A-1 thru A-35/ A-36	R00	4/79						
B-1 thru B-44	R00	4/79						
Index-1 thru Index-5/ Index-6	R00	4/79						

PREFACE

This manual provides the information necessary to install and program the Model 550 VDU and the Model 655 Thermal Printer. Chapter 1 provides installation and switch selection information. Chapter 2 provides a complete programming guide. The appendices provide 16- and 32-bit programming examples.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

APPENDICES

FIGURES

TABLE OF CONTENTS (Continued)

TABLES

CHAPTER 1
MODEL 550 VIDEO DISPLAY UNIT (VDU) INSTALLATION GUIDE

1.1 INTRODUCTION

The Model 550 VDU is a single-unit, desk-top terminal with configuration options to meet basic user requirements. The terminal (27-110) is designed around the Model 550 VDU chip. It is a self-contained unit, consisting of a display, keyboard, power supply, and associated electronics. An additional Model 655 Thermal Printer, a page printer, is detailed in this chapter.

1.2 UNPACKING

Refer to the Model 550 VDU User's/Maintenance Manual, Publication Number 29-690, for instructions on unpacking the terminal.

1.3 APPLICABLE PART NUMBERS

02-646	Model 550 - 110 VAC, 60 Hz, Standard Keyboard
27-110F00	Terminal - 110 VAC, RS-232 Communication Line Interface
02-646F01	Model 550 - 110 VAC, 60 Hz, Standard Keyboard with Printer Port
27-110F01	Terminal - 110 VAC, RS-232 Communication Line Interface with Printer Port
02-647	Model 550 - 100/110/230 VAC, 50/60 Hz Standard Keyboard with Printer Port
27-110F02	Terminal - 100/110/230 VAC, 50/60 Hz RS-232 Communication Line Interface, with Printer Port
17-379	External Current Loop Cable
17-508	Terminal to Printer Port Cable
02-648	Terminal Current Loop (Option)
02-658	Model 655 Thermal Printer 110 VAC, 50/60 Hz
02-659	230 VAC Line Cord
17-508	Printer Port Cable
02-606	Thermal Paper (1 case)
02-660	Model 655 Thermal Printer 100 VAC, 50/60 Hz

1.4 MODEL 550 AND MODEL 655 INSTALLATIONS

1.4.1 RS-232 - Type Communication Line

Figure 1-1 shows the necessary cables and their installation with various RS-232 type interfaces.

1.4.2 Current Loop Communication Type Line

Figure 1-2 shows the necessary hardware and cables required to install the Model 550 VDU and its optional Model 655 Thermal Printer on the Perkin-Elmer Computer Systems.

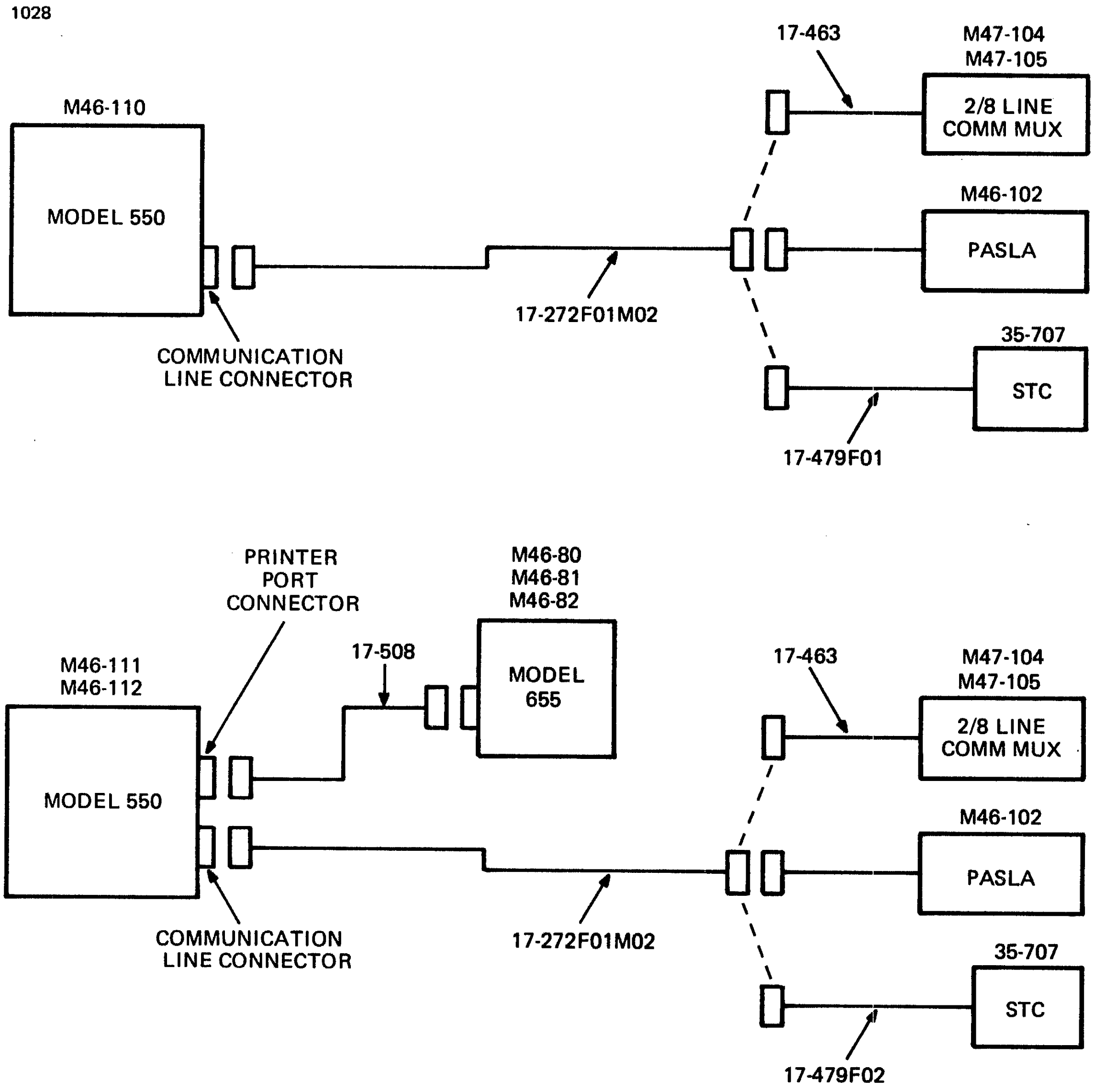


Figure 1-1 Local RS-232 Communication Line Configuration

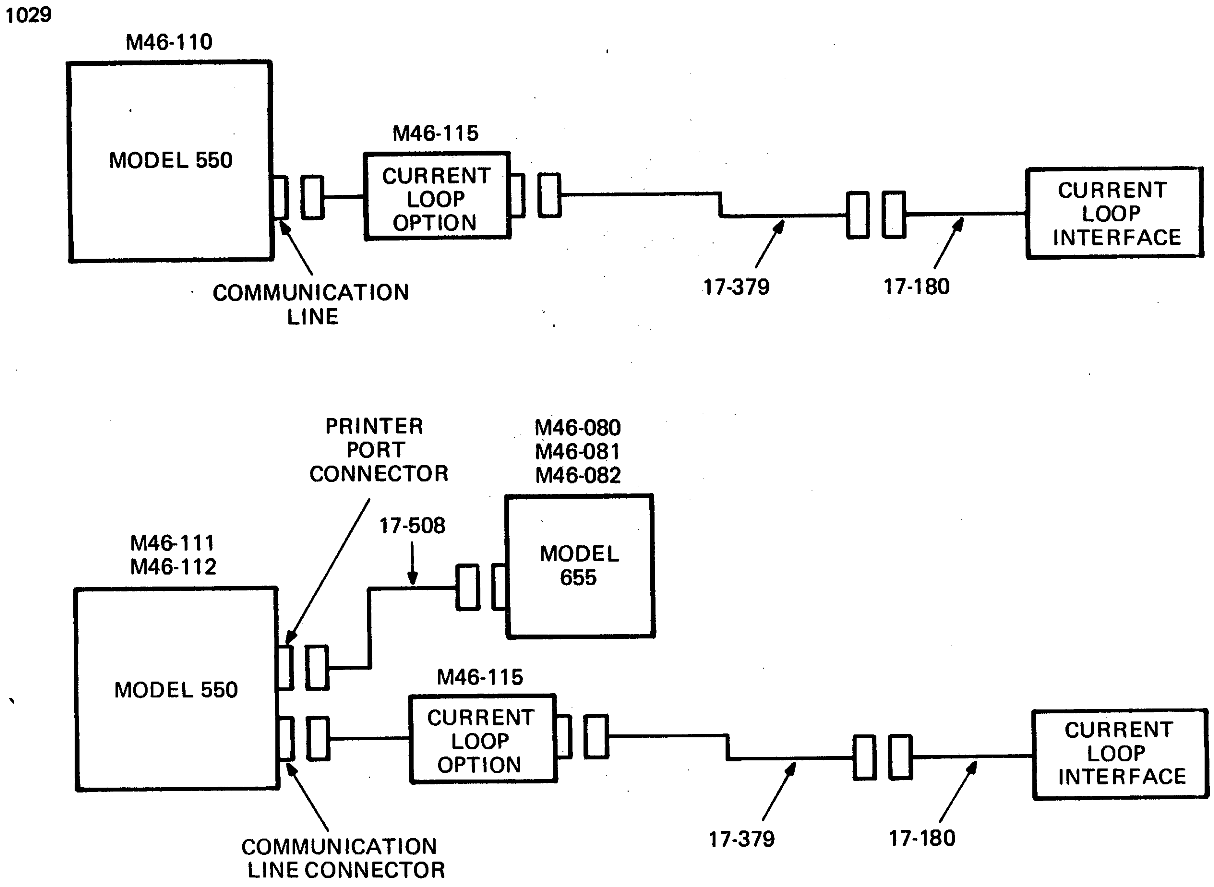


Figure 1-2 Current Loop Communication Line

1.4.3 AC Power and Fusing Requirements

Marketing Number	AC Line Voltage and Frequency	Fuse Size (AMPS)
M46-110	110 VAC, 60 Hz	.8
M46-111	110 VAC, 60 Hz	.8
M46-112	230 VAC, 50/60 Hz	3/8 amps
M46-113	110 VAC, 50/60 Hz	.8
M46-114	100 VAC, 50/60 Hz	.8

1.5 TERMINAL AND PRINTER DEVICE OPTION

1.5.1 Model 550 VDU Options

Model 550 VDU options are set by various switches located under a sliding cover at the front of the display at the rear of the keyboard. (See Figures 1-3 through 1-5). The terminal baud rate is adjustable for the following rates: 110, 200, 300, 600, 1200, 1800, 2400, 3600, 4800, 7200, and 9600 baud. The rate utilized is set using the switch marked "Baud Rate". The selection of the terminal mode is made with the nine switches as described in Table 1-1.

1030

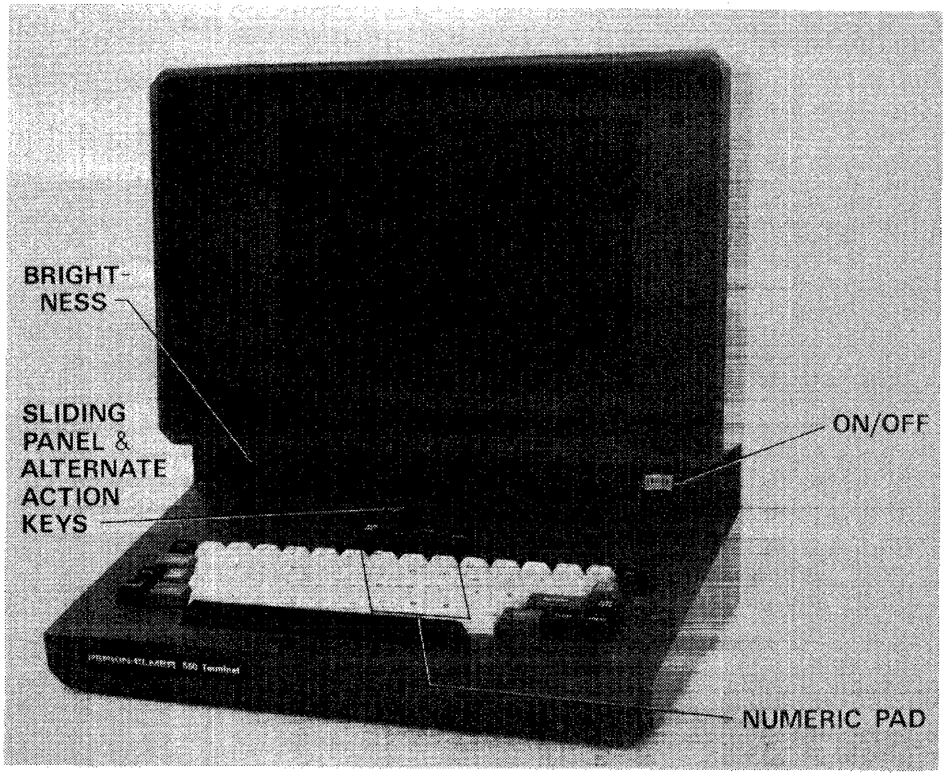


Figure 1-3 Model 550 VDU (Front View)

1031

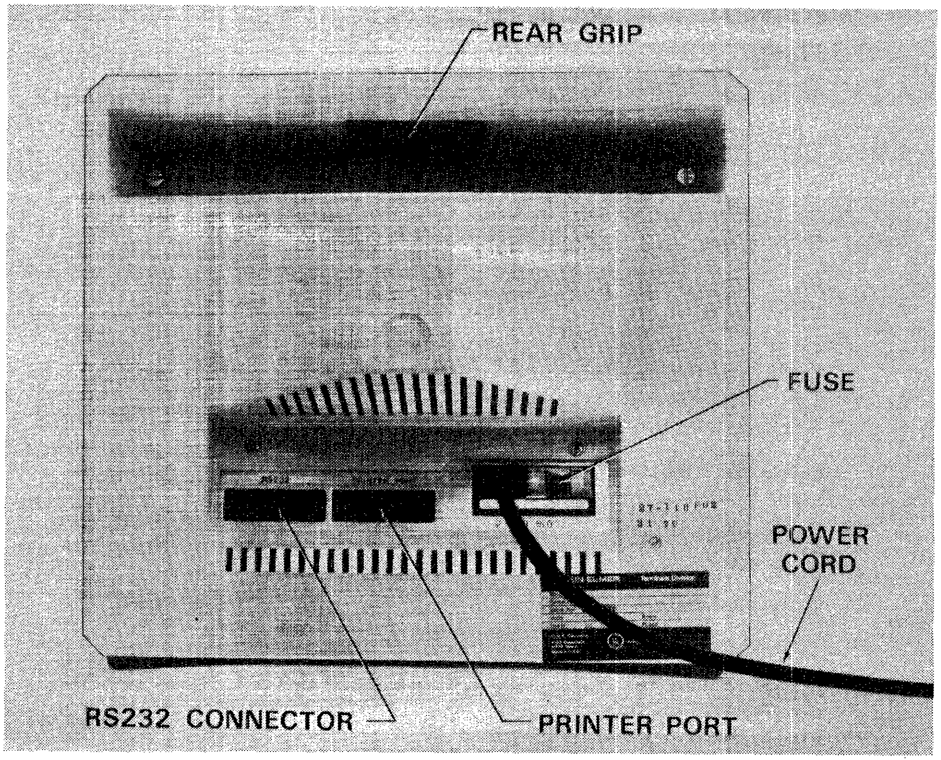


Figure 1-4 Model 550 VDU (Rear View)

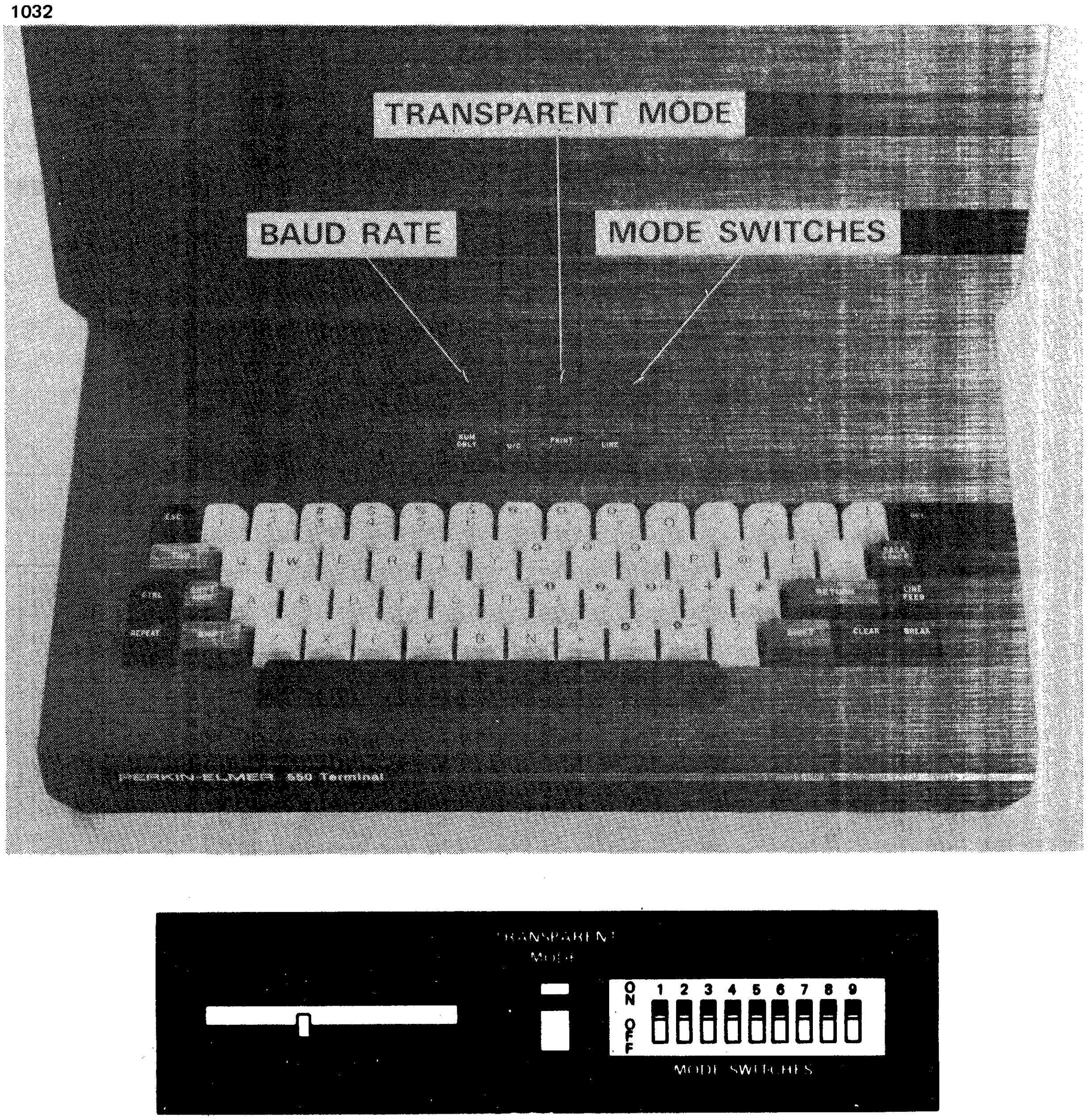


Figure 1-5 Model 550 VDU

TABLE 1-1 SELECTION SWITCHES

Parity Switch	1	2	Mode Switch	3	4	5	*6	*7	8	9
Odd	OFF	OFF	ON	1-stop	50Hz	INV. video	+12V	-12V	HDX	DTR high
Even	OFF	ON								
Mark	ON	OFF	OFF	2-stop	60Hz	STD.	open	open	FDX	Line Switch Controlled DTR
Space	ON	ON								

*Used with the current loop option

1.5.2 Model 655

The thermal printer options are set by various switches under a fabric cover located in the paper roll well. See Figures 1-6, 1-7, and 1-8.

1.5.2.1 Baud Rate

This switch is selected as shown in Figure 1-6. Note that the rate selected must match the baud rate utilized on the Model 550 VDU communication line.

1033

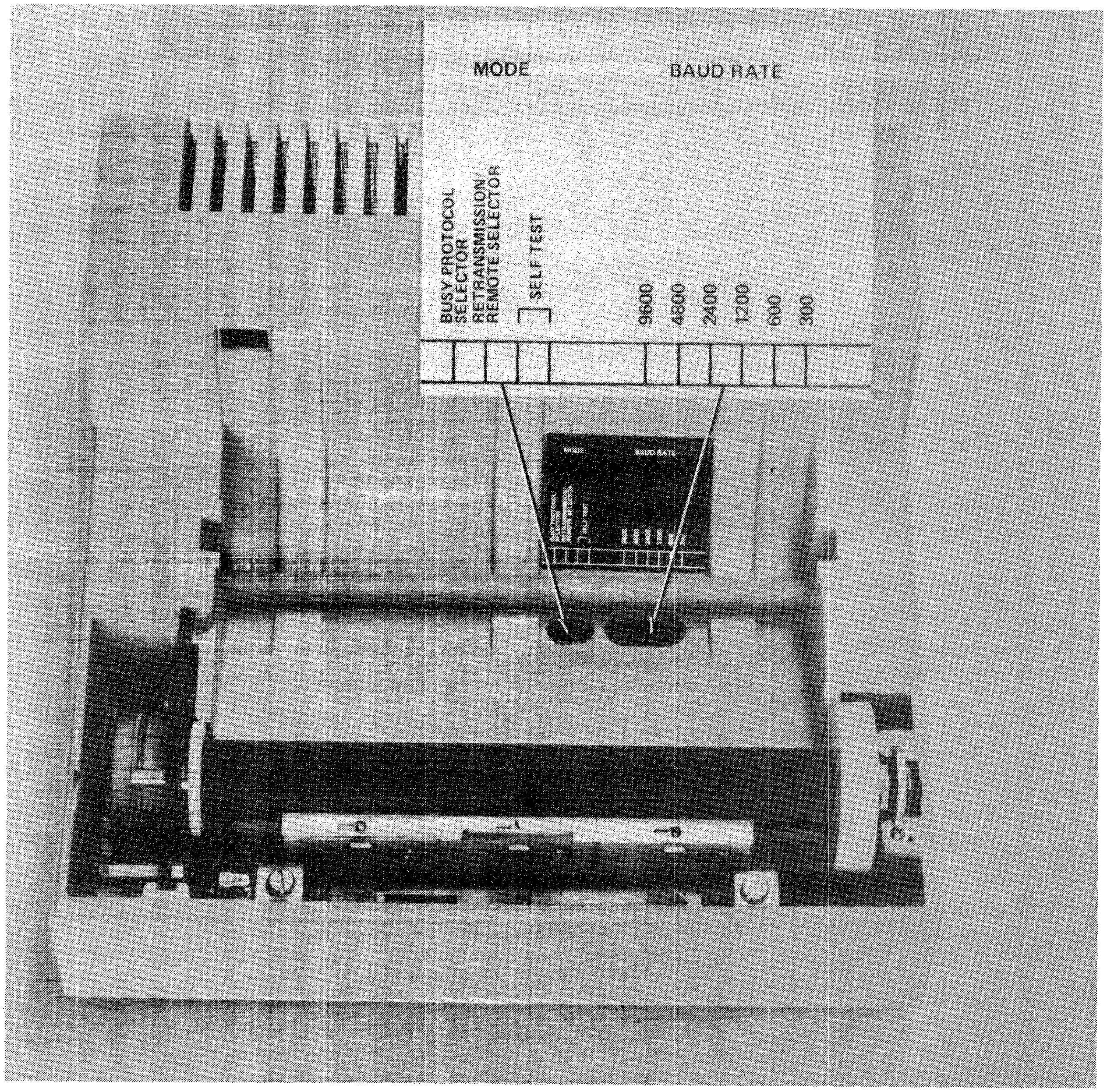


Figure 1-6 Model 655 Option Switch Locator

1034

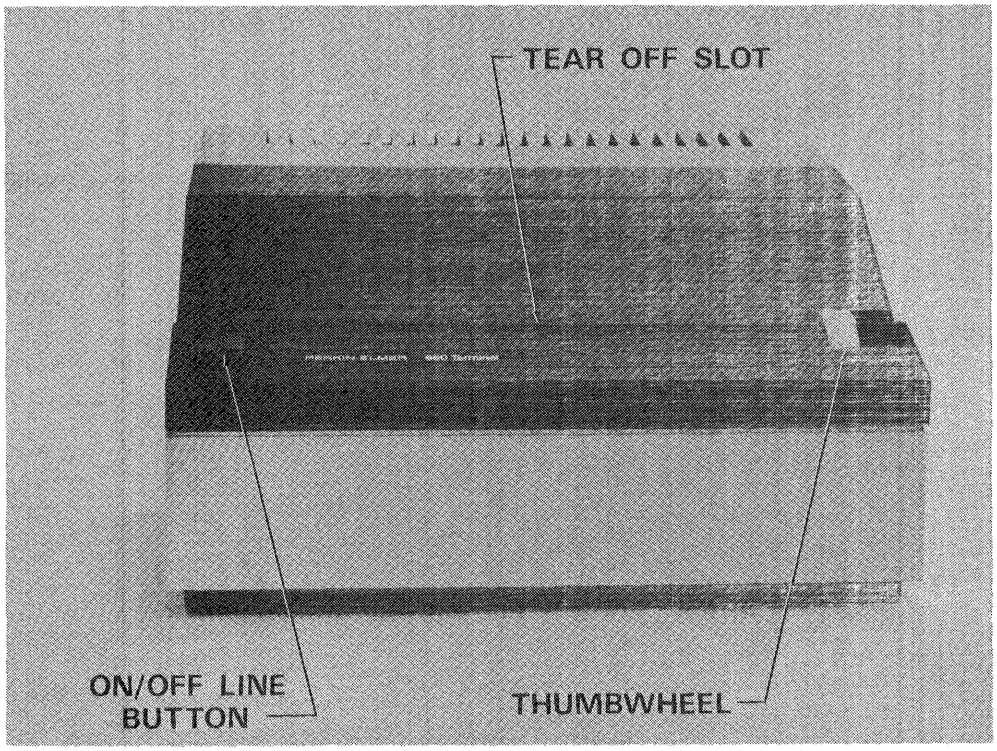


Figure 1-7 Model 655 (Front View)

1035

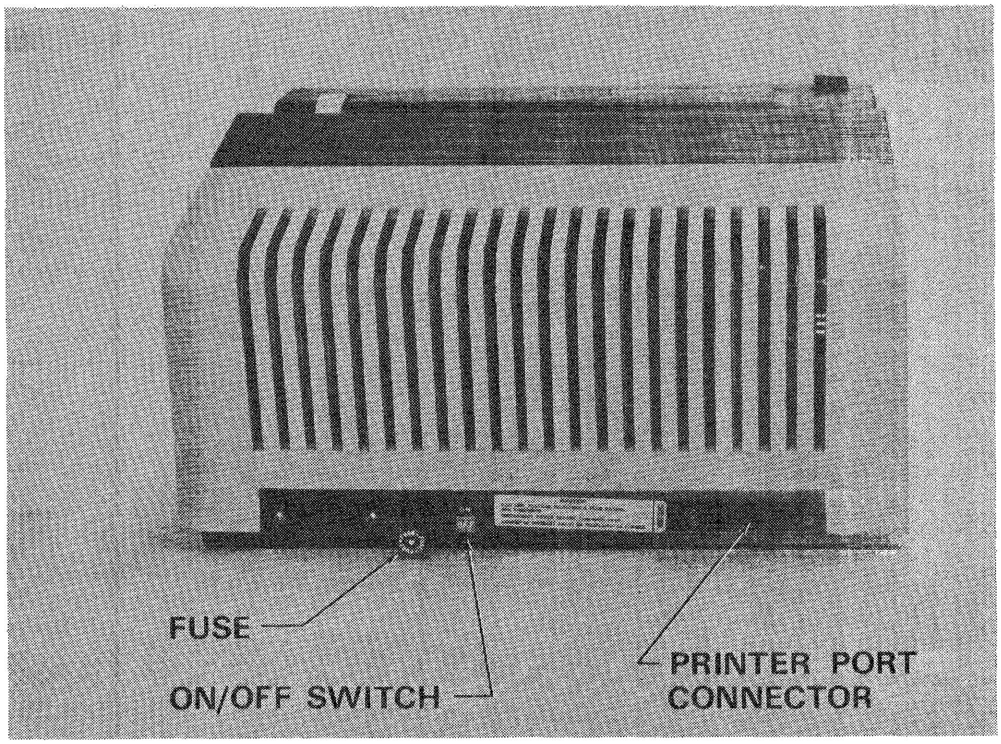


Figure 1-8 Model 655 (Rear View)

1.5.2.2 Mode Selection

Function	Switch Number	1	2	3	4
	Retransmission				
	DC4 = Busy DC2 = Busy to Ready	ON	OFF	OFF	OFF
	DC3 = Busy DC1 = Busy to Ready	OFF	OFF	OFF	OFF
	Remote Print				
	DC4 = Busy DC2 = Busy to Ready	ON	ON	OFF	OFF
	DC3 = Busy DC1 = Busy to Ready	OFF	ON	OFF	OFF
	Self Test	OFF	OFF	ON	ON

The mode used by Perkin-Elmer software is the retransmission mode using the DC2/DC4 ASCII characters.

1.6 STANDARD MODEL 550 AND MODEL 655 MODE SELECTIONS

1.6.1 Model 550 VDU

1.6.1.1 Baud Rate

Local - 9600 baud
Remote - configuration dependent

1.6.1.2 Mode Selection

Switch Number		
1 2	User Selectable	
3	User Selectable - always OFF for current loop	
4	Determined by frequency of AC power source	
5	User Selectable	
6	RS-232	OFF
	Current Loop Option	ON
7	RS-232	OFF
	Current Loop Option	ON
8	Full Duplex Standard	
9	Always OFF	

1.6.2 Model 655

1.6.2.1 Baud Rate

Set to match the baud rate setting of the Model 550 VDU.

1.6.2.2 Mode Selector

Switch position	1	2	3	4
Standard setting	ON	OFF	OFF	OFF

1.7 INTERFACE STRAPPING AND ADJUSTMENTS

1.7.1 PASLA (RS-232)

1.7.1.1 Straps

Remove	Hd	to	G4	
	G5	to	7	
Add	7	to	8	
Remove	Control Options			
	CF	to	G0	(Carrier)
	CC	to	G2	(Data Set Ready)
	CB	to	GB	(Clear to Send)

1.7.1.2 Baud Rate

Adjust the interface for the desired baud rate. For the correct procedure refer to the Programmable Asynchronous Single Line Adapter Manual (PASLA), Publication Number 29-301.

1.7.2 Current Loop

Adjust the current loop interface for the desired baud rate. (Refer to the Current Loop Interface Maintenance Manual, Publication Number 29-444.)

1.7.3 2/8 Line COMM MUX (RS-232)

Refer to Publication Number 29-650.

1. Select RS-232 Strapping.
2. Select and adjust the interface for the appropriate baud rate.
3. Select full-duplex.

CHAPTER 2
MODEL 550/MODEL 655 PROGRAMMING GUIDE

2.1 INTRODUCTION

This chapter contains a description of the Model 550 VDU, and the information necessary to program the terminal when interfaced to a Perkin-Elmer processor. The terminal interfaces to the multiplexor bus through the RS-232, current loop, and micro-I/O bus interfaces.

2.2 CONFIGURATION

The Model 550 VDU can be used with 16-bit or 32-bit processors, and is interfaced via the PASLA/Communication Multiplexor (RS-232); the current loop interface; or the micro-I/O bus on a SERIES SIXTEEN processor.

2.3 OPERATING PROCEDURES

Power

Before connecting the Model 550 VDU to the AC line, the power switch, located on the front of the terminal, must be in the OFF position.

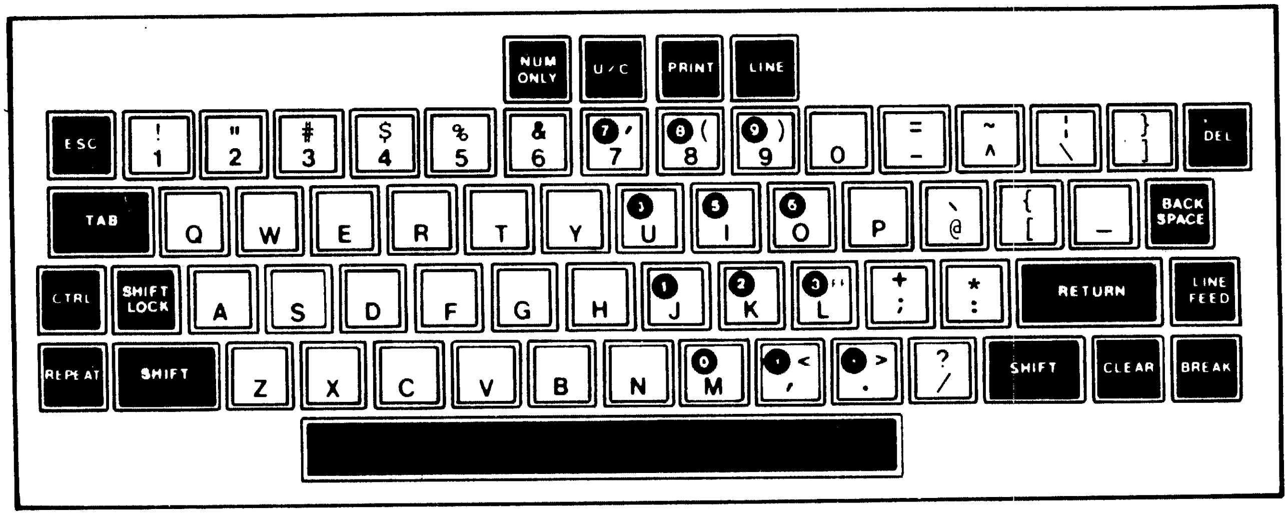
Control Panel

The Model 550 has a control panel (see Figure 1-3) located at the top of the keyboard under a slide cover. The switches must be set correctly before beginning transmission. (Refer to Section 1.5 for further details.)

Keyboard

The Model 550 VDU uses the keyboard layout shown in Figure 2-1. When a printable key is depressed, the corresponding character is printed on the terminal and the cursor moves right one position.

0582



NOTE: EARLY UNITS MAY NOT HAVE IDENTICAL KEYBOARD LAYOUT.

Figure 2-1 Model 550 Keyboard

Character Set

The Model 550 VDU can display all 128 characters of the ASCII character set. When in the upper case only mode, all alphabetic characters, entered via the keyboard or received by the terminal, are automatically converted to upper case. Numeric and special characters are not affected. The position of the SHIFT and SHIFT LOCK keys has no effect on this function. Refer to Table 2-1 for the ASCII code generated for each character.

TABLE 2-1 ASCII CHARACTER CODES

1036

					*** CONTROL CODES		* **					
BITS				b7 b6 b5	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
b4 1	b3 1	b2 1	b1 1	COL→ ROW↓	0	1	2	3	4	5	6	7
0	0	0	0	0	NUL	DLE	SP	0	@	P	\	p
0	0	0	1	1	SOH	DC1	!	1	A	Q	a	q
0	0	1	0	2	STX	DC2	"	2	B	R	b	r
0	0	1	1	3	ETX	DC3	#	3	C	S	c	s
0	1	0	0	4	EOT	DC4	$	4	D	T	d	t
0	1	0	1	5	ENQ	NAK	%	5	E	U	e	u
0	1	1	0	6	ACK	SYN	&	6	F	V	f	v
0	1	1	1	7	BEL	ETB	/	7	G	W	g	w
1	0	0	0	8	BS	CAN	(	8	H	X	h	x
1	0	0	1	9	HT	EM	)	9	I	Y	i	y
1	0	1	0	10	LF	SUB	*	:	J	Z	j	z
1	0	1	1	11	VT	ESC	+	;	K	[	k	{
1	1	0	0	12	FF	FS	,	<	L	\	l	¦
1	1	0	1	13	CR	GS	-	=	M	]	m	}
1	1	1	0	14	SO	RS	.	>	N	⌒	n	~
1	1	1	1	15	SI	US	/	?	O	—	o	DEL

* Standard 96-character ASCII set

** 64-character ASCII set displayed when the U/C key is enabled. (DEL is a legal character in this mode and is stored and displayed when preceded by an escape character sequence.)

*** In transparent mode, all control codes are displayed. In normal display mode, control characters are not displayed. DEL is displayed as a quadrangle with alternate dots.

The keyboard includes keys associated with alphabetic, numeric, and special characters. The following keys are also included:

BACK SPACE (BS) — Causes the cursor to be positioned to the left.

BRK (Break) — Causes the transmission of a spacing signal (line break) that can serve to interrupt the host processor when the terminal is in the on-line mode, and the processor interrupts are enabled.

RETURN (Carriage Return) — Causes the cursor to be positioned at the first print position of the same line.

CTRL (Control) — Causes the respective characters from columns 0 and 1 in Table 2-1 to be generated, when depressed in conjunction with keys from columns 4 and 5 of Table 2-1. This is also used to perform terminal housekeeping functions, i.e., CLEAR.

DEL (Delete) — Causes the transmission of a X'7F' with appropriate parity primarily used to erase or obliterate a previously transmitted character as dictated by the software used.

CTRL and CLEAR — Causes the terminal electronics to assume the reset condition when this key is depressed momentarily. The function in progress is terminated. No character is generated by this key. The communication switches are read and tab stops are set to every eighth position and the screen is cleared.

ESC — Causes the transmission of the ESC character normally used to provide a prefix for software interpretation of a suffix or escape sequence.

ESC SEQUENCES — If ESC and then an escape sequence key are depressed, the escape sequence is operational on the terminal. See Table 2-13. If ESC and an undefined (non-escape sequence) key, including non-displayable keys, are depressed sequentially, no escape sequence occurs and the undefined key is displayed on the terminal screen.

LINE

Causes the terminal to be put in the on-line mode, when depressed (down). When depressed again, the switch is restored to the off-line position (up). The terminal is then logically disconnected from the interface, and the Device Unavailable (DU) line is activated. Data transmitted to the interface from the keyboard is printed on the terminal screen. No character is generated by this key.

NOTE

> If terminal option switch 9 is on, DTR is always on. If terminal option switch 9 is off, DTR follows the line switch.

LF (Line Feed)

Causes the cursor to be positioned to the next line.

NUM (Numeric) only

This alternate action switch, when depressed, configures the keyboard as a numeric pad. The pad is identified by white on black circled designators and forms a calculator format. The ASCII characters that are transmitted are 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, comma, and period. In addition, the following keys generate their respective ASCII codes: Escape, Return, Line Feed, Back Space, Delete, Tab, Plus, Minus, and Slash.

PRINT

When this key is depressed, the data from the communications line is sent to both the screen and the print device (Model 655).

REPEAT

When depressed in conjunction with a printable character key, causes the particular key (character) to be repeated at a rate of 12 characters/second at 60 Hz or until either key is released.

SHIFT

Causes selection of the upper case character set, for nonalphabetic keys, when depressed. When released, the lower case character set is selected. For alphabetic keys, the upper case character set is selected. No character is generated by this key.

CTRL and SHIFT

When using the CTRL and SHIFT keys, the CTRL key takes precedence. Character codes from columns 4 through 7 of the ASCII Character Codes (see Table 2-1) are translated to the codes in columns 0 and 1, respectively.

Use of the keys listed in columns 2 and 3 (see Table 2-1), in conjunction with the CTRL key, is an illegal function (CTRL key is ignored).

SHIFT LOCK

Causes selection of the upper case character set, when depressed. When depressed again, restores the key to the lower case position (up). (See SHIFT.) No character is generated by this key.

TAB

Causes the cursor to be positioned at the next sequential print position selected as a tab stop. (Set at every eighth column, beginning with column 1, from initialize terminal.)

U/C (Upper Case only)

This switch, when in the UP position, allows all printable characters to be received or transmitted. When in the U/C ONLY position, lower case alphabetic characters are automatically converted to uppercase when received or transmitted. Special characters, numeric characters, and control codes are not affected.

2.4 CURRENT LOOP INTERFACE (CLI) PROGRAMMING INFORMATION

2.4.1 Data Format

The Model 550 VDU transfers data at 110, 200, 300, 600, 1200, 1800, or 2400 baud on a current loop interface. Each character is composed of 1 start bit (space or zero), 8 information bits including the parity bit as selected, and 1 or 2 stop bits (mark or space), for a total of 10 or 11 bits per character. Figure 2-2 shows a sample character format.

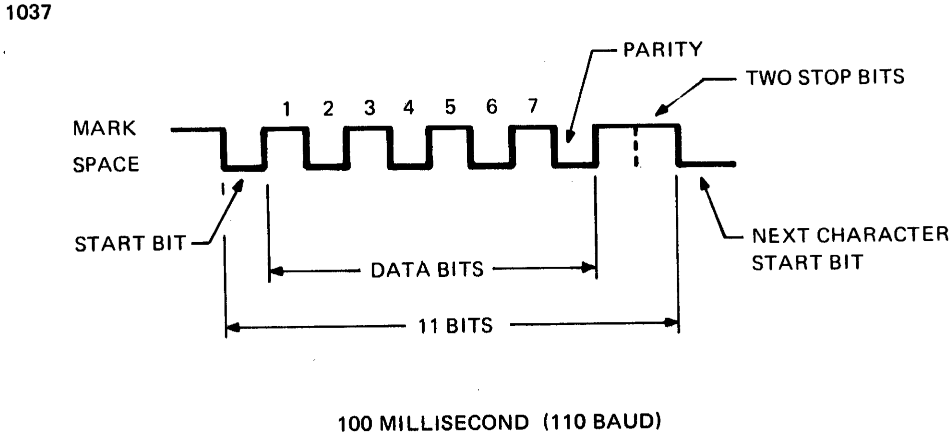


Figure 2-2 Character Format

2.4.2 Programming Instructions Using Current Loop Interface (CLI)

The following processor Input/Output (I/O) instructions are used to control and communicate with the Model 550 VDU through the appropriate CLI:

1. Sense Status (SS or SSR) - This instruction is used to examine the status bits of the interface.

2. Output Command (OC or OCR) - This instruction is used to set the interface to the receive or transmit mode, to select the Block/Unblock mode, or to select the interface interrupt ability.

3. Write Data (WD or WDR) - This instruction is used to output a character to the Model 550 VDU.

4. Read Data (RD or RDR) - This instruction is used to input a character byte from the Model 550 VDU.

5. Acknowledge Interrupt (ACK or ACKR) - This instruction (applicable to 16-bit processors only) is used to service interrupts. Execution of this instruction returns the interrupting device address in the register specified by R1, and returns the status of the interrupt device in the second operand. Acknowledgement of interrupts is automatic for 32-bit processors.

2.4.3 Status and Command Bytes

Status and command bytes for the CLI are shown in Table 2-2.

TABLE 2-2 STATUS AND COMMAND BYTE DEFINITIONS

BIT	0	1	2	3	4	5	6	7
STATUS	OV	-	BRK	-	BSY	EX	-	DU
COMMAND	DISABLE	ENABLE	UNBLOCK	BLOCK	WRITE	READ	-	-

$\underbrace{\text{DISABLE} \quad \text{ENABLE}}_{\text{DISARM}}$

Example: If the command byte X'64' (0110 0100) is issued to the interface, the resulting state of the device is:

Disable	-	0	-	OFF
Enable	-	1	-	ON-interrupts enabled
Unblock	-	1	-	ON-echo keys to printer
Block	-	0	-	OFF
Write	-	0	-	OFF
Read	-	1	-	ON-Send data to processor

Status Bytes

OV (Overflow)

When Model 550 is in the read mode, OV is set if the character previously received is not read before the present character is assembled. The previously received character is lost when the character causing the overflow condition is assembled. For example, double character buffering permits a character to be transmitted in 67 milliseconds at 110 baud. Overflow (OV) is reset with an Output Command (OC), Read Data (RD), or upon initialization. In the write mode, OV is forced reset.

BRK (Line Break)

When the serial data from Model 550 is a zero (space) for longer than one character time, or if the first stop bit is missing, the BRK bit is set. If the BREAK key is depressed while in the read mode, the character is assembled (all zeros). The BRK bit is set after one character time when BSY resets, and an interrupt occurs, if enabled. The BRK status bit remains set following depression of the Break key on the Model 550 keyboard.

When Model 550 is in the write mode and the Line Break key is depressed, BSY is set, but BRK remains reset. After one character time, and while the line remains spacing, BRK is set and BSY is reset.

BSY (Busy)

When it is not possible to transfer a character through the interface, BSY is set. In the read mode, BSY is normally set and is reset when a character has been received and assembled by the interface.

In the write mode, BSY is normally reset and is set while the interface is transferring a character output by the processor. When BSY resets, the character transfer is complete and the interface is ready for the next character. The transition of BSY from set to reset generates an interrupt, if enabled.

EX (Examine)

This bit is set if either or both of the status bits OV and BRK are set, or if the interface does not respond to the device address sent.

DU (Device Unavailable)

DU is set whenever the power switch on the Model 550 is in the OFF position, or the Line key is not depressed (terminal is in local mode). (If switch 9, the option switch, is on, then the terminal is always on line.)

Command Bytes

ENABLE When this bit is set and DISABLE is reset, interrupts are enabled.

DISABLE When this bit is set and ENABLE is reset, interrupts are not enabled, and are queued by the interface.

DISARM When both ENABLE and DISABLE are set, interrupts are not enabled, and are not queued by the interface. All pending interrupts are cleared.

UNBLOCK When this bit is set, the characters read from the terminal are echoed back to the Model 550 VDU, in addition to being assembled at the interface.

BLOCK When this bit is set, characters read from the terminal are not echoed back to the Model 550 screen, but are assembled at the interface; action caused by the control codes (DC1 TAB, etc.) received from the terminal are prevented.

WRITE When this bit is set, the interface is put in the write mode, allowing data to be transferred from the processor to the Model 550 Terminal.

READ When this bit is set, the interface is put in the read mode, allowing data to be transferred to the processor from the Model 550 VDU. When making the transition from write mode to read mode while transferring a character, the hardware waits until the character transfer is complete before changing to the read mode. Busy (BSY) remains set during this time and resets when a character has been assembled at the interface in the read mode.

NOTE

> If the transition is made from the write mode to the read mode while a character transfer is in progress (BSY is set in the write mode), BSY does not reset when the character transfer is complete; that is, no interrupt is generated or queued when the transfer is complete, even if enabled at the interface.
>
> When the transition is made from the read mode to the write mode, an interrupt is generated as BSY resets (if enabled), indicating that the interface is ready to transfer a character.

2.4.4 Programming Notes

By programming convention, a Carriage Return (CR) is always followed by a Line Feed (LF) at the end of a print line.

Keyboard

If the keyboard is operated while the processor is transferring data to the Model 550, the output data may be garbled or lost. In this situation, the conflict in the data transfers can be detected in the program by the response of the V flag in the processor condition code following a Write Data (WD or WDR) instruction. If the V flag is set following a Write Data instruction, the write operation was not successful.

2.4.5 Initialization

Interface Initialization

When the initialize switch on the processor display panel is depressed (or during the power failure/restore sequence), the interface is placed in the read mode and interrupts are disarmed.

Terminal Power Fail/Restore

As the result of power failure and restore, data in the Model 550 VDU buffers is lost, and the terminal assumes the reset condition; the cursor resets to the home position and the buffer memory is cleared to spaces.

2.4.6 Interrupts

When processor and interface interrupts are enabled, the interface generates an interrupt when:

1. BSY goes from set to reset.
2. The transition is made from read mode to write mode. (BSY goes from set to reset.)
3. BRK goes from reset to set.

NOTE

> To force an interrupt to initiate a write operation, the interface may be placed in the read mode before issuing the Output command to enter the write mode. This ensures an interrupt on the read mode to write mode transition of BSY. An alternate method is to execute a Simulate Interrupt (SINT) instruction while in the write mode. By convention, the interface is always left in the read mode.

2.4.7 Device Number

The preferred device address for the CLI is X'02'. For a chart of preferred addresses, see the 16-Bit Processor User's Manual or the 32-Bit Series Reference Manual, Publication Numbers 29-509 and 29-365, respectively.

2.4.8 Status Monitoring I/O

This I/O programming technique uses a program loop that interrogates the status of a device until the device is ready for data transmission. (See 32-Bit Series Reference Manual, Publication Number 29-365, for general programming procedures.)

2.4.9 Interrupt I/O

16-Bit Processors (See Appendices for Examples)

When using immediate interrupts on a 16-bit processor, the appropriate Interrupt Service Pointer Table must be set up to accommodate a PSW swap. The address of the interrupt service routine is placed in the new PSW location so that, when the interrupt occurs, control is transferred to the interrupt service routine. This routine must begin with an Acknowledge Interrupt (ACK or ACKR) instruction, which acknowledges the interrupt and obtains the interrupting device's address and status.

32-Bit Processors (See Appendices for Programming Examples)

Interrupts are handled via the Interrupt Service Pointer Table. Acknowledgement of the interrupt is automatic. (See 32-Bit Series Reference Manual, Publication Number 29-365, for general programming procedures.)

2.4.10 Auto Driver Channel I/O (32-Bit)

This I/O programming technique is applicable to 32-bit processors only. The Auto Driver Channel eliminates the need to process every interrupt. Operation of the Auto Driver Channel is controlled by the contents of a Channel Command Block (CCB). (See 32-Bit Series Reference Manual, Publication Number 29-365, for general programming procedures.)

2.4.11 Automatic I/O Channel (16-Bit)

Refer to the 16-Bit Processor User's Manual, Publication Number 29-509, for information on the 16-bit auto driver I/O.

2.5 FROGRAMMING ON A PASLA

2.5.1 Data Format

The PASLA may be programmed to accommodate a variety of character formats and baud rates. (See the PASLA Programming Manual, Publication Number 29-446.)

Half-duplex or full-duplex operation is switch-selectable from the front panel of the Model 550 to accommodate the corresponding strap options of the PASLA interface.

Data transfer rates of 110, 200, 300, 600, 1200, 1800, 2400, 3600, 4800, 7200, or 9600 baud are switch-selectable from the front panel of the Model 550. The baud rate selected must equal the baud rate of the interface. At 300 baud, each character is composed of one start bit (space or zero), seven information bits, a parity bit (even, odd, or space), and one stop bit (mark or one), for a total of 10 bits per character. (See Figure 2-3.)

At 110 baud, a second stop bit is transferred, for a total of 11 bits per character. Figure 2-3 shows character format for a PASLA.

1038

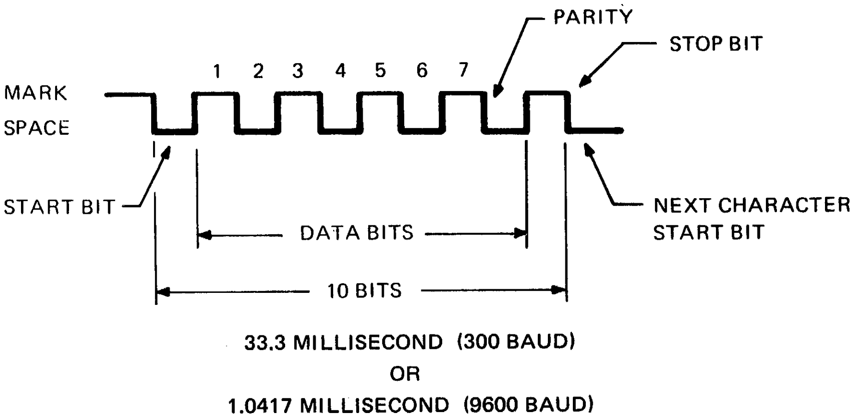


Figure 2-3 Character Format

NOTES

1. The PASLA interface may be programmed for seven data bits plus parity, or for eight data bits with no parity. When programmed for eight data bits, the parity bit transmitted by the terminal is included as part of the character assembled by the interface.

2. When the parity type selected is none, a space (zero) is transmitted by the PASLA as the parity bit for each character.

2.5.2 Programming Instructions

The following processor I/O instructions are used to control and communicate with the Model 550 VDU:

1. Sense Status (SS or SSR) - This instruction is used to determine whether the terminal is ready to transfer data, and to determine whether character transfers are complete.

2. Output Command (OC or OCR) - This instruction is used to set the interface to the receive or transmit mode and to select character format. Two command bytes are required to perform these functions with the PASLA: Command 1 and Command 2. Command 2 is required only once, unless the PASLA interface is initialized, or unless a character format change is required. PASLA Command 2 may be issued to either the receive or transmit address. (See Status and Command Bytes, Section 2.5.3, for more information.)

3. Write Data (WD or WDR) - This instruction is used to output a character byte to the Model 550 VDU.

4. Read Data (RD or RDR) - This instruction is used to read a character byte from the Model 550 VDU.

5. Acknowledge Interrupt (ACK or ACKR) - This instruction (applicable to 16-bit processors only) is used to service interrupts. Execution of this instruction returns the address in the register specified by R1 and returns the status of the interrupting device to the second operand.

2.5.3 Status and Command Bytes

Table 2-3 contains the PASLA status and command byte data.

TABLE 2-3 PASLA STATUS AND COMMAND BYTE DATA

1047

INSTRUCTION \ BIT NUMBER		0	1	2	3	4	5	6	7
PASLA STATUS		OV	PF or $\overline{\text{CL2S}}$	FR ERR	RCR	BSY	EX	CARR OFF	RING
PASLA COMMAND 1	RCV	DIS	EN	DTR	ECHO-PLEX	RCT or DTB	TRANS LB	WRT or RD	1
	SND	DIS	EN						
PASLA COMMAND 2		X	CLK	DATA BITS		STOP BIT	PARITY		0

X = UNUSED, MUST BE ZERO FOR PASLA

2.5.4 PASLA Status

OV* The Overflow (OV) status bit is set, in the receive side, if the previously received character is not read before the present character is assembled. The previously received character is lost when the character causing the OV condition is assembled. Double-character buffering in the PASLA permits a full-character grab-time. The OV status bit is reset at the next end-of-character, only if the failure condition has been cleared by a Read Data instruction. In the read mode, when an OV occurs, the BSY status bit is reset and a Read Data instruction must be issued to set the BSY bit. OV is disabled in the write mode.

* These status bits are set in the read side at end-of-character time, when BSY is reset. Since resetting BSY causes an interrupt (if enabled), these bits do not generate individual interrupts. A Read Data instruction must be issued to set the BSY bit.

PF* The PF status bit is set, in the read side, when the received parity disagrees with the programmed parity. If parity is not selected via an output command, this bit remains reset. Once set, the PF status bit remains set until a character with correct parity is assembled. PF is disabled in the write side.

$\overline{\text{CL2S}}$ The Clear to Send ($\overline{\text{CL2S}}$) status bit is ignored.

FR ERR* The Framing Error status bit is set to indicate that the received character does not have one or more stop bits; that is, the line is in the space state instead of the mark state at stop bit time. If the character has two stop bits, only the first is tested, and character assembly terminates after the first stop bit. A zero character can signify the beginning of a line break sequence. In the case of a line break (prolonged space), only the first character is assembled. Subsequent characters are not assembled until a mark to space transition is received. Note that because of this characteristic, a line break decision must be based on a single zero character with framing error. Once set, this bit remains set until a space to mark transition is received.

RCR The RCR status bit is not used with the Model 550 VDU.

BSY=1 The Busy (BSY) bit is set when:

1. Data Set Ready (DSR) from the terminal is off (EX=1).

2. A character is not yet assembled in the read mode.

3. The interface has not yet transmitted the last character in the write mode.

BSY=0 The Busy (BSY) bit is reset when the interface is able to transfer data in the selected mode. An interrupt is generated, if enabled, when BSY changes from one to zero.

Data set not ready is indicated by status X'0C'.

* These status bits are set in the read side at end-of-character time, when BSY is reset. Since resetting BSY causes an interrupt (if enabled), these bits do not generate individual interrupts. A Read Data instruction must be issued to set the BSY bit.

EX EXAMINE=OV+PF+$\overline{\text{DATA SET READY}}$+FRERR. The EX status bit is disabled in Full Duplex (FDX) on the write side. Loss of data set ready cannot be detected on the write side in FDX operation. On the receive side, data set ready is indicated by Busy and Examine bits being set; other bits may also be set. This is equivalent to Device Unavailable (DU).

CARR CFF The CARR OFF status bit is forced reset by the Model 550 VDU.

RING The RING status bit is not used with the Model 550 VDU and is forced reset.

Refer to Figure 2-4 for PASLA data set communication lines.

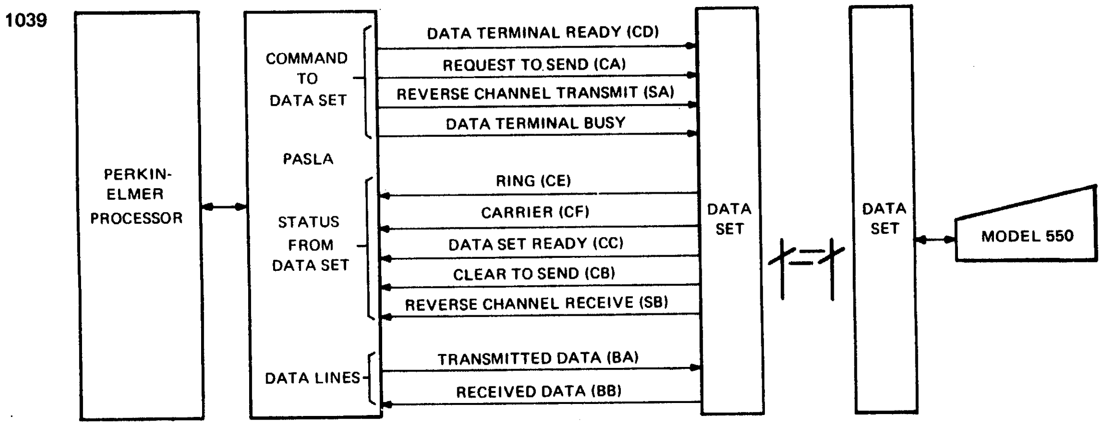


Figure 2-4 PASLA - Data Set Communication Lines (Remote Terminal Connection)

2.5.5 PASLA Command 1 Bits

The PASLA demands two 1-byte output commands. In PASLA Command 1, the DTR, ECHO-PLEX, RCT/DTB, TRANS LB, and WRT/RD bits are shared by the PASLA transmitter and receiver. The DIS/EN bits are separated for transmit and receive sides.

DIS/EN In 2-wire Half Duplex (HDX) operation, the unused side interrupts remain disarmed; the used side interrupts can be enabled or disabled. In 4-wire Full Duplex (FDX) operation, these bits must be programmed independently as follows: To change DIS/EN on the receive side, issue a command with the WRT/RD bit reset. To change DIS/EN on the transmit side, issue a command with the WRT/RD bit set. (See Table 2-4, Disable/Enable Bits.)

NOTE

See WRT/RD description for proper command sequence.

TABLE 2-4 DISABLE/ENABLE BITS

DISABLE	ENABLE	RESULTING STATE
0	0	NO CHANGE
0	1	ENABLE
1	0	DISABLE (INTERRUPT QUEUED)
1	1	COMPLEMENT (CHANGE STATE)

DTR When the Data Terminal Ready (DTR) command bit is set, DTR is turned on to permit the terminal to transfer data. When DTR is reset, the line indicator on the Model 550 VDU is turned off, and data cannot be transferred.

ECHO-PLEX When the ECHO-PLEX command bit is set, data received from the terminal is transmitted back to the terminal on the transmitted data (BA) line. The PASLA also assembles the character as in the normal data mode. When this bit is reset, characters received from the terminal are not transmitted back to the terminal printer. ECHO-PLEX is used in the read mode in HDX operation only, to provide visual verification of the data received from the terminal. This command must not be issued to the transmit side. This bit takes effect immediately; therefore, a write to read (with ECHO-PLEX) mode change requires transmitting an ASCII DEL character (X'7F') as the last character.

CAUTION

THIS BIT MUST NOT BE SET WITH THE MODEL 550 VDU IN THE FDX MODE; OTHERWISE DOUBLE CHARACTERS APPEAR ON CRT SCREEN.

RCT/DTB The RCT/DTB command bit must always be reset for the Model 550 VDU.

TRANS LB When the TRANS LB command bit is set, a continuous space is transmitted to the terminal. This condition overrides the ECHO-PLEX feature. If this command is issued while data is being transmitted, the transmitted data is mutilated.

WRT/RD The WRT/RD command bit controls Request to Send (CA) to the terminal. When WRT/RD is set, Request to Send is gated to the terminal if Data Set Ready* (DSR) is active. When WRT/RD is reset, the hardware deactivates Request to Send (CA) after the following delay: if character transfers are in progress, the hardware ensures that the last character has been transmitted; it then delays one millisecond to permit the last data bit to clear the data set before dropping Request to Send (CA) (except as noted under ECHO-PLEX). Busy (BSY) is set during this line turnaround and does not reset until a character is received. However, CL2S, CARR OFF, RING, RCR, and DSRDY may still generate interrupts, if enabled. The WRT/RD bit is gated to the data set as Request to Send. Therefore, in 4-wire Full Duplex (FDX) operation, it is essential that a command with WRT/RD reset be followed by a command with WRT/RD set to ensure that Request to Send does not deactivate.

In 2-wire, Half Duplex (HDX) operation, set WRT/RD to place the PASLA in the write mode; reset WRT/RD to place the PASLA in the read mode.

* Data Set Ready (DSR) does not appear in the status byte on the transmit side in FDX operation. The operator must rely on the receive side of the adapter for notification of the loss of DSR. Loss of DSR on the transmit side in FDX operation does not cause an interrupt. It does, however, hold Busy high, thus preventing any more end-of-character interrupts.

2.5.6 PASLA Command 2 Bits

Clock Select (CLK SEL) enables one of two baud rates. (See Table 2-5.)

TABLE 2-5 PASLA COMMAND 2 CLOCK BITS

BIT POS	1	CLOCK
	0	CLKA (Lowest Baud Rate)
	1	CLKB

DATA BITS selects the number of data bits/character (not including parity). (See Table 2-6.)

TABLE 2-6 PASLA COMMAND 2 DATA BITS

BIT POS	2	3	NO. OF DATA BITS
	0	0	5
	0	1	6
	1	0	7
	1	1	8

If fewer than eight bits are selected when a Write Data is issued in the write mode, the data must be right justified and unused bits are Don't Care. In the read mode, when a Read Data is issued, the character is presented to the processor right justified with unused bits (this includes selected parity and stop bits) forced to the zero state.

Stop Bit 0=1 Stop Bit
1=2 Stop Bits

When the line is programmed for two stop bits, the PASLA transmits both. However, the receiver samples only the first stop bit.

PARITY selects the state of the parity bit to be transmitted and received. (See Table 2-7.)

TABLE 2-7 PASLA COMMAND 2 PARITY BITS

BIT POS	5	6	PARITY
	1	0	ODD
	1	1	EVEN
	0	X	NONE

In the write mode, if parity is enabled (bit 5=1), the PASLA generates and transmits the selected parity.

In the read mode, if parity is enabled, the PASLA compares the received parity with the selected parity and generates the PF status if a discrepancy is detected.

If parity is disabled (bit 5=0), it is ignored by the hardware. When transmitting, the hardware appends a stop bit after the last data bit and, when receiving, disables the parity detection circuit.

NOTE

> The least significant bit of the command byte must be a one or a zero, as indicated, to permit the hardware to distinguish between the two commands.

2.5.7 Programming Notes

Half Duplex (HDX) Operation

When programming in Half Duplex (HDX) mode with the ECHO-PLEX command bit set, enough delay must be allowed for turning the line from write to read and vice versa. Transmission of the last character must be followed by a null character before a write to read line turnaround. This prevents mutilation of the character. (See Status and Command Bytes.)

Miscellaneous

Some test programs and all operating system programs use a back-arrow (←) to delete the last character input. To generate the equivalent code (ASCII '5F'), the underline (_) must be used.

2.5.8 Interrupts

An interrupt is generated, if enabled by the PASLA interface, for any of the reasons shown in Table 2-8.

TABLE 2-8 INTERRUPT CONDITIONS

STATUS BIT TRANSITION	HDX	FDX	
		REC	TRANS
RING → 1	X	X	
CARR OFF → 1	X (RD)	X	
CARR OFF → 0	X (RD)	X	
RCR → 1	X	X	
RCR → 0	X	X	
DSRDY → 0	X	X	
*BSY → 0	X	X	X
$\overline{\text{CL2S}}$ → 1	X (WRT)		X

* An interrupt is also generated in HDX operation when going from Read to Write mode if CL2S initially equals 0; i.e., when CL2S goes from 0 to 1, which causes a BSY interrupt.

NOTES

1. When Model 550 is connected as a local terminal, certain transitions shown in Table 2-8 are inhibited at the interface. (See Status and Command Bytes.)

2. The major difference between the remote and local configurations, shown in Figure 2-5, is that the local termination does not require the extensive hand-shaking sequences necessary for switched-line operation. Consequently, many of the standard RS-232 control and status lines are not terminated in the PASLA or the terminal for local connections.

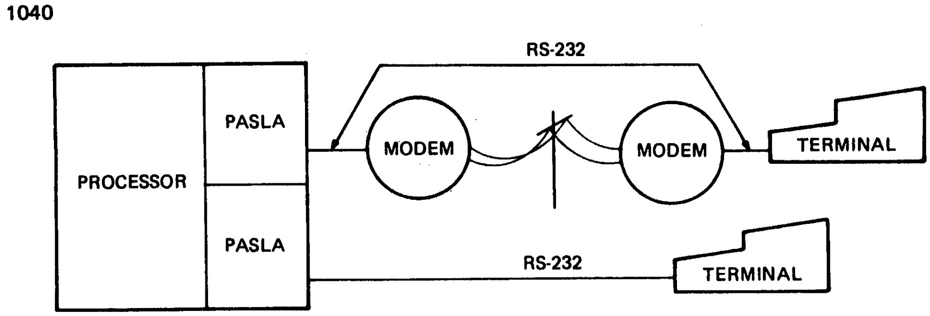


Figure 2-5 Local and Remote Terminal Connections

2.5.9 Initialization

Interface Initialization

When the initialize switch on the processor display panel is depressed (or during the power failure/restore sequence), the PASLA is placed in the disable mode. PASLA commands 1 and 2 should be issued to select format and baud rates. The state of the OV, PF, and FRERR status bits cannot be guaranteed; a Read Data (RD or RDR) instruction should be issued to ensure that Busy is set to prevent an interrupt from the PASLA as the result of initialization, before interrupts are enabled. The DTR bit is reset.

The state of command bits DTR, ECHO-PLEX, RCT, TRANS LB, and WRT/RD cannot be guaranteed when power is initially applied.

Terminal Power Fail/Restore

As the result of power failure and restore, data in the Model 550 VDU buffers is lost, and the Model 550 electronics assumes the reset condition. Every eighth print position is set as a tab stop; thus, a manual reset is necessary following a power fail/restore sequence.

Terminal Clear Switch

Momentary operation of the CLEAR and CNTL switches on the control panel causes the Model 550 electronics to assume the reset condition. The state of the PASLA interface is not affected.

PASLA Interface

VOLTAGE	BINARY	SINGLE LINE	CONTROL
+5V to +15V	0	SPACE	ON
-5V to -15V	1	MARK	OFF

2.5.10 Device Number

The PASLA may be strapped to provide a Full Duplex (FDX) interface with an even device address for the receive side and an odd device address for the transmit side. The PASLA is normally strapped for device addresses X'10' and X'11'. There is an interrupt flip-flop associated with each side. When strapped to provide a Half Duplex (HDX) interface, each side responds to either address.

2.5.11 Sample Programs

Programming examples for 16-bit and 32-bit processors are shown in Appendices A and B, respectively.

2.6 MICRO-I/O BUS PROGRAMMING INFORMATION

2.6.1 Data Format

The M51-100 serial Input/Output port is a current loop interface on the micro-I/O bus as device number X'C0'.

2.6.2 Status and Command Bytes

Tables 2-9 and 2-10 contain the serial input/output port status and ccmmand bytes.

TABLE 2-9 SERIAL INPUT/OUTPUT PORT STATUS DATA

BIT NUMBER	0	1	2	3	4	5	6	7
STATUS	OV	0	BRK	0	BSY	EX	0	DU

OV (Overflow)

The Overflow (OV) status bit indicates that one or more characters in the data stream were lost; that is, one or more new characters were received before the processor could read the present character. The Overflow indication does not set until the processor actually reads the present character. The Busy bit then remains reset until the OV bit resets. This occurs the next time the processor does a read operation. Overflow is also reset by the RESET command.

BRK (Line Break)

The Line Break (BRK) bit is set when the serial input data line is a zero (space) for longer than one character period. Specifically, BRK sets when a character is received that appears to have a Start bit, but when the first Stop bit becomes necessary, the line is zero. Break status remains active until the input data line goes to a one (mark), and the character is read by the program.

BSY (Busy)

The Busy (BSY) status bit is set when the serial input/output port cannot yet transfer a character. In the read mode, BSY is normally set. It resets when a complete character has been received and is ready to be read by the processor. In the write mode, BSY is normally reset. It sets after the processor writes a character to the interface and remains set until the interface transmits the character. Thus, when BSY is reset, the processor can transfer data with the serial input/output port. In the transition from BSY set to BSY reset, an interrupt is generated.

EX (Examine)

The Examine (EX) status bit is set when the terminal device is powered down or off-line.

DU (Device Unavailable)

The Device Unavailable (DU) status bit is set when the terminal device is powered down or off-line.

TABLE 2-10 SERIAL INPUT/OUTPUT PORT COMMAND DATA

RESET COMMAND	0	0	0	0	0	0	1	1
WRITE DISABLE	0	0	0	1	0	0	1	0
WRITE ENABLE	0	0	1	1	0	0	1	0
READ	1	0	0	1	0	0	1	0

Commands

WRITE DISABLE — An output command instruction, X'12', places the interface in the write mode with write interrupts disabled. In this mode when BSY is reset, the processor can write data to the interface.

WRITE ENABLE — An output command instruction, X'32', places the interface in the write mode with interrupts enabled. In this mode, when BSY is reset, the processor can write data to the interface.

READ — An output command instruction, X'92', places the interface in the read mode. In this mode, when BSY is reset, a character has been received from the Model 550 VDU and is available for the processor to read. Interrupts cannot be disabled in the read mode.

RESET — This output command, X'03', initializes the interface. The OV, BRK, BSY and EX status bits are reset and any pending interrupts are cleared. The RESET command should be issued only once during a program to initialize the micro-I/O bus.

CAUTION

INVALID OUTPUT COMMANDS CAUSE THE MICRO-I/O BUS TO HANG IN AN UNUSABLE STATE.

2.6.3 Programming Sequences

Programming Notes

Characters read from the keyboard are not echoed back for display. The program must perform the character echo. This is accomplished by doing a Write Data instruction immediately after reading the character. It is not necessary to switch from read mode to write mode for the character echo.

Status Monitoring I/O

This form of I/O programming uses loops to continually interrogate the device status until a specified condition is met. Processor interrupts must be disabled to use this type of I/O. (See appendices for programming examples.)

Interrupt I/O Control

When using the immediate interrupt mechanism, the appropriate Interrupt Service Pointer Table entry must be set up to accommodate a PSW swap. (See appendices for programming examples.)

2.6.4 Interrupts

In the read mode, interrupts are always enabled in the serial input/output port interface. The interface generates an interrupt for the following conditions:

1. In the read mode, when a character is present in the interface (BSY = zero).
2. In the write enable mode, if the interface can accept a character from the processor (BSY = zero).
3. EXAMINE goes active.

Interrupts pending in the interface may be cleared by:

- system initialization
- output command reset
- read data or write data instruction

The fcllowing is a list of salient serial input/output port specifications:

1. Baud rate is selectable to yield 110, 300, 600, 1200, 1800, or 2400 baud.

2. Character format is an 8-level, 11-unit code (1 start bit, 8 data bits - no parity - and 2 stop bits).

3. Data is double buffered to permit a full character "grab time".

2.6.5 Programming Instructions

When not being used for operation control, the Model 550 VDU on the serial input/output port is available to any running program as an I/O device. The following processor I/O instructions may be used to control and communicate with the Model 550 VDU.

1. Sense Status (SS or SSR) - This instruction is used to determine if the Model 550 VDU is ready to transfer data.

2. Output Command (OC or OCR) - This instruction is used to initialize the interface and to establish the read or write modes.

3. Write Data (WD or WDR) - This instruction is used to output a character to the Model 550 VDU.

4. Read Data (RD or RDR) - This instruction is used to read a character from the Model 550 VDU.

5. Acknowledge Interrupt (ACK or ACKR) - This instruction is used to service interrupt request. Execution of this instruction returns the device number in the first operand register and the device status in the second operand. The interrupt condition is not reset. The interrupt is cleared by a read operation in the read mode, a write operation in the write mode, or by a RESET command.

2.7 COMMUNICATIONS MULTIPLEXOR PROGRAMMING INFORMATION

2.7.1 Data Format

The fixed, optional, and programmable features of the 2-line and 8-line COMM MUX boards are described in the following sections.

2.7.2 Specifications

The following is a list of COMM MUX specifications:

- Baud Rates - The following groups of baud rates are provided:

Group 1:	50, 110, 1,800, 2,400
Group 2:	75, 134.5, 2,000, 3,600
Group 3:	150, 600, 4,800, 9,600
Group 4:	300, 1,200, 7,200, 19,200

Group selection is a strappable option, as described in the COMM MUX Maintenance Manual, Publication Number 29-650. Within a group, baud rate selection is under program control.

- Maximum number of lines - 2 lines or 8 lines per COMM MUX; any combination of 2-wire/4-wire.

- Character format - The following three independent variables on the character format are programmable:

Character size - 5, 6, 7, or 8 data bits
Parity - Odd, even, or none
Stop bits - One or two

- Data Set Control (Programmable)

a. Data terminal ready (CD) - Program control is provided over CD to allow for automatic call reception, disconnect, and lockout.

NOTE

Parentheses indicate RS-232C designations for indicated functions.

b. Reverse channel transmit (SA)* - Permits a supervisory signal to be transmitted over a secondary data path while simultaneously receiving data.

* Optional features in some data sets.

c. Request to send (CA) - Active to maintain the COMM MUX in the transmit mode. In 2-wire operation, the inactive state maintains the COMM MUX in the receive mode.

d. Data terminal busy* - Enables the "make busy" feature when available.

- Data set status - The following lines from the data set affect the status bits: CLEAR TO SEND (CB), CARRIER (CF), RING (CE), REVERSE CHANNEL RECEIVE (SB), and DATA SET READY (CC).

- Echoplex - A programmable feature for transmitting received data back to the data set in addition to assembling the character.

- Other features - The CCMM MUX provides a double-buffered character to permit a full character "grab-time." The hardware automatically generates and transmits the start bit in the receive mode, and it must be present for at least one-half a bit time before the character assembly begins to reduce the system's noise level.

- Method of transmission - Serial, asynchronous by character, synchronous by bit.

- Distortion:

  Transmit - The transmit data distortion is +3% per character.

  Receive - The COMM MUX adjusts the data sampling strobe with each character received and tolerates a data bit distortion of +43%. In addition, the long term transmission rate may vary by +5%.

2.7.3 Transfer Format

Asynchronous operation requires all characters to be preceded by one start bit (=0) and have at least one stop bit (=1) appended after the last data bit or the parity bit, if selected. Start and stop bits delineate characters. A typical format (110 baud) is shown in Figure 2-6.

* Optional features in some data sets.

1041

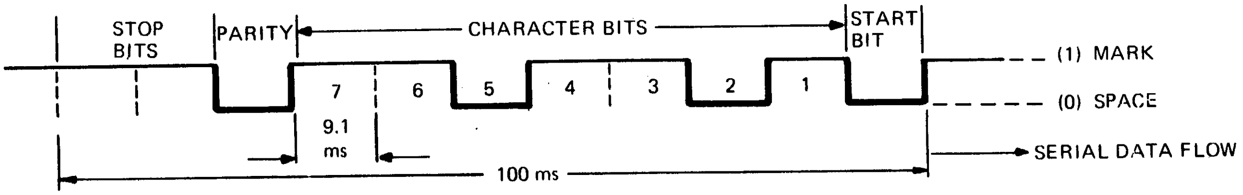


Figure 2-6 Typical ASCII Character Format

NOTE

> To send 7 data bits plus parity, 11 code elements (bits) are required. Therefore, to send 70 bits of useful information per second, the system must operate at 110 baud.

The hardware generates the single start bit; the character size/parity and number of stop bits are under program control.

2.7.4 Order of Transmission

Figure 2-7 shows the order of transmission to and from the communication line.

1042

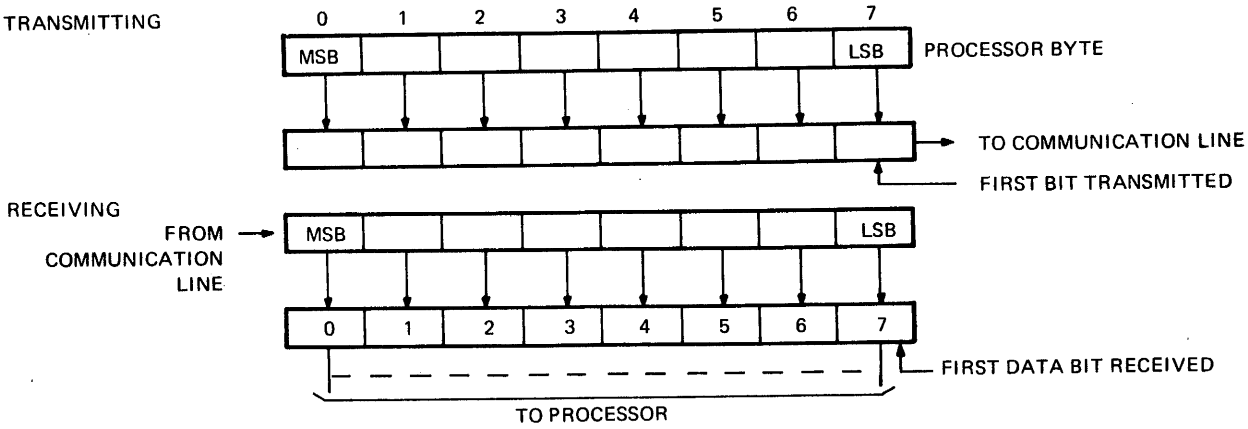


Figure 2-7 Order of Transmission for Transmitting/Receiving

NOTE

> Some terminals require the MSB to be transmitted (or received) first. For such terminals, the program should present the data with the MSB of the data in byte position 7; this may be performed through the translation option of the auto-driver channel on 32-bit processors.

2.7.5 Programming Instructions

The COMM MUX continually multiplexes signals being transferred across up to eight separate lines. Each line requires its own format characteristics while in use. Processor line drivers must be programmed differently for each type of line use. The drivers incorporate processor I/O instructions to communicate with the COMM MUX and to affect appropriate COMM MUX register changes as required for each line type. The following paragraphs describe the instructions.

2.7.6 COMM MUX Program Instructions

Sense Status (SS or SSR)

The Sense Status instruction is used to determine if character transfers are complete and correct, and to interrogate the associated data set status.

Output Command (OC or OCR)

The Output Command instruction is used to answer or disconnect calls, to set the COMM MUX in the receive or transmit mode, and to select the character format. Two command bytes are required to perform these functions.

Write Data (WD or WDR)

The Write Data instruction is used to load the output character into the COMM MUX data register.

Read Data (RD or RDR)

The Read Data instruction is used to read an assembled character into the processor.

Acknowledge Interrupt (AI or AIR)

The Acknowledge Interrupt instruction is used to service interrupts. Execution of this instruction returns the address and status of an interrupting line. This instruction is used only for the 16-bit processors.

2.7.7 Communication Instructions

The CCMM MUX accommodates communication instructions in the communication processors (Models 50, 55, and 60).

2.7.8 Auto Driver Channel (ADC)

The CCMM MUX may be used with ADC on the 32-bit processors.

2.7.9 Status and Command Bytes

Table 2-11 contains the COMM MUX status and command byte data.

TABLE 2-11 COMM MUX STATUS AND COMMAND BYTE DATA

1048

<table>
<tr><td colspan="2">BIT NUMBER</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td></tr>
<tr><td colspan="2">COMM MUX STATUS</td><td>OV</td><td>PF
or
$\overline{\text{CL2S}}$</td><td>FR
ERR</td><td>RCR</td><td>BSY</td><td>EX</td><td>CARR
OFF</td><td>RING</td></tr>
<tr><td rowspan="2">COMM MUX
COMMAND 1</td><td>RCV</td><td>DIS</td><td>EN</td><td rowspan="2">DTR</td><td rowspan="2">ECHO-
PLEX</td><td rowspan="2">RCT
or
DTB</td><td rowspan="2">TRANS
LB</td><td rowspan="2">WRT
or
RD</td><td rowspan="2">1</td></tr>
<tr><td>SND</td><td>DIS</td><td>EN</td></tr>
<tr><td colspan="2">COMM MUX COMMAND 2</td><td>CLKB</td><td>CLKA</td><td colspan="2">BIT SEL</td><td>STOP
BIT</td><td colspan="2">PARITY</td><td>0</td></tr>
</table>

2.7.10 COMM MUX Status Bits

Refer to Figure 2-8 for COMM MUX - Data Set Communication lines.

OV* The Overflow status bit is set to one if the previously received character is not read before the present character is assembled. Double-character buffering in the COMM MUX permits a full-character "grab-time". The OV status bit can be one in the receive side only. It is reset at the next end-of-character, only if the failure condition disappears (i.e., is cleared by a Read Data instruction). The character causing overflow is assembled and the previous character is lost.

* These status bits are set at end-of-character time when the BSY bit is zero. Since resetting Busy causes an interrupt (if enabled), these bits do not generate individual interrupts. At this point, a read data instruction must be issued to set the BSY bit.

PF* In the read mode, this bit is one when the received parity disagrees with the programmed parity. If parity is not selected via an output command, this bit remains zero. Once set, the PF status bit remains set until the failure condition disappears (i.e., a character with correct parity is assembled).

$\overline{\text{CL2S}}$ The lack of Clear-To-Send signifies that the modem can no longer transmit data. In the write mode, this status bit set indicates that clear-to-send (CB) is not being received from the data set. (See Figure 2-8.) This condition also forces BSY=1 on the transmit side. A transition from CL2S=0 to CL2S=1 causes an interrupt, if enabled.

FR ERR* The Framing Error status bit is set to one to indicate that the received character has no stop bit(s); that is, the line is in the space state instead of the mark state at stop bit time. If the character has two stop bits, only the first is tested, and the character assembly terminates after the first stop bit. If a framing error occurs, the character is assembled. A zero character can signify the beginning of a line break sequence. In the case of a line break (prolonged space), if the line remains spacing, only the first character is assembled. Subsequent space characters are not assembled until a mark to space transition is received. Note that because of this characteristic where the line break facility is being employed, a line break decision must be based on a single zero character with framing error. Once set, this bit remains set until the assembly of a character with a stop bit.

* These status bits are set at end-of-character time when the BSY bit is zero. Since resetting Busy causes an interrupt (if enabled), these bits do not generate individual interrupts. At this point, a read data instruction must be issued to set the BSY bit.

RCR Reverse Channel Receive (SB) is an option in some 2-wire data sets (e.g., 202C). This status bit is set if the reverse channel line from data set is on. This bit is reset if the reverse channel line from data set is off. If the data set does not have the reverse channel option, this status bit is always inactive. Either transition of this signal causes an interrupt, if enabled.

BSY If the BSY status bit is set, the following occurs:

1. Data set ready (CC) from the data set is off (EX=1).
2. Character is not assembled in read mode.
3. Clear to send (CB) is off (CL2S=1) in write mode.
4. When the interface has not yet transmitted, the last character is in the write mode.

If the busy status bit is reset, the interface can transfer data in the read/write mode. An interrupt is generated, if enabled, when the busy status bit changes from a one to a zero. When an overflow occurs in the read mode, the BSY status bit is reset to zero and a read data instruction must be issued to set the BSY bit to its correct (one) state.

EX Examine=OV+PF+DATA SET READY + FRERR. This bit is disabled in FDX on the write side. Loss of Data Set Ready (CC) cannot be detected on the write side in FDX operation. On the receive side, Data Set Ready is indicated by busy and examine being one.

CARR OFF CARR OFF is one to indicate that no valid incoming data is being received. In the received side, this bit is one to indicate that Carrier (CF) is not being received from the data set (see Figure 2-8). In the write mode, this status bit is zero when Request to Send (CA) is active. If enabled, a transition of this status bit in either direction causes an interrupt.

RING RING is one when the RING (CE) signal from the data set is active. This indicates the receiving of a call. An interrupt is generated, if enabled, when RING changes to one. In 4-wire operation, RING is always zero on the transmit (send) side. The ring status represents the present state of the equivalent data set signal. (See Figure 2-8.)

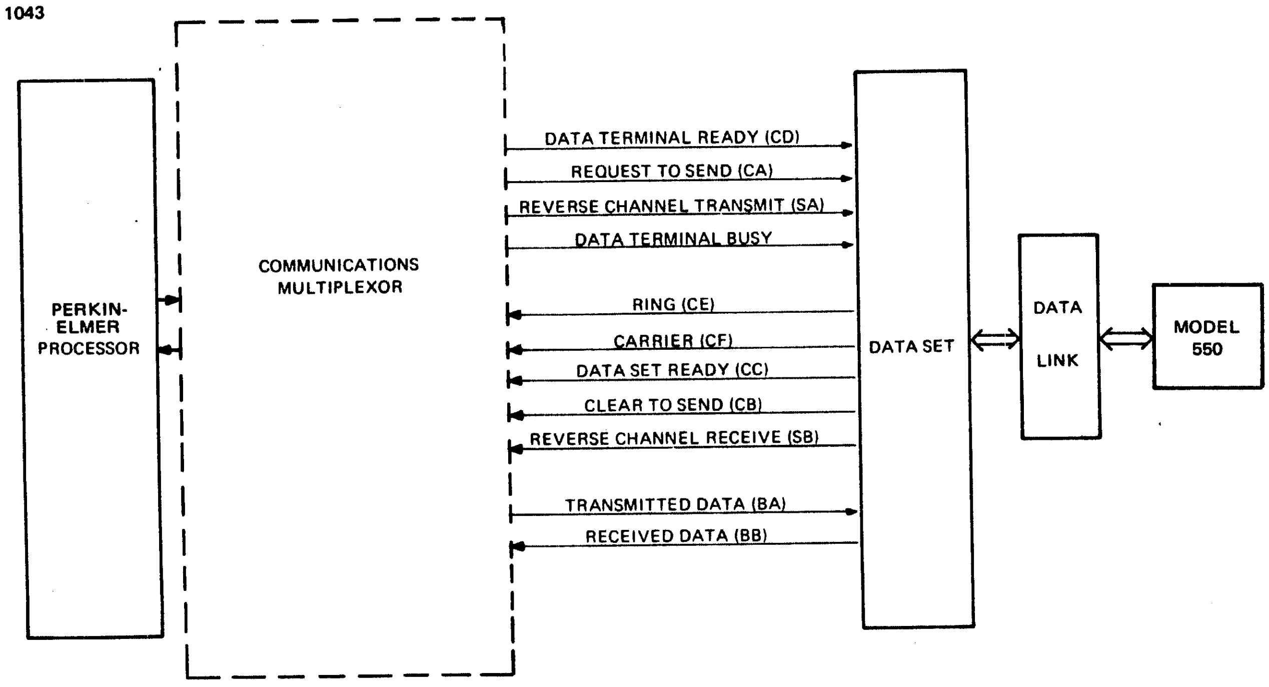


Figure 2-8 COMM MUX Data Set Communication Lines

2.7.11 COMM MUX Command 1 Instruction Bits

The COMM MUX needs two 1-byte output command instructions, Command 1 and Command 2. The Command 1 instruction byte sets up the interrupt conditions and the modem controls. In the Command 1 instruction, the DTR, ECHOPLEX, RCT/DTB, TRANS LB, and WRT/RD bits are shared by the transmitter and receiver. The EN/DIS bits, however, are separate for transmit and receive sides. Command 1 instruction bits are:

DIS,EN If the disable bit is reset and the enable bit is set, interrupts are enabled. Interrupts are queued if the disabled bit is set and the enable bit is reset. If both the disable bit and the enable bits are set, interrupts are disarmed (no interrupts are queued); but if both are reset, no change occurs.

In 2-wire operation, the unused side interrupts remain disarmed. The used side interrupts can be enabled or disabled. In 4-wire operation, these bits must be independently programmed as follows:

- To change EN/DIS on the receive side, issue a command with the WRT/RD bit = 0.

- To change EN/DIS on the transmit side, issue a command with the WRT/RD bit = 1.
- The WRT/RD bit is gated to the data set as Request to Send. Therefore, in 4-wire operation, a command with WRT/RD = 0 must be followed with a command WRT/RD = 1 to ensure that Request to Send does not deactivate.

DTR Data Terminal Ready (CD) to the data set. When this command bit is set, CD is turned on, allowing answering an incoming call automatically. This line must be on to permit the data set to enter and remain in the data mode. When this bit is reset, it does not permit automatic answering of an incoming call and causes an existing connection to disconnect if held reset for a period specified by the manufacturer of the data set. (See Figure 2-8.)

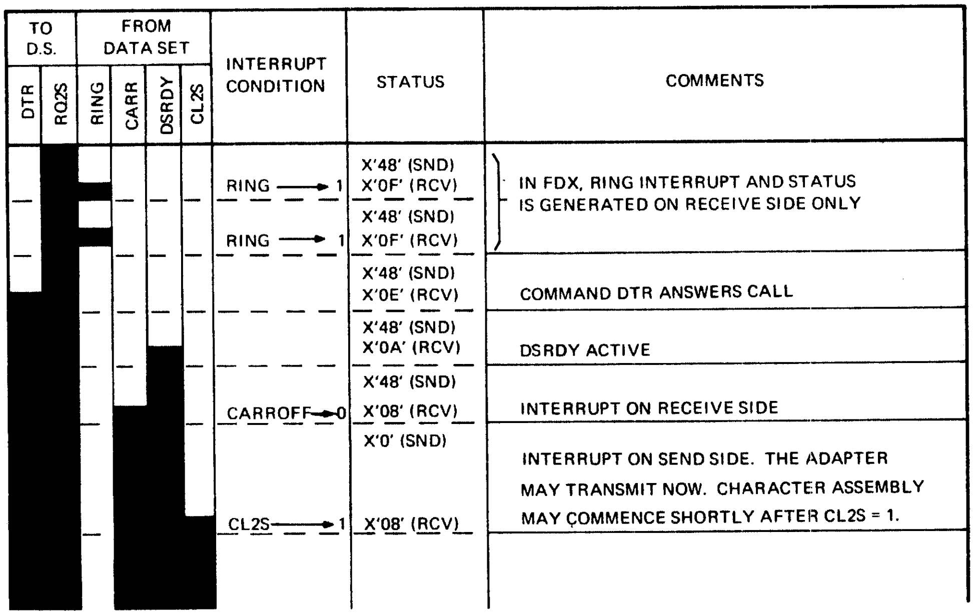
ECHO-PLEX When this bit is set, data received from the data set is transmitted back to the data set on the Transmitted Data (BA) line. (See Figure 2-8.) The COMM MUX also assembles the character as in the normal data mode. This feature is normally used for 4-wire FDX operation in the read mode tc provide visual verification at the terminal of data received by the computer. This command must not be issued to the transmit side. In the 2-wire HDX read mode, the RQ2S line is not active. If the associated data set requires RQ2S to be active, the data does not pass to the communication link. This bit takes effect immediately. Therefore, a write to read (with echo-plex) mode change requires transmitting X'FF' (an ASCII DEL character) as the last character.

RCT/DTB Not used with 550 VDU.

TRANS LB When this bit is set, a continuous space is transmitted to the data set. This condition overrides the echo-plex feature. If this command is issued while data is being transmitted, the transmitted data is mutilated.

WRT/RD This command bit controls Request to Send (CA) to the data set. (See Figure 2-8.) When this bit is set, Request to Send is gated to the data set if Data Set Ready* (CC) is active. When this bit is reset, the hardware deactivates Request to Send (CA) after the following delays: If character transfers are in progress, the hardware ensures that the last character has been transmitted. It then delays one millisecond to permit the last data bit to clear the data set before dropping Request to Send (CA) (except as noted under ECHO-PLEX). BSY is set during this line turnaround and does not reset until a character is received. However, CL2S, CARR OFF, RING, RCR, and DSRDY may still generate interrupts, if enabled. (See Figure 2-9.) In 2-wire operation, setting this bit places COMM MUX in the write mode; resetting this bit places COMM MUX in the read mode. In 4-wire operation, this bit is normally programmed set except as noted in DIS,EN.

1044



NOTE 1: CARR OFF AND DSRDY FORCED LOW ON TRANSMIT SIDE.
NOTE 2: REVERSE CHANNEL NOT APPLICABLE IN FDX (RCR = 0)

Figure 2-9 Answering Calls

* Data Set Ready (CC) does not appear in the status byte on the transmit side in FDX operation. The receive side of the adapter shows the loss of Data Set Ready (CC). Loss of Data Set Ready on the transmit side in FDX operation does not cause an interrupt. It does, however, hold Busy high, thus preventing further end-of-character interrupts.

2.7.12 COMM MUX Command 2 Instruction Bits

The Command 2 instruction byte sets up the Universal Asynchronous Receiver/Transmitter (UART) and the baud rate generator within the COMM MUX. The Command 2 instruction bits are as follows:

CLKB,CLKA Without strapping, each COMM MUX normally is set to allow for the group 4 baud rates as shown below. As described in the COMM MUX Maintenance Manual, Publication Number 29-650, any one of groups 1, 2, and 3 may be chosen optionally, instead, by proper strapping during installation. The CLK bits (CLKB and CLKA) within the Command 2 instruction select one of the four baud rates within the installed group as follows:

STRAP OPTION	CLK BITS		BAUD RATE
	CLKB (Bit 8)	CLKA (Bit 9)	
Group 1	0	0	50
	0	1	110
	1	0	1,800
	1	1	2,400
Group 2	0	0	75
	0	1	134.5
	1	0	2,000
	1	1	3,600
Group 3	0	0	150
	0	1	600
	1	0	4,800
	1	1	9,600
Group 4	0	0	300
	0	1	1,200
	1	0	7,200
	1	1	19,200

BIT SELECT These bits select the number of data bits/character (not including parity).

BIT		NUMBER OF DATA BITS
10	11	
0	1	5
0	1	6
1	0	7
1	1	8

If fewer than eight data bits are selected when a write data is issued in the write mode, the data must be right-justified and unused bits are Don't Care. In the read mode, when a read data is issued, the character is presented to the processor right-justified with unused bits forced to the zero state.

STOP BIT

0=1	Stop bit
1=2	Stop bit

When the line is programmed for two stop bits, the COMM MUX transmits both. However, the receiver samples only the first stop bit.

PARITY

These bits select odd, even, or no parity.

BIT		PARITY
13	14	
1	0	ODD
1	1	EVEN
0	X	NONE

In the write mode, if parity is enabled (bit 13=1), the CCMM MUX generates and transmits the selected parity.

In the read mode, if parity is enabled, the COMM MUX compares the received parity with the selected parity and generates the PF status if a disagreement is detected.

If parity is disabled (bit 13=0), the hardware ignores parity. When transmitting, the hardware appends stop bit(s) after the last data bit and, when receiving, disables the parity detection circuit.

NOTE

> The least significant bit (bit 15) of the command byte must be a 1 or 0, as indicated, to permit the hardware to distinguish between the two commands. Command 2 should never be issued while data transfer is in progress.

2.7.13 Operations

COMM MUX programmers must consider the programming sequences for specific line types, interrupt control logic, initialization procedure, and device numbers.

2.7.14 Programming Sequences

The following paragraphs illustrate switched line, leased line, and half-duplex operation.

2.7.15 Switched Line Operation

To originate a call, the operator depresses the TALK key on the data set and dials the desired number. When the call is answered, a carrier is heard (being sent by the data set receiving the call). The operator then depresses the DATA key.

The cperator can now hang up and depress the AUTO key to return the equipment to automatic receive following this call. When the DATA key is depressed (the data light remains lit for the duration of the call), the EX status bit resets (Data Set Ready) (CC) as does the CARR OFF status bit. The EX status bit does not reset if the last state was an overflow, framing error, or parity error. The COMM MUX should be initialized to the read mode and thus be interrupted by the receiving carrier. When CARR OFF resets, the COMM MUX should be switched to the write mode to transmit data. Following the call, both sets (originating and receiving) should be issued a command read with DTR reset to disconnect. This procedure is typical, but not necessary. The user can design his own handshaking sequence. Figures 2-9 and 2-10 show timing sequences for answering calls and line turnaround. Wide variations between data set characteristics and common carrier procedures may modify the operating procedures. The user should ensure that the characteristics of the devices connected to the COMM MUX are compatible with the descriptions in this specification.

In Figure 2-9A, DTR and RQ2S are initially off. The status is X'0E' before RINGing starts. The RING causes an interrupt and a status of X'0F'. The RING status bit is set for the period of the RING from the data set.

When RING resets, the status is X'0E' and another interrupt is generated each time RING ⟶ 1. RING continues until the program sets DTR to answer the call. Shortly after DTR is set, the data set responds with (DSRDY=1), causing EX ⟶ 0. The status at this time is X'0A' (BSY=1 and CARR OFF=1).

When the data link is established, the data set turns CARR ON, thus generating an interrupt and a status of X'08' (BSY=1). If the adapter remains in the receive mode, busy stays active until a character is received. If the adapter transmits first, the program turns RQ2S on (command with the WRT bit set). With RQ2S on, the data set responds with CL2S=1. Since this bit is initially reset, an interrupt is generated when RQ2S is turned on and another interrupt is generated when the data set responds with CL2S=1. The adapter may now transmit.

Figures 2-9 and 2-10 assume ideal conditions. For example, in a typical switched network environment, more than one interrupt may be generated as carrier is initially established, or the received data from the local data set may be active during a connect or disconnect sequence. These problems can be attributed to the type (manufacturer) of data set used, the options implemented in the data set and the switched network. In particular, if the received data from the data set is active before carrier is established, the COMM MUX commences to assemble a garbled character. This can result in a receive busy interrupt with any or all of the character status bits set (PF, FRERR, OV). These status bits then remain set until a read data is executed (to set BSY and reset OV) and a valid character is received (to reset PF and FRERR).

2.7.16 Leased Line Operation

Because a connection is permanently established in leased line operation, no dial-up or disconnect is needed. Both stations are normally initialized to the read mode. Either end can originate a transfer by going into the write mode, which causes the receiving station to interrupt when the carrier appears. Upon receiving characters, the receiving end is in the read mode and a data transfer takes place. The exact handshaking protocol depends on the user.

2.7.17 Half-Duplex Operation

In half-duplex operation, only one terminal can transmit at any one time. To change the direction of transmission, the channel must be turned around. The processor turns the line around when it has a message to transmit. Data sets (e.g. 202C type), normally used in half-duplex operation, may be equipped with the reverse channel option that is used to signal the requirement to reverse the direction of transmission or to break the data flow. An important operating convention affecting reverse channel operation results from the presence of echo suppressors in long-distance lines. These suppressors normally disallow transmission of an echo.

In data communications, the echo suppressor must be disabled, so simultaneous transmission in both directions (main channel and reverse channel) is possible. The echo suppressor becomes reenabled if the tone on the line is absent for a period exceeding 100 milliseconds. To prevent reenabling of the echo suppressor, the reverse channel should be held on (high) when the main channel is off and vice versa. This convention ensures that a tone is on the line at all times.

The reverse channel is normally held on when the processor is accepting data. The processor signals its desire to transmit by lowering (off) the reverse channel and switching to the write mode. The receiving terminal introduces a program delay allowing its reverse channel to be turned on and enabling its read mode. This delay can be 200 to 1200 milliseconds. If the receiving device signals through the reverse channel that it is ready to receive data, this can be used instead of a program delay. When the processor is transmitting, a break condition sent from the terminal signifies that it wants to transmit. This break condition is indicated by the receive reverse channel going from on (high) to off (low). The processor should then raise its reverse channel lead high (on) and transfer to the read mode. The interface automatically introduces the necessary time delay before presenting data to the processor, ensuring valid data transfer instead of transition noise.

2.7.18 Interrupts

The COMM MUX has interrupt control logic that scans all interrupt sources in the system. This logic has the following characteristics (see Table 2-12 for interrupt conditions):

1. If an interrupt is detected, the hardware assigns priority to the line with the lowest address. In 4-wire operation, this is always the receive side.

2. When an interrupt is detected on the transmit or receive side of a given line in 2-wire operation, the address of the receive side (even) is always returned.

3. In 2-wire operation, the side not being used has interrupts disarmed (not queued).

4. If an interrupt is present on an enabled line, it can become queued in the interrupt scanner even if the interrupt is disabled before it is serviced. This condition can result in an interrupt from a disabled line. Servicing this interrupt clears the attention flip-flop on this line while disabled. Software is designed to ignore any interrupt from the device once it is disabled.

TABLE 2-12 INTERRUPT CONDITIONS

INTERRUPT CONDITION	HDX	FDX	
		REC	TRANS
RING → 1	X	X	
CARR OFF → 1	X (RD)	X	
CARR OFF → 0	X (RD)	X	
RCR → 1	X	X	
RCR → 0	X	X	
DSRDY → 0	X	X	
*BSY → 0	X	X	X
$\overline{\text{CL2S}}$ → 1	X (WRT)		X

* An interrupt is also generated in 2-wire operation when going from read to write mode if CL2S initially equals a 0; i.e., CL2S going from a zero to a one causes a BSY interrupt.

2.7.19 Initialization

When the initialize pushbutton on the display panel is depressed (or power failure restart sequence), the OV, PF, and FR ERR status bits cannot be guaranteed; therefore, the programmer should ignore these bits on the first interrupt. The COMM MUX is issued an output command to disable interrupts. Because an interrupt line may be active upon initialize or power up, a number of interrupt acknowledge instructions should be executed at that time. Also, a read data should be issued to all receivers to ensure that the busy bit equals a one.

2.7.20 Device Number

The COMM MUX has contiguous addressing with the lowest address X'10'. This is a switch option. Two consecutive addresses are assigned to each 4-wire line, with the even address for the receive side and the odd address for the transmit side. In 2-wire operation, each side responds to either address. In 4-wire operation, only one Command 2 is required. Command 2 should never be issued while a character transfer is in progress, because it may mutilate the character (transmit or receive).

2.7.21 Sample Programs

Appendix A and Appendix B contain sample flow charts and program listings for 16-bit and 32-bit processors, respectively.

2.8 ESCAPE PROGRAMMING INFORMATION

The Model 550 VDU can enable the program to accomplish special functions that format the screen or put the display into the desired mode. Cursor positioning, moving the cursor one position vertically or horizontally, clearing of one line or the whole screen, and "clear" functions are available.

To initiate an ESCAPE function, the ESCAPE character is output (normally "ESC"=X'1B') and followed with the upper-case letter to execute the desired result. This must be followed by a third character to execute the desired function. This third character is either the next character to be displayed or a pad character.

When reading the cursor address on a half-duplex device, the line must be immediately turned around with an output command and a dummy read is issued to clear the present interface buffer. When busy goes active on either a half-duplex or a full-duplex interface, the 2-character cursor position can then be read. If a read is not issued before the end of the character period, an overflow takes place and the character is lost.

Refer to Table 2-13 for the available multicode sequences.

TABLE 2-13 ESCAPE CODE SEQUENCE

CODE	ACTION
ESC A	Cursor up. Moves cursor up by 1 line. If in line 1, cursor wraps around to line 24.
ESC B	Cursor down. Moves cursor down by 1 line. If in line 24, moves display up by 1 line.
ESC C	Cursor right. Moves cursor 1 position to the right. When cursor is moved beyond position 80, the cursor wraps around to position 1 of the following line.
ESC D	Cursor left. Moves cursor 1 position to the left. When cursor is moved beyond position 1, the cursor wraps around to position 80 of the previous line. The code is ignored if the cursor is at the home position.
ESC H	Cursor home. Moves cursor to position 1, line 1 (home).
ESC X character	Direct cursor address - line position. Moves cursor vertically to any line as specified by the character following "X". The code is ignored if it exceeds 24 lines.
ESC Y character	Direct cursor address - character position. Moves cursor horizontally to any position on a line. The character following "Y" specifies the character position.
ESC K	Clear all. Clears display memory to spaces and homes the cursor. ESC K needs 20 ms to complete the operation. Thus, the programmer should insert sufficient nulls as filler data (20 at 9600 baud).
ESC I	Clear line. Clears the line (reset to spaces), starting with the present position of the cursor through the end of the line. ESC I needs 20 ms to complete the operation. Thus, the programmer should insert sufficient nulls as filler data (20 at 9600 baud).
ESC	(followed by a non-ESC character) ASCII characters other than A, B, C, D, H, X, Y, K, and I are stored in the display memory as data.
ESC ESC	When a second ESC code follows an ESC, the second is displayed, but is not used to generate any ESC code sequences with following characters.

TABLE 2-14 CURSOR ADDRESSING (IN FULL ASCII MODE)

1049

ASCII CHARACTER	LINE or COLUMN		ASCII CHARACTER	LINE or COLUMN
(SP)	1		H	41
!	2		I	42
"	3		J	43
#	4		K	44
$	5		L	45
%	6		M	46
&	7		N	47
'	8		O	48
(	9		P	49
)	10		Q	50
*	11		R	51
+	12	LINE NUMBER	S	52
,	13		T	53
-	14		U	54
.	15		V	55
/	16		W	56
0	17		X	57
1	18		Y	58
2	19		Z	59
3	20		[	60
4	21		\	61
5	22		]	62
6	23		^	63*
7	24		_	64
8	25		`	65
9	26		a	66
:	27		b	67
;	28		c	68
<	29		d	69
=	30		e	70
>	31		f	71
?	32		g	72
@	33		h	73
A	34		i	74
B	35		j	75
C	36		k	76
D	37		l	77
E	38		m	78
F	39		n	79
G	40		o	80

NOTE

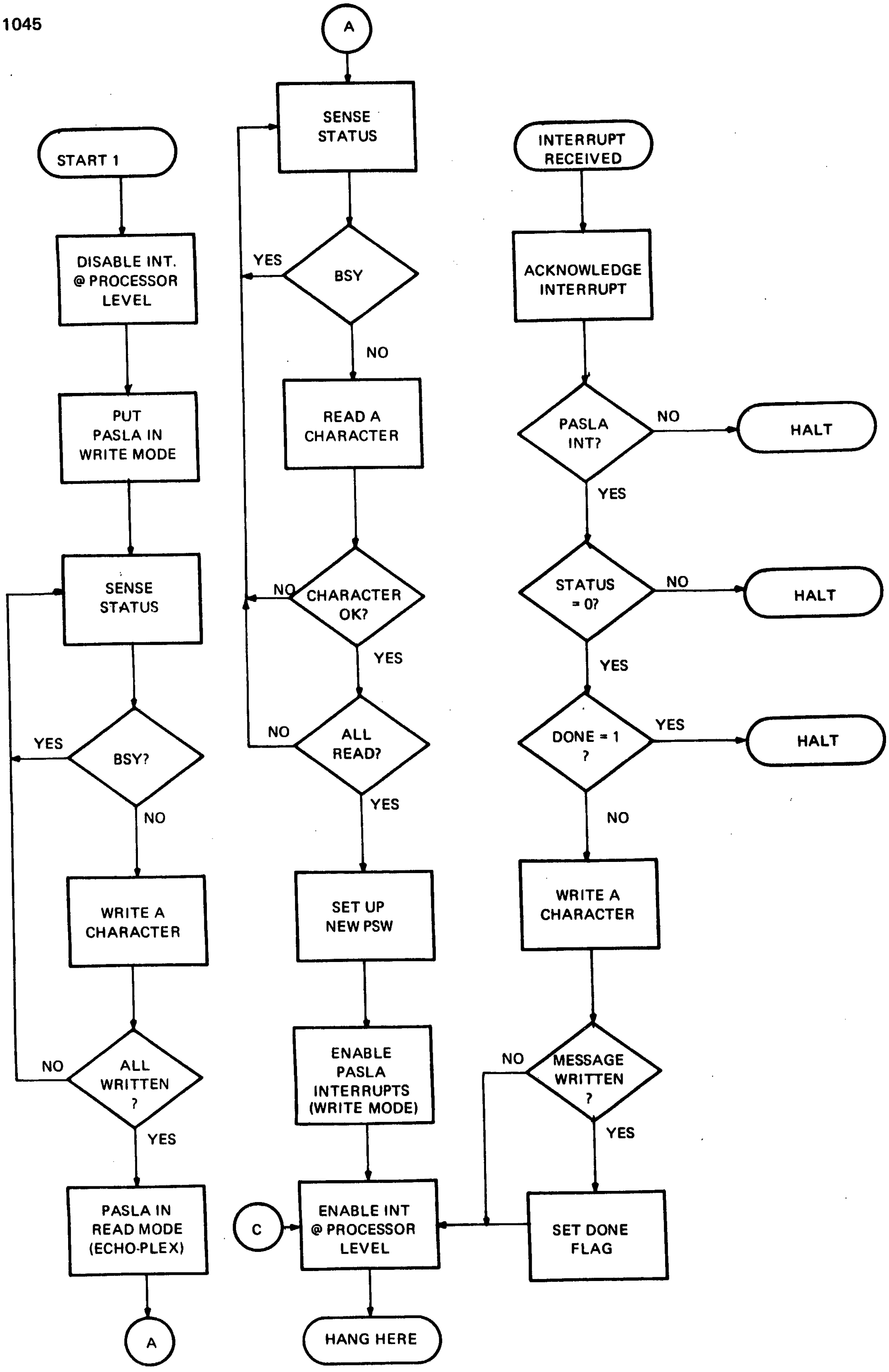
To move cursor to column 50, line 5, depress (in sequence) MULT, Y, Q, and MULT, X, $, (NULL).

*Can be positioned or read up to 10/1/77. After 10/1/77, full table applies.

APPENDIX A
16-BIT PROGRAMMING EXAMPLES

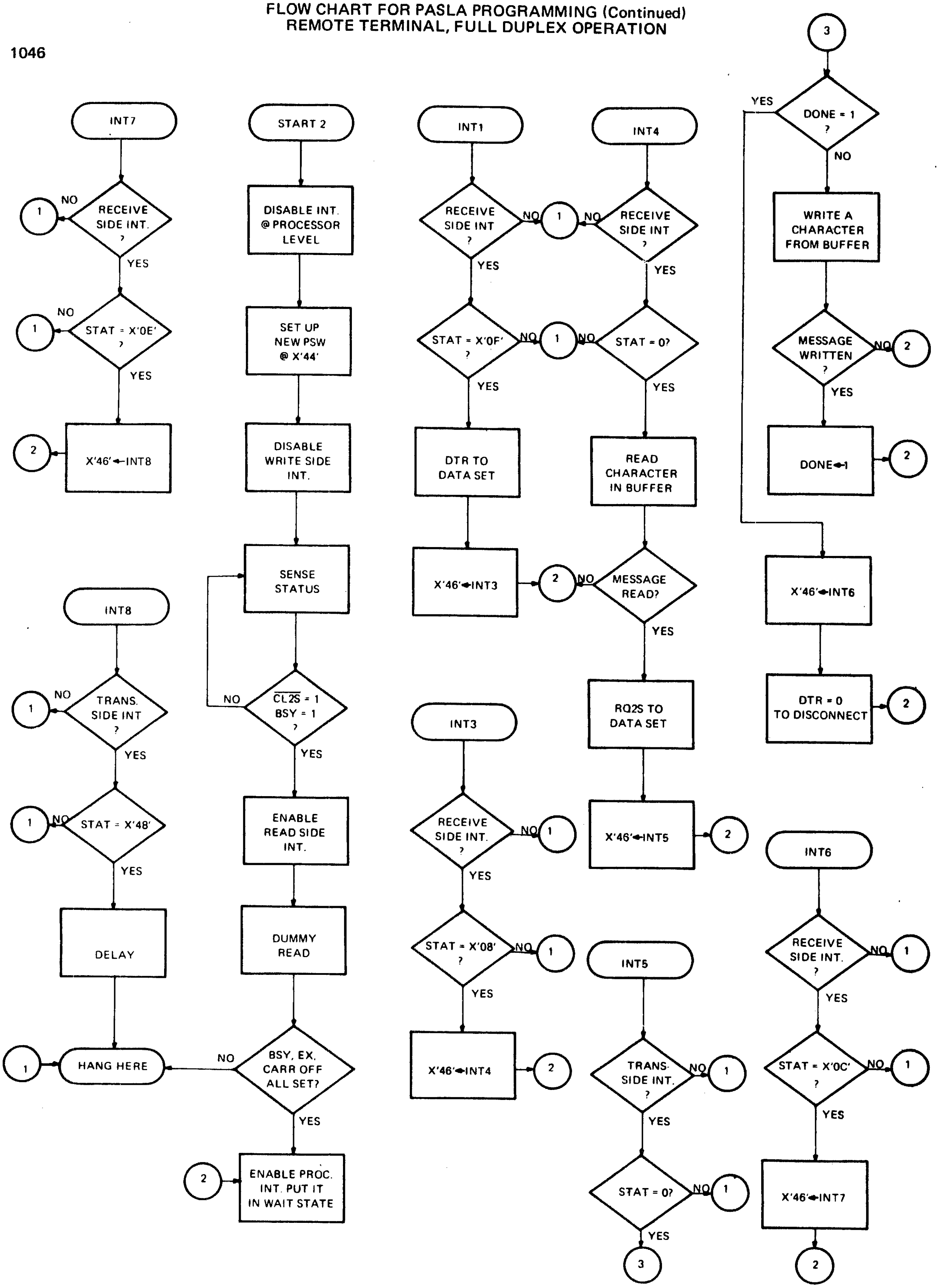
FLOW CHART FOR PASLA PROGRAMMING LOCAL TERMINAL, HALF DUPLEX OPERATION

1045



FLOW CHART FOR PASLA PROGRAMMING (Continued)
REMOTE TERMINAL, FULL DUPLEX OPERATION

1046



```
PASLA PROGRAMMING EXAMPLES FOR 16-BIT PROCESSOR          PAGE   1  14:16:05  11/30/79
PROG=           ASSEMBLED BY CAL 03-066R07-00   (32-BIT)
                                   PROG  PASLA PROGRAMMING EXAMPLES FOR 16-BIT PROCESSOR         PS210000
                                   CROSS                                                         PS210020
                                   TARGT 16
                                   WIDTH 120                                                     PS210030
                             *                                                                   PS210040
                             *  THESE EXAMPLES DEMONSTRATE SEQUENCES TO PROGRAM PASLA IN         PS210050
                             *  VARIOUS ENVOIRNMENTS. THE TERMINAL & PASLA INTERFACE SHOULD      PS210060
                             *  BE STRAPPED / CONNECTED AS MENTIONED IN INDIVIDUAL EXAMPLE.      PS210070
                             *                                                                   PS210080
                             * REGISTER ASSIGNMENTS                                              PS210090
                             *                                                                   PS210100
     0000 0001               MSG      EQU   1              MESSAGE START ADDRESS                 PS210110
     0000 0002               RCV      EQU   2              PASLA (RECEIVE) ADDRESS               PS210120
     0000 0003               SND      EQU   3              PASLA (SEND) ADDRESS                  PS210130
     0000 0004               CHAR     EQU   4              CHARACTER BEING TRANSFERRED           PS210140
     0000 0005               WORK     EQU   5              WORK REGISTER                         PS210150
     0000 0006               DONE     EQU   6              EXAMPLF DONE FLAG                     PS210160
     0000 0007               DEV      EQU   7              PASLA (HDX) ADDRESS                   PS210170
     0000 0008               STAT     EQU   8              PASLA STATUS                          PS210180
     0000 0009               REPEAT   EQU   9              EXAMPLE START ADDRESS                 PS210190
     0000 000F               LINK     EQU   15             LINK REGISTER                         PS210200
                             *                                                                   PS210210
                             *                                                                   PS210220
     0000 0004               EX       EQU   4              EX BIT IN STATUS BYTE IS ONE          PS210230
     0000 0008               BSY      EQU   8              BSY BIT IN STATUS BYTE IS ONE         PS210240
     0000 0008               CARRY    EQU   8              CARRY FLAG IN PSW IS ONE              PS210250
                             *                                                                   PS210260
```

```
PASLA PROGRAMMING EXAMPLES FOR 16-BIT PROCESSOR              PAGE  2  14:16:05  11/30/79
LOCAL TERMINAL, HALF-DUPLEX PASLA OPERATION
                             29  *                                                                          PS210280
                             30  *  TTY, CRT OR GDT SHOULD BE INTERFACED THROUGH PASLA(HDX) INTERFACE        PS210290
                             31  *  CRT/GDT SHOULD BE STRAPPED FOR HALF-DUPLEX OPERATION                    PS210300
                             32  *  THE FOLLOWING EXAMPLE IS FOR 16-BIT PROCESSOR                           PS210310
                             33  *  START EXECUTION @ START1                                                PS210320
                             34  *                                                                          PS210330
                             35  *  WRITE CHARACTERS & READ KEYS USING SENSE STATUS LOOP                    PS210340
                             36  *  WRITE CHARACTERS UNDER INTERRUPT CONTROL                                PS210350
                             37  *                                                                          PS210360
                             38  *                                                                          PS210370
0000R  0766                  39  START1    XAR   DONE,DONE                                                  PS210380
0002R  9556                  40            EPSR  WORK,DONE          DISABLE INT @ PROCESSOR LEVEL           PS210390
0004R  C890 0000R            41            LDAI  REPEAT,START1                                              PS210400
0008R  4870 0208R            42            LH    DEV,DEVADR         GET PASLA DEVICE ADDRESS                PS210410
000CR  DE70 020ER            43            OC    DEV,SECOND         SET PASLA AS PER SECOND COMMAND         PS210420
0010R  DE70 020FR            44            OC    DEV,DISWRT         WRITE MODE                              PS210430
0014R  C810 0218R            45            LDAI  MSG,MSG1           (MSG) = MESSAGE START ADDRESS           PS210440
                             46  *                                                                          PS210450
0018R  D341 0000             47  EXMP1A    LB    CHAR,0(MSG)                                                PS210460
001CR  41F0 00A2R            48            BAL   LINK,OUTCHR        OUTPUT A CHARACTER                      PS210470
0020R  2611                  49            AIS   MSG,1                                                      PS210480
0022R  C510 022AR            50            CLAI  MSG,MSG1END                                                PS210490
0026R  2087                  51            BLS   EXMP1A             LOOP TILL 'TYPE 1234567890' OUTPUTED    PS210500
0028R  DE70 0210R            52            OC    DEV,DISRD          SELECT READ MODE (ECHO-PLEX)            PS210510
002CR  9875                  53            RDR   DEV,WORK           DUMMY READ TO SET BSY                   PS210520
002ER  41F0 00AER            54            BAL   LINK,DELAY         LINE TURN AROUND DELAY                  PS210530
0032R  0711                  55            XAR   MSG,MSG                                                    PS210540
                             56  *                                                                          PS210550
0034R  41F0 0092R            57  EXMP1B    BAL   LINK,READ          READ A KEY CODE WHEN DEPRESSED          PS210560
0038R  D441 021DR            58            CLB   CHAR,TYPED(MSG)    COMPARE WITH THE EXPECTED               PS210570
003CR  4230 003CR            59            BNE   *                                                          PS210580
0040R  2611                  60            AIS   MSG,1                                                      PS210590
0042R  C510 000A             61            CLAI  MSG,10                                                     PS210600
0046R  2089                  62            BLS   EXMP1B             LOOP TILL 10 DIGITS TYPED               PS210610
                             63  *                                                                          PS210620
                             64  * WRITE UNDER INTERRUPT CONTROL                                            PS210630
                             65  *                                                                          PS210640
0048R  C850 006AR            66            LDAI  WORK,INT                                                   PS210650
004CR  4060 0044             67            STA   DONE,X'44'         SET UP NEW PSW FOR                      PS210660
0050R  4050 0046             68            STA   WORK,X'46'         IMMEDIATE INTERRUPT                     PS210670
0054R  C810 022AR            69            LDAI  MSG,MSG2                                                   PS210680
0058R  DE70 0211R            70            OC    DEV,ENWRT          ENABLE INT @ PASLA LEVEL (1 PENDING)    PS210690
005CR  41F0 00AER            71            BAL   LINK,DELAY         LINE TURNAROUND DELAY                   PS210700
0060R  C850 4000             72  EXMP1C    LDAI  WORK,X'4000'                                               PS210710
0064R  9545                  73            EPSR  CHAR,WORK          ENABLE INT @ PROCESSOR LEVEL            PS210720
0066R  4300 0066R            74            B     *                                                          PS210730
                             75  *                                                                          PS210740
                             76  * IMMEDIATE INTERRUPT IS RECEIVED                                          PS210750
                             77  *                                                                          PS210760
006AR  9F58                  78  INT       AIR   WORK,STAT          ACKNOWLEDGE AN INTERRUPT                PS210770
006CR  0557                  79            CLAR  WORK,DEV           COMPARE DEVICE THAT INTERRUPTED         PS210780
006ER  4230 006ER            80            BNE   *                                                          PS210790
0072R  0888                  81            LDAR  STAT,STAT          STATUS SHOULD BE ZERO (BSY = 0)         PS210800
```

```
     PASLA PROGRAMMING EXAMPLES FOR 16-BIT PROCESSOR          PAGE   3  14:16:05  11/30/79

     LOCAL TERMINAL, HALF-DUPLEX PASLA OPERATION
                                                                                                   PS210810
0074R  4230 0074R          82          BNZ   *                                                     PS210820
0078R  0866                83          LDAR  DONE,DONE                                             PS210830
007AR  4230 00B6R          84          BNZ   HALT           HALT IF DONE                           PS210840
                           85  *                                                                   PS210850
007ER  DA71 0000           86          WD    DEV,0(MSG)     WRITE A CHARACTER                      PS210860
0082R  2611                87          AIS   MSG,1                                                 PS210870
0084R  C510 0238R          88          CLAI  MSG,MSG2END                                           PS210880
0088R  4280 0060R          89          BL    EXMP1C         LOOP TILL 'CORRECT J' OUTPUTED         PS210890
008CR  2461                90          LIS   DONE,1         SET DONE FLAG                          PS210900
008ER  4300 0060R          91          B     EXMP1C         ACKNOWLEDGE LAST INTERRUPT             PS210910
                           92  *                                                                   PS210920
                           93  *                                                                   PS210930
0092R  9078                94  READ    SSR   DEV,STAT       SENSE PASLA STATUS                     PS210940
0094R  4240 00B6R          95          BTC   EX,HALT        HALT IF 'EX' IS SET                    PS210950
0098R  2083                96          BTBS  BSY,3          LOOP IN 'BSY'                          PS210960
009AR  9B74                97          RDR   DEV,CHAR       READ CHARACTER WHEN BSY DROPS          PS210970
009CR  C440 007F           98          NAI   CHAR,X'7F'     REMOVE PARITY BIT                      PS210980
00A0R  030F                99          BR    LINK           RETURN                                 PS210990
                          100  *                                                                   PS211000
00A2R  9D78               101  OUTCHR  SSR   DEV,STAT                                              PS211010
00A4R  2081               102          BTBS  BSY,1          LOOP ON BSY                            PS211020
00A6R  9A74               103          WDR   DEV,CHAR                                              PS211030
00A8R  9D78               104          SSR   DEV,STAT                                              PS211040
00AAR  2081               105          BTBS  BSY,1          WAIT FOR PASLA TO BECOME NOT BSY       PS211050
00ACR  030F               106          BR    LINK           RETURN                                 PS211060
                          107  *                                                                   PS211070
00AER  0755               108  DELAY   XAR   WORK,WORK                                             PS211080
00B0R  2651               109          AIS   WORK,1                                                PS211090
00B2R  2281               110          BFBS  CARRY,1                                               PS211100
00B4R  030F               111          BR    LINK                                                  PS211110
                          112  *                                                                   PS211120
00B6R  2451               113  HALT    LIS   WORK,1                                                PS211130
00B8R  915F               114          SLHLS WORK,15        WORK = X'8000'                         PS211140
00BAR  9515               115          EPSR  MSG,WORK       HALT PROCESSOR                         PS211150
00BCR  0309               116          BR    REPEAT                                                PS211160
                          117  *
```

```
                   *                                                                  PS211180
                   * PHONE RECEIVER IS IN IT'S CRADDLE                                PS211190
                   *                                                                  PS21550
                   *                                                                  PS211210
                   *                                                                  PS211220
                   * START THE PROGRAM @ START2. DIAL IN THE PROCESSOR TELEPHONE      PS211230
                   * NUMBER FROM THE TELETYPE TERMINAL. WHEN CARRIER IS HEARD,ENGAGE  PS211240
                   * THE PHONE RECEIVER IN THE COUPLER. OBSERVE DATA SET LIGHTS TO    PS211250
                   * MAKE SURE THAT CONNECTION IS ESTABLISHED. THE PROCESSOR SHOULD BE PS211260
                   * IN WAIT STATE. TYPE 10 CHARACTER MESSAGE ON TELETYPE. THE        PS211270
                   * PROGRAM ECHOES BACK THE SAME MESSAGE & DISCONNECTS THE LINE.     PS211280
                   * USER SHOULD REPLACE THE RECEIVER IN IT'S CRADDLE.                PS211290
                   *                                                                  PS211300
                   *                                                                  PS211310
003ER  0766        START2   XHR   DONE,DONE                                           PS211320
00C0R  9516                 EPSR  MSG,DONE       DISABLE INT @ PROCESSOR LEVEL        PS211330
00C2R  C810 010AR           LHI   MSG,INT1                                            PS211340
00C6R  4060 0044            STH   DONE,X'44'     SET UP NEW PSW FOR IMMEDIATE INT.    PS211350
00CAR  4010 0046            STH   MSG,X'46'                                           PS211360
00CER  C890 00BER           LDAI  REPEAT,START2                                       PS211370
00D2R  4820 0204R           LH    RCV,RCVADR     GET DEVICE ADDRESSES                 PS211380
00D6R  4830 0206R           LH    SND,SNDADR                                          PS211390
00DAR  DE20 020ER           OC    RCV,SECOND     SET UP PASLA                         PS211400
00DER  DE30 0212R           OC    SND,DISWRT2                                         PS211410
                   *                                                                  PS211420
00E2R  9D38                 SSR   SND,STAT                                            PS211430
00E4R  C580 0048            CLHI  STAT,X'48'     CL2S NOT & BSY MUST BE SET           PS211440
00E8R  4230 00E8R           BNE   *                                                   PS211450
00ECR  DE20 0213R           OC    RCV,ENREAD                                          PS211460
00F0R  DE20 0214R           OC    RCV,WRTRD                                           PS211470
00F4R  9B25                 RDR   RCV,WORK       DUMMY READ                           PS211480
00F6R  9D28                 SSR   RCV,STAT                                            PS211490
00F8R  C480 001F            NHI   STAT,X'1F'     IGNORE OV,PF,FR ERROR AS THEY ARE    PS211500
                   *                             NOT GUARANTEED ON INITIALIZATION     PS211510
00FCR  278E                 SIS   STAT,14        BSY,EX,CARR OFF MUST BE SET          PS211520
00FER  4230 00FER           BNZ   *                                                   PS211530
0102R  C810 023AR           LHI   MSG,MSG4                                            PS211540
0106R  C200 020AR           LPSW  WAIT                                                PS211550
                   *                                                                  PS211560
010AR  9F78        INT1     AIR   DEV,STAT                                            PS211570
010CR  0572                 CLHR  DEV,RCV                                             PS211580
010ER  4230 010ER           BNE   *                                                   PS211590
0112R  C580 000F            CLHI  STAT,X'0F'     RING INTERRUPT                       PS211600
0116R  4230 0116R           BNE   *                                                   PS211610
011AR  DE20 0215R           OC    RCV,DTR        INDICATE DTR TO DATA SET,ECHO,DTB    PS211620
011ER  C850 0134R           LHI   WORK,INT3                                           PS211630
0122R  4050 0046            STH   WORK,X'46'                                          PS211640
                   *                                                                  PS211650
0126R  9D28        SENSE2   SSR   RCV,STAT                                            PS211660
0128R  C580 000A            CLHI  STAT,X'0A'     DATA SET READY SHOULD DROP           PS211670
012CR  2033                 BNES  SENSE2         TO ESTABLISHE                        PS211680
012ER  2034                 BNES  SENSE2         TO ESTBLISH DATA LINK                PS211690
0130R  C200 020AR           LPSW  WAIT                                                PS211700
```

REMOTE TERMINAL DATA TRANSFER (16-BIT)

```
                     *                                                                   PS211710
0134R  9F78          INT3     AIR    DEV,STAT                                            PS211720
0136R  0572                   CLHR   DEV,RCV                                             PS211730
0138R  4230 0138R             BNE    *                                                   PS211740
013CR  C580 0008              CLHI   STAT,8         CARR OFF DROPS                       PS211750
0140R  4230 0140R             BNE    *                                                   PS211760
0144R  C850 0150R             LHI    WORK,INT4                                           PS211770
0148R  4050 0046              STH    WORK,X'46'                                          PS211780
014CR  C200 020AR    RDTEN    LPSW   WAIT                                                PS211790
                     *                                                                   PS211800
0150R  9F78          INT4     AIR    DEV,STAT                                            PS211810
0152R  0572                   CLHR   DEV,RCV                                             PS211820
0154R  4230 0154R             BNE    *                                                   PS211830
0158R  0888                   LHR    STAT,STAT      BSY DROPS WHEN KEY IS DEPRESSED      PS211840
015AR  4230 015AR             BNZ    *                                                   PS211850
015ER  DB21 0000              RD     RCV,0(MSG)     READ CHARACTER INTO MESSAGE BUFFER   PS211860
0162R  2611                   AIS    MSG,1                                               PS211870
0164R  C510 0244R             CLHI   MSG,MSG4END                                         PS211880
0168R  4280 014CR             BL     RDTEN                                               PS211890
                     *                                                                   PS211900
016CR  C810 0238R             LHI    MSG,MSG3                                            PS211910
0170R  DE30 0211R             OC     SND,ENWRT      SEND RQ2S                            PS211920
0174R  C850 0180R             LHI    WORK,INT5                                           PS211930
0178R  4050 0046              STH    WORK,X'46'                                          PS211940
017CR  C200 020AR    WRTEN    LPSW   WAIT                                                PS211950
                     *                                                                   PS211960
0180R  9F78          INT5     AIR    DEV,STAT                                            PS211970
0182R  0573                   CLHR   DEV,SND                                             PS211980
0184R  4230 0184R             BNE    *                                                   PS211990
0188R  0888                   LHR    STAT,STAT      CL2S NOT DROPS, BSY DROPS            PS25500
018AR  4230 018AR             BNZ    *                                                   PS212010
018ER  0866                   LHR    DONE,DONE                                           PS212020
0190R  4230 01A8R             BNZ    DISCONN                                             PS212030
0194R  DA31 0000              WD     SND,0(MSG)                                          PS212040
0198R  2611                   AIS    MSG,1                                               PS212050
019AR  C510 024AR             CLHI   MSG,MSG3END                                         PS212060
019ER  4280 017CR             BL     WRTEN                                               PS212070
                     *                                                                   PS212080
01A2R  2461                   LIS    DONE,1                                              PS212090
01A4R  4300 017CR             B      WRTEN          CLEAR LAST WRITE INT.                PS212100
                     *                                                                   PS212110
01A8R  C850 01B8R    DISCONN  LHI    WORK,INT6                                           PS212120
01ACR  4050 0046              STH    WORK,X'46'                                          PS212130
01B0R  DE20 0216R             OC     RCV,DTROFF     DTR IS OFF TO DISCONNECT             PS212140
01B4R  C200 020AR             LPSW   WAIT                                                PS212150
                     *                                                                   PS212160
01B8R  9F78          INT6     AIR    DEV,STAT                                            PS212170
01BAR  0572                   CLHR   DEV,RCV                                             PS212180
01BCR  4230 01BCR             BNE    *                                                   PS212190
01C0R  C580 000C              CLHI   STAT,X'0C'     DSRDY DROP CAUSES AN INTERRUPT       PS212200
01C4R  4230 01C4R             BNE    *                                                   PS212210
01C8R  C850 01D4R             LHI    WORK,INT7                                           PS212220
01CCR  4050 0046              STH    WORK,X'46'                                          PS212230
```

```
PASLA PROGRAMMING EXAMPLES FOR 16-BIT PROCESSOR          PAGE   6  14:16:05  11/30/79
REMOTE TERMINAL DATA TRANSFER (16-BIT)
01D0R   C200 020AR            225            LPSW  WAIT                                                    PS212240
                              226   *                                                                      PS212250
01D4R   9F78                  227   INT7     AIR   DEV,STAT                                                PS212260
01D6R   0572                  228            CLHR  DEV,RCV                                                 PS212270
01D8R   4230 01D8R            229            BNE   *                                                       PS212280
01DCR   C580 000E             230            CLHI  STAT,X'0E'          CARR OFF SETS CAUSING AN INT.       PS212290
01E0R   4230 01E0R            231            BNE   *                                                       PS212300
01E4R   C850 01F0R            232            LHI   WORK,INT8                                               PS212310
01E8R   4050 0046             233            STH   WORK,X'46'                                              PS212320
01ECR   C200 020AR            234            LPSW  WAIT                                                    PS212330
                              235   *                                                                      PS212340
01F0R   9F78                  236   INT8     AIR   DEV,STAT                                                PS212350
01F2R   0573                  237            CLHR  DEV,SND                                                 PS212360
01F4R   4230 01F4R            238            BNE   *                                                       PS212370
01F8R   C580 0048             239            CLHI  STAT,X'48'          CL2S NOT SETS. BSY IS ALSO SET      PS212380
01FCR   4230 01FCR            240            BNE   *                                                       PS212390
0200R   4300 00B6R            241            B     HALT                                                    PS212400
                              242   *                                                                      PS212410
                              243   * CONSTANTS & MESSAGES USED IN ABOVE EXAMPLES                          PS212420
                              244   *                                                                      PS212430
0204R   0032                  245   RCVADR   DCX   32                  PASLA (RECEIVE) ADDRESS             PS212440
0206R   0033                  246   SNDADR   DCX   33                  PASLA (SEND) ADDRESS                PS212450
0208R   0010                  247   DEVADR   DCX   10                  PASLA (HDX) ADDRESS                 PS212460
020AR   C000                  248   WAIT     DCX   C000                PSW FOR WAIT                        PS212470
020CR   00BER                 249            DC    Z(START2)                                               PS212480
020ER   78                    250   SECOND   DB    X'78'               PASLA SET UP FOR HIGHER BAUD RATE,  PS212490
                              251   *                                  8 DATA BITS/CHAR, 2 STOP BITS,      PS212500
                              252   *                                  NO PARITY CHECK.                    PS212510
020FR   AB                    253   DISWRT   DB    X'AB'               DISABLE INT, WRITE MODE             PS212520
0210R   B9                    254   DISRD    DB    X'B9'               DISABLE INT, READ WITH ECHO-BACK    PS212530
0211R   6B                    255   ENWRT    DB    X'6B'               ENABLE INT, WRITE MODE              PS212540
0212R   83                    256   DISWRT2  DB    X'83'               DIS,WRITE                           PS212550
0213R   41                    257   ENREAD   DB    X'41'               EN,READ                             PS212560
0214R   03                    258   WRTRD    DB    X'03'               WRT/RD = 1                          PS212570
0215R   3B                    259   DTR      DB    X'3B'               DATA TERMINAL READY                 PS212580
0216R   03                    260   DTROFF   DB    X'03'               DTR = 0                             PS212590
                              261   *                                                                      PS212600
0218R   5459 5045 2031 3233   262   MSG1     DC    C'TYPE 1234567890',X'0D0A'                              PS212610
0220R   3435 3637 3839 3020
0228R   0D0A
        0000 022AR            263   MSG1END  EQU   *                                                       PS212620
        0000 021DR            264   TYPED    EQU   *-13                                                    PS212630
022AR   0D0A                  265   MSG2     DC    X'0D0A',C'CORRECT',X'2121',X'0D0A'                      PS212640
022CR   434F 5252 4543 5420
0234R   2121
0236R   0D0A
        0000 0238R            266   MSG2END  EQU   *                                                       PS212650
0238R   0D0A                  267   MSG3     DC    X'0D0A'                                                 PS212660
023AR                         268   MSG4     DS    10                                                      PS212670
        0000 0244R            269   MSG4END  EQU   *                                                       PS212680
0244R   0D0A                  270            DC    X'0D0A',0,0                                             PS212690
0246R   0000
0248R   0000
```

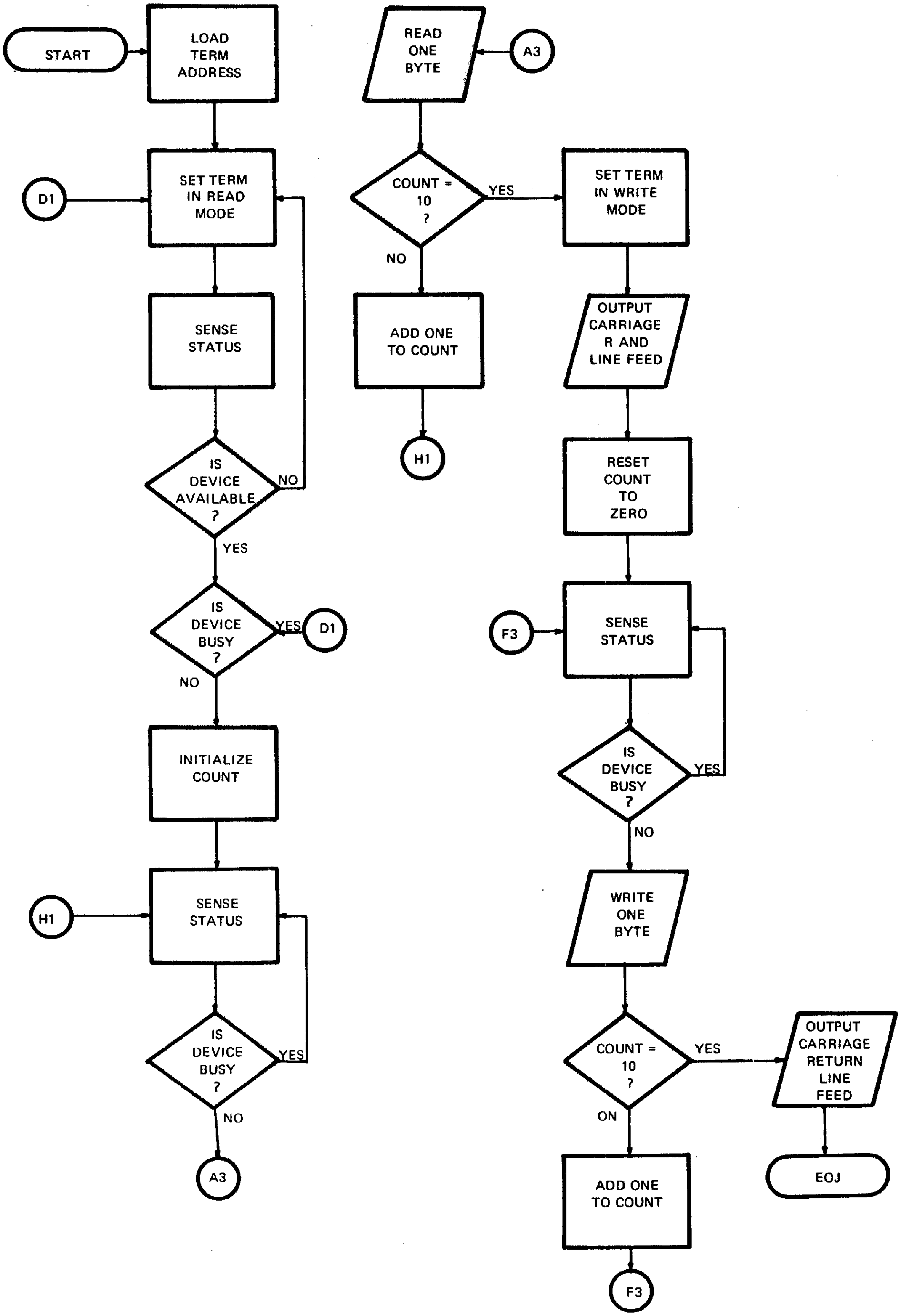
```
     PASLA PROGRAMMING EXAMPLES FOR 16-BIT PROCESSOR          PAGE   7  14:16:05  11/30/79

     REMOTE TERMINAL DATA TRANSFER (16-BIT)

         0000 024AR             271  MSG3END  EQU    *                                                      PS212700
                                272  *                                                                      PS212710
  024AR                         273           END                                                           PS212720
```

```
    PASLA PROGRAMMING EXAMPLES FOR 16-BIT PROCESSOR          PAGE   8  14:16:05  11/30/79
    REMOTE TERMINAL DATA TRANSFER (16-BIT)
ASSEMBLED BY CAL 03-066R07-00   (32-BIT)

START OPTIONS:  T=16,ERLST

   NO CAL ERRORS
   NO CAL WARNINGS
    2 PASSES

ABSTOP     0000 0000
ADC        0000 0002
BSY        0000 0008       25*     96     102     105
CARRY      0000 0008       26*    110
CHAR       0000 0004       15*     47      58      73      97      98     103
DELAY      0000 00AER      54      71     108*
DEV        0000 0007       18*     42      43      44      52      53      70      79      86      94      97     101     103
                          104     158     159     173     174     182     183     198     199     218     219     227     228
                          236     237
DEVADR     0000 0208R      42     247*
DISCONN    0000 01A8R     204     213*
DISRD      0000 0210R      52     254*
DISWRT     0000 020FR      44     253*
DISWRT2    0000 0212R     142     256*
DONE       0000 0006       17*     39      39      40      67      83      83      90     133     133     134     136     203
                          203     210
DTR        0000 0215R     163     259*
DTROFF     0000 0216R     215     260*
ENREAD     0000 0213R     147     257*
ENWRT      0000 0211R      70     193     255*
EX         0000 0004       24*     95
EXMP1A     0000 0018R      47*     51
EXMP1B     0000 0034R      57*     62
EXMP1C     0000 0060R      72*     89      91
HALT       0000 00B6R      84      95     113*    241
IMPTOP     0000 024AR
INT        0000 006AR      66      78*
INT1       0000 010AR     135     158*
INT3       0000 0134R     164     173*
INT4       0000 0150R     178     182*
INT5       0000 0180R     194     198*
INT6       0000 01B8R     213     218*
INT7       0000 01D4R     223     227*
INT8       0000 01F0R     232     236*
LADC       0000 0001
LINK       0000 000F       21*     48      54      57      71      99     106     111
MSG        0000 0001       12*     45      47      49      50      55      55      58      60      61      69      86      87
                           88     115     134     135     137     155     187     188     189     192     205     206     207
MSG1       0000 0218R      45     262*
MSG1END    0000 022AR      50     263*
MSG2       0000 022AR      69     265*
MSG2END    0000 0238R      88     266*
MSG3       0000 0238R     192     267*
MSG3END    0000 024AR     207     271*
MSG4       0000 023AR     155     268*
```

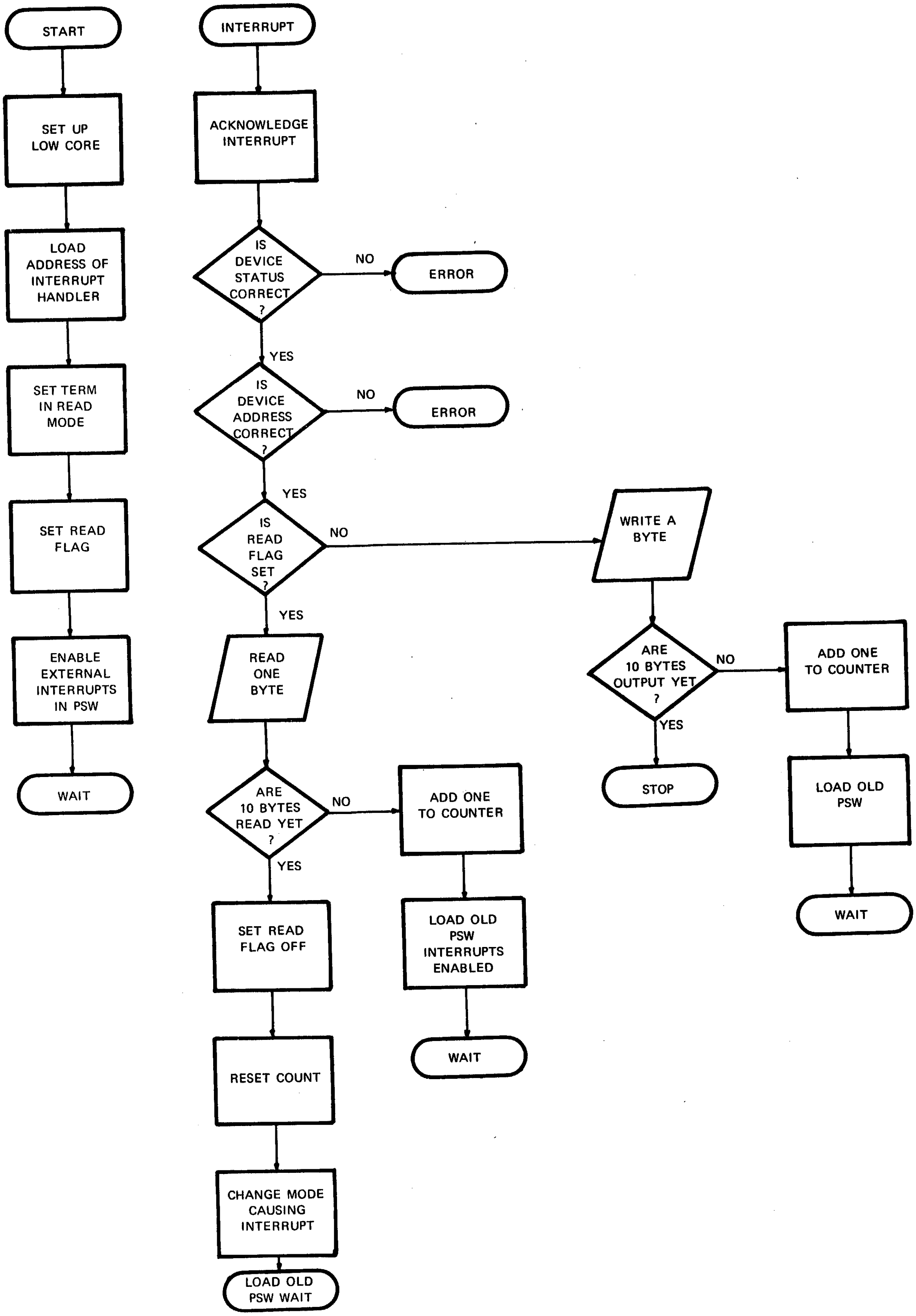
REMOTE TERMINAL DATA TRANSFER (16-BIT)

```
MSG4END    0000 0244R      189     269*
OUTCHR     0000 00A2R       48     101*
PURETOP    0000 0000R
RCV        0000 0002        13*    139     141     147     148     149     150     159     163     167     174     183     187
                           215     219     228
RCVADR     0000 0204R      139     245*
RDTEN      0000 014CR      180*    190
READ       0000 0092R       57      94*
REPEAT     0000 0009        20*     41     116     138
SECOND     0000 020ER       43     141     250*
SENSE2     0000 0126R      167*    169     170
SND        0000 0003        14*    140     142     144     193     199     205     237
SNDADR     0000 0206R      140     246*
START1     0000 0000R       39*     41
START2     0000 00BER      133*    138     249
STAT       0000 0008        19*     78      81      81      94     101     104     144     145     150     151     153     158
                           161     167     168     173     176     182     185     185     198     201     201     218     221
                           227     230     236     239
TYPED      0000 021DR       58     264*
WAIT       0000 020AR      156     171     180     196     216     225     234     248*
WORK       0000 0005        16*     40      53      66      68      72      73      78      79     108     108     109     113
                           114     115     149     164     165     178     179     194     195     213     214     223     224
                           232     233
WRTEN      0000 017CR      196*    208     211
WRTRD      0000 0214R      148     258*
```

CLI
UNDER STATUS SENSING



CLI UNDER INTERRUPTS

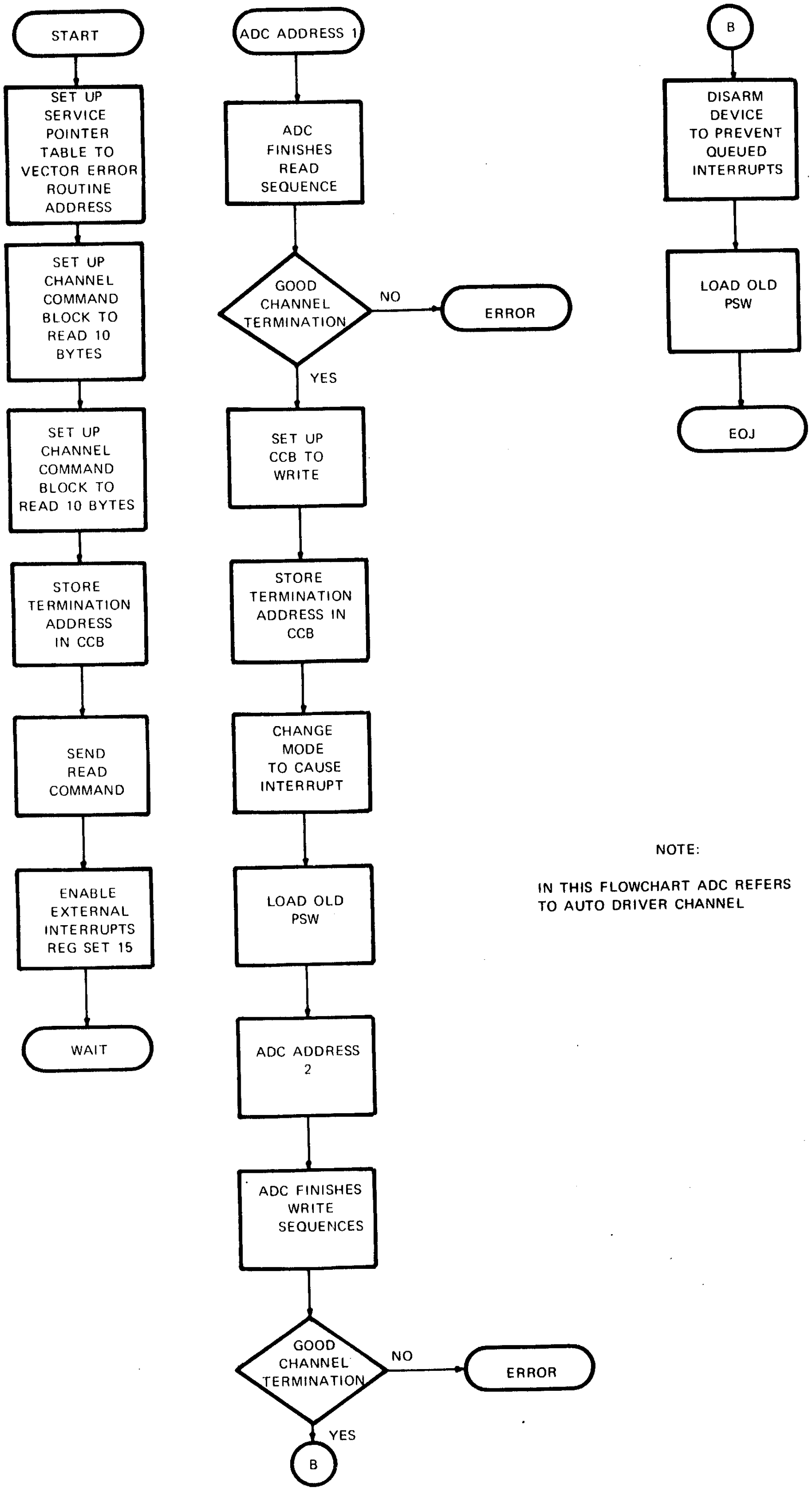


```
      SAMPLE PROGRAM WITH THE CURRENT LOOP INTERFACE           PAGE   1  16:02:07  12/03/79
PROG= CLI16S     ASSEMBLED BY CAL 03-066R07-00   (32-BIT)

                                CLI16S   PROG  SAMPLE PROGRAM WITH THE CURRENT LOOP INTERFACE
                                               CROSS
                                               TARGT 16
                                               WIDTH 120
                                *
                                *  SAMPLE PROGRAM FOR THE MODEL-550 OR A 16-BIT PROCESSOR
                                *  THIS PROGRAM UTILIZES SENSE STATUS LOOPS TO READ/WRITE
                                *  TO THE MODEL-550 WITH THE CURRENT LOOP INTERFACE
                                *
                                *  THE PROGRAM ACCEPTS 10 CHARACTERS AND OUTPUTS A CARRIAGE RETURN &
                                *  LINE FEED, THE SAME 10 CHARACTERS, AND ANOTHER CR,LF.
                                *
                                *  REGISTER ASSIGNMENTS
                                *
         0000 0001              DU       EQU   1                     DEVICE UNAVAILABLE = 1
         0000 0002              DEVADR   EQU   2                     SET DEVADR TO REGISTER 2
         0000 0003              STATUS   EQU   3                     SET STATUS TO REGISTER 3
         0000 0004              COUNT    EQU   4                     SET COUNT TO REGISTER 4
         0000 0005              WORK     EQU   5                     SET WORK TO REGISTER 5
         0000 0008              BSY      EQU   8                     BUSY = 8
                                *
                                *
 0000R   4820 006AR             START    LH    DEVADR,MODADR         DEVADR= MODEL-550 (CLI) ADDRESS
 0004R   DE20 006CR             OC1      OC    DEVADR,RDCMD          CLI IN READ MODE
 0008R   9023                   SENS1    SSR   DEVADR,STATUS         STATUS=CLI STATUS
 000AR   4210 0004R                      BTC   DU,OC1
 000ER   4380 0008R                      BFC   BSY,SENS1             CHECK NOT BUSY
 0012R   2440                            LIS   COUNT,0               COUN!=COUNT 10 KEYS
                                *
                                *  READ A TOTAL OF 10 CHARACTERS FROM MODEL-550 INTO MODBUF
                                *
 0014R   9D23                   SENS2    SSR   DEVADR,STATUS         STATUS= CLI STATUS
 0016R   4290 0014R                      BTC   DU+BSY,SENS2          CHECK DU,BUSY
 001AR   9B25                            RDR   DEVADR,WORK           WORK= 1 KEY READ FROM MODEL-550
 001CR   C450 007F                       NHI   WORK,X'7F'            ZERO OUT PARITY BIT
 0020R   D254 006ER                      STB   WORK,MODBUF(COUNT)    STORE BYTE IN MODBUF
 0024R   2641                            AIS   COUNT,1               ADD 1 TO INDEX COUNT
 0026R   C540 000A                       CLHI  COUNT,10              ARE THERE 10 BYTES READ YET
 002AR   4230 0014R                      BNE   SENS2                 IF NO, PRANCH TO SENS2 TO READ MORE
                                *
                                *  THIS SECTION OF THE PROGRAM OUTPUTS A CR TO MODEL-550
                                *
 002ER   DE20 0060R                      OC    DEVADR,WRTCMD         CLI IN WRITE MODE
 0032R   245D                            LIS   WORK,13               13 IS A CR
 0034R   9A25                            WDR   DEVADR,WORK           OUTPUT A CR
 0036R   9D23                            SSR   DEVADR,STATUS         SENSE STATUS
 0038R   2091                            BTBS  DU+BSY,1              BRANCH TO SSR COMMAND IF NOT READY
                                *
                                *  THIS SECTION OUTPUTS THE 10 BYTES OF DATA
                                *  FROM THE MODBUF TO MODEL-550
                                *
 003AR   2440                            LIS   COUNT,0               RESET INDEX COUNT TO ZERO
 003CR   DA24 006ER             WRITE    WD    DEVADR,MODBUF(COUNT) OUTPUT ONE CHARACTER TO MODEL-550
```

```
   SAMPLE PROGRAM WITH THE CURRENT LOOP INTERFACE             PAGE   2  16:02:07  12/03/79
 0040R  9D23                             SSR   DEVADR,STATUS        STATUS= CLI STATUS
 0042R  2091                             BTBS  DU+BSY,1             BRANCH TO SSR COMMAND IF NOT READY
 0044R  2641                             AIS   COUNT,1              ADD ONE TO INDEX COUNT
 0046R  C540 000A                        CLHI  COUNT,10             ARE THERE 10 BYTES WRITTEN YET
 004AR  4230 003CR                       BNE   WRITE                IF NO BRANCH TO WRITE TO WRITE MORE
                                 *
                                 *  THIS ROUTINE OUTPUTS A CR AND LINE FEED TO THE MODEL-550.
                                 *
 004ER  DE20 006DR                       OC    DEVADR,WRTCMD        CLI IN WRITE MODE
 0052R  2450                             LIS   WORK,13              13 IS A CR
 0054R  9A25                             WDR   DEVADR,WORK          OUTPUT A CR
 0056R  9D23                             SSR   DEVADR,STATUS        SENSE STATUS
 0058R  2091                             BTBS  DU+BSY,1             BRANCH TO SSR COMMAND IF NOT READY
 005AR  245A                             LIS   WORK,10              10 IS A LF
 005CR  9A25                             WDR   DEVADR,WORK          OUTPUT A LF
 005ER  9D23                             SSR   DEVADR,STATUS        SENSE STATUS
 0060R  2091                             BTBS  DU+BSY,1             BRANCH TO SSR COMMAND IF NOT READY
 0062R  C200 0066R               WAIT    LPSW  WAIT2
                                 *
                                 *  THIS SECTION SETS UP DATA CONSTANTS AND STORAGE AREAS
                                 *
 0066R  8000                     WAIT2   DCX   8000
 0068R  0062R                            DC    Z(WAIT)
 006AR  0002                     MODADR  DC    X'0002'              MODEL-550 (CLI) ADDRESS
 006CR  C4C8                     RDCMD   DC    X'C4C8'              CLI READ COMMAND, DISARM
        0000 006DR               WRTCMD  EQU   RDCMD+1              CLI WRITE COMMAND, DISARM
 006ER                           MODBUF  DS    10                   10 BYTE STORAGE FOR MODEL-550 BUFF
 0078R                                   END
```

```
    SAMPLE PROGRAM WITH THE CURRENT LOOP INTERFACE           PAGE   3  16:02:07  12/03/79
ASSEMBLED BY CAL 03-066R07-00   (32-BIT)

START OPTIONS:  T=16,ERLST

   NO CAL ERRORS
   NO CAL WARNINGS
    2 PASSES

ABSTOP     0000 0000
ADC        0000 0002
BSY        0000 0008        20*    27     33     47     55     66     70
COUNT      0000 0004        18*    28     36     37     38     52     53     56     57
DEVADR     0000 0002        16*    23     24     25     32     34     43     45     46     53     54     62     64
                            65     68     69
DU         0000 0001        15*    26     33     47     55     66     70
IMPTOP     0000 0078R
LADC       0000 0001
MODADR     0000 006AR       23     77*
MODBUF     0000 006ER       36     53     80*
OC1        0000 0004R       24*    26
PURETOP    0000 0000R
RDCMD      0000 006CR       24     78*    79
SENS1      0000 0008R       25*    27
SENS2      0000 0014R       32*    33     39
START      0000 0000R       23*
STATUS     0000 0003        17*    25     32     46     54     65     69
WAIT       0000 0062R       71*    76
WAIT2      0000 0066R       71     75*
WORK       0000 0005        19*    34     35     36     44     45     63     64     67     68
WRITE      0000 003CR       53*    58
WRTCMD     0000 006DR       43     62     79*
```

EXAMPLES USING AUTO DRIVER CHANNEL


START
SET UP SERVICE POINTER TABLE TO VECTOR ERROR ROUTINE ADDRESS
SET UP CHANNEL COMMAND BLOCK TO READ 10 BYTES
SET UP CHANNEL COMMAND BLOCK TO READ 10 BYTES
STORE TERMINATION ADDRESS IN CCB
SEND READ COMMAND
ENABLE EXTERNAL INTERRUPTS REG SET 15
WAIT
ADC ADDRESS 1
ADC FINISHES READ SEQUENCE
GOOD CHANNEL TERMINATION
NO
ERROR
YES
SET UP CCB TO WRITE
STORE TERMINATION ADDRESS IN CCB
CHANGE MODE TO CAUSE INTERRUPT
LOAD OLD PSW
ADC ADDRESS 2
ADC FINISHES WRITE SEQUENCES
GOOD CHANNEL TERMINATION
NO
ERROR
YES
B
B
DISARM DEVICE TO PREVENT QUEUED INTERRUPTS
LOAD OLD PSW
EOJ

NOTE:

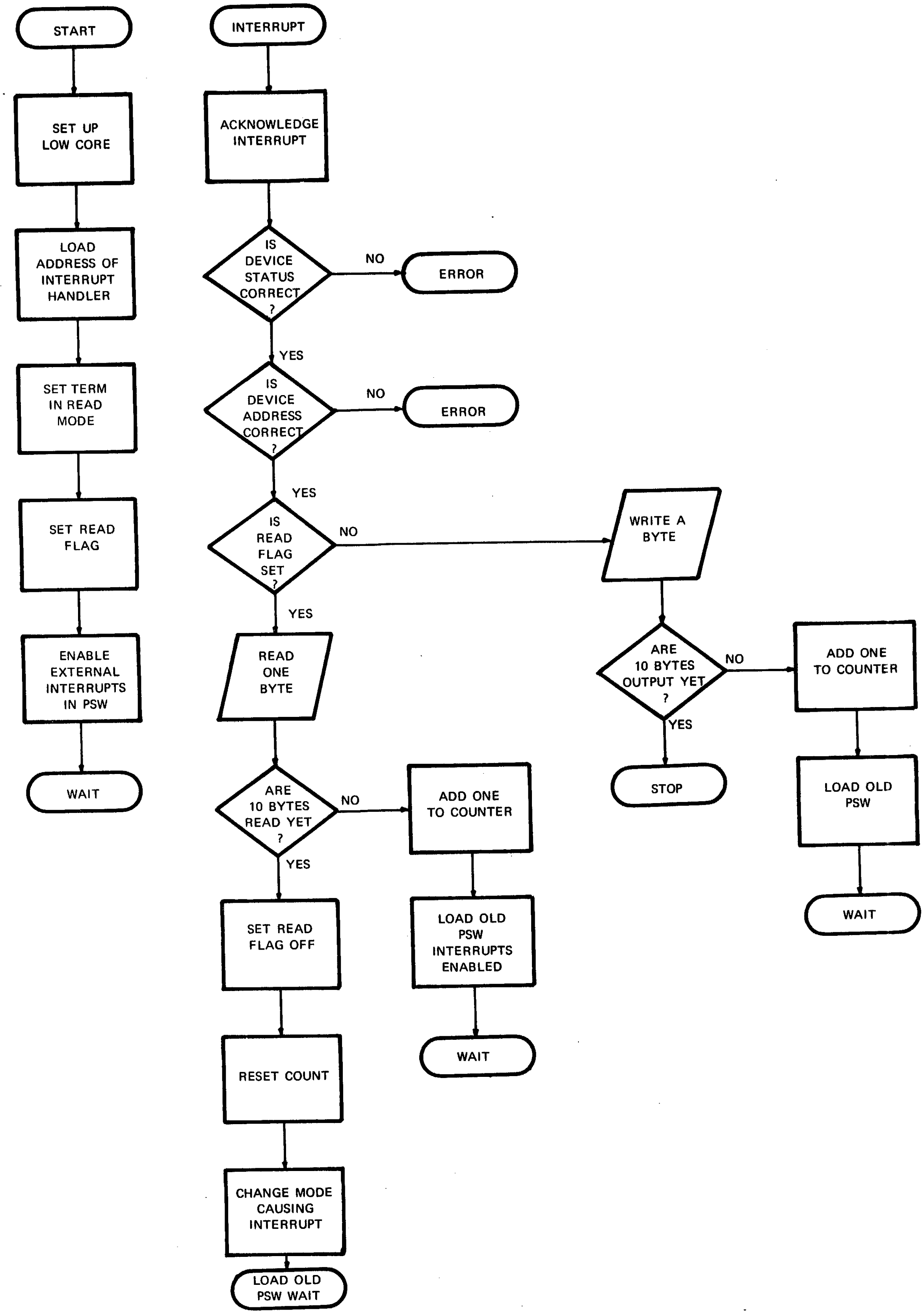
IN THIS FLOWCHART ADC REFERS TO AUTO DRIVER CHANNEL

```
        SAMPLE PROG WITH THE CURRENT LOOP INTERFACE                PAGE   1  14:13:08  11/30/79
     PROG= CLI16I     ASSEMBLED BY CAL 03-066R07-00   (32-BIT)

                                        1  CLI16I   PROG  SAMPLE PROG WITH THE CURRENT LOOP INTERFACE
                                        2                 CROSS
                                        3                 TARGT 16
                                        4                 WIDTH 120
                                        5  *
                                        6  *  SAMPLE PROGRAM FOR THE MODEL-550 ON A 16-BIT PROCESSOR
                                        7  *  THIS PROGRAM DEMONSTRATES THE READ/WRITE OPERATIONS FOR THE
                                        8  *  MODEL550 WITH THE CURRENT LOOP INTERFACE UNDER INTERRUPT CONTROL
                                        9  *  THE PROGRAM READS 10 CHARACTERS THEN OUTPUTS A CARRIAGE RETURN
                                       10  *  AND LINE FEED, ECHOS BACK THE TEN CHARACTERS, AND OUTPUTS
                                       11  *  ANOTHER CARRIAGE RETURN AND LINE FEED.
                                       12  *
               0000 0000               13  R0       EQU   0                   WORK REGISTER
               0000 0002               14  DEVADR   EQU   2                   DEVICE ADDRESS REGISTER
               0000 0004               15  COUNT    EQU   4
               0000 0005               16  WORK     EQU   5                   WORK REGISTER
               0000 0006               17  INT1     EQU   6                   INTERRUPT DEVICE REGISTER
               0000 0007               18  INTSTA   EQU   7                   INTERRUPT STATUS REGISTER
               0000 000E               19  R14      EQU   14                  WORK REGISTER
               0000 000F               20  PSWLOC   EQU   15                  PSW LOCATION SAVE REGISTER
                                       21  *
                                       22  *
        0000R  4820 0094R              23  START    LH    DEVADR,MODADR       DEVADR= CLI ADDRESS
        0004R  2451                    24           LIS   WORK,1
        0006R  4050 0098R              25           STH   WORK,FLAG           SET READ FLAG
                                       26  *
                                       27  *    SET UP LOW CORE
                                       28  *
        000AR  C8E0 0000               29           LHI   R14,0
        000ER  40E0 0040               30           STH   R14,X'40'           STORE ZERO IN OLD PSW
        0012R  40E0 0042               31           STH   R14,X'42'           STORE ZERO IN OLD LOC
        0016R  40E0 0044               32           STH   R14,X'44'           STORE ZERO IN NEW PSW
        001AR  C880 0030R              33           LHI   8,MODINT            LOAD INTERRUPT HANDLER INTO REG 8
        001ER  4080 0046               34           STH   8,X'46'             STORE ADDRESS OF HANDLER IN NEW LOC
                                       35  *
                                       36  *    SET MODEL-550 IN CORRECT MODE
                                       37  *
        0022R  950E                    38           EPSR  R0,R14
        0024R  DE20 0096R              39           OC    DEVADR,RDCMD        SET READ MODE
        0028R  C840 0000               40           LHI   COUNT,0             INITIALIZE COUNT
                                       41  *
                                       42  *    ENABLE INTERRUPTS IN PSW
        002CR  C200 009AR              43  WAIT1    LPSW  WAIT                WAIT FOR AN INTERRUPT
                                       44  *
                                       45  *    INTERRUPT HANDLER
                                       46  *
                                       47  *    ACKNOWLEDGE INTERRUPT AND MAKE SURE INTERRUPT IS FROM MODEL-550
                                       48  *
        0030R  9F67                    49  MODINT   ACKR  INT1,INTSTA         ACKNOWLEDGE INTERRUPT
        0032R  0562                    50           CLHR  INT1,DEVADR         IS INTERRUPT FROM MODEL-550 (CLI) ?
        0034R  4230 0082R              51           BNE   LOOK                IF NOT THE SAME GO TO ERROR ROUTINE
        0038R  C370 000D               52           THI   INTSTA,X'D'         TEST STATUS
        003CR  4230 0084R              53           BNZ   STSERR              IF STATUS NOT RIGHT GO TO ERROR ROUT
```

```
0040R  4850 0098R                  LH    WORK,FLAG          LOAD READ FLAG
0044R  4230 005ER                  BNZ   READBR             IF=1 READ, IF=0 WRITE
                         *
                         *    WRITE 10 BYTES OF DATA TO THE MODEL-550
                         *
0048R  DA24 0086R        WRTBR     WD    DEVADR,WRTBUF(COUNT)
004CR  2641                        AIS   COUNT,1            ADD ONE TO COUNTER
004ER  C540 000E                   CLHI  COUNT,14           COMPARE NUMBER OF BYTES TO 14
0052R  4330 005AR                  BE    DONE               IF FINISHED GO TO DONE
0056R  C200 0040                   LPSW  X'40'              LOAD OLD PSW
005AR  C200 009ER        DONE      LPSW  EOJ                FINISHED PUT PROCESSOR IN WAIT STATE
                         *
                         *    READ 10 BYTES OF DATA FROM BUF MODEL-550
                         *
005ER  DB24 0088R        READBR    RD    DEVADR,MODBUF(COUNT) * READ ONE BYTE IN MODBUF
0062R  2641                        AIS   COUNT,1            ADD ONE TO COUNTER
0064R  C540 000A                   CLHI  COUNT,10           COMPARE TO 10
0068R  4330 0070R                  BE    GOWRITE            IF FINISHED GO TO WRITE A BYTE
006CR  C200 0040                   LPSW  X'40'              LOAD OLD PSW
0070R  2450              GOWRITE   LIS   WORK,0             LOAD ZERO INTO WORK
0072R  4050 0098R                  STH   WORK,FLAG          RESET FLAG TO WRITE
0076R  C840 0000                   LHI   COUNT,0            INITIALIZE COUNT TO ZERO
007AR  DE20 0097R                  OC    DEVADR,WRTCMD      CLI IN WRITE MODE
007ER  C200 0040                   LPSW  X'40'              LOAD OLD PSW
                         *
                         *    NOT A MODEL-550 (CLI) INTERRUPT
                         *
0082R  2200              LOOK      DCX   2200               STOP
                         *
                         *    STATUS ERROR FROM MODEL-550 (CLI)
                         *
0084R  2200              STSERR    DCX   2200               STOP
                         *
                         *    SET UP DATA CONSTANTS AND STORAGE AREAS
                         *
0086R  0D0A              WRTBUF    DC    X'0D0A'            OD=CARRIAGE RETURN, OA=LINE FEED
0088R                    MODBUF    DS    10                 RESERVE STORAGE FOR 10 BYTES
0092R  0D0A                        DC    X'0D0A'            OD=CARRIAGE RETURN, OA=LINE FEED
0094R  0002              MODADR    DC    X'0002'            MODEL-550 (CLI) ADDRESS
0096R  6458              RDCMD     DC    X'6458'            CLI SET READ,UNBLOCK,ENABLE
       0000 0097R        WRTCMD    EQU   RDCMD+1            CLI SET WRITE, ENABLE
0098R  0000              FLAG      DCX   0
009AR  C000              WAIT      DC    X'C000'            SET EXTERNAL INTERRUPT
009CR  002CR                       DC    Z(WAIT1)
009ER  8000              EOJ       DC    X'8000'            PUT PROCESSOR IN WAIT STATE
00A0R  005AR                       DC    Z(DONE)
00A2R                              END
```

```
    SAMPLE PROG WITH THE CURRENT LOOP INTERFACE              PAGE   3  14:13:08  11/30/79
ASSEMBLED BY CAL 03-066R07-00   (32-BIT)

START OPTIONS:  T=16,ERLST

   NO CAL ERRORS
   NO CAL WARNINGS
    2 PASSES

 ABSTOP     0000 0000
 ADC        0000 0002
 COUNT      0000 0004         15*    40     59     60     61     68     69     70     75
 DEVADR     0000 0002         14*    23     39     50     59     68     76
 DONE       0000 005AR        62     64*    99
 EOJ        0000 009ER        64     98*
 FLAG       0000 0098R        25     54     74     95*
 GOWRITE    0000 0070R        71     73*
 IMPTOP     0000 00A2R
 INT1       0000 0006         17*    49     50
 INTSTA     0000 0007         18*    49     52
 LADC       0000 0001
 LOOK       0000 0082R        51     81*
 MODADR     0000 0094R        23     92*
 MODBUF     0000 0088R        68     90*
 MODINT     0000 0030R        33     49*
 PSWLOC     0000 000F         20*
 PURETOP    0000 0000R
 R0         0000 0000         13*    38
 R14        0000 000E         19*    29     30     31     32     38
 RDCMD      0000 0096R        39     93*    94
 READBR     0000 005ER        55     68*
 START      0000 0000R        23*
 STSERR     0000 0084R        53     85*
 WAIT       0000 009AR        43     96*
 WAIT1      0000 002CR        43*    97
 WORK       0000 0005         16*    24     25     54     73     74
 WRTBR      0000 0048R        59*
 WRTBUF     0000 0086R        59     89*
 WRTCMD     0000 0097R        76     94*
```

IMMEDIATE INTERRUPTS

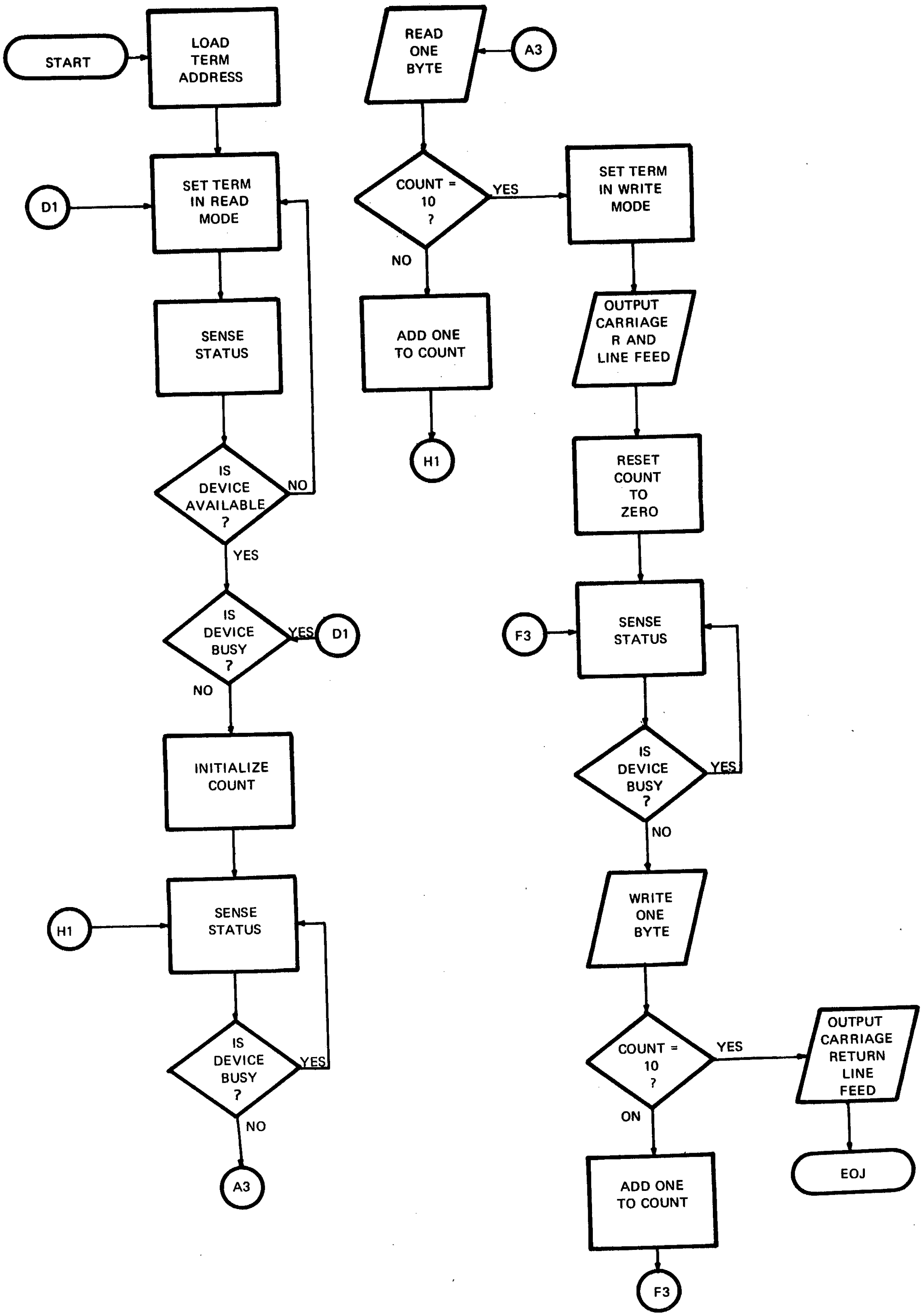


```
      SAMPLE PROGRAM WITH THE 5/16 MICRO I/O BUS                    PAGE   1  14:19:43  12/03/79
PROG= MIOBIN     ASSEMBLED BY CAL 03-066R07-00   (32-BIT)

                                     1  MIOBIN   PROG  SAMPLE PROGRAM WITH THE 5/16 MICRO I/O BUS
                                     2                 CROSS
                                     3                 TARGT 16
                                     4                 WIDTH 120
                                     5  *
                                     6  *    SAMPLE PROGRAM FOR THE MODEL-550 ON A 16-BIT PROCESSOR
                                     7  *    THIS PRPGRAM DEMONSTRATES THE READ/WRITE OPERATIONS FOR THE
                                     8  *    MODEL-550 WITH THE MICRO-I/O-BUS UNDER INTERRUPT CONTROL
                                     9  *    THE PROGRAM READS 10 CHARACTERS THEN OUTPUTS A CARRIAGE RETURN
                                    10  *    AND LINE FEED. ECHOS BACK THE TEN CHARACTERS, AND OUTPUTS
                                    11  *    ANOTHER CARRIAGE RETURN AND LINE FEED.
                                    12  *
              0000 0000             13  RD       EQU   0                     WORK REGISTER
              0000 0002             14  DEVADR   EQU   2                     DEVICE ADDRESS REGISTER
              0000 0004             15  COUNT    EQU   4
              0000 0005             16  WORK     EQU   5                     WORK REGISTER
              0000 0006             17  INT1     EQU   6                     INTERRUPT DEVICE REGISTER
              0000 0007             18  INTSTA   EQU   7                     INTERRUPT STATUS REGISTER
              0000 0008             19  R8       EQU   8
              0000 000E             20  R14      EQU   14                    WORK REGISTER
              0000 000F             21  PSWLOC   EQU   15                    PSW LOCATION SAVE REGISTER
                                    22  *
                                    23  *
0000R    4820 0096R                 24  START    LH    DEVADR,MODADR         DEVADR= MODEL-550 (CLI) ADDRESS
0004R    2451                       25           LIS   WORK,1
0006R    4050 009AR                 26           STH   WORK,FLAG             SET READ FLAG
                                    27  *
                                    28  *    SET UP LOW CORE
                                    29  *
000AR    C8E0 0000                  30           LHI   R14,0
000ER    40E0 0040                  31           STH   R14,X'40'             STORE ZERO IN OLD PSW
0012R    40E0 0042                  32           STH   R14,X'42'             STORE ZERO IN OLD LOC
0016R    40E0 0044                  33           STH   R14,X'44'             STORE ZERO IN NEW PSW
001AR    C880 0030R                 34           LHI   R8,MODINT             LOAD INTERRUPT HANDLER INTO REG 8
001ER    4080 0046                  35           STH   R8,X'46'              STORE ADDRESS OF HANDLER IN NEW LOC
                                    36  *
                                    37  *    SET MODEL-550 IN CORRECT MODE
                                    38  *
0022R    950E                       39           EPSR  RD,R14
0024R    DE20 0098R                 40           OC    DEVADR,RDCMD          SET READ MODE
0028R    C840 0000                  41           LHI   COUNT,0               INITIALIZE COUNT
                                    42  *
                                    43  *    ENABLE INTERRUPTS IN PSW
                                    44  *
002CR    C200 009CR                 45  WAIT1    LPSW  WAIT                  WAIT FOR AN INTERRUPT
                                    46  *
                                    47  *    INTERRUPT HANDLER
                                    48  *
                                    49  *    ACKNOWLEDGE INTERRUPT AND MAKE SURE INTERRUPT IS FROM MODEL-550
                                    50  *
0030R    9F67                       51  MODINT   ACKR  INT1,INTSTA           ACKNOWLEDGE INTERRUPT
0032R    C560 0002                  52           CLHI  INT1,DEVADR           IS INTERRUPT FROM MODEL-550  ?
0036R    4230 0084R                 53           BNE   LOOK                  IF NOT THE SAME GO TO ERROR ROUTINE
```

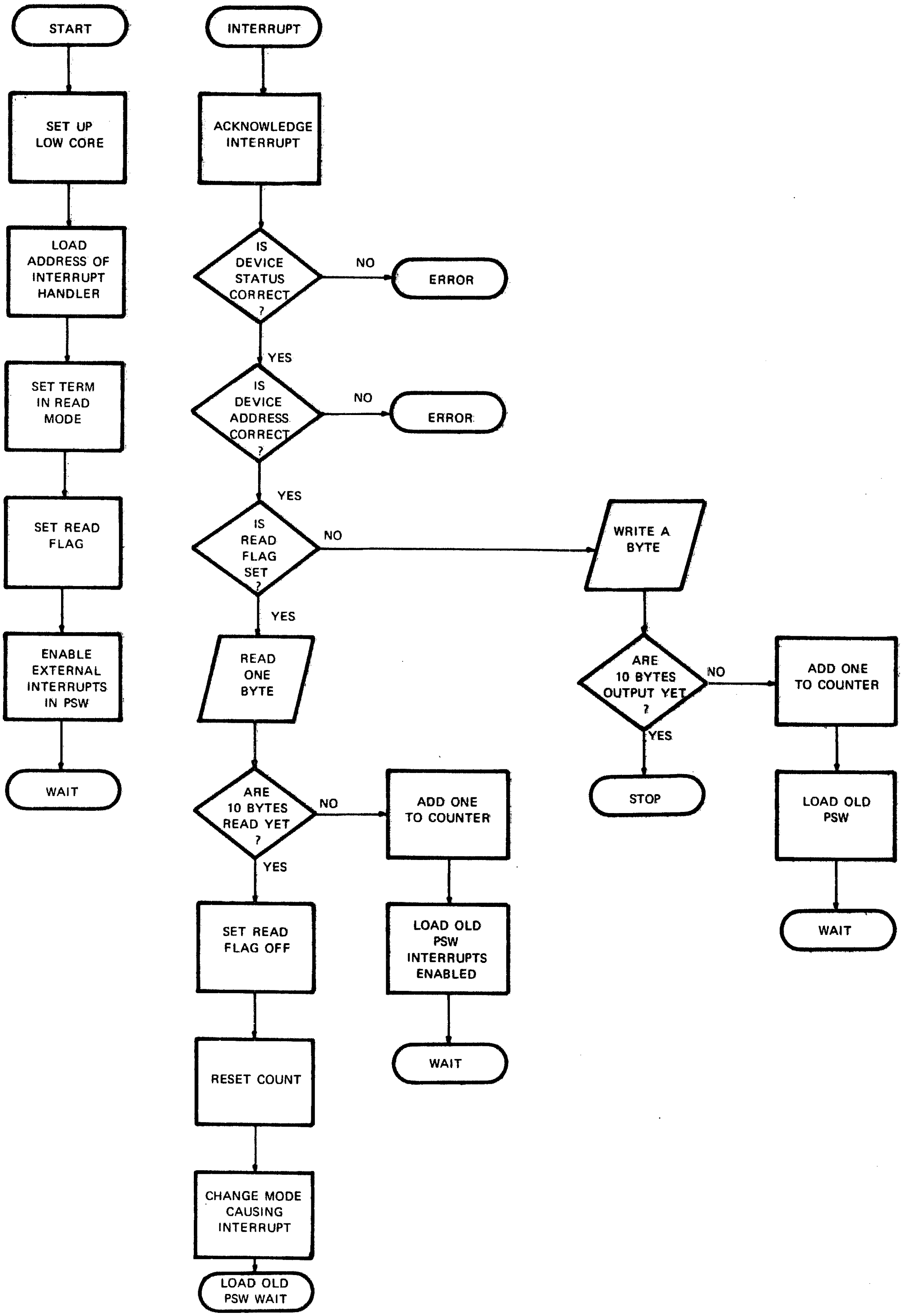
```
        SAMPLE PROGRAM WITH THE 5/16 MICRO I/O BUS             PAGE   2  14:19:43  12/03/79
003AR  C370 000D            54          THI    INTSTA,X'D'          TEST STATUS
003ER  4230 0086R           55          BNZ    STSERR               IF STATUS NOT RIGHT GO TO ERROR ROUT
0042R  4850 009AR           56          LH     WORK,FLAG            LOAD READ FLAG
0046R  4230 0060R           57          BNZ    READBR               IF=1 READ, IF=0 WRITE
                            58 *
                            59 *   WRITE 10 BYTES OF DATA TO MODEL-550.
                            60 *
004AR  DA24 0088R           61 WRTBR    WD     DEVADR,WRTBUF(COUNT)
004ER  2641                 62          AIS    COUNT,1              ADD ONE TO COUNTER
0050R  C540 000E            63          CLHI   COUNT,14             COMPARE NUMBER OF BYTES TO 14
0054R  4330 005CR           64          BE     DONE                 IF FINISHED GO TO DONE
0058R  C200 0040            65          LPSW   X'40'                LOAD OLD PSW
005CR  C200 00A0R           66 DONE     LPSW   EOJ                  FINISHED PUT PROCESSOR IN WAIT STATE
                            67 *
                            68 *   READ 10 BYTES OF DATA FROM BUF MODEL-550
                            69 *
0060R  DB24 008AR           70 READBR   RD     DEVADR,MODBUF(COUNT) READ ONE BYTE IN MODBUF
0064R  2641                 71          AIS    COUNT,1              ADD ONE TO COUNTER
0066R  C540 000A            72          CLHI   COUNT,10             COMPARE TO 10
006AR  4330 0072R           73          BE     GOWRITE              IF FINISHED GO TO WRITE A BYTE
006ER  C200 0040            74          LPSW   X'40'                LOAD OLD PSW
0072R  2450                 75 GOWRITE  LIS    WORK,0               LOAD ZERO INTO WORK
0074R  4050 009AR           76          STH    WORK,FLAG            RESET FLAG TO WRITE
0078R  C840 0000            77          LHI    COUNT,0              INITIALIZE COUNT TO ZERO
007CR  DE20 0099R           78          OC     DEVADR,WRTCMD        MICRO I/O IN WRITE MODE
0080R  C200 0040            79          LPSW   X'40'                LOAD OLD PSW
                            80 *
                            81 *   NOT A MODEL-550 INTERRUPT
                            82 *
0084R  2200                 83 LOOK     DCX    2200                 STOP
                            84 *
                            85 *   STATUS ERROR FROM MODEL-550
                            86 *
0086R  2200                 87 STSERR   DCX    2200                 STOP
                            88 *
                            89 *   SET UP DATA CONSTANTS AND STORAGE AREAS
                            90 *
0088R  0D0A                 91 WRTBUF   DC     X'0D0A'              OD=CARRIAGE RETURN, OA=LINE FEED
008AR                       92 MODBUF   DS     10                   RESERVE STORAGE FOR 10 BYTES
0094R  0D0A                 93          DC     X'0D0A'              OD=CARRIAGE RETURN, OA=LINE FEED
0096R  00C0                 94 MODADR   DC     X'00C0'              MODEL-550 (MICRO-IO-BUS) ADDRESS
0098R  9232                 95 RDCMD    DC     X'9232'              MICRO I/O SET READ,UNBLOCK,ENABLE
       0000 0099R           96 WRTCMD   EQU    RDCMD+1              MICRO I/O SET WRITE, ENABLE
009AR  0000                 97 FLAG     DCX    0
009CR  C000                 98 WAIT     DC     X'C000'              SET EXTERNAL INTERRUPT
009ER  002CR                99          DC     Z(WAIT1)
00A0R  8000                100 EOJ      DC     X'8000'              PUT PROCESSOR IN WAIT STATE
00A2R  005CR               101          DC     Z(DONE)
00A4R                      102          END
```

```
    SAMPLE PROGRAM WITH THE 5/16 MICRO I/O BUS                PAGE   3  14:19:43  12/03/79
ASSEMBLED BY CAL 03-066R07-00    (32-BIT)

START OPTIONS:  T=16,ERLST

   NO CAL ERRORS
   NO CAL WARNINGS
    2 PASSES

 ABSTOP     0000 0000
 ADC        0000 0002
 COUNT      0000 0004      15*     41     61     62     63     70     71     72     77
 DEVADR     0000 0002      14*     24     40     52     61     70     78
 DONE       0000 005CR     64      66*   101
 EOJ        0000 00A0R     66     100*
 FLAG       0000 009AR     26      56     76     97*
 GOWRITE    0000 0072R     73      75*
 IMPTOP     0000 00A4R
 INT1       0000 0006      17*     51     52
 INTSTA     0000 0007      18*     51     54
 LADC       0000 0001
 LOOK       0000 0084R     53      83*
 MODADR     0000 0096R     24      94*
 MODBUF     0000 008AR     70      92*
 MODINT     0000 0030R     34      51*
 PSWLOC     0000 000F      21*
 PURETOP    0000 0000R
 R14        0000 000E      20*     30     31     32     33     39
 R8         0000 0008      19*     34     35
 RD         0000 0000      13*     39
 RDCMD      0000 0098R     40      95*    96
 READBR     0000 0060R     57      70*
 START      0000 0000R     24*
 STSERR     0000 0086R     55      87*
 WAIT       0000 009CR     45      98*
 WAIT1      0000 002CR     45*     99
 WORK       0000 0005      16*     25     26     56     75     76
 WRTBR      0000 004AR     61*
 WRTBUF     0000 0088R     61      91*
 WRTCMD     0000 0099R     78      96*
```

UNDER STATUS SENSING



IMMEDIATE INTERRUPTS

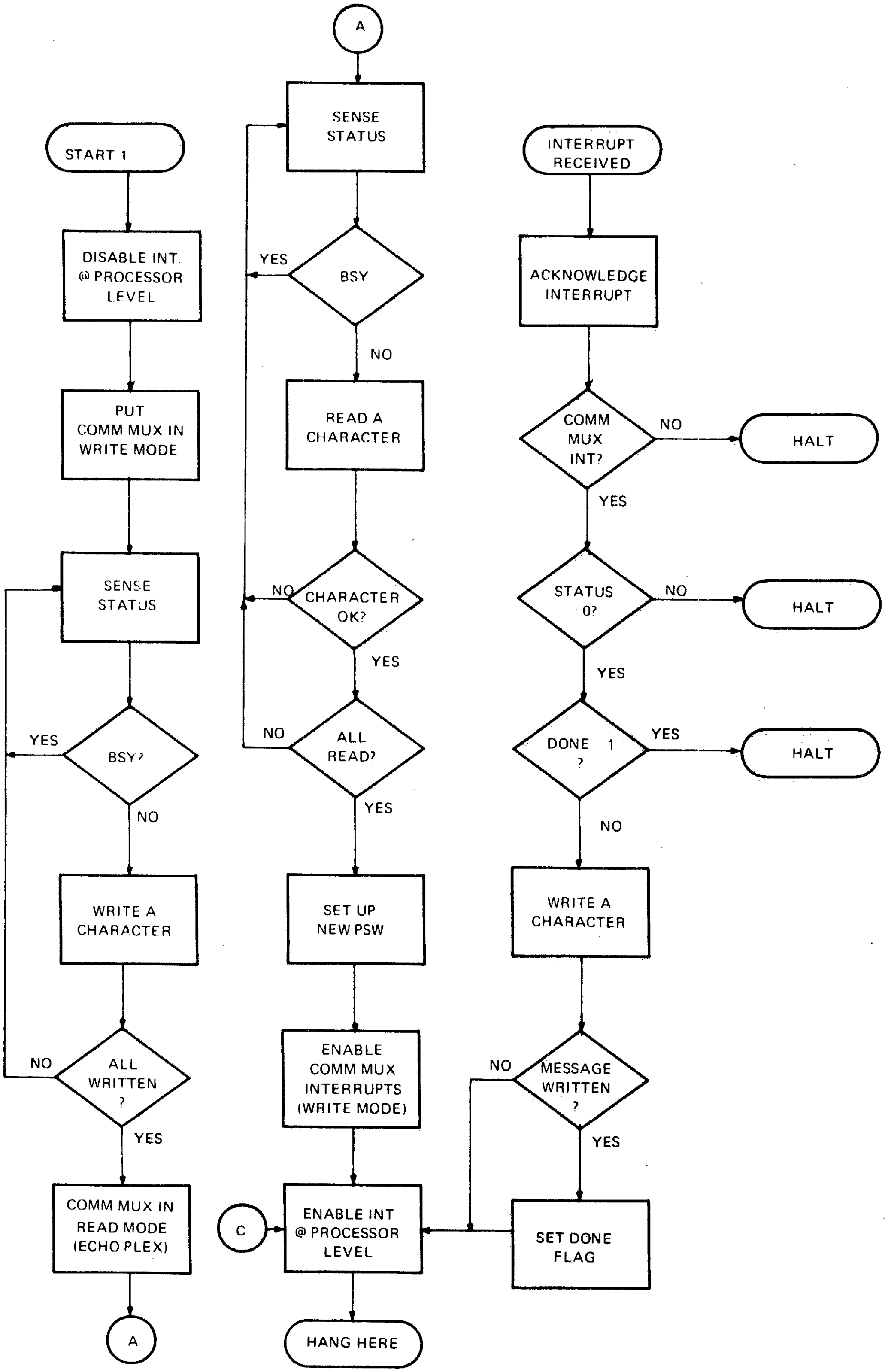


```
          SAMPLE PROGRAM USING THE MICRO I/O BUS                     PAGE   1  09:12:47  11/29/79
      PROG= MIOBSS     ASSEMBLED BY CAL 03-066R07-00    (32-BIT)
                                          1  MIOBSS    PROG  SAMPLE PROGRAM USING THE MICRO I/O BUS
                                          2                CROSS
                                          3                TARGT 16
                                          4                WIDTH 120
                                          5  *
                                          6  *     SAMPLE PROGRAM FOR THE MODEL-550 ON A 16-BIT PROCESSOR
                                          7  *     THIS PROGRAM UTILIZES SENSE STATUS LOOPS TO READ/WRITE
                                          8  *     TO THE MODEL-550 WITH THE MICRO-I/O-BUS
                                          9  *
                                         10  *     THE PROGRAM ACCEPTS 10 CHARACTERS AND OUTPUTS A CARRIAGE RETURN
                                         11  *     LINE FEED, THE SAME 10 CHARACTERS, AND ANOTHER CR,LF.
                                         12  *
                                         13  *     REGISTER ASSIGNMENTS
                                         14  *
                 0000 0001               15  DU        EQU    1                       DEVICE UNAVAILABLE = 1
                 0000 0002               16  DEVADR    EQU    2                       SET DEVADR TO REGISTER 2
                 0000 0003               17  STATUS    EQU    3                       SET STATUS TO REGISTER 3
                 0000 0004               18  COUNT     EQU    4                       SET COUNT TO REGISTER 4
                 0000 0005               19  WORK      EQU    5                       SET WORK TO REGISTER 5
                 0000 0008               20  BSY       EQU    8                       BUSY = 8
                                         21  *
                                         22  *
      0000R  4820 0072R                  23  START     LH     DEVADR,MODADR           DEVADR=MODEL-550 (CLI) ADDRESS
      0004R  DE20 0074R                  24  OC1       OC     DEVADR,RDCMD            CLI IN READ MODE
      0008R  9D23                        25  SENS1     SSR    DEVAUR,STATUS           STATU S=CLI STATUS
      000AR  4210 0004R                  26            BTC    DU,OC1
      000ER  4380 0008R                  27            BFC    BSY,SENS1               CHECK NOT BUSY
      0012R  2440                        28            LIS    COUNT,0                 COUNT=COUNT 10 KEYS
                                         29  *
                                         30  *     READ A TOTAL OF 10 CHARACTERS FROM MODEL-550 INTO MODBUF
                                         31  *
      0014R  9D23                        32  SENS2     SSR    DEVADR,STATUS           STATUS= CLI STATUS
      0016R  4290 0014R                  33            BTC    DU+BSY,SENS2            CHECK DU,BUSY
      001AR  9B25                        34            RDR    DEVADR,WORK             WORK= 1 KEY READ FROM THE MODEL-550
      001CR  CA50 007F                   35            AHI    WORK,X'7F'              ZERO OUT PARITY BIT
      0020R  D254 0076R                  36            STB    WORK,MODBUF(COUNT)      STORE BYTE IN MODBUF
      0024R  2641                        37            AIS    COUNT,1                 ADD 1 TO INDEX COUNT
      0026R  C540 000A                   38            CLHI   COUNT,10                ARE THERE 10 BYTES READ YET
      002AR  4230 0014R                  39            BNE    SENS2                   IF NO BRANCH TO SENS2 TO READ MORE
                                         40  *
                                         41  *     THIS SECTION OF THE PROGRAM OUTPUTS A CR TO THE MODEL-550.
                                         42  *
      002ER  DE20 0075R                  43            OC     DEVADR,WRTCMD           CLI IN WRITE MODE
      0032R  245D                        44            LIS    WORK,13                 13 IS A CR
      0034R  9A25                        45            WDR    DEVADR,WORK             OUTPUT A CR
      0036R  9D23                        46            SSR    DEVADR,STATUS           SENSE STATUS
      0038R  2091                        47            BTBS   DU+BSY,1                BRANCH TO SSR COMMAND IF NOT READY
      003AR  245A                        48            LIS    WORK,10                 10 IS A LINE FEED
      003CR  9A25                        49            WDR    DEVADR,WORK             OUTPUT A LINE FEED
      003ER  9D23                        50            SSR    DEVADR,STATUS           SENSE STATUS
      0040R  2091                        51            BTBS   DU+BSY,1                BRANCH TO SSR COMMAND IF NOT READY
                                         52  *
                                         53  *     THIS SECTION OUTPUTS THE 10 BYTES OF
```

```
     SAMPLE PROGRAM USING THE MICRO I/O BUS                        PAGE   2  09:12:47  11/29/79
                                  54  *   DATA FROM MODBUF TO THE MODEL-550.
                                  55  *
0042R  2440                       56            LIS   COUNT,0                RESLT INDEX COUNT TO ZERO
0044R  DA24 0076R                 57  WRITE     WD    DEVADR,MODBUF(COUNT) OUTPUT ONE CHARACTER TO MODEL-550
0048R  9D23                       58            SSR   DEVADR,STATUS          STATUS= CLI STATUS
004AR  2091                       59            BTBS  DU+BSY,1               BRANCH TO SSR COMMAND IF NOT READY
004CR  2641                       60            AIS   COUNT,1                ADD ONE TO INDEX COUNT
004ER  C540 000A                  61            CLHI  COUNT,10               ARE THERE 10 BYTES WRITTEN YET
0052R  4230 0044R                 62            BNE   WRITE                  IF NO, BRANCH TO 'WRITE', TO CONTIN
                                  63  *
                                  64  *   THIS ROUTINE OUTPUTS A CR AND LINE FEED TO THE MODEL-550.
                                  65  *
0056R  DE20 0075R                 66            OC    DEVADR,WRTCMD          CLI IN WRITE MODE
005AR  245D                       67            LIS   WORK,13                13 IS A CR
005CR  9A25                       68            WDR   DEVADR,WORK            OUTPUT A CR
005ER  9D23                       69            SSR   DEVADR,STATUS          SENSE STATUS
0060R  2091                       70            BTBS  DU+BSY,1               BRANCH TO SSR COMMAND IF NOT READY
0062R  245A                       71            LIS   WORK,10                LINE FEED TO MODEL-550
0064R  9A25                       72            WDR   DEVADR,WORK            OUTPUT A LINE FEED
0066R  9D23                       73            SSR   DEVADR,STATUS          SENSE STATUS
0068R  2091                       74            BTBS  DU+BSY,1               BRANCH TO SSR COMMAND IF NOT READY
006AR  C200 006ER                 75  WAIT      LPSW  WAIT2
                                  76  *
                                  77  *   THIS SECTION SETS UP DATA CONSTANTS AND STORAGE AREAS.
                                  78  *
006ER  8000                       79  WAIT2     DCX   8000
0070R  006AR                      80            DC    Z(WAIT)
0072R  00C0                       81  MODADR    DC    X'00C0'                MODEL-550 (CLI) ADDRESS
0074R  9212                       82  RDCMD     DC    X'9212'                CLI READ COMMAND, ENABLE
       0000 0075R                 83  WRTCMD    EQU   RDCMD+1                CLI WRITE COMMAND, DISABLE
0076R                             84  MODBUF    DS    10                     10 BYTE STORAGE FOR MODEL-1100 BUFF
0080R                             85            END
```

```
     SAMPLE PROGRAM USING THE MICRO I/O BUS                  PAGE   3  09:12:47  11/29/79

 ASSEMBLED BY CAL 03-066R07-00   (32-BIT)

 START OPTIONS:  T=16,ERLST

    NO CAL ERRORS
    NO CAL WARNINGS
     2 PASSES

  ABSTOP     0000 0000
  ADC        0000 0002
  BSY        0000 0008         20*     27     33     47     51     59     70     74
  COUNT      0000 0004         18*     28     36     37     38     56     57     60     61
  DEVADR     0000 0002         16*     23     24     25     32     34     43     45     46     49     50     57     58
                               66      68     69     72     73
  DU         0000 0001         15*     26     33     47     51     59     70     74
  IMPTOP     0000 0080R
  LADC       0000 0001
  MODADR     0000 0072R        23      81*
  MODBUF     0000 0076R        36      57     84*
  OC1        0000 0004R        24*     26
  PURETOP    0000 0000R
  RDCMD      0000 0074R        24      82*    83
  SENS1      0000 0008R        25*     27
  SENS2      0000 0014R        32*     33     39
  START      0000 0000R        23*
  STATUS     0000 0003         17*     25     32     46     50     58     69     73
  WAIT       0000 006AR        75*     80
  WAIT2      0000 006ER        75      79*
  WORK       0000 0005         19*     34     35     36     44     45     48     49     67     68     71     72
  WRITE      0000 0044R        57*     62
  WRTCMD     0000 0075R        43      66     83*
```

COMM MUX


A
SENSE STATUS
START 1
INTERRUPT RECEIVED
DISABLE INT @ PROCESSOR LEVEL
YES
BSY
ACKNOWLEDGE INTERRUPT
NO
PUT COMM MUX IN WRITE MODE
READ A CHARACTER
COMM MUX INT?
NO
HALT
YES
SENSE STATUS
NO
CHARACTER OK?
STATUS 0?
NO
HALT
YES
YES
YES
BSY?
NO
ALL READ?
DONE 1 ?
YES
HALT
NO
YES
NO
WRITE A CHARACTER
SET UP NEW PSW
WRITE A CHARACTER
NO
ALL WRITTEN ?
ENABLE COMM MUX INTERRUPTS (WRITE MODE)
NO
MESSAGE WRITTEN ?
YES
YES
COMM MUX IN READ MODE (ECHO-PLEX)
C
ENABLE INT @ PROCESSOR LEVEL
SET DONE FLAG
A
HANG HERE

```
       COMM MUX PROGRAMMING EXAMPLES FOR 16- BIT PROCESSOR       PAGE   1  15:01:46  12/03/79
PROG=           ASSEMBLED BY CAL 03-066R07-00   (32-BIT)
                                   1          PROG  COMM MUX PROGRAMMING EXAMPLES FOR 16- BIT PROCESSOR          CM160010
                                   2          CROSS                                                              CM160020
                                   3          TARGT 16                                                           CM160030
                                   4          WIDTH 120                                                          CM160040
                                   5  *                                                                          CM160050
                                   6  * THESE EXAMPLES DEMONSTRATE SEQUENCES TO PROGRAM THE                      CM160060
                                   7  * COMMUNICATIONS MUX IN VARIOUS ENVIRONMENTS. THE TERMINAL AND             CM160070
                                   8  * THE  COMM MUX INTERFACE SHOULD BE STRAPPED/CONNECTED                     CM160080
                                   9  * AS MENTIONED IN THE INDIVIDUAL EXAMPLES.                                 CM160090
                                  10  *                                                                          CM160100
                                  11  * REGISTER ASSIGNMENTS                                                     CM160110
                                  12  *                                                                          CM160120
     0000 0001                    13  MSG       EQU   1                MESSAGE START ADDRESS                     CM160130
     0000 0004                    14  CHAR      EQU   4                CHARACTER BEING TRANSFERRED               CM160140
     0000 0005                    15  WORK      EQU   5                WORK REGISTER                             CM160150
     0000 0006                    16  DONE      EQU   6                EXAMPLE DONE FLAG                         CM160160
     0000 0007                    17  DEV       EQU   7                MUX HDX ADDRESS                           CM160170
     0000 0008                    18  STAT      EQU   8                MUX STATUS                                CM160180
     0000 0009                    19  REPEAT    EQU   9                EXAMPLE START ADDRESS                     CM160190
     0000 000F                    20  LINK      EQU   15               LINK REGISTER                             CM160200
                                  21  *                                                                          CM160210
                                  22  *                                                                          CM160220
     0000 0004                    23  EX        EQU   4                EX BIT IN STATUS BYTE IS ONE              CM160230
     0000 0008                    24  BSY       EQU   8                BSY BIT IN STATUS BYTE IS ONE             CM160240
     0000 0008                    25  CARRY     EQU   8                CARRY FLAG IN PSW IS ONE                  CM160250
                                  26  *                                                                          CM160260
```

```
COMM MUX PROGRAMMING EXAMPLES FOR 16- BIT PROCESSOR       PAGE   2  15:01:46  12/03/79

LOCAL TERMINAL, HALF-DUPLEX  COMM MUX OPERATION

                                 *                                                                 CM160280
                                 *   TTY, CRT OR GDT SHOULD BE INTERFACED THROUGH MUX HDX INTERFACE   CM160290
                                 *   CRT/GDT SHOULD BE STRAPPED FOR HALF-DUPLEX OPERATION          CM160300
                                 *   THE FOLLOWING EXAMPLE IS FOR 16-BIT PROCESSOR                 CM160310
                                 *   START EXECUTION @ START1                                      CM160320
                                 *                                                                 CM160330
                                 *   WRITE CHARACTERS & READ KEYS USING SENSE STATUS LOOP          CM160340
                                 *   WRITE CHARACTERS UNDER INTERRUPT CONTROL                      CM160350
                                 *                                                                 CM160360
                                 *                                                                 CM160370
0000R  0766                      START1    XAR   DONE,DONE                                         CM160380
0002R  9556                                EPSR  WORK,DONE          DISABLE INT @ PROCESSOR LEVEL  CM160390
0004R  C890 0000R                          LDAI  REPEAT,START1                                     CM160400
0008R  4870 00BER                          LH    DEV,DEVADR         GET  COMM MUX DEVICE ADDRESS   CM160410
000CR  DE70 00C0R                          OC    DEV,SECOND         SET  MUX AS PER SECOND COMMAND CM160420
0010R  DE70 00C1R                          OC    DEV,DISWRT         WRITE MODE                     CM160430
0014R  C810 00C4R                          LDAI  MSG,MSG1           (MSG) = MESSAGE START ADDRESS  CM160440
                                 *                                                                 CM160450
0018R  D341 0000                 EXMP1A    LB    CHAR,0(MSG)                                       CM160460
001CR  41F0 00A2R                          BAL   LINK,OUTCHR        OUTPUT A CHARACTER             CM160470
0020R  2611                                AIS   MSG,1                                             CM160480
0022R  C510 00D8R                          CLAI  MSG,MSG1END                                       CM160490
0026R  2087                                BLS   EXMP1A             LOOP TILL 'TYPE 1234567890' OUTPUT   CM160500
0028R  DE70 00C2R                          OC    DEV,DISRD          SELECT READ MODE (ECHO-PLEX)   CM160510
002CR  9B75                                RDR   DEV,WORK           DUMMY READ TO SET BSY          CM160520
002ER  41F0 00AER                          BAL   LINK,DELAY         LINE TURN AROUND DELAY         CM160530
0032R  0711                                XAR   MSG,MSG                                           CM160540
                                 *                                                                 CM160550
0034R  41F0 0092R                EXMP1B    BAL   LINK,READ          READ A KEY CODE WHEN DEPRESSED CM160560
0038R  D441 00CBR                          CLB   CHAR,TYPED(MSG)    COMPARE WITH THE EXPECTED      CM160570
003CR  4230 003CR                          BNE   *                                                 CM160580
0040R  2611                                AIS   MSG,1                                             CM160590
0042R  C540 000A                           CLAI  MSG,10                                            CM160600
0046R  2089                                BLS   EXMP1B             LOOP TILL 10 DIGITS TYPED      CM160610
                                 *                                                                 CM160620
                                 * WRITE UNDER INTERRUPT CONTROL                                   CM160630
                                 *                                                                 CM160640
0048R  C850 006AR                          LDAI  WORK,INT                                          CM160650
004CR  4060 0044                           STA   DONE,X'44'         SET UP NEW PSW FOR             CM160660
0050R  4050 0046                           STA   WORK,X'46'         IMMEDIATE INTERRUPT            CM160670
0054R  C810 00D8R                          LDAI  MSG,MSG2                                          CM160680
0058R  DE70 00C3R                          OC    DEV,ENWRT          ENABLE INT  @ MUX LEVEL 1 PENDING   CM160690
005CR  41F0 00AER                          BAL   LINK,DELAY         LINE TURNAROUND DELAY          CM160700
0060R  C850 4000                 EXMP1C    LDAI  WORK,X'4000'                                      CM160710
0064R  9545                                EPSR  CHAR,WORK          ENABLE INT @ PROCESSOR LEVEL   CM160720
0066R  4300 0066R                          B     *                                                 CM160730
                                 *                                                                 CM160740
                                 * IMMEDIATE INTERRUPT IS RECEIVED                                 CM160750
                                 *                                                                 CM160760
006AR  9F58                      INT       AIR   WORK,STAT          ACKNOWLEDGE AN INTERRUPT       CM160770
006CR  0557                                CLAR  WORK,DEV           COMPARE DEVICE THAT INTERRUPTED   CM160780
006ER  4230 006ER                          BNE   *                                                 CM160790
0072R  0888                                LDAR  STAT,STAT          STATUS SHOULD BE ZERO (BSY = 0)   CM160800
```

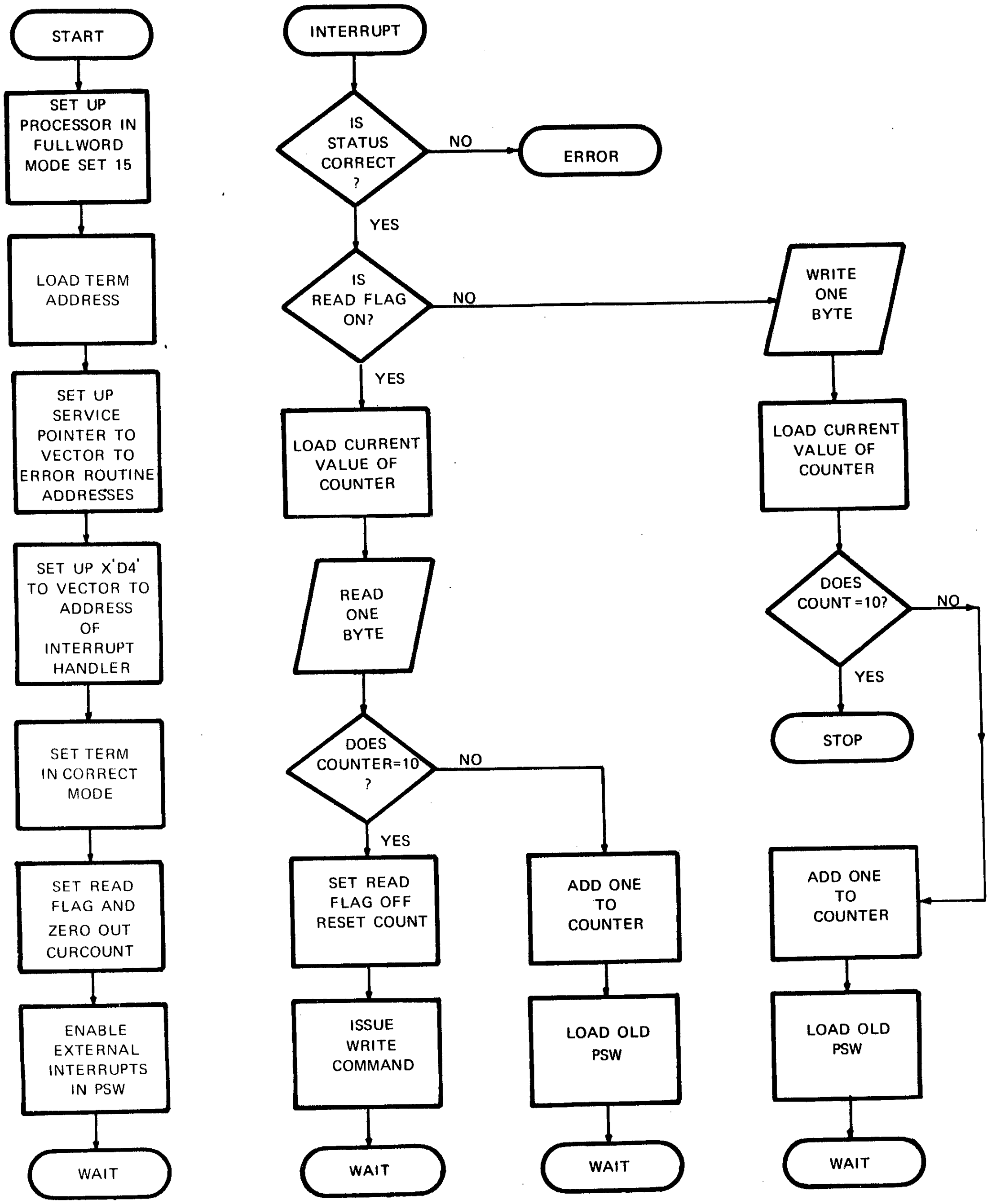
```
    COMM MUX PROGRAMMING EXAMPLES FOR 16- BIT PROCESSOR       PAGE   3  15:01:46  12/03/79

    LOCAL TERMINAL, HALF-DUPLEX  COMM MUX OPERATION
                                                                                                CM160810
 0074R  4230 0074R        81            BNZ    *                                                CM160820
 0078R  0866              82            LDAR   DONE,DONE                                        CM160830
 007AR  4230 0086R        83            BNZ    HALT               HALT IF DONE                  CM160840
                          84   *                                                                CM160850
 007ER  0A71 0000         85            WD     DEV,0(MSG)         WRITE A CHARACTER             CM160860
 0082R  2611              86            AIS    MSG,1                                            CM160870
 0084R  C510 00E6R        87            CLAI   MSG,MSG2END                                      CM160880
 0088R  4280 0060R        88            BL     EXMP1C             LOOP TILL 'CORRECT !' OUTPUT  CM160890
 008CR  2461              89            LIS    DONE,1             SET DONE FLAG                 CM160900
 008ER  4300 0060R        90            B      EXMP1C             ACKNOWLEDGE LAST INTERRUPT    CM160910
                          91   *                                                                CM160920
                          92   *                                                                CM160930
 0092R  9D78              93   READ     SSR    DEV,STAT           SENSE  COMM MUX STATUS        CM160940
 0094R  4240 00B6R        94            BTC    EX,HALT            HALT IF 'EX' IS SET           CM160950
 0098R  2083              95            BTBS   BSY,3              LOOP IN 'BSY'                 CM160960
 009AR  9B74              96            RDR    DEV,CHAR           READ CHARACTER WHEN BSY DROPS CM160970
 009CR  C440 007F         97            NAI    CHAR,X'7F'         REMOVE PARITY BIT             CM160980
 00A0R  030F              98            BR     LINK               RETURN                        CM160990
                          99   *                                                                CM161000
 00A2R  9D78             100   OUTCHR   SSR    DEV,STAT                                         CM161010
 00A4R  2081             101            BTBS   BSY,1              LOOP ON BSY                   CM161020
 00A6R  9A74             102            WDR    DEV,CHAR                                         CM161030
 00A8R  9D78             103            SSR    DEV,STAT                                         CM161040
 00AAR  2081             104            BTBS   BSY,1              WAIT FOR MUX TO BECOME NOT BUSY  CM161050
 00ACR  030F             105            BR     LINK               RETURN                        CM161060
                         106   *                                                                CM161070
 00AER  0755             107   DELAY    XAR    WORK,WORK                                        CM161080
 00B0R  2651             108            AIS    WORK,1                                           CM161090
 00B2R  2281             109            BFBS   CARRY,1                                          CM161100
 00B4R  030F             110            BR     LINK                                             CM161110
                         111   *                                                                CM161120
 00B6R  2451             112   HALT     LIS    WORK,1                                           CM161130
 00B8R  915F             113            SLHLS  WORK,15            WORK = X'8000'                CM161140
 00BAR  9515             114            EPSR   MSG,WORK           HALT PROCESSOR                CM161150
 00BCR  0309             115            BR     REPEAT                                           CM161160
                         116   * CONSTANTS & MESSAGES USED IN ABOVE EXAMPLES                    CM161170
                         117   *                                                                CM161180
 00BER  0010             118   DEVADR   DCX    10                 COMM MUX HDX ADDRESS          CM161190
 00C0R  F8               119   SECOND   DB     X'F8'                                            CM161200
                         120   *                                  8 DATA BITS/CHAR, 2 STOP BITS,  CM161210
                         121   *                                  NO PARITY CHECK.              CM161220
 00C1R  AB               122   DISWRT   DB     X'AB'              DISABLE INT, WRITE MODE       CM161230
 00C2R  B9               123   DISRD    DB     X'B9'              DISABLE INT, READ WITH ECHO-BACK  CM161240
 00C3R  6B               124   ENWRT    DB     X'6B'              ENABLE INT, WRITE MODE        CM161250
                         125   *                                                                CM161260
 00C4R  5459 5045 2031 3233  126   MSG1     DC     C'TYPE 1234567890',X'0D0A'
 00CCR  3435 3637 3839 3020
 00D4R  0D0A
 00D6R  0D0A             127            DC     X'0D0A'                                          CM161270
        0000 00D8R       128   MSG1END  EQU    *                                                CM161280
        0000 00CBR       129   TYPED    EQU    *-13                                             CM161290
 00D8R  0D0A             130   MSG2     DC     X'0D0A',C'CORRECT',X'2121',X'0D0A'               CM161300
 00DAR  434F 5252 4543 5420
```

```
      COMM MUX PROGRAMMING EXAMPLES FOR 16- BIT PROCESSOR       PAGE   4  15:01:46  12/03/79
      LOCAL TERMINAL, HALF-DUPLEX  COMM MUX OPERATION
 00E2R  2121
 00E4R  0D0A
        0000 00E6R            131  MSG2END  EQU   *                                              CM161310
                              132  *                                                             CM161320
 00E6R                        133           END                                                  CM161330
```

```
     COMM MUX PKOGRAMMING EXAMPLES FOR 16- BIT PROCESSOR       PAGE   5  15:01:46  12/03/79

     LOCAL TERMINAL, HALF-DUPLEX  COMM MUX OPERATION

ASSEMBLED BY CAL 03-066R07-00   (32-BIT)

START OPTIONS:  T=16,ERLST

   NO CAL ERRORS
   NO CAL WARNINGS
    2 PASSES

 ABSTOP     0000 0000
 ADC        0000 0002
 BSY        0000 0008         24*     95     101     104
 CARRY      0000 0008         25*    109
 CHAR       0000 0004         14*     46      57      72      96      97     102
 DELAY      0000 00AER        53      70     107*
 DEV        0000 0007         17*     41      42      43      51      52      69      78      85      93      96     100     102
                             103
 DEVADR     0000 00BER        41     118*
 DISRD      0000 00C2R        51     123*
 DISWRT     0000 00C1R        43     122*
 DONE       0000 0006         16*     38      38      39      66      82      82      89
 ENWRT      0000 00C3R        69     124*
 EX         0000 0004         23*     94
 EXMP1A     0000 0018R        46*     50
 EXMP1B     0000 0034R        56*     61
 EXMP1C     0000 0060R        71*     88      90
 HALT       0000 00B6R        83      94     112*
 IMPTOP     0000 00E6R
 INT        0000 006AR        65      77*
 LADC       0000 0001
 LINK       0000 000F         20*     47      53      56      70      98     105     110
 MSG        0000 0001         13*     44      46      48      49      54      54      57      59      60      68      85      86
                              87     114
 MSG1       0000 00C4R        44     126*
 MSG1END    0000 00D8R        49     128*
 MSG2       0000 00D8R        68     130*
 MSG2END    0000 00E6R        87     131*
 OUTCHR     0000 00A2R        47     100*
 PURETOP    0000 0000R
 READ       0000 0092R        56      93*
 REPEAT     0000 0009         19*     40     115
 SECOND     0000 00C0R        42     119*
 START1     0000 0000R        38*     40
 STAT       0000 0008         18*     77      80      80      93     100     103
 TYPED      0000 00CBR        57     129*
 WORK       0000 0005         15*     39      52      65      67      71      72      77      78     107     107     108     112
                             113     114
```

APPENDIX B
32-BIT PROGRAMMING EXAMPLES

MICRO I/O BUS UNDER INTERRUPTS


START
SET UP PROCESSOR IN FULLWORD MODE SET 15
LOAD TERM ADDRESS
SET UP SERVICE POINTER TO VECTOR TO ERROR ROUTINE ADDRESSES
SET UP X'D4' TO VECTOR TO ADDRESS OF INTERRUPT HANDLER
SET TERM IN CORRECT MODE
SET READ FLAG AND ZERO OUT CURCOUNT
ENABLE EXTERNAL INTERRUPTS IN PSW
WAIT
INTERRUPT
IS STATUS CORRECT ?
NO
ERROR
YES
IS READ FLAG ON?
NO
YES
LOAD CURRENT VALUE OF COUNTER
READ ONE BYTE
DOES COUNTER=10 ?
NO
YES
SET READ FLAG OFF RESET COUNT
ISSUE WRITE COMMAND
WAIT
ADD ONE TO COUNTER
LOAD OLD PSW
WAIT
WRITE ONE BYTE
LOAD CURRENT VALUE OF COUNTER
DOES COUNT =10?
NO
YES
STOP
ADD ONE TO COUNTER
LOAD OLD PSW
WAIT

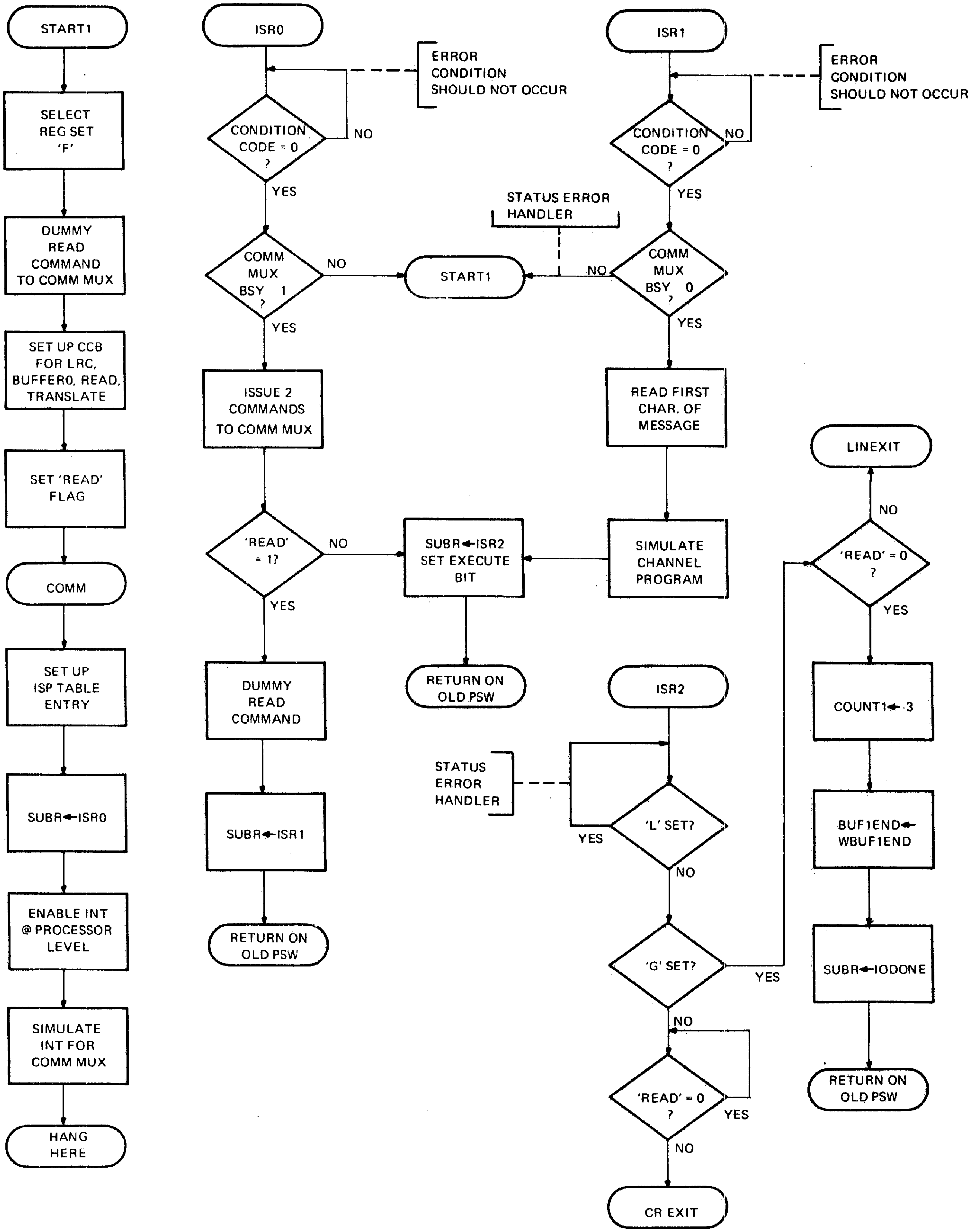
```
       SAMPLE PROGRAM USING MODEL 550 ON CURRENT LOOP INTERFACE  PAGE   1  14:18:58  12/03/79
PROG= CLI32I      ASSEMBLED BY CAL 03-066R07-00   (32-BIT)

                                    1  CLI32I     PROG  SAMPLE PROGRAM USING MODEL 550 ON CURRENT LOOP INTERFACE
                                    2                 CROSS
                                    3                 TARGT 32
                                    4                 WIDTH 120
                                    5                 NORX3
                                    6  *
                                    7  *   SAMPLE PROGRAM FOR THE MODEL-550 ON A 32-BIT PROCESSOR
                                    8  *   THIS PROGRAM DEMONSTRATES THE READ/WRITE OPERATIONS FOR THE
                                    9  *   MODEL-550 WITH THE CURRENT LOOP INTERFACE UNDER INTERRUPT CONTROL
                                   10  *
                                   11  *   THE PROGRAM READS 10 CHARACTERS THEN OUTPUTS A CARRIAGE RETURN
                                   12  *   AND LINE FEED, ECHOS BACK THE TEN CHARACTERS, AND OUTPUTS
                                   13  *   ANOTHER CARRIAGE RETURN AND LINE FEED.
                                   14  *
                                   15  *   SET UP REGISTERS
                                   16  *
          0000 0000                17  R0         EQU   0                     WORK REGISTER
          0000 0001                18  R1         EQU   1                     WORK REGISTER
          0000 0002                19  R2         EQU   2                     WORK REGISTER
          0000 0003                20  R3         EQU   3                     CONDITION CODE TEST REGISTER
          0000 0004                21  DEVADR     EQU   4                     DEVICE ADDRESS REGISTER
          0000 0005                22  WORK       EQU   5                     WORK REGISTER
          0000 0006                23  COUNT      EQU   6                     COUNT REGISTER
          0000 0007                24  R7         EQU   7                     INTERRUPT ADDRESS REGISTER
                                   25  *
                                   26  *   SET UP PROCESSOR IN FULL WORD MODE & REGISTER SET 15
                                   27  *
000000I   C800 00F0                28  START      LHI   R0,X'00F0'
000004I   9510                     29             EPSR  R1,R0                 REGISTER SET 15 ACTIVATED
000006I   4840 80AE =0000B8I       30             LH    DEVADR,MODADR         LOAD CLI DEVICE ADDRESS
                                   31  *
                                   32  *   SET UP INTERRUPT TABLE
                                   33  *
00000AI   0722                     34             XAR   R2,R2                 ZERO OUT REGISTER 2
00000CI   C810 0098I               35             LHI   R1,ERROR              SET UP ADDRESS OF ERROR
000010I   4012 00D0                36  LOAGIN     STH   R1,X'D0'(R2)          STORE ERROR IN FULL TABLE
000014I   2622                     37             AIS   R2,2                  ADVANCE TO THE NEXT ENTRY
000016I   C520 0200                38             CLHI  R2,X'200'             IS TABLE FULL YET
00001AI   4280 FFF2 =000010I       39             BL    LOAGIN                BRANCH TO LOAD ANOTHER ERROR ADDRESS
00001EI   4850 8096 =0000B8I       40             LH    WORK,MODADR           LOAD DEVICE ADDRESS INTO WORK REG
000022I   1151                     41             SLLS  WORK,1
000024I   C870 0046I               42             LHI   R7,INT                LOAD INTERRUPT ADDRESS INTO REG 7
000028I   4075 00D0                43             STH   R7,X'D0'(WORK)        STORE ADDRESS IN SERVICE POINTER TBL
                                   44  *
                                   45  *   ENABLE THE MODEL-550 INTERRUPT IN READ MODE
                                   46  *
00002CI   DE40 808A =0000BAI       47             OC    DEVADR,RDCMD          SET STATUS OF CLI
000030I   2451                     48             LIS   WORK,1                LOAD ONE INTO WORK REGISTER
000032I   4050 8086 =0000BCI       49             STH   WORK,FLAG             SET READ FLAG
000036I   0766                     50             XAR   COUNT,COUNT           ZERO OUT COUNT
000038I   4060 806C =0000A8I       51             STH   COUNT,CURCOUNT        STORE ZERO IN CURCOUNT
                                   52  *
                                   53  *   ENABLE EXTERNAL INTERRUPTS IN PSW
```

```
     SAMPLE PROGRAM USING MODEL 550 ON CURRENT LOOP INTERFACE  PAGE   2  14:18:58  12/03/79

                                          54  *
00003CI  F810 0000 C0F0                   55  GOAGAIN  LI    R1,Y'C0F0'          WAIT STATE, ENABLE INTRPT, REG SET F
000042I  9501                             56           EPSR  R0,R1               INTERRUPTS ENABLED
000044I  2204                             57           BS    GOAGAIN             HANG, SHOULD NEVER EXEC THIS INSTRUCT  ION
                                          58  *
                                          59  *   INTERRUPT HANDLER
                                          60  *
000046I  0833                             61  INT      LDAR  R3,R3               SET CONDITION CODE
000048I  4230 804C =000098I               62           BNZ   ERROR               IF STATUS IS BAD GO TO ERROR
00004CI  4850 806C =0000BCI               63           LH    WORK,FLAG           TEST FLAG
000050I  4230 8018 =00006CI               64           BNZ   READBR              IF 1 GO TO READ 10 BYTES
                                          65  *
                                          66  *   WRITE 10 BYTES OF DATA TO MODEL-550
                                          67  *
0000541  4860 8050 =0000A8I               68           LH    COUNT,CURCOUNT      LOAD VALUE OF CURCOUNT INTO COUNT
000058I  DA46 804E =0000AAI               69  WRTBR    WD    DEVADR,WRTBUF(COUNT) WRITE OUT A BYTE OF DATA
00005CI  2661                             70           AIS   COUNT,1             ADD ONE TO COUNT
00005EI  4060 8046 =0000A8I               71           STH   COUNT,CURCOUNT      UPDATE CURCOUNT
000062I  C560 000E                        72           CLHI  COUNT,14            COMPARE COUNT TO 14
000066I  4330 8032 =00009CI               73           BE    DONE                IF FINISHED GO TO DONE
00006AI  1800                             74           LPSWR R0
                                          75  *
                                          76  *   READ 10 BYTES OF DATA FROM MODEL-550 TO MODBUF
                                          77  *
00006CI  4840 8048 =0000B8I               78  READBR   LH    DEVADR,MODADR       LOAD CLI ADDRESS
000070I  C860 00A8I                       79           LHI   COUNT,CURCOUNT      INITIALIZE COUNT REGISTER
000074I  DB46 8034 =0000ACI               80           RD    DEVADR,MODBUF(COUNT) READ ONE BYTE
000078I  2661                             81           AIS   COUNT,1             ADD ONE TO COUNT
00007AI  4060 802A =0000A8I               82           STH   COUNT,CURCOUNT      UPDATE CURCOUNT
00007EI  C560 000A                        83           CLHI  COUNT,10            COMPARE TO 10
000082I  4330 8002 =000088I               84           BE    GOWRITE             BRANCH TO ROUTINE THAT SETS UP WRITE
000086I  1800                             85           LPSWR R0                  LOAD OLD PSW
                                          86  *
000088I  2450                             87  GOWRITE  LIS   WORK,0              LOAD ZERO INTO WORK REGISTER
00008AI  4050 802E =0000BCI               88           STH   WORK,FLAG           STORE ZERO IN FLAG TO WRITE
00008EI  4050 8016 =0000A8I               89           STH   WORK,CURCOUNT
000092I  DE40 8025 =0000BBI               90           OC    DEVADR,WRTCMD       SET WRITE MODE
000096I  1800                             91           LPSWR R0                  LOAD OLD PSW
                                          92  *
                                          93  *   SET UP DATE CONSTANTS AND STORAGE AREAS
                                          94  *
000098I  4300 FF64 =000000I               95  ERROR    B     START               BRANCH TO START OF PROGRAM
                                          96  *
00009CI  C200 8000 =0000A0I               97  DONE     LPSW  WAIT
                                          98  *
0000A0I  0000 8000                        99  WAIT     DCF   Y'8000'
0000A4I  0000 009CI                      100           DC    A(DONE)
0000A8I  0000                            101  CURCOUNT DC    X'0'
0000AAI  0D0A                            102  WRTBUF   DC    X'0D0A'             0D=CARRIAGE RETURN, 0A=LINE FEED
0000ACI                                  103  MODBUF   DS    10                  RESERVE STORAGE FOR 10 BYTES
0000B6I  0D0A                            104           DC    X'0D0A'             0D=CARRIAGE RETURN,0A=LINE FEED
0000B8I  0002                            105  MODADR   DC    X'0002'             MODADR = 02
0000BAI  6458                            106  RDCMD    DC    X'6458'             ENABEL UNBLOCK READ/ENABLE WRITE
         0000 00BBI                      107  WRTCMD   EQU   RDCMD+1
0000BCI  0000 0000                       108  FLAG     DC    Y'0'
```

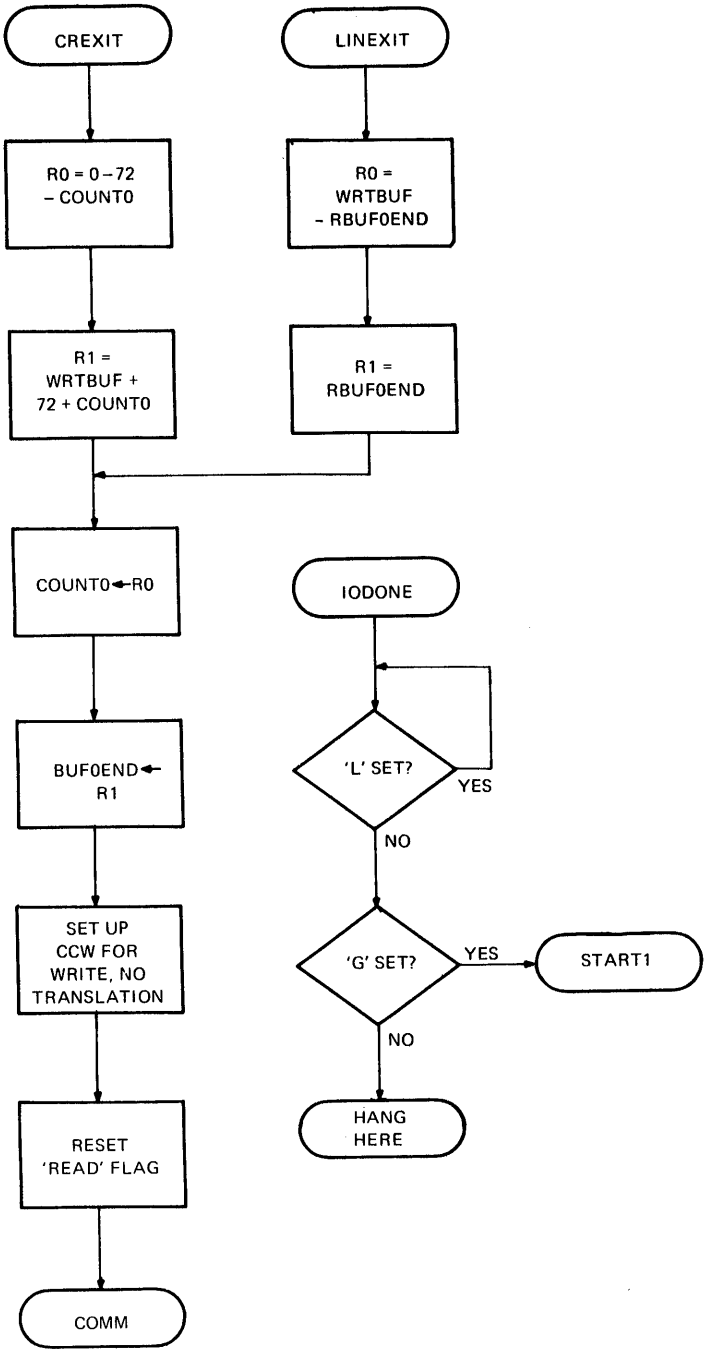
```
     SAMPLE PROGRAM USING MODEL 550 ON CURRENT LOOP INTERFACE  PAGE   3  14:18:58  12/03/79
0000C01                        109            END
```

```
    SAMPLE PROGRAM USING MODEL 550 ON CURRENT LOOP INTERFACE  PAGE   4  14:18:58  12/03/79

ASSEMBLED BY CAL 03-066R07-00   (32-BIT)

START OPTIONS:  T=32,ERLST

   NO CAL ERRORS
   NO CAL WARNINGS
    2 PASSES

 ABSTOP     0000 0000
 ADC        0000 0004
 COUNT      0000 0006         23*     50      50      51      68      69      70      71      72      79      80      81      82
                              83
 CURCOUNT   0000 00A8I        51      68      71      79      82      89      101*
 DEVADR     0000 0004         21*     30      47      69      78      80      90
 DONE       0000 009CI        73      97*     100
 ERROR      0000 0098I        35      62      95*
 FLAG       0000 00BCI        49      63      88      108*
 GOAGAIN    0000 003CI        55*     57
 GOWRITE    0000 0088I        84      87*
 IMPTOP     0000 00C0I        109
 INT        0000 0046I        42      61*
 LADC       0000 0002
 LOAGIN     0000 0010I        36*     39
 MODADR     0000 00B8I        30      40      78      105*
 MODBUF     0000 00ACI        80      103*
 PURETOP    0000 0000P        109
 R0         0000 0000         17*     28      29      56      74      85      91
 R1         0000 0001         18*     29      35      36      55      56
 R2         0000 0002         19*     34      34      36      37      38
 R3         0000 0003         20*     61      61
 R7         0000 0007         24*     42      43
 RDCMD      0000 00BAI        47      106*    107
 READBR     0000 006CI        64      78*
 START      0000 0000I        28*     95
 WAIT       0000 00A0I        97      99*
 WORK       0000 0005         22*     40      41      43      48      49      63      87      88      89
 WRTBR      0000 0058I        69*
 WRTBUF     0000 00AAI        69      102*
 WRTCMD     0000 00BBI        90      107*
```

COMM MUX


START1
SELECT REG SET 'F'
DUMMY READ COMMAND TO COMM MUX
SET UP CCB FOR LRC, BUFFER0, READ, TRANSLATE
SET 'READ' FLAG
COMM
SET UP ISP TABLE ENTRY
SUBR←ISR0
ENABLE INT @ PROCESSOR LEVEL
SIMULATE INT FOR COMM MUX
HANG HERE
ISR0
ERROR CONDITION SHOULD NOT OCCUR
CONDITION CODE = 0 ?
NO
YES
COMM MUX BSY 1 ?
NO
YES
START1
ISSUE 2 COMMANDS TO COMM MUX
'READ' = 1?
NO
YES
DUMMY READ COMMAND
SUBR←ISR1
RETURN ON OLD PSW
SUBR←ISR2 SET EXECUTE BIT
RETURN ON OLD PSW
STATUS ERROR HANDLER
ISR1
ERROR CONDITION SHOULD NOT OCCUR
CONDITION CODE = 0 ?
NO
YES
COMM MUX BSY 0 ?
NO
YES
READ FIRST CHAR. OF MESSAGE
SIMULATE CHANNEL PROGRAM
ISR2
STATUS ERROR HANDLER
'L' SET?
YES
NO
'G' SET?
YES
NO
'READ' = 0 ?
YES
NO
CR EXIT
LINEXIT
NO
'READ' = 0 ?
YES
COUNT1←-3
BUF1END← WBUF1END
SUBR←IODONE
RETURN ON OLD PSW

COMM MUX (Continued)



```
        COMM MUX PROGRAMMING EXAMPLES FOR 32-BIT PROCESSOR        PAGE   1  11:10:15  08/16/78
    PROG= CM32A      ASSEMBLED BY CAL 03-066R05-00   (32-BIT)

                             1  CM32A    PROG  COMM MUX PROGRAMMING EXAMPLES FOR 32-BIT PROCESSOR       CM320010
                             2                 CROSS                                                    CM320020
                             3                 WIDTH 120                                                CM320030
                             4                 TARGT 32                                                 CM320040
                             5                 NORX3                                                    CM320050
                             6  *                                                                       CM320060
                             7  *                                                                       CM320070
                             8  *  USER INPUTS A MESSAGE OF UP TO 72 CHARACTERS THROUGH THE TERMINAL    CM320080
                             9  * HOOKED TO THE COMMUNICATIONS MUX INTERFACE. A MESSAGE OF LESS THAN    CM320090
                            10  *  72 CHARACTERS MUST BE TERMINATED BY DEPRESSING 'CR' KEY TWICE. THE   CM320100
                            11  *  AUTO-DRIVER CHANNEL READS THE CHARACTERS & DOES ASCII TO ASCII       CM320110
                            12  *  TRANSLATION, AND GENERATES A BUFFER CALLED 'MESSAGE'.               CM320120
                            13  *  THEN CCB IS SET UP TO WRITE, FAST MODE, NO TRANSLATION. THE          CM320130
                            14  *  'CR', 'LF' OR JUST 'LF' IS OUTPUT FOLLOWED BY THE 'MESSAGE'          CM320140
                            15  *  BUFFER. THEN 'CR', 'LF' ARE OUTPUT.                                  CM320150
                            16  *                                                                       CM320160
                            17  *  THE ENTIRE PROGRAM LOOPS ON ITSELF.                                  CM320170
                            18  *                                                                       CM320180
                            19  *                                                                       CM320190
                            20  * REGISTER EQUATES                                                      CM320200
                            21  *                                                                       CM320210
                            22  *                                                                       CM320220
          0000 0000         23  R0       EQU   0                                                        CM320230
          0000 0001         24  R1       EQU   1                                                        CM320240
          0000 0007         25  DEV      EQU   7                                                        CM320250
          0000 000D         26  WORK     EQU   13                                                       CM320260
```

```
COMM MUX PROGRAMMING EXAMPLES FOR 32-BIT PROCESSOR       PAGE   2  11:10:15  08/16/78

DATA TRANS THRU AUTO DRIVER CHANNEL 4 WIRE OPERATION
                                                                                CM320280
000000I                           ORG    X'A00'                                 CM320290
                           *                                                    CM320300
                           *                                                    CM320310
000A00   C810 00F0        START1  LHI    R1,X'F0'                               CM320320
000A04   9501                     EPSR   R0,R1          REGISTER SET F          CM320330
                           *                                                    CM320340
000A06   4870 0BB2                LH     DEV,RCVADR     GET RECEIVE SIDE ADDRESS  CM320350
000A0A   9B7D                     RDR    DEV,WORK       DUMMY READ TO SET BUSY  CM320360
000A0C   4070 0BB0                STH    DEV,DEVADR                             CM320370
                           *                                                    CM320380
000A10   C810 F702                LHI    R1,CCWSTA+TLATE                        CM320390
000A14   4010 0B48                STH    R1,CCW         LRC, BUFFER 0, READ, TRANSLATE  CM320400
000A18   E610 0BA7                LDAI   R1,RBUFOEND                            CM320410
000A1C   E600 0B60                LDAI   R0,MESSAGE                             CM320420
000A20   0B01                     SAR    R0,R1                                  CM320430
000A22   4000 0B4A                STH    R0,COUNTO                              CM320440
000A26   5010 0B4C                STA    R1,BUFOEND                             CM320450
000A2A   D300 0DBE                LB     R0,ENREAD      GET READ COMMAND        CM320460
000A2E   D200 0BB4                STB    R0,CMD                                 CM320470
000A32   24D1                     LIS    WORK,1                                 CM320480
000A34   40D0 0BAC                STH    WORK,READ      SET FLAG                CM320490
                           *                                                    CM320500
000A38   C810 0B48        COMM    LHI    R1,CCB                                 CM320510
000A3C   2611                     AIS    R1,1                                   CM320520
000A3E   4017 4700 00D0           STH    R1,ISPTAB(DEV,DEV)  SET UP ISP TABLE ENTRY  CM320530
000A44   2418                     LIS    R1,8                                   CM320540
000A46   7610 0B48                RBT    R1,CCW         RESET EX BIT            CM320550
000A4A   C810 0AAC                LHI    R1,ISR0                                CM320560
000A4E   4010 0B5C                STH    R1,SUBR        SUBROUTINE ADDRESS      CM320570
                           *                                                    CM320580
000A52   C810 40F0                LHI    R1,X'40F0'                             CM320590
000A56   9501                     EPSR   R0,R1          ENABLE INT @ PROCESSOR LEVEL  CM320600
000A58   4870 0BB0                LH     DEV,DEVADR                             CM320610
000A5C   E207 0000                SINT   0(DEV)                                 CM320620
000A60   4300 0A60                B      *              HANG                    CM320630
                           *                                                    CM320640
                           *  TO WRITE CR, LF OR JUST LF FOLLOWED BY MESSAGE & CR, LF.  CM320650
                           *                                                    CM320660
000A64   0700             CREXIT  XAR    R0,R0                                  CM320670
000A66   4810 0B4A                LH     R1,COUNTO                              CM320680
000A6A   CA10 0048                AHI    R1,72                                  CM320690
000A6E   0B01                     SAR    R0,R1                                  CM320700
000A70   FA10 0000 0B5E           AAI    R1,WRTBUF                              CM320710
000A76   2306                     BS     OK                                     CM320720
000A78   E610 0BA7        LINEXIT LDAI   R1,RBUFOEND                            CM320730
000A7C   E600 0B5E                LDAI   R0,WRTBUF                              CM320740
000A80   0B01                     SAR    R0,R1                                  CM320750
000A82   4000 0B4A        OK      STH    R0,COUNTO                              CM320760
000A86   5010 0B4C                STA    R1,BUFOEND                             CM320770
000A8A   C810 F704                LHI    R1,CCWSTA+WRITE  WRITE, NO TRANSLATION  CM320780
000A8E   4010 0B48                STH    R1,CCW                                 CM320790
000A92   D300 0DC2                LB     R0,ENWRT                               CM320800
```

```
         COMM MUX PROGRAMMING EXAMPLES FOR 32-BIT PROCESSOR        PAGE   3  11:10:15  08/16/78

         DATA TRANS THRU AUTO DRIVER CHANNEL 4 WIRE OPERATION

000A96   D2C0 0BB4           81          STB   R0,CMD                                                        CM320810
000A9A   4870 0DBC           82          LH    DEV,SNDADR                                                    CM320820
000A9E   4070 0BB0           83          STH   DEV,DEVADR                                                    CM320830
000AA2   07DD                84          XAR   WORK,WORK                                                     CM320840
000AA4   40D0 0BAC           85          STH   WORK,READ          RESET FLAG                                 CM320850
000AA8   43C0 0A38           86          B     COMM                                                          CM320860
                             87    *                                                                         CM320870
                             88    *  COME HERE AFTER SINT                                                   CM320880
                             89    *                                                                         CM320890
000AAC   42F0 0AAC           90    ISR0  BTC   C+V+G+L,*                                                     CM320900
000AB0   C330 0008           91          THI   3,8                                                           CM320910
000AB4   4330 0A00           92          BZ    START1                                                        CM320920
000AB8   DE20 0DC3           93          OC    2,SECOND           SET UP FOR  COMM MUX                       CM320930
000ABC   DE20 0BB4           94          OC    2,CMD              ISSUE COMMAND                              CM320940
000AC0   48D0 0BAC           95          LH    WORK,READ                                                     CM320950
000AC4   4330 0AD4           96          BZ    ISR01                                                         CM320960
000AC8   9B2D                97          RDR   2,WORK             SET BSY                                    CM320970
000ACA   E6D0 0AE4           98          LDAI  WORK,ISR1                                                     CM320980
000ACE   40D0 0B5C           99          STH   WORK,SUBR          TO READ A MESSAGE                          CM320990
000AD2   1800               100          LPSWR R0                                                            CM321000
                            101    *                                                                         CM321010
000AD4   E6D0 0B04          102    ISR01 LDAI  WORK,ISR2                                                     CM321020
000AD8   40D0 0B5C          103          STH   WORK,SUBR          TO WRITE THE MESSAGE                       CM321030
000ADC   24D8               104          LIS   WORK,8                                                        CM321040
000ADE   75D0 0B48          105          SBT   WORK,CCW           SET EX BIT                                 CM321050
000AE2   1800               106          LPSWR R0                                                            CM321060
                            107    *                                                                         CM321070
                            108    * COME HERE TO READ FIRST CHARACTER                                       CM321080
                            109    *                                                                         CM321090
000AE4   42F0 0AE4          110    ISR1  BTC   C+V+G+L,*                                                     CM321100
000AE8   C330 0008          111          THI   3,8                                                           CM321110
000AEC   4230 0A00          112          BNZ   START1                                                        CM321120
000AF0   9B2D               113          RDR   2,WORK             READ FIRST CHARACTER OF MESSAGE            CM321130
000AF2   E3D0 0B48          114          SCP   WORK,CCB           PUT IT IN BUFFER                           CM321140
000AF6   4300 0AD4          115          B     ISR01                                                         CM321150
                            116    *                                                                         CM321160
                            117    *                                                                         CM321170
000AFA   1800               118    IGNORE   LPSWR R0                                                         CM321180
                            119    *                                                                         CM321190
                            120    *                                                                         CM321200
000AFC   07DD               121    CR    XAR   WORK,WORK                                                     CM321210
000AFE   40D0 0B48          122          STH   WORK,CCW           RESET EX BIT                               CM321220
000B02   1800               123          LPSWR R0                                                            CM321230
                            124    *                                                                         CM321240
                            125    *                                                                         CM321250
                            126    *  COME HERE AFTER ADC TERMINATION                                        CM321260
                            127    *                                                                         CM321270
000B04   4210 0B04          128    ISR2  BTC   L,*                                                           CM321280
000B08   4220 0B18          129          BTC   G,BUFULL                                                      CM321290
000B0C   48D0 0BAC          130          LH    WORK,READ                                                     CM321300
000B10   4330 0B10          131          BZ    *                                                             CM321310
000B14   4300 0A64          132          B     CREXIT                                                        CM321320
000B18   48D0 0BAC          133    BUFULL LH   WORK,READ                                                     CM321330
```

```
      COMM MUX PROGRAMMING EXAMPLES FOR 32-BIT PROCESSOR        PAGE   4  11:10:15  08/16/78

      DATA TRANS THRU AUTO DRIVER CHANNEL 4 WIRE OPERATION
                                                                                           CM321340
000B1C   4230 0A78          134            BNZ   LINEXIT                                   CM321340
000B20   C8D0 FFFD          135            LHI   WORK,-3           TO OUTPUT CR, LF, NULL, NULL.   CM321350
000B24   40D0 0B52          136            STH   WORK,COUNT1                               CM321360
000B28   E6D0 0BAB          137            LDAI  WORK,WBUF1END                             CM321370
000B2C   50D0 0B54          138            STA   WORK,BUF1END                              CM321380
000B30   C8D0 0B3A          139            LHI   WORK,IODONE                               CM321390
000B34   40D0 0B5C          140            STH   WORK,SUBR                                 CM321400
000B38   1800               141            LPSWR R0                                        CM321410
                            142  *                                                         CM321420
                            143  *  COME HERE WHEN ALL I/O OPERATION IS OVER               CM321430
                            144  *                                                         CM321440
                            145  *                                                         CM321450
000B3A   4210 0B3A          146  IODONE    BTC   L,*                                       CM321460
000B3E   4220 0A00          147            BTC   G,START1                                  CM321470
000B42   4300 0B42          148            B     *                                         CM321480
                            149  *                                                         CM321490
000B48                      150            ALIGN ADC                                       CM321500
                            151  *-----------------------------------                      CM321510
                            152  *                                                         CM321520
         0000 0B48          153  CCB       EQU   *                                         CM321530
000B48   0000               154  CCW       DCX   0                                         CM321540
000B4A   0000               155  COUNTO    DCX   0                                         CM321550
000B4C   0000 0000          156  BUFOEND   DC    0                                         CM321560
000B50   0000               157  CHKWORD   DCX   0                                         CM321570
000B52   0000               158  COUNT1    DCX   0                                         CM321580
000B54   0000 0000          159  BUF1END   DC    0                                         CM321590
000B58   0000 0BB8          160            DC    A(TLATETAB)                               CM321600
000B5C   0000               161  SUBR      DCX   0                                         CM321610
                            162  *                                                         CM321620
                            163  *-----------------------------------                      CM321630
                            164  *  BUFFERS                                                CM321640
                            165  *                                                         CM321650
000B5E   0D                 166  WRTBUF    DB    13                                        CM321660
000B5F   0A                 167            DB    10                                        CM321670
000B60                      168  MESSAGE   DS    72                                        CM321680
         0000 0BA7          169  RBUFOEND  EQU   *-1                                       CM321690
000BA8   0DCA               170            DCX   0D0A,0                                    CM321700
000BAA   0000
         0000 0BAB          171  WBUF1END  EQU   *-1                                       CM321710
                            172  *-----------------------------------                      CM321720
                            173  *                                                         CM321730
                            174  *  EQUATES                                                CM321740
                            175  *                                                         CM321750
         0000 F700          176  CCWSTA    EQU   X'F700'                                   CM321760
         0000 0004          177  WRITE     EQU   X'0004'                                   CM321770
         0000 0002          178  TLATE     EQU   X'0002'                                   CM321780
                            179  *                                                         CM321790
         0000 00D0          180  ISPTAB    EQU   X'D0'                                     CM321800
         0000 0008          181  C         EQU   8                                         CM321810
         0000 0004          182  V         EQU   4                                         CM321820
         0000 0002          183  G         EQU   2                                         CM321830
         0000 0001          184  L         EQU   1                                         CM321840
                            185  *                                                         CM321850
```

```
COMM MUX PROGRAMMING EXAMPLES FOR 32-BIT PROCESSOR        PAGE  5  11:10:15  08/16/78

DATA TRANS THRU AUTO DRIVER CHANNEL 4 WIRE OPERATION

                                186  *                                                          CM321860
                                187  *  CONSTANTS                                               CM321870
                                188  *                                                          CM321880
                                189  *                                                          CM321890
000BAC   0000 0000              190  READ      DC    0                                          CM321900
000BB0   0000                   191  DEVADR    DCX   0                                          CM321910
000BB2   0010                   192  RCVADR    DCX   10                                         CM321920
000BB4   00                     193  CMD       DB    0                                          CM321930
                                194  *                                                          CM321940
                                195  *---------------------------------                         CM321950
                                196  *                                                          CM321960
000BB8                          197            ALIGN ADC+ADC                                    CM321970
                                198  *                                                          CM321980
000BB8                          199  TLATETAB DO    13                                          CM321990
000BB8   057D                   200            DC    T(IGNORE)                                  CM322000
000BBA   057D                   200            DC    T(IGNORE)
000BBC   057D                   200            DC    T(IGNORE)
000BBE   057D                   200            DC    T(IGNORE)
000BC0   057D                   200            DC    T(IGNORE)
000BC2   057D                   200            DC    T(IGNORE)
000BC4   057D                   200            DC    T(IGNORE)
000BC6   057D                   200            DC    T(IGNORE)
000BC8   057D                   200            DC    T(IGNORE)
000BCA   057D                   200            DC    T(IGNORE)
000BCC   057D                   200            DC    T(IGNORE)
000BCE   057D                   200            DC    T(IGNORE)
000BD0   057D                   200            DC    T(IGNORE)
000BD2   057E                   201            DC    T(CR)                                      CM322010
000BD4                          202            DO    9                                          CM322020
000BD4   057D                   203            DC    T(IGNORE),T(IGNORE)                        CM322030
000BD6   057D
000BD8   057D                   203            DC    T(IGNORE),T(IGNORE)
000BDA   057D
000BDC   057D                   203            DC    T(IGNORE),T(IGNORE)
000BDE   057D
000BE0   057D                   203            DC    T(IGNORE),T(IGNORE)
000BE2   057D
000BE4   057D                   203            DC    T(IGNORE),T(IGNORE)
000BE6   057D
000BE8   057D                   203            DC    T(IGNORE),T(IGNORE)
000BEA   057D
000BEC   057D                   203            DC    T(IGNORE),T(IGNORE)
000BEE   057D
000BF0   057D                   203            DC    T(IGNORE),T(IGNORE)
000BF2   057D
000BF4   057D                   203            DC    T(IGNORE),T(IGNORE)
000BF6   057D
         0000 0BF8              204  TABLE     EQU   *                                          CM322040
000BF8                          205            DO    32                                         CM322050
000BF8   8020 8021              206            DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2      CM322060
000BFC   8022 8023              206            DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C00   8024 8025              206            DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C04   8026 8027              206            DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
```

```
         COMM MUX PROGRAMMING EXAMPLES FOR 32-BIT PROCESSOR        PAGE   6  11:10:15  08/16/78

         DATA TRANS THRU AUTO DRIVER CHANNEL 4 WIRE OPERATION
000C08   8028 8029          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C0C   802A 802B          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C10   802C 802D          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C14   802E 802F          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C18   8030 8031          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C1C   8032 8033          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C20   8034 8035          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C24   8036 8037          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C28   8038 8039          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C2C   803A 803B          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C30   803C 803D          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C34   803E 803F          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C38   8040 8041          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C3C   8042 8043          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C40   8044 8045          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C44   8046 8047          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C48   8048 8049          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C4C   804A 804B          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C50   804C 804D          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C54   804E 804F          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C58   8050 8051          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C5C   8052 8053          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C60   8054 8055          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C64   8056 8057          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C68   8058 8059          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C6C   805A 805B          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C70   805C 805D          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C74   805E 805F          206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C78                      207        DO    16                                               CM322070
000C78   057D               208        DC    T(IGNORE),T(IGNORE)                              CM322080
000C7A   057D
000C7C   057D               208        DC    T(IGNORE),T(IGNORE)
000C7E   057D
000C80   057D               208        DC    T(IGNORE),T(IGNORE)
000C82   057D
000C84   057D               208        DC    T(IGNORE),T(IGNORE)
000C86   057D
000C88   057D               208        DC    T(IGNORE),T(IGNORE)
000C8A   057D
000C8C   057D               208        DC    T(IGNORE),T(IGNORE)
000C8E   057D
000C90   057D               208        DC    T(IGNORE),T(IGNORE)
000C92   057D
000C94   057D               208        DC    T(IGNORE),T(IGNORE)
000C96   057D
000C98   057D               208        DC    T(IGNORE),T(IGNORE)
000C9A   057D
000C9C   057D               208        DC    T(IGNORE),T(IGNORE)
000C9E   057D
000CA0   057D               208        DC    T(IGNORE),T(IGNORE)
000CA2   057D
000CA4   057D               208        DC    T(IGNORE),T(IGNORE)
000CA6   057D
```

```
000CA8   057D                208           DC      T(IGNORE),T(IGNORE)
000CAA   057D
000CAC   057D                208           DC      T(IGNORE),T(IGNORE)
000CAE   057D
000CB0   057D                208           DC      T(IGNORE),T(IGNORE)
000CB2   057D
000CB4   057D                208           DC      T(IGNORE),T(IGNORE)
000CB6   057D
                             209  *                                                          CM322090
                             210  *                                                          CM322100
000CB8                       211           DO      13                                        CM322110
000CB8   057D                212           DC      T(IGNORE)                                 CM322120
000CBA   057D                212           DC      T(IGNORE)
000CBC   057D                212           DC      T(IGNORE)
000CBE   057D                212           DC      T(IGNORE)
000CC0   057D                212           DC      T(IGNORE)
000CC2   057D                212           DC      T(IGNORE)
000CC4   057D                212           DC      T(IGNORE)
000CC6   057D                212           DC      T(IGNORE)
000CC8   057D                212           DC      T(IGNORE)
000CCA   057D                212           DC      T(IGNORE)
000CCC   057D                212           DC      T(IGNORE)
000CCE   057D                212           DC      T(IGNORE)
000CD0   057D                212           DC      T(IGNORE)
000CD2   057E                213           DC      T(CR)                                     CM322130
000CD4                       214           DO      9                                         CM322140
000CD4   057D                215           DC      T(IGNORE),T(IGNORE)                       CM322150
000CD6   057D
000CD8   057D                215           DC      T(IGNORE),T(IGNORE)
000CDA   057D
000CDC   057D                215           DC      T(IGNORE),T(IGNORE)
000CDE   057D
000CE0   057D                215           DC      T(IGNORE),T(IGNORE)
000CE2   057D
000CE4   057D                215           DC      T(IGNORE),T(IGNORE)
000CE6   057D
000CE8   057D                215           DC      T(IGNORE),T(IGNORE)
000CEA   057D
000CEC   057D                215           DC      T(IGNORE),T(IGNORE)
000CEE   057D
000CF0   057D                215           DC      T(IGNORE),T(IGNORE)
000CF2   057D
000CF4   057D                215           DC      T(IGNORE),T(IGNORE)
000CF6   057D
         0000 0CF8           216  TABLEH   EQU     *                                         CM322160
000CF8                       217           DO      32                                        CM322170
000CF8   8020 8021           218           DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2   CM322180
000CFC   8022 8023           218           DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D00   8024 8025           218           DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D04   8026 8027           218           DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D08   8028 8029           218           DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D0C   802A 802B           218           DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D10   802C 802D           218           DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
```

```
        COMM MUX PROGRAMMING EXAMPLES FOR 32-BIT PROCESSOR        PAGE   8  11:10:15  08/16/78

        DATA TRANS THRU AUTO DRIVER CHANNEL 4 WIRE OPERATION
000D14    802E 802F            218          DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D18    8030 8031            218          DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D1C    8032 8033            218          DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D20    8034 8035            218          DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D24    8036 8037            218          DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D28    8038 8039            218          DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D2C    803A 803B            218          DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D30    803C 803D            218          DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D34    803E 803F            218          DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D38    8040 8041            218          DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D3C    8042 8043            218          DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D40    8044 8045            218          DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D44    8046 8047            218          DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D48    8048 8049            218          DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D4C    804A 804B            218          DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D50    804C 804D            218          DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D54    804E 804F            218          DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D58    8050 8051            218          DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D5C    8052 8053            218          DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D60    8054 8055            218          DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D64    8056 8057            218          DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D68    8058 8059            218          DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D6C    805A 805B            218          DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D70    805C 805D            218          DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D74    805E 805F            218          DB      X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D78                         219          DO      16                                                CM322190
000D78    057D                 220          DC      T(IGNORE),T(IGNORE)                               CM322200
000D7A    057D
000D7C    057D                 220          DC      T(IGNORE),T(IGNORE)
000D7E    057D
000D80    057D                 220          DC      T(IGNORE),T(IGNORE)
000D82    057D
000D84    057D                 220          DC      T(IGNORE),T(IGNORE)
000D86    057D
000D88    057D                 220          DC      T(IGNORE),T(IGNORE)
000D8A    057D
000D8C    057D                 220          DC      T(IGNORE),T(IGNORE)
000D8E    057D
000D90    057D                 220          DC      T(IGNORE),T(IGNORE)
000D92    057D
000D94    057D                 220          DC      T(IGNORE),T(IGNORE)
000D96    057D
000D98    057D                 220          DC      T(IGNORE),T(IGNORE)
000D9A    057D
000D9C    057D                 220          DC      T(IGNORE),T(IGNORE)
000D9E    057D
000DA0    057D                 220          DC      T(IGNORE),T(IGNORE)
000DA2    057D
000DA4    057D                 220          DC      T(IGNORE),T(IGNORE)
000DA6    057D
000DA8    057D                 220          DC      T(IGNORE),T(IGNORE)
000DAA    057D
000DAC    057D                 220          DC      T(IGNORE),T(IGNORE)
```

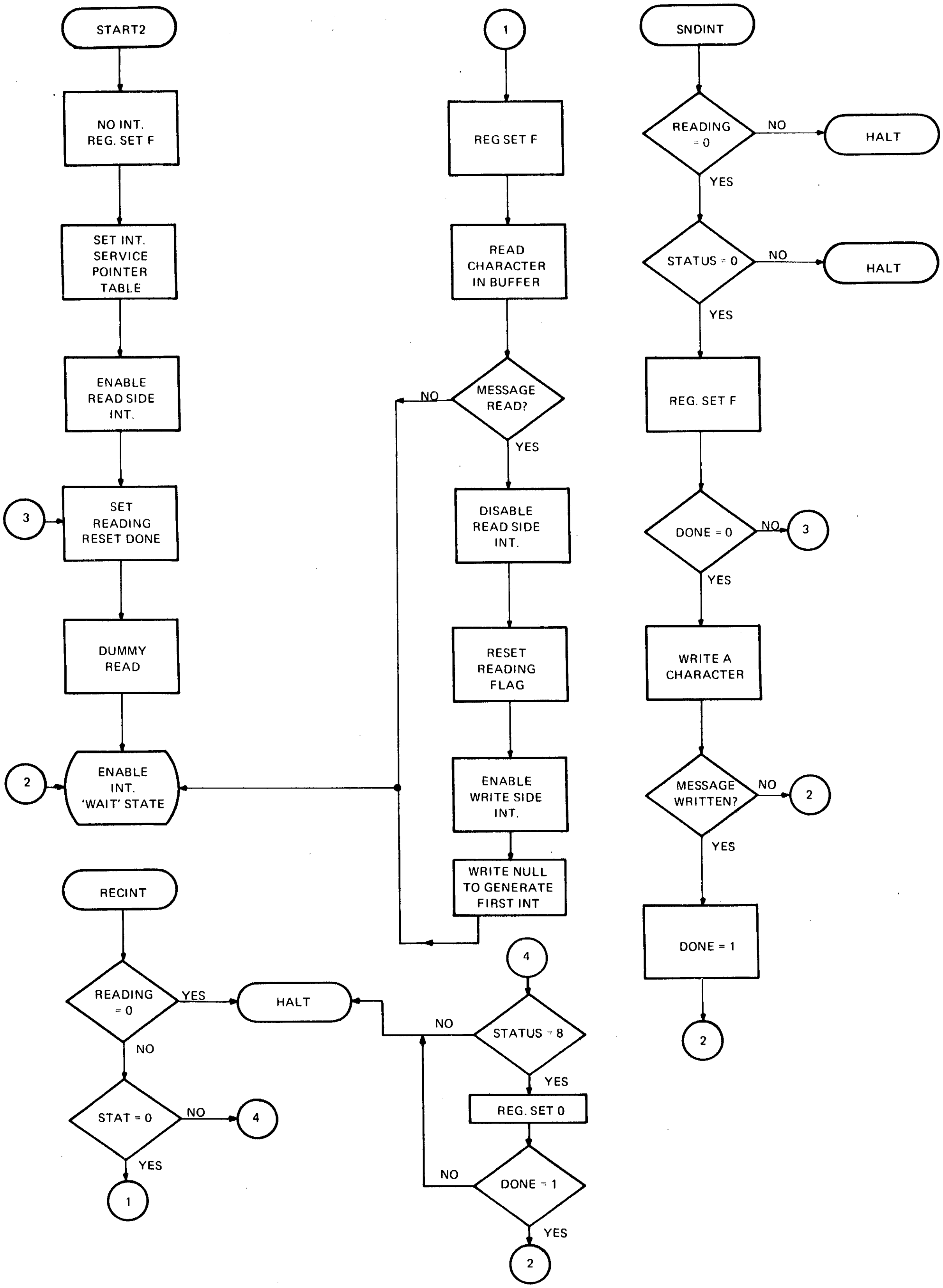
```
     COMM MUX PROGRAMMING EXAMPLES FOR 32-BIT PROCESSOR        PAGE   9  11:10:15  08/16/78

     DATA TRANS THRU AUTO DRIVER CHANNEL 4 WIRE OPERATION
000DAE   057D
000DB0   057D                  220          DC    T(IGNORE),T(IGNORE)
000DB2   057D
000DB4   057D                  220          DC    T(IGNORE),T(IGNORE)
000DB6   057D
                               221  *                                                           CM322210
                               222  *                                                           CM322220
                               223  * CONSTANTS USED IN ABOVE EXAMPLES                          CM322230
                               224  *                                                           CM322240
000DB8   0000                  225  READING  DCX   0                   READ FLAG                CM322250
000DBA   0010                  226  RECADR   DCX   10                  COMM MUX RECEIVE ADDRESS CM322260
000DBC   0011                  227  SNDADR   DCX   11                  COMM MUX SEND ADDRESS    CM322270
000DBE   79                    228  ENREAD   DB    X'79'               ENABLE,ECHO,READ         CM322280
000DBF   B9                    229  DISRD    DB    X'B9'               DISABLE,ECHO,READ        CM322290
000DC0   AB                    230  DISWRT   DB    X'AB'               DISABLE,WRITE            CM322300
000DC1   3B                    231  RQ2S     DB    X'3B'               REQUEST TO SEND (WRT/RD = 1) CM322310
000DC2   6B                    232  ENWRT    DB    X'6B'               ENABLE,WRITE             CM322320
000DC3   F8                    233  SECOND   DB    X'F8'                                        CM322330
                               234  *                                  2 STOP BITS, NO PARITY CHECK CM322340
000DC4                         235           END                                                CM322350
```

```
     COMM MUX PROGRAMMING EXAMPLES FOR 32-BIT PROCESSOR        PAGE  10   11:10:15  08/16/78

     DATA TRANS THRU AUTO DRIVER CHANNEL 4 WIRE OPERATION

ASSEMBLED BY CAL 03-066R05-00   (32-BIT)

START OPTIONS:  T=32,CROSS,ERLST,

    NO CAL ERRORS
    NO CAL WARNINGS
     2 PASSES

  ABSTOP    0000 0DC4
  ADC       0000 0004        150     197     197
  BUFOEND   0000 0B4C         45      77     156*
  BUF1END   0000 0B54        138     159*
  BUFULL    0000 0B18        129     133*
  C         0000 0008         90     110     181*
  CCB       0000 0B48         51     114     153*
  CCW       0000 0B48         40      55      79     105     122     154*
  CCWSTA    0000 F700         39      78     176*
  CHKWORD   0000 0B50        157*
  CMD       0000 0BB4         47      81      94     193*
  COMM      0000 0A38         51*     86
  COUNTO    0000 0B4A         44      68      76     155*
  COUNT1    0000 0B52        136     158*
  CR        0000 0AFC        121*    201     213
  CREXIT    0000 0A64         67*    132
  DEV       0000 0007         25*     35      36      37      53      53      61      62      82      83
  DEVADR    0000 0BB0         37      61      83     191*
  DISRD     0000 0DBF        229*
  DISWRT    0000 0DC0        230*
  ENREAD    0000 0DBE         46     228*
  ENWRT     0000 0DC2         80     232*
  G         0000 0002         90     110     129     147     183*
  IGNORE    0000 0AFA        118*    200     203     203     208     208     212     215     215     220     220
  IMPTOP    0000 0000I
  IODONE    0000 0B3A        139     146*
  ISPTAB    0000 00D0         53     180*
  ISRO      0000 0AAC         56      90*
  ISR01     0000 0AD4         96     102*    115
  ISR1      0000 0AE4         98     110*
  ISR2      0000 0B04        102     128*
  L         0000 0001         90     110     128     146     184*
  LADC      0000 0002
  LINEXIT   0000 0A78         73*    134
  MESSAGE   0000 0B60         42     168*
  OK        0000 0A82         72      76*
  PURETOP   0000 0000P
  RO        0000 0000         23*     33      42      43      44      46      47      60      67      67      70      74      75
                              76      80      81     100     106     118     123     141
  R1        0000 0001         24*     32      33      39      40      41      43      45      51      52      53      54      55
                              56      57      59      60      68      69      70      71      73      75      77      78      79
  RBUFOEND  0000 0BA7         41      73     169*
  RCVADR    0000 0BB2         35     192*
  READ      0000 0BAC         49      85      95     130     133     190*
  READING   0000 0DB8        225*
```

```
     COMM MUX PROGRAMMING EXAMPLES FOR 32-BIT PROCESSOR        PAGE  11  11:10:15  08/16/78

     DATA TRANS THRU AUTO DRIVER CHANNEL 4 WIRE OPERATION

RECADR    0000 0DBA        226*
RQ2S      0000 0DC1        231*
SECOND    0000 0DC3         93    233*
SNDADR    0000 0DBC         82    227*
START1    0000 0A00         32*    92    112    147
SUBR      0000 0B5C         57     99    103    140    161*
TABLE     0000 0BF8        204*   206    206
TABLEH    0000 0CF8        216*   218    218
TLATE     0000 0002         39    178*
TLATETAB  0000 0BB8        160    199*
V         0000 0004         90    110    182*
WBUF1END  0000 0BAB        137    171*
WORK      0000 000D         26*    36     48     49     84     84     85     95     97     98     99    102    103
                           104    105    113    114    121    121    122    130    133    135    136    137    138
                           139    140
WRITE     0000 0004         78    177*
WRTBUF    0000 0B5E         71     74    166*
```

APPENDIX B (Continued)

COMM MUX UNDER INTERRUPTS



```
        COMM MUX PROGRAMMING EXAMPLES FOR 32-BIT PROCESSOR        PAGE   1  14:10:39  08/16/78

  PROG= CM32B      ASSEMBLED BY CAL 03-066R05-00   (32-BIT)
                                   1  CM32B     PROG  COMM MUX PROGRAMMING EXAMPLES FOR 32-BIT PROCESSOR          CM322360
                                   2            CROSS                                                             CM322370
                                   3            WIDTH 120                                                         CM322380
                                   4            TARGT 32                                                          CM322390
                                   5            NORX3                                                             CM322400
                                   6  *                                                                           CM322410
                                   7  *                                                                           CM322420
```

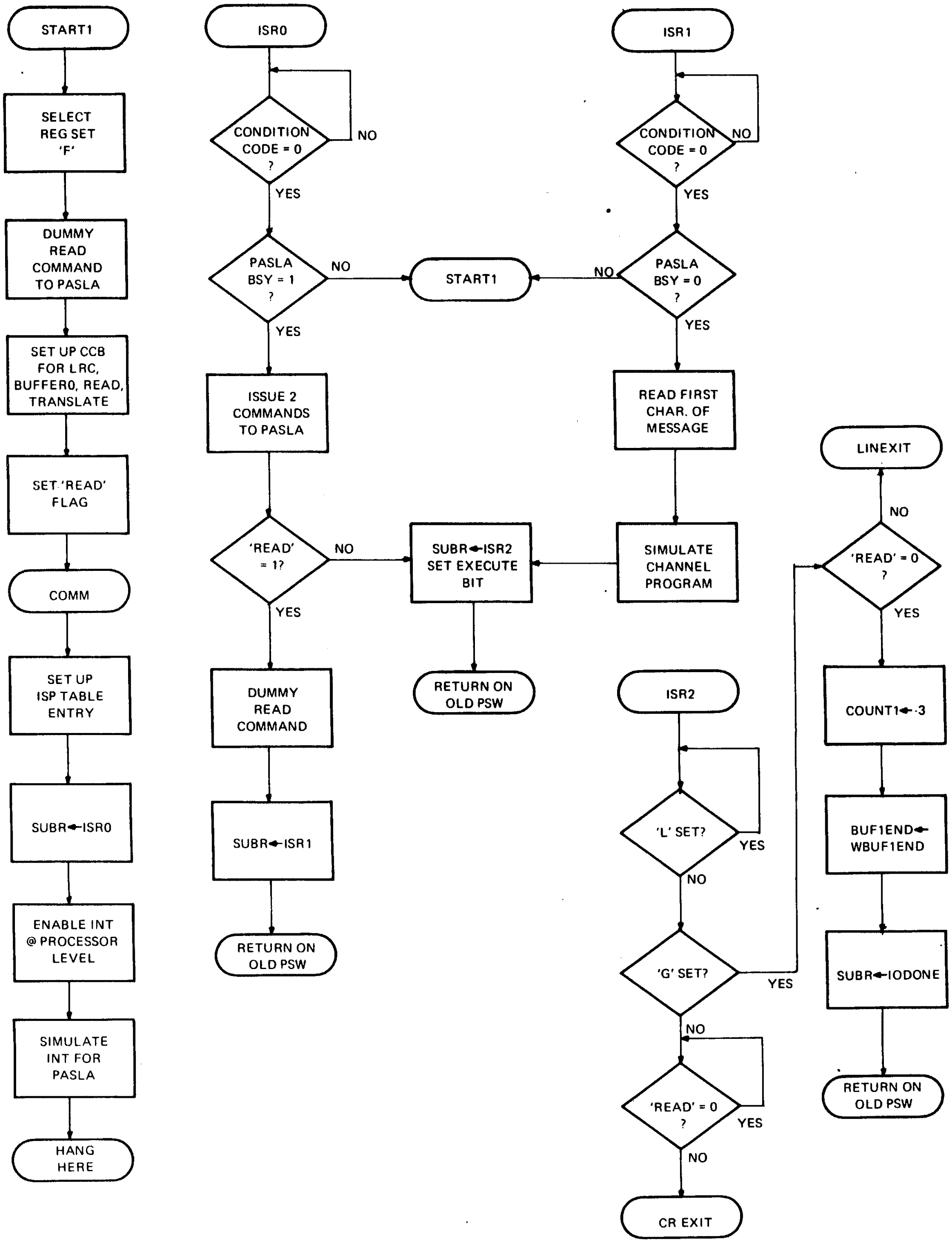
```
COMM MUX PROGRAMMING EXAMPLES FOR 32-BIT PROCESSOR        PAGE   2  14:10:39  08/16/78

LOCAL TERMINAL, FULL DUPLEX  COMM MUX OPERATION
                                                                                               CM322440
                                 9  *                                                          CM322450
                                10  *  THE TERMINAL SHOULD BE INTERFACED THROUGH COMM MUX FDX INTERFACE   CM322450
                                11  *  THE FOLLOWING EXAMPLE IS FOR 32-BIT PROCESSOR           CM322460
                                12  *  START EXECUTION @ START                                  CM322470
                                13  *                                                          CM322480
                                14  *  READ 10 KEYS & WRITE THOSE 10 CHARACTERS UNDER INTERRUPT CONTROL  CM322490
                                15  *                                                          CM322500
                                16  *                                                          CM322510
                                17  *  REGISTER ASSIGNMENTS                                    CM322520
                                18  *                                                          CM322530
          0000 0000             19  DONE      EQU    0                EXAMPLE DONE FLAG        CM322540
          0000 0002             20  DEV0      EQU    2                COMM MUX RECEIVE ADDRESS CM322550
          0000 0003             21  DEV1      EQU    3                COMM MUX SEND ADDRESS    CM322560
          0000 0003             22  R3        EQU    3                STATUS OF INTERRUPTING DEVICE  CM322570
          0000 0004             23  MSG       EQU    4                MESSAGE ADDRESS          CM322580
          0000 0005             24  STAT      EQU    5                COMM MUX STATUS          CM322590
          0000 0008             25  BSY       EQU    8                                         CM322600
          0000 000A             26  R10       EQU    10                                        CM322610
          0000 000B             27  R11       EQU    11               WORK REGISTER            CM322620
          0000 000C             28  R12       EQU    12               WORK REGISTER            CM322630
          0000 000D             29  WORK      EQU    13                                        CM322640
                                30  *                                                          CM322650
000000I                         31            ORG    X'A00'                                    CM322660
000A00    F8D0 0000 00F0        32  START     LI     WORK,Y'F0'                                CM322670
000A06    954D                  33            EPSR   MSG,WORK         NO INT, REG SET F        CM322680
                                34  *                                                          CM322690
                                35  * SET UP INTERRUPT SERVICE POINTER TABLE                   CM322700
                                36  *                                                          CM322710
000A08    E640 0AF6             37            LDAI   MSG,HALT                                  CM322720
000A0C    F6A0 00D0             38            LDAI   R10,X'D0'                                 CM322730
000A10    24B2                  39            LIS    R11,2                                     CM322740
000A12    E6C0 02CE             40            LDAI   R12,X'2CE'                                CM322750
000A16    404A 0000             41  EXMP2A    STH    MSG,0(R10)       SET UP ENTIRE TABLE WITH 'HALT'  CM322760
000A1A    C1A0 0A16             42            BXLE   R10,EXMP2A                                CM322770
                                43  *                                                          CM322780
000A1E    4820 0AFC             44            LH     DEV0,RECADR      GET BOTH ADDRESSES       CM322790
000A22    4830 0AFE             45            LH     DEV1,SNDADR                               CM322800
000A26    DE20 0B05             46            OC     DEV0,SECOND      SET UP  COMM MUX         CM322810
000A2A    08A2                  47            LR     R10,DEV0         SET UP 2 TABLE ENTRIES FOR  CM322820
000A2C    0AAA                  48            AAR    R10,R10          RECEIVE & TRANSMIT DEVICES.  CM322830
000A2E    C8D0 0A62             49            LHI    WORK,RECINT                               CM322840
000A32    40DA 00D0             50            STH    WORK,X'D0'(R10)                           CM322850
000A36    08A3                  51            LR     R10,DEV1 •                                CM322860
000A38    0AAA                  52            AAR    R10,R10                                   CM322870
000A3A    C8D0 0AC8             53            LHI    WORK,SNDINT                               CM322880
000A3E    40DA 00D0             54            STH    WORK,X'D0'(R10)                           CM322890
                                55  *                                                          CM322900
000A42    0700                  56  REPEAT2   XAR    DONE,DONE                                 CM322910
000A44    E640 0B0A             57            LDAI   MSG,MSG33                                 CM322920
000A48    4040 0AFA             58            STH    MSG,READING      RESET FLAG               CM322930
000A4C    DE20 0B00             59            OC     DEV0,ENREAD      ENABLE READ SIDE INT, ECHO MODE  CM322940
000A50    DE20 0B03             60            OC     DEV0,RQ2S                                 CM322950
000A54    9B2D                  61            RDR    DEV0,WORK        DUMMY READ TO SET BSY    CM322960
```

```
          COMM MUX PROGRAMMING EXAMPLES FOR 32-BIT PROCESSOR        PAGE   3  14:10:39  08/16/78

          LOCAL TERMINAL, FULL DUPLEX  COMM MUX OPERATION
000A56    F8D0 0000 C0F0       62 EXMP2B  LI    WORK,Y'C0F0'        ENABLE INT & HALT TILL DEPRESSION     CM322970
000A5C    95AD                 63         EPSR  R10,WORK            OF A KEY INTERRUPTS                   CM322980
000A5E    4300 0A5E            64         B     *                                                         CM322990
                               65 *                                                                       CM323000
                               66 * RECEIVE SIDE HAS INTERRUPTED                                          CM323010
                               67 *                                                                       CM323020
000A62    48A0 0AFA            68 RECINT  LH    R10,READING         HALT IF RECEIVE SIDE INTERRUPTS       CM323030
000A66    4330 0A66            69         BZ    *                   WHEN 'READING' FLAG IS RESET          CM323040
000A6A    0833                 70         LR    R3,R3               (R3) = RECEIVE SIDE STATUS            CM323050
000A6C    4330 0A8A            71         BZ    TEST1                                                     CM323060
000A70    C530 0008            72         CLHI  R3,8                                                      CM323070
000A74    4230 0AF6            73         BNE   HALT                                                      CM323080
000A78    C8D0 00F0            74         LHI   WORK,X'F0'                                                CM323090
000A7C    95AD                 75         EPSR  R10,WORK                                                  CM323100
000A7E    08C0                 76         LR    DONE,DONE                                                 CM323110
000A80    4230 0AF6            77         BNZ   HALT                                                      CM323120
000A84    2401                 78         LIS   DONE,1                                                    CM323130
000A86    4300 0A56            79         B     EXMP2B                                                    CM323140
          0000 0A8A            80 TEST1   EQU   *                                                         CM323150
000A8A    F8D0 0000 00F0       81         LI    WORK,Y'F0'                                                CM323160
000A90    95AD                 82         EPSR  R10,WORK            REGISTER SET F                        CM323170
000A92    2401                 83         LIS   DONE,1                                                    CM323180
000A94    DB24 0000            84         RD    DEV0,0(MSG)         READ BYTE INTO MESSAGE BUFFER         CM323190
000A98    2641                 85         AIS   MSG,1                                                     CM323200
000A9A    F540 0000 0B14       86         CLAI  MSG,MSG33END                                              CM323210
000AA0    4280 0A56            87         BL    EXMP2B              LOOP TILL 10 KEYS ARE READ IN.        CM323220
                               88 *                                                                       CM323230
                               89 * TO WRITE THE MESSAGE JUST READ IN                                     CM323240
                               90 *                                                                       CM323250
000AA4    DE20 0B01            91         OC    DEV0,DISRD          DISABLE READ SIDE INTERRUPTS          CM323260
000AA8    DE20 0B03            92         OC    DEV0,RQ2S                                                 CM323270
000AAC    E640 0B06            93         LDAI  MSG,MSG3                                                  CM323280
000AB0    2400                 94         LIS   DONE,0                                                    CM323290
000AB2    4000 0AFA            95         STH   DONE,READING        RESET FLAG                            CM323300
000AB6    DE30 0B04            96         OC    DEV1,ENWRT          ENABLE WRITE SIDE INTERRUPTS          CM323310
000ABA    9A30                 97         WDR   DEV1,DONE           WRITE NULL TO GENERATE FIRST INT.     CM323320
000ABC    F8D0 0000 C0F0       98 EXMP2C  LI    WORK,Y'C0F0'                                              CM323330
000AC2    95AD                 99         EPSR  R10,WORK            ENABLE INT @ PROCESSOR LEVEL & HALT   CM323340
000AC4    4300 0AC4           100         B     *                                                         CM323350
                              101 *                                                                       CM323360
                              102 * TRANSMIT SIDE HAS INTERRUPTED                                         CM323370
                              103 *                                                                       CM323380
000AC8    48A0 0AFA           104 SNDINT  LH    R10,READING                                               CM323390
000ACC    4230 0ACC           105         BNZ   *                                                         CM323400
000AD0    0833                106         LR    R3,R3               STATUS SHOULD BE ZERO                 CM323410
000AD2    4230 0AD2           107         BNZ   *                                                         CM323420
000AD6    F8D0 0000 00F0      108         LI    WORK,Y'F0'                                                CM323430
000ADC    95AD                109         EPSR  R10,WORK            REGISTER SET F                        CM323440
000ADE    DA34 0000           110         WD    DEV1,0(MSG)                                               CM323450
000AE2    2641                111         AIS   MSG,1                                                     CM323460
000AE4    F540 0000 0B18      112         CLAI  MSG,MSG3END                                               CM323470
000AEA    4280 0ABC           113         BL    EXMP2C                                                    CM323480
000AEE    DE30 0B02           114         OC    R3,DISWRT                                                 CM323490
```

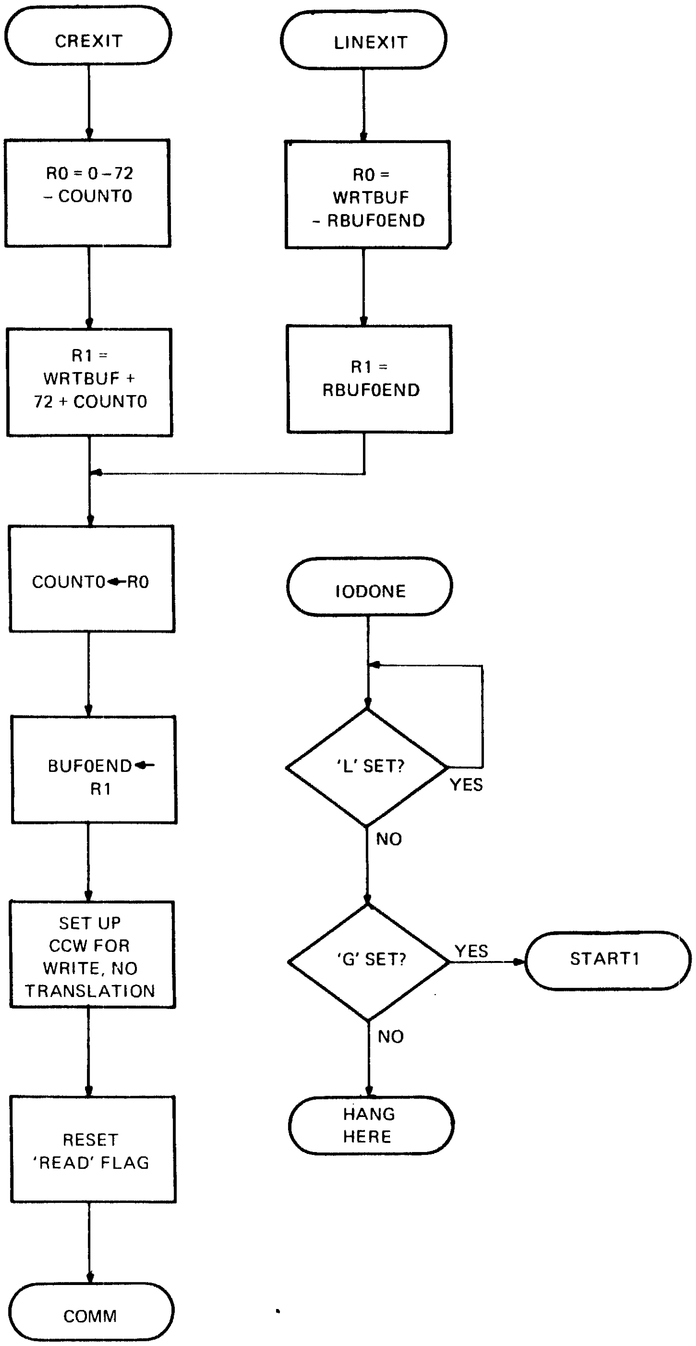
```
     COMM MUX PROGRAMMING EXAMPLES FOR 32-BIT PROCESSOR        PAGE   4  14:10:39  08/16/78

     LOCAL TERMINAL, FULL DUPLEX  COMM MUX OPERATION
000AF2   4300 0A42              115            B     REPEAT2                                                 CM323500
000AF6   4300 0AF6              116  HALT      B     *                                                       CM323510
                                117  *                                                                       CM323520
                                118  *                                                                       CM323530
                                119  * CONSTANTS USED IN ABOVE EXAMPLES                                      CM323540
                                120  *                                                                       CM323550
000AFA   0000                   121  READING   DCX   0                      READ FLAG                        CM323560
000AFC   0010                   122  RECADR    DCX   10                     COMM MUX RECEIVE ADDRESS         CM323570
000AFE   0011                   123  SNDADR    DCX   11                     COMM MUX SEND ADDRESS            CM323580
000B00   79                     124  ENREAD    DB    X'79'                  ENABLE,ECHO,READ                 CM323590
000B01   B9                     125  DISRD     DB    X'B9'                  DISABLE,ECHO,READ                CM323600
000B02   AB                     126  DISWRT    DB    X'AB'                  DISABLE,WRITE                    CM323610
000B03   3B                     127  RQ2S      DB    X'3B'                  REQUEST TO SEND (WRT/RD = 1)     CM323620
000B04   6B                     128  ENWRT     DB    X'6B'                  ENABLE,WRITE                     CM323630
000B05   F8                     129  SECOND    DB    X'F8'                                                   CM323640
                                130  *                                      2 STOP BITS, NO PARITY CHECK     CM323650
000B06   FFFF                   131  MSG3      DCX   FFFF                                                    CM323660
000B08   0D0A                   132            DC    X'0D0A'                                                 CM323670
000B0A                          133  MSG33     DS    10                     10 CHARACTER MESSAGE BUFFER      CM323680
         0000 0B14              134  MSG33END  EQU   *                                                       CM323690
000B14   0D0A                   135            DC    X'0D0A'                                                 CM323700
000B16   0000                   136            DCX   0                      2 NULL CHARACTERS                CM323710
         0000 0B18              137  MSG3END   EQU   *                                                       CM323720
                                138  *                                                                       CM323730
000B18                          139            END                                                           CM323740
```

```
        COMM MUX PROGRAMMING EXAMPLES FOR 32-BIT PROCESSOR        PAGE   5  14:10:39  08/16/78

        LOCAL TERMINAL, FULL DUPLEX  COMM MUX OPERATION

ASSEMBLED BY CAL 03-066R05-00   (32-BIT)

START OPTIONS:  T=32,CROSS,ERLST,

    NO CAL ERRORS
    NO CAL WARNINGS
     2 PASSES

 ABSTOP      0000 0B18
 ADC         0000 0004
 BSY         0000 0008          25*
 DEV0        0000 0002          20*    44     46     47     59     60     61     84     91     92
 DEV1        0000 0003          21*    45     51     96     97    110
 DISRD       0000 0B01          91    125*
 DISWRT      0000 0B02         114    126*
 DONE        0000 0000          19*    56     56     76     76     78     83     94     95     97
 ENREAD      0000 0B00          59    124*
 ENWRT       0000 0B04          96    128*
 EXMP2A      0000 0A16          41*    42
 EXMP2B      0000 0A56          62*    79     87
 EXMP2C      0000 0ABC          98*   113
 HALT        0000 0AF6          37     73     77    116*
 IMPTOP      0000 0000I
 LADC        0000 0002
 MSG         0000 0004          23*    33     37     41     57     58     84     85     86     93    110    111    112
 MSG3        0000 0B06          93    131*
 MSG33       0000 0B0A          57    133*
 MSG33END    0000 0B14          86    134*
 MSG3END     0000 0B18         112    137*
 PURETOP     0000 0000P
 R10         0000 000A          26*    38     41     42     47     48     48     50     51     52     52     54     63
                                68     75     82     99    104    109
 R11         0000 000B          27*    39
 R12         0000 000C          28*    40
 R3          0000 0003          22*    70     70     72    106    106    114
 READING     0000 0AFA          58     68     95    104    121*
 RECADR      0000 0AFC          44    122*
 RECINT      0000 0A62          49     68*
 REPEAT2     0000 0A42          56*   115
 RQ2S        0000 0B03          60     92    127*
 SECOND      0000 0B05          46    129*
 SNDADR      0000 0AFE          45    123*
 SNDINT      0000 0AC8          53    104*
 START       0000 0A00          32*
 STAT        0000 0005          24*
 TEST1       0000 0A8A          71     80*
 WORK        0000 000D          29*    32     33     49     50     53     54     61     62     63     74     75     81
                                82     98     99    108    109
```

FLOW CHART FOR PASLA PROGRAMMING USING AUTO DRIVER CHANNEL



FLOW CHART FOR PASLA PROGRAMMING USING AUTO DRIVER CHANNEL (Continued)



```
        PASLA PROGRAMMING EXAMPLES FOR 32-BIT PROCESSOR          PAGE   1  16:16:49  11/29/79
   PROG=           ASSEMBLED BY CAL 03-066R07-00   (32-BIT)
                                  1          PROG  PASLA PROGRAMMING EXAMPLES FOR 32-BIT PROCESSOR          PS310000
                                  2          CROSS                                                          PS310020
                                  3          WIDTH 120                                                      PS310030
                                  4          TARGT 32                                                       PS310040
                                  5          NORX3                                                          PS310050
                                  6  *                                                                      PS310060
                                  7  *                                                                      PS310070
                                  8  *  USER INPUTS A MESSAGE OF UP TO 72 CHARACTERS THROUGH THE TERMINAL    PS310080
                                  9  *  HOOKED ON TO THE PASLA INTERFACE. A MESSAGE OF LESS THAN 72         PS310090
                                 10  *  CHARACTERS MUST BE TERMINATED BY DEPRESSING 'CR' KEY TWICE. THE     PS310100
                                 11  *  AUTO-DRIVER CHANNEL READS THE CHARACTERS & DOES ASCII TO ASCII      PS310110
                                 12  *  TRANSLATION, AND GENERATES A BUFFER CALLED 'MESSAGE'.              PS310120
                                 13  *  THEN CCB IS SET UP TO WRITE, FAST MODE, NO TRANSLATION. THE        PS310130
                                 14  *  'CR', 'LF' OR JUST 'LF' IS OUTPUT FOLLOWED BY THE 'MESSAGE'        PS310140
                                 15  *  BUFFER. THEN 'CR', 'LF' ARE OUTPUT.                                PS310150
                                 16  *                                                                      PS310160
                                 17  *  THE ENTIRE PROGRAM LOOPS ON ITSELF.                                 PS310170
                                 18  *                                                                      PS310180
                                 19  *                                                                      PS310190
                                 20  * REGISTER EQUATES                                                     PS310200
                                 21  *                                                                      PS310210
                                 22  *                                                                      PS310220
         0000 0000               23  R0        EQU   0                                                      PS310230
         0000 0001               24  R1        EQU   1                                                      PS310240
         0000 0007               25  DEV       EQU   7                                                      PS310250
         0000 000A               26  WORK      EQU   10                                                     PS310260
```

```
PASLA PROGRAMMING EXAMPLES FOR 32-BIT PROCESSOR          PAGE   2  16:16:49  11/29/79

DATA TRANSFER THROUGH AUTO-DRIVER CHANNEL

                                   28  *                                                               PS310280
000000I                            29           ORG   X'A00'                                           PS310290
                                   30  *                                                               PS310300
                                   31  *                                                               PS310310
000A00   C810 00F0                 32  START1   LHI   R1,X'F0'                                         PS310320
000A04   9501                      33           EPSR  R0,R1              REGISTER SET F                PS310330
                                   34  *                                                               PS310340
000A06   4870 0BB2                 35           LH    DEV,RCVADR         GET RECEIVE SIDE ADDRESS      PS310350
000A0A   9B7A                      36           RDR   DEV,WORK           DUMMY READ TO SET BUSY        PS310360
000A0C   4070 0BB0                 37           STH   DEV,DEVADR                                       PS310370
                                   38  *                                                               PS310380
000A10   C810 F702                 39           LHI   R1,CCWSTA+TLATE                                  PS310390
000A14   4010 0B48                 40           STH   R1,CCW             LRC, BUFFER 0, READ, TRANSLATE  PS310400
000A18   E610 0BA7                 41           LDAI  R1,RBUF0END                                      PS310410
000A1C   E600 0B60                 42           LDAI  R0,MESSAGE                                       PS310420
000A20   0B01                      43           SAR   R0,R1                                            PS310430
000A22   4000 0B4A                 44           STH   R0,COUNT0                                        PS310440
000A26   5010 0B4C                 45           STA   R1,BUF0END                                       PS310450
000A2A   D300 0BA2                 46           LB    R0,ENREAD          GET READ COMMAND              PS310460
000A2E   D200 0BB4                 47           STB   R0,CMD                                           PS310470
000A32   24A1                      48           LIS   WORK,1                                           PS310480
000A34   40A0 0BAC                 49           STH   WORK,READ          SET FLAG                      PS310490
                                   50  *                                                               PS310500
000A38   C810 0B48                 51  COMM     LHI   R1,CCB                                           PS310510
000A3C   2611                      52           AIS   R1,1                                             PS310520
000A3E   4017 4700 00D0            53           STH   R1,ISPTAB(DEV,DEV) SET UP ISP TABLE ENTRY        PS310530
000A44   2418                      54           LIS   R1,8                                             PS310540
000A46   7610 0B48                 55           RBT   R1,CCW             RESET EX BIT                  PS310550
000A4A   C810 0AAC                 56           LHI   R1,ISR0                                          PS310560
000A4E   4010 0B5C                 57           STH   R1,SUBR            SUBROUTINE ADDRESS            PS310570
                                   58  *                                                               PS310580
000A52   C810 40F0                 59           LHI   R1,X'40F0'                                       PS310590
000A56   9501                      60           EPSR  R0,R1              ENABLE INT a PROCESSOR LEVEL  PS310600
000A58   4870 0BB0                 61           LH    DEV,DEVADR                                       PS310610
000A5C   E207 0000                 62           SINT  0(DEV)                                           PS310620
000A60   4300 0A60                 63           B     *                  HANG                          PS310630
                                   64  *                                                               PS310640
                                   65  *  TO WRITE CR, LF OR JUST LF FOLLOWED BY MESSAGE & CR, LF.     PS310650
                                   66  *                                                               PS310660
000A64   0700                      67  CREXIT   XAR   R0,R0                                            PS310670
000A66   4810 0B4A                 68           LH    R1,COUNT0                                        PS310680
000A6A   CA10 0048                 69           AHI   R1,72                                            PS310690
000A6E   0B01                      70           SAR   R0,R1                                            PS310700
000A70   FA10 0000 0B5E            71           AAI   R1,WRTBUF                                        PS310710
000A76   2306                      72           BS    OK                                               PS310720
000A78   E610 0BA7                 73  LINEXIT  LDAI  R1,RBUF0END                                      PS310730
000A7C   E600 0B5E                 74           LDAI  R0,WRTBUF                                        PS310740
000A80   0B01                      75           SAR   R0,R1                                            PS310750
000A82   4000 0B4A                 76  OK       STH   R0,COUNT0                                        PS310760
000A86   5010 0B4C                 77           STA   R1,BUF0END                                       PS310770
000A8A   C810 F704                 78           LHI   R1,CCWSTA+WRITE      WRITE, NO TRANSLATION       PS310780
000A8E   4010 0B48                 79           STH   R1,CCW                                           PS310790
000A92   D300 0BA6                 80           LB    R0,ENWRT                                         PS310800
```

```
        PASLA PROGRAMMING EXAMPLES FOR 32-BIT PROCESSOR          PAGE   3  16:16:49  11/29/79

        DATA TRANSFER THROUGH AUTO-DRIVER CHANNEL
000A96   D200 0BB4            81          STB   R0,CMD                                        PS310810
000A9A   4870 0BA0            82          LH    DEV,SNDADR                                    PS310820
000A9E   4070 0BB0            83          STH   DEV,DEVADR                                    PS310830
000AA2   07AA                 84          XAR   WORK,WORK                                     PS310840
000AA4   40A0 0BAC            85          STH   WORK,READ           RESET FLAG                PS310850
000AA8   4300 0A38            86          B     COMM                                          PS310860
                              87  *                                                           PS310870
                              88  *  COME HERE AFTER SINT                                     PS310880
                              89  *                                                           PS310890
000AAC   42F0 0AAC            90  ISR0    BTC   C+V+G+L,*                                     PS310900
000AB0   C330 0008            91          THI   3,8                                           PS310910
000AB4   4330 0A00            92          BZ    START1                                        PS310920
000AB8   DE20 0BA7            93          OC    2,SECOND            SET UP PASLA              PS310930
000ABC   DE20 0BB4            94          OC    2,CMD               ISSUE COMMAND             PS310940
000AC0   48A0 0BAC            95          LH    WORK,READ                                     PS310950
000AC4   4330 0AD4            96          BZ    ISR01                                         PS310960
000AC8   9B2A                 97          RDR   2,WORK              SET BSY                   PS310970
000ACA   E6A0 0AE4            98          LDAI  WORK,ISR1                                     PS310980
000ACE   40A0 0B5C            99          STH   WORK,SUBR           TO READ A MESSAGE         PS310990
000AD2   1800                100          LPSWR R0                                            PS311000
                             101  *                                                           PS311010
000AD4   E6A0 0B04           102  ISR01   LDAI  WORK,ISR2                                     PS311020
000AD8   40A0 0B5C           103          STH   WORK,SUBR           TO WRITE THE MESSAGE      PS311030
000ADC   24A8                104          LIS   WORK,8                                        PS311040
000ADE   75A0 0B48           105          SBT   WORK,CCW            SET EX BIT                PS311050
000AE2   1800                106          LPSWR R0                                            PS311060
                             107  *                                                           PS311070
                             108  * COME HERE TO READ FIRST CHARACTER                         PS311080
                             109  *                                                           PS311090
000AE4   42F0 0AE4           110  ISR1    BTC   C+V+G+L,*                                     PS311100
000AE8   C330 0008           111          THI   3,8                                           PS311110
000AEC   4230 0A00           112          BNZ   START1                                        PS311120
000AF0   9B2A                113          RDR   2,WORK              READ FIRST CHARACTER OF MESSAGE PS311130
000AF2   E3A0 0B48           114          SCP   WORK,CCB            PUT IT IN BUFFER          PS311140
000AF6   4300 0AD4           115          B     ISR01                                         PS311150
                             116  *                                                           PS311160
                             117  *                                                           PS311170
000AFA   1800                118  IGNORE  LPSWR R0                                            PS311180
                             119  *                                                           PS311190
                             120  *                                                           PS311200
000AFC   07AA                121  CR      XAR   WORK,WORK                                     PS311210
000AFE   40A0 0B48           122          STH   WORK,CCW            RESET EXE BIT             PS311220
000B02   1800                123          LPSWR R0                                            PS311230
                             124  *                                                           PS311240
                             125  *                                                           PS311250
                             126  *  COME HERE AFTER ADC TERMINATION                          PS311260
                             127  *                                                           PS311270
000B04   4210 0B04           128  ISR2    BTC   L,*                                           PS311280
000B08   4220 0B18           129          BTC   G,BUFULL                                      PS311290
000B0C   48A0 0BAC           130          LH    WORK,READ                                     PS311300
000B10   4330 0B10           131          BZ    *                                             PS311310
000B14   4300 0A64           132          B     CREXIT                                        PS311320
000B18   48A0 0BAC           133  BUFULL  LH    WORK,READ                                     PS311330
```

```
PASLA PROGRAMMING EXAMPLES FOR 32-BIT PROCESSOR          PAGE   4  16:16:49  11/29/79

DATA TRANSFER THROUGH AUTO-DRIVER CHANNEL

000B1C   4230 0A78         134           BNZ   LINEXIT                                  PS311340
000B20   C8A0 FFFD         135           LHI   WORK,-3          TO OUTPUT CR, LF, NULL, NULL.   PS311350
000B24   40A0 0B52         136           STH   WORK,COUNT1                              PS311360
000B28   E6A0 0BAB         137           LDAI  WORK,WBUF1END                            PS311370
000B2C   50A0 0B54         138           STA   WORK,BUF1END                             PS311380
000B30   C8A0 0B3A         139           LHI   WORK,IODONE                              PS311390
000B34   40A0 0B5C         140           STH   WORK,SUBR                                PS311400
000B38   1800              141           LPSWR R0                                       PS311410
                           142   *                                                      PS311420
                           143   *  COME HERE WHEN ALL I/O OPERATION IS OVER            PS311430
                           144   *                                                      PS311440
                           145   *                                                      PS311450
000B3A   4210 0B3A         146   IODONE  BTC   L,*                                      PS311460
000B3E   4220 0A00         147           BTC   G,START1                                 PS311470
000B42   4300 0B42         148           B     *                                        PS311480
                           149   *                                                      PS311490
000B48                     150           ALIGN ADC                                      PS311500
                           151   *-----------------------------------                   PS311510
                           152   *                                                      PS311520
         0000 0B48         153   CCB     EQU   *                                        PS311530
000B48   0000              154   CCW     DCX   0                                        PS311540
000B4A   0000              155   COUNT0  DCX   0                                        PS311550
000B4C   0000 0000         156   BUF0END DC    0                                        PS311560
000B50   0000              157   CHKWORD DCX   0                                        PS311570
000B52   0000              158   COUNT1  DCX   0                                        PS311580
000B54   0000 0000         159   BUF1END DC    0                                        PS311590
000B58   0000 0BB8         160           DC    A(TLATETAB)                              PS311600
000B5C   0000              161   SUBR    DCX   0                                        PS311610
                           162   *                                                      PS311620
                           163   *-----------------------------------                   PS311630
                           164   *  BUFFERS                                             PS311640
                           165   *                                                      PS311650
000B5E   0D                166   WRTBUF  DB    13                                       PS311660
000B5F   0A                167           DB    10                                       PS311670
000B60                     168   MESSAGE DS    72                                       PS311680
         0000 0BA7         169   RBUF0END EQU  *-1                                      PS311690
000BA8   0D0A              170           DCX   0D0A,0                                   PS311700
000BAA   0000
         0000 0BAB         171   WBUF1END EQU  *-1                                      PS311710
                           172   *-----------------------------------                   PS311720
                           173   *                                                      PS311730
                           174   *  EQUATES                                             PS311740
                           175   *                                                      PS311750
         0000 F700         176   CCWSTA  EQU   X'F700'                                  PS311760
         0000 0004         177   WRITE   EQU   X'0004'                                  PS311770
         0000 0002         178   TLATE   EQU   X'0002'                                  PS311780
                           179   *                                                      PS311790
         0000 00D0         180   ISPTAB  EQU   X'D0'                                    PS311800
         0000 0008         181   C       EQU   8                                        PS311810
         0000 0004         182   V       EQU   4                                        PS311820
         0000 0002         183   G       EQU   2                                        PS311830
         0000 0001         184   L       EQU   1                                        PS311840
                           185   *                                                      PS311850
```

```
          PASLA PROGRAMMING EXAMPLES FOR 32-BIT PROCESSOR          PAGE   5  16:16:49  11/29/79

          DATA TRANSFER THROUGH AUTO-DRIVER CHANNEL
                                          186  *                                                    PS311860
                                          187  *  CONSTANTS                                        PS311870
                                          188  *                                                    PS311880
                                          189  *                                                    PS311890
000BAC    0000 0000                       190  READ     DC    0                                     PS311900
000BB0    0000                            191  DEVADR   DCX   0                                     PS311910
000BB2    0010                            192  RCVADR   DCX   10                                    PS311920
000BB4    00                              193  CMD      DB    0                                     PS311930
                                          194  *                                                    PS311940
                                          195  *----------------------------------                  PS311950
                                          196  *                                                    PS311960
000BB8                                    197           ALIGN ADC+ADC                               PS311970
                                          198  *                                                    PS311980
000BB8                                    199  TLATETAB DO    13                                    PS311990
000BB8    057D                            200           DC    T(IGNORE)                             PS312000
000BBA    057D                            200           DC    T(IGNORE)
000BBC    057D                            200           DC    T(IGNORE)
000BBE    057D                            200           DC    T(IGNORE)
000BC0    057D                            200           DC    T(IGNORE)
000BC2    057D                            200           DC    T(IGNORE)
000BC4    057D                            200           DC    T(IGNORE)
000BC6    057D                            200           DC    T(IGNORE)
000BC8    057D                            200           DC    T(IGNORE)
000BCA    057D                            200           DC    T(IGNORE)
000BCC    057D                            200           DC    T(IGNORE)
000BCE    057D                            200           DC    T(IGNORE)
000BD0    057D                            200           DC    T(IGNORE)
000BD2    057E                            201           DC    T(CR)                                 PS312010
000BD4                                    202           DO    9                                     PS312020
000BD4    057D                            203           DC    T(IGNORE),T(IGNORE)                   PS312030
000BD6    057D
000BD8    057D                            203           DC    T(IGNORE),T(IGNORE)
000BDA    057D
000BDC    057D                            203           DC    T(IGNORE),T(IGNORE)
000BDE    057D
000BE0    057D                            203           DC    T(IGNORE),T(IGNORE)
000BE2    057D
000BE4    057D                            203           DC    T(IGNORE),T(IGNORE)
000BE6    057D
000BE8    057D                            203           DC    T(IGNORE),T(IGNORE)
000BEA    057D
000BEC    057D                            203           DC    T(IGNORE),T(IGNORE)
000BEE    057D
000BF0    057D                            203           DC    T(IGNORE),T(IGNORE)
000BF2    057D
000BF4    057D                            203           DC    T(IGNORE),T(IGNORE)
000BF6    057D
          0000 0BF8                       204  TABLE    EQU   *                                     PS312040
000BF8                                    205           DO    32                                    PS312050
000BF8    8020 8021                       206           DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2 PS312060
000BFC    8022 8023                       206           DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C00    8024 8025                       206           DB    X'80',*-TABLE+63/2,X'80''*-TABLE+63/2
000C04    8026 8027                       206           DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
```

```
          PASLA PROGRAMMING EXAMPLES FOR 32-BIT PROCESSOR          PAGE   6  16:16:49  11/29/79

          DATA TRANSFER THROUGH AUTO-DRIVER CHANNEL

000C08    8028 8029           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C0C    802A 802B           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C10    802C 802D           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C14    802E 802F           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C18    8030 8031           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C1C    8032 8033           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C20    8034 8035           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C24    8036 8037           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C28    8038 8039           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C2C    803A 803B           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C30    803C 803D           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C34    803E 803F           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C38    8040 8041           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C3C    8042 8043           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C40    8044 8045           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C44    8046 8047           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C48    8048 8049           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C4C    804A 804B           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C50    804C 804D           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C54    804E 804F           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C58    8050 8051           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C5C    8052 8053           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C60    8054 8055           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C64    8056 8057           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C68    8058 8059           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C6C    805A 805B           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C70    805C 805D           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C74    805E 805F           206        DB    X'80',*-TABLE+63/2,X'80',*-TABLE+63/2
000C78                        207        DO    16                                                     PS312070
000C78    057D                208        DC    T(IGNORE),T(IGNORE)                                    PS312080
000C7A    057D
000C7C    057D                208        DC    T(IGNORE),T(IGNORE)
000C7E    057D
000C80    057D                208        DC    T(IGNORE),T(IGNORE)
000C82    057D
000C84    057D                208        DC    T(IGNORE),T(IGNORE)
000C86    057D
000C88    057D                208        DC    T(IGNORE),T(IGNORE)
000C8A    057D
000C8C    057D                208        DC    T(IGNORE),T(IGNORE)
000C8E    057D
000C90    057D                208        DC    T(IGNORE),T(IGNORE)
000C92    057D
000C94    057D                208        DC    T(IGNORE),T(IGNORE)
000C96    057D
000C98    057D                208        DC    T(IGNORE),T(IGNORE)
000C9A    057D
000C9C    057D                208        DC    T(IGNORE),T(IGNORE)
000C9E    057D
000CA0    057D                208        DC    T(IGNORE),T(IGNORE)
000CA2    057D
000CA4    057D                208        DC    T(IGNORE),T(IGNORE)
000CA6    057D
```

DATA TRANSFER THROUGH AUTO-DRIVER CHANNEL

```
000CA8   057D                 208            DC     T(IGNORE),T(IGNORE)
000CAA   057D
000CAC   057D                 208            DC     T(IGNORE),T(IGNORE)
000CAE   057D
000CB0   057D                 208            DC     T(IGNORE),T(IGNORE)
000CB2   057D
000CB4   057D                 208            DC     T(IGNORE),T(IGNORE)
000CB6   057D
                              209  *                                                 PS312090
                              210  *                                                 PS312100
000CB8                        211            DO     13                               PS312110
000CB8   057D                 212            DC     T(IGNORE)                        PS312120
000CBA   057D                 212            DC     T(IGNORE)
000CBC   057D                 212            DC     T(IGNORE)
000CBE   057D                 212            DC     T(IGNORE)
000CC0   057D                 212            DC     T(IGNORE)
000CC2   057D                 212            DC     T(IGNORE)
000CC4   057D                 212            DC     T(IGNORE)
000CC6   057D                 212            DC     T(IGNORE)
000CC8   057D                 212            DC     T(IGNORE)
000CCA   057D                 212            DC     T(IGNORE)
000CCC   057D                 212            DC     T(IGNORE)
000CCE   057D                 212            DC     T(IGNORE)
000CD0   057D                 212            DC     T(IGNORE)
000CD2   057E                 213            DC     T(CR)                            PS312130
000CD4                        214            DO     9                                PS312140
000CD4   057D                 215            DC     T(IGNORE),T(IGNORE)              PS312150
000CD6   057D
000CD8   057D                 215            DC     T(IGNORE),T(IGNORE)
000CDA   057D
000CDC   057D                 215            DC     T(IGNORE),T(IGNORE)
000CDE   057D
000CE0   057D                 215            DC     T(IGNORE),T(IGNORE)
000CE2   057D
000CE4   057D                 215            DC     T(IGNORE),T(IGNORE)
000CE6   057D
000CE8   057D                 215            DC     T(IGNORE),T(IGNORE)
000CEA   057D
000CEC   057D                 215            DC     T(IGNORE),T(IGNORE)
000CEE   057D
000CF0   057D                 215            DC     T(IGNORE),T(IGNORE)
000CF2   057D
000CF4   057D                 215            DC     T(IGNORE),T(IGNORE)
000CF6   057D
         0000 0CF8            216  TABLEH    EQU    *                                PS312160
000CF8                        217            DO     32                               PS312170
000CF8   8020 8021            218            DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2   PS312180
000CFC   8022 8023            218            DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D00   8024 8025            218            DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D04   8026 8027            218            DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D08   8028 8029            218            DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D0C   802A 802B            218            DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D10   802C 802D            218            DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
```

DATA TRANSFER THROUGH AUTO-DRIVER CHANNEL

```
000D14    802E 802F          218         DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D18    8030 8031          218         DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D1C    8032 8033          218         DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D20    8034 8035          218         DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D24    8036 8037          218         DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D28    8038 8039          218         DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D2C    803A 803B          218         DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D30    803C 803D          218         DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D34    803E 803F          218         DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D38    8040 8041          218         DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D3C    8042 8043          218         DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D40    8044 8045          218         DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D44    8046 8047          218         DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D48    8048 8049          218         DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D4C    804A 804B          218         DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D50    804C 804D          218         DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D54    804E 804F          218         DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D58    8050 8051          218         DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D5C    8052 8053          218         DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D60    8054 8055          218         DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D64    8056 8057          218         DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D68    8058 8059          218         DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D6C    805A 805B          218         DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D70    805C 805D          218         DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D74    805E 805F          218         DB     X'80',*-TABLEH+63/2,X'80',*-TABLEH+63/2
000D78                       219         DO     16                                                 PS312190
000D78    057D               220         DC     T(IGNORE),T(IGNORE)                                PS312200
000D7A    057D
000D7C    057D               220         DC     T(IGNORE),T(IGNORE)
000D7E    057D
000D80    057D               220         DC     T(IGNORE),T(IGNORE)
000D82    057D
000D84    057D               220         DC     T(IGNORE),T(IGNORE)
000D86    057D
000D88    057D               220         DC     T(IGNORE),T(IGNORE)
000D8A    057D
000D8C    057D               220         DC     T(IGNORE),T(IGNORE)
000D8E    057D
000D90    057D               220         DC     T(IGNORE),T(IGNORE)
000D92    057D
000D94    057D               220         DC     T(IGNORE),T(IGNORE)
000D96    057D
000D98    057D               220         DC     T(IGNORE),T(IGNORE)
000D9A    057D
000D9C    057D               220         DC     T(IGNORE),T(IGNORE)
000D9E    057D
000DA0    057D               220         DC     T(IGNORE),T(IGNORE)
000DA2    057D
000DA4    057D               220         DC     T(IGNORE),T(IGNORE)
000DA6    057D
000DA8    057D               220         DC     T(IGNORE),T(IGNORE)
000DAA    057D
000DAC    057D               220         DC     T(IGNORE),T(IGNORE)
```

DATA TRANSFER THROUGH AUTO-DRIVER CHANNEL

```
000DAE   057D
000DB0   057D               220          DC     T(IGNORE),T(IGNORE)
000DB2   057D
000DB4   057D               220          DC     T(IGNORE),T(IGNORE)
000DB6   057D
```

```
PASLA PROGRAMMING EXAMPLES FOR 32-BIT PROCESSOR          PAGE  10  16:16:49  11/29/79

LOCAL TERMINAL, FULL-DUPLEX PASLA OPERATION

                               *                                                         PS312220
                               *  THE TERMINAL SHOULD BE INTERFACED THROUGH PASLA(FDX) INTERFACE   PS312230
                               *  THE FOLLOWING EXAMPLE IS FOR 32-BIT PROCESSOR          PS312240
                               *  START EXECUTION @ START2                               PS312250
                               *                                                         PS312260
                               *  READ 10 KEYS & WRITE THOSE 10 CHARACTERS UNDER INTERRUPT CONTROL   PS312270
                               *                                                         PS312280
                               *                                                         PS312290
                               *  REGISTER ASSIGNMENTS                                   PS312300
                               *                                                         PS312310
         0000 0000             DONE     EQU   0               EXAMPLE DONE FLAG          PS312320
         0000 0002             DEV0     EQU   2               PASLA (RECEIVE) ADDRESS    PS312330
         0000 0003             DEV1     EQU   3               PASLA (SEND) ADDRESS       PS312340
         0000 0004             MSG      EQU   4               MESSAGE ADDRESS            PS312350
         0000 0005             STAT     EQU   5               PASLA STATUS               PS312360
         0000 000A             R10      EQU   10              WORK REGISTER              PS312370
         0000 000B             R11      EQU   11              WORK REGISTER              PS312380
         0000 000C             R12      EQU   12              WORK REGISTER              PS312390
         0000 0003             R3       EQU   3               STATUS OF INTERRUPTING DEVICE   PS312400
                               *                                                         PS312410
         0000 0008             BSY      EQU   8                                          PS312420
                               *                                                         PS312430
                               *                                                         PS312440
000DB8   F8A0 0000 00F0        START2   LI    WORK,Y'F0'                                 PS312450
000DBE   954A                           EPSR  MSG,WORK        NO INT, REG SET F          PS312460
                               *                                                         PS312470
                               * SET UP INTERRUPT SERVICE POINTER TABLE                  PS312480
                               *                                                         PS312490
000DC0   E640 0E98                      LDAI  MSG,HALT                                   PS312500
000DC4   E6A0 00D0                      LDAI  R10,X'D0'                                  PS312510
000DC8   24B2                           LIS   R11,2                                      PS312520
000DCA   E6C0 02CE                      LDAI  R12,X'2CE'                                 PS312530
000DCE   404A 0000             EXMP2A   STH   MSG,0(R10)      SET UP ENTIRE TABLE WITH 'HALT'   PS312540
000DD2   C1A0 0DCE                      BXLE  R10,EXMP2A                                 PS312550
                               *                                                         PS312560
000DD6   4820 0E9E                      LH    DEV0,RECADR     GET BOTH ADDRESSES         PS312570
000DDA   4830 0EA0                      LH    DEV1,SNDADR                                PS312580
000DDE   DE20 0EA7                      OC    DEV0,SECOND     SET UP PASLA               PS312590
000DE2   08A2                           LR    R10,DEV0        SET UP 2 TABLE ENTRIES FOR   PS312600
000DE4   0AAA                           AAR   R10,R10         RECEIVE & TRANSMIT DEVICES.   PS312610
000DE6   C8A0 0E1A                      LHI   WORK,RECINT                                PS312620
000DEA   40AA 00D0                      STH   WORK,X'D0'(R10)                            PS312630
000DEE   08A3                           LR    R10,DEV1                                   PS312640
000DF0   0AAA                           AAR   R10,R10                                    PS312650
000DF2   C8A0 0E66                      LHI   WORK,SNDINT                                PS312660
000DF6   40AA 00D0                      STH   WORK,X'D0'(R10)                            PS312670
                               *                                                         PS312680
000DFA   0700                  REPEAT2  XAR   DONE,DONE                                  PS312690
000DFC   E640 0EAA                      LDAI  MSG,MSG33                                  PS312700
000E00   4040 0E9C                      STH   MSG,READING     RESET FLAG                 PS312710
000E04   DE20 0EA2                      OC    DEV0,ENREAD     ENABLE READ SIDE INT, ECHO MODE   PS312720
000E08   DE20 0EA5                      OC    DEV0,RQ2S                                  PS312730
000E0C   982A                           RDR   DEV0,WORK       DUMMY READ TO SET BSY      PS312740
```

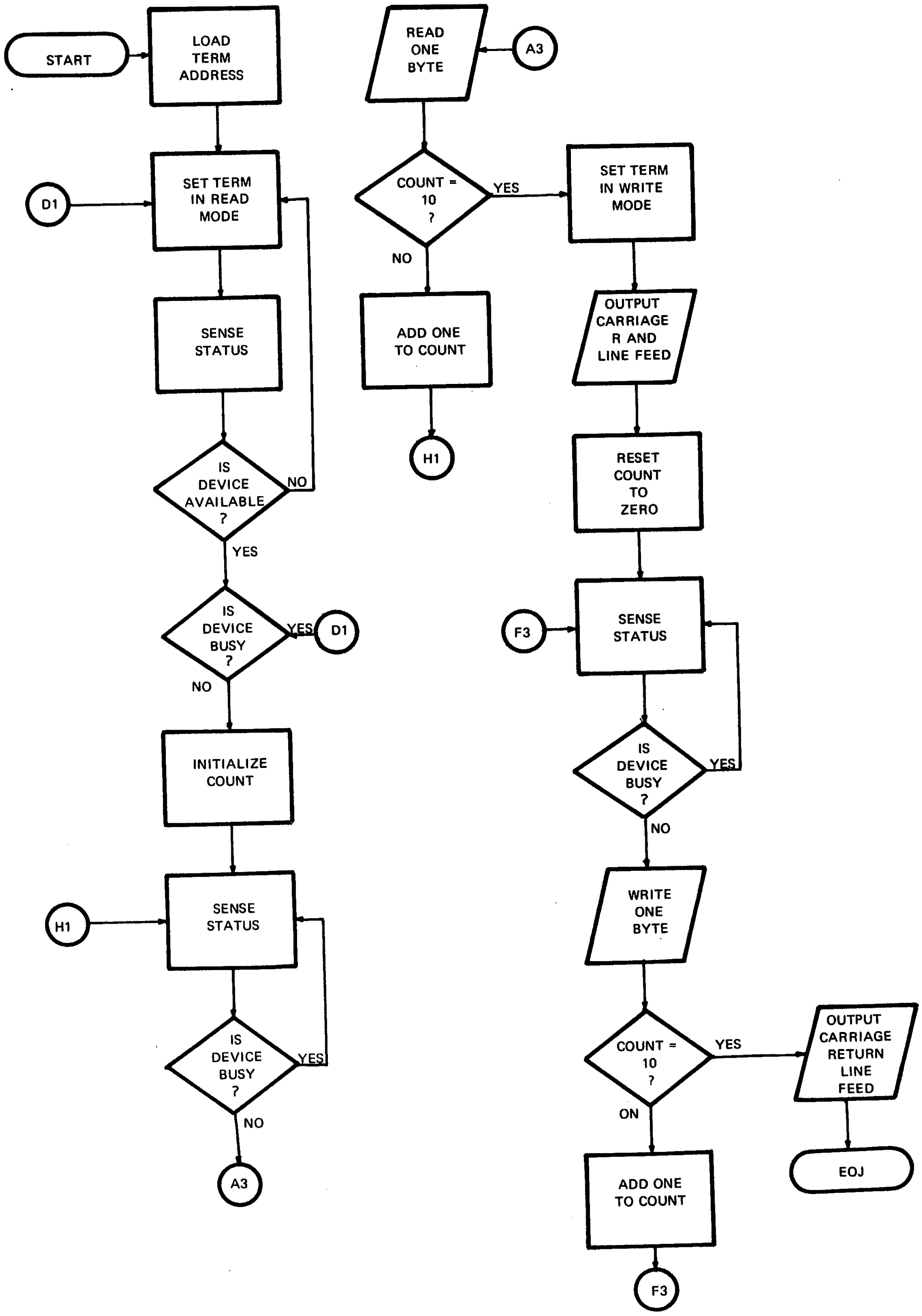
```
000E0E   F8A0 0000 C0F0       275  EXMP2B   LI    WORK,Y'C0F0'     ENABLE INT & HALT TILL DEPRESSION    PS312750
000E14   95AA                 276           EPSR  R10,WORK         OF A KEY INTERRUPTS                  PS312760
000E16   4300 0E16            277           B     *                                                     PS312770
                              278  *                                                                    PS312780
                              279  * RECEIVE SIDE HAS INTERRUPTED                                       PS312790
                              280  *                                                                    PS312800
000E1A   48A0 0E9C            281  RECINT   LH    R10,READING      HALT IF RECEIVE SIDE INTERRUPTS      PS312810
000E1E   4330 0E1E            282           BZ    *                WHEN 'READING' FLAG IS RESET         PS312820
000E22   0833                 283           LR    R3,R3            (R3) = RECEIVE SIDE STATUS           PS312830
000E24   4230 0E24            284           BNZ   *                HALT IF NOT ZERO                     PS312840
000E28   F8A0 0000 00F0       285           LI    WORK,Y'F0'                                            PS312850
000E2E   95AA                 286           EPSR  R10,WORK         REGISTER SET F                       PS312860
000E30   DB24 0000            287           RD    DEV0,0(MSG)      READ BYTE INTO MESSAGE BUFFER        PS312870
000E34   2641                 288           AIS   MSG,1                                                 PS312880
000E36   F540 0000 0EB4       289           CLAI  MSG,MSG33END                                          PS312890
000E3C   4280 0E0E            290           BL    EXMP2B           LOOP TILL 10 KEYS ARE READ IN.       PS312900
                              291  *                                                                    PS312910
                              292  * TO WRITE THE MESSAGE JUST READ IN                                  PS312920
                              293  *                                                                    PS312930
000E40   DE20 0EA3            294           OC    DEV0,DISRD       DISABLE READ SIDE INTERRUPTS         PS312940
000E44   DE20 0EA5            295           OC    DEV0,RQ2S                                             PS312950
000E48   E640 0EA8            296           LDAI  MSG,MSG3                                              PS312960
000E4C   4000 0E9C            297           STH   DONE,READING     RESET FLAG                           PS312970
000E50   DE30 0EA6            298           OC    DEV1,ENWRT       ENABLE WRITE SIDE INTERRUPTS         PS312980
000E54   9D35                 299           SSR   DEV1,STAT                                             PS312990
000E56   2081                 300           BTBS  BSY,1                                                 PS313000
000E58   9A30                 301           WDR   DEV1,DONE        WRITE NULL TO GENERATE FIRST INT.    PS313010
000E5A   F8A0 0000 C0F0       302  EXMP2C   LI    WORK,Y'C0F0'                                          PS313020
000E60   95AA                 303           EPSR  R10,WORK         ENABLE INT @ PROCESSOR LEVEL & HALT  PS313030
000E62   4300 0E62            304           B     *                                                     PS313040
                              305  *                                                                    PS313050
                              306  * TRANSMIT SIDE HAS INTERRUPTED                                      PS313060
                              307  *                                                                    PS313070
000E66   48A0 0E9C            308  SNDINT   LH    R10,READING                                           PS313080
000E6A   4230 0E6A            309           BNZ   *                                                     PS313090
000E6E   0833                 310           LR    R3,R3            STATUS SHOULD BE ZERO                PS313100
000E70   4230 0E70            311           BNZ   *                                                     PS313110
000E74   F8A0 0000 00F0       312           LI    WORK,Y'F0'                                            PS313120
000E7A   95AA                 313           EPSR  R10,WORK         REGISTER SET F                       PS313130
000E7C   0800                 314           LR    DONE,DONE                                             PS313140
000E7E   4230 0DFA            315           BNZ   REPEAT2                                               PS313150
000E82   DA34 0000            316           WD    DEV1,0(MSG)                                           PS313160
000E86   2641                 317           AIS   MSG,1                                                 PS313170
000E88   F540 0000 0EB8       318           CLAI  MSG,MSG3END                                           PS313180
000E8E   4280 0E5A            319           BL    EXMP2C                                                PS313190
000E92   2401                 320           LIS   DONE,1                                                PS313200
000E94   4300 0E5A            321           B     EXMP2C           TO CLEAR LAST WRITE INT              PS313210
                              322  *                                                                    PS313220
000E98   4300 0E98            323  HALT     B     *                                                     PS313230
                              324  *                                                                    PS313240
                              325  *                                                                    PS313250
                              326  * CONSTANTS USED IN ABOVE EXAMPLES                                   PS313260
                              327  *                                                                    PS313270
```

```
      PASLA PROGRAMMING EXAMPLES FOR 32-BIT PROCESSOR          PAGE  12  16:16:49  11/29/79

      LOCAL TERMINAL, FULL-DUPLEX PASLA OPERATION

000E9C   0000                     328 READING   DCX   0            READ FLAG                              PS313280
000E9E   0010                     329 RECADR    DCX   10           PASLA (RECEIVE) ADDRESS                PS313290
000EA0   0011                     330 SNDADR    DCX   11           PASLA (SEND) ADDRESS                   PS313300
000EA2   79                       331 ENREAD    DB    X'79'        ENABLE,ECHO,READ                       PS313310
000EA3   B9                       332 DISRD     DB    X'B9'        DISABLE,ECHO,READ                      PS313320
000EA4   AB                       333 DISWRT    DB    X'AB'        DISABLE,WRITE                          PS313330
000EA5   3B                       334 RQ2S      DB    X'3B'        REQUEST TO SEND (WRT/RD = 1)           PS313340
000EA6   6B                       335 ENWRT     DB    X'6B'        ENABLE,WRITE                           PS313350
000EA7   78                       336 SECOND    DB    X'78'        HIGHER BAUD RATE, 8 BITS/CHAR,         PS313360
                                  337 *                            2 STOP BITS, NO PARITY CHECK           PS313370
000EA8   0D0A                     338 MSG3      DC    X'0D0A'                                             PS313380
000EAA                            339 MSG33     DS    10           10 CHARACTER MESSAGE BUFFER            PS313390
         0000 0EB4                340 MSG33END  EQU   *                                                   PS313400
000EB4   0D0A                     341           DC    X'0D0A'                                             PS313410
000EB6   0000                     342           DCX   0            2 NULL CHARACTERS                      PS313420
         0000 0EB8                343 MSG3END   EQU   *                                                   PS313430
                                  344 *                                                                   PS313440
000EB8                            345           END                                                       PS313450
```

```
    PASLA PROGRAMMING EXAMPLES FOR 32-BIT PROCESSOR          PAGE  13  16:16:49  11/29/79

    LOCAL TERMINAL, FULL-DUPLEX PASLA OPERATION

ASSEMBLED BY CAL 03-066R07-00   (32-BIT)

START OPTIONS:  T=32,ERLST

   NO CAL ERRORS
   NO CAL WARNINGS
    2 PASSES

 ABSTOP     000U 0EB8
 ADC        000U 0004        150     197     197
 BSY        000U 0008        242*    300
 BUF0END    000U 0B4C         45      77     156*
 BUF1END    0000 0B54        138     159*
 BUFULL     000U 0B18        129     133*
 C          000U 0008         90     110     181*
 CCB        000U 0B48         51     114     153*
 CCW        0000 0B48         40      55      79     105     122     154*
 CCWSTA     0000 F700         39      78     176*
 CHKWORD    0000 0B50        157*
 CMD        000U 0BB4         47      81      94     193*
 COMM       0000 0A38         51*     86
 COUNT0     0000 0B4A         44      68      76     155*
 COUNT1     000U 0B52        136     158*
 CR         000U 0AFC        121*    201     213
 CREXIT     0000 0A64         67*    132
 DEV        0000 0007         25*     35      36      37      53      53      61      62      82      83
 DEV0       000U 0002        233*    257     259     260     272     273     274     287     294     295
 DEV1       000U 0003        234*    258     264     298     299     301     316
 DEVADR     0000 0BB0         37      61      83     191*
 DISRD      000U 0EA3        294     332*
 DISWRT     0000 0EA4        333*
 DONE       0000 0000        232*    269     269     297     301     314     314     320
 ENREAD     000U 0EA2         46     272     331*
 ENWRT      0000 0EA6         80     298     335*
 EXMP2A     000U 0DCE        254*    255
 EXMP2B     0000 0E0E        275*    290
 EXMP2C     0000 0E5A        302*    319     321
 G          0000 0002         90     110     129     147     183*
 HALT       0000 0E98        250     323*
 IGNORE     000U 0AFA        118*    200     203     203     208     208     212     215     215     220     220
 IMPTOP     0000 0000I       345
 IODONE     000U 0B3A        139     146*
 ISPTAB     0000 00D0         53     180*
 ISR0       000U 0AAC         56      90*
 ISR01      000U 0AD4         96     102*    115
 ISR1       0000 0AE4         98     110*
 ISR2       000U 0B04        102     128*
 L          0000 0001         90     110     128     146     184*
 LADC       0000 0002
 LINEXIT    000U 0A78         73*    134
 MESSAGE    0000 0B60         42     168*
 MSG        000U 0004        235*    246     250     254     270     271     287     288     289     296     316     317     318
 MSG3       0000 0EA8        296     338*
```

```
      PASLA PROGKAMMING EXAMPLES FOR 32-BIT PROCESSOR          PAGE  14  16:16:49  11/29/79

      LOCAL TERMINAL, FULL-DUPLEX PASLA OPERATION

   MSG33       0000 0EAA        270   339*
   MSG33END    0000 0EB4        289   340*
   MSG3END     0000 0EB8        318   343*
   OK          0000 0A82         72    76*
   PURETOP     0000 0000P       345
   R0          0000 0000         23*   33    42    43    44    46    47    60    67    67    70    74    75
                                 76    80    81   100   106   118   123   141
   R1          0000 0001         24*   32    33    39    40    41    43    45    51    52    53    54    55
                                 56    57    59    60    68    69    70    71    73    75    77    78    79
   R10         0000 000A        237*  251   254   255   260   261   261   263   264   265   265   267   276
                                281   286   303   308   313
   R11         0000 000B        238*  252
   R12         0000 000C        239*  253
   R3          0000 0003        240*  283   283   310   310
   RBUF0END    0000 0BA7         41    73   169*
   RCVADR      0000 0BB2         35   192*
   READ        0000 0BAC         49    85    95   130   133   190*
   READING     0000 0E9C        271   281   297   308   328*
   RECADR      0000 0E9E        257   329*
   RECINT      0000 0E1A        262   281*
   REPEAT2     0000 0DFA        269*  315
   RQ2S        0000 0EA5        273   295   334*
   SECOND      0000 0EA7         93   259   336*
   SNDADR      0000 0EA0         82   258   330*
   SNDINT      0000 0E66        266   308*
   START1      0000 0A00         32*   92   112   147
   START2      0000 0DB8        245*
   STAT        0000 0005        236*  299
   SUBR        0000 0B5C         57    99   103   140   161*
   TABLE       0000 0BF8        204*  206   206
   TABLEH      0000 0CF8        216*  218   218
   TLATE       0000 0002         39   178*
   TLATETAB    0000 0BB8        160   199*
   V           0000 0004         90   110   182*
   WBUF1END    0000 0BAB        137   171*
   WORK        0000 000A         26*   36    48    49    84    84    85    95    97    98    99   102   103
                                104   105   113   114   121   121   122   130   133   135   136   137   138
                                139   140   245   246   262   263   266   267   274   275   276   285   286
                                302   303   312   313
   WRITE       0000 0004         78   177*
   WRTBUF      0000 0B5E         71    74   166*
```

WITH CURRENT LOOP INTERFACE UNDER SENSE STATUS



```
      SAMPLE PROGRAM USING MODEL-550 WITH CURRENT LOOP         PAGE   1  09:23:17  11/29/79

   PROG= CLI32S     ASSEMBLED BY CAL 03-066R07-00   (32-BIT)

                                     1  CLI32S   PROG  SAMPLE PROGRAM USING MODEL-550 WITH CURRENT LOOP
                                     2                 CROSS
                                     3                 TARGT 32
                                     4                 WIDTH 120
                                     5                 NORX3
                                     6  *
                                     7  *   SAMPLE PROGRAM FOR THE MODLE-550 ON A 32-BIT PROCESSOR
                                     8  *   THIS PROGRAM UTILIZES SENSE STATUS LOOPS TO READ/WRITE
                                     9  *   TO THE MODEL-550 WITH A CURRENT LOOP INTERFACE
                                    10  *
                                    11  *
                                    12  *   REGISTER ASSIGNMENTS
                                    13  *
         0000 0001                  14  DU       EQU   1                   DEVICE UNAVAILABLE =1
         0000 0002                  15  DEVADR   EQU   2                   SEI DEVADR TO REGISTER 2
         0000 0003                  16  STATUS   EQU   3                   SEI STATUS TO REGISTER 3
         0000 0004                  17  COUNT    EQU   4                   SEI COUNT TO REGISTER 4
         0000 0005                  18  WORK     EQU   5                   SEI WORK TO REGISTER 5
         0000 0008                  19  BSY      EQU   8                   BUSY = 8
                                    20  *
                                    21  *
000000I  4820 8074 =000078I         22  START    LH    DEVADR,MODADR       DEVADR= MODEL-550 (CLI) ADDRESS
000004I  DE20 8072 =00007AI         23  OC1      OC    DEVADR,RDCMD        CLI IN READ MODE
000008I  9D23                       24  SENS1    SSR   DEVADR,STATUS       STATUS= CLI STATUS
00000AI  4210 FFF6 =000004I         25           BTC   DU,OC1
00000EI  4380 FFF6 =000008I         26           BFC   BSY,SENS1           CHECK NOT BUSY
000012I  2440                       27           LIS   COUNT,0             COUNT=COUNT 10 KEYS
                                    28  *
                                    29  *   READ A TOTAL OF 10 CHARACTERS FROM MODEL-550 INTO MODBUF
                                    30  *
000014I  9D23                       31  SENS2    SSR   DEVADR,STATUS       STATUS= CLI STATUS
000016I  4290 FFFA =000014I         32           BTC   DU+BSY,SENS2        CHECK DU,BUSY
00001AI  9B25                       33           RDR   DEVADR,WORK         WORK= 1 KEY READ FROM OWL
00001CI  C450 007F                  34           NHI   WORK,X'7F'          ZERO OUT PARITY BIT
000020I  D254 8058 =00007CI         35           STB   WORK,MODBUF(COUNT)  STORE BYTE IN MODBUF
000024I  2641                       36           AIS   COUNT,1             ADD 1 TO INDEX COUNT
000026I  C540 000A                  37           CLHI  COUNT,10            ARE THERE 10 BYTES READ YET
00002AI  4230 FFE6 =000014I         38           BNE   SENS2               IF NO BRANCH TO SENS2 TO READ MORE
                                    39  *
                                    40  *   THIS SECTION OF THE PROGRAM OUTPUTS A CR TO THE MODEL-550.
                                    41  *
00002EI  DE20 8049 =00007BI         42           OC    DEVADR,WRTCMD       CLI IN WRITE MODE
000032I  245D                       43           LIS   WORK,13             13 IS A CR
000034I  9A25                       44           WDR   DEVADR,WORK         OUTPUT A CR
000036I  9D23                       45           SSR   DEVADR,STATUS       SENSE STATUS
000038I  2091                       46           BTBS  DU+BSY,1            BRANCH TO SSR COMMAND IF NOT READY
00003AI  245A                       47           LIS   WORK,10             10 IS A LINE FEED
00003CI  9A25                       48           WDR   DEVADR,WORK         OUTPUT A LF
00003EI  9D23                       49           SSR   DEVADR,STATUS       SENSE STATUS
000040I  2091                       50           BTBS  DU+BSY,1            BRANCH TO SSR COMMAND IF NOT READY
                                    51  *
                                    52  *   THIS SECTION OUTPUTS THE 10 BYTES OF
                                    53  *   DATA FROM MODBUF TO THE MODEL-550
```

```
     SAMPLE PROGRAM USING MODEL-550 WITH CURRENT LOOP          PAGE   2  09:23:17  11/29/79

                                 *
000042I  2440                             LIS    COUNT,0               RESET INDEX COUNT TO ZERO
000044I  DA24 8034 =00007CI      WRITE    WD     DEVADR,MODBUF(COUNT)  OUTPUT ONE CHARACTER TO MOD-550
000048I  9D23                             SSR    DEVADR,STATUS         STATUS= CLI STATUS
00004AI  2091                             BTBS   DU+BSY,1              BRANCH TO SSR COMMAND IF NOT READ
00004CI  2641                             AIS    COUNT,1               ADD ONE TO INDEX COUNT
00004EI  C540 000A                        CLHI   COUNT,10              ARE THERE 10 BYTES WRITTEN YET
000052I  4230 FFEE =000044I               BNE    WRITE                 IF NO BRANCH OT WRITE TO WRITE MORE
                                 *
                                 *   THIS ROUTINE OUTPUTS A CR AND LINE FEED TO THE MODEL-550.
                                 *
000056I  DE20 8021 =00007BI               OC     DEVADR,WRTCMD         CLI IN WRITE MODE
00005AI  245D                             LIS    WORK,13               13 IS A CR
00005CI  9A25                             WDR    DEVADR,WORK           OUTPUT A CR
00005EI  9D23                             SSR    DEVADR,STATUS         SENSE STATUS
000060I  2091                             BTBS   DU+BSY,1              BRANCH TO SSR COMMAND IF NOT READY
000062I  245A                             LIS    WORK,10               10 IS A LF
000064I  9A25                             WDR    DEVADR,WORK           OUTPUT A LF
000066I  9D23                             SSR    DEVADR,STATUS         SENSE STATUS
000068I  2091                             BTBS   DU+BSY,1              BRANCH TO SSR COMMAND IF NOT READY
00006AI  C200 8002 =000070I      WAIT     LPSW   WAIT2
                                 *
                                 *   THIS SECTION SETS UP DATA CONSTANTS AND STORAGE AREAS
                                 *
000070I  0000 8000               WAIT2    DCF    Y'8000'
000074I  0000 006AI                       DC     A(WAIT)
000078I  0002                    MODADR   DC     X'0002'               MODEL-550 (CLI) ADDRESS
00007AI  04C8                    RDCMD    DC     X'04C8'               CLI READ COMMAN,DISARM,UNBLOCK
         0000 007BI              WRTCMD   EQU    RDCMD+1               CLI WRITE COMMAND,DISARM
00007CI                          MODBUF   DS     10                    10 BYTE STORAGE FOR MODEL-550 BUFF
000086I                                   END
```

```
    SAMPLE PROGRAM USING MODEL-550 WITH CURRENT LOOP         PAGE   3  09:23:17  11/29/79
ASSEMBLED BY CAL 03-066R07-00   (32-BIT)

START OPTIONS:  T=32,ERLST

   NO CAL ERRORS
   NO CAL WARNINGS
    2 PASSES

 ABSTOP     0000 0000
 ADC        0000 0004
 BSY        0000 0008          19*   26   32   46   50   58   69   73
 COUNT      0000 0004          17*   27   35   36   37   55   56   59   60
 DEVADR     0000 0002          15*   22   23   24   31   33   42   44   45   48   49   56   57
                               65    67   68   71   72
 DU         0000 0001          14*   25   32   46   50   58   69   73
 IMPTOP     0000 0086I         84
 LADC       0000 0002
 MODADR     0000 0078I         22    80*
 MODBUF     0000 007CI         35    56   83*
 OC1        0000 0004I         23*   25
 PURETOP    0000 0000P         84
 RDCMD      0000 007AI         23    81*  82
 SENS1      0000 0008I         24*   26
 SENS2      0000 0014I         31*   32   38
 START      0000 0000I         22*
 STATUS     0000 0003          16*   24   31   45   49   57   68   72
 WAIT       0000 006AI         74*   79
 WAIT2      0000 0070I         74    78*
 WORK       0000 0005          18*   33   34   35   43   44   47   48   66   67   70   71
 WRITE      0000 0044I         56*   61
 WRTCMD     0000 007BI         42    65   82*
```

INDEX

CUT ALONG LINE

PUBLICATION COMMENT FORM

Please use this postage-paid form to make any comments, suggestions, criticisms, etc. concerning this publication.

From ______________________ Date ______________________

Title ______________________ Publication Title ______________________

Company ______________________ Publication Number ______________________

Address ______________________

____

____

FOLD FOLD

Check the appropriate item.

□ Error Page No. ________ Drawing No. ______________

□ Addition Page No. ________ Drawing No. ______________

□ Other Page No. ________ Drawing No. ______________

Explanation:

FOLD FOLI

Fold and Staple
No postage necessary if mailed in U.S.A.

STAPLE STAPLE

FOLD FOLD

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 22 OCEANPORT, N.J.

POSTAGE WILL BE PAID BY ADDRESSEE

PERKIN-ELMER
Computer Systems Division
2 Crescent Place
Oceanport, NJ 07757

TECH PUBLICATIONS DEPT. MS 322A

FOLD FOLD

STAPLE STAPLE