Hazeltine 2000

Video Display Terminal Maintenance Manual

Issued Nov. 1972

Hazeltine Corporation
Computer Peripheral Equipment
Greenlawn, N.Y. 11740 (516) 549-8800

HI-1007-1000

This manual is published and distributed by Hazeltine Corporation, Industrial Products Division. The contents of this manual are subject to change at any time and without prior notice by Hazeltine. The information presented herein may not reflect latest changes in the product. Confirmation and any required clarification of this information can be obtained from your local Hazeltine sales representative.

TABLE OF CONTENTS

SECTION 3. THEORY OF OPERATION (continued)

Page

SECTION 3. THEORY OF OPERATION (continued)

SECTION 4. OPERATION

SECTION 5. MAINTENANCE

Page

SECTION 5. MAINTENANCE (continued)

Page

SECTION 5. MAINTENANCE (continued)

LIST OF TABLES

Table Page

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (continued)

Figure 1-1. Video Display Terminal

Section 1

INTRODUCTION

1.1 GENERAL (figure 1-1)

This manual contains operation and service instructions for the Model 2000 Video Display Terminal (hereinafter called the VDT), manufactured by the Hazeltine Corporation, Industrial Products Division, Greenlawn, New York 11740. As shown in figure 1-1, the VDT consists of a keyboard unit and a display unit, which are electrically interconnected.

1.2 ORGANIZATION OF THIS MANUAL

This manual is divided into the following sections:

a. Section 1 - Introduction. This section describes the purpose of the VDT, and contains quick reference data.

b. Section 2 - Description. This section contains physical and functional descriptions of the VDT.

c. Section 3 - Theory of Operation. This section contains overall and detailed theory of operation of the VDT, supported by block and logic diagrams.

d. Section 4 - Operation. This section contains operator data required by maintenance technicians in servicing the VDT.

e. Section 5 - Maintenance. This section contains checkout and troubleshooting procedures that permit detection and isolation of faults to the plug-in assembly level. Also included are adjustment procedures, removal and replacement instructions, and cleaning instructions.

1.3 PURPOSE OF THE EQUIPMENT

The VDT is designed to replace a standard teletype terminal with a smaller, quieter, and more efficient terminal facility. The paper printout portion of the teletype machine is replaced by a video display. All other remaining features of the standard teletype are retained, and a number of additional features make the VDT much more versatile than a teletype.

The VDT can function as a "stand-alone" terminal with the addition of optional hard-copy printers and/or tape cassette units. The VDT can also function as a highly flexible on-line input/output device in a data processing system. Communication with such systems can be accomplished through direct connection to an I/O controller, or connection to a communications modem that sends and receives data via telephone lines.

Solid-state circuits and plug-in printed circuit boards are used throughout for high reliability and ease of maintenance.

1.4 QUICK-REFERENCE DATA (table 1-1)

Table 1-1 contains a technical summary of the VDT features.

Table 1-1. Technical Description

Characteristic	Description
DISPLAY CHARACTERISTICS	
Character Capacity	27 lines of characters; 74 characters per line; screen capacity - 1998 characters.
CRT Display	12-inch diagonal CRT; Standard raster - 525 lines, 30 frames/sec.; Optional raster - 625 lines, 25 frames/sec.
Character Style	5 x 7 dot matrix pattern using 525-line standard TV raster.
Character Repertoire	64 alphamerics and symbols. 32 ASCII control codes. All 128 ASCII codes can be keyed.
Character Size	Nominal character height - 0.119 inch (6-inch raster height). Nominal character width - 0.082 inch (8½-inch line width).

Table 1-1. Technical Description (continued)

Characteristic	Description
DISPLAY CHARACTERISTICS (continued)	
Split Screen	Two video intensities are available and can be controlled by the CPU. Full intensity is foreground and low intensity is background.
Memory Type	2048 x 8 magnetic core.
Editing Features	15 distinct operations from the keyboard, including character and line insert/delete. 12 distinct operations under CPU control, including backspace and non-stored carriage return.
Cursor Addressability	Program may direct the cursor to any character position on the screen simply by transmitting X-Y coordinates.
Formatting	Protected "background" data may be held on screen while unprotected "foreground" data is transmitted and cleared.
Tab Function	Provides instant cursor advance - horizontally, vertically, or diagonally in fixed format operation.
Status Lights	Five status lights are provided to indicate the operational state of the system.
Refresh Rate	60 fields per second.
COMMUNICATION INTERFACE	
Data Transmission Rates	Standard - 110, 300, 1200, 2400, 9600 baud: Optional - 110, 1200, 2400, 4800, 9600 baud; Optional - 110, 150, 300, 600, 1200 baud.
Data Interface	EIA RS-232-C with: (a) Bell 103A type data set compatible (b) Bell 202C type data set compatible (c) 8X baud rate clock Front panel toggle switch permits channel turn around on EOT code or supervisory channel control in CR position. Data set cable is 10 feet long and fixed to the terminal. A Cinch or Cannon DB-25P connector is used.

Table 1-1. Technical Description (continued)

Characteristic	Description
COMMUNICATION INTERFACE (continued)	
Modes of Operation	Batch, Half-duplex, Full-duplex. A three-position rotary switch on the front panel (under monitor) permits operational mode selection.
Parity	Generates and checks parity. Four-position rotary switch on front panel (under monitor) permits selection of (1) odd parity, (2) even parity, (3) parity bit always "1", and (4) parity bit always "0".
KEYBOARD	
Solid State	The keyboard has no mechanical contacts, and contains three key groups: (1) teletype key arrangement, (2) ten-key adding machine arrangement, and (3) 13-key editing and cursor control arrangement. The keyboard enclosure also contains the following: Indicators: (1) TRANSMIT (2) PRINT Pushbuttons: (1) Power on/off (2) Break (3) System Reset Indicator/pushbutton combinations: (1) Receive mode indicator/set to Receive and Local mode. (2) Local mode indicator/set to Local mode only. (3) Parity error indicator/Reset Parity error.
Keyboard is Removable	An additional power on/off switch is on the front panel of the Display for power turn-on when keyboard is removed.
POWER REQUIREMENTS	
Primary Power	300 volt-amp maximum, 50 hertz or 60 hertz 115/230 volts nominal. Low: 90/180 - 110/220 vac; Medium: 104/208 - 126/250 vac; High: 114/224 - 136/272 vac.
Circuit Protection	DC power supply shuts down automatically for over-voltage, short-circuit, or over-temperature condition. Power supply may be reset by turning primary power off for five seconds.

Table 1-1. Technical Description (continued)

Characteristic	Description
ENVIRONMENTAL	
Temperature	10^{o} to 40^{o}C.
Humidity	90% relative humidity - non-condensing
PHYSICAL CHARACTERISTICS (nominal)	
Dimensions with Keyboard	Height: 12.5 inches (31.8 cm) Width: 18.5 inches (47.0 cm) Depth: 22.0 inches (55.9 cm)
Keyboard Removed	Depth: 16 inches (40.6 cm)
Keyboard Dimensions	Height: 2.8 inches (7.1 cm) Width: 18.5 inches (47.0 cm) Depth: 6.1 inches (15.6 cm)
Weight with Keyboard	62 pounds (28.2 kg)
Electrical Cable/ Connector Interfaces:	
Keyboard to Display	5-foot (1.52 meters) cable from keyboard terminated with 54-pin HDR-series connector (AMP Inc.). Mating connector on display rear panel.
Display to Data Set	10-foot (3.05 meters) cable from display terminated with DB25-P connector (Cannon or Cinch).
Display to Printer	54-pin HDR-series connector (AMP Inc.) on display rear panel.
Display to Tape Cassette	54-pin HDR-series connector (AMP Inc.) on display rear panel.
Display to Remote Monitor	14-pin Series "M" connector (AMP Inc.) on display rear panel.
Display to Power Source	7-foot (1.78 meters) cable, 3-wire, fixed to display (50-Hz equipments have no connector, 60-Hz equipments have molded 3-conductor plugs).

Section 2

DESCRIPTION

2.1 GENERAL

This section contains physical and functional descriptions of the VDT. Refer to Section 3, Theory of Operation, for detailed functional descriptions of the unit.

2.2 PHYSICAL DESCRIPTION

The VDT consists of two separate units -- the keyboard unit and the display unit -- which are electrically interconnected.

2.2.1 KEYBOARD (figure 2-1)

The keyboard unit contains all character and function keys for the VDT. In addition, mode indicators and pushbutton/indicators are provided to control and indicate the status of the VDT. Electronic circuits in the keyboard unit are mounted on a printed circuit board.

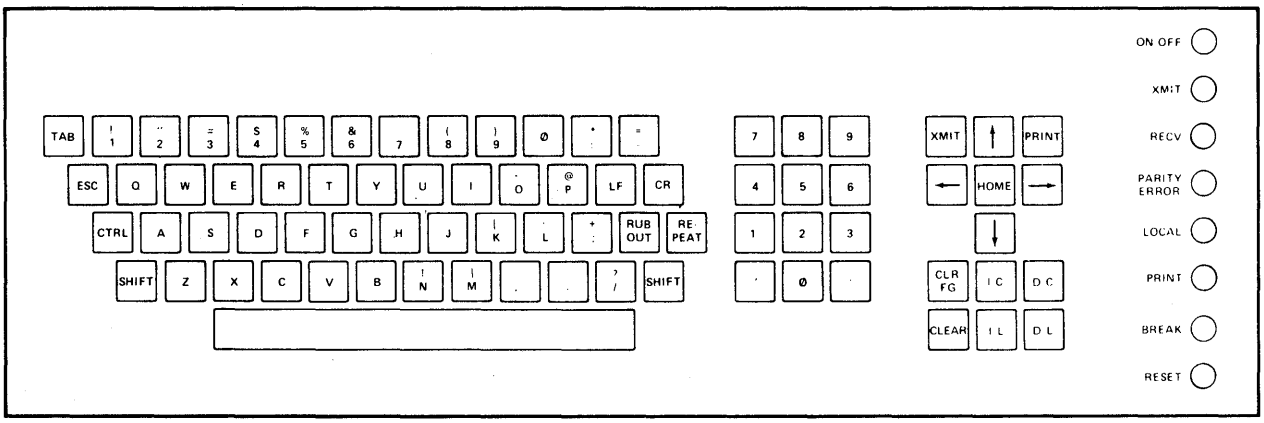


Figure 2-1. Keyboard Unit

2.2.2 DISPLAY UNIT (figures 2-2 and 2-3)

The display unit contains the TV monitor/cage fan assembly, the power supply circuits, the logic rack, the display control panel, and the connector panel. The display control panel is accessible through a tip-in, spring-loaded door in the front of the housing. The connector panel is accessible from the rear. All other assemblies are accessible when the housing is removed.

2.2.2.1 TV Monitor/Cage Fan Assembly. This assembly contains the cathode-ray tube (CRT) and the electrical cage fan that circulates cooling air through the display unit. Adjustments for the TV monitor are contained on the printed circuit board that contains the electronic circuits associated with the CRT. With the exception of the CRT, solid-state electronics are used throughout. The CRT, the printed circuit board, and the cage fan are all mounted to a cube-shaped tubular frame.

2.2.2.2 Logic Rack (figure 2-4). The logic rack is mounted on the right side of the display unit, and contains all electronic circuits with the exception of those in the power supply and the TV monitor. All circuits in the logic rack are contained on 33 plug-in printed circuit boards, plus a two-board memory sandwich that contains the display memory. Off-center keying blades in each printed circuit board slot prevent incorrect insertion of the boards in the slots.

Test points and adjustments on the printed circuit boards are accessible without removing the boards from the rack. Access to internal points on the boards while the unit is in operation is obtained by inserting the board into an extender board, which, in turn, is installed into the logic rack.

By removing all but one of the screws that mount the logic rack to the bottom plate, the rack may be pivoted out, exposing the backplane wiring.

2.2.2.3 Power Supply Circuits. All components of the power supply are mounted directly or indirectly to the bottom plate of the display unit. Most of the voltage regulator circuits are contained on a single printed circuit board which inserts into a connector mounted to a bracket on the left side of the display unit. The bracket also contains test points for each of the dc supply voltages. Separate connectors on the same bracket distribute the dc supply voltages to the TV monitor and to the logic rack components.

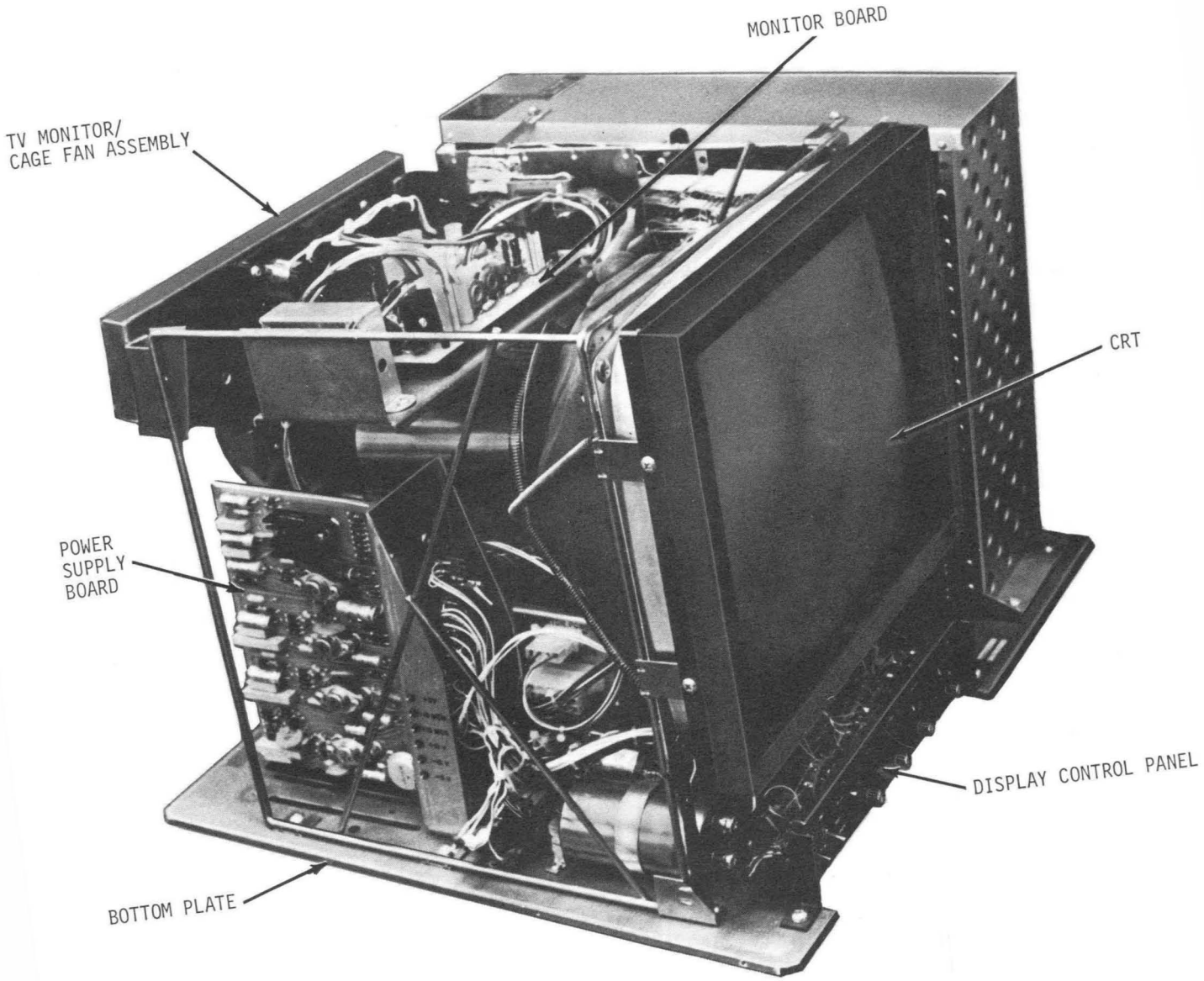


Figure 2-2. Display Unit, Front Left Side View

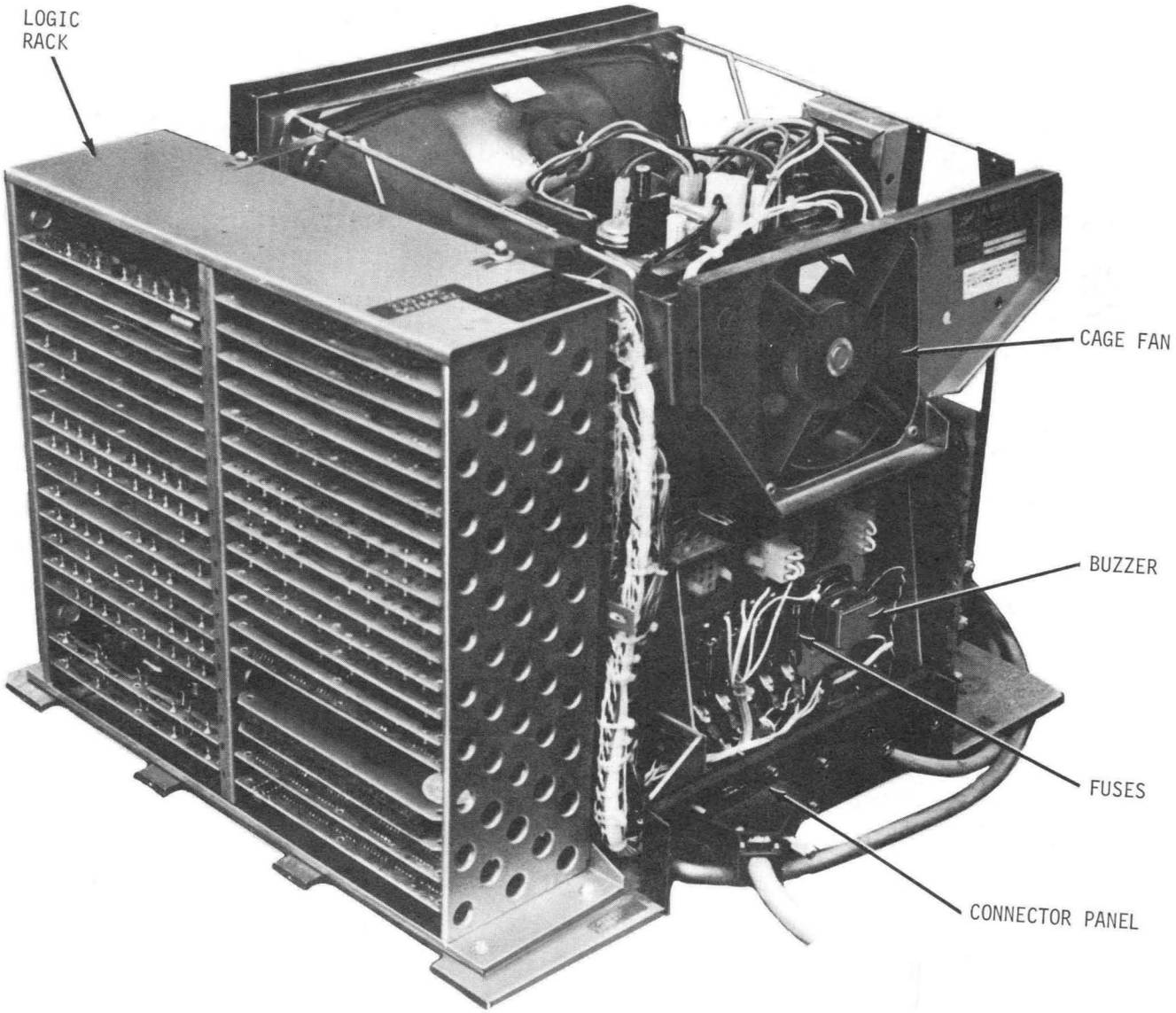
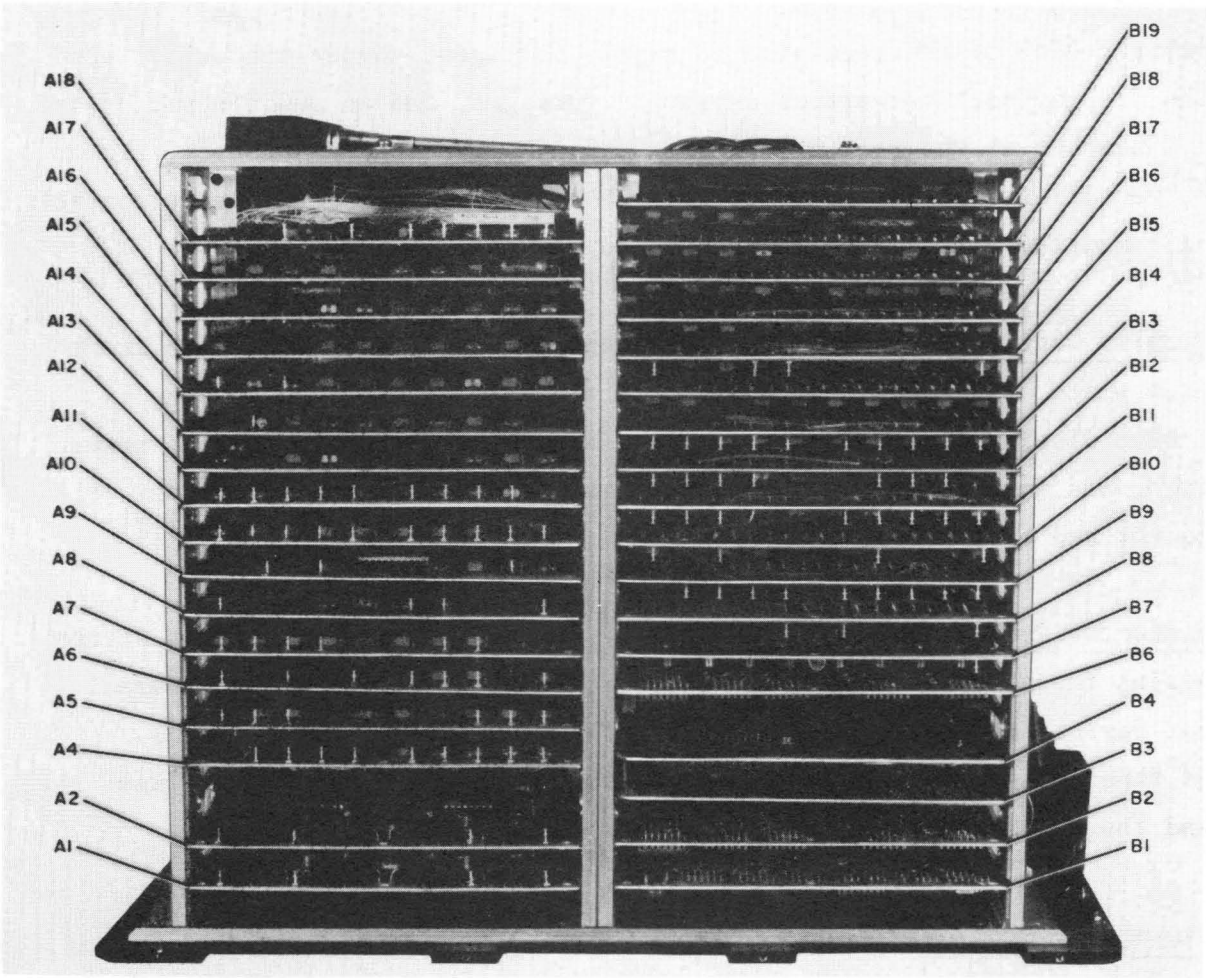


Figure 2-3. Display Unit, Rear Right Side View



ROW A			ROW B
SPARE	19	19	I/O PERIPHERALS CONTROL LOGIC
I/O DATA INTERFACE	18	18	BAUD RATE CLOCK
I/O MODE CONTROL	17	17	I/O PERIPHERAL INTERFACE
I/O DATA CONTROL	16	16	I/O OUTPUT DRIVER
KEYBOARD ENTRY LOGIC	15	15	ROLL UP LOGIC
FULL DUPLEX I/O REGISTER	14	14	I/O READ/WRITE CONTROL
LINE INSERT, CLEAR, PROGRAM COUNTER LOGIC	13	13	60-Hz TV SYNC or 50-Hz TV SYNC
EXPANSION/COMPRESSION	12	12	CURSOR ADDRESS CONTROL LOGIC
PROGRAM COUNTER	11	11	CURSOR ADDRESS Y COUNTER, COINCIDENCE COUNTER
TV SYNC CLOCK	10	10	CURSOR ADDRESS X COUNTER, COINCIDENCE COUNTER
CHARACTER GENERATOR	9	9	REFRESH ADDRESS COUNTER AND REFRESH CONTROL
CHARACTER ROW COUNTER	8	8	MEMORY ADDRESS CONVERTER
MEMORY COMMAND/VIDEO GENERATOR	7	7	DRIVER 3, Y ROW READ/WRITE DRIVERS
LINE COUNTER	6	6	DRIVER 1, Y COLUMN READ/WRITE DRIVERS
DATA REGISTER	5	5	SPARE
MEMORY CONTROL	4	4	CORE MEMORY
SPARE	3	3	
SENSE DIGIT BITS 5-8	2	2	DRIVER 1, X COLUMN READ/WRITE DRIVERS
SENSE DIGIT BITS 1-4	1	1	DRIVER 2, X ROW READ/WRITE DRIVERS

Figure 2-4. Logic Rack

Fuses (five) for each of the regulated dc supply voltages, jumper connectors used to adapt the display unit to various ac supply voltages, and an audible buzzer are mounted to a bracket at the rear of the unit. The main power transformer, filter capacitors, rectifiers, power transistors (with heat sinks), and the power control relay are all mounted directly or indirectly to the bottom plate.

2.2.2.4 Display Control Panel. The display control panel, mounted at the front of the bottom plate, is accessible through the tip-in door when the housing is installed. The display control panel contains a video contrast control, a power on/off switch, and four switches associated with input/output communications between the VDT and external devices.

2.2.2.5 Connector Panel. The connector panel, mounted at the rear of the bottom plate, contains the ac power cable, the data cable that connects to an external input/output device, and cable connectors for the keyboard and the optional printer and tape cassette units. The panel also contains the main power fuse (3½ amp) and the power transformer fuse (1/8 amp).

2.3 FUNCTIONAL DESCRIPTION (figure 2-5)

The VDT can function as a stand-alone system for record production and updating when optional printer and/or tape cassette units are included. Alternatively, the VDT can function as an on-line input/output device for conducting two-way communications with data processing systems. In the latter application, the VDT may be connected directly to the standard teletype channel of the CPU, or may be connected to a communications modem that transmits and receives data from the CPU via ordinary telephone lines.

When the VDT functions as an input/output device in a data processing system, it can operate in any of three modes, depending upon the capabilities of the communications link.

2.3.1 FULL-DUPLEX MODE

In the full-duplex mode, simultaneous two-way communications can be conducted. Characters and function command codes entered via the keyboard are transmitted to the CPU as they are typed. The CPU may simultaneously send alphameric characters

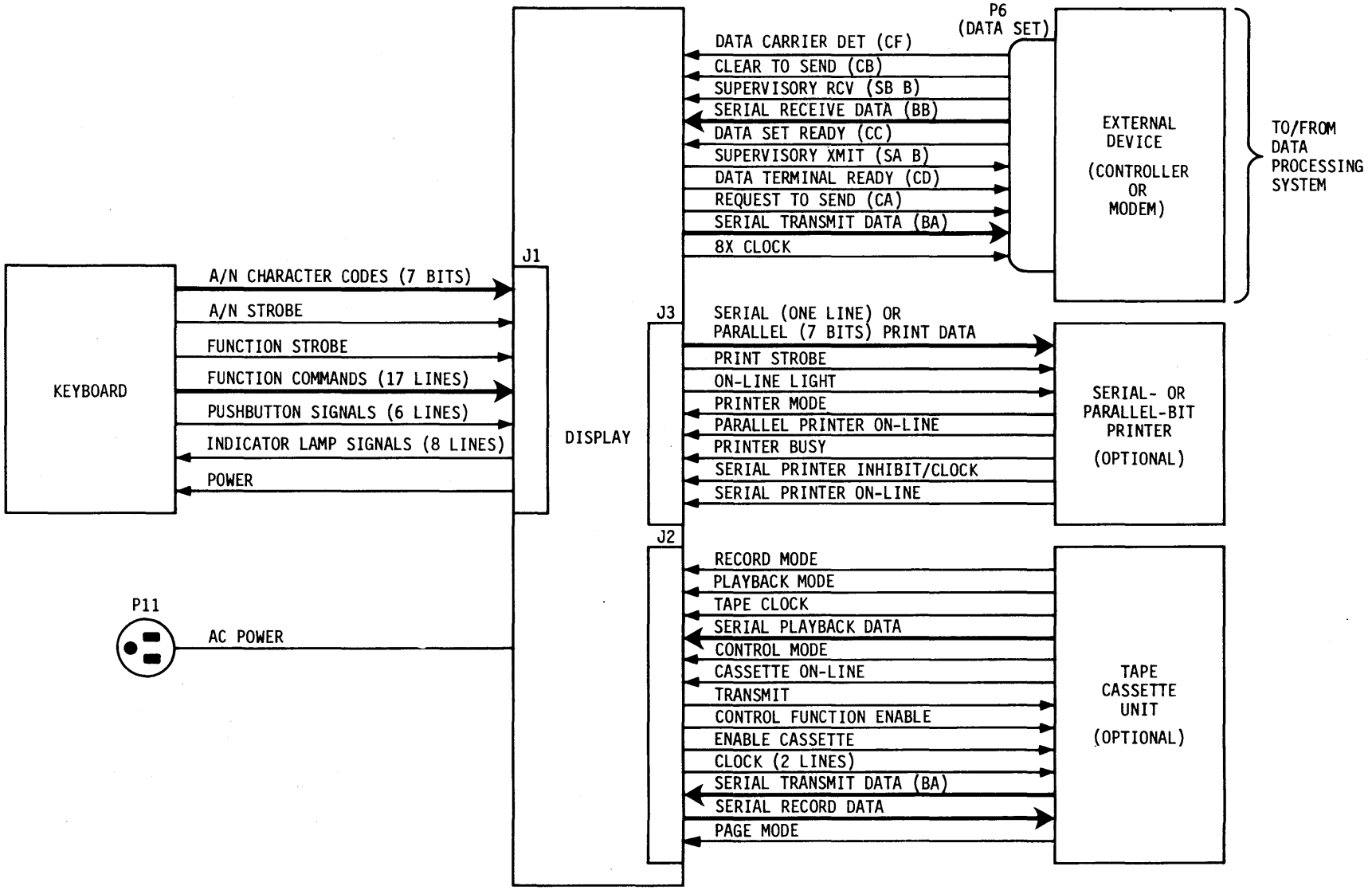


Figure 2-5. System Block Diagram

and function commands to the VDT, producing a display on the CRT. Entries made via the keyboard are not displayed on the CRT. If, however, the CPU "echoes" the typed characters back to the VDT, the effect of displaying the typed characters is achieved.

2.3.2 HALF-DUPLEX MODE

In the half-duplex mode, simultaneous transmission and reception of data cannot be performed. Thus, the VDT is either in a sending or a receiving state. In the sending state, characters and function command codes are entered via the keyboard, stored in memory and displayed, and are also sent to an external device. In the receiving state, characters and function commands received from an external device produce a display on the CRT, and the keyboard is disabled.

2.3.3 BATCH MODE

The batch mode is compatible with input/output devices that operate in either the full- or half-duplex mode. Characters entered via the keyboard are displayed at foreground (bright) intensity on the CRT. Characters received from external devices may be displayed at either foreground or background (dim) intensity. All characters entered via the CPU and the keyboard are stored in a display memory that contains one word for each display location on the CRT. At any point, the keyboard operator may initiate a transmit cycle, during which all foreground data not previously transmitted is retrieved from memory and sent, one character at a time, to the CPU.

2.3.4 OPERATION WITH SERIAL- OR PARALLEL-BIT PRINTER

When the VDT is used with an optional serial- or parallel-bit printer, all data appearing on the CRT may be transmitted to the printer for hard-copy record production. Either the keyboard operator or the data processing system may initiate a print cycle, during which all foreground and background characters not previously printed are retrieved from memory and sent, one character at a time, to the printer.

2.3.5 OPERATION WITH TAPE CASSETTE UNIT

When the VDT is used with the optional tape cassette unit, data entered into the VDT memory from the CPU or the keyboard can be sent to the tape cassette unit for recording on magnetic tape. Data previously stored on magnetic tape may be entered on the CRT in the playback mode, and can also be transmitted (via the VDT) to the CPU. Record and playback clocks may be generated by either the VDT or the tape cassette unit.

2.3.6 CHARACTER GENERATION AND DISPLAY TECHNIQUES

The display contains 27 lines, each containing 74 columns, giving a total of 1998 character positions. The VDT contains a 2048-word core memory (50 spare words) that stores the code for the character displayed in each position. Each word contains eight bits (seven ASCII character code bits and one bit that defines whether the character is to be displayed at foreground or background intensity).

Any of the 1998 positions on the display is defined by an X-Y coordinate set defining the column (X coordinate) and line (Y coordinate). A visible cursor (short horizontal line) can be moved to any position on the display, defining the current character entry point by addressing the corresponding location in the core memory. An ASCII code received from either the keyboard or CPU is entered into the memory location accessed by the cursor. During periodic display refresh cycles, the core memory is scanned sequentially at a rapid rate, and the character codes read from each memory location (non-destructive readout) are decoded by a character generator that produces the dot matrix pattern required to create each character on the screen at the correct X-Y coordinates.

For some function commands, including insert line, delete line, insert character, and delete character, the character codes stored in certain memory locations are moved to other locations as part of the command execution.

2.3.7 DISPLAYABLE SYMBOL REPERTOIRE (table 2-1)

The VDT symbol repertoire consists of 65 displayable symbols, including 64 USASCII (ASCII) characters (no lower-case letters) and a special transmission symbol. These 65 symbols, and their associated ASCII 7-bit binary codes, are listed in table 2-1.

As shown in table 2-1, most characters may be generated by two or more different keying strokes. A keying stroke is here defined as the pressing or striking of one or more of the keys on the keyboard to produce a desired symbol. With the exception of the unlabeled space bar (assigned the abbreviation SP), all keys in table 2-1 are called out by their keytop labels. The superscripts in table 2-1 are used to indicate that the SHIFT or CTRL key should be depressed first, and while still depressed, the appropriate character key should be struck. The superscript definitions are as follows:

^{s}X - press SHIFT key and strike character key

^{c}X - press CTRL key and strike character key

^{cs}X - press both the SHIFT and CTRL keys, and strike the character key.

Bit 7 (MSB) of the ASCII code associated with the basic character key is complemented when the CTRL key is pressed. Bit 5 (third MSB) is complemented when the SHIFT key is pressed. For example, the ASCII code associated with the key having keytop label "A" is 1000001. When the CTRL key is pressed, this code is changed to 0000001, and when the shift key is pressed, the code is changed to 1010001 (Q). If both the CTRL and SHIFT keys are pressed, the code is changed to 0010001.

It can also be seen from an analysis of table 2-1 that, for letters A through Z, [, \ and], an additional keying stroke, not available in the batch mode, can be used to produce the letter in the full- or half-duplex mode. This feature derives from the fact that, in these modes, the ASCII codes for lower-case letters are displayed as upper-case letters (the VDT does not have lower-case letters in its symbol repertoire). In full-duplex mode, computer echo-back is assumed. In half-duplex mode, echo-back is accomplished by an internal VDT connection.

Table 2-1. Symbol Repertoire

Character	ASCII Code*	Keying Strokes	
		Batch Mode	Full- or Half-Duplex Mode
A	1Ø00001	A, ^{s}Q	A, ^{s}Q, ^{cs}1
B	1Ø00010	B, ^{s}R	B, ^{s}R, ^{cs}2
C	1Ø00011	C, ^{s}S	C, ^{s}S, ^{cs}3
D	1Ø00100	D, ^{s}T	D, ^{s}T, ^{cs}4
E	1Ø00101	E, ^{s}U	E, ^{s}U, ^{cs}5
F	1Ø00110	F, ^{s}V	F, ^{s}V, ^{cs}6
G	1Ø00111	G, ^{s}W	G, ^{s}W, ^{cs}7
H	1Ø01000	H, ^{s}X	H, ^{s}X, ^{cs}8
I	1Ø01001	I, ^{s}Y	I, ^{s}Y, ^{cs}9
J	1Ø01010	J, ^{s}Z	J, ^{s}Z, ^{c}LF, ^{cs}:
K	1Ø01011	K	K, ^{cs};, ^{cs}ESC
L	1Ø01100	L	L, ^{c},
M	1Ø01101	M, ^{c}CR	M, ^{c}CR, ^{c}-
N	1Ø01110	N	N, ^{c}.
O	1Ø01111	O	O, ^{c}/, ^{s}RUBOUT
P	1Ø10000	P	P, ^{c}O, ^{cs}SP
Q	1Ø10001	Q, ^{s}A	Q, ^{s}A, ^{c}1
R	1Ø10010	R, ^{s}B	R, ^{s}B, ^{c}2
S	1Ø10011	S, ^{s}C	S, ^{s}C, ^{c}3
T	1Ø10100	T, ^{s}D	T, ^{s}D, ^{c}4
U	1Ø10101	U, ^{s}E	U, ^{s}E, ^{c}5
V	1Ø10110	V, ^{s}F	V, ^{s}F, ^{c}6
W	1Ø10111	W, ^{s}G	W, ^{s}G, ^{c}7
X	1Ø11000	X, ^{s}H	X, ^{s}H, ^{c}8

Table 2-1. Symbol Repertoire (continued)

Character	ASCII Code*	Keying Strokes	
		Batch Mode	Full- or Half-Duplex Mode
Y	1Ø11001	Y, ^{s}I	Y, ^{s}I, $^{c}9$
Z	1Ø11010	Z, ^{s}J, ^{cs}LF	Z, ^{s}J, ^{cs}LF, $^{c}:$
0	0110000	0, ^{s}SP	0, ^{s}SP
1	0110001	1	1
2	0110010	2	2
3	0110011	3	3
4	0110100	4	4
5	0110101	5	5
6	0110110	6	6
7	0110111	7	7
8	0111000	8	8
9	0111001	9	9
:	0111010	:	:
-	0101101	-	-
!	0100001	$^{s}!$	$^{s}!$
"	0100010	$^{s}"$	$^{s}"$
#	0100011	$^{s}\#$	$^{s}\#$
$	0100100	$^{s}\$$	$^{s}\$$
%	0100101	$^{s}\%$	$^{s}\%$
&	0100110	$^{s}\&$	$^{s}\&$
'	0100111	$^{s}'$	$^{s}'$
(	0101000	$^{s}($	$^{s}($
)	0101001	$^{s})$	$^{s})$
*	0101010	$^{s}*$	$^{s}*$

Table 2-1. Symbol Repertoire (continued)

Character	ASCII Code*	Keying Strokes	
		Batch Mode	Full- or Half-Duplex Mode
=	0111101	$s_{=}$	$s_{=}$
←	1011111	$s_{\leftarrow}$	$s_{\leftarrow}$
↑	1011110	$s_{\uparrow}$	$s_{\uparrow}$
@	1000000	$s_{@}$	$s_{@}$
[	1Ø11011	$s_{[}$	$s_{[}$, $^{c}_{;}$, $^{c}_{ESC}$
]	1Ø11101	$s_{]}$	$s_{]}$, $^{cs}_{-}$, $^{cs}_{CR}$
\	1Ø11100	$s_{\backslash}$	$s_{\backslash}$, $^{cs}_{,}$
/	0101111	/	/, $^{cs}_{RUBOUT}$
;	0111011	;	;
+	0101011	s_{+}	s_{+}
,	0101100	,	,
.	0101110	.	.
?	0111111	$s_{?}$, $^{c}_{RUBOUT}$	$s_{?}$, $^{c}_{RUBOUT}$
<	0111100	$s_{<}$	$s_{<}$
>	0111110	$s_{>}$	$s_{>}$
Space	0100000	SP, $^{s}_{0}$	SP, $^{s}_{0}$
■ (Transmit)	1100000	$^{cs}_{0}$, $^{c}_{SP}$	$^{cs}_{0}$, $^{c}_{SP}$

* Ø indicates that bit can be a 1 or a 0. For upper-case letters, the bit is a 0; for lower case letters (displayed as upper case) the bit is a 1. The lower-case key stroke only produces a visible character in full- or half-duplex mode.

2.3.8 FUNCTION COMMAND REPERTOIRE (table 2-2)

As is the case with the symbols listed in table 2-1, the function commands listed in table 2-2 can be generated either by the keyboard or by the CPU via remote commands. Those function commands for which no ASCII code exists cannot be generated remotely.

Of the 30 different function commands listed in table 2-2, 20 can be produced by special function keys on the keyboard. The remaining commands are performed on receiving only, and can be produced at the keyboard in full- or half-duplex mode by pressing either (or both) the CTRL or SHIFT key and striking the appropriate character key. As is the case with symbol characters, two or more keying strokes can produce the same function command in some instances. All commands represented by an ASCII code can be performed in batch mode if the proper code sequence is received by the VDT.

Keying strokes having a single asterisk appended are Class I remote commands, which do not require a lead-in code. Keying strokes having a double asterisk appended are Class II remote commands, which require the command to be preceded by a special lead-in code (1111110). The lead-in code can be entered in half- or full-duplex mode via the keyboard by pressing the CTRL and SHIFT keys and striking the period (.) key. After the lead-in code has been entered, the keying stroke indicated in table 2-2 must be performed to execute the command.

It should be noted that the 7-bit ASCII code format is capable of producing 256 different character and function commands (128 commands without the Class II remote command lead-in, and 128 Class II remote commands preceded by the remote lead-in). Of the 256 possible combinations, only 116 produce any action in the VDT. The remaining 115 code possibilities in the Class II remote category produce no action (null, with no cursor movement). The 25 unused codes for which no lead-in is required cause the VDT to display a space (blank is produced and cursor advances one position).

Table 2-2. Function Commands

No.	Command	ASCII Code	Keying Strokes	
			Batch Mode	Full- or Half-Duplex Mode
1	Cursor-Right	None	→	→
2	Cursor-Left	None	←	←
3	Cursor-Up	None	↑	↑
4	Cursor-Down	None	↓	↓
5	Repeat	None	REPEAT (cursor or character)	REPEAT (cursor only)
6	Tab	None	TAB	TAB
7	Insert Character	None	I/C	None
8	Delete Character	None	D/C	D/C
9	Carriage Return (with erase)	0001101	CR, ^{c}M	CR, ^{c}M*
10	Carriage Return (without erase)	0001101	None	CR**, ^{c}M**
11	Line Feed	0001010	LF, ^{c}J, ^{cs}Z	LF, ^{c}J, ^{cs}Z
12	Rub-Out	1111111	RUBOUT, ^{cs}/	
13	Escape	0011011	ESC, ^{cs}K	ESC, ^{cs}K
14	Null (No Action)	0000000	None	^{cs}P*
15	Sound Buzzer	0000111	None	^{c}G*, ^{cs}W*
16	End of Transmission (line turn-around)***	0000100	None	^{c}D*, ^{cs}T*
17	Backspace Cursor	0001000	None	^{c}H*
18	Lead-In for Remote Command	1111110	None	^{cs}.
19	Set Background	0011001	None	^{c}Y**, ^{cs}I**
20	Set Foreground	0011111	None	^{cs}O**

Table 2-2. Function Commands (continued)

No.	Command	ASCII Code	Keying Strokes	
			Batch Mode	Full- or Half-Duplex Mode
21	Address Cursor****	0010001	None	^{c}Q**, ^{cs}A**
22	Printer On Line	0101111	None	/**, ^{cs}RUBOUT**
23	Printer Off Line	0111111	None	^{s}?**, ^{c}RUBOUT**
24	Insert Line	0011010	I/L	I/L, ^{c}Z**, ^{s}LF**, ^{cs}J**
25	Delete Line	0010011	D/L	D/L, ^{c}S**, ^{cs}C**
26	Transmit	0001110	^{s}XMIT	^{c}N**
27	Print	0011110	^{s}PRINT	^{cs}N**, ^{s}PRINT
28	Home Cursor	0010010	HOME	HOME, ^{c}R**, ^{cs}B**
29	Clear Display	0011100	^{s}CLEAR	^{s}CLEAR, ^{cs}L**
30	Clear Foreground	0011101	^{s}CLR/FG	^{s}CLR/FG, ^{c}CR**, ^{cs}M**

* Remote command that does not require lead-in (Class I).

** Remote command that requires lead-in character (^{cs}.) to be typed before the indicated keying stroke (Class II).

*** In full- or half-duplex mode with CA AUTO selected and EOT/CR switch set to EOT, this command controls the CA signal (Request to Send). Each time an EOT command is issued, signal CA and the RECEIVE indicator change state (line turnaround).

**** Following the Address Cursor command, a two-character cursor address is sent. The first character defines the cursor X-coordinate, and the second character defines the Y-coordinate.

2.3.8.1 Cursor Commands. Commands in this category affect the cursor, and include commands 1 through 4, 6, 9, 10, 17, 21, 22, 23 and 28 in table 2-2.

Commands 1 through 4 permit the cursor to be positioned by means of the associated keyboard editing keys at any point on the display. Movement of the cursor with these keys has no effect on data already displayed. The keys are active in all modes. Since no ASCII code is associated with these keys, the commands cannot be generated remotely.

Command 6 (Tab) permits the cursor to be moved sequentially to preset points defined by transitions from background to foreground characters. When background characters are present, striking the TAB key causes the cursor to be moved to the first foreground character that follows the end of the next background field. When there are no background characters displayed, striking the TAB key causes the cursor to move to the lower right-hand corner of the display.

Command 10 (carriage return without erase) is a Class II remote command that causes the cursor to be moved to the start of the next line. If a character was displayed at the position of the cursor before the command is received, this character is unaffected, unlike the standard carriage return command (command 9) which overwrites the character with the CR code. The CR code is a non-display code; therefore the previous character becomes a blank.

Command 17 (backspace cursor) is a Class I remote command. Each execution of this command causes the cursor to be moved one position leftward on the display without affecting displayed characters.

Command 21 (address cursor) is a Class II remote command that permits the cursor position to be remotely controlled by the CPU. The address cursor command homes the cursor, and must be followed by a two-word address that defines the X-Y coordinates of the desired cursor position. The first address word defines the X-coordinate, and the second word defines the Y-coordinate.

Command 22 (Printer On Line) causes the printer to create a hard copy of data transmitted and received. This command can be performed directly by pressing the parallel printer ON LINE button.

Command 28 (home cursor) causes the cursor to be moved to the home position at the upper left-hand corner of the display.

2.3.8.2 Character Insertion, Deletion, and Repetition Commands. Commands in this category control insertion, deletion, and repetition of characters, and include commands 5, 7, 8, 24 and 25.

Command 5 (repeat) is used in conjunction with any alphameric character key, and with the cursor-up, -down, -left, and -right keys. When used with an alphameric character key, the character is repetitively generated as long as the REPEAT key is pressed. When used with the cursor-up, -down, -left, and -right key., the

cursor moves in the desired direction as long as the REPEAT key is pressed. When the REPEAT and TAB keys are both pressed and an alphameric character key is struck, the entire display is instantly filled with that character. In the full- and half-duplex modes, the REPEAT key only affects the cursor (alphameric characters cannot be repeated). There is no ASCII character code associated with this command, thus it cannot be generated remotely.

Command 7 (insert character) is used to insert foreground characters in the batch mode only. The command has no effect on background characters. Since there is no ASCII code for this command, it cannot be generated remotely. In the batch mode, with the cursor positioned in a foreground field, the I/C key is pressed, and the desired character key is struck. This action will insert the character above the cursor, which remains stationary. All foreground characters to the right and below the cursor that appear before any intervening background field are moved rightward one position each time a character is inserted. Foreground characters beyond the next background field are unaffected.

Command 8 (delete character) is used to delete foreground characters in any mode. The command has no effect on background characters. Since there is no ASCII code for this command, it cannot be generated remotely. With the cursor positioned in a foreground field, the character at the stationary cursor position is deleted each time the D/C key is struck. All foreground characters to the right and below the cursor that appear before any intervening background field are moved leftward one position each time a character is deleted. Foreground characters beyond the next background field are unaffected.

Command 24 (insert line) is used in any mode to insert a line of characters on the line at which the cursor is positioned. The command affects both foreground and background characters. When the I/L key is struck, the cursor is moved to the beginning of the line, and all lines below the cursor are moved down one position. This command can also be generated remotely.

Command 25 (delete line) is used in any mode to delete the line of characters on the line at which the cursor is positioned. The command affects both foreground and background characters. When the D/L key is struck, the cursor is moved to the beginning of the line, the line on which the cursor is located is deleted, and all lines below the cursor are moved up one position. This command can also be generated remotely.

2.3.8.3 Foreground/Background Control Commands. Commands 19 and 20 permit the CPU to remotely control the display intensity of characters on the display. When the batch mode is selected, the keyboard-entered data intensity is foreground; received data intensity is determined by foreground or background remote commands. When the half-duplex mode is selected, the intensity of all data is determined by foreground and background remote commands, regardless of the source of the data. The Class II set foreground command (command 20) causes all characters that follow the command to be displayed at foreground intensity. The Class II set background command (command 19) causes all characters that follow the command to be displayed at background intensity.

2.3.8.4 Transmission Control Commands. Commands 16, 26 and 27 are associated with the transmission of data from the VDT to external devices.

Command 26 (transmit) is used to transmit data to the CPU. Only foreground characters on the display are transmitted. In the batch mode, the command is executed by pressing the SHIFT key and striking the XMIT key. A Class II remote command can be utilized in all modes to transmit data. When command execution is initiated, a single transmit symbol appears at the current cursor position. Then the cursor is moved to the first character position of the line beneath the next preceding single transmit symbol. If there is no intervening transmit symbol, the cursor moves to the top left corner of the display. Starting at this point, the cursor begins moving to the right, and each foreground character in each line is read out. The cursor skips over background fields, which are not transmitted. When the cursor passes the transmit symbol, transmission stops and the cursor remains stationary at the beginning of the line below tne line containing the transmit symbol. If the EOT/CR switch was set to EOT, the EOT code (command 16) is sent to the external device when the EOT symbol is reached on the display. If the EOT/CR switch was in the CR position, the CR code (command 9) is sent.

Command 27 (print) is used to send data to the optional printer. Both foreground and background characters are sent to the printer. This command is executed by pressing the SHIFT key and striking the PRINT key. A Class II remote command can also be utilized to perform the print function. When command execution is initiated, a double transmit symbol appears at the current cursor position. Then the cursor is moved to the first character position of the line beneath the next

preceding double transmit symbol, or, if there are no earlier double transmit symbols, to the top left corner of the display. Starting at this point, the cursor begins moving to the right, and each character in each line is read out. When the cursor passes the double transmit symbol, transmission stops, and the cursor remains stationary at the beginning of the line below the line containing the double EOT symbol. A CR is sent when the first transmit symbol is reached and an LF is sent for the second. CR's and LF's are inserted in the data to insure proper formatting at the printer.

2.3.8.5 Clear Commands. Commands 29 and 30 cause the cursor to be homed, and also cause the display to be completely or partially cleared. Command 29 (clear display) deletes both foreground and background characters. Command 30 only deletes the foreground characters. Command 29 is executed by pressing the SHIFT key and striking the CLEAR key. Command 30 is executed by pressing the SHIFT key and striking the CLR/FG key. Both commands can also be executed by Class II remote command codes.

2.3.8.6 Buzzer Command. Command 15 (sound buzzer) causes the audible buzzer on the VDT to be sounded. This command is in the Class I remote category.

2.3.8.7 Miscellaneous No-Action Commands. In the batch mode, command codes 11 (line feed), 12 (rub-out) and 14 (null) can be entered from the keyboard. The codes are stored in memory, the cursor advances, no character is displayed, and the code is transmitted (in batch transmit). In half- or full-duplex mode, these codes are transmitted when the key is pressed; the codes are not stored and the cursor does not advance. These codes received via the EIA connector or cassette interface will not advance the cursor and the codes are not stored in memory, regardless of mode setting.

Command 13 (escape) advances the cursor and the code is stored in memory when entered from the keyboard or via the EIA connector or tape cassette interface, regardless of mode setting.

THEORY OF OPERATION

INTRODUCTION

This section describes the theory of operation for the Hazeltine 2000 Video Display Terminal. The section is divided into the following parts.

1.	General Description	Capabilities of the terminal.
2.	Communication Modes	Various modes used by the terminal.
3.	System Terminology	Major signal names and function terms.
4.	System Functional Block Description	Operation of the terminal at the system level.
5.	Major Functions of Logic Boards	Listing of the logic printed boards by functional grouping.
6.	System Functional Circuit Discussion	Detailed major function flow between associated logic boards that comprise a functional grouping.
7.	Program Function Flowchart Discussion	Terminal programming functions.

3.1 GENERAL DESCRIPTION

The Hazeltine 2000 Video Display Terminal can operate under three basic communication modes: Full-Duplex, Half-Duplex, or Batch mode. The terminal functions either as a low, medium or high-baud rate machine. The selectable baud rates in each category are: Low (110, 150, 300, 600 and 1200), Medium (110, 300, 1200, 2400 and 9600), High (110, 1200, 2400, 4800 and 9600). Baud rate selection depends upon the speed at which data can be entered into or retrieved from the CPU. The terminal has twelve distinct functions that can be controlled by the operator.

3.2 COMMUNICATION MODES

3.2.1 FULL-DUPLEX MODE

This mode allows for simultaneous two-way data transmission. Data entered on the keyboard is sent to the CPU only and is not stored or displayed. Data received by the terminal via the Input/Output Logic is stored and displayed.

3.2.2 HALF-DUPLEX MODE

In this mode, keyboard-entered data is sent to the CPU via the Input/Output Logic, is stored and displayed.

3.2.3 BATCH MODE

The Batch mode is used when it is desired to enter data from the keyboard directly into memory instead of transmitting it immediately to the CPU. This feature enables the operator to utilize some of the editing functions of the terminal before transmitting to the CPU.

The Batch mode can be used with either Full or Half-Duplex communication systems.

3.3 SYSTEM TERMINOLOGY

Following is a list of the nomenclature of major signals and function terms used in the discussion of the terminal. Refer to EIA standard RS-232 for description of interface signal names that are used with the terminal.

Signal Name or Function Term	Description
A/N	Alpha-Numeric
C/R	Carriage Return
C/W	Clear-Write (of data in memory)
CLR	Clear
COIN	Coincidence Counter
CPU	Central Processing Unit
CTR	Counter
CTRL	Control
CL CAX	Clear Cursor Address Register (the X register, which defines character positions horizontally across the screen)
CL CAY	Clear Cursor Address Register (the Y register, which defines the character rows on lines vertically on the screen)
D/C	Delete Character

Signal Name or Function Term	Description
D/L	Delete Line
ENA	Enable
EOT	End-of-Transmission (indicates end-of-message to the CPU)
ESC	Escape (used to generate a program interrupt signal)
FD	Full-duplex
FG	Foreground (unprotected data in memory) displayed at high intensity on the Terminal display.
$\overline{FG}$	Background (protected data in memory) displayed at low intensity on the Terminal display.
I/C	Insert Character
I/L	Insert Line
I/O	Input/Output Connector (being one data source with the other being the keyboard)
H/D	Half-duplex
INH	Inhibit
KB	Keyboard
LF	Line Feed
P-CTR-1,-2,-3,-4, -5,-6,-7	Program Counter sequence pulses.
R/R	Read/Restore (of data in memory)
REC	Receiver
REG	Register
RPT	Repeat
TOGGLE	Change count by "1"
XMIT	Transmit
XFR	Transfer

Signal Name or Function Term	Description
(+1X) ($\overline{X=73, Y=26}$)	A command which defines the operation of advancing the cursor address X counter and displacing the cursor symbol to the right, once for each character entered, and provided the cursor is not at the last character position in the last row.

3.4 SYSTEM FUNCTIONAL BLOCK DESCRIPTION

The Hazeltine 2000 Video Display Terminal can be broken down into eleven functional sections (figure 3-1): Keyboard Logic, Keyboard Input Logic, Decision Logic, Memory Logic, Refresh Logic, Internal Program Counter, Internal Terminal Clock Sync, Cursor Address Logic, Character Video Logic, Monitor, and Input/Output Logic.

3.4.1 KEYBOARD LOGIC

The Keyboard Logic consists of a keyboard and associated logic circuitry. The keyboard provides the operator with a means to enter alphanumeric data into the terminal. It also allows the operator to enter control signals, such as transmit (XMIT) and editing instructions, such as delete line (D/L). The alphanumeric data and control signals from the keyboard, in the form of ASCII code bits, are applied to the Keyboard Input Logic. Alphanumeric data is applied in parallel form on seven lines. All data entered from the keyboard are initiated by capacitive coupled switches (keys) and do not involve any mechanical contacts. The KEYBOARD CHARACTER BIT signals are also applied to Output Logic in parallel form, where it is processed out as (BA) SERIAL DATA (FD+HD) to the CPU.

3.4.2 KEYBOARD INPUT LOGIC

This functional section generates KEYBOARD FUNCTION COMMANDS in response to inputs from Keyboard Logic. It also controls the flow of data between the Keyboard Logic and the Decision Logic. A REPEAT signal is applied to the Internal Program Counter to initiate repetitive operations when required. An ALLOWED KEYBOARD STROBE (BATCH) signal is applied to the Internal Program Counter to enable generation of counter pulses for character entry in Batch mode. The ALLOWED KEYBOARD STROBE (FD or HD), applied to the Input/Output Logic, enables the Input/Output Logic to transmit data received from the keyboard to the CPU.

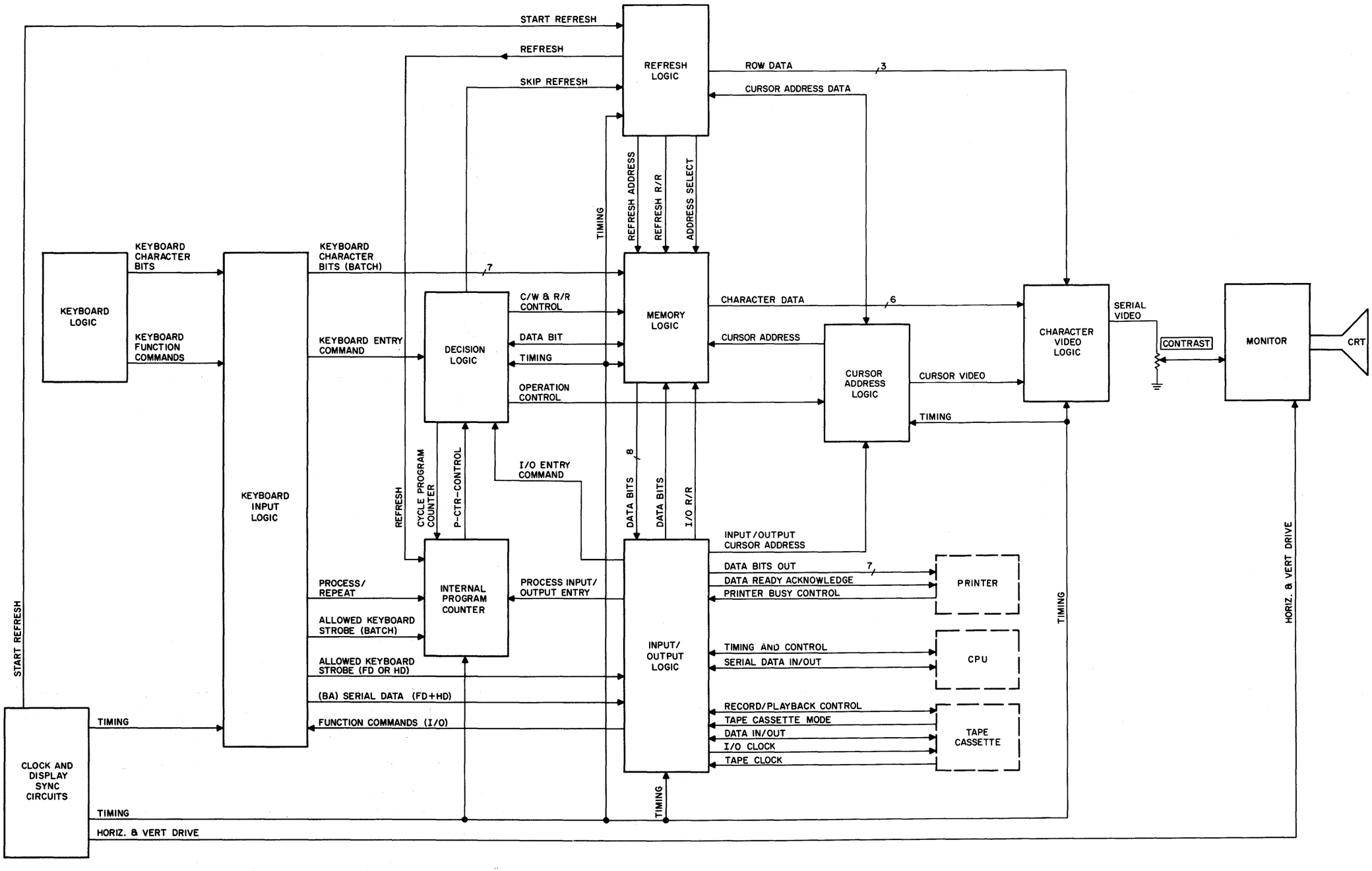


Figure 3-1. Video Display Terminal Functional Block Diagram

3.4.3 DECISION LOGIC

The Decision Logic controls the internal program actions, other than display refresh, of the terminal. These program actions are either initiated by incoming KEYBOARD FUNCTION COMMANDS or I/O FUNCTION COMMAND signals, from Keyboard Input Logic or Input/Output Logic, respectively. The enabling of the program actions are governed by P-CTR-CONTROL signals from the Internal Program Counter. The OPERATION CONTROL signal, that is applied to the Cursor Address Logic, is used to control the entry or routing of cursor position information, for data which is read out of or written into memory. The C/W or R/R CONTROL signals, that are applied to the Memory Logic, are used to control the step-by-step procedure required to read out or write in data into memory. The Decision Logic recycles the Internal Program Counter with a CYCLE PROGRAM COUNTER signal when a program is repetitive in nature. When programs requiring recycling may not be completed during the time of one horizontal retrace, the Decision Logic sends a SKIP REFRESH signal to the Refresh Logic, which causes an interruption of the display refresh cycle, making the memory available for the longer period of time required for completion of the program.

3.4.4 MEMORY LOGIC

The Memory Logic consists of a core memory and associated input and output logic circuits. The core memory has a capacity of 2048 8-bit storage locations. The terminal only utilizes 1998 address locations for character storage. All of these character locations can be displayed on 1998 corresponding positions on the CRT. They are addressed by REFRESH ADDRESS or CURSOR ADDRESS signals from Refresh Logic or Cursor Address Logic, respectively. The REFRESH ADDRESS signal, in conjunction with the R/R CONTROL signal from Decision Logic, enables the Memory Logic to read out of memory a series of 74 characters in sequential order for display per refreshed line. The REFRESH ADDRESS signal is used when the lines are being scanned on the CRT. The resultant refresh signal BIT DATA is sent to the Character Video Logic. When a line is not being scanned on the CRT, the CURSOR ADDRESS signal can be used in conjunction with the C/W CONTROL from the Decision Logic. Upon receipt of the CURSOR ADDRESS and C/W CONTROL signals, DATA BITS from the Input/Output Logic or the Decision Logic are written into memory. If the Memory Logic receives a R/R CONTROL signal instead of a C/W CONTROL, data is read out of memory and sent as DATA BITS to the Decision Logic and Input/Output Logic. A

memory operation of reading out or writing into a memory location requires a time duration of 700 nanoseconds.

3.4.5 REFRESH LOGIC

The function of the Refresh Logic is to determine the exact location on the CRT being scanned at all times. This is accomplished by the presence of the timing signal START REFRESH being sent by the Internal Terminal Clock Sync. When the character lines are being scanned on the CRT, the Memory Logic receives a REFRESH ADDRESS, and also an ADDRESS SELECT signal which causes the Memory Logic to accept the REFRESH ADDRESS. The REFRESH ADDRESS represents the position in memory that contains the data for the character position being scanned. After the acceptance of the REFRESH ADDRESS, a REFRESH R/R signal is sent to the Memory Logic, causing a readout of the character bits at that location. These character bits, CHARACTER DATA, from Memory Logic are sent to the Character Video Logic in conjunction with the ROW DATA signal, which enables the Character Video Logic to determine which character row is being scanned on the CRT at the time of refresh. A REFRESH signal is sent to the Internal Program Counter to indicate that character lines are being scanned on the CRT. Presence of a SKIP REFRESH signal from the Decision Logic, causes the Refresh Logic to interrupt the refresh cycle. Interruption of refresh cycle is needed when a program action is repetitive in nature and cannot be completed within the time required for one horizontal retrace.

3.4.6 INTERNAL PROGRAM COUNTER

The Internal Program Counter generates P-CTR-CONTROL timing counts that are sent to the Decision Logic to be used to time the program acting of the terminal. The Internal Program Counter generates program counts up to 7 stages, where each program count is equivalent to the time (700 nanoseconds) it takes to scan one character position on the CRT. Receipt of an ALLOWED KEYBOARD STROBE (BATCH) from Keyboard Input Logic or PROCESS INPUT/OUTPUT ENTRY from Input/Output Logic starts the program counter. The Decision Logic enables the Program Counter to recycle portions of the program until the program is completed. The REPEAT signal from Keyboard Input Logic causes the program counter to restart 15 times-per-second. This results in the repetition of the program cycle at a 15 cycle-per-second rate.

3.4.7 INTERNAL TERMINAL CLOCK SYNC

The timing signals used throughout the terminal are generated by the Internal Terminal Clock Sync. The timing signals, including horizontal and vertical drive, are derived by counting down a master clock signal of approximately 20 MHz. By supplying all the timing signals for the terminal, the clock ensures that all the program operations performed are synchronized with each other, and are synchronized with the generation of the raster in the Monitor. The master clock signal is also counted down to produce the I/O baud clock rates.

3.4.8 CURSOR ADDRESS LOGIC

The CURSOR ADDRESS signal that is sent to the Memory Logic is used to perform program actions during refresh down time. This means that an I/O character or instruction can be written into or read out of the memory address corresponding to the cursor address. The CURSOR VIDEO signal is mixed with character video to produce the underscore cursor the width of two non-refresh lines (character rows 9 and 10) at the character position corresponding to the cursor address on the CRT. This cursor position on the CRT will be directly under the position where the next operation will take place. This resultant underscore cursor will be represented by a bright video line. The CPU, via Input/Output Logic, can place the cursor at any position on the CRT by sending a cursor address command and the cursor address. This is accomplished by the Input/Output Logic decoding the Input/Output CURSOR ADDRESS command, causing the terminal to load the cursor address into the cursor address register.

3.4.9 CHARACTER VIDEO LOGIC

The Character Video Logic generates a SERIAL VIDEO signal which represents a video dot-by-dot and line-by-line presentation to the Monitor. This SERIAL VIDEO signal forms a character in a five-by-seven dot matrix format within a seven-by-eleven dot window on the CRT. The row element of the character to be presented (simultaneously with the scan of a TV line) will be determined by the ROW DATA signal from the Refresh Logic. Following the five dot positions of each character are two blank dot positions which make up the horizontal space between characters. Beneath the seventh row of dots, which make up each character, are four rows which make up the space between character lines; two rows (9 and 10) are used to display the cursor. The CURSOR VIDEO signal from Cursor Address Logic is timed so as to

cause the ninth and tenth lines to be illuminated when the character position is coincident with the cursor address being scanned. The Character Video Logic can generate up to 65 different characters, including an end-of-transmission (EOT) symbol. When an EOT symbol is displayed, all 35 dots are illuminated. All characters, including the EOT symbol, can be generated at one of two intensity levels. The intensity level indicates whether a symbol is displayed as a background (dim) or foreground (bright) symbol.

3.4.10 MONITOR

The Monitor consists of a CRT, horizontal and vertical deflection circuits, video amplifier circuit, high-voltage power supply circuit, and deflection yoke. The CRT has a 12-inch diagonal screen. The HORIZ and VERT DRIVE signals from the Internal Terminal Clock Sync are used by the Monitor to generate the horizontal and vertical sweep currents for the CRT deflection yoke. The Monitor uses a direct drive scanning system. Horizontal deflection signals are derived from signals from the Internal Terminal Clock. Vertical deflection signals utilize a local monitor vertical oscillator which is synchronized by vertical timing signals from the Internal Terminal Clock. The horizontal deflection circuit and flyback transformer develop the high voltage for the CRT. Cursor and character video are sent to the Monitor by the Character Video Logic.

3.4.11 INPUT/OUTPUT LOGIC

The Input/Output Logic controls the timing of data flow between the terminal and accessory units, such as the CPU, Printer and Tape Cassette. Data from the accessory units must be retimed by the Input/Output Logic for entry into the Memory Logic. Data transmitted from the terminal is retimed by the Input/Output Logic so it can be used by the accessory units. Function command signals from the CPU are decoded by the Input/Output Logic, and sent to the Keyboard Logic as I/O ENTRY COMMAND SIGNALS. Each accessory unit is described separately since some of the control signals associated with the accessory units are unique.

Central Processing Unit (CPU). When transmitting to the CPU in FD or HD modes, parallel data from the Keyboard Input Logic is sent out via Input/Output Logic. In the BATCH mode, parallel data is taken from memory and processed by the Input/Output Logic. The character data received by the Input/Output Logic is temporarily stored and then sent out as serial data at the preselected terminal baud rate.

Printer. Data is sent to the Printer Unit by the terminal, in parallel seven character bits at a time, on seven lines. The Input/Output Logic sends the Printer a PRINTER STROBE signal when the data is ready. The Printer sends back a PRINTER BUSY signal until it is ready for the next set of character bits. The Input/Output Logic waits until the Printer is ready and then sends the next set of character bits along with a PRINTER STROBE signal. If a baud rate higher than 300 is selected, the resulting waiting of the Input/Output Logic will automatically reduce the data throughput rate to 30 characters per second.

Tape Cassette. Operation with the Tape Cassette unit has many modes. Operation is controlled by timing and control signals which flow in both directions between the Input/Output Logic and the Tape Cassette unit. Data to be recorded is sent to the Tape Cassette unit either at the selected terminal baud rate or at a rate determined by a recorder clock signal from the Tape Cassette unit. When DATA IN/OUT signal is sent at the selected baud rate, the Tape Cassette unit reads and stores the data at the baud rate using the I/O CLOCK signal from the Input/Output Logic. The data is then recorded on the tape at a rate of 2400 baud. Data can also be sent to the Tape Cassette unit at a rate determined by the tape cassette clock (the Tape Cassette unit sends the Input/Output Logic a TAPE CLOCK signal). This signal is used by the Input/Output Logic to time the readout of data from the Memory logic. The resultant transfer of data occurs at the 2400 baud rate at which the data is recorded on tape. Playback of data from the Tape Cassette unit occurs at either 2400 baud or at the selected terminal baud rate, depending upon whether the Page or Character mode is selected at the Tape Cassette unit. If the Page mode is selected, data is played back in blocks at 2400 baud, entered into memory, and displayed. If the Tape Cassette unit is operating in the On Line condition, an automatic transmit will be initiated after the playback of each block, and the block will then be read out of the memory and transmitted in a batch at the selected baud rate. If the Tape Cassette unit is also in the Continuous mode, the playback and transmit of subsequent blocks will automatically be initiated until the last block in that file has been transmitted. If the Character mode is selected, the Tape Cassette unit uses the input/output clock to play back data to the terminal at the selected baud rate for direct transmission via the Input/Output Logic. In the Character mode, the playback data will be displayed only in Half-Duplex mode.

3.5 MAJOR FUNCTIONS OF LOGIC BOARDS

The following tables (3-1 through 3-9) list the printed circuit boards of the logic rack by functional groupings, and include the nomenclature, reference designation, and a brief description of the major function of each board.

3.5.1 POWER SUPPLY PRINTED CIRCUIT BOARD

The power supply printed circuit board supplies the necessary voltages needed by the logic boards and the Monitor circuits. The voltages supplied are +15, +13, ±12, and ±5 volts, which are regulated by monolithic voltage regulators.

Table 3-1. Keyboard Input Logic Circuit Boards

Circuit Board	Function
PART OF FULL-DUPLEX I/O REGISTER - A14	Controls the flow of data between the keyboard and the memory. Serializes keyboard character bit data in FD and HD.
KEYBOARD ENTRY LOGIC- A15	Generates keyboard entry commands in response to keyboard inputs.

Table 3-2. Decision Logic Circuit Boards

Circuit Board	Function
DATA REGISTER - A5	Provides temporary storage for data write-in and readout from memory.
LINE COUNTER - A6	Controls the initiation of Internal Program Counter. Controls line deletion and insertion functions.
EXPANSION/COMPRESSION - A12	Decision logic for character insertion and deletion functions. Initiates skip refresh.
LINE INSERT, CLEAR, PROGRAM COUNTER LOGIC - A13	Decision logic for line insertion, full and partial clear. Decision logic for keyboard lockout control and system reset circuits.
PART OF I/O MODE CONTROL - A17	Decision logic for PRINT and batch XMIT functions; bit tracking logic for incoming data (playback or rcv).
CURSOR ADDRESS CONTROL - B12	Generates control signals which position the cursor by incrementing or decrementing the count stored in the cursor address X and Y counter.

Table 3-2. Decision Logic Circuit Boards (cont)

Circuit Board	Function
PART OF ROLL-UP LOGIC - B15	Decision involved in roll-up and PRINT and batch XMIT functions; conditioning for REMOTE COMMAND processing.

Table 3-3. Memory Logic Circuit Boards

Circuit Board	Function
SENSE DIGIT - A1	Sense amplifiers and write inhibit drivers for bits 1 through 4.
SENSE DIGIT - A2	Sense amplifiers and write inhibit drivers for bits 5 through 8.
MEMORY CONTROL - A4	Generates timing and control signals for data write-in and readout from core memory; supplies memory threshold adjustment.
DRIVER NO. 2 - B1	X row drivers.
DRIVER NO. 1 (X COLUMN) - B2	X column drivers.
MEMORY CORE ARRAY - B3 MEMORY CORE ARRAY - B4	Stores character bits (2048 characters x 8 bits).
DRIVER NO. 1 (Y COLUMN) - B6	Y column drivers.
DRIVER NO. 3 - B7	Y row drivers.
MEMORY ADDRESS CONVERTER - B8	Converts 74 by 27 character position format of the raster to the 64 by 32 format of the storage address location format in the core memory.

Table 3-4. Refresh Logic Circuit Boards

Circuit Board	Function
PART OF CHARACTER ROW COUNTER - A8	Generates row data for application to the Character Video Logic.
REFRESH ADDRESS COUNTER AND REFRESH CONTROL - B9	Refresh X and Y address counters.

Table 3-5. Internal Program Counter Circuit Board

Circuit Board	Function
PROGRAM COUNTER - A11	Generates program counts in response to control signals from the Decision Logic, Refresh Logic and Keyboard Input Logic.

Table 3-6. Internal Terminal Clock Sync Circuit Boards

Circuit Board	Function
PART OF CHARACTER ROW COUNTER - A8	Generates horizontal drive signal for Monitor.
TV SYNC CLOCK - A10	Generates an internal master clock signal. Counts down the master clock signal to generate timing signals which synchronize all operations within the terminal with the generation of the TV raster.
TV SYNC 50 Hz/60 Hz- B13	Generates all timing signals equal to or slower than the horizontal drive frequency of 15.75 kHz. Generates vertical sync at 2:1 interlace for Monitor and other circuits.

Table 3-7. Cursor Address Logic Circuit Boards

Circuit Board	Function
CURSOR ADDRESS (X COUNTER AND Y COINCIDENCE COUNTER) - B10	Contains the cursor address X counter which stores the cursor X address. This counter responds to control signals from the Decision Logic. Also contains the Y coincidence counter, which is preset to the Y position, and counts down to a final value to provide a signal for cursor generation.
CURSOR ADDRESS (Y COUNTER AND X COINCIDENCE COUNTER) - B11	Stores the Y address of the cursor. The Y address of the cursor is changed within the vertical limits of the display by control signals from the Decision Logic. Also contains X coincidence counter which is preset to the X position, and counts down to a final value to provide a signal for cursor generation.

Table 3-8. Character Video Logic Circuit Boards

Circuit Boards	Function
MEMORY COMMAND/VIDEO GENERATOR - A7	Controls the write in and read out of data from memory in response to clear/write (c/w) and read/restore (r/r) commands from the Refresh Logic and Decision Logic. Controls background/foreground intensity levels. Mixes character and EOT video and cursor video for TV monitor.
CHARACTER GENERATOR - A9	Decodes ASCII character bits from memory and row data (Row 1-7) from the Refresh Logic and generates character (dot) video (serial).

Table 3-9. Input/Output Logic Circuit Boards

Circuit Boards	Function
PART OF FULL-DUPLEX I/O REGISTER - A14	Controls data output from the terminal in the full-duplex mode.
I/O DATA CONTROL - A16	Converts parallel output data of terminal to serial form for output at the selected baud rate. Converts serial input data to parallel form for use by the terminal. Decodes command signals from the CPU, and generates the corresponding input/output entry commands.
PART OF I/O MODE CONTROL - A17	Input/Output logic for internal clock control in RECEIVE, PRINT and batch XMIT functions.
I/O DATA SET INTERFACE - A18	Detects parity errors of data received and adds the parity bit for output data. Generates the break signal and controls the receive inhibit logic. Generates the Request to Send and Supervisory Transmitted Data Signals.
I/O READ/WRITE CONTROL - B14	Generates control signals for the readout of memory for PRINT and batch XMIT functions. Delays the response to received data until the memory is ready for use by the Decision Logic. Controls the homing of the cursor and positioning the cursor on command from the CPU.
I/O OUTPUT LOGIC - B16	The input/output driver/receivers for the CPU and Tape Cassette unit interface data and clock generation.

Table 3-9. Input/Output Logic Circuit Boards (cont)

Circuit Boards	Function
I/O PERIPHERAL INTERFACE - B17	The input/output logic for the Printer and Tape Cassette interface.
BAUD RATE - B18	Generates internal baud rate clock for receive, PRINT and batch XMIT. Also generates baud rate clock for full and half-duplex transmit.
I/O PERIPHERAL CONTROL LOGIC - B19	The input/output control logic for the Tape Cassette, I/O clock control and Printer control.

3.6 SYSTEM FUNCTIONAL CIRCUITS

This section describes the major logic functions in the Video Display Terminal. Signal flow is traced between related printed circuit cards that contain the logic elements for each function. Those included are:

Keyboard Logic (par. 3.6.1)
Keyboard Input Logic (par. 3.6.2)
Internal Terminal Clock (par. 3.6.3)
Decision Logic (par. 3.6.4)
Internal Program Counter (par. 3.6.5)
Memory Logic (par. 3.6.6)
Refresh Logic (par. 3.6.7)
Input/Output Logic (par. 3.6.8)
Cursor Address Logic (par. 3.6.9)
Character Video Logic (par. 3.6.10)
Monitor (par. 3.6.11)

3.6.1 KEYBOARD LOGIC

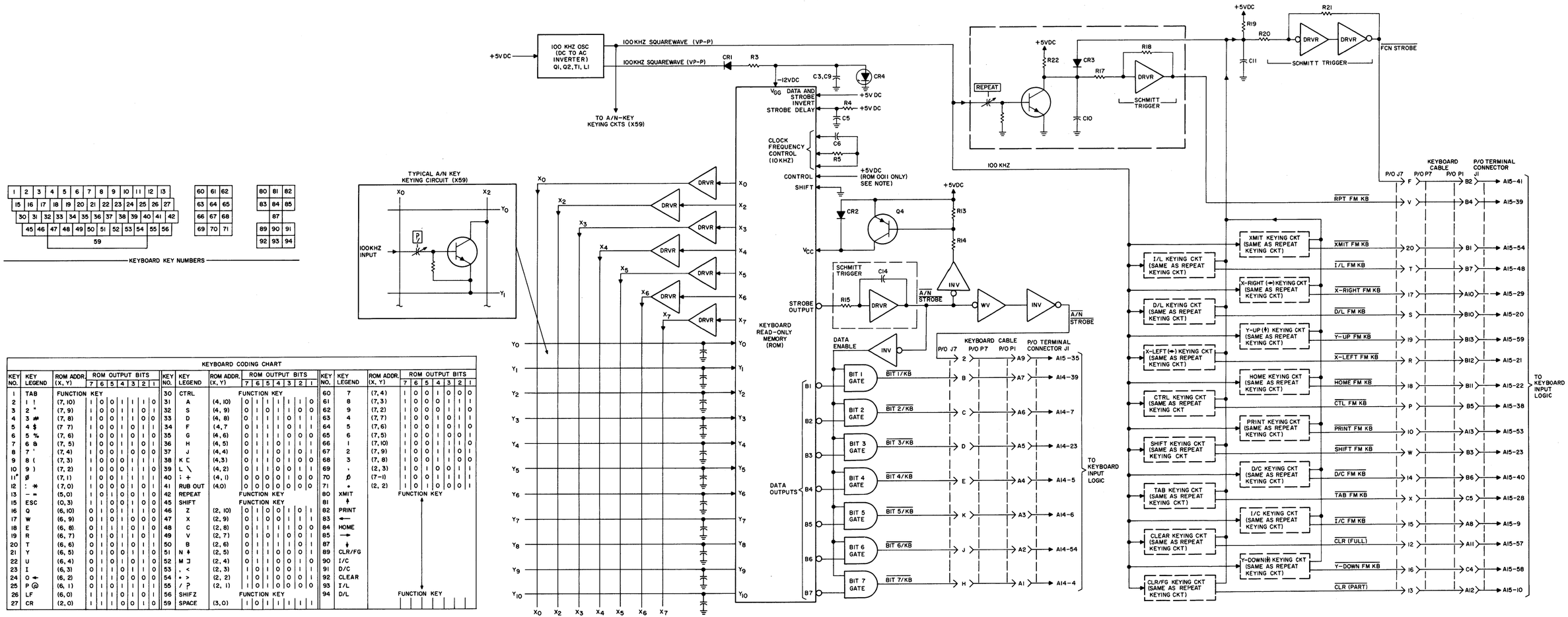
The keyboard logic, comprising the 77 keyboard keys and their associated logic circuitry, provides a means of data entry and control for the operator. Alphanumeric data, in the form of 7-bit ASCII code, are applied to the keyboard input logic of the terminal on seven parallel lines. Control signals, designated function commands, are applied to the keyboard input logic on 18 discrete lines. The individual function commands are initiated by each of the cursor control and editing keys, plus the TAB, CTRL, REPEAT, and SHIFT keys. The remaining 59 keys are

designated alphanumeric (A/N) keys, each of which is assigned a unique 7-bit ASCII code. The circuit operation for both the A/N and function command keys are described in the following paragraphs referenced to figure 3-2.

3.6.1.1 A/N Keys. The circuit for the A/N keys includes a read-only memory (ROM) which, when addressed by one of the 59 A/N keys, produces the complement of the 7-bit ASCII code for that particular key at its B1 through B7 data outputs. The ROM also generates an alphanumeric strobe, $\overline{\text{A/N STRB}}$, after the data outputs have stabilized, to signify that the data is ready and valid. The negative-going strobe is fed through a Schmitt trigger circuit and an inverter to the bit 1 through bit 7 gates at the ROM's data outputs. This gates the 7-bit ASCII code to the keyboard input logic.

Internally, the ROM contain a 2376-bit memory, an oscillator circuit, 8-stage and 11-stage ring counters, and an 11-bit comparator. The 264-word by 9-bit memory is divided into three 88-word by 9-bit groups. The appropriate levels on the shift and control inputs selects one of the three 88-word groups. The 88 individual word locations of the selected group are then addressed by the two ring counters. Thus, the memory address is formed by combining the shift and control inputs with the outputs of the two ring counters. Selecting one of the three 88-word groups of the memory is defined as mode selection. The three modes are normal, shift, and control. Because the keyboard only requires 59 address locations (one for each A/N key), the shift and control inputs are hard-wired to ground or +5 vdc to permanently select one mode of operation. The keyboards for the Hazeltine 2000 are supplied with one of two coded ROM's: ROM 0010 and ROM 0011. As noted in figure 3-2, both the control and shift inputs to ROM 0010 are grounded while the control and shift inputs to ROM 0011 are connected to +5 vdc and ground, respectively. This permanently selects the normal mode for ROM 0010 and the control mode for ROM 0011. However, because the 59 address locations for the normal mode of ROM 0010 and for the control mode of ROM 0011 are precoded with the same data, both configurations produce identical B1 through B7 data outputs for the same A/N keys depressions. In addition, although each word stored in the memory contains nine bits (8-bit ASCII code, plus a parity bit), only bits 1 through 7 (7-bit ASCII code) are applied to the Keyboard Input Logic.

The internal oscillator clocks the 8-stage and 11-stage ring counters to sequentially address each of the 88 (8 by 11) word locations of the memory. The rate



KEYBOARD CODING CHART

KEY NO.	KEY LEGEND	ROM ADDR. (X, Y)	ROM OUTPUT BITS 7	6	5	4	3	2	1
1	TAB	FUNCTION KEY							
2	1 !	(7, 10)	1	0	0	1	1	1	0
3	2 "	(7, 9)	1	0	0	1	1	0	1
4	3 #	(7, 8)	1	0	0	1	1	0	0
5	4 $	(7 7)	1	0	0	1	0	1	1
6	5 %	(7, 6)	1	0	0	1	0	1	0
7	6 &	(7, 5)	1	0	0	1	0	0	1
8	7 '	(7, 4)	1	0	0	1	0	0	0
9	8 (	(7, 3)	1	0	0	0	1	1	1
10	9)	(7, 2)	1	0	0	0	1	1	0
11	Ø	(7, 1)	1	0	0	0	1	0	1
12	: *	(7, 0)	1	0	0	0	1	0	0
13	- =	(5, 0)	1	0	1	0	0	1	0
15	ESC	(0, 3)	1	1	0	0	1	0	0
16	Q	(6, 10)	0	1	0	1	1	1	0
17	W	(6, 9)	0	1	0	1	0	0	0
18	E	(6, 8)	0	1	0	0	0	1	0
19	R	(6, 7)	0	1	0	0	1	0	1
20	T	(6, 6)	0	1	0	0	1	1	0
21	Y	(6, 5)	0	1	0	0	1	1	1
22	U	(6, 4)	0	1	0	1	0	1	0
23	I	(6, 3)	0	1	0	1	0	0	1
24	O ←	(6, 2)	0	1	0	1	0	0	0
25	P @	(6, 1)	0	1	0	1	1	1	1
26	LF	(6, 0)	0	1	0	1	1	0	1
27	CR	(2, 0)	0	1	0	1	1	0	0

KEY NO.	KEY LEGEND	ROM ADDR. (X, Y)	ROM OUTPUT BITS 7	6	5	4	3	2	1
30	CTRL	FUNCTION KEY							
31	A	(4, 10)	0	1	1	1	1	1	0
32	S	(4, 9)	0	1	1	1	1	0	1
33	D	(4, 8)	0	1	1	1	0	1	0
34	F	(4, 7	0	1	1	1	0	0	1
35	G	(4, 6)	0	1	1	1	0	0	0
36	H	(4, 5)	0	1	1	0	1	1	1
37	J	(4, 4)	0	1	1	0	1	0	1
38	K [	(4, 3)	0	1	1	0	1	0	0
39	L \	(4, 2)	0	1	1	0	0	1	1
40	; +	(4, 1)	0	0	0	0	1	0	0
41	RUB OUT	(4,0)	0	0	0	0	0	0	0
42	REPEAT	FUNCTION KEY							
45	SHIFT	FUNCTION KEY							
46	Z	(2, 10)	0	1	0	0	1	0	1
47	X	(2, 9)	0	1	0	0	1	1	1
48	C	(2, 8)	0	1	1	1	1	0	0
49	V	(2, 7)	0	1	0	1	0	0	1
50	B	(2, 6)	0	1	1	1	1	0	1
51	N ↓	(2, 5)	0	1	1	0	0	0	1
52	M ⌴	(2, 4)	0	1	1	0	0	1	0
53	, <	(2, 3)	1	0	1	0	0	1	1
54	. >	(2, 2)	1	0	1	0	0	0	1
55	/ ?	(2, 1)	1	0	1	0	0	0	0
56	SHIFZ	FUNCTION KEY							
59	SPACE	(3,0)	1	0	1	1	1	1	1

KEY NO.	KEY LEGEND	ROM ADDR. (X, Y)	ROM OUTPUT BITS 7	6	5	4	3	2	1
60	7	(7, 4)	1	0	0	1	0	0	0
61	8	(7, 3)	1	0	0	0	1	1	1
62	9	(7, 2)	1	0	0	0	1	1	0
63	4	(7, 7)	1	0	0	0	1	0	1
64	5	(7, 6)	1	0	0	0	1	0	0
65	6	(7, 5)	1	0	0	0	0	1	1
66	1	(7, 10)	1	0	0	0	0	1	0
67	2	(7, 9)	1	0	0	0	0	0	1
68	3	(7, 8)	1	0	0	0	0	0	0
69	.	(2, 3)	1	0	1	0	0	1	1
70	Ø	(7-1)	1	0	0	1	1	1	1
71	+	(2, 2)	1	0	1	0	0	0	1
80	XMIT	FUNCTION KEY							
81	↓								
82	PRINT								
83	←								
84	HOME								
85	→								
87	↓								
89	CLR/FG								
90	I/C								
91	D/C								
92	CLEAR								
93	I/L								
94	D/L								

Figure 3-2. Keyboard Logic Block Diagram

(10 kHz) at which the word locations are addressed is determined by the oscillators external RC network consisting of capacitor C6 and resistor R5. In addition to addressing the memory, the outputs of the 8-stage ring counter are applied to the X_0 through X_7 terminals of the ROM and the outputs of the 11-stage ring counter are fed to the 11-bit comparator. The X_0 and X_2 through X_7 outputs (X, is not used) are applied through drivers to form an X-Y matrix with the Y_0 through Y_{10} ROM inputs. The Y_0 through Y_{10} inputs are applied to the 11-bit comparator where they are compared with the outputs of the 11-stage ring counter. The 59 A/N keying circuits (see insert on figure 3-2) are connected across 59 of the 77 X-Y intersections. A 100-kHz squarewave, developed by a dc to ac inverter, is routed to the capacitive-type switches of each keying circuit. When no A/N key is pressed, the capacitive switches appear as open circuits to the 100-kHz signal and the transistors are cut off. During this condition, the two ring counters are clocked to sequentially address the memory and the strobe output is high. With the strobe output high, the bit 1 through bit 7 gates are inhibited and the $\overline{\text{bit1/KB}}$ through $\overline{\text{bit7/KB}}$ outputs to the keyboard input logic are all low.

When an A/N key is pressed, the 100-kHz square wave is coupled through the respective capacitive switch to the base of the keying circuit transistor. This effectively creates a path between one output of the 8-stage ring counter (X_0 and X_2 through X_7), via the respective driver and keying circuit transistor, and one input of the 11 stage comparator (Y_0 through Y_{10}). Each time the selected X line goes high, the positive-going pulse is passed through the transistor to the selected Y input. After a number of clock cycles, a condition will occur where the level on the selected path to the comparator input (Y input) will match the level on the corresponding comparator input from the 11-stage ring counter. When this occurs, the comparator generates a signal to inhibit the clocks to the ring counters. Thus, the ring counters are stopped at the selected address location and the complement of the 7-bit ASCII code for that particular key appears at the B1 through B7 data outputs. Approximately 1.5 milliseconds after the clocks are inhibited, the strobe output goes low to indicate the data outputs are valid. As previously mentioned, the negative-going strobe is applied through the Schmitt trigger circuit and then inverted to enable the bit 1 through bit 7 gates. The output of the Schmitt trigger is also double inverted (buffered) and applied to the keyboard input logic of the terminal. This negative going $\overline{\text{A/N STRB}}$ pulse signals the keyboard that an A/N entry is being initiated at the keyboard and the 7-bit ASCII code is present on its data input lines. The Schmitt trigger circuit functions to block any noise

or ringing that may occur at the strobe output of the ROM. Once the strobe output drops below the threshold level of the driver, the positive (regenerative) feedback path through capacitor C14 causes the Schmitt output to switch to the logic zero level. Similarly, when the strobe output goes high again, the Schmitt output switches to the logic one state when the input level reaches the threshold level of the driver.

When the A/N key is released, the capacitive switch removes the 100-kHz signal from the keying circuit transistor to open the path between the ROM X output and Y input. When this occurs, the internal ring counters are clocked again to sequentially address the memory and the strobe output goes high to inhibit the bit 1 through bit 7 output gates.

As shown in figure 3-2, the $\overline{\text{A/N STROBE}}$ signal at the output of the Schmitt trigger circuit is also applied through an inverter to the base circuit of transistor Q4. Transistor Q4 acts as a switch which is turned on or off by its base signal to increase or decrease the voltage at the Vcc input of the ROM. This functions to further stabilize the B1 through B7 data outputs by decreasing the Vcc supply voltage when the strobe pulse is generated. When the strobe output is high, the inverter output is low, and transistor Q4 is turned on. With Q4 conducting, the Vcc input is approximately 4.9 VDC due to the small voltage drop across the transistor. However, when the strobe output goes low, the inverter output goes high to cut the transistor off. The Vcc input then drops to about 4.2 VDC due to the approximate 0.8 VDC drop across diode CR2. This decrease in Vcc voltage helps to pull down the appropriate B1 through B7 outputs. Erroneous data cannot be generated even if an A/N key is held in a partially depressed state.

The keyboard coding chart in figure 3-2 lists each of the keys on the keyboard, along with the key legends, and the ROM X-Y addresses and B1 through B7 data outputs for each of the 59 A/N keys.

3.6.1.2 Function Command Keys. Because all 17 function command keying circuits are identical, only the keying circuit for the REPEAT function command key will be described. When the REPEAT key is not pressed, the keying circuit transistor is cut off and +5 vdc is applied althrough collector resistor R22 to the input of the Schmitt trigger circuit. Thus, the output of the Schmitt trigger ($\overline{\text{RPT RM KB}}$) is normally high. When the key is pressed, the 100-kHz squarewave from the dc to ac inverter is coupled through the REPEAT CAPACITIVE switch to the base of the transistor. This turns the transistor on for the positive half cycles of the

input signal. Thus, capacitor C10, which was previously charged up to +5 vdc, discharges through the transistor during the positive half cycles and charges through resistor R22 during the negative half cycles. Because the RC time constant of the discharge path is relatively small as compared to the charging path, the charge on the capacitor decreases to about zero volts in a few cycles. However, when the voltage level at the input to resistor R17 drops below approximately 0.8 volt, the Schmitt trigger output switches to the logic zero state. Thus a $\overline{\text{RPT FM KB}}$ function command is applied to the keyboard input logic of the terminal. When the REPEAT key is released, the capacitive switch opens to remove the 100-kHz signal from the base of the keying circuit transistor. Thus, the transistor is turned off and capacitor C10 begins to charge toward +5 vdc again. When the charge on C10 reaches about 2 volts, the Schmitt output switches to the logic one state to terminate the $\overline{\text{RPT FM KB}}$ signal.

The keyboard logic also generates a delayed negative-going function strobe, $\overline{\text{FCN STROBE}}$, to signal the keyboard input logic when a function command is initiated at the keyboard. The collectors of the 17 keying circuit transistors are ORed together, via their respective diodes (CR3), and applied to the junction of capacitor C11 and resistor R19 at the input to a Schmitt trigger circuit. When any one of the transistors is turned on, capacitor C11 starts to discharge through the respective diode and transistor. When the charge on the capacitor drops below the threshold level of the Schmitt, the $\overline{\text{FCN STROBE}}$ output goes low. The delay of the function strobe is obtained as a result of the strobe Schmitt trigger level being lower than that of the Schmitt for the function outputs, and that the inputs to both Schmitts decrease simultaneously and relatively slowly (> 1MS) as a key is depressed. When the function command key is released, the transistor turns off and capacitor C11 begins to charge toward +5 volts again. At about 2 volts, the Schmitt output goes high to terminate the $\overline{\text{FCN STROBE}}$ signal.

3.6.2 KEYBOARD INPUT LOGIC (See figure 3-3).

The Keyboard Input Logic controls the flow of character bit data between the keyboard and the memory and display circuits, and gener-tes function commands in accordance with keyboard-selected function inputs. It also controls the flow of data in Half or Full-Duplex modes of transmission. In the Batch mode, the data is stored in memory for possible editing prior to transmission.

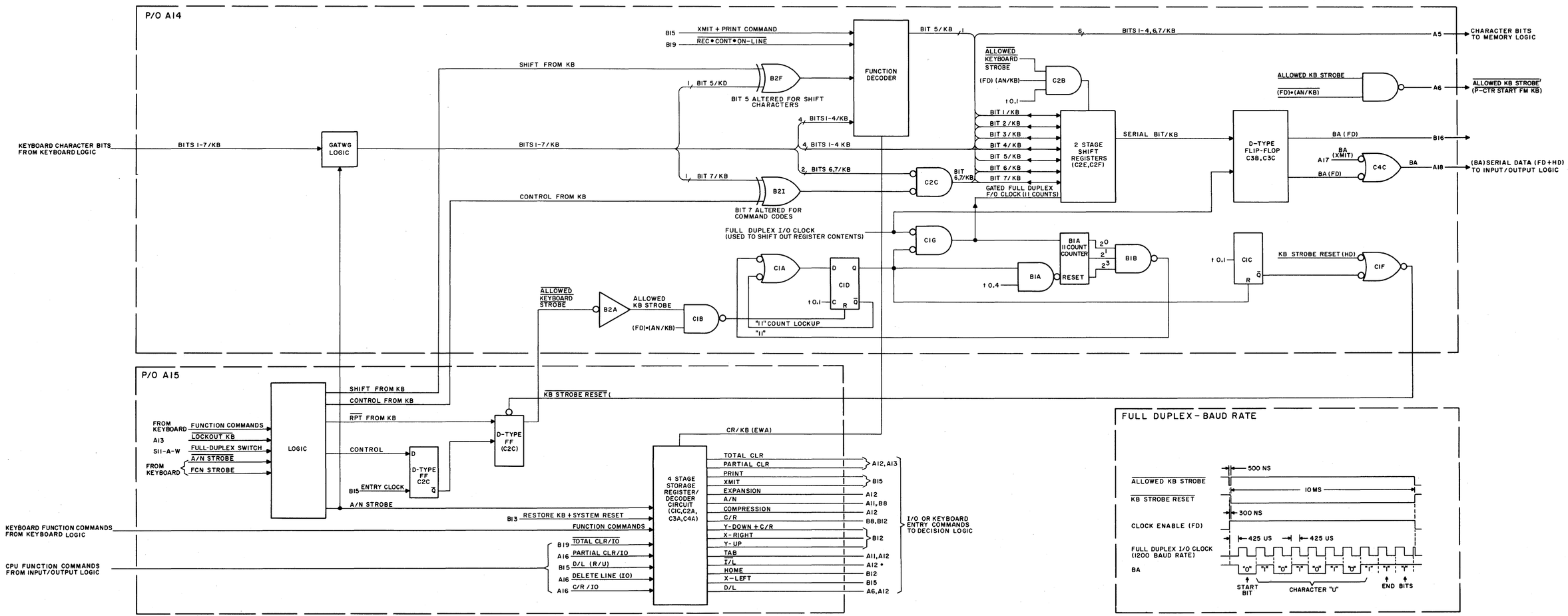


Figure 3-3. Keyboard Input Logic Block Diagram

3.6.2.1 Batch Mode. The Keyboard Input Logic samples the incoming $\overline{\text{BITS 1-7/KB}}$ data from the Keyboard Logic and routes it to data register A5, where it will be further routed to the memory and display circuits for storage and display. The Function Decoder samples certain simultaneous conditions such as shift and transmit or shift and print. These combinations initiate a transmission or a print operation depending upon the function selected. Should a carriage return be initiated from the keyboard, the Function Decoder develops CR/KB (ENA) which is applied as an input to the 4-stage Storage Register/Decoder circuit. A low level $\overline{\text{STROBE KB REG}}$ signal permits the transfer of CR/KB (ENA) or any other selected function command to the Decision Logic. The 4-stage Storage Register/Decoder circuit stores the selected function commands until the terminal is ready to perform that particular function. As an example, if the function to be performed is a delete line (D/L), the terminal will sample the function command and, under control of the Internal Program Counter, perform the delete line operation.

3.6.2.2 Full-Duplex or Half-Duplex Modes. The keyboard character bits are strobed into the 2-stage Shift Registers and are shifted out serially through two additional D-type flip-flops by the FULL DUPLEX I/O CLOCK signal when enabled by CLOCK ENABLE (FD) signal. The serial data (BA) is routed through the Input/Output Logic for parity processing prior to transmission to the CPU. The CLOCK ENABLE (FD) signal is generated by a D-type flip-flop when reset by the coincidence of the (FD) · (AN/KB) and $\overline{\text{ALLOWED KB STROBE}}$ signals. The CLOCK ENABLE (FD) signal enables the FULL-DUPLEX I/O CLOCK signal to shift the data out serially at the baud rate selected by the front panel BAUD switch. The CLOCK ENABLE (FD) signal is disabled after a period of eleven counts or 10 clock pulses. Additional counts are required to provide start, stop and parity bits along with the seven data bits, totaling ten bits, for proper serial transmission.

3.6.3 CLOCK AND DISPLAY SYNC CIRCUITS

The clock and display sync circuits generate the basic clock and timing signals for the digital processor portion of the terminal, as well as the horizontal and vertical drive (display sync) signals for the terminal monitor circuits. All timing signals, including the horizontal and vertical drive signals, are derived from an internally generated 19.845 MHz master clock. Thus, all terminal

operations are synchronized with the display raster. A functional description of the clock and display sync circuits is provided in the following paragraphs referenced to figure 3-4.

3.6.3.1 Master Clock and t-Pulse Generator. When the terminal is energized, +5 vdc is applied to a crystal-controlled oscillator which, in turn, generates the 19.845 MHz master clock. A divide-by-two flip-flop then divides the 19.845 MHz clock (figure 3-5) down to an approximate 10 MHz (9.9225 MHz) for the reset circuit and t-pulse generator. A 10 MHz shift clock output is also routed to the character video logic circuits where it is used to shift the character video pulses to the monitor for display on the crt screen.

The t-pulse generator is a 7-bit shift register which generates seven individual pulse to.0 through to.6), each consisting of 100 nanosecond pulses, spaced 700 nanoseconds apart (leading edge to leading edge). The shift register mode (reset or shift right) is controlled by the reset circuit. When power is initially applied to the terminal, a capacitor at the input to the reset circuit begins to charge from 0 vdc toward +5 vdc. For approximately 100 milliseconds after power is applied, the voltage on the capacitor is sufficiently low to appear as a logic zero input to the reset circuit. This causes the register reset output to the shift register to be high. With the mode input to the shift register high, shift right operation is inhibited and parallel entry is enabled. Thus, the high register reset signal at the to.0 parallel input causes the t0.0 output to go high and the grounded to.1 through to.6 parallel inputs force the to.1 through to.6 parallel outputs low. During this initial reset period the reset circuit also generates SYSTEM RESET and $\overline{\text{SYSTEM RESET}}$ signals for the Input/Output Logic, Decision Logic and Memory Logic circuits. When the charge on the capacitor reaches approximately 2 volts, the input to the reset circuit appears as a logic one. This terminates the SYSTEM RESET and $\overline{\text{SYSTEM RESET}}$ signals and on the next positive-going edge of the 10 MHz input the register reset output goes low. When the shift register mode input is low, parallel entry is inhibited and shift right operation is enabled. The high at the t0.0 output is then shifted one bit to the right on each negative-going edge of the 10 MHz clock input. Since the t0.6 output is fed back to the shift register serial input, the logic one bit is continuously re-circulated to produce the t0.0 through t0.6 pulse trains shown in figure 3-5. The t0.0 through to.6 and $\overline{\text{t0.0}}$ through $\overline{\text{t0.6}}$ outputs of the clock and display sync circuits are distributed

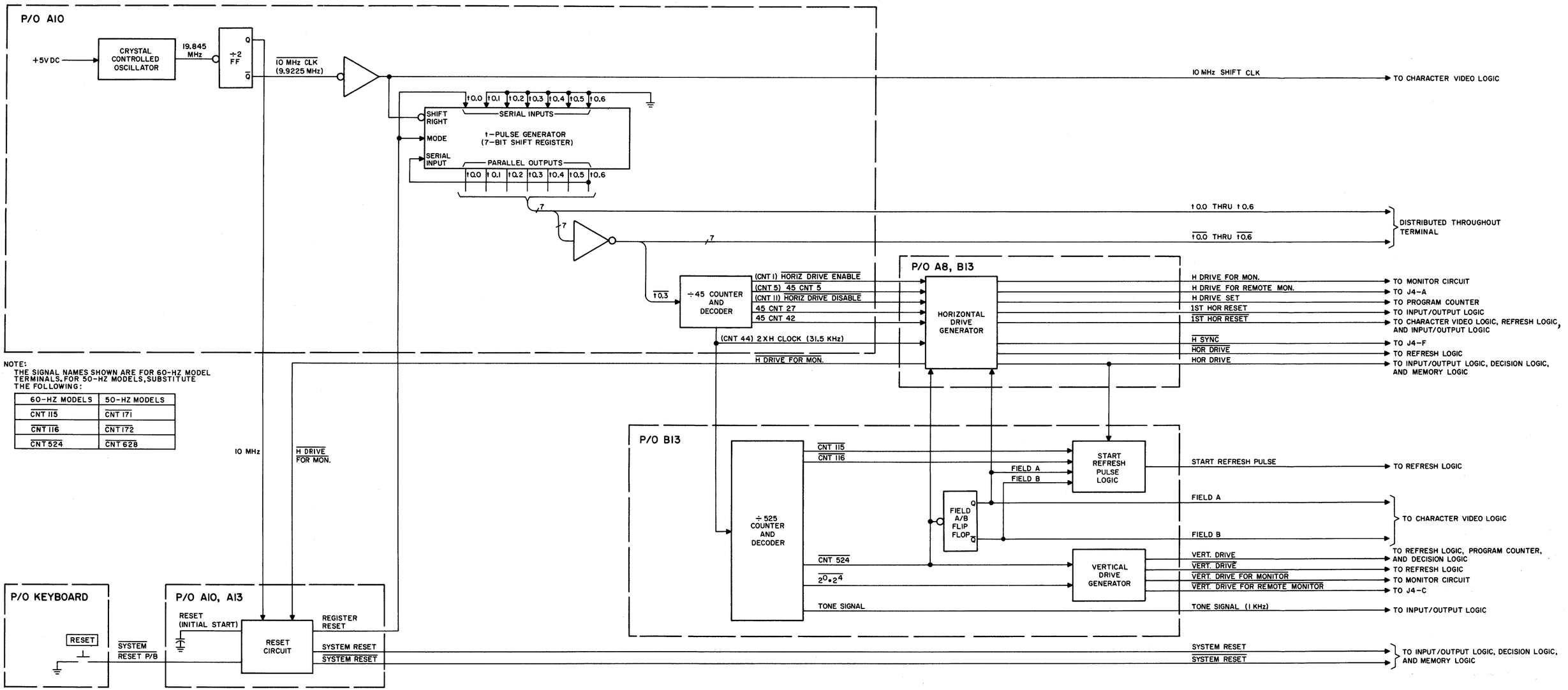


Figure 3-4. Clock and Display Sync, Functional Block Diagram

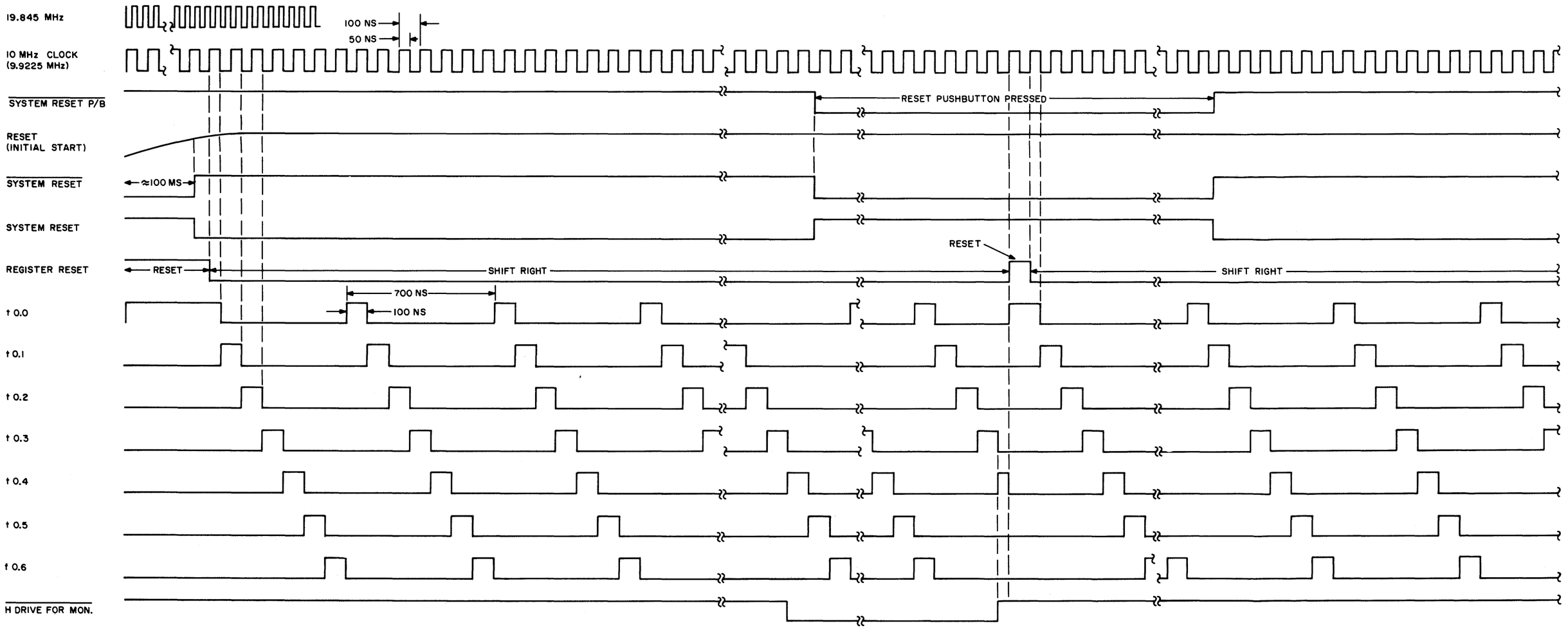


Figure 3-5. Master Clock and t-Pulse Generator, Timing Diagram

throughout the terminal where they are used to control the timing of various terminal operations. The $\overline{\text{to.3}}$ pulse train is also applied to the divide by 45 counter and decoder.

In addition to the initial start reset discussed above, the keyboard is provided with a RESET pushbutton for manual reset. When the RESET pushbutton is pressed, a low level (ground) SYSTEM RESET P/B signal is applied to the reset circuit. This causes the SYSTEM RESET output to go high and the SYSTEM RESET output to go low. As shown in figure 3-5, these signals remain in the reset state until the RESET pushbutton is released. However, the register reset signal goes high on the leading edge of a 10 MHz clock pulse and goes low again on the leading edge of the next 10 MHz clock pulse. Thus, the t-pulse generator is held reset for only 100 nanoseconds. Since the to.3 pulse train is used to derive the horizontal drive pulses that are required by the monitor circuits to develop the dc voltages for the crt, limiting the t-pulse generator reset time to 100 nanoseconds prevents the crt voltages from being affected when the RESET pushbutton is held pressed. In addition, to further prevent the reset function from affecting the monitor circuits, the register reset pulse is inhibited when the $\overline{\text{H DRIVE FOR MON}}$ input to the reset circuit is low. Thus, if the $\overline{\text{H DRIVE FOR MON}}$ input is low when the RESET pushbutton is pressed, the register reset pulse is delayed until after the $\overline{\text{H DRIVE FOR MON}}$ pulse goes high again.

3.6.3.2 Divide by 45 Counter and Decoder. The divide by 45 counter and decoder accepts the 1.4175 MHz $\overline{\text{to.3}}$ pulse train to produce the following 31.5 kHz timing pulses: $\overline{\text{HORIZ DRIVE ENABLE}}$, $\overline{\text{45 CNT 5}}$, $\overline{\text{HORIZ DRIVE DISABLE}}$, $\overline{\text{45 CNT 27}}$, $\overline{\text{45 CNT 42}}$, and 2XH CLOCK. The timing relationships of these signals are illustrated in figure 3-6. The divide by 45 counter advances one count every 700 nanoseconds on the trailing edge of each $\overline{\text{to.3}}$ input pulse. Counts 1, 5, 11, 27, 42, and 44 of the counter are then decoded to produce the required timing pulses. After each count of 44 is reached, the counter is reset to zero again by the following (45th) $\overline{\text{to.3}}$ pulse. The counter then begins to count up again on each subsequent $\overline{\text{to.3}}$ pulse. The $\overline{\text{HORIZ DRIVE ENABLE}}$, $\overline{\text{45 CNT 5}}$, $\overline{\text{HORIZ DRIVE DISABLE}}$, $\overline{\text{45 CNT 27}}$, $\overline{\text{45 CNT 42}}$ are applied to the horizontal drive generator; the 2XH CLOCK is applied to both the horizontal drive generator and the divide by 525 (or 629 for 50-Hz model terminals) counter and decoder.

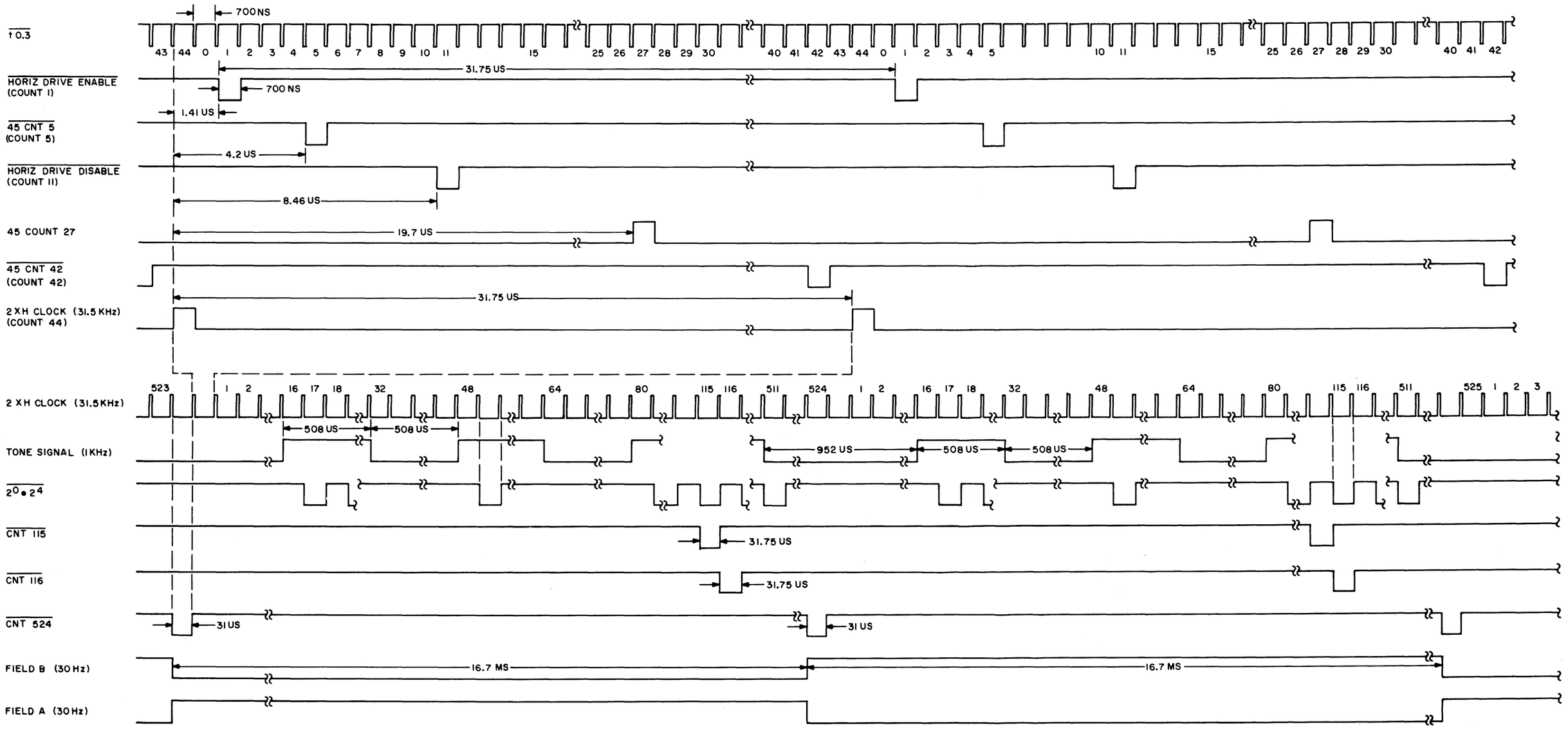


Figure 3-6. Divide by 45 and Divide by 525 Counters (50-Hz Models), Timing Diagram

3.6.3.3 Divide by 525/629 Counter and Decoder. The function of the divide by 525/629 counter and decoder is to divide the 31.5 kHz 2XH CLOCK down to the proper vertical sweep rate (field rate) to control the field A/B flip-flop, start refresh pulse logic, and vertical drive generator. Sixty-Hz models employ a divide by 525 counter and decoder to develop the required 60-Hz field rate, while 50-Hz models use divide by 629 counters and decoders to produce the 50-Hz field rates. Since the divide by 525 and divide by 629 counter and decoder circuits are similar, the operation of the divide by 525 counter and decoder will be described first, followed by a brief description of the differences in the operation of the divide-by-629 counter and decoder. Refer to figure 3-6 for the timing of the divide by 525 counter and decoder output signals.

The divide by 525 counter is advanced one count every 31.75 microseconds by the trailing edge of each 2XH clock input from the divide by 45 counter and decoder. Each time the counter reaches a count of 524, vertical drive is set and the leading edge of the next clock input resets the counter to zero. The 2^0 and 2^4 outputs of the counter are ANDed together to provide the $2^0.2^4$ counter to reset the vertical drive generator. This pulse occurs every 16.7 milliseconds, or 60 times each second. Counts 115, 116, and 524 of the counter are also decoded every 16.7 milliseconds to provide the 60-Hz $\overline{\text{CNT115}}$, $\overline{\text{CNT116}}$, and $\overline{\text{CNT524}}$ pulse trains. The 31.75-microsecond, negative going $\overline{\text{CNT115}}$ and $\overline{\text{CNT116}}$ pulses are applied to the start refresh pulse logic; the 31-microsecond, negative-going $\overline{\text{CNT524}}$ pulses are applied to the horizontal drive generator, vertical drive generator, and field A/B flip-flop. In addition, the 2^4 output of the divide by 525 counter provides a 1-kHz tone signal for the Input/Output Logic.

The leading edge of each $\overline{\text{CNT524}}$ pulse toggles the field A/B flip-flop to alternately produce the 16.7-microsecond field A and field B gates. The field A gate is applied to the horizontal drive generator; both gates are applied to the start refresh pulse logic. The field A and field B gates are also routed to the Character Video Logic where they preset the row counters for row 1 or row 2 for fields A and B, respectively.

The operation of the divide by 629 counter and decoder differs from the divide by 525 counter and decoder as follows. The divide by 629 counter counts up to 628 before being reset by the leading of the next clock input, and counts 171, 172, and 628 are decoded as outputs in place of counts 115, 116, and 524, respectively. The timing of the divide by 629 counter and decoder outputs is shown in figure 3-7. The $2^0.2^4$ count 17 pulse and the $\overline{\text{CNT171}}$, $\overline{\text{CNT172}}$, and

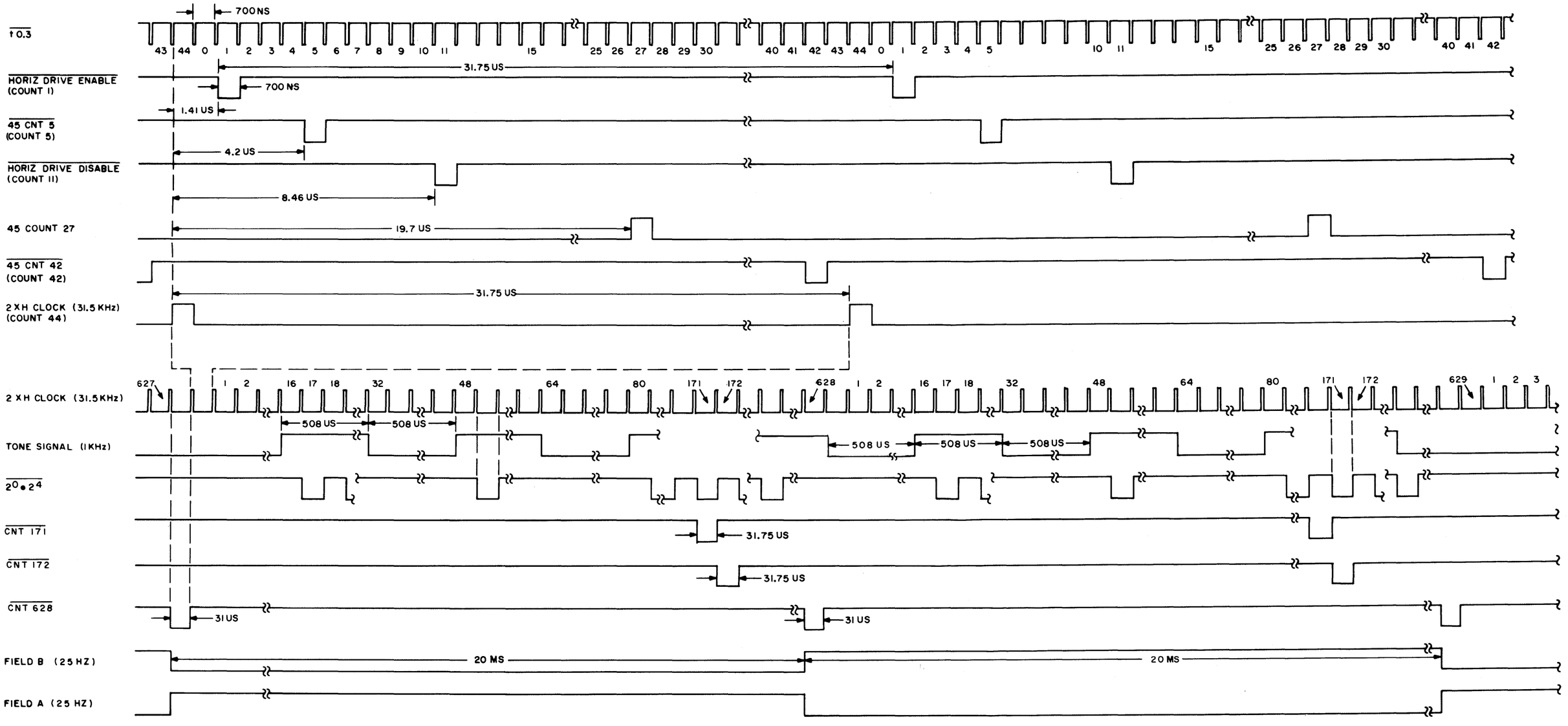


Figure 3-7. Divide by 45 and Divide by 629 Counters (60-Hz Models), Timing Diagram

$\overline{\text{CNT628}}$ pulses are decoded every 20 milliseconds, or 50 times each second. In addition, since the field A/B flip-flop is toggled by the $\overline{\text{CNT628}}$ pulses, 20-millisecond field A and field B gates are generated by the 50-Hz models.

3.6.3.4 Vertical Drive Generator. The vertical drive generator receives timing pulses from the divide by 525 (or 629 for 50-Hz models) counter and decoder to generate 570-microsecond vertical drive sync pulses for the Monitor, Refresh Logic, Internal Program Counter, and Decision Logic circuits. As shown in figure 3-8, the vertical drive pulses are initiated by the leading edge of each $\overline{\text{CNT524}}$ (or $\overline{\text{CNT628}}$) pulse and are terminated by the leading edge of the first $2^0.2^4$ pulse after each $\overline{\text{CNT524}}$ (or $\overline{\text{CNT628}}$) pulse. This produces a 570-microsecond vertical drive pulse at the beginning of each field. The negative-going $\overline{\text{VERT DRIVE FOR MON}}$ pulses are routed to the monitor to synchronize the vertical sweep sawtooth voltages with the 60 (or 50) hertz field rate. Thus, a vertical sweep is initiated every 16.7 milliseconds in the 60-Hz model and 20-milliseconds in the 50-Hz models. The vertical drive outputs to the Refresh Logic, Internal Program Counter, and Decision Logic circuits are used to synchronize various functions with the vertical sweep of the display raster. These functions are described when the individual circuit groups are discussed. A buffered $\overline{\text{VERT DRIVE FOR REMOTE MONITOR}}$ output is made available at pin C of connector J4 for remote monitor installations.

3.6.3.5 Horizontal Drive Generator. The function of the horizontal drive generator is to generate horizontal drive pulses for the monitor circuit and horizontal sync pulses for various other circuits of the terminal. The horizontal drive pulses are applied to the horizontal deflection circuits of the monitor to produce both the horizontal scans for the raster and the dc voltages for the CRT. The horizontal sync pulses are used to synchronize certain terminal operations with the horizontal scans. Figure 3-9 illustrates the timing of the horizontal generator.

The 2XH clock pulses from the divide by 45 counter and decoder are applied to a divide by 2 flip-flop which divides the 31.5 kHz input clock rate down to approximately 15 kHz for the horizontal drive generator outputs. When the Q-output of the flip-flop is high, the leading edge of the $\overline{\text{45CNT42}}$ input pulse initiates an $\overline{\text{H DRIVE FOR MON}}$ pulse. The pulse is then terminated by the trailing edge of the next 45CNT27 pulses from the divide by 45 counter and decoder. Thus, a 21.8 microsecond, negative-going pulse is applied to the monitor every 63.5 microseconds to initiate a horizontal scan. As shown in figure 3-9, the first pulse for

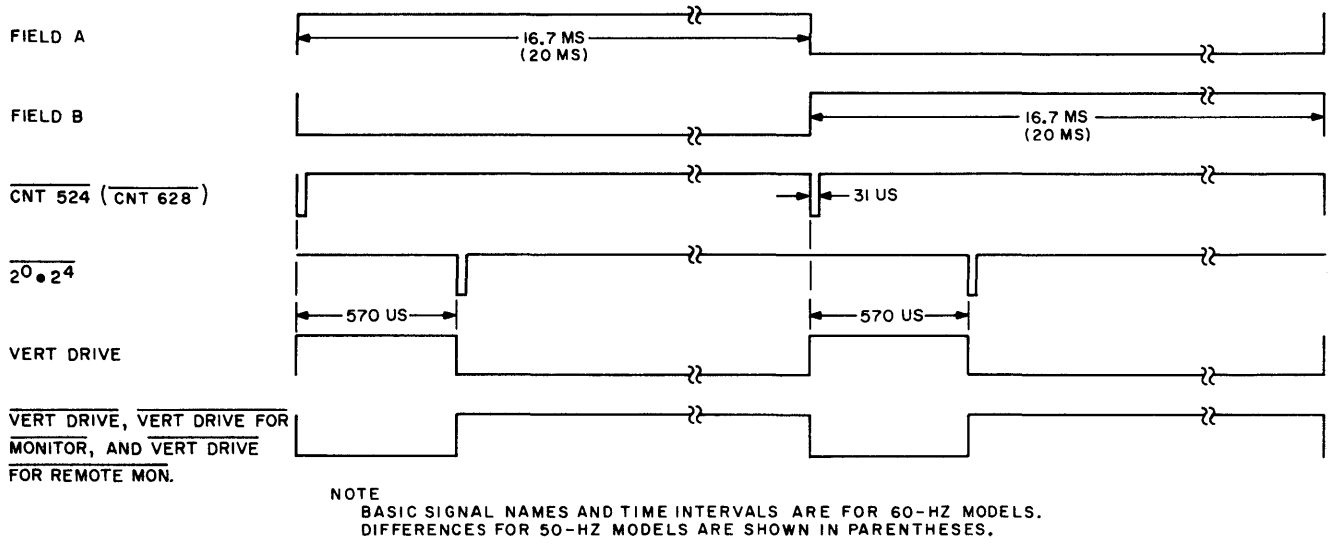
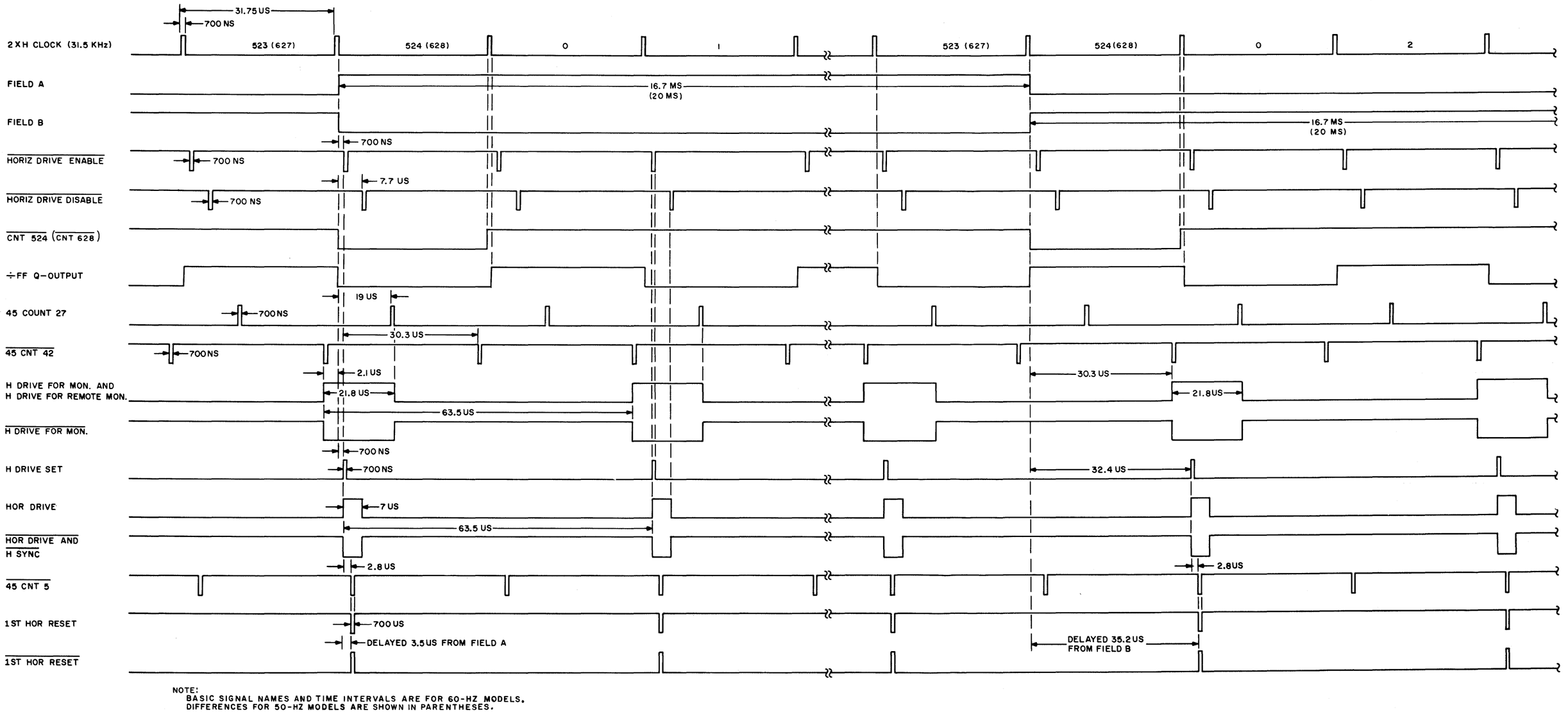


Figure 3-8. Vertical Drive Generator, Timing Diagram



NOTE:
BASIC SIGNAL NAMES AND TIME INTERVALS ARE FOR 60-HZ MODELS.
DIFFERENCES FOR 50-HZ MODELS ARE SHOWN IN PARENTHESES.

Figure 3-9. Horizontal Drive Generator, Timing Diagram

field A occurs 2.1 microseconds before the field A gate while the first pulse for field B occurs 30.3 microsecond after the field B gate. Since the vertical scans for both fields are initiated at the start of each gate, the horizontal scans during field B will occur in between the horizontal scans of field A. This produces the 2:1 interlace raster. To ensure that the first H DRIVE FOR MON pulse for field A occurs just prior to the field A gate, the divide by 2 flip-flop is initially reset by the first $\overline{\text{HORIZ DRIVE ENABLE}}$ pulse after the field A input goes high. This orients the flip-flop such that the Q-output will always be high just prior to field A, and low prior to field B. For the 60-hertz unit, 262-1/2 horizontal scans are produced during field A and 262-1/2 for field B. For the 50-Hz units, 314-1/2 horizontal scans are produced for field A and 314-1/2 for field B. The horizontal drive generator also provides a buffered H DRIVE FOR REMOTE MON output for driving remote monitors. This signal is available at pin A of connector J4. In addition, a negative-going $\overline{\text{H DRIVE FOR MON}}$ is routed to the reset circuit for the t-pulse generator.

The Q-output of the divide-by-2 flip-flop also controls the generation of the horizontal sync pulses. The circuits that develop the H DRIVE SET and HOR DRIVE outputs are enabled when the Q-output of the flip-flop is low. The 700-nanosecond H DRIVE SET pulse is produced by the $\overline{\text{HORIZ DRIVE ENABLE}}$ input and the HOR DRIVE pulse is generated from the leading edge of the $\overline{\text{HORIZ DRIVE ENABLE}}$ pulse to the leading edge of the $\overline{\text{HORIZ DRIVE DISABLE}}$ pulse. The $\overline{\text{HOR DRIVE}}$ pulses are, in turn, ANDed with the $\overline{\text{45CNT5}}$ pulses from the divide by 45 counter and decoder to produce the 1st HOR RESET and 1st $\overline{\text{HOR RESET}}$ outputs. The destinations of the horizontal sync pulses are shown on figure 3-4. The HOR DRIVE pulse is also applied to the start refresh pulse logic circuit.

3.6.3.6 Start Refresh Pulse Logic. The function of the start refresh pulse logic is to signal the Refresh Logic circuit when to initiate the refresh cycle for each field (vertical scan). This synchronizes the displayed video with the CRT raster. For the 60-Hz units, a START REFRESH pulse is applied to the Refresh Logic circuit at the start of the 58th horizontal scan of each field. Thus, the first row of video to be refreshed each frame is painted during the 58th horizontal scan of field A: the first 57 horizontal scans are always blanked. For the 50-Hz units, the START REFRESH pulses are not generated until the 86th horizontal scan of each field. Because the start refresh pulse logic circuits for all models are identical, only the operation of 60-Hz units are described. The start refresh pulse logic

timing for 60-hertz units is shown in figure 3-10, and in figure 3-11, for 50-hertz models.

The $\overline{\text{CNT115}}$ and $\overline{\text{CNT116}}$ inputs are ANDed with the field A and field B gates, respectively, to produce the FIELD A · $\overline{\text{CNT115}}$ and FIELD B · $\overline{\text{CNT116}}$ signals shown in figure 3-10. These signals are then ORed to produce a signal having positive pulses at count 115 of field A and count 116 of field B. This signal, (field A · $\overline{\text{CNT115}}$) + (field B · $\overline{\text{CNT116}}$), is used to gate the 58th HOR DRIVE input pulse of each field to the start refresh pulse output.

3.6.4 DECISION LOGIC

The Decision Logic is the traffic control section of the terminal. Its main function is to allow orderly implementation of terminal functions whether routine refresh or entry and editing. The functions are implemented by utilizing the functional sections of the terminal in a predetermined sequence dependent upon the function called for. The refresh function is described in paragraph 3.6.7 and the other functions are discussed in section 3.7.

3.6.4.1 Refresh Cycle. The refresh cycle is a repetitive operation that takes place under control of timing signals from the Clock and Display Sync Circuits synchronized with the display raster.

The normal reset signal for the refresh cycle is VERT DRIVE which is ORed together with SYSTEM RESET + SKIP REF CYCLE from the Decision Logic. The refresh cycle generates a $\overline{\text{R/R REFRESH}}$ signal which goes to the Decision Logic and in conjunction with $\overline{\text{ROW 1 THRU 7 (REFRESH)}}$ enables the R/R Command F/F. When timing signal t0.6 sets the R/R Command F/F it develops R/R COMMAND TO MEMORY and $\overline{\text{R/R COMMAND TO MEMORY}}$. R/R COMMAND TO MEMORY is combined with timing signal t0.4 developing $\overline{\text{CLEAR A DATA REG}}$ and STROBE FG/BG. When timing signal to.5 occurs STROBE A DATA REG is developed. $\overline{\text{R/R COMMAND TO MEMORY}}$ is routed to the Memory Logic and Character Video togic. It is also combined with $\overline{\text{R/R REFRESH}}$ to develop $\overline{\text{CHAR GEN ENABLE}}$ for the Character Video Logic.

3.6.4.2 C/W Commands. The prime function of the Decision Logic during non-refresh time is the steering and enabling of the Internal Program Counter. An additional function is the development of C/W or C/W ZERO signals. The C/W or C/W ZERO signals enable the C/W Command F/F. The C/W ZERO signal is also routed

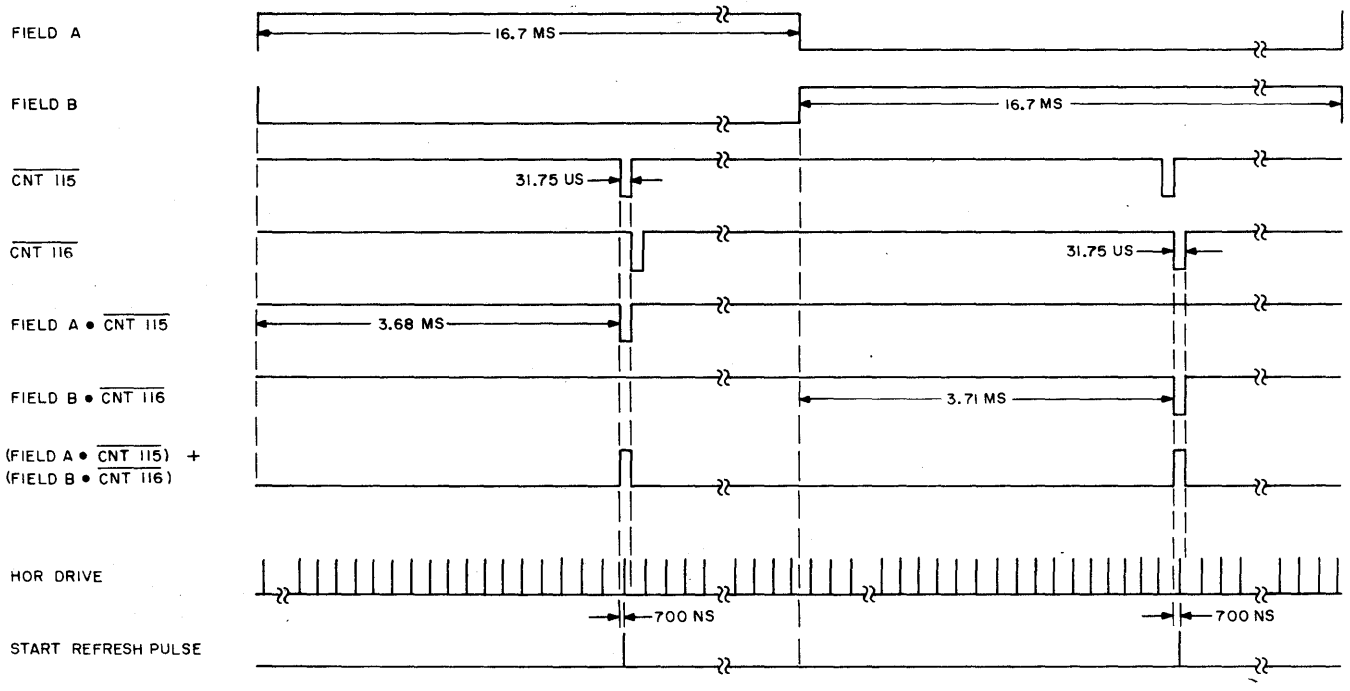


Figure 3-10. Start Refresh Pulse Logic (60-Hz Models), Timing Diagram

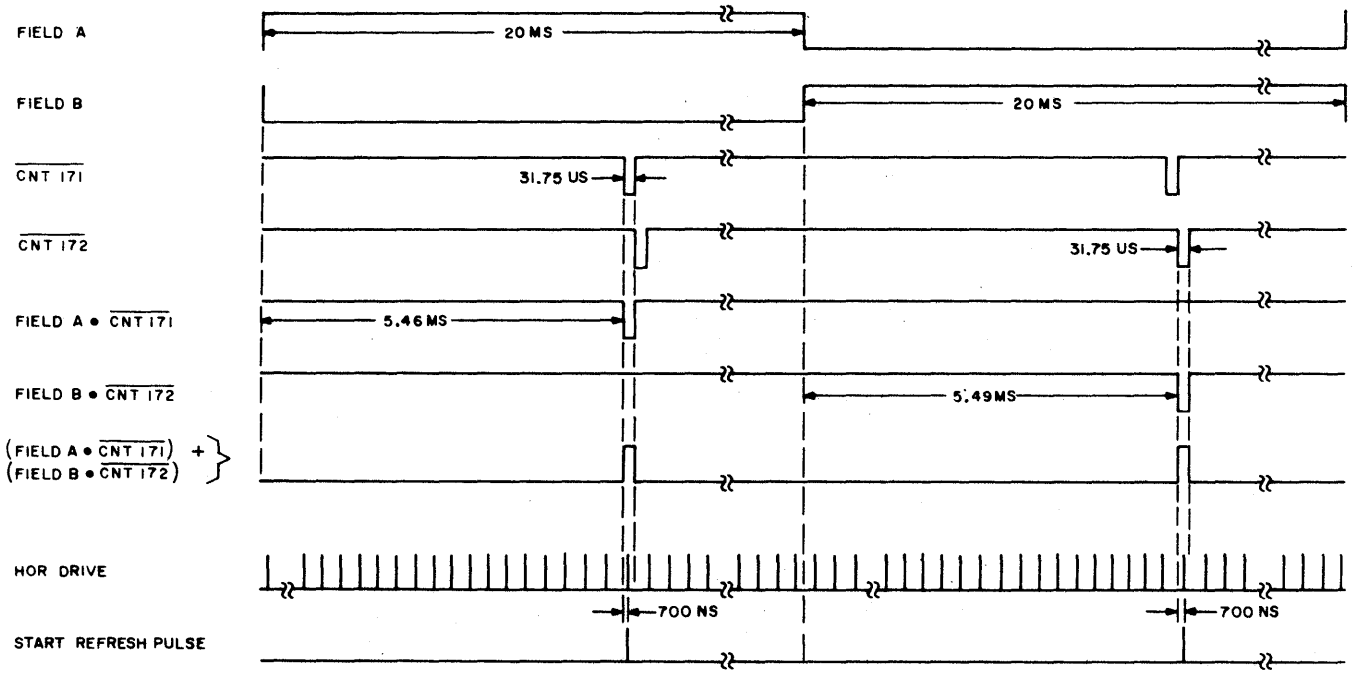


Figure 3-11. Start Refresh Pulse Logic (50-Hz Models), Timing Diagram

to the Memory Logic where it is used with $\overline{\text{CLEAR B DATA REG EXCEPT BIT 6 \& 8}}$ which is C/W ZERO ANDed with timing signal t0.6. The C/W Command F/F is set by timing signal t0.6 and develops $\overline{\text{C/W COMMAND}}$ to the Memory Logic.

3.6.5 INTERNAL PROGRAM COUNTER

The Internal Program Counter (P-CTR) determines the conduct of the terminal when an entry or command is initiated by the keyboard operator or the CPU. The P-CTR is enabled, when required, during the horizontal retrace interval. Some functions of the terminal may be accomplished in one horizontal retrace period with no impact on the normal refresh cycle. However, for certain functions more time is required than is available during a normal retrace period therefore a skip refresh signal is developed which interrupts the normal refresh cycle. During a skip refresh interval a momentary blanking of the display occurs. Once the P-CTR is enabled it is clocked thru its seven stages sequentially by timing signal t0.1. Each stage generates an output, P-CTR-1 thru P-CTR-7, and their complements which enable various logic operations to take place implementing the function initiated. Some functions require operations to be done repeatedly; for these the P-CTR and the Decision Logic will provide loops enabling the respective P-CTR stages to recycle until the entire function has been implemented. There are fourteen specific functions that are implemented under P-CTR control and they are described in detail in paragraphs 3.7.

3.6.5.1 P-CTR Enabling (figure 3-12). There are two signals that enable the P-CTR Entry Control F/F; $\overline{\text{ALLOWED KB STROBE}}$ and $\overline{\text{PROCESS (I/O) ENTRY}}$ for keyboard and CPU initiated functions respectively. For REPEAT operations a third enabling input, derived from the Line Repeat Counter acting as a divide-by-four element clocked by VERT DRIVE, enables the P-CTR Entry Control F/F during the interval the REPEAT key is depressed. The enabled Entry Control F/F, set by the H DRIVE SET signal, enables the "A" flip-flop. When timing signal t0.0 occurs, its trailing edge sets the enabled "A" flip-flop. The set states of the Entry Control flip-flop and the "A" flip-flop enable the "C" flip-flop which is clocked by timing signal t0.4. The "C" flip-flop then causes the "A" flip-flop to be reset at the trailing edge of the next t0.0. The "C" flip-flop itself will remain locked in the set state until the ENTRY CONTROL FF is reset, which occurs when its enabling signal goes false. The "A" flip-flop, in addition to enabling the "B" and "C" flip-flops, develops a STROBE KB REG signal which loads the keyboard command

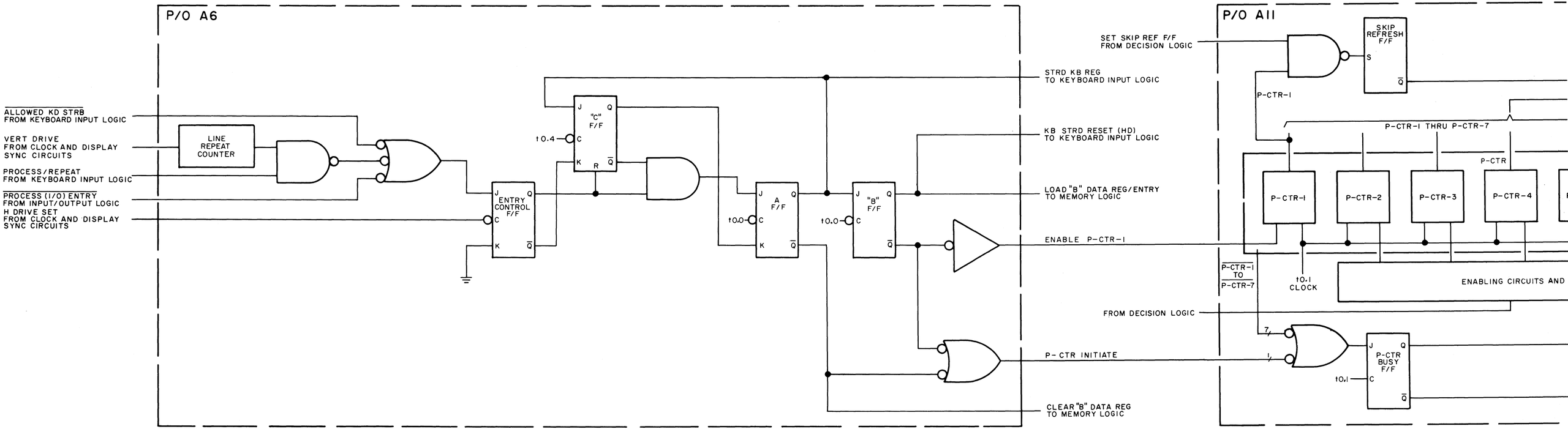
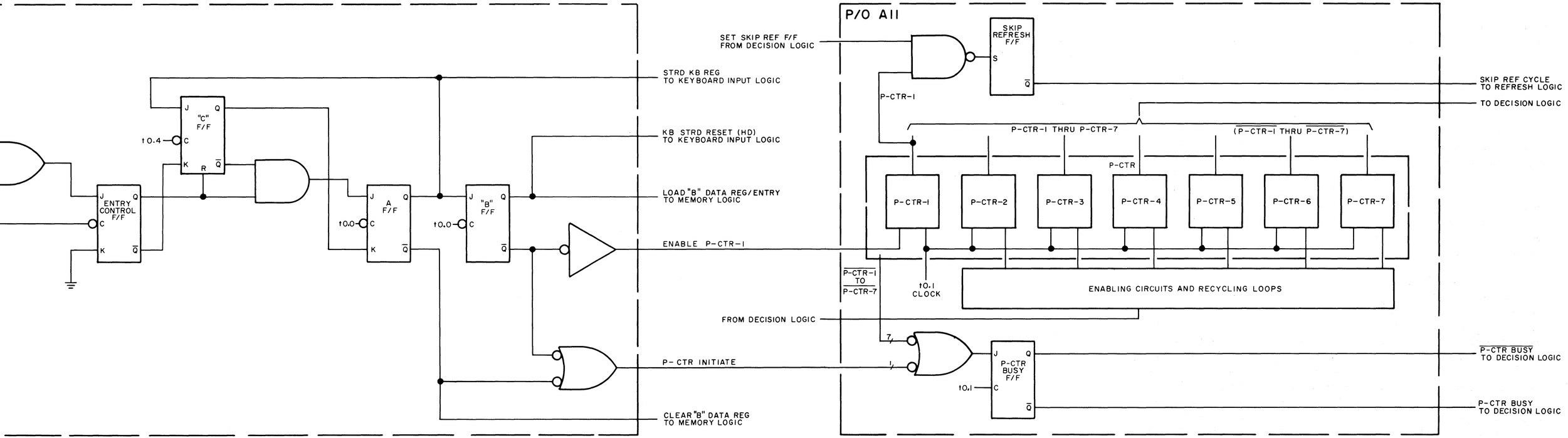


Figure 3-12. Internal Program Counter Block Diagram

minal


P/O A11
SET SKIP REF F/F FROM DECISION LOGIC
SKIP REFRESH F/F
S
Q
SKIP REF CYCLE TO REFRESH LOGIC
TO DECISION LOGIC
P-CTR-1
P-CTR-1 THRU P-CTR-7
(P-CTR-1 THRU P-CTR-7)
P-CTR
P-CTR-1
P-CTR-2
P-CTR-3
P-CTR-4
P-CTR-5
P-CTR-6
P-CTR-7
STRD KB REG TO KEYBOARD INPUT LOGIC
KB STRD RESET (HD) TO KEYBOARD INPUT LOGIC
LOAD "B" DATA REG/ENTRY TO MEMORY LOGIC
ENABLE P-CTR-1
P-CTR-1 TO P-CTR-7
t0.1 CLOCK
ENABLING CIRCUITS AND RECYCLING LOOPS
FROM DECISION LOGIC
P-CTR INITIATE
P-CTR BUSY F/F
t0.1
P-CTR BUSY TO DECISION LOGIC
P-CTR BUSY TO DECISION LOGIC
CLEAR "B" DATA REG TO MEMORY LOGIC
"C" F/F
t0.4
ENTRY CONTROL F/F
A F/F
t0.0
"B" F/F
t0.0

into the 4 stage Storage Register/Decoder Circuit of the Keyboard Input Logic. A $\overline{\text{CLR "B" DATA REG/ENTRY}}$ signal is also developed which is applied to the Memory Logic to clear the "B" Data Register. The same signal line carrying the $\overline{\text{CLR "B" DATA REG/ENTRY}}$ signal is used to develop the $\overline{\text{P-CTR INITIATE}}$ signal which is connected to the P-CTR busy OR circuit. The enabled "B" flip-flop is clocked by the trailing edge of timing signal t0.0 and develops the following signals. $\overline{\text{KB STRB RESET (HD)}}$ clears the input flip-flops of the Keyboard Entry Logic. LOAD "B" DATA REG/ENTRY is applied to the Memory Logic and allows the keyboard data to be set into the "B" Data Register. $\overline{\text{P-CTR INITIATE}}$ is again developed and applied to the P-CTR Busy OR circuit. ENABLE P-CTR-1 enables the first stage of the P-CTR.

3.6.5.2 P-CTR Operation. Once the P-CTR is enabled it is clocked sequentially thru its seven stages by timing signal t0.1. One output from each stage, in addition to being routed to other destinations, is connected to the P-CTR Busy OR circuit. The output of the P-CTR Busy OR circuit enables the P-CTR Busy F/F which generates P-CTR BUSY and its compliment when clocked by timing signal $\overline{\text{t0.1}}$. These signals indicate the status of the P-CTR to the Decision Logic and the Input/Output Logic. When a particular function is to be implemented under P-CTR control that requires more time than is available during one horizontal retrace period, P-CTR-1 in conjunction with the SET SKIP REF F/F signal sets the Skip Refresh F/F. The Skip Refresh F/F develops the $\overline{\text{SKIP REF CYCLE}}$ signal which inhibits the Refresh F/F in the Refresh Logic. The outputs of the P-CTR are routed primarily to Decision Logic which in turn develops enabling signals that are fed back to the P-CTR. The Decision Logic determines what outputs are to be used to develop signals to implement the respective function called for. The enabling of successive stages of the P-CTR are dependent upon the function called for and the display position at which it is implemented. The functions implemented under P-CTR control are discussed in detail in paragraphs 3.7 which show, by means of logic flow charts, the signal flow for each step in the implementation of a function.

3.6.6 MEMORY LOGIC

The main function of the Memory Logic, figure 3-13, is to store data in the form of 8-bit words and allow access to this data for information or modification. The 8-bit words are stored in a magnetic-core memory consisting of one integral unit composed of cards B3 and B4. The core-memory locations are addressed for read/write operations by X and Y driver boards B1, B2 and B6, B7 respectively. The

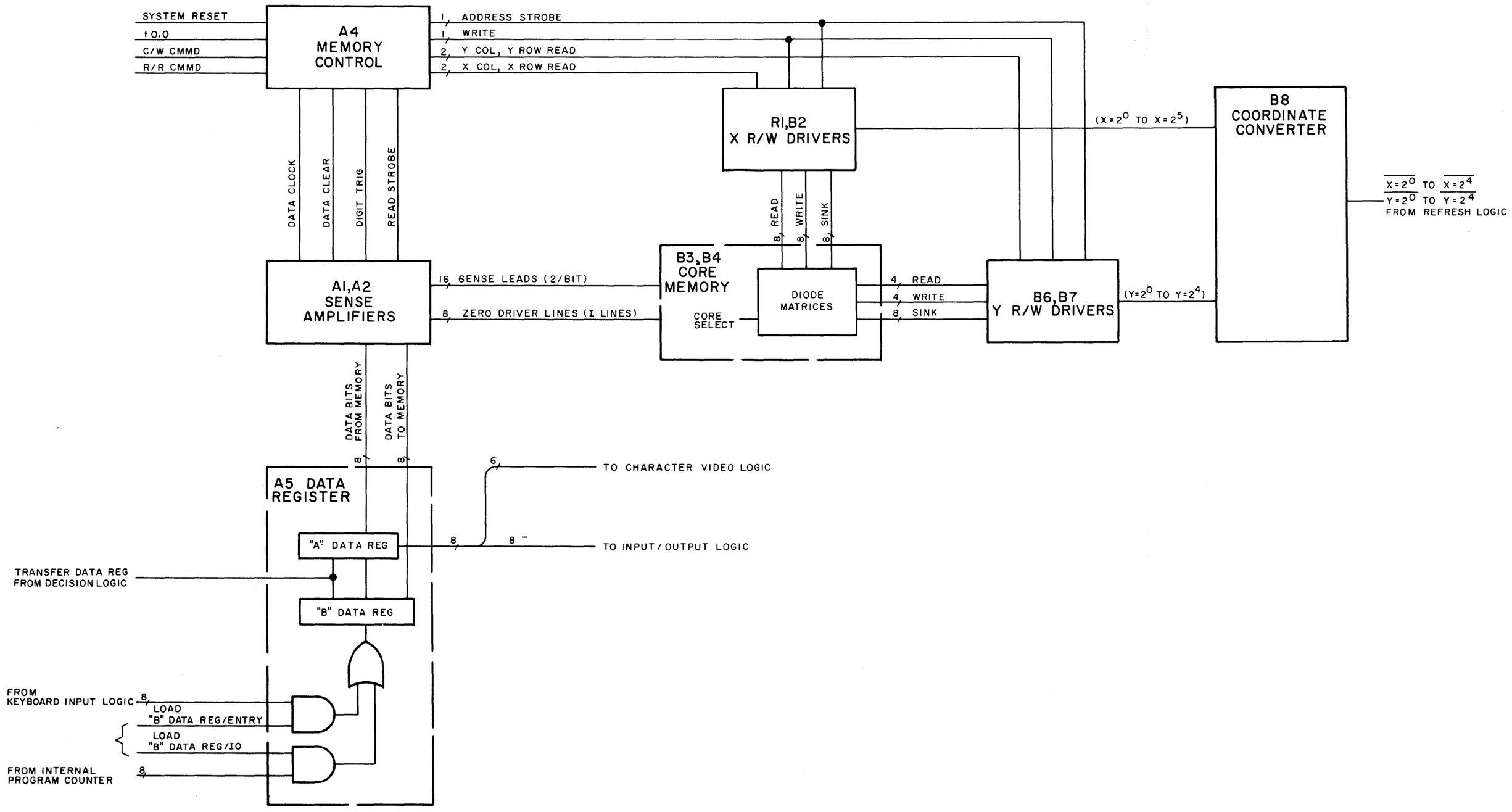


Figure 3-13. Memory Logic Block Diagram

address information is provided to the driver boards by Coordinate Converter, B8, which converts the 74x27 display coordinates to 64x32 memory coordinates. The read/write signals are generated in their proper sequence, upon commands from Refresh Logic or Decision Logic, by Memory Control A4. The sense amplifiers which are used during read operations are contained in Sense Amplifiers A1 and A2. The main interface between the magnetic-core memory and Keyboard Logic, Character Video Logic and Input Output Logic is A5, the "A" (read) and "B" (write) Data Register.

3.6.6.1 Magnetic Core Memory (figure 3-14). The magnetic-core memory has 2048 addressable locations each containing eight bits. Each core is threaded by three conductors, two drive lines, X and Y, and a sense/digit line. The sense/digit line is threaded thru all the cores in each bit (or plane). The ends of the sense/digit lines are terminated at the differential inputs to the sense amplifiers. The X and Y drive lines are the outputs of two matrices. The X matrix, 8x8, yields 64 x selections and the Y matrix, 4x8, yields 32 Y selections. The matrices are supplied with write, read and sink lines using two diodes per line for read-write selection. Read current (+Is) is defined as current entering the memory. Write current (-Is) is defined as current leaving the memory.

3.6.6.2 R/W Drivers (figure 3-14). Since B1 and B2 are functionally comparable to B7 and B6 respectively only a typical B1 and B2 circuit is discussed. B1 contains the read/write drivers which are addressed by the Coordinate Converter and the selected x row driver is enabled by a $\overline{\text{READ TRIG}}$ or $\overline{\text{WRITE TRIG}}$ signal. B2 contains sink circuits which are also addressed by the coordinate converter and the selected x column sink circuit is enabled by a $\overline{\text{READ TRIG}}$ or $\overline{\text{WRITE TRIG}}$ signal.

3.6.6.3 Coordinate Converter. The Coordinate Converter consists of four 4-bit binary full adders. The inputs to the adders are supplied from the Refresh Logic where a selection of refresh address or cursor address is made. The input address data is in display dimensions $x=2^0$ thru $x=2^6$, and $y=2^0$ thru $y=2^4$. The input address data is converted to memory address dimensions of $x=2^0$ thru $x=2^5$, and $y=2^0$ thru $y=2^4$.

3.6.6.4 Memory Control. Memory Control generates control signals for the R/W Drivers and the Sense Amplifiers. The relationship of the signals developed is

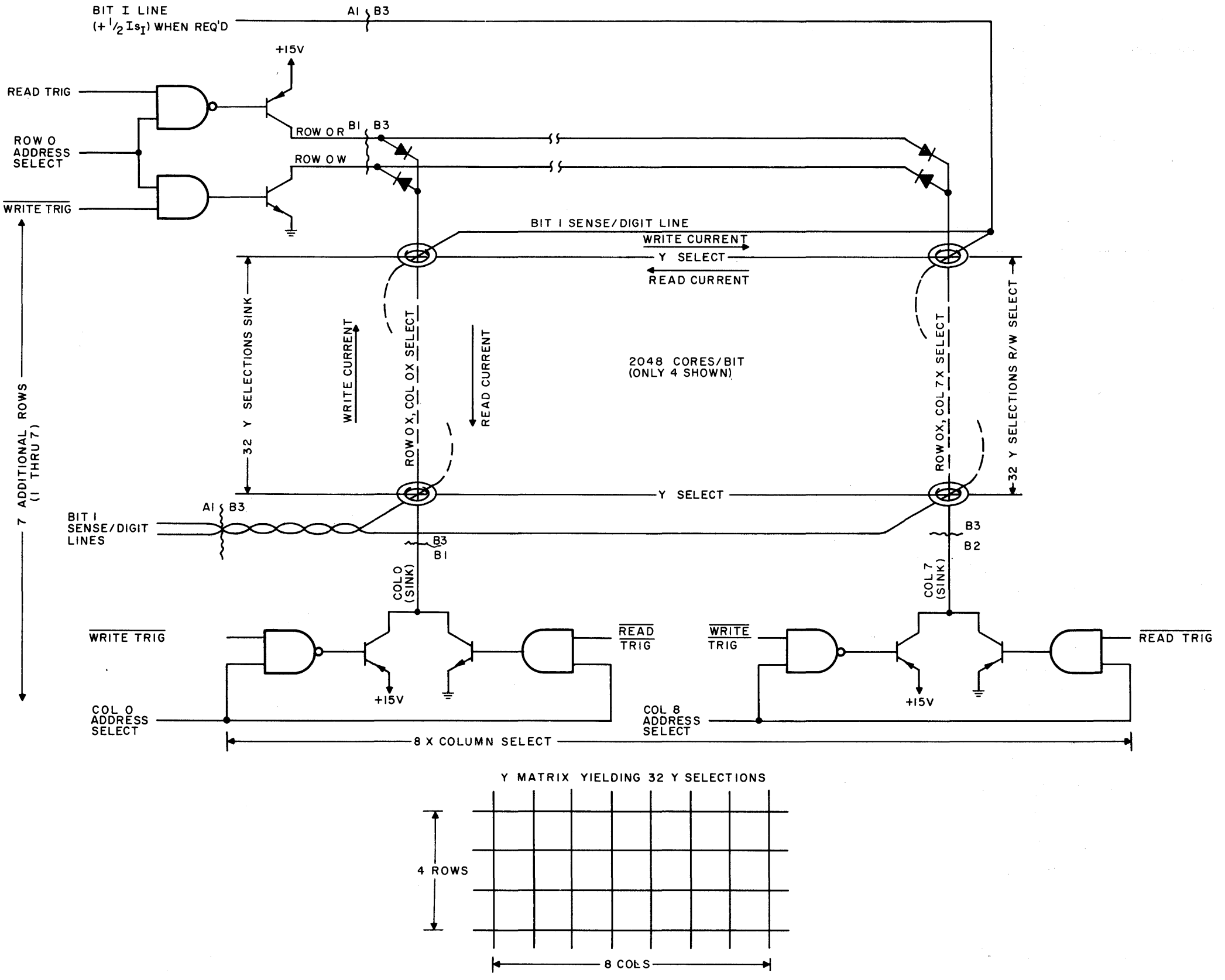


Figure 3-14. Read, Write, Core Selection

shown in figure 3-15. Figure 3-15(A) illustrates signals which are developed without regard to whether a C/W or R/R command has been given. Figure 3-15(B) and (C) illustrate signals developed where R/R (Read/Restore) or C/W (Clear/Write) commands are present respectively. Figure 3-15-1, ADD STROBE, is sent to the R/W Drivers and gate in the address information corresponding to the memory location to be operated upon. Figure 3-15-2, DATA CLEAR, is the last signal in the sequence and clears the data storage flip-flops, located with the Sense Amplifiers, that contained the data just previously written or restored into memory.

3.6.6.4.1 R/R Command, Read/Restore. When a R/R operation is called for, the R/R CMMD signal enables the R/R F/F which is clocked by timing signal t0.0. Setting the R/R F/F allows the Y ROW READ, y COL READ, X COL READ and X ROW READ signals to be developed for the R/W Drivers. The READ signals, figure 3-15-3 enable the read lines in the R/W Drivers and the bit data in memory is detected by the sense amplifiers. When the read lines are energized one half of the sense current (+Is) is contributed by the x lines (+1/2Isx) and one half by the y lines (+1/2Isy). If the data in a particular addressed core is a "zero" (0), no change will take place concerning the magnetic state of the core and the sense amplifiers produce no output. If however the core contains a magnetic state corresponding to a "one" (1) a change in its magnetic state takes place, due to the presence of (+Is), and the sense amplifiers produce a logic level output sufficient to preset the data storage flip-flops when enabled by READ STROBE figure 3-15-4. If the core contained a 1 prior to the read operation, the read operation has placed the core in its 0 magnetic status and the core must now be restored to its 1 condition by a restore (write) operation. If the core was in its 0 magnetic status the read operation has not altered its magnetic state. Therefore the write operation must be conditioned so as not to alter the cores magnetic state during the restore (write) operation. The write current conditioning is accomplished by means of the I lines from the sense amplifiers to the memory and are connected to the center tap of the sense leads for each bit. To write a 1 back into a core, which was read out, requires a current of (-Is). This current is supplied by the write lines, (-1/2Isx) by the x lines and (-1/2Isy) by the y lines. If the core contained a 0 its magnetic state was not altered and the write current must be modified to prevent a 1 being written into the core. The state of the data storage flip-flops depends upon what the sense amplifiers produce during the read operation. If a 1 is read out of memory the flip-flops are not preset and the $\overline{Q}$ output is at its

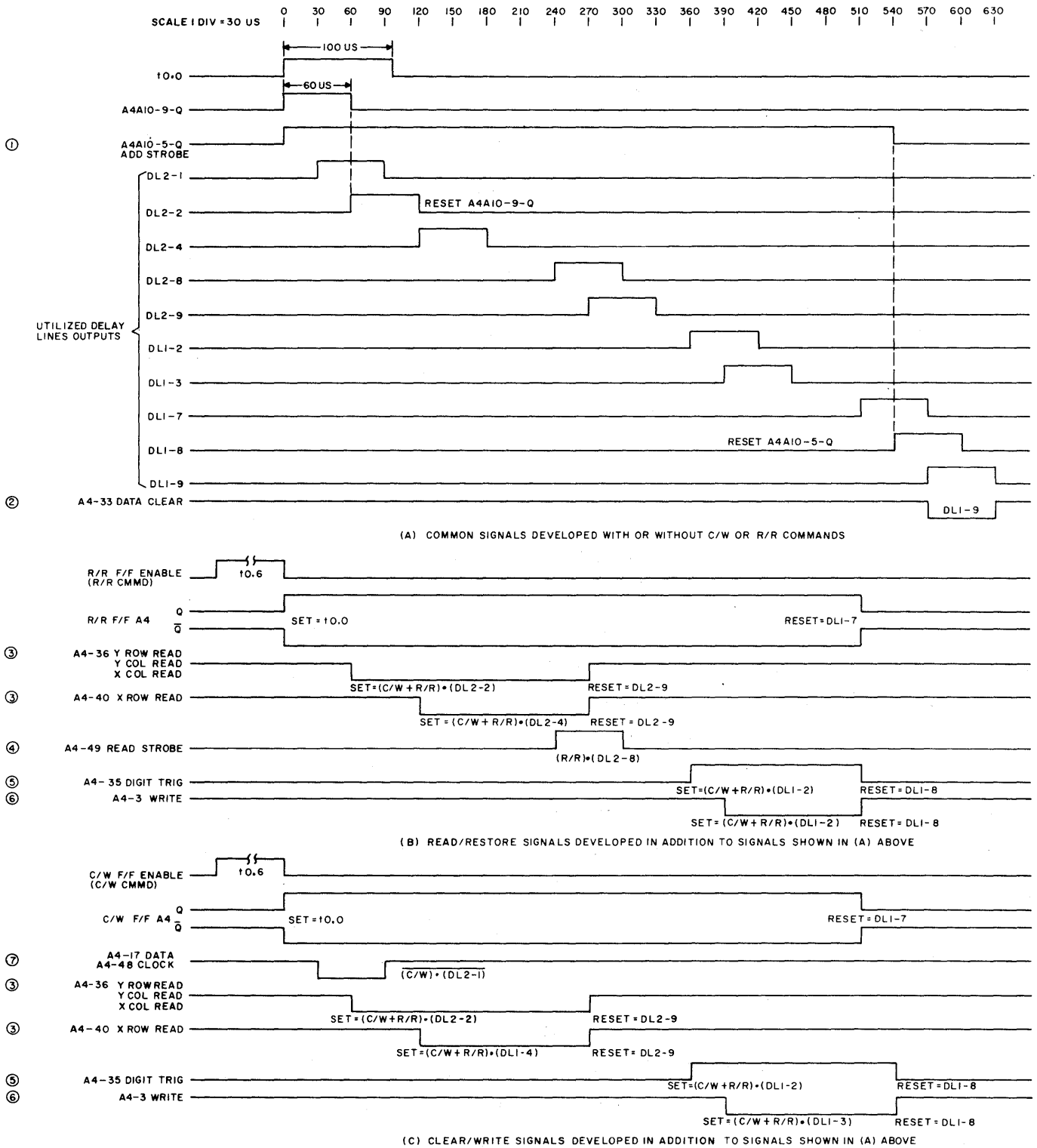


Figure 3-15. Memory Control Timing Diagram

high logic level. With the $\overline{Q}$ output at its high logic level and the occurrence of the DIGIT TRIG signal which occurs 30 nanoseconds prior to the WRITE signal, figure 3-15-5, and -6 respectively, the I line is enabled to carry current into the memory. The magnitude of the I line current is (+1/2Is) and is in the direction to oppose the normal write current provided by the R/W Drivers. The total effective write current under these conditions is $(+1/2Is_I) + (-1/2Is_x) + (-1/2Is_y)$ = (-1/2Is) which is not sufficient to switch the magnetic state of the core and it remains in a magnetic state representing a 0. Had a 1 been read out of the core a nominal write current of (-Is) will be developed due to the inhibiting of $(+1/2Is_I)$ and $(-1/2IS_x) + (-1/2Is_y) = (-Is)$. The $\overline{Q}$ outputs of the data storage flip-flops are connected to the Data Register and are strobed into the "A" Data Reg at the proper time.

3.6.6.4.2 C/W Command, Clear/Write. When a C/W operation is called for the C/W CMMD signal enables the C/W F/F which is clocked by timing signal t0.0. Setting the C/W F/F enables delay line signal DL2-1 to generate the DATA CLOCK signal, figure 3-15-7, which clocks the data storage flip-flops. The resultant state of the data storage flip-flops depends upon what logic levels are applied to their steering inputs (D terminals). The READ signals described above in 3.6.6.4.1 are also generated but have no affect on the data storage flip-flops since the READ STROBE is not generated. Once the data has been set into the data storage flip-flops the write mechanics are identical as those described in 3.6.6.4.1 above.

3.6.6.5 Sense Amplifiers. The sense amplifiers are basically differential amplifiers with the differential inputs supplied by the sense lines. The sense line for each bit is threaded thru all 2048 cores related to that bit and the ends of the sense line are connected to the sense amplifier inputs. The center tap of the sense line is connected to the I line which utilizes the sense leads to modify the write current as discussed in paragraph 3.6.6.4.1. There is a threshold adjustment located on A4, nominally set at 18 millivolts, that sets the threshold for the sense amplifiers. Where a core magnetic state corresponding to a "1" is read the sense amplifiers provide an output voltage level that represents a high logic level. That logic level in conjunction with a READ STROBE signal causes the data storage flip-flops to be preset. If a "0" is read out of a core the voltage developed, if any, is not sufficient to drive the preset logic and the data storage flip-flops remain in their reset state. The data storage flip-flops

are also clocked by the DATA CLOCK signal during a C/W operation. In this operation the preset input is not enabled and the flip-flop state is modified by logic levels applied to its steering input as supplied from the "B" Data Register. The state of the data storage flip-flops are fed back to the "A" Data Register and also determine whether the I lines will be enabled or disabled during the write interval.

3.6.6.6 Data Register, "A" or Read Register and "B" or Write Register. The Data Register consists of two separate 8-bit storage registers, an "A" or Read register and a "B" or Write register. The "A" register is denoted the read register due to the fact that data bits residing in the Sense Amplifier data storage flip-flops are set into the "A" register when a STROBE "A" DATA REG signal occurs. The outputs of the "A" Data Register are provided to the Character Video Logic, Input/Output Logic, Decision Logic and the "B" Data Register. The "B" register is denoted the write register since the outputs of the "B" register steer the Sense Amplifiers data storage flip-flops. When the data storage flip-flops are clocked by the DATA CLOCK signal they then contain the same data as is in the "B" register and await a DIGIT TRIG signal that will condition the bit write currents.

3.6.6.6.1 R/R Operation. When a read/restore operation is to take place, the data bits are read out and stored in the "A" Register which was cleared by a CLEAR "A" DATA REGISTER signal. The $\overline{\text{STROBE "A" DATA REG}}$ signal sets the data in. The restore operation is identical to those described in paragraph 3.6.6.4.1.

3.6.6.6.2 C/W Operation. Prior to the occurrence of the DATA CLOCK signal, which transfers the data bits from the "B" Data Register into the storage data flip-flops, data may be set into the "B" Data Register from several sources. One source is the "A" Data Register; the other sources are the Keyboard Logic and the Input/Output Logic. The "A" Data Register contents are written into the "B" Register by a $\overline{\text{TRANSFER DATA REG}}$ signal, which is developed during I/L, I/C, D/L, and D/C operations. The Keyboard Logic provides seven bits of data for bits one thru seven. The eighth bit which denotes the foreground/background status of the data word is always entered as a "1" denoting foreground data. Keyboard data is set into the "B" Data Register by the LOAD "B" DATA REG/ENTRY signal. The Input Output Logic supplies eight bit data words which are set into the "B" Data Register by the LOAD "B" DATA REG/IO signal. Prior to any data entry the "B" Data Register is cleared by the $\overline{\text{CLEAR "B" DATA REG/ENTRY}}$ signal developed by the Internal

Program Counter Entry Logic. If data is to be entered from the Input Output Logic another clearing signal is developed, $\overline{\text{CLEAR "B" DATA REG/IO}}$, as part of the A/N Entry routine, which again clears the "B" Data Register to insure proper data entry.

3.6.6.6.3 C/W Zero. A C/W ZERO command occurs when a blank character is to be displayed as when the SPACE bar on the keyboard is depressed. A ($\overline{\text{CLEAR "B" DATA REG}}$) $\overline{\text{EXCEPT BIT 6 \& 8}}$ signal is generated which resets the "B" Data Register stages except those for bits six and eight which are preset.

3.6.6.6.4 EOT Symbols (XMIT + PRINT). When a transmit (XMIT) command is initiated, the bit six and seven memory locations corresponding to the cursor position will be set to a "one" logic state. This data will be decoded as an EOT symbol and displayed as such. When a PRINT command is initiated the bit six and seven memory locations corresponding to the cursor position and the next character position (+1X) will be set to a "one" logic state. EOT symbols will be displayed for these two positions.

3.6.7 REFRESH LOGIC (figure 3-16)

The Refresh Logic develops the signals required to regenerate or refresh the CRT display, unless a skip refresh signal has been developed. A skip refresh will be developed when certain editing functions are called for. The Refresh Logic is controlled by signals, from the Clock and Display Sync Circuits, which primarily occur at the horizontal sweep rate. Once the refresh cycle is initiated, the X and Y coincidence counters are loaded with cursor address information, the character row counter is enabled and a read/restore command is generated. The cursor coincidence counters are incremented as the refresh cycle progresses and, at the proper coincidence of signals the Cursor Address Logic will develop a signal allowing the Character Video Logic to generate a cursor on the CRT. The character row counter generates character row information for the Character Video Logic. The X and Y outputs of the refresh CA-X and CA-Y counters address the Memory Logic and extract character data corresponding to that address location. The CRT display to be pointed or refreshed is developed as a raster scan presentation where each horizontal sweep is intensity modulated and generates the character in a row by row, dot by dot procedure. The character code and row information determine the dot pattern to be displayed for that character on that sweep. The character generation is discussed in greater detail in paragraph 3.6.10.

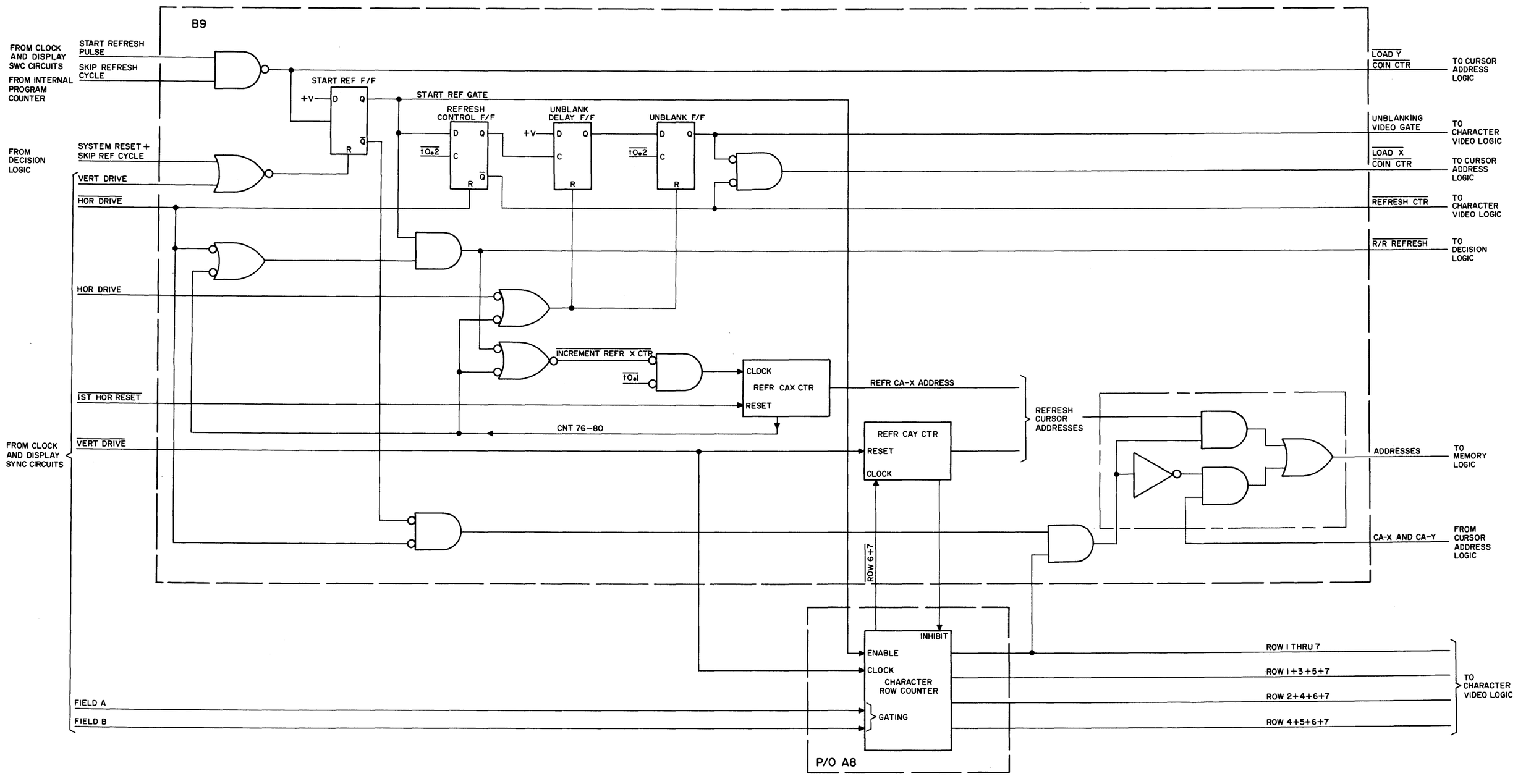


Figure 3-16. Refresh Logic Diagram

3.6.7.1 Refresh Enable Logic. The Start Refresh F/F is set by START REFRESH PULSE and $\overline{\text{SKIP REF CYCLE}}$ which also generate the $\overline{\text{LOAD Y COIN CTR}}$ signal. The outputs of the Start Refresh F/F enable the Character Row Decoder, Refresh Control F/F and with HOR DRIVE provide one enabling signal to the memory address gates. The Refresh Control F/F clocks the Unblank Delay F/F which enables the Unblank F/F. The Refresh Enable Logic now provides the UNBLANKING VIDEO GATE, $\overline{\text{R/R REFRESH}}$, $\overline{\text{REFRESH CTL}}$, $\overline{\text{LOAD X COIN CTR}}$ and $\overline{\text{INCREMENT REFR X CTR}}$ signals. The Character Row Decoder, now enabled, is clocked along by the $\overline{\text{1st HOR RESET}}$ signal and provides character row data to the Character Video Logic. The row data is developed in a sequence that allows the 2:1 interlace feature of the display to be achieved. When the Character Row Counter is gated by the FIELD A signal the row sequence starts out as Row 1, Row 3, Row 5, Row 7, Row 9, Row 2, Row 4, Row 6, Row 8, Row 10, Row 1, Row 3, etc. When, at the occurrance of the next field which is field B, the FIELD B signal gates the Character Row Counter the row sequence starts out as Row 2, Row 4, Row 6, Row 8, Row 10, Row 1, Row 3, Row 5, Row 7, Row 9, Row 2, Row 4, etc. A signal, ROW 1-7 (REFRESH) completes the enabling of the memory address gates which address the memory corresponding to the refresh CA-X and CA-Y location. The $\overline{\text{ROW 6+7}}$ signal which indicates the end of a character row sequence increments the Refresh CA-Y Counter. When the CA-Y Counter reaches a Y line count of 27 an INHIBIT ROW COUNTER signal is developed which inhibits the Character Row Counter. The Refresh CA-Y counter is cleared by $\overline{\text{VERT DRIVE}}$. The Refresh CA-Y Counter enabled by the $\overline{\text{INCREMENT REFR X CTR}}$ signal is clocked along by timing signal to.1. When an X position count of 76 is achieved a COUNT 76-80 signal is developed which inhibits the $\overline{\text{R/R REFRESH}}$ signal and the Refresh CA-X Counter until the next horizontal trace occurs. The Refresh CA-X Counter is cleared by the $\overline{\text{1st HOR RESET}}$ signal.

3.6.8 INPUT/OUTPUT LOGIC

The Input/Output Logic functions as the data interface between the Terminal, Central Processing Unit (CPU), Printer, and Tape Cassette. Under control of function commands and/or control signals from the terminal circuits, CPU, Printer, and Tape Cassette, the Input/Output Logic performs the following functions: transmits data entered at the keyboard to the CPU (Half and Full Duplex-Standard Transmission modes); extracts and processes selected data from the terminal memory for transmission to the CPU, Printer, or Tape Cassette (Batch Transmission and Print modes); accepts and processes data from the CPU or Tape Cassette for display on the

terminal screen (Receive and Full Duplex Standard Transmission modes). In addition, the Input/Output Logic provides the capabilities for printing or recording all communications between the terminal and CPU, as well as printing data being received from the Tape Cassette. The Input/Output Logic is divided into two functional circuit groups, the Input/Output (I/O) Processor and the I/O Interface, which are described separately under paragraphs 3.6.8.1 and 3.6.8.2, respectively.

3.6.8.1 I/O Processor. The I/O Processor functions as the control and data processing portion of the Input/Output Logic. Essentially, The I/O Processor circuits operate in four basic modes, Transmit (Batch), Print, Receive, and Local. In the Transmit mode, the I/O Processor reads each foreground character of the selected data from memory and serially shifts the data to the I/O Interface at the selected band rate. The I/O Interface inserts the proper parity bit in each character word and routes the aerial data to the CPU. In addition, if the Printer is in the on-line mode during Transmit, the I/O Processor automatically inserts a line feed (LF) code after each carriage return (CR). This signals the Printer to do a line feed after each carriage return. The CPU must furnish a LF to the terminal after an EDT is sent as(EOT or CR)in order to maintain correct Printer format. When the first end-of-transmission (EOT) symbol is read from memory, the I/O Processor transmits an end-of-message code (EOT or CR) to the CPU and then switches to the Receive mode. The terminal is then ready to receive and process data from the CPU or Tape Cassette. In addition, if the EOT occurs on the last line of the display, the I/O Processor initiates a delete/line-roll/up function. This deletes the first line and moves all the remaining lines up one line. The bottom line is then left blank for received data.

The Print mode is used for outputting selected data to the Printer or Tape Cassette. The operation of the I/O Processor for the Print mode is similar to the Transmit-Printer on-line operation. However, in the Print mode, both foreground and background characters of the selected data are read from memory and transmitted to the Printer or Tape Cassette. In addition after the last character on each line is sent out, the I/O Processor sends a carriage return (CR) code followed by a line feed (LF) code to the Printer before the first character of the next line is read from memory. This signals the Printer to do a carriage return and line feed at the end of each line. Printer/Tape Cassette priority is accomplished by the I/O Interface. With the Tape Cassette connected and set to the Record mode, serial data is directed to the Tape Cassette; the Printer is not accessed. Data is shifted to the Tape Cassette at the selected baud rate or by

an external clock from the Tape Cassette Unit. If the Tape Cassette is not in the Record mode, data is directed to the Printer and not to the Tape Cassette. Both serial and parallel outputs are provided for the Printer. When the terminal is used with the Hazeltine Printer, data is read out in parallel form, seven character bits at a time. When the Printer receives a set of character bits it sends back a printer busy signal until it is ready to accept the next character. The I/O Processor than waits until the Printer is not busy to send out the next set of character bits. Thus, if a baud rate higher than 300 is selected at the terminal, the resulting waiting automatically reduces the baud rate to about 300. Whenever an EOT symbol is read from memory, the I/O Processor generates a carriage return followed by a line feed code to advance the Printer to the first position on the next line. At the same time, the cursor advances to the first position of the next line to read the next character out of memory. However, it two consecutive EOT symbols are read out of memory, the carriage return and line feed are sent out, the cursor advances to the start of the next line, and the I/O Processor switches to the Receive mode.

In the receive mode, the I/O Processor accepts serial data from either the CPU or Tape Cassette, checks parity (from CPU only), and decodes each character as either a function command or one of the 65 displayable characters. Decoded function commands are routed to the keyboard input logic or decision logic as I/O entry commands. These commands control the internal program actions of the terminal similar to function commands entered at the keyboard. Displayable characters are entered into memory and displayed on the terminal screen. When a received character is displayed at the last character position (X=73, Y=26) on the screen, a delete/line-roll/up function is performed and the next character received is displayed at the first character position on the last line.

When the I/O Processor is in the Local mode, all external communication with the terminal is inhibited and the keyboard is in full control of all terminal operations. When the terminal is to be operated in either the Half or Full Duplex-Standard Transmission modes, the I/O Processor is placed in the Receive mode by pressing either the RECEIVE or RESET pushbuttons on the keyboard. Data entered at the keyboard is then transmitted to the CPU, via the I/O Interface, and data received from the CPU (Full Duplex) or from the keyboard (Half Duplex) is processed by the I/O Processor and inserted into memory for display on the terminal screen. When Printing in the Standard Transmission modes, each character is gated to the Printer as it is received and inserted into memory.

The I/O Processor circuits are discussed in detail in the following paragraphs. Because the internal I/O register clock is utilized in all I/O Processor modes, excluding the Local mode, the operation of the I/O Register Clock circuit is given first, followed by the descriptions of the I/O Processor circuits for the Transmit, Print, Receive, and Local modes.

3.6.8.1.1 I/O Register Clock Circuit. The I/O Register Clock circuit (figure 3-17) generates the internal I/O register clock which is used to shift input/output data between the terminal and the CPU, Printer (serial), and Tape Cassette. Baud rates of 110, 300, 1200, 2400, and 9600 may be selected on the front panel BAUD switch. The I/O Register Clock circuit is turned on and off by the CLOCK ENABLE (HD) signal generated within the I/O Process.

When the CLOCK ENABLE (HD) input is low, gate B18-B3A is inhibited and flip-flop B18-C4D is held reset. Thus, the 12-bit counter, consisting of 4-bit binary counters B18-B3C, -B3D, and -B4B, remains at a count of zero from the previous cycle and the I/O REG CLOCK (INTERNAL) output is held low. When data is ready to be shifted in or out of the terminal, the CLOCK ENABLE (HD) signal goes high. This enables gate B18-B3A and releases the output flip-flop B18-C4D. The 1.4175 MHz t0.3 clock pulses to flip-flop B18-B2A are divided by 2 to produce a 708.75 kHz squarewave at the other input to B18-B3A. With the gate enabled, the squarewave is applied to the 12-bit counter which begins to count up at a 708.75 kHz rate. Counts 37, 148, 296, 1184, and 3216 of the counter are then decoded and applied to gates B18-C3E,-C3F, -C3G, -C3H, and -C4C, respectively. Each of these gates is enabled by one of the baud rate selection inputs from the front panel Baud selector switch, S12. Thus, the appropriate decoder output is fed to the D-input of flip-flop B18-C4B, via OR gate B18-C4E/D2A.

For example, when the BAUD switch is set to 9600, the D-input to the flip-flop goes low each time the 12-bit counter reaches a count of 37. Five hundred nanoseconds after the D-input goes low, the inverted t0.1 pulse clocks the flip-flop and the Q-output goes low. This causes the output of OR gate B18-B3B to go high, resetting the 12-bit counter. Three hundred nanoseconds later, the leading edge of the inverted t0.5 pulse direct sets the flip-flop to release the counter. This operation divides the 708.75 kHz counter input by 37 down to 19.155 kHz. The 19.155 kHz, 300-nanosecond pulses at the $\overline{Q}$-output of B18-C4B are used to clock divide by 2 flip-flop B18-C4D to produce an approximate 9600 (9577) Hz squarewave at the I/O REG CLOCK (INTERNAL) output. With the exception of the output

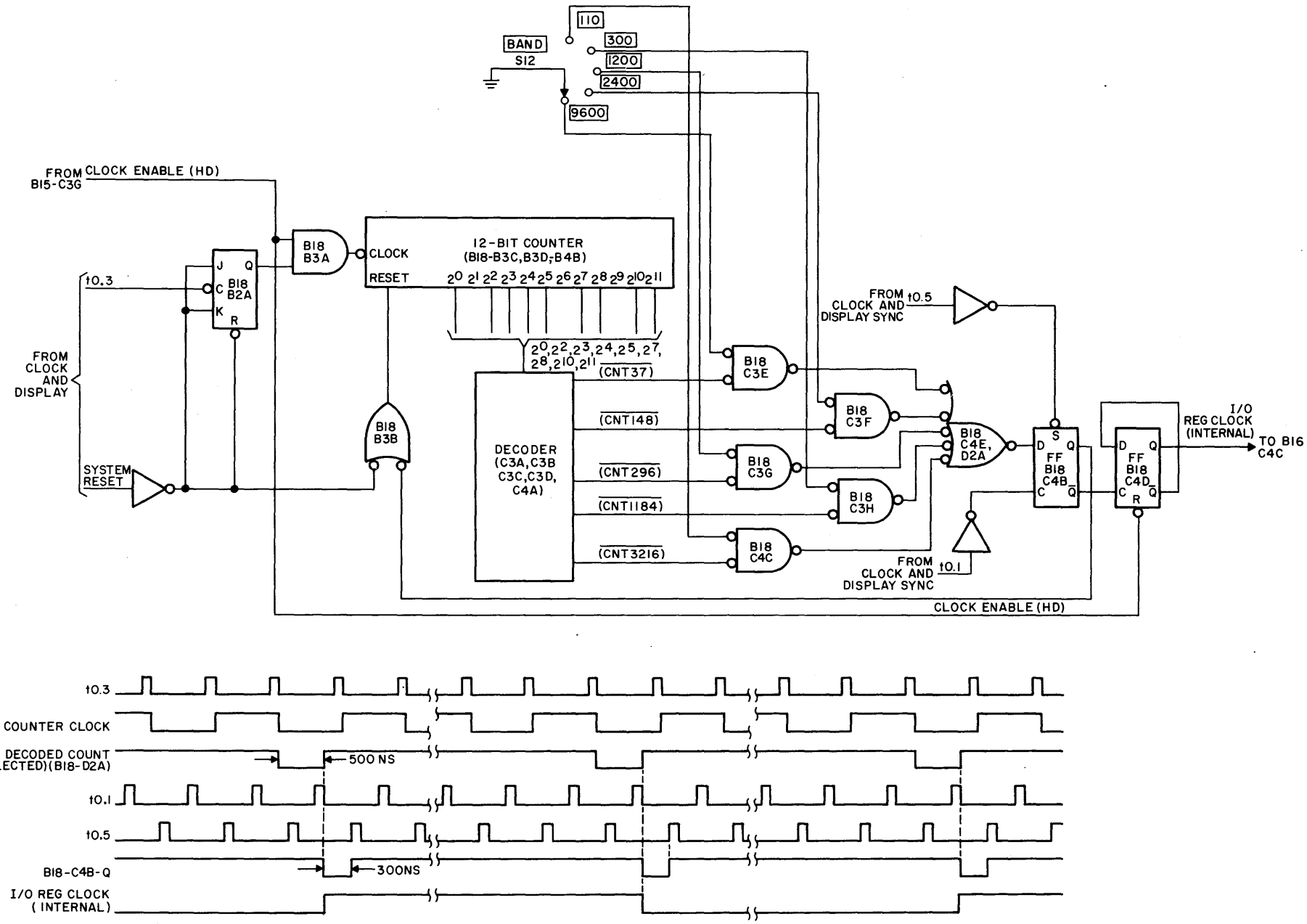


Figure 3-17. I/O Register Clock, Logic Diagram

frequency (baud rate), the operation for the 110, 300, 1200, and 2400 baud selections is identical to the operation for the 9600 baud selection. The SYSTEM RESET input resets the 12-bit counter and flip-flop E18-B2A when power is initially applied to the terminal, or whenever the RESET pushbutton switch on the keyboard is pressed.

3.6.8.1.2 Transmit (Xmit) Mode. The Xmit mode may be initiated at the keyboard or by a decoded Xmit function command from either the CPU or the Tape Cassette. When the terminal is placed in the Xmit mode, an EOT symbol is inserted into memory after the last character position of the message and the program counter initiates a search for the start of the message. When the start of the message (the previous EOT or the HOME position) has been located, a $\overline{\text{START READOUT MEM FOR XMIT}}$ signal is generated by the Decision Logic and applied to the I/O Processor to initiate the transmission of data to the CPU. Figure 3-18 illustrates the I/O Processor logic for the Xmit mode.

Before the I/O Processor circuits receive the $\overline{\text{START READOUT MEM FOR XMIT}}$ signal, the following initial conditions exist: EOT counter, comprising flip-flops A17-B2A and A17-B2B, equals zero; Clock Enable flip-flop A17-B4B is set; the CLOCK ENABLE (HD) input to the I/O Register Clock circuit is low; SHIFT LOAD ENABLE (A17-C4A) is high; and I/O bit counter A17-B3G equals zero. Because the keyboard is locked in the Xmit and Print modes, the I/O Processor must be in either the Receive or Local mode before the Xmit mode can be initiated. In the Receive mode, the Q-output of Receive flip-flop A17-D2A is high and the t0.3 pulses are gated through AND gate A17-B1G and OR gate A17-B1H to reset the EOT counter flip-flops. If the terminal was previously placed in the Local mode, pressing the LOCAL pushbutton switch on the keyboard momentarily grounded one input to OR gate A17-A2B. This causes the $\overline{\text{LOCKOUT KB OVERRIDE}}$ signal to go low which also resets the EOT counter, via OR gate A17-A2B. The remaining initial conditions are satisfied as follows. If the Clock Enable flip-flop is not set, the high CLOCK ENABLE FF input to gate B15-C3G causes the CLOCK ENABLE (HD) output to be high (the $\overline{\text{CLOCK ENABLE INH CASSETTE OR PRINTER}}$ input to X B15-C3G is high for the Local and Receive mode, except for the Receive/cassette playback mode). With the CLOCK ENABLE (HD) signal high, the I/O Register clock is enabled, and the output pulses are applied through OR gate B16-C4E and AND gate A17-A3A to the clock input of the I/O bit counter. Once a count of 10 in the Receive mode, or a count of 12 in the Local mode, is decoded, the output of OR gate A17-D3B goes high to set-enable the Clock Enable flip-flop. The flip-flop is then set by the next t0.0 pulse. The

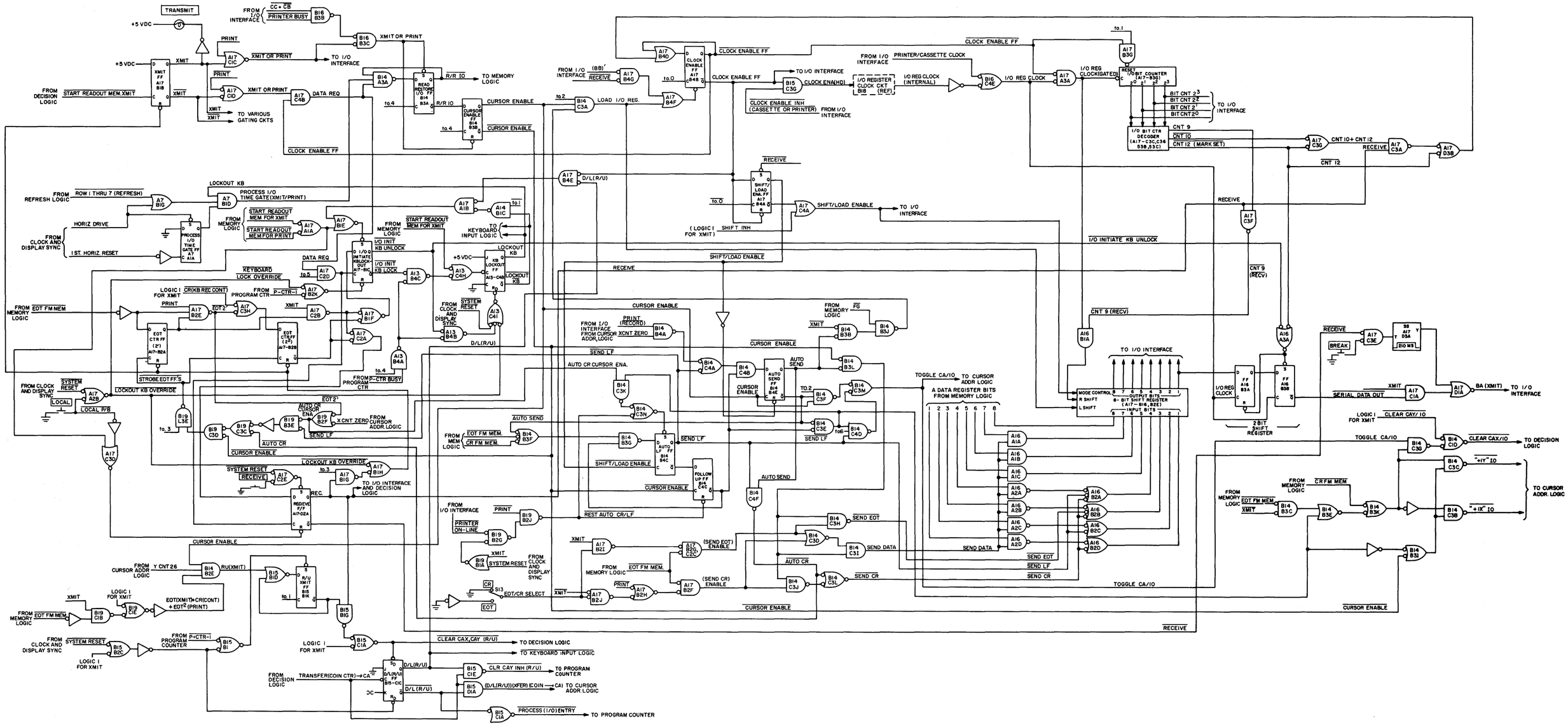


Figure 3-18. I/O Processor Logic (Xmit Mode), Logic Diagram

high $\overline{\text{CLOCK ENABLE FF}}$ output keeps the flip-flop set and enables the t0.1 pulse to reset the I/O bit counter. The low CLOCK ENABLE FF output causes the CLOCK ENABLE (HD) signal to be low and resets Shift/Load Enable flip-flop A17-B4A, causing the SHIFT LOAD ENABLE signal to go high.

When the $\overline{\text{START READOUT MEM FOR XMIT}}$ signal is generated, the 700-nanosecond, negative-going pluse is applied to Xmit flip-flop A17-B1B and OR gates A17-A1A and A13-C4H. Thus, the Xmit flip-flop is set, the output of A17-A1A sets I/O Initiate KB Lockout flip-flop A17-B1C, via OR gate A17-B1E, and resets Receive flip-flop A17-D2A, via OR gate A17-C3D, and the output of A13-C4H sets KB Lockout flip-flop A13-C4B. The LOCKOUT KB output of A13-C4B is routed to the Keyboard Input Logic to electronically disable the keyboard during the Xmit mode of operation. Simultaneously, the low $\overline{\text{XMIT}}$ signal from the Xmit flip-flop causes the DATA REQ signal at the output of AND gate A17-C4B to go high. With Read Restore flip-flop B14-B3A previously set by the XMIT OR PRINT Signal, prior to the initiation of the Xmit mode, two of three inputs to AND gate B14-A3A are now high. The third input to the gate, PROCESS I/O TIME GATE (XMIT/PRINT), is developed by OR gate A7-B1G and Process I/O Time Gate flip-flop A7-A1A.

The PROCESS I/O TIME GATE (XMIT/PRINT) signal determines the periods when the I/O Processor may read data from the memory. For the horizontal scans corresponding to rows 8 through 11 of each character line, the $\overline{\text{ROW 1 THRU 7 (REFRESH)}}$ input to A7-B1G is high, and the PROCESS I/O TIME GATE (XMIT/PRINT) signal follows the Q-output of flip-flop A7-A1A. See figure 3-19. Flip-flop A7-A1A is set by the trailing edge of each HORIZ DRIVE pulse and reset by the trailing edge of each 1ST HORIZ RESET pulse. For character rows 1 through 7 (refresh rows), the PROCESS I/O TIME RATE (XMIT/PRINT) signal is high only from the leading edge of each HORIZ DRIVE pulse to the trailing edge of each 1ST HORIZ RESET pulse. Therefore, the I/O Processor has access to the memory for the entire scans of the non-refresh rows (rows 8 through 11) and for 3.5 microseconds (during the horizontal retraces) of rows 1 through 7; during the horizontal scans of rows 1 through 7, the Refresh Logic has access to the memory to refresh the video on the display. In addition, after the last refresh row of the last character line on the display is scanned, the $\overline{\text{ROW 1 THRU 7 (REFRESH)}}$ signal goes high again until the first refresh row of the next vertical scan. Thus, the I/O Processor also has access to the memory from row 8 (or 9) of the last line of each vertical scan until row 1 (or 2) of the first line of the following vertical scan.

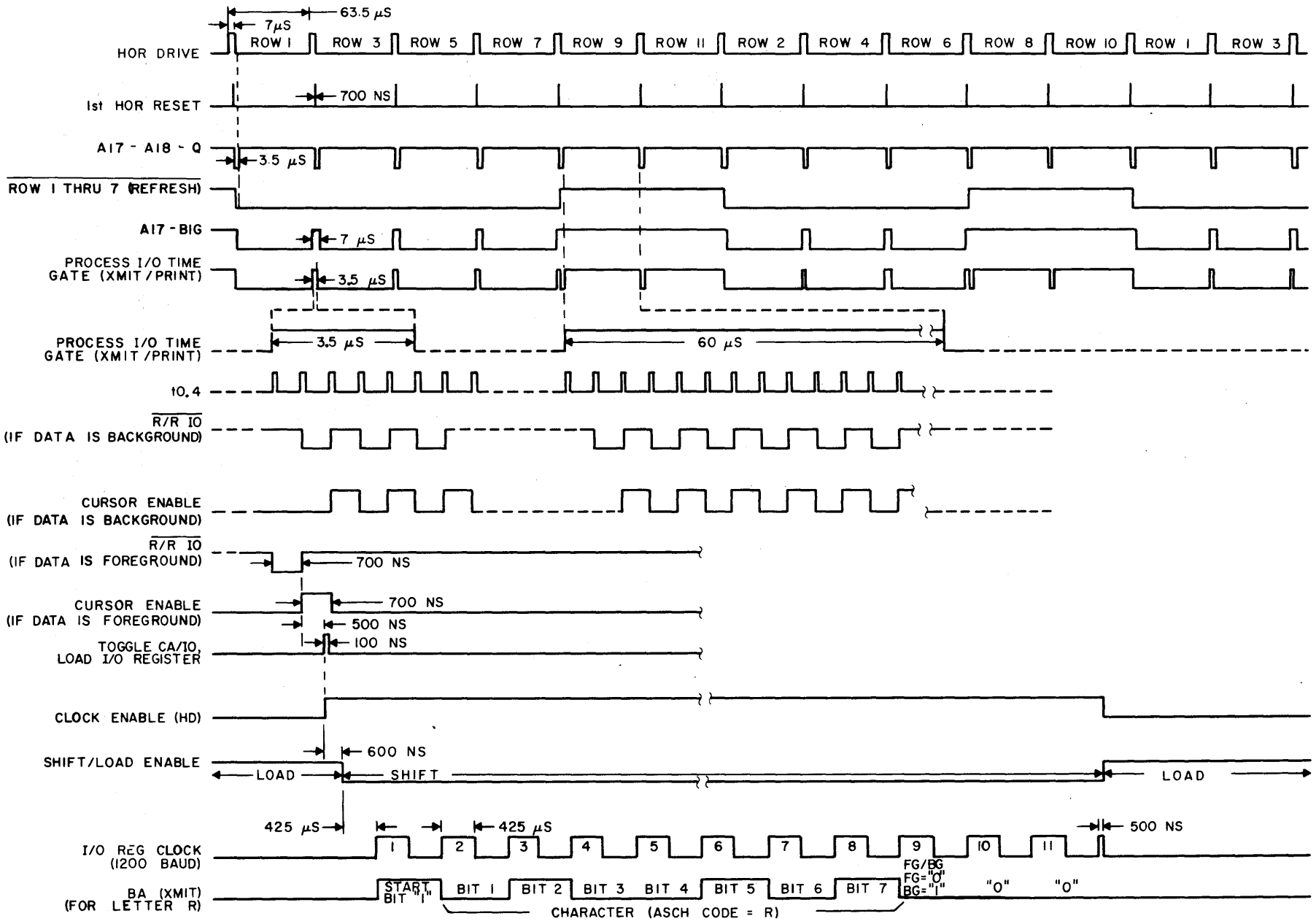


Figure 3-19. I/O Processor (Xmit Mode), Timing Diagram

When the PROCESS I/O TIME GATE (XMIT/PRINT) signal goes high, the t0.4 pulse toggles the Read Restore flip-flop to produce a 700-nanosecond $\overline{\text{R/R IO}}$ pulse. This reads the first character of the message out of memory. If the character is a background (protected) character, the $\overline{\text{FG}}$ input to gate B14-B3J from the Memory Logic will be high. For the Xmit, Printer off-line mode, the other input to the gate is also high. Thus, the output of the gate is low, and subsequently, the LOAD I/O REG signal remains low, $\overline{\text{CLOCK ENABLE FF}}$, remains high, and DATA REQ remains high. This keeps gate B14-A3A enabled and a $\overline{\text{R/R IO}}$ pulse is again initiated by the next t0.4 pulse. However, immediately following the first $\overline{\text{R/R IO}}$ pulse a CURSOR ENABLE pulse was generated by Cursor Enable F/F B14-B3H (figure 3-19). The negative-going $\overline{\text{CURSOR ENABLE}}$ pulse is inverted by OR gate B14-B3I and applied to AND gate B14-C3B. Assuming the first character of a message is not an EOT or carriage return (CR) symbol, the other input to B14-C3B will also be high. This applies a $+\overline{\text{IX IO}}$ enable signal to the Cursor Address Logic. Simultaneously, the positive-going CURSOR ENABLE pulse enables AND gate B14-C3F and the t0.2 pulse is gated through B14-C3F and OR gate B14-C3M to produce a TOGGLE CA/IO pulse for the Cursor Address Logic. With a $+\overline{\text{IX IO}}$ applied to the Cursor Address Logic, the TOGGLE CA/IO causes the cursor to advance to the next character position. When the next $\overline{\text{R/R IO}}$ pulse is generated, the second character of the message is read from the memory. Therefore, whenever a background character is read from memory, the character is not transmitted to the CPU, and the I/O Processor immediately addresses the next character of the message. However, when a foreground (unprotected) character is read from memory, the $\overline{\text{FG}}$ input to B14-B3J goes low, forcing the output high to enable gate B14-C3A. Subsequently, when the CURSOR ENABLE and t0.2 inputs are simultaneously high, the LOAD I/O REG signal goes high. At the same time, the negative-going $\overline{\text{CURSOR ENABLE}}$ pulse is gated through AND gates B14-B3L and B14-C3I to produce a positive-going SEND DATA pulse. This gates bits 1 through 7 of the character to the parallel inputs of 8-bit shift register A17-B1C, -B2E. The 8th bit contains foreground/background data and is applied directly to the register. This bit is routed to the Printer only and is substituted by a parity bit in the I/O Interface before being shifted to the CPU. The 100-nanosecond LOAD I/O REG pulse then loads the data into the register. The parallel outputs of the register are applied to the I/O Interface where they are gated to the Printer for Printer on-line operation.

The LOAD I/O REG pulse also resets the Clock Enable flip flop, via OR gate A17-B4F. Thus, $\overline{\text{CLOCK ENABLE FF}}$ goes low, to enable AND gate A17-A3A, release the I/O bit counter, and remove the DATA REQ signal from B14-A3A. With DATA REQ low, the

$\overline{\text{R/R IO}}$ pulses are inhibited until the present character is shifted out to the CPU. With CLOCK ENABLE FF now high, the CLOCK ENABLE (HD) signal goes high to enable the I/O Register clock and Shift/Load enable flip-flop A17-B4A is set by the next $\overline{\text{t0.0}}$ pulse. The SHIFT/LOAD ENABLE input to the 8-bit register then goes low (shift) and the I/O REG CLOCK (INTERNAL) is applied through OR gate B16-C4E to flip-flops A16-B3A and A16-B3B, and further gated through gates A17-A3A and A16-B1A to the right shift input of the register. This shifts the character word out of the register and flip-flops A16-B3A and A16-B3B through AND gate A17-C1A and OR gate A17-D1A to the BA(XMIT) input of the I/O Interface. Since flip-flop A16-B3A is reset by each decoded count 12 of the I/O bit counter, the first bit (start bit) of each 11-bit character word is always a logic "0" bit (figure 3-19). The start bit is then followed by the 7 character code bits, the foreground/background identification bit, and two logic "1" bits.

NOTE: The output signal from the register is data, so logic "1" = 0 Volt, and logic "0" = +4V.

As each character word is shifted out of the register, the I/O REG CLOCK (GATED) output of gate A17-A3A also advances the I/O bit counter. Subsequently, when the I/O bit counter decoder decodes a count of 12, the $\overline{\text{CNT 12}}$ output goes low. This resets flip-flop A16-B3A and sets flip-flop A16-B3B, via OR gate A16-A3A, and set enables the Clock Enable flip-flop, via OR gates A17-D3B and A17-B4D. The Clock Enable flip-flop is then set by the next t0.0 pulse. Thus, the $\overline{\text{CLOCK ENABLE FF}}$ output goes high and the CLOCK ENABLE FF output goes low. $\overline{\text{CLOCK ENABLE FF}}$ inhibits gate A17-A3A, resets the I/O bit counter, and initiates another DATA REQ; CLOCK ENABLE FF removes the CLOCK ENABLE (HD) input from the I/O Register Clock and forces the SHIFT/LOAD ENABLE signal high (load). The I/O Processor is then ready to process the next character.

When a carriage return (CR) code is read from memory, it is gated into the register and shifted out to the CPU the same as any foreground character. However, instead of advancing the cursor one space to the right, the cursor must be sent to first character position (X=0) of the next line. Whenever a CR code is read from memory, the Memory Logic applies a $\overline{\text{CR FM MEM}}$ signal to OR gate B14-B3K. This inhibits gate B14-C3B and enables gates B14-C3C and B14-C3G. Thus, when the CURSOR ENABLE pulse is generated (figure 3-19), the $\overline{\text{CURSOR ENABLE}}$ pulse initiates a $+\overline{\text{1Y IO}}$ and the TOGGLE CA/IO pulse is gated through B14-C3G to produce a $\overline{\text{CLEAR CAX/IO}}$. These pulses are applied to the Cursor Address Logic to clear the CAX counter and advance the CAY counter one count.

When the EOT code is read from memory, indicating the end of the message, the following operations occur: an EOT or CR code (depending on the EOT/CR switch setting) is transmitted to the CPU; the cursor is positioned at X=0 of the next line; the keyboard is unlocked; the Xmit flip-flop is reset; and the Receive flip-flop is set. The terminal is now ready to receive and process a response from the CPU. However, if the EOT symbol is on the last line (Y=26), a delete/line-roll/up function is also performed. This deletes the top line, rolls all the remaining lines up one line, and inserts all blanks in the bottom line. The cursor ends up at the first position of the last line, ready to receive and display data from the CPU.

When the $\overline{\text{R/R IO}}$ pulse addresses the memory and an EOT symbol is decoded, an $\overline{\text{EOT FM MEM}}$ signal is applied to EOT CTR flip-flop A17-B2A and AND gates B19-C1B, A17-B2G,-C2C, A17-B2F, and B14-B3C. When the CURSOR ENABLE pulse occurs, the low output of B14-B3C enables $+\overline{\text{1Y IO}}$ and $\overline{\text{CLEAR CAX IO}}$ cursor commands to position the cursor at X=0 of the next line. The $\overline{\text{CURSOR ENABLE}}$ pulse also enables AND gate B19-C3E, via B19-C3D, and the t0.3 pulse toggles the EOT CTR flip-flop. The Q-output of the flip-flop then goes high, causing the output of AND gate A17-C2B to go low. This reset-enables the I/O Initiate KB Lockout flip-flop and set-enables the Receive flip-flop, via OR gates A17-B1F and A17-C2A, respectively. Gates A17-B2G, -C2C and A17-B2F generate either a (SEND EOT) ENABLE or a (SEND CR) ENABLE signal, respectively depending on the position of the EOT/CR select switch. If the switch is in the EOT position, A17-B2F is inhibited, and A17-B2G, -C2C is enabled. The (SEND EOT) ENABLE signal is then gated through AND gate B14-C3H at CURSOR ENABLE to enable OR gate A16-B2B. Thus, the bit 3 input to the shift register is high and the remaining 6 character bits are low. Gates A16-A1A through A16-A2D are inhibited high (SEND EOT) ENABLE input to OR gate B14-C3D. When LOAD I/O REG occurs, the EOT code is loaded into the register and subsequently shifted out to the CPU. After the word is shifted out and the DATA REQ signal goes high again, AND gate A17-C2D is enabled and the t0.5 pulse resets the I/O Initiate KB Lockout flip-flop and sets the Receive flip-flop. The $\overline{\text{I/O INITIATE KB UNLOCK}}$ signal resets the KB Lockout flip-flop, via A13-B4B and A13-C4I, to unlock the Keyboard; the RECEIVE output of the Receive flip-flop resets the EOT and Xmit flip-flops, via A17-B1G and A17-B1H.

The $\overline{\text{EOT FM MEM}}$ input to AND gate B19-C1B causes one input to AND gate B14-B2E to go high. Therefore, if the cursor is on the last line, the Y CNT 26 input will be high and the CURSOR ENABLE pulse causes R/U Xmit flip-flop B15-B1K to be reset at

t0.1. When the RECEIVE signal goes high, a $\overline{\text{CLEAR CAX, CAY (R/U)}}$ signal is generated at the output of OR gate B15-C1A and applied to the Decision Logic to reset the cursor address X and Y counters. This places the cursor at the HOME position. Simultaneously, D/L (R/U) flip-flop B15-C1C is set. The D/L (R/U) signal is routed to the Program Counter, via an OR gate in the Keyboard Input Logic, to enable a delete line function. At the same time, a $\overline{\text{PROGRESS (I/O) ENTRY}}$ signal from OR gate B15-C1F is applied to the Program Counter to initiate the function. Since the cursor is at the HOME position, the top line is deleted and all other lines are moved up one line. At the end of a normal delete line function, when the cursor is at X=73, Y=26, the CAX and CAY counters are first cleared and then the cursor X and Y addresses from the coincidence counter are preset back into the CAX and CAY counters. This places the cursor back at the location from which the function was initiated, which in this case would be the HOME position. However, for the delete/line-roll/up function, the cursor must end up at X=0 of the bottom line. This is accomplished by the $\overline{\text{CLR CAY INH (R/U)}}$ output of AND gate B15-C1E and the $\overline{\text{D/L (R/U)}}$ output of flip-flop B15-C1C. The $\overline{\text{CLR CAY INH (R/U)}}$ is applied to the Program Counter to inhibit the CLEAR CAY CTR output to the Cursor Address Logic and the $\overline{\text{D/L (R/U)}}$ signal inhibits the $(\overline{\text{D/L (R/U)}})$(XFER COIN→CA) output to the Cursor Address Logic. Thus, when the cursor is at the X=73, Y=26 position, the CAX counter is cleared and the cursor address (X=0) in the coincidence X counter is transfered to the CAX counter; however, the address in the CAY counter remains at Y=26. R/U Xmit flip-flop B15-B1K is set by P-CTR-1 during the "roll up" operation and D/L (R/U) flip-flop B15-C1C is reset by the TRANSFER (COIN CTR) --CA pulse at the end of "roll up".

As previously mentioned, when the Printer is on-line during the Xmit mode, the I/O Processor automatically follows each carriage return with a line feed code. This operation is performed by an auto-send state counter, consisting of Auto Send flip-flop B14-C4E, Auto LF flip-flop B14-B4C, Follow Up flip-flop B14-C4C, and their associated logic circuitry. During the normal Xmit mode (Printer not on line), flip-flop B14-B4C is held set and flip-flop B14-C4C is held reset by the low $\overline{\text{RESET AUTO CR/LF}}$ signal from AND gate B19-B2J. Flip-flop B14-C4E also remains reset because of the high inputs to OR gate B14-C4A. However, when the Printer is on-line, the output of AND gate B19-B2G goes low, and subsequently, the $\overline{\text{RESET AUTO CR/LF}}$ signal goes high to release flip-flops B14-C4C and B14-B4C. Therefore, when a $\overline{\text{CR FM MEM}}$ signal is applied to OR gate B14-B3F, the counter is set into operation. Since the operation of the auto-send state counter for the Xmit-Printer

on-line mode is similar to the operation for the Print mode, described under paragraph 3.6.8.1.3, only the sequence of operation for the Xmit Printer on-line mode will be described at this time. When a CR code is read from memory, the SEND DATA signal goes high at CURSOR ENABLE and the CR code (data) is inserted into the shift register. At the same time, +$\overline{\text{1Y IO}}$ and $\overline{\text{CLEAR CAX/IO}}$ signals are applied to the Cursor Address Logic to position the cursor at the first character position on the next line. When the SHIFT/LOAD ENABLE signals goes low to shift the contents of the register to the CPU, $\overline{\text{SHIFT/LOAD ENABLE}}$ goes high to reset flip-flop B14-B4C. Thus, $\overline{\text{SEND LF}}$ goes low to enable gate B14-C4B, via OR gate B14-C4A, and to apply the LF ASCII code to the shift register, via gates A16-B2A and A16-B2C. When CURSOR ENABLE goes high again, flip-flop B14-C4E is set and the LF code is gated into the register. The TOGGLE CA/IO is then inhibited by the low $\overline{Q}$-output of flip-flop B14-C4E to gate B14-C3F to prevent the cursor from advancing when the LF code is being sent out. At the trailing edge of CURSOR ENABLE, flip-flop B14-C4C is set and it's $\overline{Q}$-output goes low to inhibit gates B14-C4D and B14-B3G. When $\overline{\text{SHIFT/LOAD ENABLE}}$ goes high again (shift), flip-flop B14-C4E is reset, flip-flop B14-B4C is set, and the LF code is shifted out of the register. During the next CURSOR ENABLE, the character at X=0 is loaded into the register and the cursor advances to the next position (X=1). At the trailing edge of CURSOR ENABLE, flip flop B14-C4C is reset to return the counter to its static state.

A summary of the basic operations of the I/O Processor for the Xmit mode is given in sequence in the following steps:

Step 1. Initial conditions: CLOCK ENABLE (HD) low; SHIFT/LOAD ENABLE high (load); I/O bit counter =0; and EOT counter = 0.

Step 2. When $\overline{\text{START READOUT MEM FOR XMIT}}$ occurs; reset Receive flip flop; set Xmit flip-flop; $\overline{\text{I/O INIT KB LOCK}}$ low; and LOCKOUT KB high.

Step 3. DATA REQ high.

Step 4. If EOT counter $\neq$ 0: $\overline{\text{I/O INIT KB UNLOCK}}$ low; set Receive flip flop; and reset Xmit flip-flop and EOT counter. If R/U Xmit flip-flop reset: set D/L (R/U) flip flop; $\overline{\text{PROCESS I/O ENTRY}}$ low. (Reset D/L (R/U) flip-flop when XFER COIN CTR--CA occurs at end of "roll up".)

Step 5. When PROCESS I/O TIME GATE (XMIT OR PRINT) occurs; $\overline{\text{R/R IO}}$ low for 700 nanoseconds.

Step 6. If $\overline{\text{FG}}$ (background character), go back to step 5.

Step 7. If EOT: $\overline{\text{SEND EOT}}$ or $\overline{\text{SEND CR}}$ pulse; $\overline{+1\text{Y IO}}$ pulse; $\overline{\text{CLEAR CAX/IO}}$ pulse; TOGGLE CA/IO pulse; and increment EOT counter. If EOT and Y=26, reset R/U (Xmit) flip-flop. Proceed to step 10.

Step 8. If CR: $\overline{+1\text{Y IO}}$ pulse; $\overline{\text{CLEAR CAX/IO}}$ pulse; TOGGLE CA/IO pulse; and SEND DATA pulse. Proceed to step 10.

Step 9. SEND DATA pulse; $\overline{+1\text{X IO}}$ pulse; and $\overline{\text{TOGGLE CA/IO}}$ pulse.

Step 10. LOAD I/O REG pulse; DATA REQ low; reset Clock Enable flip-flop; CLOCK ENABLE (HD) high; set Shift/Load Enable flip-flop; SHIFT/LOAD ENABLE low (shift); enable bit counter; and start I/O REG CLOCK.

Step 11. When bit counter reaches a count of 12: set Clock Enable flip-flop; CLOCK ENABLE (HD) low; reset Shift/Load Enable flip-flop; SHIFT/LOAD ENABLE high (load); and reset bit counter. Go back to step 3.

3.6.8.1.3 Print Mode. The Print mode may also be initiated from either the keyboard or by a decoded Print function command from the CPU or Tape Cassette. When the Print mode is initiated, two adjacent EOT symbols are inserted into memory after the last character position of the message and a search is initiated for the start of the message. When the start of the message (the previous two adjacent EOT's or the HOME position) is located, a $\overline{\text{START READOUT MEM FOR PRINT}}$ signal is applied to the I/O Processor Logic to initiate the Print operation. Figure 3-20 illustrates the I/O Processor Logic for the Print mode. Because the initial conditions and basic operations for the Print mode are identical to those for the Xmit mode previously described, only the differences in the operations for the Print mode, along with the operation of the auto-send state counter, are described in detail in the following paragraphs.

When the $\overline{\text{START READOUT MEM FOR PRINT}}$ signal is applied to the I/O Processor Logic, the Print, I/O Initiate KB Lockout, and KB Lockout flip-flops are set, and the Receive flip-flop is reset. Thus, the keyboard is disabled and the DATA REQ signal goes high to enable the Read Restore flip-flop input AND gate. The $\overline{\text{R/R IO}}$ and CURSOR ENABLE pulses are then generated during the PROCESS I/O TIME GATE (XMIT/PRINT) periods to read each character from memory and to advance the cursor to each next character position. However, when data is being sent to the Printer, the Printer sends back a PRINTER BUSY during the time intervals (approximately 30 milliseconds) it takes the Printer to process and print each character. The inverted $\overline{\text{PRINTER BUSY}}$ signal is, in turn, applied to AND gate B16-B3C to inhibit

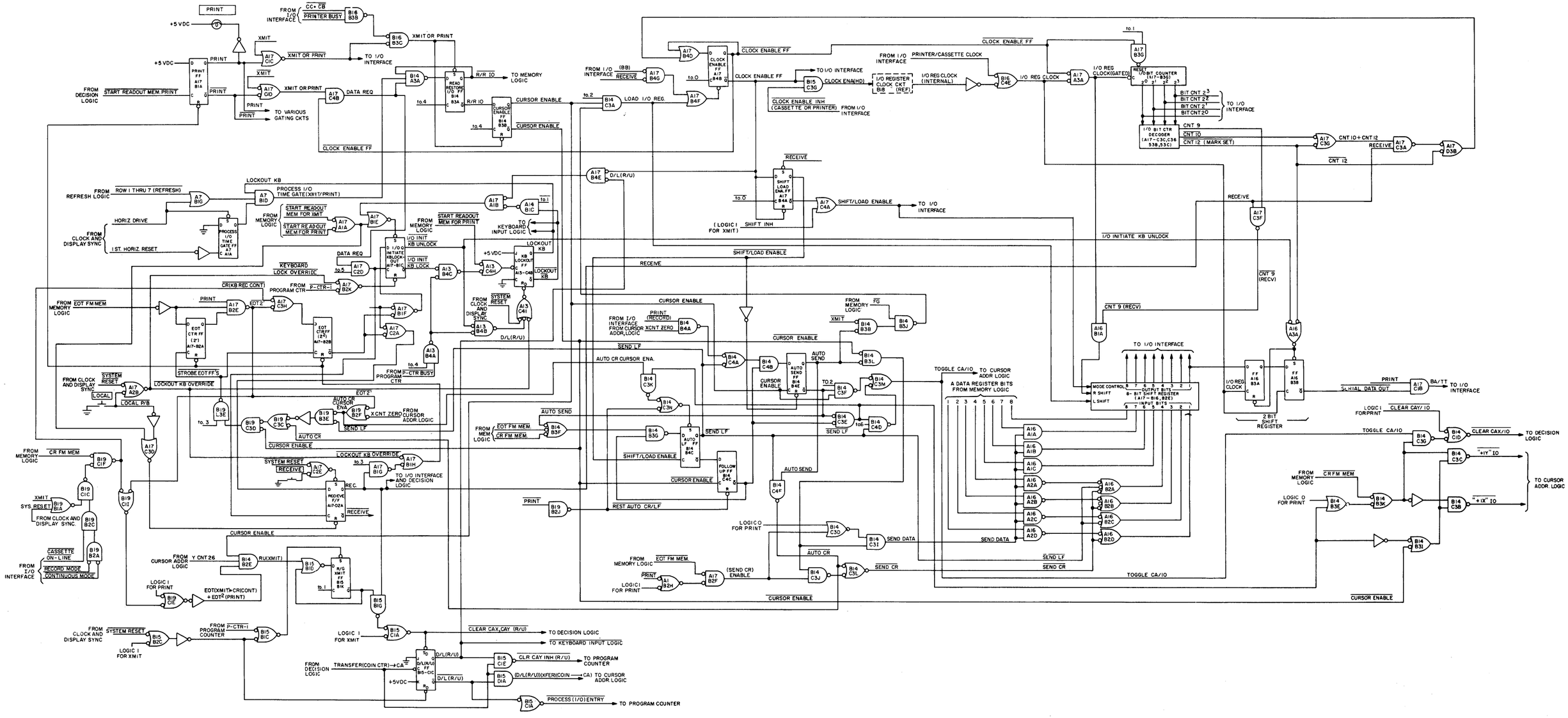


Figure 3-20. I/O Processor Logic (Print Mode), Logic Diagram

the $\overline{\text{R/R IO}}$ and CURSOR ENABLE pulses until the Printer is ready to accept the next character. When data is being sent to the Tape Cassette, and not the Printer, the $\overline{\text{PRINTER BUSY}}$ signal remains high. In addition, contrary to the Xmit mode, in which only foreground characters are sent to the CPU, both foreground and background characters are sent to the Printer or Tape Cassette during the Print mode. With the $\overline{\text{XMIT}}$ input to AND gate B14-B3B high, the output of AND gate B14-B3J remains high, and a LOAD I/O REG pulse is generated at t0.2 for each CURSOR ENABLE pulse. When each LOAD I/O REG pulse is generated, the data at the input to the shift register is loaded into the register where it is applied to the I/O Interface in parallel form for the parallel Printer and subsequently shifted out to the I/O Interface for either the Tape Cassette or a serial Printer. Each LOAD I/O REG pulse also resets the Clock Enable flip-flop to initiate the shift operation. However, data may be shifted to the Tape Cassette or serial Printer by the I/O REG CLOCK (INTERNAL), at the selected baud rate, or by an external clock input from the Tape Cassette or Printer. When the internal clock is used, the CLOCK ENABLE (HD) signal goes high to enable the I/O Register Clock circuit. When an external clock is used, the $\overline{\text{CLOCK ENABLE INHIBIT (CASSETTE OR PRINTER)}}$ input to AND gate B15-C3G goes low to inhibit the CLOCK ENABLE(HD) signal and the external clock is routed through the I/O Interface to the PRINTER/CASSETTE CLOCK input of OR gate B16-C4E. Regardless of the clock selected, when the bit counter reaches a count of 12, the Clock Enable flip-flop is set again. This inhibits AND gate A17-A3A, resets the I/O bit counter, and restores the DATA REQ signal. When the next $\overline{\text{R/R IO}}$ pulse occurs, the next character is read from memory and the process is repeated.

If the Tape Cassette is not in the Record mode, the I/O Processor sends out a CR code, followed by a LF code, at the end of each character line to signal the Printer to do a carriage return and line feed before it receives the first character of the following line. The sequence of this operation is shown in figure 3-21. When the first $\overline{\text{R/R IO}}$ pulse on the diagram occurs, the cursor is at X=72 and the next to last character on the line is read from memory. When CURSOR ENABLE occurs, the AUTO SEND signal from the Auto Send flip-flop is low to enable AND gate B14-B3L. The $\overline{\text{CURSOR ENABLE}}$ pulse is then applied through gate B14-B3L to the input of AND gate B14-C3I. If the character is not an EOT, the other input to the gate will be high and a SEND DATA pulse is initiated. This enables AND gates A16-A1A through A16-A2D and the character bits from the Memory Logic are loaded into the shift register by the LOAD I/O REG pulse. Simultaneously, the

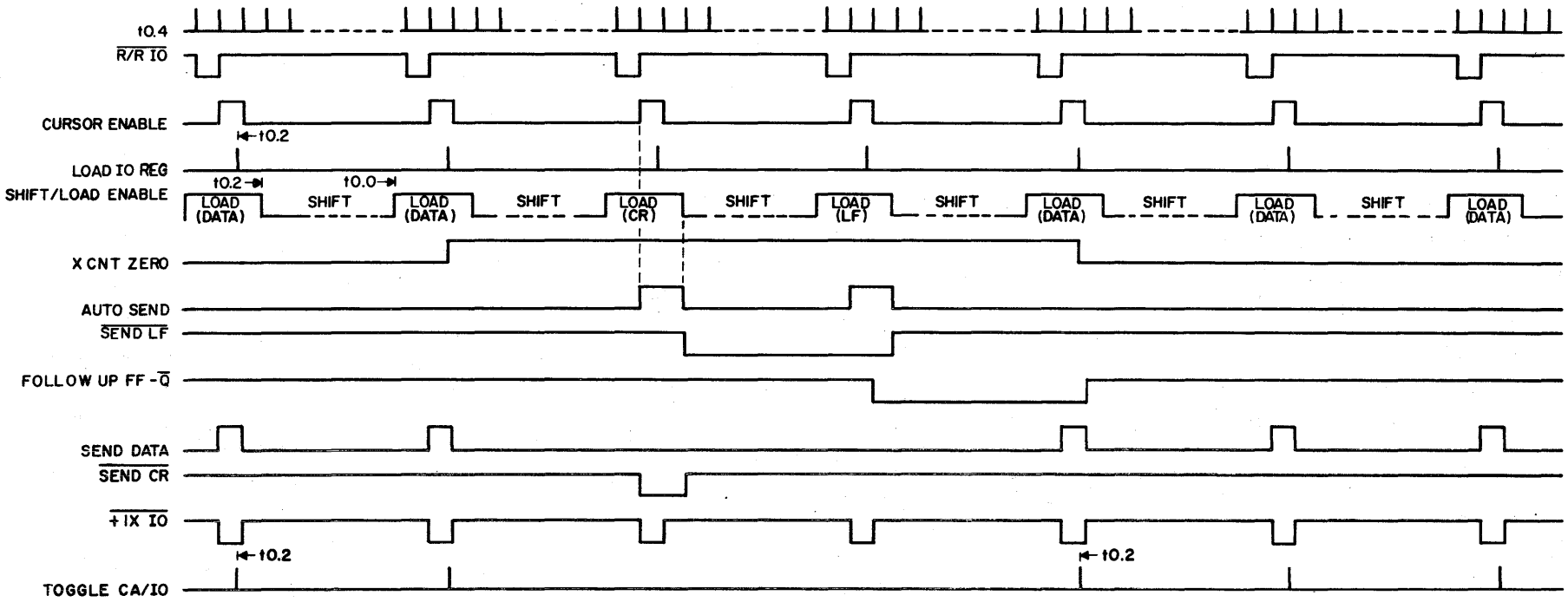


Figure 3-21. Auto-Send State Counter (Print, $\overline{\text{Record}}$, X-CNT=0), Timing Diagram

high $\overline{Q}$-output of the Auto Send flip-flop enables gate B14-C3F and a TOGGLE CA/IO pulse is generated at t0.2. Since the $\overline{+1X\ IO}$ signal is low, the cursor is advanced to the X=73 position. When the SHIFT/LOAD ENABLE signal goes low, the next to last character is shifted out of the register. Since the Auto Send flip-flop is still reset when the next CURSOR ENABLE pulse occurs, the last character on the line is loaded into the register. At the same time, a TOGGLE CA/IO pulse is generated which advances the cursor to the X=0 position of the following line. This causes the X CNT ZERO input to AND gate B14-B4A. With the terminal in the Print mode and the Tape Cassette not in the Record mode, the PRINT ($\overline{RECORD}$) input to gate B14-B4A is also high. Since the $\overline{Q}$-output of the follow up flip-flop is high, enabling gate B14-C4B, the X CNT ZERO input causes the D-input to the Auto Send flip-flop to go high. After the last character on the previous line is shifted out of the register, the $\overline{R/R\ IO}$ pulse reads the character at X=0 out of the memory. However, the leading edge of the CURSOR ENABLE pulse sets the Auto Send flip-flop causing the AUTO SEND signal to go high. This forces the output of AND gate B14-B3L low to inhibit the SEND DATA pulse. Thus, the character bits from the Memory Logic are not loaded into the register. Instead, the AUTO SEND signal and the high $\overline{SEND\ LF}$ output of the Send LF flip-flop enable gate B14-C4F to generate an $\overline{AUTO\ CR}$. This is then applied through OR gate B14-C3L to produce the $\overline{SEND\ CR}$ signal. When the LOAD I/O REG pulse occurs, the CR code is loaded into the register. In addition, the low $\overline{Q}$-output of the Auto Send flip-flop inhibits gate B14-C3F to inhibit the TOGGLE CA/IO pulse and enables AND gate B14-B3G, via OR gate B14-B3F. Thus, the cursor does not advance (remains at X=0) and the Auto LF flip-flop is reset by the positive-going edge of the $\overline{SHIFT/LOAD\ ENABLE}$ signal. Although the Auto Send flip-flop is also reset by the SHIFT/LOAD ENABLE, it is again set by the leading edge of the next CURSOR ENABLE pulse. At this time, the AUTO SEND signal inhibits the SEND DATA pulse and the low $\overline{SEND\ LF}$ signal inhibits AND gate B14-C4F to inhibit the $\overline{SEND\ CR}$ signal. However, the $\overline{SEND\ LF}$ signal produces the LF code at the inputs to the register. When the LOAD I/O REG pulse occurs, the LF code is loaded into the register and subsequently shifted out. Again the TOGGLE CA/IO pulse is inhibited by the $\overline{Q}$-output of the Auto Send flip-flop and the cursor remains at X=0. With the $\overline{Q}$-output of the Auto LF flip-flop high, the Follow Up flip-flop is set by the trailing edge of the $\overline{CURSOR\ ENABLE}$ pulse. The output of the Follow Up flip-flop then goes low to inhibit AND gates B14-B3G and B14-C4B. This causes the Auto LF flip-flop to be set by the $\overline{SHIFT/LOAD\ ENABLE}$ pulse and prevents the Auto Send flip-flop from being set by the next CURSOR ENABLE pulse. Therefore, when the next CURSOR ENABLE pulse occurs, a

SEND DATA pulse is initiated to load the X=0 character bits into the register. At the same time, a TOGGLE CA/IO pulse is generated to advance the cursor to the X=1 position. At the trailing edge of the $\overline{\text{CURSOR ENABLE}}$ pulse, the Follow-Up flip-flop is reset to return the state counter to its initial state.

When a CR code is read from memory, the CR code is sent out, the cursor is advanced to X=0 of the following line, and a LF code is sent out before the X=0 character. The timing of the Auto-state counter for a CR code is illustrated in figure 3-22. Whenever the $\overline{\text{R/R IO}}$ pulse reads a CR code from memory, a $\overline{\text{CR FM MEM}}$ signal is applied to OR gates B14-B3F and B14-B3K. Since both inputs to OR gate B14-C4A are still high at the leading edge of the CURSOR ENABLE pulse, the Auto Send flip-flop remains reset. Thus, AND gate B14-B3L remains enabled and the $\overline{\text{CURSOR ENABLE}}$ pulse produces a SEND DATA signal. The CR code from the Memory Logic is then loaded into the register by the LOAD I/O REG pulse. However, the $\overline{\text{CR FM MEM}}$ input to OR gate B14-B3K produces a $\overline{+1\text{Y IO}}$ instead of a $\overline{+1\text{X IO}}$ signal at CURSOR ENABLE and also enables AND gate B14-C3G. Therefore, when the TOGGLE CA/IO pulse is generated by AND gate B14-C3F, the cursor Y address increases by one count and a $\overline{\text{CLEAR CAX/IO}}$ pulse is initiated to clear the cursor X address counter. This positions the cursor to X=0 of the following line. The $\overline{\text{CR FM MEM}}$ input to OR gate B14-B3K causes the Auto LF flip-flop to be reset by the positive-going edge of the $\overline{\text{SHIFT/LOAD ENABLE}}$ pulse. This, in turn, enables gate B14-C4B, via OR gate B14-C4A, causing the Auto Send flip-flop to be set by the leading edge of the next CURSOR ENABLE pulse. Thus, the SEND DATA signal is inhibited and the LF code is loaded into the register. With the Auto Send flip-flop set, the TOGGLE CA/IO pulse is also inhibited and the cursor remains at X=0. With the $\overline{\text{Q}}$-output of the AUTO LF flip-flop high, the FOLLOW UP flip-flop is set by the trailing edge of the $\overline{\text{CURSOR ENABLE}}$ pulse. Operation then proceeds as for the Auto CR/LF combination. Thus, during the next CURSOR ENABLE period, the X=0 character is loaded into the register and the cursor advances to the X=1 position.

When an EOT is read from memory, a CR code is sent out, the cursor advances to the next character to determine if it is also an EOT, a line feed is then sent out, and the cursor is positioned to X=0 of the next line. If two adjacent EOT's are decoded, the I/O Processor Logic is set to the Receive mode. If the character after the EOT is a blank, the character at X=0 of the next line is sent out and the cursor advances to X=1 for the next character. The timing for an EOT code is shown in figure 3-23. Whenever an EOT code is read from memory, an $\overline{\text{EOT FM MEM}}$ signal is applied to gates A17-B2F and B14-B3F, and through an inverter to EOT flip-flop

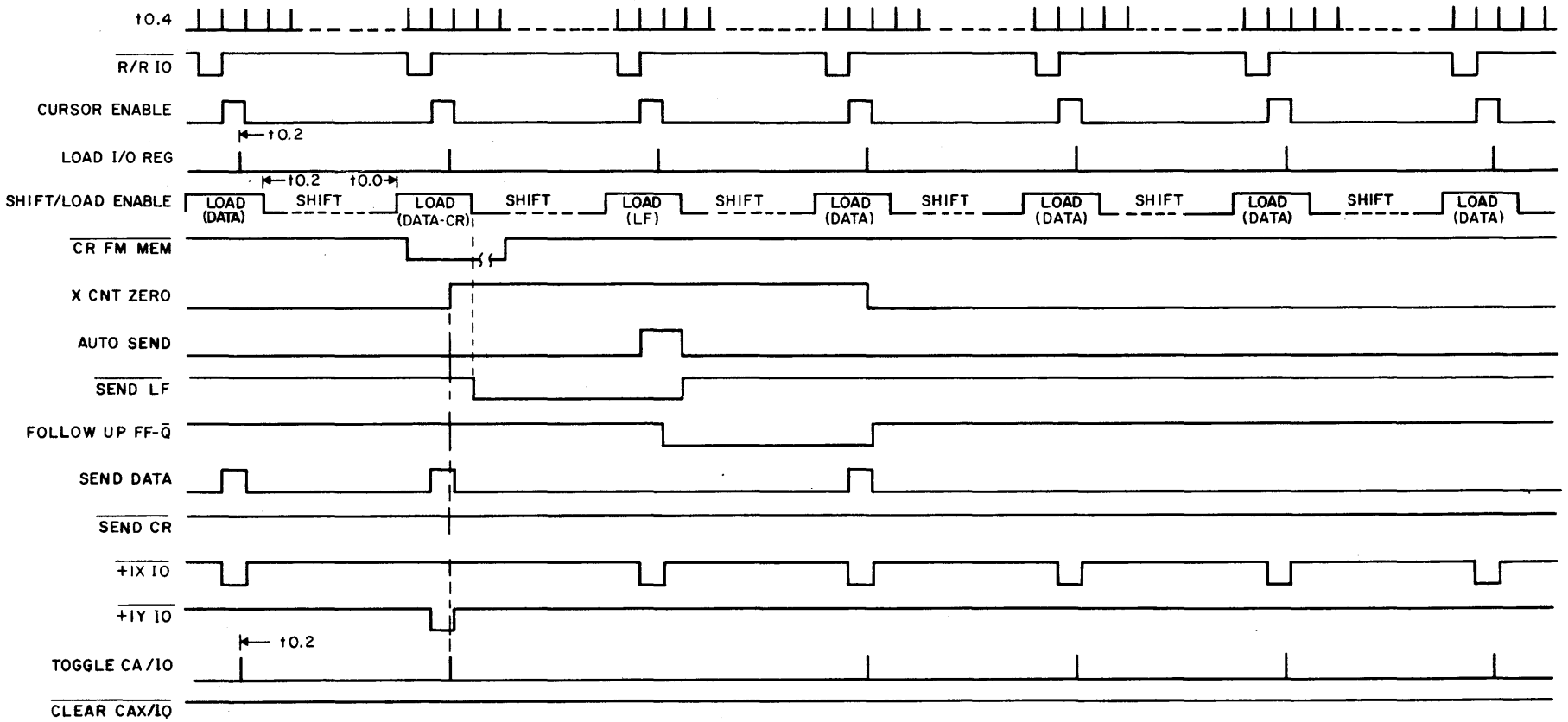


Figure 3-22. Auto-Send State Counter (CR FM MEM), Timing Diagram

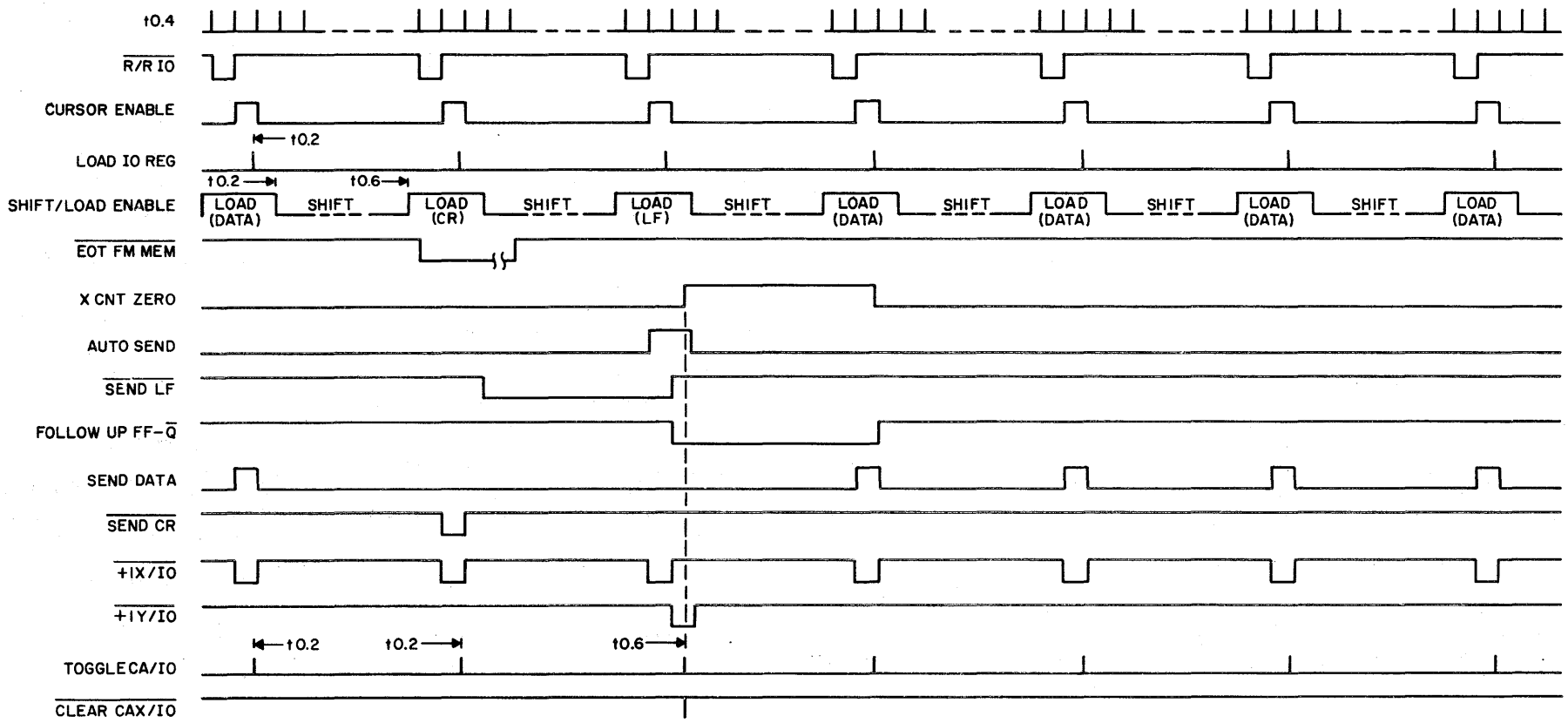


Figure 3-23. Auto-Send State Counter (EOT FM MEM), Timing Diagram

A17-B2A and gate A17 B2E. The $\overline{\text{EOT FM MEM}}$ input to gate A17-B2F produces a (SEND CR) ENABLE signal which, in turn, inhibits the SEND DATA output of B14 C3L and initiates a $\overline{\text{SEND CR}}$ pulse via OR gate B14-C3L. The CR code is then loaded into the register by the LOAD I/O REG pulse. At the same time, the $+\overline{1X\ IO}$ and TOGGLE CA/IO pulses are generated to advance the cursor to the next character position and the EOT counter is incremented to a count of one. At the positive-going edge of the $\overline{\text{SHIFT/LOAD ENABLE}}$ pulse, the Auto LF flip-flop is reset due to the $\overline{\text{EOT FM MEM}}$ input to OR gate B14-B3F. With the Auto LF flip-flop reset, the Auto Send flip-flop is set by the leading edge of the CURSOR ENABLE pulse. With the AUTO SEND signal high and the $\overline{\text{SEND LF}}$ signal low at CURSOR ENABLE, the LF code is loaded into the register by the LOAD I/O REG pulse. In addition, the $\overline{Q}$-output of the Auto Send flip-flop inhibits the TOGGLE CA/IO pulse to prevent the cursor from advancing in the x direction. When the $\overline{\text{SHIFT/LOAD ENABLE}}$ signal goes high the Follow Up flip-flop is reset. This causes the output of AND gate B14-C3E to go high, enabling AND gate B14-C3K. If the cursor is not a X=0, the low X CNT ZERO input to gate B19-B2F causes the other input to gate B14-C3K to be high. This sets the Auto LF flip-flop to enable AND gate B14-C4D. Thus, a TOGGLE CA/IO pulse is generated by gate B14-C4D at t0.6. However, the high output of gate B14-C3E also inhibits the $\not{1}\overline{\text{X IO}}$ output gate and enables the $\not{1}\overline{\text{Y IO}}$ and $\overline{\text{CLEAR CAX/IO}}$ outputs. This advances the cursor to the X=0 position of the following line. During the next CURSOR ENABLE pulse, the character at X=0 is loaded into the register and the cursor is advanced to the next position.

If the character after the first EOT symbol was also an EOT, the EOT counter would have incremented to a count of two. This causes the $\overline{Q}$-output of EOT CTR FF A17-B2B to go low to reset enable the I/O Initiate KB Lockout flip-flop and set-enable the Receive flip-flop. When the DATA REQ signal goes high again, the I/O Initiate KB Lockout is reset and the Receive flip-flop is set. The $\overline{\text{I/O INITIATE KB UNLOCK}}$ signal then resets the KB LOCKOUT flip-flop to unlock the keyboard, and the RECEIVE output of the Receive flip-flop resets the EOT counter and Print flip-flops. Gates B19-C3C, B19-B3E, and B19-B2F inhibit the $\overline{\text{STROBE EOT FF's}}$ signal whenever a auto carriage return is being performed or whenever a line feed is being initiated with the cursor at the X=0 position and the $\overline{\text{EOT 2}}$ signal is high. This ensures that a single EOT symbol does not increment the EOT counter to two when the cursor remains at the same position for the CR and LF operations.

With the Tape Cassette in the Off-Line, Record, Continuous mode, the Print mode is terminated by a single CR code. In this configuration, AND gates B19-B2A,

B19-B2C, and B19-C1C are enabled and a low enable signal is applied to AND gate B19-C1F. When a CR $\overline{\text{FM MEM}}$ signal is applied to gate B19-C1F, a $\overline{\text{CR(KB REC CONT)}}$ signal is applied to OR gate A17-C3H to enable EOT CTR FF A17-B2B. When the EOT flip-flops are strobed, the Print mode is terminated as described in the preceding paragraph.

3.6.8.1.3.7 If the Print mode is terminated on the last line of the display (Y CNT 26), an R/U (XMIT) signal is generated at the output of AND gate B14-B2E to initiate the delete-line/roll up function described in paragraph 3.6.8.1.2.8.

3.6.8.1.3.8 A summary of the basic operations of the I/O Processor Logic for the Print mode is given in sequence in the following steps:

Step 1. Initial conditions: CLOCK ENABLE (HD) low; SHIFT/LOAD ENABLE high (load); I/O bit counter=0; and EOT counter=0.

Step 2. When $\overline{\text{START READOUT MEM FOR PRINT}}$ occurs; reset Receive flip-flop;set Print flip-flop; $\overline{\text{I/O INIT KB LOCK}}$ low; and LOCKOUT KB high.

Step 3. DATA REQ high.

Step 4. If EOT counter=2; $\overline{\text{I/O INIT KB UNLOCK}}$ low; set Receive flip-flop; and reset Print flip-flop and EOT counter. If R/U Xmit flip-flop reset; set D/L (R/U) flip-flop; $\overline{\text{PROCESS I/O ENTRY}}$ low. Reset D/L (R/U) flip-flop when XFER COIN CRT-CA occurs at end of "roll up".

Step 5. If Tape Cassette in Record mode, proceed to step 9.

Step 6. If CAX=0 and LF has been sent, proceed to step 9.

Step 7. If CAX=0 and CR has been sent; $\overline{\text{SEND LF}}$ low. Proceed to step 16.

Step 8. If CAX=0 and CR has not been sent: $\overline{\text{SEND CR}}$ low. Proceed to step 16.

Step 9. When PROCESS I/O TIME GATE (XMIT OR PRINT) occurs: $\overline{\text{R/R IO}}$ low.

Step 10. If EOT and EOT counter=0; $\overline{\text{SEND CR}}$ low; $\overline{\not 1\text{X IO}}$ low; TOGGLE CA/IO pulse; and increment EOT counter. Proceed to step 16.

Step 11. If EOT and EOT counter=1; $\overline{\text{SEND LF}}$ low; $\overline{\not 1\text{Y IO}}$ low; TOGGLE CA/IO pulse, $\overline{\text{CLEAR CAX/IO}}$ pulse, and increment EOT counter. Proceed to step 16.

Step 12. If $\overline{\text{EOT}}$ and EOT counter=1: $\overline{\text{SEND LF}}$ low; $\overline{\not 1\text{Y IO}}$ low; but TOGGLE CA/IO and $\overline{\text{CLEAR CAX/IO}}$ pulse only if $\overline{\text{CAX}\neq 0}$. Reset EOT counter. Proceed to step 16.

Step 13. If $\overline{\text{EOT}}$ and EOT counter=0: reset EOT counter.

Step 14. If CR: $\overline{\not 1\text{Y IO}}$ low; TOGGLE CA/IO and $\overline{\text{CLEAR CAX/IO}}$ pulses; and SEND DATA high. If Record Off-Line Continuous: set EOT counter to 2. Proceed to step 16.

Step 15. $\overline{\not 1\text{X IO}}$ low; TOGGLE CA/IO pulse; and SEND DATA high.

Step 16. LOAD I/O REG pulse; DATA REQ low; reset Clock Enable flip-flop; CLOCK ENABLE (HD) high; set Shift/Load Enable flip-flop; SHIFT/LOAD ENABLE low (shift); enable bit counter; and start I/O REG CLOCK.

Step 17. When bit counter reaches a count of 12: set Clock Enable flip-flop; CLOCK ENABLE (HD) low; reset Shift/Load Enable flip-flop; SHIFT/LOAD ENABLE high (load); and reset bit counter. Go back to step 3.

3.6.8.1.4 Receive Mode. The I/O Processor is automatically placed in the Receive mode whenever power is first applied to the terminal or when either the Xmit or Print Mode is terminated. In addition, when the terminal is in the Xmit, Print, or Local mode, the Receive mode may be initiated by pressing the RECV pushbutton on the keyboard. When the terminal is in the Receive mode, the I/O Processor is ready to receive and process data from the CPU or Tape Cassette. Figure 3-24 illustrates the I/O Processor Logic for the Receive mode.

In the Receive mode, the $\overline{\text{RECEIVE}}$ output of the Receive flip-flop enables AND gate A17-B4G. The serial data (BB)' from the CPU or Tape Cassette is also routed to AND gate A17-B4G, via the I/O Interface. When the start bit of each word is applied to the gate, the Clock Enable flip-flop is set. The high CLOCK ENABLE FF output of the flip-flop causes the CLOCK ENABLE (HD) signal to go high and the SHIFT/LOAD ENABLE signal to go low (shift), and sets the I/O INIT KB Lockout flip-flop. The $\overline{\text{I/O INIT KB LOCK}}$ signal then resets the Lockout flip-flop to lock the keyboard. Simultaneously, the $\overline{\text{CLOCK ENABLE FF}}$ signal releases the I/O bit counter and enables AND gate A17-A3A. Thus, the I/O REG CLOCK (INTERNAL) from the I/O Register Clock circuit, or the TAPE CASSETTE clock from the I/O Interface is gated through gates B16-C4E, A17-A3A, and A16-B1A to right-shift input of the shift register. The received data at the output A17-B4G is then shifted into the shift register. When the bit counter reaches a count of 9, a $\overline{\text{CNT9 (REC)}}$ signal is produced at the output of gate A17-C3F to inhibit gate A16-B1A and interrupt the clock to the shift register. At a count of 10, the Clock Enable flip-flop is reset again and the Data Ready flip-flop is set. This indicates that the data is in the shift register and ready to be decoded or inserted into memory. At the same time, the

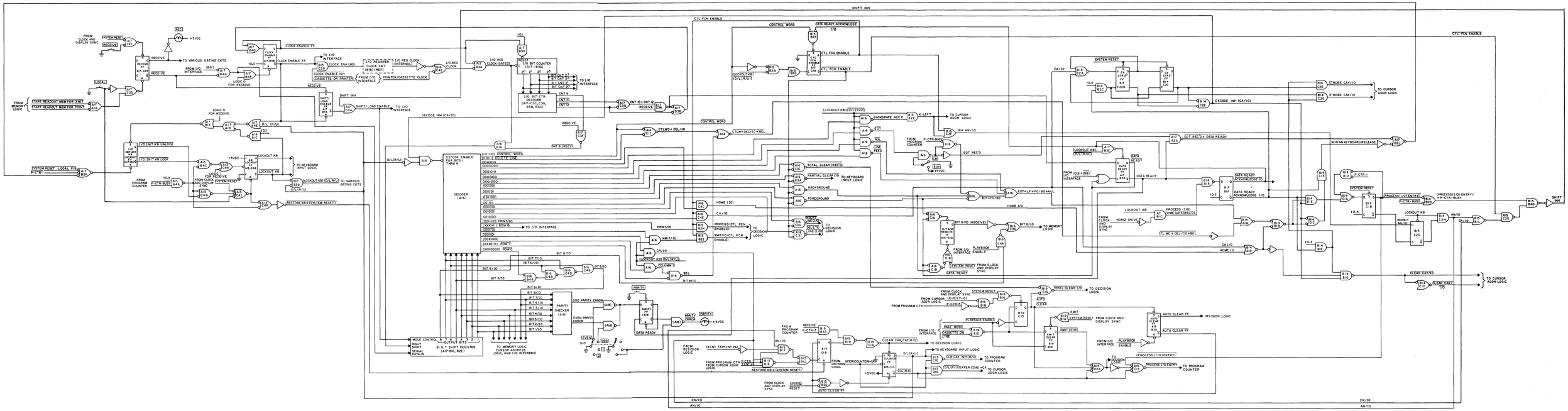


Figure 3-24. I/O Processor Logic (Receive Mode), Logic Diagram

8 bits are checked for the proper parity by parity checker A18-B4B. If a parity error exists, parity flip-flop A18-C4C is reset to light the PARITY indicator on the keyboard. If a control word code is decoded, indicating that the following word is a function command, Control Function Enable flip-flop B15-C2B is set. This enables the various function command gates on A16 and B15. The decoded control word function also initiates a NON A/N KEYBOARD RELEASE signal at the output of B14-C1E to unlock the keyboard. When the next word is applied to the (BB)' input of gate A17-B4G, the keyboard is locked again and the data is shifted into the register as described above. If the word is a function command, such as, Delete Line, Clear Screen, CA/IO, etc, the appropriate instructions are decoded and applied to the Decision Logic or Keyboard Input Logic to initiate the function. The operations for the various commands are given in the following paragraph. When an A/N character, or a function command requiring P-CTR operation, is received, a $\overline{\text{(PROCESS I/O ENTRY)}}$' signal is generated by flip-flop B14-C1B and applied through OR gate B15-C1F to the Internal Program Counter. This starts the Internal Program counter which then functions the same as it does for a keyboard initiated A/N entry or editing operation. After the operation is completed by the Internal Program Counter, the keyboard is again unlocked and the I/O Processor is ready to receive the next data word. Whenever a function command is received which does not require P-CTR operation (Set FG), a NON A/N KEYBOARD RELEASE signal is generated to unlock the keyboard. In addition, if a carriage return code is received on line 26, or an A/N character is received for X=73, Y=26, flip-flop B15-C1C is set to initiate a delete/line-roll/up function. This operation is the same as the delete/line-roll/up operation for the Xmit mode described in paragraph 3.6.8.1.2.8. The function of gates B16-B4A, B16-C4D, B16-C4B, and B16-C4G is to alter bit 6 as required to change all lower case letter ASCII codes to upper case letter ASCII codes for the Memory Logic.

The operations of the I/O Processor Logic for the Receive mode are outlined in sequence in the following steps:

Step 1. Initial conditions: CLOCK ENABLE (HD) low; SHIFT/LOAD ENABLE high (load); I/O bit counter=0; Receive flip-flop set; and LOCKOUT KB low (keyboard unlocked).

Step 2. When "start bit" received: $\overline{\text{I/O INIT KB LOCK}}$ low; LOCKOUT KB high; CLOCK ENABLE (HD) high; SHIFT/LOAD ENABLE low (shift); enable bit counter; and start I/O REG CLOCK.

Step 3. When bit counter reaches a count of 9: $\overline{\text{CT 9 (REC)}}$ low to inhibit clock to shift register.

Step 4. When bit counter reaches a count of 10: DATA READY high; SHIFT/LOAD ENABLE high (10ad); CLOCK ENABLE (HD) low; and reset bit counter.

Step 5. If Control Word received: set Control Function Enable flip-flop and unlock keyboard with NON A/N KEYBOARD RELEASE. Proceed to step 19.

Step 6. If Control Function Enable flip-flop set: decode appropriate instructions (steps 7 through 13). If Control Function Enable not set, proceed to step 14.

Step 7. If DEL, BEL, EOT, LF, FG/BG, NUL, HOME/IO, or CA/IO received: unlock keyboard with NON A/N KEYBOARD RELEASE.

Step 8. If HOME/IO received: $\overline{\text{CLEAR CAX/IO}}$ and $\overline{\text{CLEAR CAY/IO}}$ pulses. Proceed to step 19.

Step 9. If FG/IO received: set Bit 8/IO Receive flip-flop. Proceed to step 19.

Step 10. If BG/IO received: reset Bit 8/IO Receive flip-flop. Proceed to step 19.

Step 11. If CA/IO received: $\overline{\text{CLEAR CAX/IO}}$ and $\overline{\text{CLEAR CAY/IO}}$ pulses, and toggle CA/IO counter. Proceed to step 19.

Step 12. If CA/IO counter=1: STROBE CAX/IO pulse and toggle CA/IO counter. Proceed to step 19.

Step 13. If CA/IO counter=2: STROBE CAY/IO pulse and reset CA/IO counter by toggling once. Proceed to step 19.

Step 14. If DEL, BEL, EOT, LF, or NUL received, or P-CTR BUSY, unlock keyboard with NON A/N KEYBOARD RELEASE. Proceed to step 19.

Step 15. When PROCESS I/O TIME GATE occurs: reset Data Ready flip-flop with DATA READY ACKNOWLEDGE/IO and $\overline{\text{PROCESS I/O ENTRY)'}}$ low.

Step 16. At P-CTR-1: $\overline{\text{PROCESS I/O ENTRY)'}}$ high and $\overline{\text{I/O INIT KB UNLOCK}}$ low.

Step 17. If CR received on line 26, or if any A/N character received at (X=73, Y=26): set D/L (R/U) flip-flop and $\overline{\text{PROCESS I/O ENTRY}}$ low. (Reset D/L(R/U) flip-flop when XFER COIN CTR--CA occurs at end of "roll up".)

Step 18. Go back to step 2.

Step 19. Reset Data Ready flip-flop with $\overline{\text{DATA READY ACKNOWLEDGE/IO}}$. Go back to step 2.

3.6.8.2 I/O Interface Circuits. The function of the I/O interface circuits is to provide an interface between the other video display terminal circuits and modem

when transmitting or receiving, and between the other video display terminal circuits and the tape cassette and printer units. Control signals and data pass between the I/O interface circuits and the modem, and between the I/O interface circuits and the tape cassette unit on a two-way basis. Though control signals pass in both directions between the I/O interface circuits and the printer unit, data are sent to the printer unit on a one-way basis.

3.6.8.2.1 Control Signals to Modem. Three control signals are sent to the local modem by the video display terminal: request to send (CA), supervisory transmitted data (SA), and data terminal ready (CD). As shown in figure 3-25, the CD output of the video display terminal coming from line level converter B16-C2H is permanently wired to be at a constant high level, indicating that the video display terminal is always ready. The CA and SA output signals of the video display terminal in general complement each other. That is, when one is at a high level the other is at a low level. When the CA-SA generating circuit is wired as shown in figure 3-25, either of two methods are used to achieve line turnaround (from transmit to receive and vice-versa), depending upon whether it is desired to use an end-of-transmission (EOT) signal or a carriage return (CR) signal to indicate that a line turnaround is required. If EOT is selected, line turnaround is controlled by both the CPU and the video display terminal. Should CR be selected, the CPU is in exclusive control of line turnaround. A jumper option on the A18 board permits line turnaround to be controlled by a supervisory received data (SB) signal from the CPU when the EOT/CR switch is in the CR position. Each of these three methods of achieving line turnaround is individually described in the following paragraphs. Note that line turnaround is necessary only when the video display terminal is operating in the batch or half duplex modes. Operation of the request to send (CA) circuit during full duplex operation is described below. Note that pressing the LOCAL switch will cause the CA output signal to the modem to go high, thereby generating a request to send, and that pressing the BREAK switch when the video display terminal is receiving data causes a 300 millisecond duration negative-going SA pulse to the modem to inform the CPU to cease transmitting and prepare to receive.

The use of EOT to achieve line turnaround is normally employed only when the video display terminal is operating in the batch mode. It is possible, however, to achieve line turnaround using EOT when in the conversational mode (half duplex) if EOT is entered manually by simultaneously pressing the CTRL and D keys at the

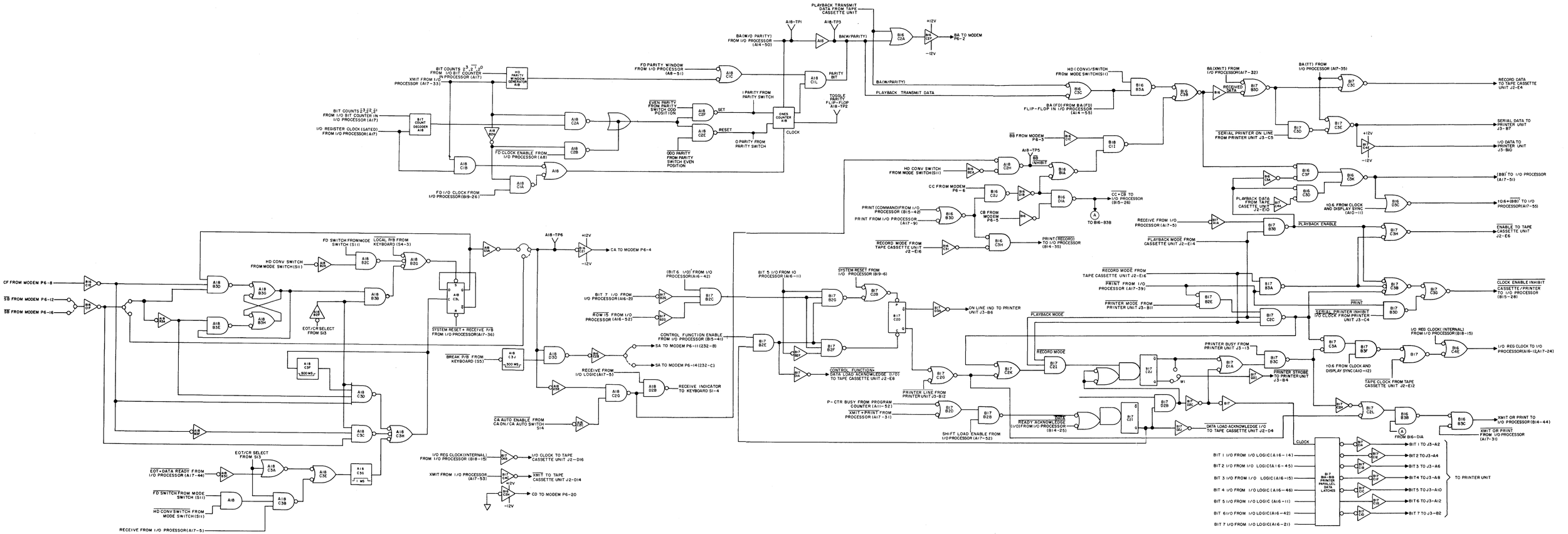


Figure 3-25. I/O Interface

conclusion of each data transmission. When the video display terminal has a complete batch of data ready for transmission to the CPU and EOT/CR switch S13 in the EOT position, an $\overline{\text{EOT} \cdot \text{DATA READY}}$ signal from the I/O processor circuits enables AND gate A18-C3A. This triggers single-shot multivibrator A18-C3G via OR gate A18-C3E, causing a 1 millisecond duration positive-going pulse to be generated. The negative-going trailing edge of this pulse toggles flip-flop A18-C3L from the reset state to the set state, causing the CA output of line level converter B16-C2I to go high. This conditions the local modem to receive data from the video display terminal and transmit them to the CPU. At the same time that the CA signal goes high, the SA output signal of line level converter B16-C2B, which goes to the modem, goes low. In addition, AND gate A18-C2G is enabled, since CA ON/CA AUTO switch S14 is in CA AUTO except when in the full duplex mode, causing the RECV indicator lamp on the keyboard to go out. Once the data have been transmitted, a high level receive signal from the I/O processor circuits enables AND gate A18-C3B. This results in flip-flop A18-C3L being toggled reset after a 1 millisecond delay, causing the CA output of the video display terminal to low, the SA output to go high, and the RECV indicator lamp on the keyboard to light. This conditions the local modem to receive data from the CPU and transmit them to the video display terminal.

When the video display terminal is turned on and the EOT/CR switch is in CR, AND gate A18-B3D is enabled, thereby setting OR flip-flop A18-B3G, B3H. This causes flip-flop A18-C3L to set and remain set (via enabled AND gate A18-B3B) even though the $\overline{\text{SYSTEM RESET}}$ signal at its reset input goes low at turn-on. Thus, at turn-on time the CA output of the video display terminal is high, while the SA output is low. When the CPU subsequently comes on line, the $\overline{\text{SB}}$ input to the video display terminal from the modem goes momentarily high. This enables AND gate A18-B3E, thereby resetting OR flip-flop A18-B3G, B3H and removing the low level from the direct set input of flip-flop A18-C3L. When the CPU is ready to transmit data to the video display terminal, the CF input to the video display terminal goes high and the $\overline{\text{SB}}$ input goes low. This enables AND gate A18-C3C, causing flip-flop A18-C3L to be toggled to the reset state via OR gate A18-C3H. As a result, the CA output of line level converter B16-C2I goes low, the SA output of line level converter B16-C2B goes high, and the RECV indicator lamp on the keyboard lights. When the CPU stops sending data, the CF input to the video display terminal goes low and the $\overline{\text{SB}}$ input goes high, thereby enabling AND gate A18-C3D. This causes flip-flop A18-C3L to be toggled to the set state and triggers single-shot delay multivibrator

A18-C3F. The latter circuit generates a 500 millisecond duration negative-going pulse that inhibits AND gate A18-C3D, thus preventing possible noise on the CF input line from causing a spurious line turnaround. During the period that AND gate A18-C3D is inhibited, the $\overline{SB}$ input to the video display terminal goes momentarily low, enabling AND gate A18-B3D. This sets OR flip-flop A18-B3G, B3H, thereby latching flip-flop A18-C3L in the set state via its direct set input. Thus, when AND gate A18-C3D is again enabled at the end of the 500 millisecond delay, its output is unable to toggle flip-flop A18-C3L back to the reset state. When flip-flop A18-C3L goes to the set state, the CA output of the video display terminal goes high, the SA output goes low, and the RECV indicator lamp on the keyboard goes out. Turnaround from receive to transmit is thus largely controlled by the CF input signal, while turnaround from transmit to receive is largely controlled by the $\overline{SB}$ input signal. The video display terminal may be operated in either the batch or conversational mode when using this method of achieving line turnaround.

The third method of achieving line turnaround is by slaving the CA output of the video display terminal to its $\overline{SB}$ input. This is done by jumpering the output of line level converter B16-C1F to the input of line level converter B16-C2I. Jumpering is done on the A18 board. EOT/CR switch S13 must be in the CR position when this method of achieving line turnaround is employed.

When the video display terminal is connected to a modem capable of simultaneous two-way transmission, the full duplex mode of operation may be employed. AND gate A18-B2C is enabled when the mode select switch (S11) is set to the FULL DUPLEX position. This causes a low level to be applied to the direct set input of flip-flop A18-C3L via OR gate A18-B2G, thus holding the flip-flop in the set state at all times when in full duplex mode. As a result, the CA output of line level converter B16-C2I is always high, permitting continuous transmission of data from the video display terminal to the modem and from thence to the CPU. Since CA ON/CA AUTO switch S14 must be in the CA ON position when operating in the full duplex mode, AND gate A18-C2G is disabled, thereby preventing the RECV indicator lamp on the keyboard from lighting.

3.6.8.2.2 Data to Modem. When the tape cassette unit associated with the video display terminal is operating in the character/on line/playback mode, serial playback transmit data from the tape cassette unit are fed to the local modem as a BA signal via OR gate B16-C2A and line level converter B16-C2F. No other circuits of the video display terminal are utilized when operating in this mode. At

all other times, data to be transmitted to the local modem for subsequent transmission to the CPU are passed through the I/O processor circuits before being sent to the modem via the I/O interface circuits. This is true whether the data are keyboard or tape cassette unit originated. Data for transmission to the modem are applied to the I/O interface circuits from the I/O processor circuits as a BA (W/O PARITY) signal. This signal is passed through buffer amplifiers on the A18 board and has a parity bit added to it before being applied to the modem via OR gate B16-C2A and line level converter B16-C2F. The operation of the parity bit generating circuit is described in the following paragraph.

The parity bit generating circuit is physically located on the A18 board. Its method of operation depends upon the mode of operation of the video display terminal and whether odd, even, 1 or 0 parity is desired. The parity bit is inserted in the ninth bit position of each data character (eighth bit after start), regardless of the mode of operation, by either a low level $\overline{\text{FD PARITY WINDOW}}$ signal from the I/O processor or a low level signal from the half duplex parity window generator circuit on the A18 board. This latter circuit generates a low level signal when the video display terminal is operating in the batch mode, the bit counter in the I/O processor circuits contains a 9, and a high level XMIT signal is applied to the half duplex parity window generator from the I/O processor circuits. In either case, the parity window signal is passed through OR gate A18-C1C to one of the two inputs to AND gate A18-C1L, thus enabling it during the ninth count of each data character. Regardless of the mode of operation of the video display terminal, the actual parity bit is generated in the ones generator. When PARITY switch S10 is in either the 1 or 0 position, the parity bit inserted in each data character is always a 1 or a 0. If the PARITY switch is in the ODD position, the parity flip-flop in the ones counter is set at the beginning of each data character by a negative-going pulse from AND gate A18-C2F, which is enabled by the output of either AND gate A18-C2A (batch mode) or AND gate A18-C2B (full or half duplex mode). The former of these two AND gates is enabled when a high level XMIT signal from the I/O processor circuits is present and the bit count decoder detects 0 or 1 in the bit counter in the I/O processor circuits. AND gate A18-C2B is enabled when the XMIT signal is at a low level and there is a high level $\overline{\text{FD CLOCK ENABLE}}$ signal from the I/O processor circuits present. The latter conditions arise only when the video display terminal is being operated in the full or half duplex modes. AND gate A18-C2E is enabled in a manner similar to AND gate A18-C2F when the PARITY switch is in the EVEN position, which results in the parity

flip-flop in the ones counter being reset at the beginning of each data character. The ones counter is subsequently clocked and reset a number of times during each data character by either an I/O REGISTER CLOCK (GATED) or FD I/O CLOCK (depending upon the mode of operation of the video display terminal) from the I/O processor circuits and the data bits in the data character. The clocking and resetting is performed in such a manner that the parity bit, when added to the data character via AND gate A18-C1L, causes the character subsequently transmitted to the modem to contain either an odd or an even number of ones.

3.6.8.2.3 Serial Data Accessories. The video display terminal is capable of transmitting serial data to either of two accessories: the tape cassette unit or the printer unit. The serial data may represent received data from the CPU, data generated in the video display terminal, or playback data from the tape cassette unit.

When the video display terminal is transmitting in the half duplex mode, a BA ($\overline{FD}$) signal from the A14 board is applied to AND gate B16-B3A, which is enabled when mode switch S11 is in the HALF DUPLEX position, and OR gate B16-C3B to inverter B16-C3J and driver B16-D3B, from which it goes to OR gate B17-B3D as RECD DATA. If the tape cassette unit is in the record mode, and thus takes precedence over the printer unit, the record data pass through OR gate B17-C3C to the tape cassette unit. If the $\overline{\text{SERIAL PRINTER ON LINE}}$ signal from the printer unit is at a high level, the data pass through AND gate B17-C3D and OR gate B17-C3E to the printer unit. The serial data output signal of OR gate B17-C3E is also passed through line level converter B17-C4E and sent to the printer unit as an I/O DATA signal.

When the video display terminal is operating in the batch mode and the data being transmitted to the modem are also to be recorded either at the tape cassette unit or at the printer unit, the data in the form of a BA (XMIT) signal from the I/O processor circuits are fed through OR gate B17-B3D to the input of OR gate B17-C3C and AND gate B17-C3D. These data are then gated through to either the tape cassette unit or the printer unit in the manner described in paragraph 3.6.8.2.3.1.

A print command initiated at the keyboard causes data in the form of a BA (TT) signal from the I/O processor circuits to be applied to the inputs of OR gate B17-C3C and OR gate B17-C3E. The data are then sent to either the tape cassette unit or the printer unit.

Received data ($\overline{BB}$) from the CPU (by way of the modem) are passed through buffer B16-C1C to AND gate B16-C1I. This gate is enabled to pass the received data on to OR gate B16-C3B, from whence they are sent to either the tape cassette unit or printer unit as described above, if its other input is at a high level. For AND gate B16-C1I to be enabled, the data set ready (CC) signal from the modem to AND gate B16-C2J must be high, and the PRINT (COMMAND) and PRINT signals from the I/O processor circuits to OR gate B16-B3D must be low. When low, the $\overline{\text{BB INHIBIT}}$ signal inhibits received data (BB) from being sent to the tape cassette unit, printer unit, and I/O processor circuits. The only time that this signal goes low is when the video display terminal is transmitting while in the half duplex mode of operation. The CC signal from the modem is high whenever the CPU is transmitting data to the video display terminal, while the PRINT (COMMAND) and PRINT signals are high whenever the video display terminal is in the print mode.

3.6.8.2.4 Parallel Data to Printer Unit. Seven parallel data bits from the I/O processor circuits are entered into printer parallel data latches B17-B1A and B17-B1B whenever they are clocked by a positive-going pulse from single-shot multivibrator B17-C2I. This can only occur if AND gate B17-D2B is enabled and the single-shot multivibrator is triggered. Single-shot multivibrator B17-C2I is triggered whenever the SHIFT LOAD ENABLE input signal to AND gate B17-B2B from the I/O processor circuits goes high at the same time that the P-CTR BUSY signal from the program counter or the $\overline{\text{XMIT + PRINT}}$ signal from the I/O processor circuits is low. Either of the latter signals enables OR gate B17-B2D, thus enabling AND gate B17-B2B. For the output pulse of the single-shot multivibrator to clock the printer parallel data latches, however, AND gate B17-D2B must be enabled. This occurs when the tape cassette unit is not in either playback or record mode, and OR gate B17-C2K is enabled, as AND gate B17-C2I is enabled under these conditions. OR gate B17-C2K is enabled whenever either of its two inputs is low. One input is derived from AND gate B17-C2C, the output of which goes low whenever the tape cassette unit is not in either playback or record mode, the $\overline{\text{PRINT}}$ signal from the I/O processor circuits is low, and the $\overline{\text{PRINTER MODE}}$ signal from the printer unit is low. The other input to OR gate B17-C2K is low whenever the printer unit is on line. Whenever the ON LINE indicator/switch on the printer unit is actuated, the $\overline{\text{PRINTER ON LINE}}$ input signal to OR gate B17-C2G is low, which results in enabling OR gate B17-C2K. OR gate B17-C2G is enabled by a low level signal from the Q output of flip-flop B17-C2D when the CPU instructs the printer unit to go on line.

This is done by sending a control shifted period followed by a slash (/) to the terminal. This results in the BIT 5 I/O, BIT 7 I/O, and $\overline{\text{ROW 15}}$ signals going low and the (BIT 6 I/O)' and CONTROL FUNCTION ENABLE signals, all from the I/O processor circuits, going high. When the SHIFT LOAD ENABLE signal subsequently goes high and fires single-shot multivibrator B17-C2I, the output of AND gate B17-B2E enables AND gate B17-B2F to clear flip-flop B17-C2D. Clearing flip-flop B17-C2D results in the ON LINE indicator on the printer unit lighting and the application of a clock pulse to the printer parallel data latches. One data character is transmitted to the printer unit on each clock pulse to the parallel data latches. To return the printer to the off line condition, a control shifted period followed by a question mark is sent to the printer unit. When this is done, the input signals to the gates controlling flip-flop B17-C2D remain in the same states as when the printer unit is to be placed on line with the exception that the BIT 5 (I/O) signal goes high. This disables AND gate B17-B2F and enables AND gate B17-B2G, which results in the setting of flip-flop B17-C2D via OR gate B17-C2B. This same OR gate can also set the flip-flop whenever the $\overline{\text{SYSTEM RESET}}$ signal from the I/O processor circuits goes low. Setting flip-flop B17-C2D causes the ON LINE indicator on the printer unit to go out and prevents the printer parallel data latches from being clocked.

3.6.8.2.5 Control Signals to Tape Cassette Unit. The video display terminal sends five control signals to the tape cassette unit. Each of these control signals is individually described in the following paragraphs.

A low level $\overline{\text{ENABLE}}$ signal is sent to the tape cassette unit enabling it to play back data when the video display terminal is in receive mode and playback has been manually selected at the tape cassette unit, or when the tape cassette unit is in record mode and there is a low level $\overline{\text{PRINT}}$ signal from the I/O processor circuits present at AND gate B17-B3A. In either case, a low level signal is applied to one of the two inputs to OR gate B17-C3H, which passes it on to the tape cassette unit as an $\overline{\text{ENABLE}}$ signal. The $\overline{\text{PLAYBACK ENABLE}}$ output signal of AND gate B17-B3B is also applied to the circuit that send the (BB)' signal to the I/O processor circuits.

An $\overline{\text{I/O CLOCK TO CASS}}$ signal is also sent to the tape cassette unit by the I/O interface circuits. This signal, which is derived from the I/O processor circuits, is sent to the tape cassette unit via inverter B17-C4B.

An $\overline{\text{XMIT}}$ signal from the I/O processor circuits is also sent to the tape cassette unit. This signal is inverted in inverter B17-C4C.

When the CONTROL FUNCTION ENABLE input signal to AND gate B17-B2E is high and single-shot multivibrator B17-C2I is triggered by the negative-going edge of a $\overline{\text{DATA READY ACKNOWLEDGE}}$ (I/O) signal from the I/O processor circuits, the AND gate is enabled. This causes a negative-going $\overline{\text{CONTROL FUNCTION}\cdot\text{DATA LOAD ACKNOWLEDGE}}$ signal to be sent to the tape cassette unit via inverter B17-D1A.

Whenever single-shot multivibrator B17-C2I is triggered in the manner described in paragraph 3.6.8.2.5.4 above, its positive-going output pulse is inverted and applied to the tape cassette unit. Inversion is performed in inverter B17-D2J.

3.6.8.2.6 Data to I/O Processor Circuits. A data signal (BB)' and a t0.6 + $\overline{(BB)'}$ signal are sent to the I/O processor circuits from the interface circuits when the video display terminal is receiving data ($\overline{BB}$) from the modem, when the video display terminal is transmitting data in the half duplex mode, and when the tape cassette unit is playing back data to the video display terminal. AND gate B16-C3F is enabled to pass either received data or transmitted data to OR gate B16-C3K and thence to the I/O processor circuits when the $\overline{\text{PLAYBACK ENABLE}}$ input signal to inverter B16-C3A is high. When the $\overline{\text{PLAYBACK ENABLE}}$ signal is low, PLAYBACK DATA from the tape cassette unit are enabled to pass through AND gate B16-C3D and OR gate B16-C3K to the I/O processor circuits. The (BB)' output signal of OR gate B16-C3K is also ORed with a t0.6 signal from the clock and display sync circuits in OR gate B16-D3C, the t0.6 + $\overline{(BB)'}$ output of which is applied to the I/O processor circuits.

3.6.8.2.7 Control Signals to I/O Processor Circuits. The I/O interface circuits send five control signals to the I/O processor circuits. These signals are $\overline{\text{CLOCK ENABLE INHIBIT CASSETTE/PRINTER}}$, I/O REGISTER CLOCK, XMIT OR PRINT, PRINT $\overline{(\text{RECORD})}$, and $\text{CC}\cdot\overline{\text{CB}}$. The I/O interface circuits affecting each of these signals is separately described in the following paragraphs.

The $\overline{\text{CLOCK ENABLE INHIBIT CASSETTE/PRINTER}}$ signal goes low and thereby inhibits the generation of the baud rate clock in the I/O processor circuits when any one of four sets of conditions apply. One of these is when AND gate B17-B3D is enabled by a low level $\overline{\text{PRINT}}$ from the I/O processor circuits and a low level $\overline{\text{SERIAL PRINTER INHIBIT I/O CLOCK}}$ signal from the printer unit. It is also low when the $\overline{\text{PLAYBACK ENABLE}}$ output signal of AND gate B17-B3B is low, as described in paragraph 3.6.8.2.6, and when a low level $\overline{\text{ENABLE}}$ signal is sent to the tape cassette unit as described

in paragraph 3.6.8.2.5. The fourth condition under which the $\overline{\text{CLOCK ENABLE INHIBIT CASSETTE/PRINTER}}$ signal goes low is when AND gate B16-C2C is enabled, as described in paragraph 3.6.8.2.4.

I/O REGISTER CLOCK signals are sent to the I/O processor circuits via OR gate B16-C4E either when it is receiving an I/O REGISTER CLOCK (INTERNAL) signal from the I/O processor circuits, a $\overline{\text{TAPE CLOCK}}$ signal is being applied to the noninverting OR gate on B17, or when AND gate B17-C3A is enabled and AND gate B17-C3F is enabled by a t0.6 pulse from the clock and display sync circuits. The latter condition occurs when AND gate B17-C2C is enabled as described in paragraph 3.6.8.2.4 and AND gate B17-B3C is enabled by a high level PRINTER BUSY signal from the printer unit and a positive-going pulse from OR gate B17-D1A. A positive-going pulse ensues at the output of OR gate B17-D1A on either the trailing edge of the $\overline{Q}$ output of a single-shot multivibrator B17-C2J or the positive-going edge of the Q output of single-shot multivibrator B17-C2I. Each of these single-shot multivibrators is triggered as described in paragraph 3.6.8.2.4.

Whenever the printer unit is one line or the PRINTER BUSY input signal to AND gate B17-B3C is low, the output of OR gate B17-C2L goes high. If at the same time the $\overline{\text{CC}\cdot\overline{\text{CB}}}$ output signal of AND gate B16-D1A (see paragraph 3.6.8.2.7) is also high, the resulting low level output of AND gate B16-B3B enables a low level $\overline{\text{XMIT OR PRINT}}$ signal from the I/O processor circuits to cause a high level XMIT OR PRINT signal to be sent to the I/O processor circuits via AND gate B16-B3C.

A high level PRINT/$(\overline{\text{RECORD}})$ signal is generated in AND gate B16-C3H and sent to the I/O processor circuits when the gate is enabled by two low level inputs. One of these inputs goes low whenever the PRINT (COMMAND) or PRINT inputs to OR gate B16-B3D are low, while the other is low when the $\overline{\text{RECORD MODE}}$ input signal to inverter B16-C3I is high.

A $\overline{\text{CC}\cdot\overline{\text{CB}}}$ signal from AND gate B16-D1A is sent to the I/O processor circuits and to one of the inputs of AND gate B16-B3B. This signal must be high for data to be transmitted by the video display terminal, and low for data to be received. The data set ready (CC) signal applied to AND gate B16-C2J from the modem must be high for the video display terminal to receive data, as must the output of OR gate B16-B3D. In addition, the clear to send (CB) signal from the modem must be low in order to enable AND gate B16-D1A. When CC is low, the $\overline{\text{CC}\cdot\overline{\text{CB}}}$ signal is high regardless of the status of the CB signal, thus permitting the video display terminal to transmit data.

3.6.9 CURSOR ADDRESS LOGIC

The Cursor Address Logic generates cursor X and Y addresses which correspond to the cursor, or character, positions on the display in the X (horizontal) and Y (vertical) axis, respectively. There are 74 (0 through 73) cursor X addresses, one for each character position on a particular line, and 27 (0 through 26) cursor Y addresses, one for each character line. Thus, there are 1998 unique (X,Y) cursor address locations, starting with address location (0,0) in the upper left-hand corner of the screen (designated HOME) and ending with address location (73,26) in the lower right-hand corner. The cursor addresses determine where the cursor is displayed on the screen, as well as where in memory the next character entered will be stored. At the start of each refresh cycle, the cursor X and Y addresses are sampled and the cursor is painted under the appropriate character location when that character is displayed. The cursor addresses are also used to address the Memory Logic for keyboard entry (typing), editing data, transmitting data, programming, etc. Cursor position commands from the Keyboard Input Logic, Decision Logic, Internal Program Counter, and Input/Output Logic increment or decrement cursor address X and Y counters to control the movement of the cursor on the screen and to address specific characters or groups of data in the Memory Logic. Functionally, the Cursor Address Logic (figure 3-26) consists of cursor address X and Y counters, X and Y coincidence counters, their respective control logic circuits, decoders, and counter preset circuits, and a cursor video circuit.

3.6.9.1 Cursor Address X (CAX) Counter and CAX Control Logic. The cursor address X counter is a 7-bit up/down binary counter whose outputs provide the 7-bit cursor X address ($CAX=2^0$ through $CAX=2^6$) and its complement ($\overline{CAX=2^0}$ through $\overline{CAX=2^6}$). The CAX control logic accepts right ($\overline{+1X}$ and X=RIGHT) and left ($\overline{-1X}$ and X-LEFT) cursor commands and generates the required signals to increment or decrement the counter accordingly. Basically, when a $\overline{+1X}$ or X-RIGHT signal is applied to the CAX control logic, the counter is advanced (counts up) one count to either move the cursor one space to the right or to address the next character location in the memory logic. Similarly, a $\overline{-1X}$ or X-LEFT command causes the counter to decrease (count down) one count to back-space the cursor or to address the character location in memory one space to the left. When a $\overline{+1X}$ (or $\overline{-1X}$) signal is received from the Internal Program Counter, the CAX control logic initiates two outputs: first, an $\overline{\text{ADD ENABLE CAX}}$ (or $\overline{\text{SUBTRACT ENABLE CAX}}$) signal is generated to enable the count up (or count down) input of the counter; and second, a $\overline{\text{TOGGLE CA}}$ signal is applied to the toggle logic.

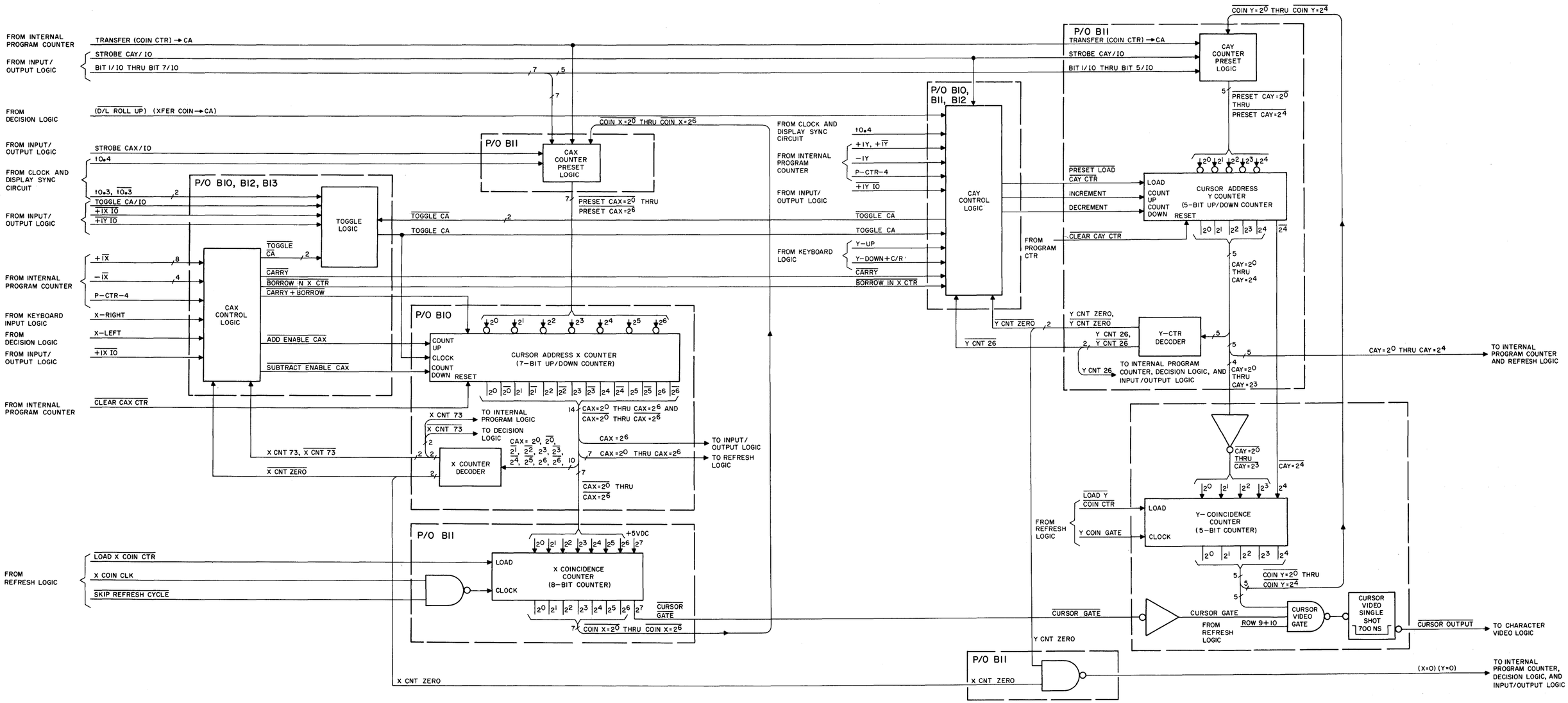


Figure 3-26. Cursor Address Logic, Functional Block Diagram

The $\overline{\text{TOGGLE CA}}$ input to the toggle logic gates the t0.3 input from the Clock and Display Sync Circuit to the TOGGLE CA output. Thus, a 100-nanosecond TOGGLE CA pulse clocks the counter to increment (or decrement) the cursor X address one count. Subsequently, for each $\overline{+1X}$ command the cursor X address increases one count, and for each $\overline{-1X}$ command the cursor X address is decreased by one count. However, the cursor address X counter outputs are applied to an X counter decoder which decodes count 0 and 73 to detect the extreme left (first) or extreme right (last) character positions on the line. When the cursor X address is at the last character position (count 73) an X CNT 73 signal is generated and applied to the CAX control logic. If a $\overline{+1X}$ command is received from the Internal Program Counter when the X CNT 73 signal is present, the CAX control logic generates $\overline{\text{CARRY+BORROW}}$ and $\overline{\text{CARRY}}$ signals, in addition to the $\overline{\text{ADD ENABLE CAX}}$ and $\overline{\text{TOGGLE CA}}$ signals. With both a $\overline{\text{CARRY+BORROW}}$ and $\overline{\text{ADD ENABLE CAX}}$ applied to the counter, the counter outputs go to zero when the TOGGLE CA pulse clocks the counter. In addition, with a $\overline{\text{CARRY}}$ signal applied to the CAY control logic, the TOGGLE CA pulse causes the cursor address Y counter to increment (count up) one count. Thus, the cursor X address goes to zero and the cursor Y address advances to the next higher count. This causes the cursor to advance to the first character position on the next lower line. A similar operation is performed when the cursor X address is at the first character position and $\overline{-1X}$ signal is received from the Internal Program Counter. With the cursor at the first character position, the X counter decoder decodes the count of zero and applies an $\overline{\text{X CNT ZERO}}$ signal to the CAX control logic. If a $\overline{-1X}$ signal is then received, the CAX control logic generates $\overline{\text{CARRY+BORROW}}$ and $\overline{\text{BORROW IN X CTR}}$ signals, in addition to the $\overline{\text{SUBTRACT ENABLE CAX}}$ and $\overline{\text{TOGGLE CA}}$ signals. With a $\overline{\text{CARRY+BORROW}}$ and $\overline{\text{SUBTRACT ENABLE CAX}}$ applied to the cursor address X counter, and $\overline{\text{BORROW IN X CTR}}$ signal applied to the CAY control logic, the TOGGLE CA pulse causes the cursor X address to go to 73 and the cursor Y address to decrement one count. Thus, the cursor moves from the first character position on the line to the last character position on the next line up.

The operations for $\overline{+1X\ IO}$ commands from the Output Logic are similar to those for the $\overline{+1X}$ commands from the Internal Program Counter. However, a $\overline{+1X\ IO}$ input to the CAX control logic does not initiate a $\overline{\text{TOGGLE CA}}$ output. Instead, the input/output logic applies $\overline{+1X\ IO}$ and TOGGLE CA/IO signals directly to the toggle logic. When the TOGGLE CA/IO input goes high the TOGGLE CA output clocks the counter. Note that the Cursor Address Logic does not receive a $\overline{-1X\ IO}$ input directly from the Input Output Logic. However, when a back-space (-1X) is initiated by the Input/Output

Logic, a BACKSPACE RECEIVED signal is applied to the Decision Logic which, in turn, sends an X-LEFT signal to the Cursor Address Logic.

The X-RIGHT input is initiated by the right-arrow cursor control key on the keyboard and the X-LEFT input is initiated by either the left-arrow cursor control key or the BACKSPACE RECEIVED signal mentioned above. The operations from the X-RIGHT and X-LEFT cursor commands are the same as for the $\overline{+1X}$ and $\overline{-1X}$ commands from the Internal Program Counter, with two exceptions. First, the X-RIGHT and X-LEFT inputs are gated by the P-CTR-4 input from the Internal Program Counter. This functions to synchronize X-RIGHT and X-LEFT command cursor movements with count 4 of the Internal Program Counter. In addition, when the CAX control logic receives an $\overline{\text{X CNT 73}}$ signal the X-RIGHT command is inhibited, and similarly, when an $\overline{\text{X CNT ZERO}}$ signal is present the X-LEFT input is inhibited. Thus, if the cursor is at the last character position on a line and the right-arrow cursor control key is pressed, the cursor remains stationary; it will not advance to the first character of the next line. Similarly, if the cursor is at the first position on a line and the left-arrow cursor control key is pressed, the cursor remains stationary.

The Cursor Address X Counter also receives a $\overline{\text{CLEAR CAX CTR}}$ input from the Internal Program Counter and PRESET CAX=2^0 through $\overline{\text{PRESET CAX=}2^6}$ inputs from the CAX counter preset logic. The $\overline{\text{CLEAR CAX CTR}}$ input is used to reset the cursor X address to zero for various terminal functions. The preset inputs are used to insert cursor X addresses from either the Input/Output Logic or X coincidence counter into the Cursor Address X Counter.

The CAX=2^0 through CAX=2^6 outputs of the cursor address X counters are routed to the Refresh Logic. During the horizontal retraces and skip refresh functions, the cursor addresses are gated through the refresh logic to address the corresponding character locations in the memory logic. The most-significant bit (CAX=2^6) of the cursor X address is also routed to the Input/Output Logic to initiate the 1-kHz audible tone signal when the cursor reaches the 64th character position on any line. This alerts the operator during keyboard entry functions that the cursor is approaching the end of a line. The $\overline{\text{CAX=}2^0}$ through $\overline{\text{CAX=}2^6}$ outputs of the cursor address X counter are applied to the X coincidence counter which is described in the following paragraph. As previously discussed, several counter outputs are also applied to the X counter decoder to decode cursor X addresses zero and 73. Decoded outputs are, in turn, routed to the CAX control logic, Internal Program Counter, and Decision Logic. In addition, the X CNT ZERO output is ANDed with the Y CNT ZERO output of the Y counter decoder to obtain a $\overline{(X=0)(Y=0)}$ signal. The $\overline{(X=0)(Y=0)}$ output of

the Cursor Address Logic is routed to the Input/Output Logic, Decision Logic, and Internal Program Counter to signal these circuits when the cursor is at the HOME position.

3.6.9.2 X Coincidence Counter. The X coincidence counter performs two separate functions: first, it samples the complement of the cursor X addresses during the refresh cycles to generate the cursor video pulses; and second, it stores the complement of the cursor X addresses during skip refresh functions while the Internal Program Counter utilizes the cursor address X counter to address various character locations in the Memory Logic. At the end of the particular skip refresh function the cursor X address is preset back into the cursor address X counter via the CAX counter preset logic.

During the refresh cycles, the refresh logic generates a $\overline{\text{LOAD X COIN CTR}}$ pulse at the beginning of each horizontal sweep period to load the complement of the cursor X address into the X coincidence counter. Simultaneously, a logic one (+5 vdc) is preset into the MSB of the counter, causing the 2^7 output of the counter ($\overline{\text{CURSOR GATE}}$) to go high. During the video unblanking portion of the horizontal sweep, the refresh logic applies 100-nanosecond X COIN CLK pulses to the Cursor Address Logic every 700 ns. These pulses are gated to the clock input of the X coincidence counter during each refresh cycle by the logic one $\overline{\text{SKIP REFRESH CYCLE}}$ signal from the Refresh Logic. An X COIN CLK pulse is applied to the X coincidence counter every 700 nanoseconds to increment the counter 100 nanoseconds before each character position is scanned on the screen. When the character being scanned is coincident with the cursor X address the $\overline{\text{CURSOR GATE}}$ output goes low to enable $\overline{\text{CURSOR GATE}}$ input of the cursor video AND gate. For example, if the cursor X address is 4, corresponding to the fifth character position on the line (the first position is address zero), the complement of the address (CAX=2^0 through CAX=2^6) 1101111 is loaded into the X coincidence counter. Thus, the counter outputs, including the CURSOR GATE output is 11011111. The first four X COIN CLK pulses then advance the counter to 11111111 and the fifth clock pulse, occurring 100 nanoseconds before the fifth character position is scanned, causes the $\overline{\text{CURSOR GATE}}$ output to go low.

For the skip refresh functions (delete character, insert character, and delete line) the Refresh Logic applies a $\overline{\text{LOAD X COIN CTR}}$ pulse to the X coincidence counter to load the complement of the cursor X address into the counter. However, during these functions the $\overline{\text{SKIP REFRESH CYCLE}}$ input from the Refresh Logic is low to inhibit X COIN CLK from clocking the counter. Thus, the cursor X address is stored in the

counter. The $\overline{\text{COIN X=2}^0}$ through $\overline{\text{COIN X=2}^6}$ outputs of the counter are applied to the CAX counter preset logic where they are gated to the cursor address X counter at the end of the skip refresh function by a TRANSFER (COIN CTR) → CA pulse from the Internal Program Counter.

3.6.9.3 CAX Counter Preset Logic. The CAX counter preset logic receives cursor X addresses from the Input/Output Logic (BIT I/IO through BIT 7/IO) and X coincidence counter ($\overline{\text{COIN X=2}^0}$ through $\overline{\text{COIN X=2}^6}$) and, under control of the STROBE CAX/IO and TRANSFER (COIN CTR) → CA x inputs, presets the respective address into the cursor address X counter. When a STROBE CAX/IO is applied from the Input/Output Logic, the BIT 1/IO through BIT 7/IO inputs are inverted and gated to the $\overline{\text{PRESET CAX=2}^0}$ through $\overline{\text{PRESET CAX=2}^6}$ outputs. When a TRANSFER ($\overline{\text{COIN CTR}}$) → CA is applied to the CAX counter preset circuit, the $\overline{\text{COIN X=2}^0}$ through $\overline{\text{COIN X=2}^6}$ outputs of the X coincidence counter are gated by the t0.4 input pulse to the preset logic outputs. Because the cursor address X counter accepts active-low preset inputs, the outputs of the CAX counter preset logic are actually the complement of the cursor address.

3.6.9.4 Cursor Address Y Counter and CAY Control Logic. The cursor address Y counter is a 5-bit up/down binary counter which provides the 5-bit cursor Y address CAY=2^0 through CAY=2^4. These outputs are applied to the Refresh Logic, Internal Program Counter and Y counter decoder. In addition, the CAY=2^0 through CAY=2^3 outputs are inverted and applied to the Y coincidence counter along with the $\overline{\text{CAY=2}^4}$ counter output. During the horizontal retraces and skip refresh functions, the Refresh Logic gates the cursor Y address, along with the cursor X address, to the Memory Logic. The Internal Program Counter utilizes the cursor Y address for the delete line function to detect the line being deleted. The Y counter decoder decodes counts zero and 26 of the counter and provides decoded outputs for the CAY control logic, Internal Program Counter, Input/Output Logic, Decision Logic, and $\overline{\text{(X=0)(Y=0)}}$ AND gate.

The CAY control logic functions similarly to the CAX control logic in that it accepts up (-1Y) and down (+1Y) and (Y-DOWN+C/R) cursor commands to decrement or increment the cursor address Y counter accordingly. Because the cursor Y addresses run from top to bottom on the display, a +1Y increments the counter to move the cursor down, and conversely, a -1Y decrements the counter to move the cursor up. When a +1Y (or -1Y) command is received from the Internal Program Counter a $\overline{\text{TOGGLE CA}}$ signal is applied to the toggle logic. This, in turn, gates the t0.3 or $\overline{\text{t0.3}}$ input to the TOGGLE CA output. The 100-nanosecond TOGGLE CA pulse is then fed

back to the CAY control logic where the +1Y (or -1Y) input steers the TOGGLE CA pulse to the increment (or decrement) output. Thus, for each +1Y or $\overline{+1Y}$ input the cursor Y address increases one count, and for each -1Y command the cursor Y address decreases one count.

When a +1Y cursor command is initiated at the Output Logic, a $\overline{+1Y\ IO}$ enable signal is applied to the CAY control logic and $\overline{+1Y\ IO}$ enable signal and 100 nanosecond TOGGLE CA/IO sync pulse are applied to the toggle logic. The $\overline{+1Y\ IO}$ input to the toggle logic gates the TOGGLE CA/IO pulse to the TOGGLE CA output and the $\overline{+1Y\ IO}$ input to the CAY control logic steers the TOGGLE CA pulse to the increment input of the counter.

The Y-UP input to the CAY control logic is initiated by the up-cursor control key on the keyboard and the (Y-DOWN + C/R) input is initiated by either the down-cursor control key or a carriage return function. The operations for the Y-UP and (Y-DOWN + C/R) cursor commands are similar to the operations for the -1Y and +1Y commands from the Internal Program Counter, with two exceptions. The Y-UP and (Y-DOWN + C/R) inputs are gated by the P-CTR-4 input to synchronize the up and down cursor movements with the Internal Program Counter. In addition, when the CAY control logic receives a Y CNT 26 signal, the (Y-DOWN + C/R) command is inhibited, and similarly, a Y CNT ZERO input inhibits the Y-UP command.

The CAY control logic also receives $\overline{\text{CARRY}}$ and $\overline{\text{BORROW IN X CTR}}$ signals from the CAX control logic to generate $\overline{\text{DECREMENT}}$ and $\overline{\text{INCREMENT}}$ signals, respectively. These signals and their related functions were described in detail when the CAX control logic was discussed in paragraph 3.6.9.1. In addition, the CAY control logic accepts STROBE CAY/IO and $\overline{\text{(D/L ROLL UP)}} \cdot$ (XFER COIN $\rightarrow$ CA) inputs from the Input/Output Logic and Decision Logic, respectively, to generate $\overline{\text{PRESET LOAD CAY CTR}}$ pulses for the cursor address Y counter. When the STROBE CAY/IO input goes high, the $\overline{\text{PRESET LOAD CAY CTR}}$ output goes low to enable the preset inputs of the counter. When the $\overline{\text{(D/L ROLL UP)}} \cdot$ (XFER COIN $\rightarrow$ CA) input goes high, the t0.4 pulse from the clock and display sync circuit is gated to the $\overline{\text{PRESET LOAD CAY CTR}}$ output.

3.6.9.5 Y Coincidence Counter. The Y coincidence counter performs the same two functions as the X coincidence counter. During the refresh cycles, the Refresh Logic generates a $\overline{\text{LOAD Y COIN CTR}}$ pulse at the start of each refresh field to load the complement of the cursor Y address into the Y coincidence counter. As row 6 or 7 of each character line is refreshed, starting with the second line, a Y COINCIDENCE GATE is applied to the counter. The trailing edge of each gate, in turn,

increments the counter one count. This subsequently causes the counter outputs to all be high when the character line being refreshed corresponds to the cursor Y address. For example, if the cursor Y address is 3, corresponding to the fourth line down from the top, the complement of the address, 00111, is loaded into the counter at the start of each field. The counter then advances to 10111 when row 6 or 7 of the second character line is refreshed, 01111 for the third line, and all "ones" for the fourth line. The logic one outputs of the counter are, in turn, applied to the cursor video AND gate.

For the skip refresh functions the Refresh Logic also generates a $\overline{\text{LOAD Y COIN CTR}}$ pulse to load the complement of the cursor Y address into the counter. However, the Y COIN GATE is inhibited during these functions and the cursor address remains stored in the counter.

3.6.9.6 CAY Counter Preset Logic. The CAY counter preset logic receives cursor Y addresses from the Input/Output Logic (BIT 1/IO through BIT 5/IO) and Y coincidence counter ($\overline{\text{COIN Y=}2^0}$ through $\overline{\text{COIN Y=}2^4}$) and, under control of the STROBE CAY/IO and TRANSFER (COIN CTR) → CA inputs, gates the respective cursor Y address to the preset inputs of the cursor address Y counter. When a STROBE CAY/IO signal is generated by the Input/Output Logic, the BIT 1/IO through BIT 5/IO inputs are inverted and gated to the $\overline{\text{PRESET CAY=}2^0}$ through $\overline{\text{PRESET CAY=}2^4}$ outputs of the CAY counter preset logic. Simultanesouly, a $\overline{\text{PRESET LOAD CAY CTR}}$ signal is generated by the CAY control logic to preset the cursor address into the cursor address Y counter. When a TRANSFER (COIN CTR) → CA pulse is generated by the Internal Program Counter, the $\overline{\text{COIN Y=}2^0}$ through $\overline{\text{COIN Y=}2^4}$ inputs from the Y coincidence counter are gated through the CAY counter preset logic. When the TRANSFER (COIN CTR) → CA pulse is generated, a $\overline{\text{(D/L ROLL UP)}} \cdot$ (XFER COIN → CA) signal is also applied to the CAY control logic. Thus, the t0.4 pulse from clock and display sync circuits produces a $\overline{\text{PRESET LOAD CAY CTR}}$ pulse to preset the cursor Y address from the Y coincidence counter into the cursor address Y counter.

3.6.9.7 Cursor Video Circuit. The cursor video circuit consists of an AND gate and a single-shot multivibrator. The $\overline{\text{COIN Y=}2^0}$ through $\overline{\text{COIN Y=}2^4}$ outputs of the Y coincidence counter, the inverted $\overline{\text{CURSOR GATE}}$ output of the X coincidence counter, and a ROW 9 + 10 signal from the refresh logic are applied to the cursor video AND gate. When all seven inputs to the AND gate are simultaneously high, its output goes low to trigger the single-shot. A 700-nanosecond $\overline{\text{CURSOR OUTPUT}}$ pulse is then routed to the Character Video Logic for the cursor video. During the refresh

cycles, the CURSOR GATE input to the AND gate goes low at the start of each horizontal sweep and goes high 100 nanoseconds before the character position corresponding to the cursor X address is scanned on the screen; the row 9 + 10 input is high each time row 9 or 10 of a character line is scanned; and the $\overline{\text{COIN Y=2}^0}$ through $\overline{\text{COIN Y=2}^4}$ all go high when row 6 or 7 of the character line corresponding to the cursor Y address is scanned, and remain high until row 6 or 7 of the next character line is scanned. Thus, the inputs to the cursor video AND gate are simultaneously high when row 9 or 10 of the character line corresponding to the cursor Y address is scanned, 100 nanoseconds before the character corresponding to the cursor X address is scanned. Because every other character row is scanned each field, a $\overline{\text{CURSOR OUTPUT}}$ pulse is generated for row 9 of one field and for row 10 of the following field. This paints a cursor under the character position corresponding to the cursor address which is two character rows in height and slightly greater than one character in width.

3.6.10 CHARACTER VIDEO LOGIC (figure 3-27)

The Character Video Logic is composed of two major functional sections, the character generator and the video logic. The character generator, a 2560-bit ROM, receives a 6-bit character address which identifies the character to be displayed and a 3-bit row address to identify which portion of the character is to be developed for the respective line being refreshed by the Refresh Logic. The output of the character generator is a 5-bit parallel data word applied to a shift register where it is serialized prior to being applied to the video logic. The video logic accepts the serial data from the character generator, the cursor output from the Cursor Address Logic and video level information from the Memory Logic. The video logic develops three levels of video brightness, blank or no video output, half brightness for background characters, and bright for foreground characters and cursor. The output of the video logic is applied to CONTRAST control R5 the output of which is routed to the Monitor. An additional output of the video logic is available for a remote monitor.

3.6.10.1 Character Generator (figure 3-28). Each character that is displayed on the CRT is allocated a 7 by 11 dot window. Each character is developed in the upper left 5 by 7 dot area of the window. The two spaces to the right of the character are inter-character spaces and the four spaces below the character are allocated for the cursor (Rows 9 and 10) and inter-line spacing. As the horizontal sweep traces across the CRT, the CRT is intensity modulated by the video logic.

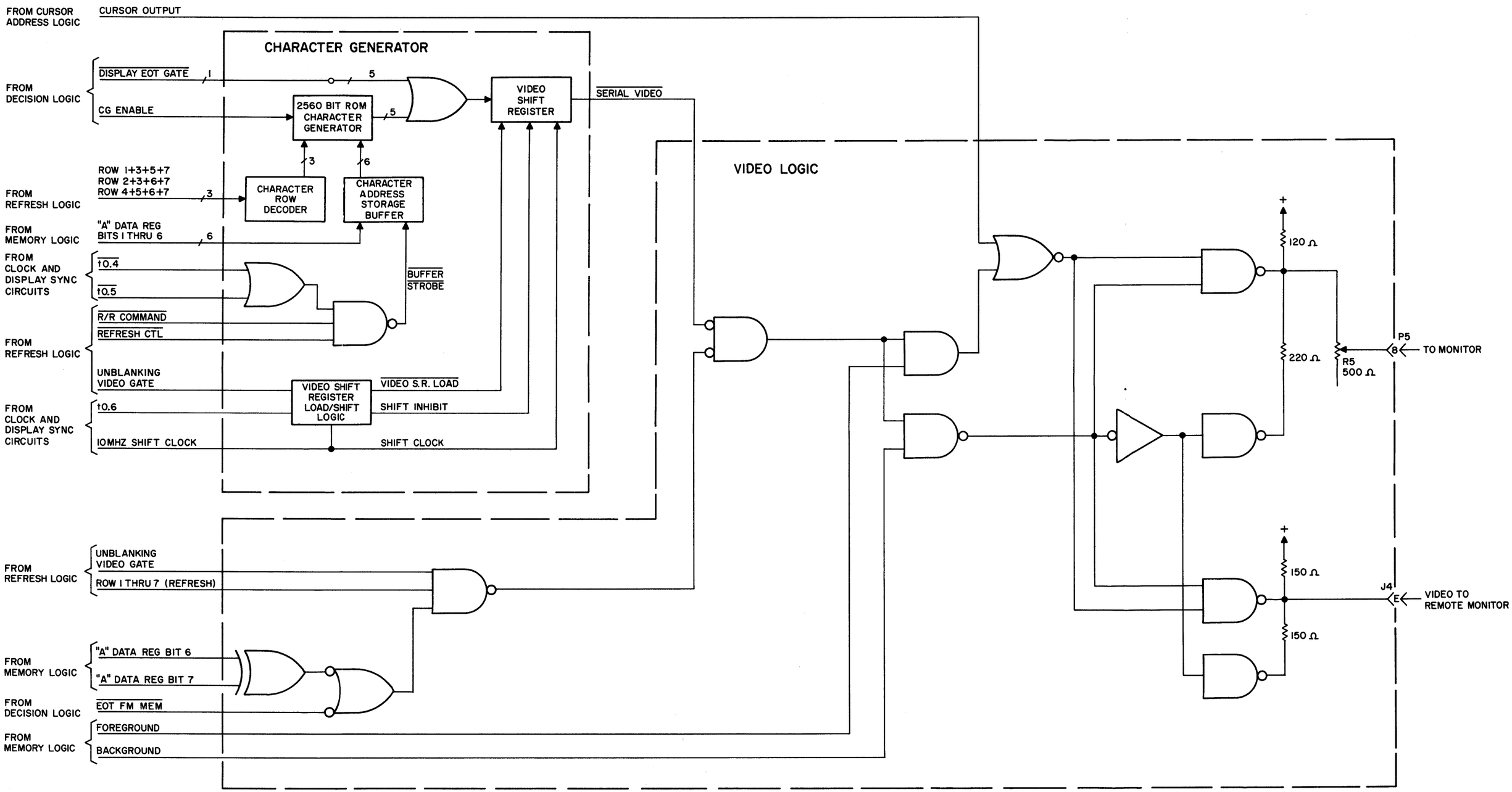
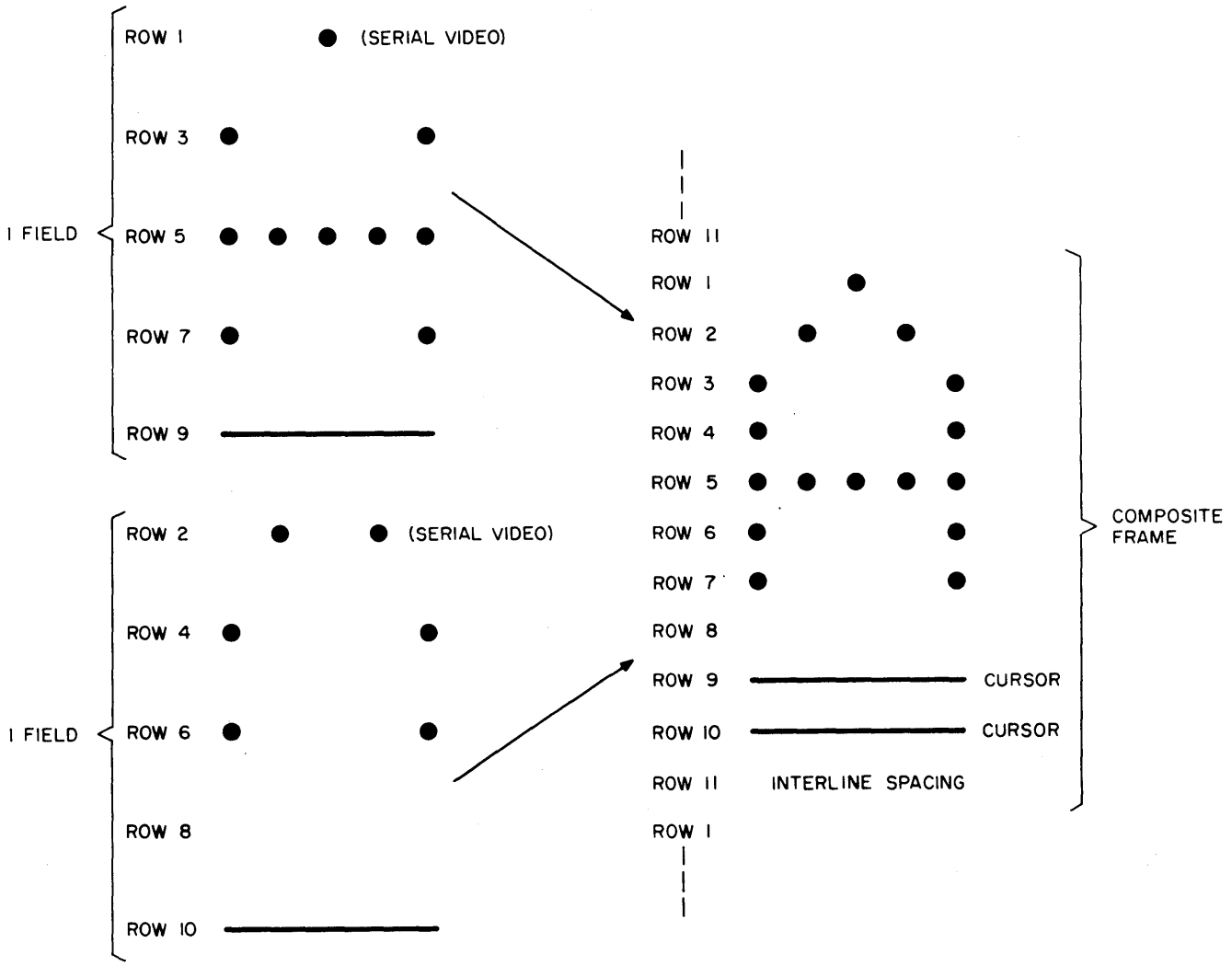


Figure 3-27. Character Video Logic Block Diagram



NOTE:
FIELD A AND FIELD B ARE INTERLACED TO PRODUCE A FRAME.

Figure 3-28. Character "A" and Cursor Development

A display frame is composed of two interlaced fields, field A and field B. The character generator receives, and stores a 6-bit character code from the Memory Logic, which is also the MSB's of the address in the ROM, of the character to be generated. The character address is held in a buffer register that is loaded by a command generated by $\overline{\text{R/R COMMAND}}$ and $\overline{\text{REFRESH CTL}}$ from the Refresh Logic and timing signal $\overline{\text{t0.4}}$ or $\overline{\text{t0.5}}$. A three-bit row address from the Refresh Logic row counter selects the row section of the characters to be generated. The output of the character generator is a 5-bit parallel data word that is applied to the video shift register thru an OR network. The additional input to the OR network is the decoded EOT symbol which is displayed as a 5x7 dot pattern for which all five dots in all seven rows are displayed. The video shift register load and shift signals are developed from timing signals 10MHz SHIFT CLOCK and t0.6 when enabled by the UNBLANKING VIDEO GATE from the Refresh Logic. The 5-bit parallel data word plus two space bits are shifted out to the video logic at a 10MHz rate.

3.6.10.2 Video Logic. The video logic accepts the $\overline{\text{CURSOR OUTPUT}}$ and $\overline{\text{SERIAL VIDEO}}$ signals which are the video information to be displayed. Control signals ROW 1 THRU 7 (REFRESH) and UNBLANKING VIDEO GATE, from Refresh Logic, are combined with additional control signals, A DATA REG BIT 6, A DATA REG BIT 7, FOREGROUND and EOT FM MEMORY, from the Memory Logic, which enable the video data and determine the brightness level. The video logic develops three levels of brightness; full brightness for foreground characters and cursor. Half brightness is developed for background or protected characters. Data locations in memory that contain blank characters or data for which no character symbols exist (i.e., CR), require no video output to be developed for their respective display locations. All memory data containing bits 6 and 7 as both ones or both zeroes will develop blanks, except the EOT symbol. For full brightness video the output levels of the output gates are high, for low level video the gate at the mid-point of the output voltage divider is high while the lower gate output is low. The output is then divided down by the ratio of the resistor values comprising the divider. For no video out the gate at the mid-point of the output voltage divider is low.

3.6.11 MONITOR CIRCUITS

The monitor is the display portion of the terminal, comprising the CRT, horizontal and vertical deflection circuits, a video amplifier and CRT control circuits. The CRT is a 12-inch, P39-phosphor tube with electrostatic focus which utilizes electromagnetic deflection to produce a horizontal raster scan. The beam is deflected by

the horizontal and vertical sync input signals while, simultaneously, the video data (serial) is presented to a video amplifier which modulates the beam intensity. The horizontal and vertical deflection circuits accept H DRIVE FOR MON and $\overline{\text{VERT DRIVE FOR MON}}$ input signals from the Clock and Display Sync Circuits to develop the sweep currents for the horizontal and vertical deflection yokes, respectively. The horizontal deflection circuit is a direct drive scanning circuit; i.e., the horizontal deflection signal is derived directly from the H DRIVE FOR MON input. The horizontal deflection circuit also develops the high voltage for the anode of the CRT, as well as the voltages for the video amplifier, focusing circuit, and CRT bias. The vertical deflection circuit uses a vertical oscillator to develop the vertical deflection signal. The vertical oscillator is triggered by the $\overline{\text{VERT DRIVE FOR MON}}$ pulses from the Clock and Display Sync circuits. The video from the Character Video Logic is applied to the video amplifier where it is amplified and dc coupled to the cathode of the CRT. The individual circuits of the monitor are described in the following paragraphs referenced to the monitor schematic shown in figure 3-29.

3.6.11.1 Vertical Deflection Circuit. Refer to figure 3-30 for the timing of the vertical deflection circuit. Silicon controlled rectifier (SCR) Q102 and its external circuitry form the vertical oscillator. When +15 volts dc is applied to the monitor circuits, capacitors C105 and C106 begin to charge exponentially through potentiometer R116 and resistor R115. Thus, the voltage at the anode of Q102 increases exponentially from zero volts towards +10 volts dc. Neglecting the $\overline{\text{VERT DRIVE FOR MON}}$ sync input for the moment, capacitors C105 and C106 continue to charge until the voltage reaches the firing point of Q102. Since Q102 is a complementary SCR (or unijunction), the device fires when the gate input is about 0.5 volts more negative than the anode (forward biased). With the voltage at the gate of Q102 fixed at approximately 5.5 volts dc, due to the voltage division of resistors R117, R118, R120, and the voltage drop across CR101, the SCR fires when the voltage at the anode reaches about 6 volts. Once the SCR fires, C105 and C106 begin to discharge through the cathode-anode junction of the SCR and resistor R120 to ground. Since the SCR appears as a virtual short circuit during conduction, the discharge time is primarily a function of the capacitance of C105 and C106 and the resistance of R120. In approximately 0.6 milliseconds the anode voltage decreases to about 1.7 volts and the current through the SCR becomes small enough to shut the SCR off. Capacitors C105 and C106 then begin to charge through R115 and R116 again to repeat the cycle. This produces a sawtooth voltage at the anode of Q102 whose

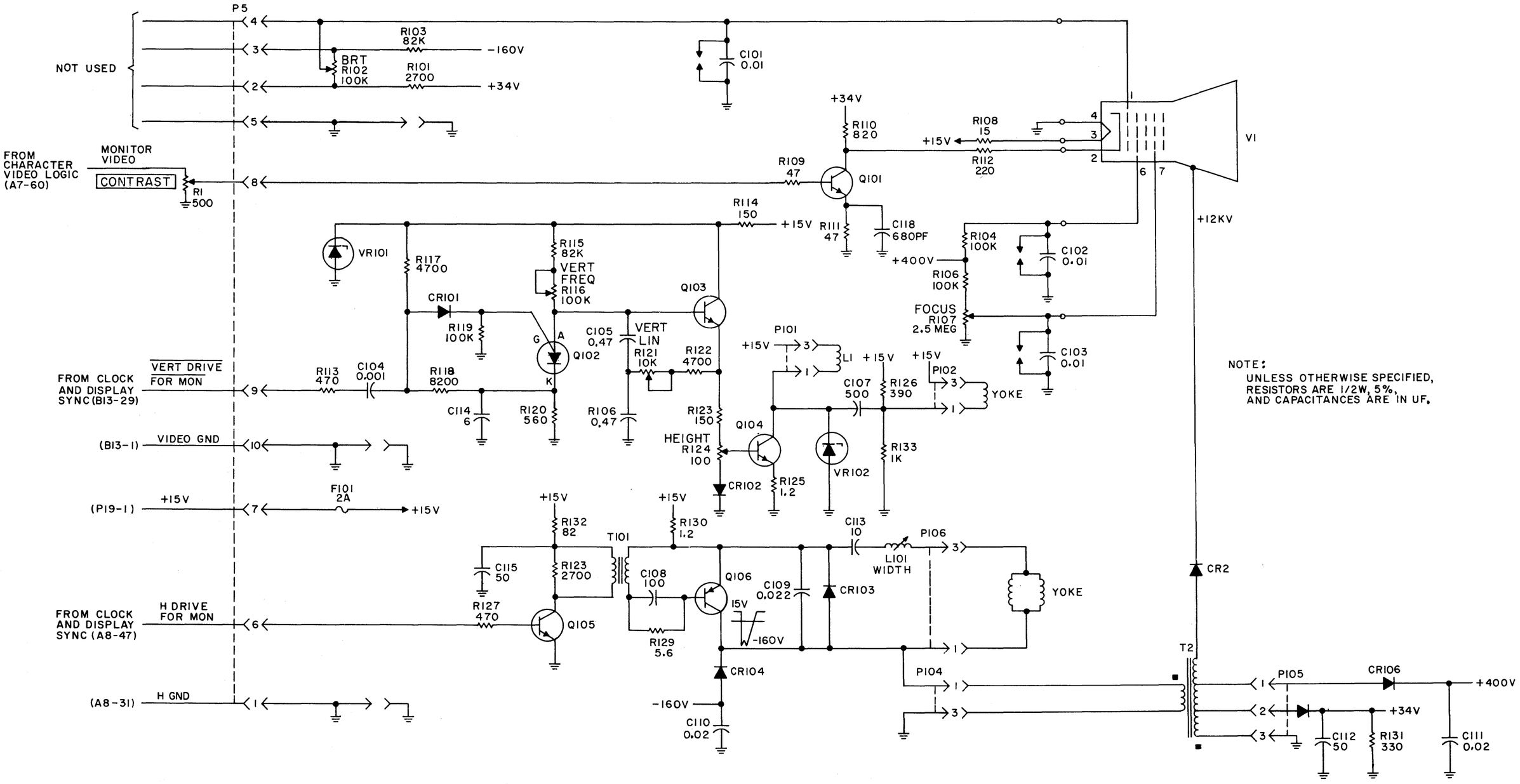
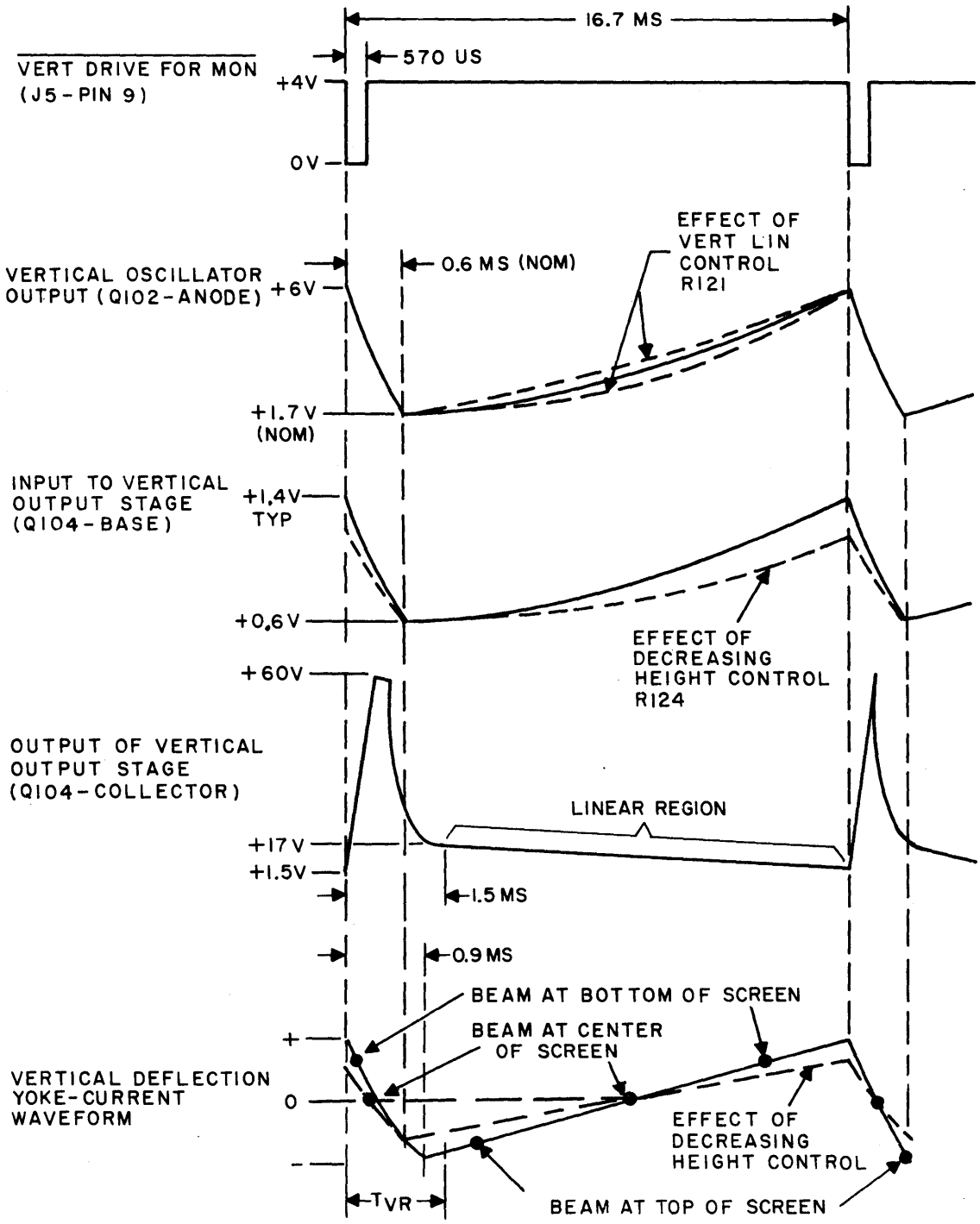


Figure 3-29. Monitor Circuits, Schematic Diagram



T_{VR} = TOTAL RETRACE TIME = 1.2 MS (INCLUDES DEFLECTION DELAY)

Figure 3-30. Vertical Deflection Circuit, Timing Diagram

amplitude is approximately 3.4 volts peak to peak. Vertical frequency potentiometer R116 adjusts the frequency (nominally 60 hertz of the sawtooth by controlling the charging rate of C105 and C106. Hence a basic free-running sawtooth is generated which can be triggered from an external source.

To synchronize the raster with vertical sync pulses of the terminal the vertical oscillator is triggered by the $\overline{\text{VERT DRIVE FOR MON}}$ pulses from the Clock and Display Sync Circuits. These pulses are applied to J5-9 of the monitor where they are coupled through resistor R113, capacitor C104, and diode CR101 to the gate of Q102. When the negative-going pulse is applied to R113, the gate of Q102 drops to about 3 volts to trigger the SCR (creating a forward biased anode). Thus, every 16.7 milliseconds Q102 is turned on by the $\overline{\text{VERT DRIVE FOR MON}}$ input to discharge capacitors C105 and C106. Since the frequency of the sawtooth is now a function of the vertical sync input, vertical frequency potentiometer R116 is adjusted to provide a free-running sawtooth period slightly greater than the period of the $\overline{\text{VERT DRIVE FOR MON}}$ input (16.7 ms). This ensures that the sawtooth voltage does not reach the firing point of the SCR before the trigger pulse is applied, providing a more stable oscillator.

The sawtooth voltage (parabolic) at the anode of Q102 is directly coupled to the base of driver amplifier Q103. Driver amplifier Q103 consists of two transistors internally connected as a darlington pair. Thus, the amplifier exhibits a high input impedance to the oscillator circuit while providing a low output impedance to drive vertical output transistor Q104. The sawtooth voltage at the output of Q103 is applied through resistor R123 and potentiometer R124 to the base of the vertical output transistor. In addition, the output of Q103 is also fed back to the junction of C105 and C106 of the oscillator circuit via resistor R122 and potentiometer R121. The feedback signal is used to modify the oscillator output waveform to improve the vertical linearity of the CRT raster. Potentiometer R121 functions as the vertical linearity control by controlling the amount of feedback current. The effect of the Vert. Lin. control on the oscillator output is indicated by the dashed lines on the vertical output waveform.

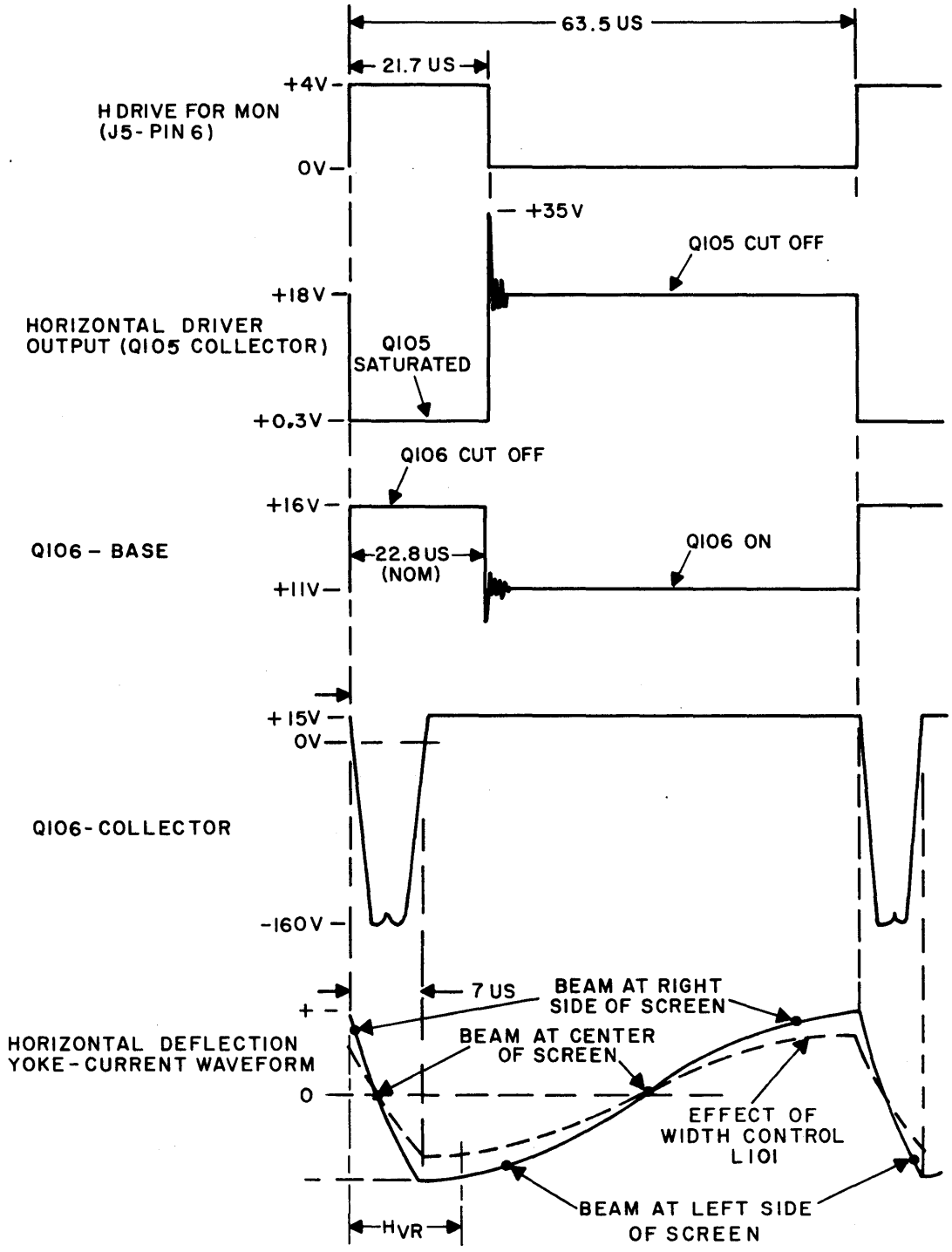
The sawtooth voltage at the base of Q104 controls the base current of the vertical output transistor and the conduction (current output). Height control R124 varies the peak amplitude of the sawtooth voltage, and hence, base current of transistor Q104 to control the vertical size of the raster. For simplicity, the discussion of the vertical output circuit will start with the sawtooth voltage at the base of Q104 increasing toward its peak value. During this interval, the current through

the deflection coil is increasing toward its peak positive value and the beam is being deflected downward towards the bottom of the screen. As the voltage at the base of Q104 continues to increase, the current through the transistor and the deflection yoke increases to deflect the beam further down, beyond the face of the CRT. During this period, current is also increasing through coil L1. When the VERT DRIVE FOR MON pulse triggers Q102, the voltage at the base of Q104 decreases sharply and the current through the transistor decreases proportionately. This sudden decrease in current is opposed by the inductance of L1 and a large positive pulse is developed at the collector of Q104 due to the counter emf of the coil. Since the coil is a relatively high impedance as compared to the yoke inductance, the positive pulse developed by L1 reverses the current through the deflection yoke to produce the vertical retrace. As the current through the yoke decreases from its maximum positive value, changes direction, and increases toward its maximum negative value, the beam deflects from beyond the bottom of the screen up past the top of the screen. At this point the sawtooth voltage at the base of Q104 starts to rise again. This causes the transistor to increase conduction and the increasing current through Q104 tries to reverse the direction of current flow through the deflection yoke. At the same time, the counter emf built up in the yoke during the retrace period opposes any change in current flow and tries to maintain current flow in the same direction. Therefore, the current through the yoke slowly starts to decrease from its maximum negative value. As the current through the yoke begins to decrease, the beam starts to move downward from the top of the screen. During the period transistor Q104 continues to increase conduction, the current through the deflection yoke linearly decreases to zero, reverses direction, and increases toward its maximum positive value again. This deflects the beam at a uniform rate downward past the bottom of the screen. When the vertical oscillator is triggered again, retrace is initiated, and the entire cycle repeats itself. The dots on the vertical deflection yoke-current waveform of figure 3.6.11.2 indicate the relative position of the beam with respect to the visual portion of the screen. Vertically, the raster actually extends above and below the face of the CRT (called "over-scan").

Capacitor C107 is a dc blocking capacitor which prevents the dc current of Q104 from passing through the vertical deflection yoke. Damping resistor R126 is connected across the yoke to prevent oscillations at the beginning of the sweep due to the high change of current during the retrace period. Resistor R133 provides a dc current path through the yoke to dc offset the vertical sweeps. This shifts the entire raster down to provide a more linear display.

3.6.11.2 Horizontal Deflection Circuit. Figure 3-31 illustrates the timing of the horizontal deflection circuit. As mentioned above, the horizontal deflection signal is derived directly from the H DRIVE FOR MON input pulses from the clock and display sync circuits. These pulses are applied through J5-6 of the monitor to the base of driver transistor Q106. The driver transistor is either cut off or driven into saturation by the signal to its base. When the base signal goes high, the transistor is turned on and its collector voltage is pulled down to about 0.3 volts. When the signal at the base is low, the transistor is cut off and its collector voltage is about +18 volts. Thus, a rectangular waveform appears at the collector of Q105, which is inverted with respect to the H DRIVE FOR MON input. The large positive spike in the horizontal driver output waveform is produced by the primary of transformer T101 when transistor Q105 is first cut off. The horizontal driver output is in turn, transformer coupled through T101 to the base of transistor Q106. Because of the phase reversal through T101, the rectangular waveform at the base of Q106 is in phase with the H DRIVE FOR MON input.

Transistor Q106 acts as a switch which is turned on or off by the rectangular waveform on its base to control the horizontal output circuit. For ease of understanding, the description of the horizontal output circuit will start when the current through the deflection coil is increasing in a positive direction and the beam is near the center of the screen, moving to the right. During this period, the signal at the base of Q106 is +11 volts and the transistor is in conduction. With Q106 conducting, the +15 volt supply voltage plus the charge built up in capacitor C113 during the negative half cycle of the current waveform causes yoke current to increase in a linear manner towards its maximum positive value. This moves the beam at a uniform rate to the right side of the screen. There, the signal at the base of Q106 goes to +16 volts. This turns the transistor off to initiate the horizontal retrace. With Q106 cut off, the currents through the deflection yoke and the primary of flyback transformer T2 start to decrease. As the current through the yoke decreases, the magnetic field built up during the horizontal sweep (trace) begins to collapse and the beam is deflected back towards the center of the screen. During this period a large negative pulse is developed at the collector of Q106 by the rapid collapsing fields of both the yoke and the primary of T2. This charges capacitor C109. The negative pulse reaches its peak amplitude when the beam approaches the center of the screen and the current through the deflection yoke is zero. Capacitor C109 now begins to discharge through the yoke to induce a current in the opposite direction. Thus, the beam is deflected to the left side of the screen to



H_{VR} = TOTAL RETRACE TIME = 9.5 US (INCLUDES DEFLECTION DELAY TIME)

Figure 3-31. Horizontal Deflection Circuit, Timing Diagram

complete the horizontal retrace. When capacitor C109 is fully discharged (collector of Q106 at zero volts), the current through the deflection yoke reaches its maximum value and subsequently starts to decrease. At this time the collapsing fields of the yoke and the primary of T2 try to produce a positive pulse at the collector of Q106. However, when the voltage at the collector starts to exceed +15 volts, damper diode CR103 is biased into conduction. Thus, the collector of Q106 is limited to +15 volts and the current created by the collapsing field of the yoke flows through CR103 to provide the first portion of the horizontal scan to charge capacitor C113. When the signal at the base of Q106 goes negative, the transistor conducts and the cycle starts over again.

Capacitor C113, in series with the yoke, also functions to block dc currents through the yoke and to provide "S" shaping of the current waveform. This "S" shaping compensates for stretching at the left and right sides of the screen due to the fact that the curvature of the CRT face and the deflected beam do not inscribe the same arc. Variable inductor L101 is an adjustable width control which allows a greater or lesser amount of current to flow through the deflection yoke, and therefore, varies the width of the raster.

The negative flyback pulses, developed during the horizontal retraces are rectified by diode CR104 and filtered by capacitor C110 to produce -160 volts dc for brightness control R102. These same pulses are transformer coupled to the secondary of T2 where they are rectified by diodes CR2, CR106, and CR105 to produce +12kvdc, +400 vdc, and +34 vdc, respectively. The +12 kvdc is the anode voltage for the CRT and the +400 vdc is applied through R104 to the screen grid and through R106 and Focus control R107 to the focusing grid of the CRT. The +34 vdc output is applied to R101 of the Brightness control circuit and R110 of the video amplifier stage.

3.6.11.3 Video Amplifier. The video from the Character Video Logic enters the monitor at J5-8 via the front panel CONTRAST control R1. The CONTRAST control varies the amplitude of the positive-going video pulses which are applied to the base of video amplifier Q101. Transistor Q101 is an inverting amplifier with a gain of about 17. The transistor remains cut off, maintaining the cathode of the CRT at +34 vdc, until a positive-going video pulse arrives at its base and turns the transistor on. This produces negative-going video pulses at the collector of Q101 to modulate the cathode of the CRT. The amplitude of the video at the CRT's cathode is controlled by the CONTRAST control which varies the signal at the base

of Q101. This controls the intensity at which the video is displayed on the screen. The overall brightness of the screen is controlled by the Brightness control which is described in the following paragraph.

3.6.11.4 CRT Control Circuits. The CRT control circuits consist primarily of the Brightness and Focus control circuitry. Brightness control R102 varies the control grid voltage between +30 and -70 vdc to increase or decrease conduction, and therefore, control the overall brightness of the raster. Varying the Focus control varies the voltage at the focusing grid between zero and +385 vdc. In addition, +400 vdc is applied through R104 to the screen grid of the CRT. The screen grid prevents interaction between the control and focusing grids and "screens" the electron beam for best deflection toward the phospor surface.

Filter capacitors C101, C102, and C103 are arc-gap ceramic type capacitors which provide built-in arc suppression to protect the transistor circuits when power to the CRT is turned on or off.

3.7 PROGRAM FUNCTION FLOW

This section describes how the control functions, initiated by means of the keyboard or from the CPU, are implemented by the major system logical functions described in paragraphs 3.4 and 3.6. The control functions are implemented under control of the Internal Program Counter which enables the proper logical functions in a predetermined sequence for a respective control function. The control functions described are:

A/N ENTRY (par. 3.7.1)
C/R (par. 3.7.2)
CURSOR MOVEMENT (par. 3.7.3)
CURSOR HOME (par. 3.7.4)
CLR/FG (par. 3.7.6)
INSERT LINE (I/L) (par. 3.7.6)
DELETE LINE (D/L) (par. 3.7.7)
INSERT CHARACTER (I/C) (par. 3.7.8)
DELETE CHARACTER (D/C) (par. 3.7.9)
XMIT (par. 3.7.10)
TAB (par. 3.7.11)
PRINT (par. 3.7.12)

RECORD (par. 3.7.13)
RECEIVE (par. 3.7.14)

3.7.1 A/N ENTRY (See figure 3-32)

Entry of A/N data from the Keyboard Input Logic or Input/Output Logic involves the Internal Program Counter and the Memory Logic.

The Internal Program Counter is enabled in the Batch mode by the $\overline{\text{ALLOWED KB STROBE}}$ signal, and in Receive mode by the $\overline{\text{PROCESS (I/O) ENTRY}}$ signal. Both of these signals occur when data is ready to be entered into memory. The keyboard-entered data (Batch mode) is available at the Data Register in the Memory Logic from the Keyboard Input Logic. In Receive, the data resides in the I/O registers of the Input/Output Logic and is available at the Data Register.

With the Internal Program Counter Entry logic enabled, the H DRIVE SET signal clocks the ENTRY CONTROL F/F enabling the "A" F/F. The "A" F/F is clocked by the t0.0 pulse, and enables the "B" F/F and clears the "B" Data Register by means of the $\overline{\text{CIR "B" DATA REG/ENTRY}}$ signal.

The next occurrence of a t0.0 pulse (700 ns later) clocks the "B" F/F which loads the "B" Data Register with the seven keyboard data bits by means of the LOAD B DATA REG/ENTRY signal. The eighth bit entered is a "1" which denotes a foreground character, as are all A/N keyboard-entered characters. At the same time, an ENABLE P-CTR-1 signal is developed which enables the first stage of the Internal Program Counter and allows it to operate upon arrival of the next $\overline{\text{t0.1}}$ clock pulse.

The seven stage Internal Program Counter is clocked along in sequence by $\overline{\text{t0.1}}$ pulses, and develops P-CTR-1 through P-CTR-7 pulses in sequence. Up to the development of P-CTR-1, the P-CTR Entry Logic operation is the same whether it is enabled by the keyboard or I/O stimulus. When in the Batch mode (keyboard stimulus), the data has already been entered into the "B" Data Register, and no further action must be taken with respect to that operation. In Receive, however, the data is in registers in the Input/Output Logic and must be entered into the "B" Data Register. When the Input/Output Logic received the data, from the CPU or by internal feedback in Half Duplex, a B DATA REG I/O ENABLE signal was developed. This signal, in conjunction with P-CTR-1, enables two gates, one of which receives a t0.4 pulse during the enabled period, and develops a $\overline{\text{CLEAR B DATA REG I/O}}$ signal which clears

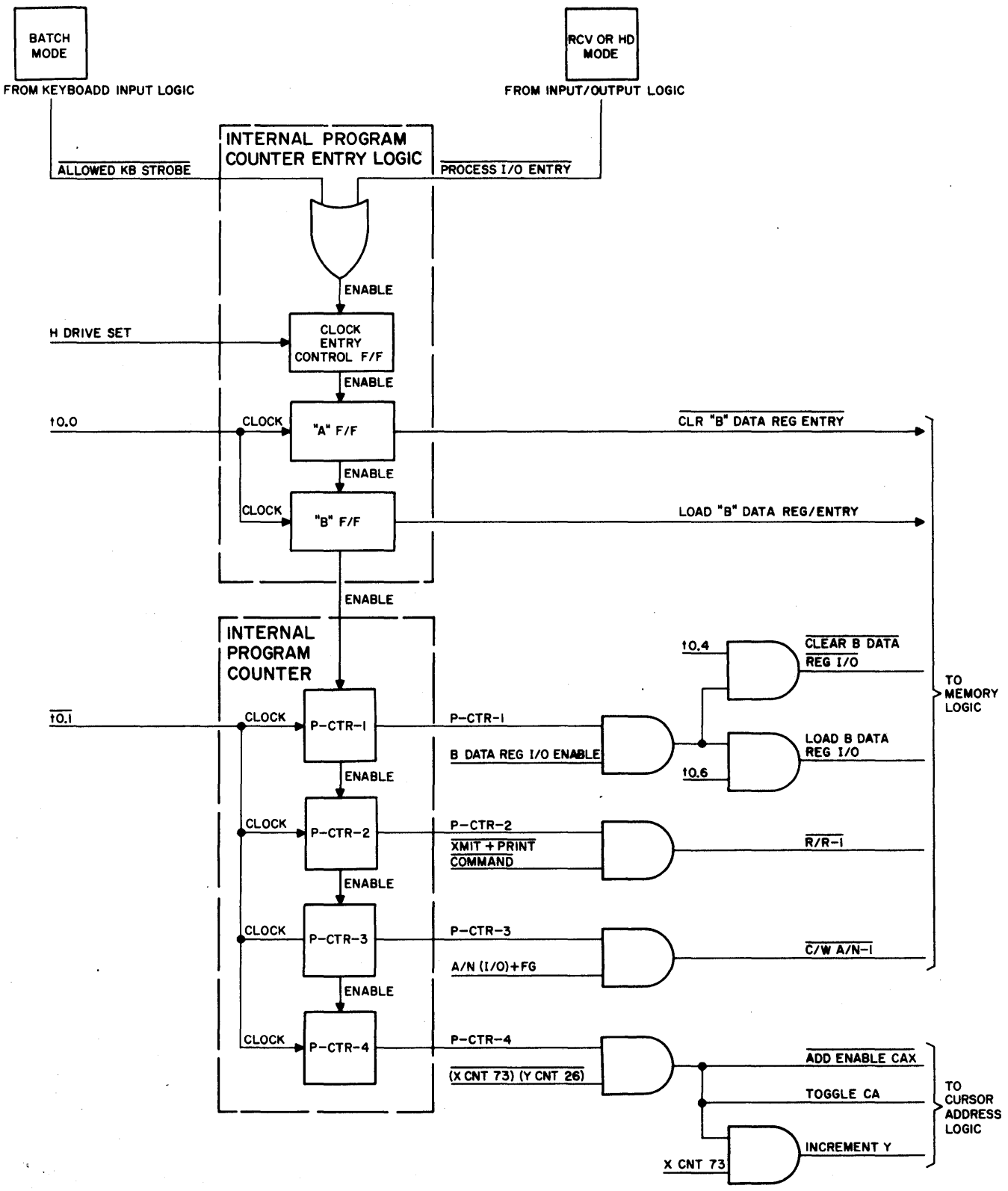


Figure 3-32. A/N ENTRY Logic Flow Diagram

the B Data Register. The other gate receives a t0.6 pulse and develops a LOAD B DATA REG I/O signal which allows the 8 bit I/O data to be loaded into the B Data Register.

The next Internal Program Counter sequence pulse, P-CTR-2, in conjunction with the $\overline{\text{XMIT + PRINT (COMMAND)}}$ signal, develops a read/restore signal $\overline{\text{R/R-1}}$. The $\overline{\text{R/R-1}}$ signal is generated primarily to determine whether the character data in memory is background or foreground data.

$(\text{P-CTR-3})^1$ is now enabled and, if the conditions A/N (I/O) + FG are met, a $\overline{\text{C/W A/N-1}}$ signal is developed. The condition FG must be met if the basic entry is from the keyboard, since $\overline{\text{FG}}$ (or BG) information is protected and not allowed to be modified by a keyboard entry. If the basic entry is from the I/O, it has the capability of modifying the memory information regardless of whether it is FG or BG. The $\overline{\text{C/W A/N-1}}$ signal is routed to the Memory Logic where it causes the contents of the "B" Data Register 8 bits, including FG or BG information, to be entered into memory.

The next operation involves incrementing the cursor to its next character position, unless the A/N Entry was just made at the X=73, Y=26 location, whereby no further cursor movement will take place. P-CTR-4 is developed and, if the signal $\overline{\text{(X CNT 73) · (Y CNT 26)}}$ is at its high logic level, signals TOGGLE CA and ADD ENABLE CAX are developed to increment the CA-X-CTR one count. If the last A/N Entry is made in X position 73, a $\overline{\text{CARRY}}$ signal is developed which in turn develops an INCREMENT Y signal, which also increments the CA-Y-CTR.

The A/N Entry has now been accomplished, and the Internal Program Counter proceeds through the remainder of its counts, P-CTR-5 thru P-CTR-7, which will have no effect.

3.7.2 C/R (See figure 3-33)

A C/R may be initiated by means of the keyboard in the Batch mode, or by the CPU in Full or Half Duplex, When the C/R is initiated by means of the keyboard, or by receipt of a C/R from the CPU or tape cassette, it is stored in the memory as a C/R and the cursor then performs a carriage return and line feed, but a blank is displayed on the terminal screen since no C/R symbol exists. When the Batch information is transmitted, the C/R is recognized by the CPU or peripheral equipment, and they respond accordingly.

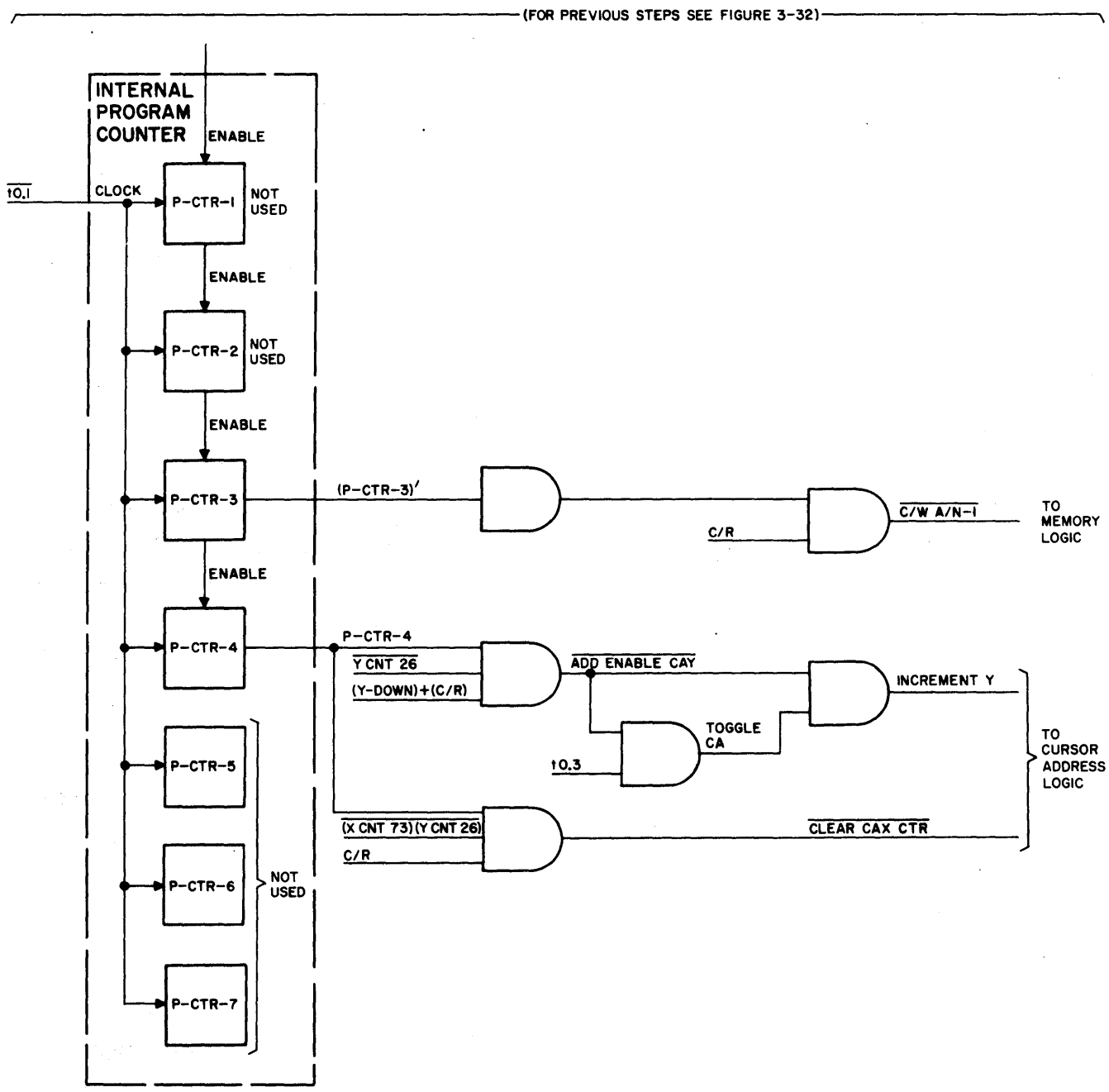


Figure 3-33. C/R Logic Flow Diagram

The initial enabling and entry sequence initiating the Internal Program Counter operation is the same as it is for A/N Entry (paragraph 3.7.1).

The first two counts developed in the Internal Program Counter, P-CTR-1 and P-CTR-2, perform the same function as for A/N Entry (paragraphs 3.7.1.4 and 3.7.1.5).

The Keyboard Input Logic develops a C/R signal from the Keyboard or from a CR/IO signal sent from the I/O logic, which enables $(\text{P-CTR-3})^1$ to develop a $\overline{\text{C/W A/N-1}}$ signal and enter the keyboard-initiated C/R into memory. $(\text{P-CTR-3})^1$ is inhibited when a CR/IO (CTL FCN) signal is present, indicating that the received CR was preceded by a lead-in code. In this case, cursor motion occurs, but no CR character is written into memory. When that memory location into which a CR has been written is sampled during the refresh cycle, the ASCII representation for a C/R is extracted from the memory, but the Bit 6 and 7 Decoder in the Character Video Logic causes that character interval to be blanked. Should a transmit or print operation be performed, the C/R ASCII code is transmitted.

The next operation involves moving the cursor down one line in the Y direction, unless it is already at line 26, and over to the X=0 position at the left-hand side of the display. The P-CTR-4 signal, in conjunction with the (Y-DOWN + C/R) and $\overline{\text{Y-CNT 26}}$ signals, develops TOGGLE CA and $\overline{\text{ADD ENABLE CAY}}$ signals to increment the CA-Y-CTR one count. This moves the cursor to the next lower line. P-CTR-4, along with the $\overline{\text{(X-CNT 73)} \cdot \text{(Y} \cdot \text{CNT 26)}}$ and the C/R signals, develops a $\overline{\text{CLEAR CAX CTR}}$ signal which resets the CA-X-CTR to zero and locates the cursor at the left-hand side of the screen.

The C/R function has been accomplished, and the Internal Program Counter proceeds through the remainder of its counts, P-CTR-5 through P-CTR-7, which will have no effect.

3.7.3 CURSOR MOVEMENT (↑ ↓ → ←) (See figure 3-34)

Cursor movement is the positioning of the cursor and primarily involves a keyboard entry. Each time the keyboard key is depressed the cursor moves one increment in the indicated direction until the display limits are reached. Should rapid movement of the cursor be desired, the RPT key should also be depressed with the desired direction key. The cursor will step along in the selected direction at a rate of approximately 15 steps per second. A CPU command Backspace Cursor will result in a signal BACKSPACE REC'D which will be interpreted by the logic as an

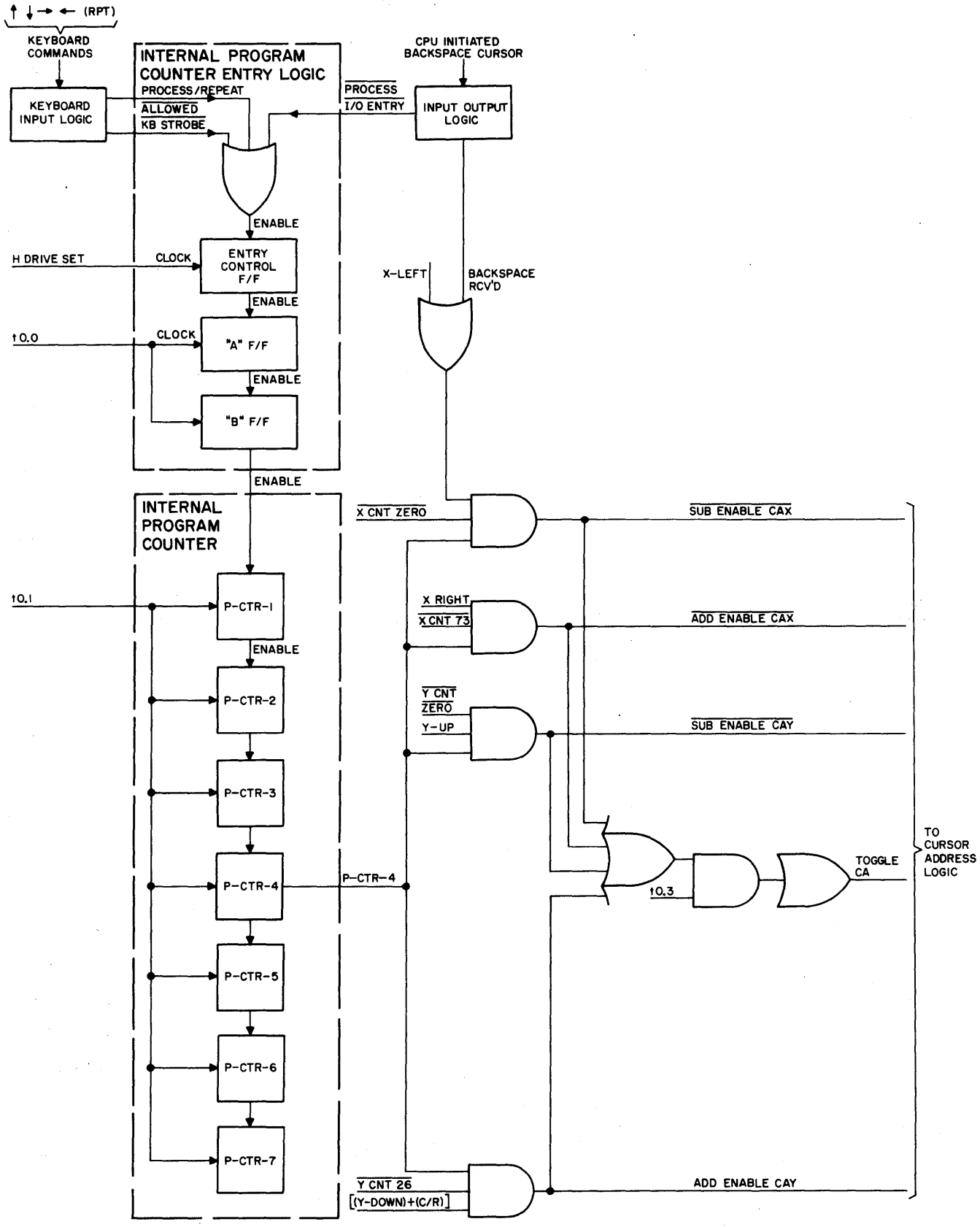


Figure 3-34. CURSOR MOVEMENT Logic Flow Diagram

X-LEFT signal. The CPU can address the cursor to any location on the display but the manner in which this is accomplished does not involve the Internal Program Counter. The Cursor Address Logic receives cursor address instructions from the Input/Output Logic under CPU stimulus.

The initial enabling and entry sequence initiating the Internal Program Counter operation is the same as it is for A/N Entry (paragraphs 3.7.1.1 thru 3.7.1.3).

The first two counts developed in the Internal Program Counter P-CTR-1 and P-CTR-2 perform the same function as for A/N Entry (paragraph 3.7.1.4 and 3.7.1.5).

P-CTR-3 is not used.

The implementation of positioning the cursor is essentially the same for any of the selected directions, therefore, only one direction, X=Left, is discussed. Figure 3-34 indicates all of the logical paths. The positioning sequence is enabled by P-CTR-4, X-LEFT and $\overline{\text{X-CNT ZERO}}$ signals which develop $\overline{\text{SUB ENABLE CAX}}$. $\overline{\text{SUB ENABLE CAX}}$ subtracts one count from the CA-X-CTR when TOGGLE CA clocks the CA-X-CTR. TOGGLE CA is developed by $\overline{\text{SUB ENABLE CAX}}$ and timing signal t0.3. The cursor address CA-X is now one count less than it was at the start of the sequence.

Cursor movement has been accomplished and the Internal Program Counter proceeds through the remainder of its counts, P-CTR-5 thru P-CTR-7, which will have no effect.

When a repeated movement is desired the RPT key is depressed, in addition to the direction key; enabling the Repeat Counter which recycles the Internal Program Counter until the desired cursor destination is achieved and both keys are released.

3.7.4 CURSOR HOME (HOME) (See figure 3-35)

The Cursor Home command function positions the cursor at the first character position, (x=0, y=0), on the display. When this function is initiated by the keyboard, the Internal Program Counter will implement the positioning by developing the $\overline{\text{CLEAR X AND Y CA-1}}$ signal. When the CPU initiates this function two signals $\overline{\text{CLEAR CAX/IO}}$ and $\overline{\text{CLEAR CAY/IO}}$ are developed and reset the cursor address counters; the Internal Program Counter is not required.

The Internal Program Counter is enabled thru the same sequence given for keyboard entry of an A/N Entry (Paragraph 3.7.1).

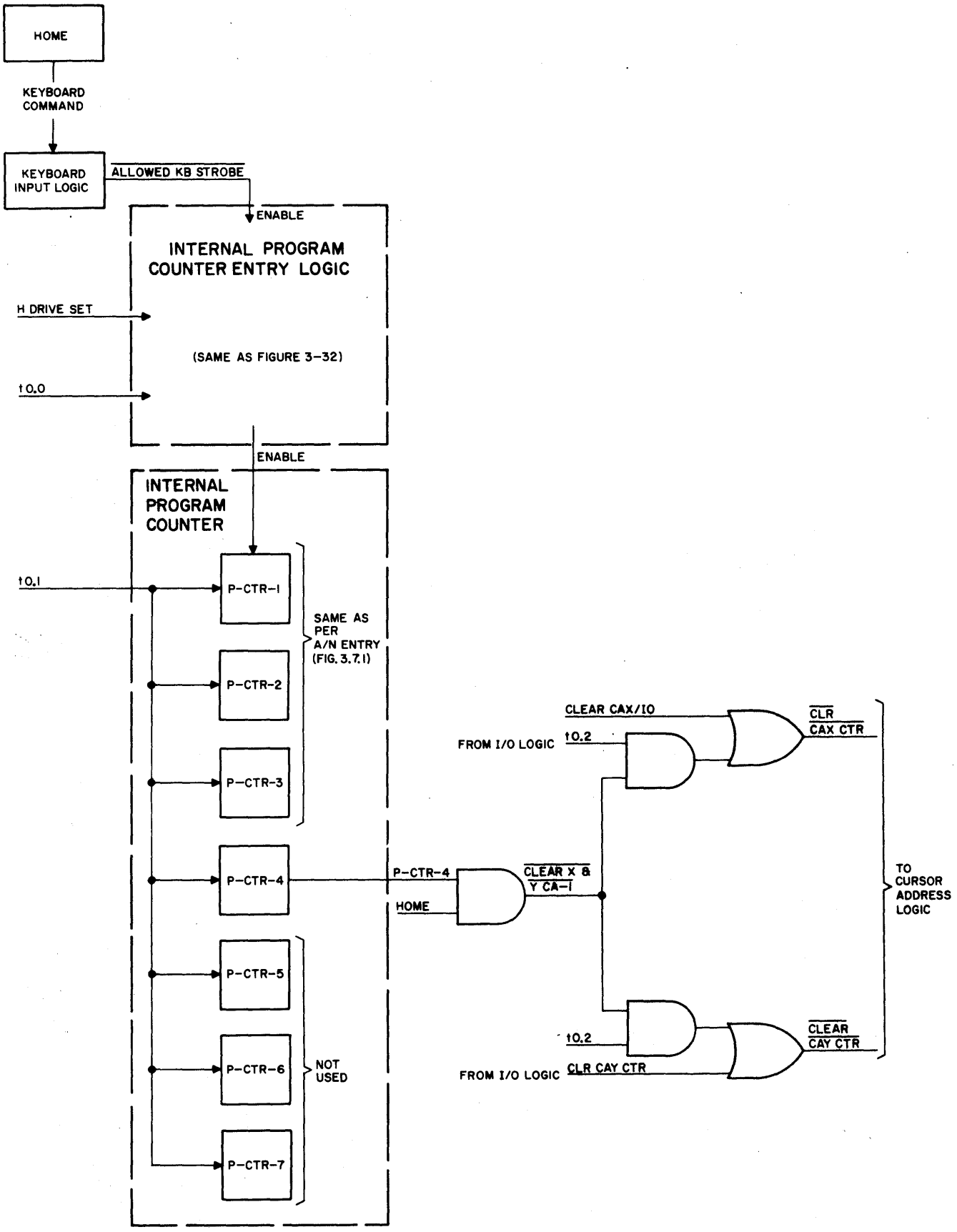


Figure 3-35. CURSOR HOME Logic Flow Diagram

The first three counts of the Internal Program Counter, P-CTR-1 thru P-CTR-3 do not affect this function.

P-CTR-4 in conjunction with the HOME signal develops $\overline{\text{CLEAR X AND Y CA-1}}$ which when clocked by to.1 develops two signals $\overline{\text{CLR CAX CTR}}$ and $\overline{\text{CLR CAY CTR}}$. The Home signals $\overline{\text{CLEAR CAX/IO}}$ and $\overline{\text{CLEAR CAY/IO}}$ from the Input/Output Logic enter their respective counters with $\overline{\text{CLR CAX CTR}}$ and $\overline{\text{CLR CAY CTR}}$ thru two OR gates.

The Cursor Home function has now been accomplished and the Internal Program Counter proceeds thru the remainder of its counts, P-CTR-4 thru P-CTR-7, which have no effect.

3.7.5 CLEAR FOREGROUND (CLR/FG) (figure 3-36)

The clear foreground function involves clearing the screen of foreground, or bright intensity, characters only. Background, or low intensity, characters are protected and are not cleared. This function is extremely useful in repetitive "form fill-out" operations. The Clear function is implemented in exactly the same manner as Clear Foreground except that the foreground test is eliminated and all characters are cleared. When either function is desired the SHIFT key must be depressed and then the respective CLEAR or CLR/FG key. The implementation of this function involves sampling every memory location and will take more time than is available during the normal refresh retrace period. The refresh cycle is interrupted by the generation of a SKIP REFRESH CYCLE signal. The cursor is initially sent to the HOME (x=0, y=0) position, memory locations are sampled for foreground status (meaningful in CLR/FG only), and characters are cleared. When a character is cleared from memory it is replaced by a blank character which is the result of a C/W-ZERO command. When the clear step has been performed the cursor is incremented along to the next position and the memory sampled. This process continues recycling P-CTR-3 thru P-CTR-5 until the cursor is again at the HOME position and the clearing function has been completed. The CLEAR and CLR/FG functions may be initiated by keyboard entry or the CPU. The implementation of the functions are essentially identical regardless of the initial stimulus. The TOTAL CLEAR and PARTIAL CLEAR are developed from $\overline{\text{TOTAL CLR/IO}} = \overline{\text{CLR FULL FM KB}}$ and $\overline{\text{PARTIAL CLR/IO}} + \overline{\text{CLR PART FM KB}}$ respectively.

The initial enabling and entry sequence initiating the Internal Program Counter operation is the same as it is for A/N Entry (paragraph 3.7.1).

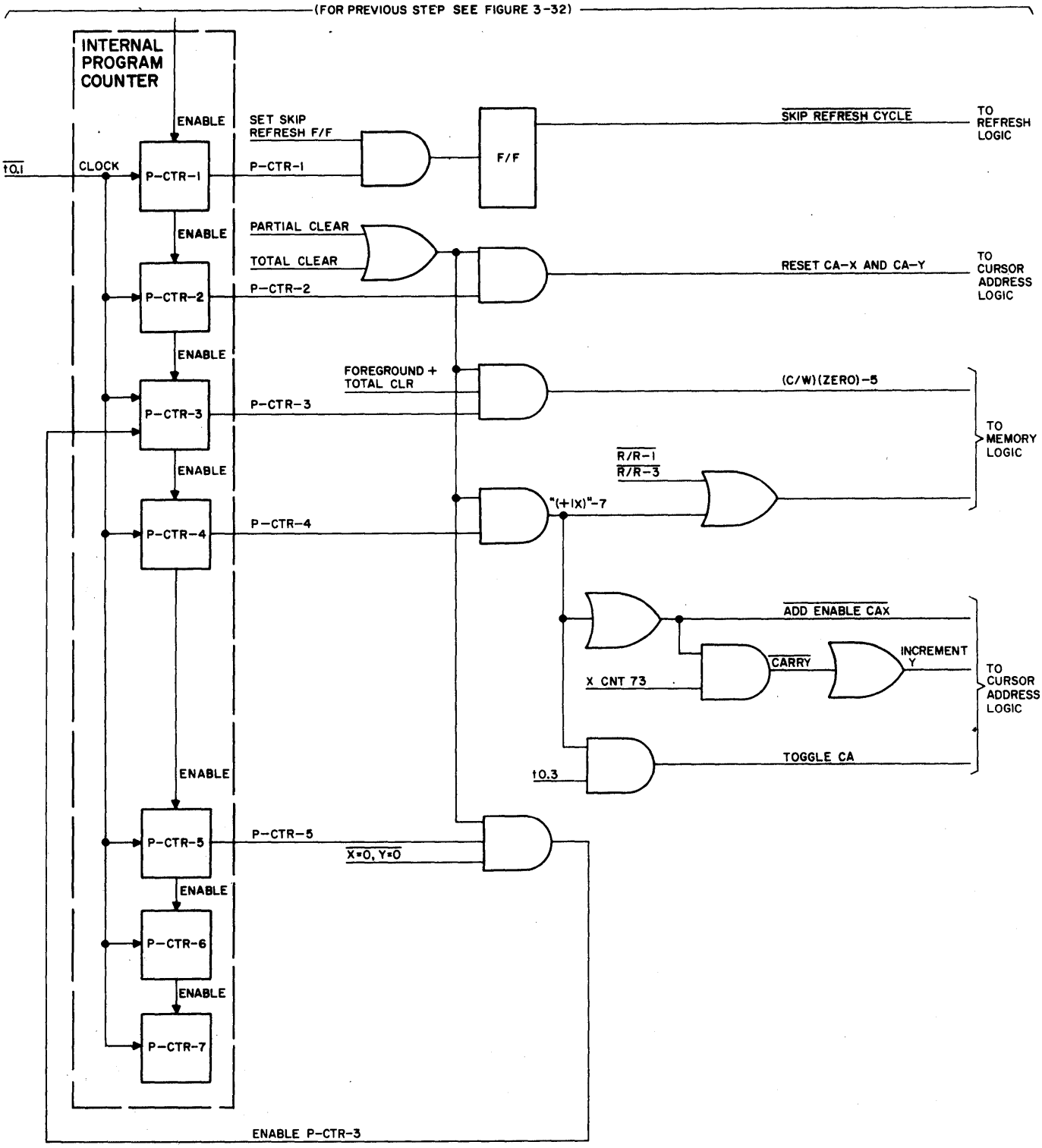


Figure 3-36. CLEAR FOREGROUND Logic Flow Diagram

The first two counts developed in the Internal Program Counter, P-CTR-1 and P-CTR-2, perform the same function as for A/N Entry (paragraph 3.7.1) and in addition, P-CTR-1 enables the SET SKIP REFRESH F/F signal. The SET SKIP REFRESH F/F signal causes the $\overline{\text{SKIP REFRESH CYCLE}}$ signal to be developed and sent to the Refresh Logic. P-CTR-2 enabled by PARTIAL CLEAR (or TOTAL CLEAR in the Clear function) resets the CA-X and CA-Y counters which position the cursor at the HOME position.

P-CTR-3 in conjunction with FOREGROUND · PARTIAL CLEAR, or TOTAL CLEAR in clear, generates clear-write signal $\overline{\text{(C/W) (ZERO)-5}}$ which writes a blank character in the memory at that respective location.

P-CTR-4 in conjunction with PARTIAL CLEAR, or TOTAL CLEAR in clear, generates the CA-X counter incrementing signal (+1X)-7, and read-restore signal $\overline{\text{R/R-3}}$. When the CA-X Counter reaches a count of 73, the $\overline{\text{CARRY}}$ signal is developed which increments the CA-Y counter.

P-CTR-5 in conjunction with $\overline{\text{(X=0) (Y=0)}}$ and PARTIAL CLEAR, or TOTAL CLEAR in clear, enables P-CTR-3. P-CTR-3, P-CTR-4 and P-CTR-5 recycle performing their respective logical operations until the condition x=0, y=0 occurs which inhibits further P-CTR-3 operation.

P-CTR-6 and P-CTR-7 have no effect in the implementation of this function.

3.7.6 INSERT LINE (I/L) (See figure 3-37)

The Insert Line function may be initiated by means of a keyboard entry or by the CPU. The implementation of the function is identical regardless of which stimulus initiates the function. To initiate the Insert Line function by means of the keyboard, the cursor must be positioned at the desired y location, where new line data is to be inserted, and the I/L key depressed. The data lines below the cursor will roll down one line. The line where the cursor is positioned will also roll down one line and a line of blank characters will be written into those line locations. Blank characters are written into the memory by C/W ZERO command and will be refreshed as blank characters until the new data is written in. The

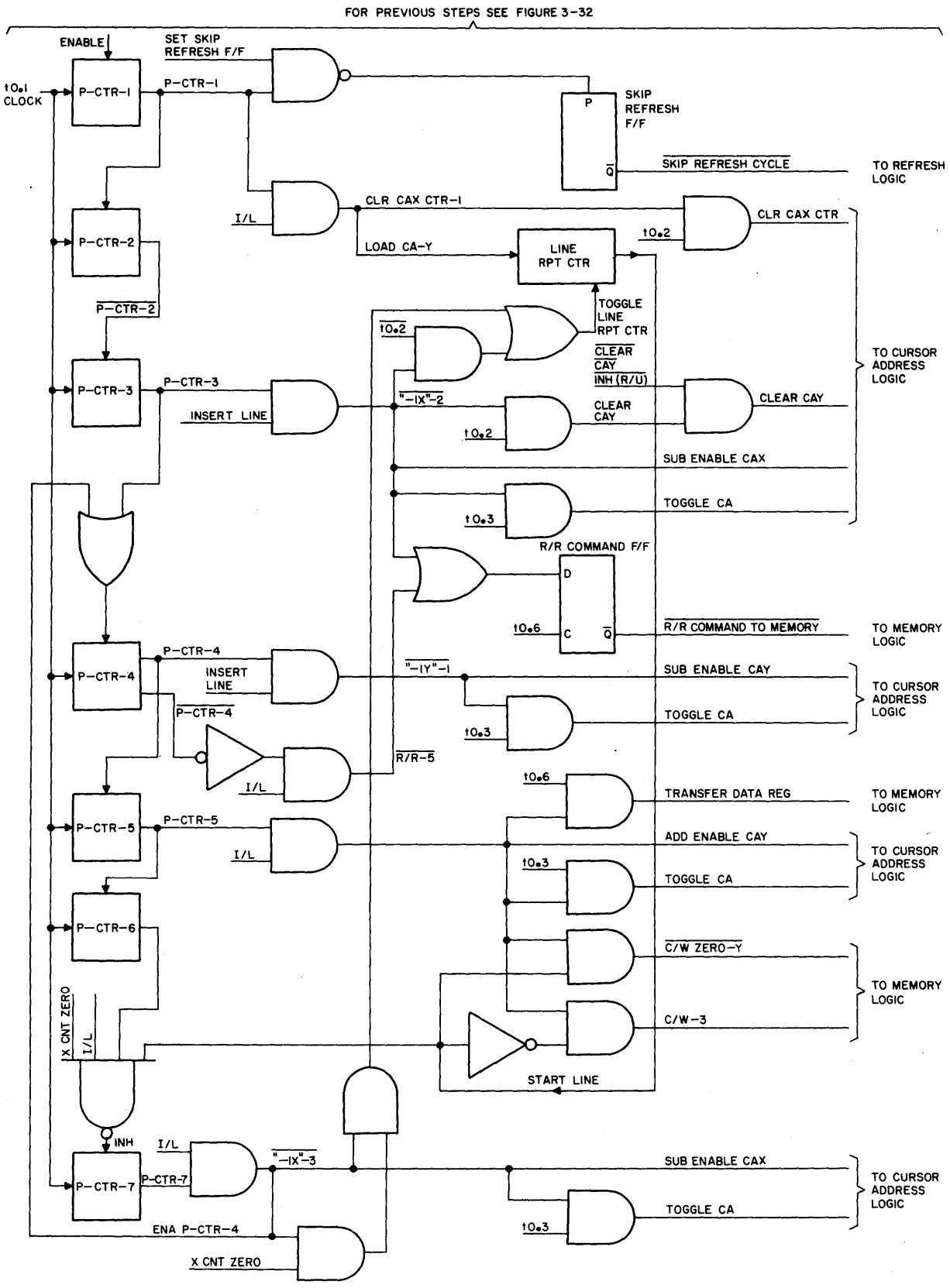


Figure 3-37. INSERT LINE Logic Flow Diagram

implementation of the function involves replacing character lines with the information contained in the line just above it, character by character until the cursor location is reached and a blank line results. The last line of data y=26 is lost being replaced in memory by the data of line y=25 or, if y=26 is the selected location for the new line, a line of blank characters.

The initial enabling and entry sequence initiating the Internal Program Counter operation is the same as it is for A/N Entry (paragraph 3.7.1).

The first count developed in the Internal Program Counter P-CTR-1, performs the same function as it does for CLR/FG (paragraph 3.7.5). In addition, P-CTR-1 in conjunction with the INSERT LINE signal which is derived from $\overline{(I/L)}$, from the keyboard, or $\overline{\text{INSERT LINE (IO)}}$ generates $\overline{\text{CLR CAX CTR-1}}$. $\overline{\text{CLR CAX CTR-1}}$ in conjunction with timing signal t0.2 generates the $\overline{\text{CLEAR CAX CTR}}$ signal which clears the CA-X counter to zero. $\overline{\text{CLEAR CAX CTR-1}}$ in conjunction with $\overline{\text{INSERT LINE}}$ loads the y location of the cursor into the Line Repeat Ctr.

P-CTR-2 develops $\overline{\text{R/R-1}}$ which has no significance with regard to this function.

P-CTR-3 enables several sequential signals which occur in the following sequence.

P-CTR-3 in conjunction with the INSERT LINE signal produces the $\overline{\text{"-1X"-2}}$ signal. $\overline{\text{"-1X"-2}}$ in conjunction with timing signal to.2 and $\overline{\text{CLEAR CAY INH (R/U)}}$ generates CLEAR CAY CTR which clears the CA-Y counter to zero. $\overline{\text{"-1X"-2}}$ and timing signal $\overline{\text{t0.2}}$ generate a signal to toggle the Line Repeat Ctr. $\overline{\text{"-1X"-2}}$ develops into the $\overline{\text{SUB ENABLE CAX}}$ signal which subtracts one increment from the CA-X counter when the TOGGLE CA signal occurs. TOGGLE CA is developed when the timing signal to.3 is enabled by signal $\overline{\text{"-1X"-2}}$. This subtraction of one count from the CA-X counter, which was reset to zero by P-CTR-1, results in the CA-X counter containing a full count. When SUB ENABLE CAX occurs with the CA-X count equal to zero, a $\overline{\text{BORROW IN X CTR}}$ signal is developed which develops DECREMENT Y signal which subtracts one increment from the CA-Y counter when toggled by TOGGLE CA. Subtracting one count from the CA-Y counter, which was cleared to zero, results in the CA-Y counter containing a full count of 31. Counts greater then 26 in the CA-Y counter will have no visible significance since neither the display or the memory recognize these counts. The insert line logic, however, will go thru the specified routine until count 26 is reached where visible valid operations

commence. P-CTR-3 in conjunction with the INSERT LINE signal produces signal $\overline{\text{R/R-4}}$ gating the R/R Command F/F which when clocked by timing signal to.6 generates $\overline{\text{R/R COMMAND TO MEMORY}}$. This R/R command has no significance with regard to this function.

P-CTR-4 with INSERT LINE develops a "-1Y"-1 signal which results in the CA-Y counter being decremented by one count. Assuming the cursor is now positioned on line 25, the character on line 25 must be read out of memory and stored to be written into the same X location on line 26. $\overline{\text{P-CTR-4}}$ in conjunction with INSERT LINE and timing signal to.6 develops $\overline{\text{R/R-5}}$ which reads data out of memory corresponding to the cursor location. P-CTR-5 will generate signals which will write the proper data into memory.

P-CTR-5 and INSERT LINE enable several sequential signals which occur in the following sequence. P-CTR-5 and INSERT LINE develops the (XFER DATA REG-2), +1Y-2 signal which generates the $\overline{\text{ADD ENABLE CAY}}$ signal, to increment the CA-Y counter one count, and in conjunction with to.3 generates the TOGGLE CA signal. The CA-Y counter is incremented one count and the information stored for the display location just above this one will be written in. (XFER DATA REG-2), +1Y-2 in conjunction with to.6 generates the $\overline{\text{TRANSFER DATA REG}}$ signal which transfers the data in the "A" Data Register into the "B" Data Register. (XFER DATA REG-2), +1Y-2 in conjunction with the START LINE INSERT COINC signal from the Line Repeat Ctr enables a $\overline{\text{C/W-3}}$ signal or $\overline{\text{C/W ZERO-4}}$ signal. If the start line, the original Y position of the cursor, has not been reached, $\overline{\text{C/W-3}}$ will write the data into memory from the "B" Data Register. If the start line has been reached and the START LINE INSERT COINC is at its high logic level $\overline{\text{C/W ZERO-4}}$ is developed which writes a blank character into memory. Either $\overline{\text{C/W-3}}$ or $\overline{\text{C/W ZERO-4}}$ will enable the C/W Command F/F which is clocked by timing signal to.6.

If the routine is on the start line and the CA-X count is zero, the Insert Line function has been completed. P-CTR-6 does not enable P-CTR-7 and the routine will cease. If these conditions have not been achieved P-CTR-7 will be enabled and the routine continued.

P-CTR-7 and INSERT LINE develop $\overline{\text{("-1X"-3)}}$, $\overline{\text{ENA P -CTR-4-3}}$ which enables several sequential signals in addition to enabling P-CTR-4. $\overline{\text{(-1"1X"-3)}}$, $\overline{\text{ENA P-CTR-4-3}}$ develops $\overline{\text{SUB ENABLE CAX}}$ which enables the subtract gating in the CA-X counter. If the count in the CA-X counter is zero, $\overline{\text{SUB ENABLE CAX}}$ and $\overline{\text{X CNT ZERO}}$ develop $\overline{\text{BORROW IN X CTR}}$ which results in the TOGGLE RPT CTR/LINE INS and DECREMENT Y

signals. Toggle RPT CTR/LINE INS enables timing signal to.2 to clock the Line Repeat Ctr one count. When timing signal to.3 occurs it is enabled to develop a TOGGLE CA signal which decrements the CA-X counter and, if the DECREMENT Y signal is present also decrements the CA-Y counter.

3.7.7 DELETE LINE (D/L) (See figure 3-38)

The Delete Line (D/L) function may be initiated by means of the keyboard or the CPU. To initiate this function from the keyboard the cursor must be positioned below the line to be deleted and D/L key depressed. The line just above the cursor will be deleted and all lower lines will roll up leaving a line of blank characters in the last line.

The initial enabling and entry sequence initiating the Internal Program Counter operation is the same as it is for A/N Entry (paragraph 3.7.1).

The first count of the Internal Program Counter performs the same functions as for Insert Line (I/L) except that the Y location of the cursor is not set into the Line Repeat ctr. and the CA-X counter is reset. $\overline{\text{CLR CAX CTR-1}}$ is enabled by the coincidence of P-CTR-1 and (D/L) + (I/L). $\overline{\text{CLR CAX CTR-1}}$ enables timing signal t0.2 which develops $\overline{\text{CLR CAX CTR}}$ which clears the CA-X counter.

$\overline{\text{P-CTR-1}}$ enables P-CTR-2 and resets the Last Line F/F.

P-CTR-2 in conjunction with the D/L signal develops (P2) (EXPANSION + COMPRESSION + D/L) which generates $\overline{\text{LOAD Y COIN CTR}}$ and $\overline{\text{LOAD X COIN CTR}}$. These two signals load the C.A.Y. Coincidence Counter and the C.A.Y. Coin. Ctr. with the complements of the numbers in the CA-Y and CA-X counters respectively for storage of the cursor location until the routine is completed. P-CTR-2 in conjunction with $\overline{\text{XMIT + PRINT (COMMAND)}}$ also develops a R/R-1 signal which performs no function in this routine.

$\overline{\text{P-CTR-2}}$ enables P-CTR-3

P-CTR-3 in conjunction with D/L enables the following sequential signals. P-CTR-3 and D/L generates a "+1Y"-1 signal which enables timing signal t0.2 in an attempt to toggle the Last Line F/F. If the Y count is 26 the Last Line F/F toggles, if the Y count has not reached 26 the Last Line F/F remains in its reset state. "+1Y"-1 develops $\overline{\text{ADD ENABLE CAY}}$ which enables timing signal t0.3 to generate a TOGGLE CA to advance to CA-Y counter one count. This removes the cursor to the

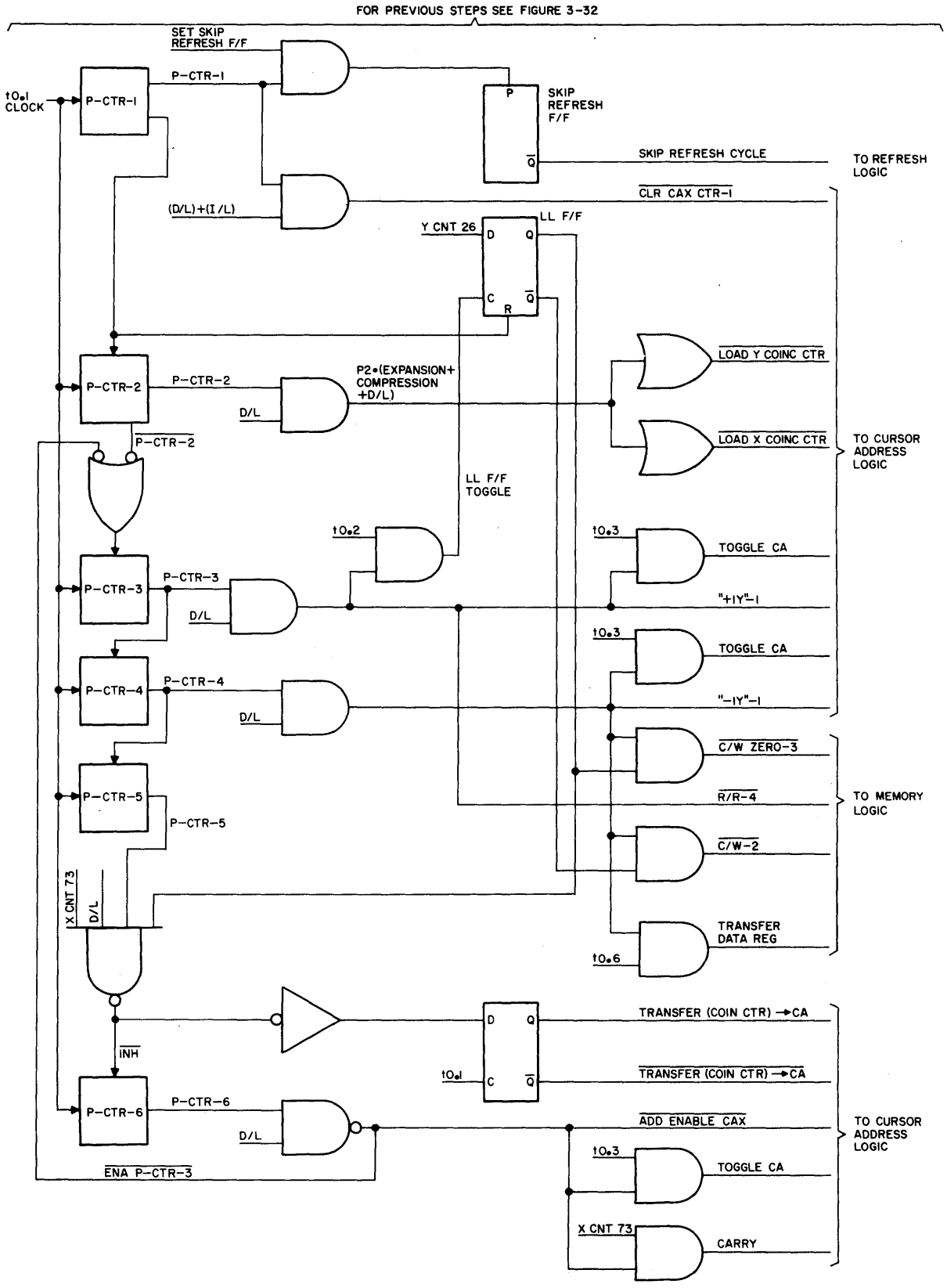


Figure 3-38. DELETE LINE Logic Flow Diagram

line just below so that data may be extracted from memory and later written into the character space just above the one read. P-CTR-3 and D/L generate R/R-4 which enables timing signal t0.6 to generate the R/R COMMAND TO MEMORY signal. The data at this respective location is read out of memory and stored.

P-CTR-4 and D/L enables the following sequential signals. "-1Y"-1 develops SUB ENABLE CAY which enables timing signal to.3 to generate a TOGGLE CA to decrement the CA-Y counter one count. This moves the cursor up to the next line, just above the position from which the data was extracted, where the data is to be written in memory. TRANSFER DATA REG. enables timing signal to.6 which transfers the data from the "A" Data Register into the "B" Data Register. If the Last Line F/F is still in its reset condition a C/W-2 signal is developed which in conjunction with timing signal t0.6 generates a C/W COMMAND. C/W COMMAND writes into memory the contents of the "B" Data Register. If the Last Line F/F has been clocked due to a Y CNT 26, or the last line has been reached, a (C/W)(ZERO)-3 signal is developed which in conjunction with timing signal to.6 generates a C/W COMMAND. C/W COMMAND now writes a blank character into memory.

P-CTR-5 and D/L are two of the four signals required to enable P-CTR-6. P-CTR-6 will be enabled by INH P-CTR-6 ENA-1 as long as the x=73 and y=26 condition has not been achieved. When the last blank character has been written, x=73, in line 26, P-CTR-6 will be inhibited. The same signal that inhibits P-CTR-6 is also (TRANS (COIN CTR) → CA)-2 which enables the XFER ADDR F/F and allows it to be clocked by timing signal to.1. The XFER ADDR F/F generates two signals TRANSFER (COIN CTR → CA and TRANSFER (COIN CTR.) → CA. TRANSFER (COIN CTR.) → CA enables timing signal to.2 which develops CLEAR CAY CTR and CLEAR CAY CTR which clear out both cursor address counters. TRANSFER (COIN CTR) → CA enables timing signal to.4 which transfers the original CA-X counter data from the CA-X Coinc Ctr to the CA-X counter. TRANSFER (COIN CTR) → CA also develops D/L(R/U) • XFER COIN→CA which enables timing signal to.4 generating PRESET LOAD CAY CTR. transfers the original CA-Y counter data from the C.A.Y. Coincidence Counter to the CA-Y counter.

As long as P-CTR-6 is enabled with the D/L signal present, ENA P-CTR-3 is generated, enabling P-CTR-3. The same signal is also "+1X"-6 which develops ADD ENABLE CAX and in conjunction with timing signal to.3 generates TOGGLE CA. This increments the CA-X counter one count. When X COUNT 73 is developed a CARRY

signal is generated which automatically develops an INCREMENT Y signal which adds once count to the CA-Y counter. Stages P-CTR-3 through P-CTR-6 recycle until the last line and last character have been completed and the routine is terminated as described above.

3.7.8 INSERT CHARACTER (I/C) (Figure 3-39)

The Insert Character (I/C) function allows a character or space to be inserted in any unprotected, or foreground, character position. This function is a keyboard function and is activated by depressing the I/C key and the desired character (or space bar) to be inserted. The character, or space, is entered at the cursor location and all other successive foreground characters are shifted one character position to the right until a background character is encountered or x=73 and y=26.

The initial enabling and entry sequence initiating the Internal Program Counter is the same as it is for a keyboard A/N Entry (paragraph 3.7.1).

P-CTR-1 is enabled by the SET SKIP REF FF signal and sets the Skip Refresh F/F which develops $\overline{\text{SKIP REF CYCLE}}$.

P-CTR-2 is enabled by the EXPANSION signal and develops $\overline{\text{LOAD Y COIN CTR}}$ and $\overline{\text{LOAD X COIN CTR}}$ signals which load the C.A.Y. Coincidence Counter and the C.A.X. Coincidence Counter with complementary data of the CA-Y and CA-X counters respectively, for storage of the cursor address until the routine is completed. P-CTR-2 in conjunction with $\overline{\text{XMIT + PRINT (COMMAND)}}$ develops $\overline{\text{R/R-1}}$ which enables the R/R Command F/F. When timing signal t0.6 clocks the R/R Command F/F a $\overline{\text{R/R COMMAND TO MEMORY}}$ is developed which reads the data out of memory corresponding to the cursor location.

P-CTR-3 combined with the EXPANSION and FG signals enables P-CTR-4. If the data read out of memory indicates a background character, $\overline{\text{FG}}$ will cause P-CTR-3 to inhibit P-CTR-4 and enable the transfer address F/F (XFR ADD FF). Timing signal t0.1 clocks the transfer address F/F generating TRANSF (COIN CTR) → CA and $\overline{\text{TRANSF (COIN CTR)} \rightarrow \text{CA}}$. TRANSF (COIN CTR) → CA enables timing signal t0.2 generating $\overline{\text{CLEAR CAX CTR}}$ and CLEAR CAY CTR which clear the CA counters. TRANSF (COIN CTR) → CA enables timing signal t0.4 which transfers the original cursor address, from the coincidence counters where it was stored, into the CA counters. For a background character the routine stops at this point and the Insert Character function has not been accomplished.

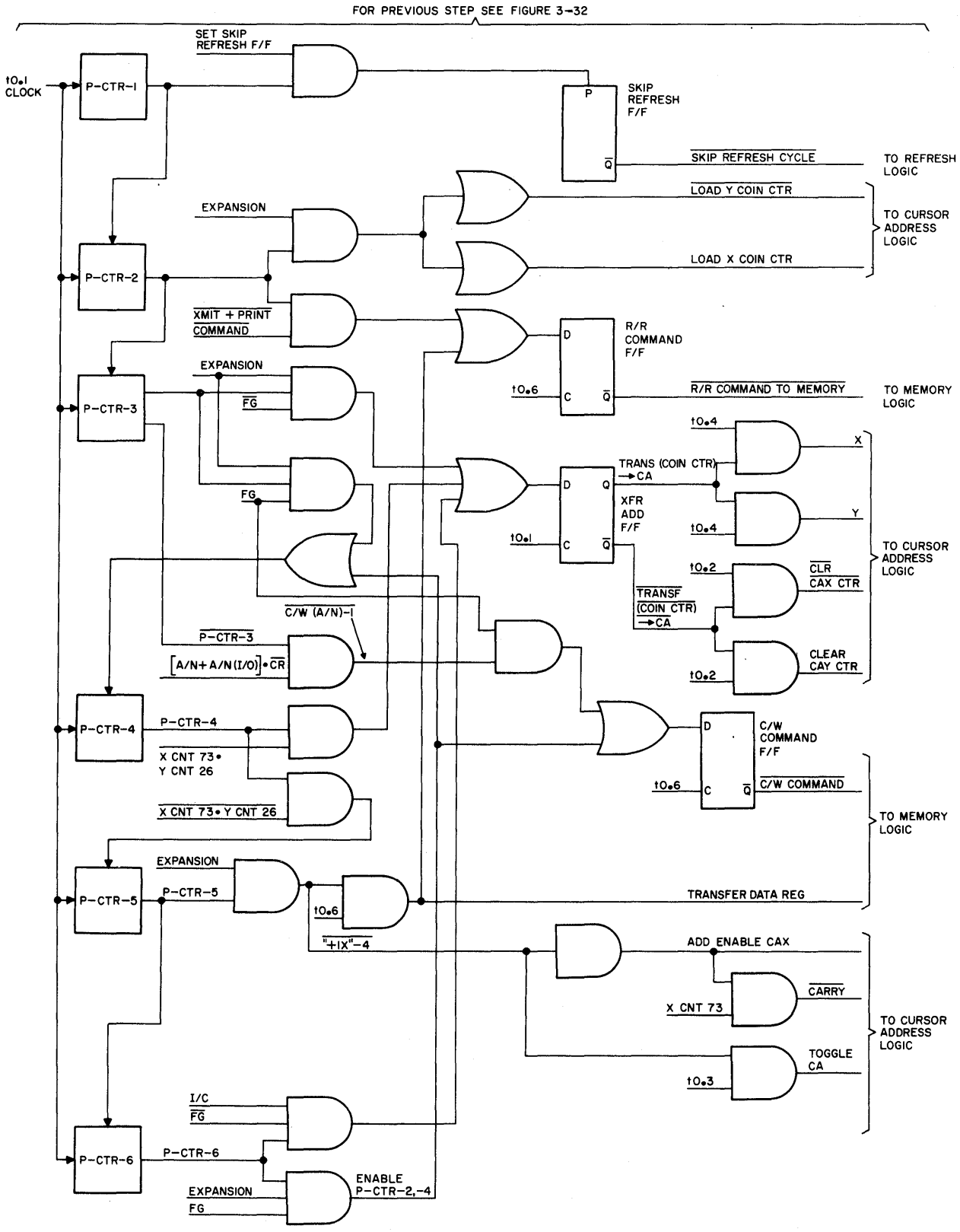


Figure 3-39. INSERT CHARACTER (I/C) Logic Flow Diagram

P-CTR-3' in coincidence with FG and (A/N + A/N (I/O)) • $\overline{(C/R)}$ generates $\overline{C/W(A/N)-1}$ which enables the C/W Command F/F. Timing signal t0.6 clocks the C/W Command F/F which generates a $\overline{C/W\ COMMAND}$ and enters the keyboard character into memory.

If the character was inserted at the x=73, y=26 position P-CTR-4 will inhibit P-CTR-5 and generate $\overline{XFR\ (COIN\ CTR) \rightarrow CA-1}$ which transfers the contents of the coincidence counters into the cursor address counters. If the insertion is made at any other than the last position the routine continues.

P-CTR-5 in conjunction with EXPANSION generates $\overline{"+1X"-4}$ and $\overline{R/R-8}$. "+1X"-4 creates $\overline{ADD\ ENABLE\ CAX}$, and in conjunction with timing signal t0.3 a TOGGLE CA. The CA-X counter is incremented one position to the right. R/R-8 in conjunction with timing signal t0.6 generates $\overline{TRANSFER\ DATA\ REG}$ which transfers the data in the "A" Data Reg into the "B" Data Reg. R/R-8 also enables the R/R Command F/F which is clocked by timing signal t0.6 and develops and $\overline{R/R\ COMMAND\ TO\ MEMORY}$. $\overline{R/R\ COMMAND\ TO\ MEMORY}$ reads the data in memory at that location and stores it in the "A" Data Reg.

P-CTR-5 enables P-CTR-6.

P-CTR-6 in conjunction with FG and EXPANSION generates $\overline{ENABLE\ P-CTR-4-2}$ which enables P-CTR-4 and allows P-CTR-4, P-CTR-5 and P-CTR-6 to recycle. P-CTR-6, EXPANSION, FG and I/C generate $\overline{CW-4}$ writing the contents of the "B" Register into memory. P-CTR-6 and $\overline{FG}$ will inhibit P-CTR-4, the routine stops, no C/W signal occurs and the $\overline{XFR\ (COIN\ CTR) \rightarrow CA-1}$ signal is developed.

3.7.9 DELETE CHARACTER (D/C) (See Figure 3-40).

The Delete Character (D/C) function allows any unprotected, or foreground, character to be deleted by locating the cursor under the character to be deleted and depressing the D/C key. If the cursor is located under a protected character, no deletion takes place. When a character is deleted all foreground data moves one location to the left and a blank character is inserted at the end. This blank character will be inserted just to the left of the first protected character encountered or will be in the last character position of the display, (X=73, Y=26).

The initial enabling and entry sequence initiating the Internal Program Counter is the same as it is for a keyboard A/N Entry (paragraphs 3.7.1.1 thru 3.7.1.3).

P-CTR-1 and SET SKIP REFRESH generate the $\overline{SKIP\ REFRESH\ CYCLE}$ signal.

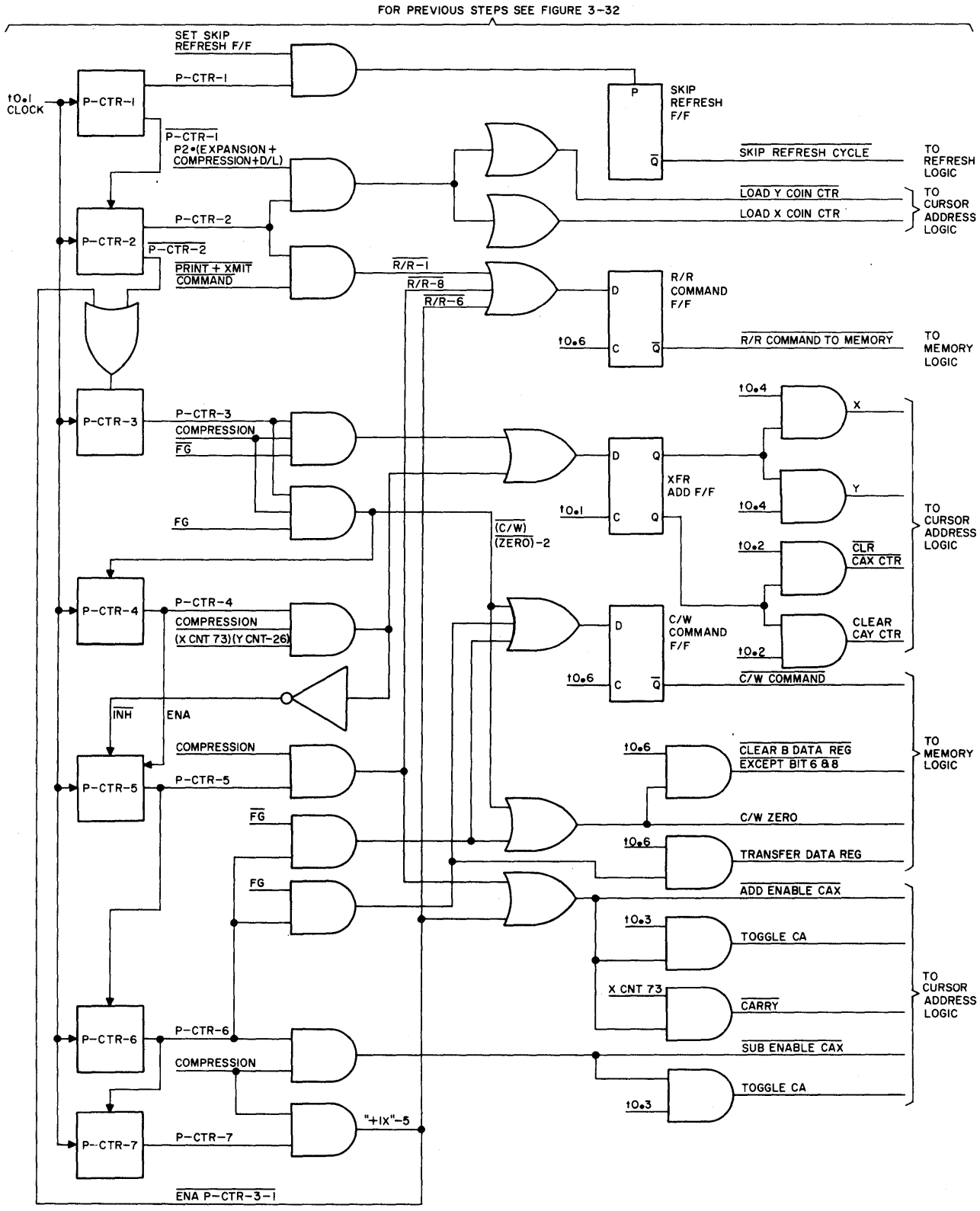


Figure 3-40. DELETE CHARACTER Logic Flow Diagram

P-CTR-2 in conjunction with the Expansion + Compression signal generates $\overline{\text{LOAD Y COINC CTR}}$ and $\overline{\text{LOAD X COINC CTR}}$. These signals load the C.A.Y. Coincidence Counter and the C.A.X. Coincidence Counter with complementary data from the CA-Y and CA-X counters respectively, for storage of the cursor address until the routine is completed. P-CTR-2 in conjunction with $\overline{\text{XMIT + PRINT (COMMAND)}}$ develops $\overline{\text{R/R-1}}$ which enables the R/R Command F/F. When timing signal to.6 clocks the R/R Command F/F, $\overline{\text{R/R COMMAND TO MEMORY}}$ is developed which reads the data out of memory corresponding to the cursor location.

$\overline{\text{P-CTR-2}}$ enables P-CTR-3.

If the character at the cursor address is $\overline{\text{FG}}$, P-CTR-3 will inhibit P-CTR-4 and develop $\overline{\text{TRANSFER (COIN CTR)} \rightarrow \text{CA-1}}$. $\overline{\text{TRANSFER (COIN CTR)} \rightarrow \text{CA-1}}$ enables the Xfr F/F which when clocked by timing signal t0.1 develops XFR (COIN CTR) → CA and $\overline{\text{XFR(COIN CTR)} \rightarrow \text{CA}}$. These signals transfer the cursor address out of the coincidence counters into the cursor address counters. The routine will stop at this point and the character is not deleted. If however the character read out was FG, P-CTR-4 is enabled and a $\overline{\text{(C/W) (ZERO)-2}}$ is developed. $\overline{\text{(CW)(ZERO)-2}}$ causes a blank character to be written into the cursor location.

If the cursor is at (X=73, y=26) P-CTR-5 is inhibited, $\overline{\text{TRANSFER (COIN CTR)} \rightarrow \text{CA-1}}$ is developed and the routine stops. If $\overline{\text{(XCNT 73)} \bullet \text{(y CNT 26)}}$ and (EXPANSION + COMPRESSION). P-CTR-5 is enabled and the routine continues.

P-CTR-5 with (EXPANSION + COMPRESSION) develops the following sequential signals. $\overline{\text{"+1X"-4}}$ is developed, which becomes $\overline{\text{ADD ENABLE CA X}}$, and enables timing signal t0.3 which generates TOGGLE CA. These signals increment the CA-X counter one count. If the X count is 73, a $\overline{\text{CARRY}}$ is generated which also increments the CA-Y counter. $\overline{\text{R/R-8}}$ is developed which enables the R/R Command F/F. When the R/R Command F/F is clocked by timing signal t0.6 a $\overline{\text{R/R COMMAND TO MEMORY}}$ is generated. This signal causes the memory at the present location to be read and stored. A $\overline{\text{XFR DATA REG}}$ signal is also developed but has no effect at this point in the routine. P-CTR-5 enables P-CTR-6.

P-CTR-6 in addition to enabling P-CTR-7 develops the following sequential signals. P-CTR-6 with COMPRESSION develops "-1X"-1 which generates $\overline{\text{SUB ENABLE CA-X}}$ and TOGGLE CA which decrement the CA-X counter one count. "-1X"-1 enables timing signal t0.6 which generates $\overline{\text{XFR DATA REG}}$ which transfers the data in the "A" Data Reg into the "B" Data Reg. If the character read out by P-CTR-5 is a FG character a $\overline{\text{C/W-4}}$ signal is generated which enables the C/W Command F/F. When the

C/W Command F/F is clocked by timing signal t0.6 a $\overline{\text{C/W COMMAND}}$ is generated and sent to the Memory Logic. The contents of the "B" Data Register are then written into memory. If however the character read out by P-CTR-5 is a $\overline{\text{FG}}$ character a $\overline{\text{C/W ZERO-2}}$ is generated and a blank character is written into memory.

P-CTR-7 in conjunction with the COMPRESSION signal generates $\overline{\text{ENA P-CTR-3-1}}$ which enables P-CTR-3. Also developed are $\overline{\text{"+1X"-5}}$ which increments the CA-X counter and a $\overline{\text{R/R-6}}$ which reads out the memory data at the new location.

3.7.10 TRANSMIT (XMIT) See Figure 3-41)

In the keyboard-initiated TRANSMIT (XMIT) mode of the terminal, the operator selects the end point of the message to be transmitted by positioning the cursor just to the right of the last character of the message. When the XMIT pushbutton is depressed or an external transmit command is initiated, an EOT symbol is generated at that display location and a search is initiated, under Internal Program Counter Control, for the start of the message. The start of the current message may be at X=0, Y-0 or at a previous EOT which terminated a previous transmission. When the start point has been determined, a $\overline{\text{START READOUT MEM FOR XMIT}}$ is generated which sets the I/O Logic into operation. The I/O Logic then operates at the selected output baud rate until the transmission has been completed. When the $\overline{\text{START READOUT MEM FOR XMIT}}$ signal has been generated, the Internal Program Counter ceases operation.

The initial enabling and entry sequence initiating the Internal Program Counter operation is the same as it is for A/N Entry (paragraph 3.7.1), except that when the "B" F/F generates the LOAD B DATA REG/ENTRY signal, data bits 6 and 7 are loaded into the "B" Data Register of the Memory Logic. Bits 6 and 7 are enabled by the (XMIT + PRINT) CMD signal.

The first count of the Internal Program Counter, P-CTR-1, enables the SET SKIP REFRESH F/F SIGNAL. This signal causes the $\overline{\text{SKIP REFRESH CYCLE}}$ signal to be developed and sent to the Refresh Logic.

P-CTR-2, enabled by the XMIT signal, generates a $\overline{\text{C/W-4}}$ command which writes bits 6 and 7 into memory. Bits 6 and 7 were loaded into the "B" Data Register during the Internal Program Counter entry and enabling routine. When a refresh read/restore operation is performed on the memory, bits 6 and 7 will be decoded and displayed as an EOT symbol.

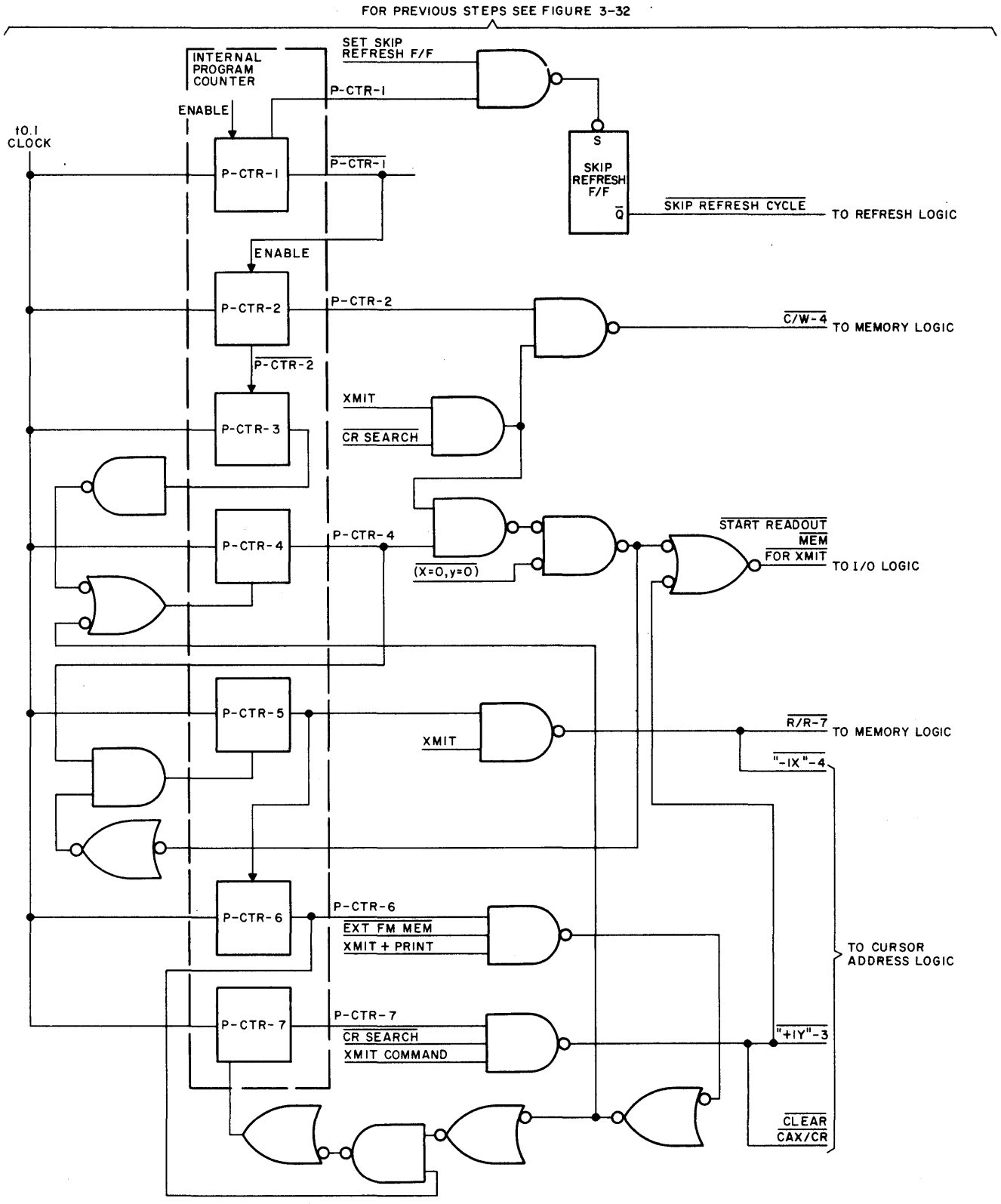


Figure 3-41. TRANSMIT Logic Flow Diagram

P-CTR-3 is not utilized in this routine except to enable P-CTR-4.

P-CTR-4 samples the condition of the ($\overline{X=0, Y=0}$) signal. If the cursor is at (X=0, Y=0), P-CTR-5 is inhibited and a $\overline{\text{START READOUT MEM FOR XMIT}}$ signal is generated. If however the cursor is not at (X=0, Y=0), P-CTR-5 is enabled and the search for (X=0, Y=0) or an EOT continues.

P-CTR-5, with the XMIT signal, causes the $\overline{\text{"=1X"-4}}$ and $\overline{\text{R/R-7}}$ signals to be generated. The $\overline{\text{"-1X"-4}}$ moves the cursor one increment in the direction toward (X=0, Y=0) position in search of the (X=0, Y=0) or EOT condition. $\overline{\text{R/R-7}}$ samples the memory location for an EOT in memory.

As P-CTR-5 moves the cursor and samples the memory, P-CTR-6 activates the enabling circuitry for P-CTR-4 and P-CTR-7. If an EOT is not at the respective cursor location, P-CTR-6 enables P-CTR-4 and inhibits P-CTR-7. The routine involving P-CTR-4, P-CTR-5 and P-CTR-6 continues until (X=0, Y=0) or an EOT is sensed. When an EOT is sensed, it is recognized as the end of a previous transmission. Therefore the start of the current transmission starts on the next line at the X=0 position. When the EOT is sensed, P-CTR-4 is inhibited and P-CTR-7 is enabled.

P-CTR-7 generates $\overline{\text{"+1Y"-3}}$, $\overline{\text{CLEAR CAX/CR}}$ and $\overline{\text{START READOUT MEM FOR XMIT}}$. The $\overline{\text{"+1Y"-3}}$ moves the cursor to the next Y line. $\overline{\text{CLEAR CAX/CR}}$ moves the cursor to the X=0 position so that the cursor is now at the first character position of the new message to be transmitted. $\overline{\text{START READOUT MEM FOR XMIT}}$ is then generated and the I/O Logic enabled. The Internal Program Counter has completed its task and the I/O Logic takes over to perform the actual data transmission.

3.7.11 TAB (See figure 3-42)

The TAB key automatically moves the cursor to the next tab stop, if any exist. Tab stops are set automatically when the display contains both background and foreground fields. The first foreground character following a background field is a tab stop. If there are no tab stops set, the TAB key will move the cursor to the lower right hand corner of the display.

The initial enabling and entry sequence initiating the Internal Program Counter operation is the same as it is for A/N Entry (paragraph 3.7.1).

P-CTR-1 enables the SET SKIP REFRESH F/F signal. This signal causes the $\overline{\text{SKIP REFRESH CYCLE}}$ signal to be developed and sent to the Refresh Logic. $\overline{\text{P-CTR-1}}$ enables P-CTR-2.

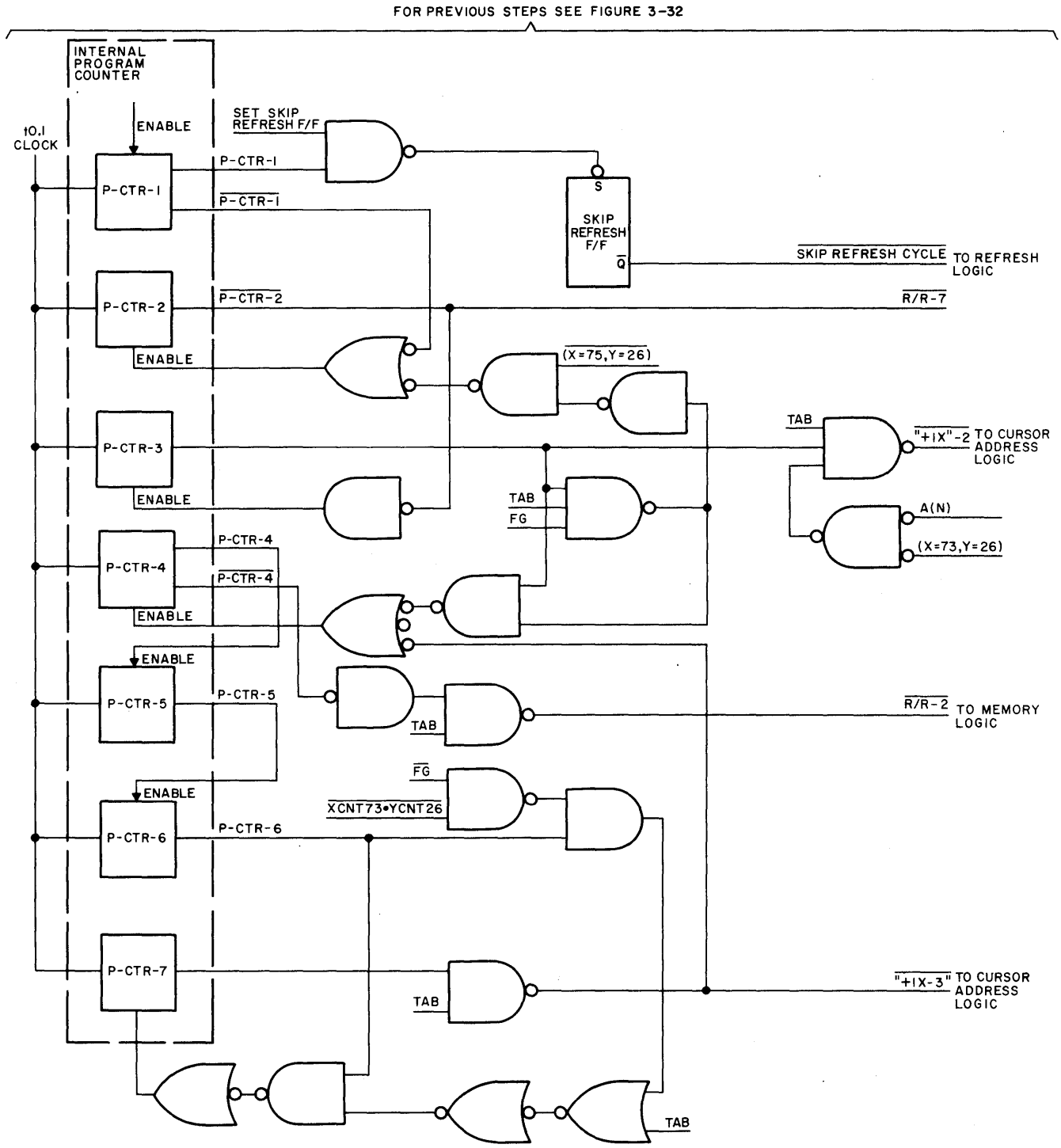


Figure 3-42. TAB Logic Flow Diagram

P-CTR-2, enabled by $\overline{\text{XMIT + PRINT (COMMAND)}}$, generates a R/R-1 signal to determine whether this cursor location contains a background or foreground character. P-CTR-2 also enables P-CTR-3.

P-CTR-3, in conjunction with $\overline{(x = 73, y = 26)}$, TAB, and (A/N), generates a $\overline{\text{"+1X"-2}}$ which moves the cursor to the next character location. P-CTR-3, in conjunction with the foreground (FG) signal, if the character readout with P-CTR-2 is a FG character, will inhibit P-CTR-4. If the character is a foreground character and $\overline{(x = 73, y = 26)}$, P-CTR-2 will be enabled, and the search for the end of the foreground field is continued. If (x = 73, y = 26), and an A/N signal is not present, P-CTR-2 will be inhibited and the cursor will remain in the lower right hand corner of the screen. If a $\overline{\text{FG}}$ character is read out and $\overline{(x = 73,}$ $\overline{y = 26)}$, then P-CTR-4 is enabled and the search for the end of the background field is initiated.

$\overline{\text{P-CTR-4}}$, in conjunction with the TAB signal, generates a $\overline{\text{R/R-2}}$ to sample the memory at the respective cursor location. P-CTR-4 enables P-CTR-5.

The only function performed by P-CTR-5 is to enable P-CTR-6.

P-CTR-6 must sample for a background character, foreground character of (x = 73, y = 26). If a background character ($\overline{\text{FG}}$) is sensed and $\overline{(x = 73, y = 26)}$, the search for the end of the background field must continue by causing P-CTR-7 to enable P-CTR-4. If a foreground character is sensed, P-CTR-7 is inhibited and the cursor remains at its present location. If (x = 73, y = 26) is sensed, P-CTR-7 will also be inhibited.

P-CTR-7, under the condition of $\overline{\text{FG}} \cdot \overline{(x = 73, y = 26)}$, will generate a $\overline{\text{"+ 1X" - 3}}$ which moves the cursor to the next location and enables P-CTR-4. When a FG + (x = 73) (y = 26) condition is sensed, P-CTR-7 is inhibited and the routine ceases; the cursor remains at its respective location.

3.7.12 PRINT (See figure 3-43)

Depressing the PRINT key (and also the SHIFT key) initiates a batch readout of the contents of the display to Printer accessory equipment by switching the Video Display Terminal from the Ready or Local modes to the Print mode. When the SHIFT-PRINT keys are depressed, a print symbol (■■) will appear on the screen at the cursor position. The print symbol indicates to the user and to the terminal that all information preceding the symbol has been printed. Consequently, on a

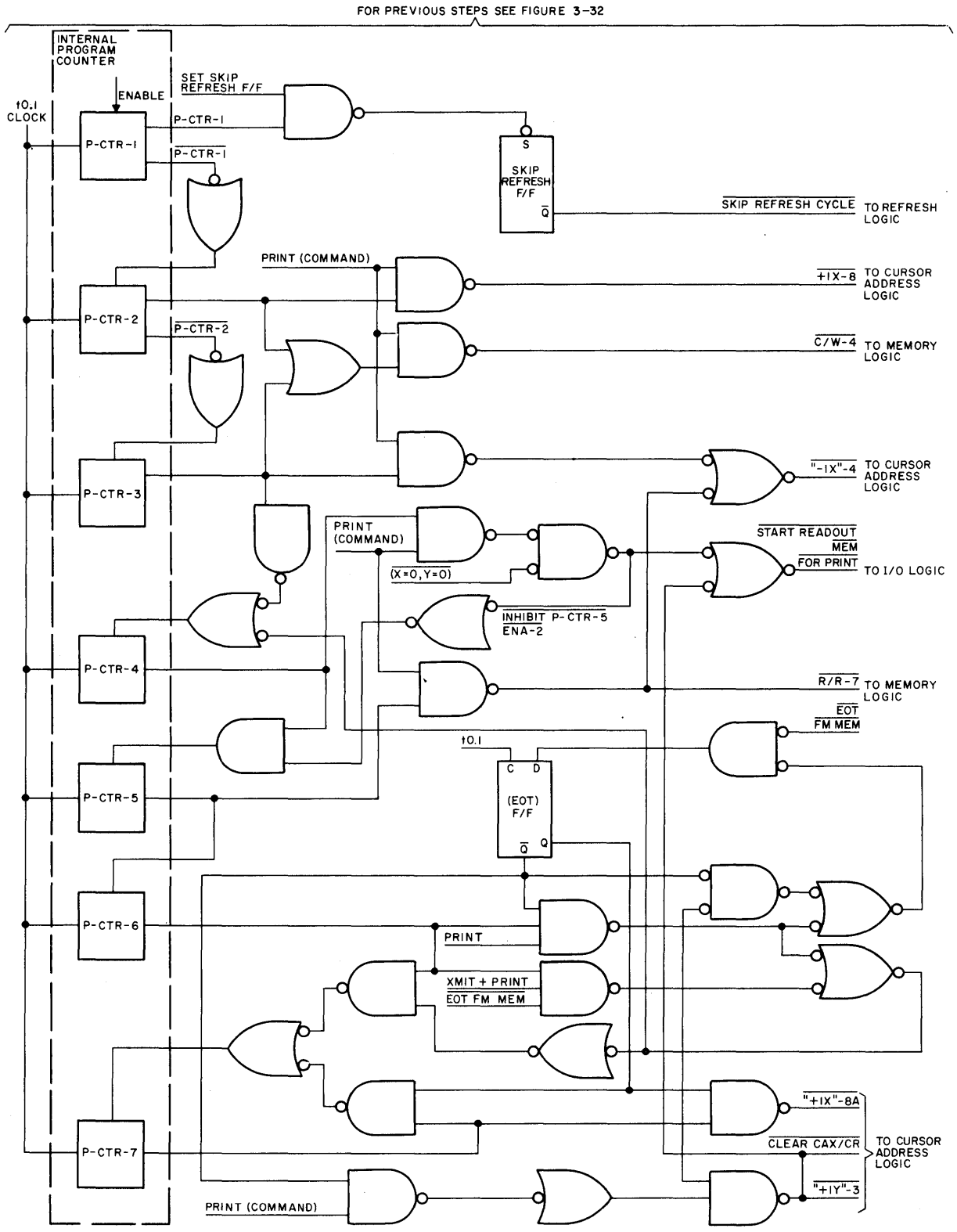


Figure 3-43. PRINT Logic Flow Diagram

subsequent print request, the cursor will return to either the home position or to the line following the previous print symbol to begin printing again. The Internal Program Counter causes the print symbol to be generated and searches for the start of the message which will be at the home position or following a previous print symbol. When the message has been printed, the cursor will be positioned on the next line at the x = 0 position.

The initial enabling and entry sequence initiating the Internal Program Counter operation is the same as it is for TRANSMIT (paragraph 3.7.10).

P-CTR-1 causes the $\overline{\text{SKIP REFRESH CYCLE}}$ signal to be developed and sent to the Refresh Logic. $\overline{\text{P-CTR-1}}$ enables P-CTR-2 and resets the $(\text{EOT})^2$ SEARCH F/F.

P-CTR-2, in conjunction with the PRINT (COMMAND) signal, generates a $\overline{\text{"+ 1x" -8}}$ which causes the cursor to move to the next x increment since two EOT symbols must be generated and displayed. A $\overline{\text{C/W-4}}$ signal is generated to write an EOT into memory at the new cursor location.

P-CTR-3, in conjunction with the PRINT (COMMAND) signal, generates a $\overline{\text{"-1x" - 4}}$ which moves the cursor to its original end of print position. A $\overline{\text{C/W-4}}$ signal is again generated to write an EOT into memory at the original cursor position. Two EOT symbols have now been generated at the end of print message.

P-CTR-4 samples for the (x = 0, y = 0) condition. If (x = 0, y = 0), P-CTR-5 is inhibited and a $\overline{\text{START READOUT MEM FOR PRINT}}$ signal is generated. If ($\overline{\text{x = 0}}$, $\overline{\text{y = 0}}$), then P-CTR-5 is enabled and the search for two EOT symbols or (x = 0, y = 0) continues.

P-CTR-5, in conjunction with the PRINT (COMMAND) signal, generates $\overline{\text{"- 1X" - 4}}$ to move the cursor one increment toward the home position, and $\overline{\text{R/R-7}}$ in search of EOT symbols. P-CTR-5 also enables P-CTR-6.

P-CTR-6 now controls the search for EOT symbols and the recycling of the routine back to P-CTR-4 through P-CTR-6. Several events may occur at the P-CTR-6 stage as follows. The character readout is $\overline{\text{EOT}}$ whereby P-CTR-4 is enabled, the $(\text{EOT})^2$ Search F/F is reset and P-CTR-7 is inhibited. The character readout is an EOT and the $(\text{EOT})^2$ Search F/F is not set. P-CTR-4 is enabled, the EOT Search F/F is set and P-CTR-7 is inhibited. The character readout is an EOT and the $(\text{EOT})^2$ Search F/F is set. This condition is a result of a second EOT in succession being read out of memory and signifies the end of a previous print cycle. As a result of this condition P-CTR-7 is enabled.

When P-CTR-7 occurs and the $(EOT)^2$ Search F/F is set, P-CTR-7 enables itself for another count, the $(EOT)^2$ Search F/F is reset and a "+ 1x" -8 signal is generated to move the cursor from its present position. If however the $(EOT)^2$ Search F/F is reset, then a "+ 1y" -3 is generated to move the cursor down one line. A $\overline{\text{CLEAR CAY/CR}}$ is also generated to move the cursor to the x = 0 position and a $\overline{\text{START READOUT MEM FOR PRINT}}$ is generated and routed to the I/O Logic. The Internal Program Counter has completed its operation and the I/O Logic will perform the transmission of data to the peripheral devices.

3.7.13 RECORD (OFF-LINE CONTINUOUS) (See figure 3-44)

When the Tape Cassette Accessory Unit is in the Off-Line Continuous Record mode, recording may be initiated by depressing the CR (Carriage Return) key on the Video Display Terminal keyboard. This permits the user to prepare small blocks of data such as program statements, either formatted or free form. When this mode of operation is initiated, the terminal searches for a previous CR character, or returns to the (x = 0, y = 0) position if no CR is found to start recording. The Internal Program Counter controls the CR search and develops a $\overline{\text{START READOUT MEM FOR PRINT}}$ signal which sets the I/O Logic into operation. The I/O Logic interfacing with the Tape Cassette Unit will control the recording. When the $\overline{\text{START READOUT MEM FOR PRINT}}$ signal has been generated, the Internal Program Counter ceases operation.

The initial enabling and entry sequence initiating the Internal Program Counter operation is the same as it is for A/N Entry (paragraphs 3.7.1).

The first count of the Internal Program Counter, P-CTR-1, sets the KB Rec Cont F/F to the state where the $\overline{Q}$ output is high. The KB Rec Cont F/F is enabled by RECORD MODE, $\overline{\text{CONTINUOUS MODE}}$ and CR/KB. When the KB Rec Cont F/F is set, the $\overline{Q}$ output enables P-CTR-1 to set the Skip Refresh F/F. The $\overline{\text{SKIP REFRESH CYCLE}}$ signal is developed and sent to the Refresh Logic. $\overline{\text{P-CTR-1}}$ enables P-CTR-2.

P-CTR-2, in conjunction with $\overline{\text{XMIT + PRINT (COMMAND)}}$, generates $\overline{\text{R/R-1}}$ which samples the memory with regard to whether the character at that location is FG or $\overline{\text{FG}}$. $\overline{\text{P-CTR-2}}$ enables P-CTR-3.

P-CTR-3 samples the result of the read/restore operation generated by P-CTR-2. If a $\overline{\text{FG}}$ character is sampled, the CR character cannot be entered at that location and the routine ceases. The operator must recognize that something invalid has

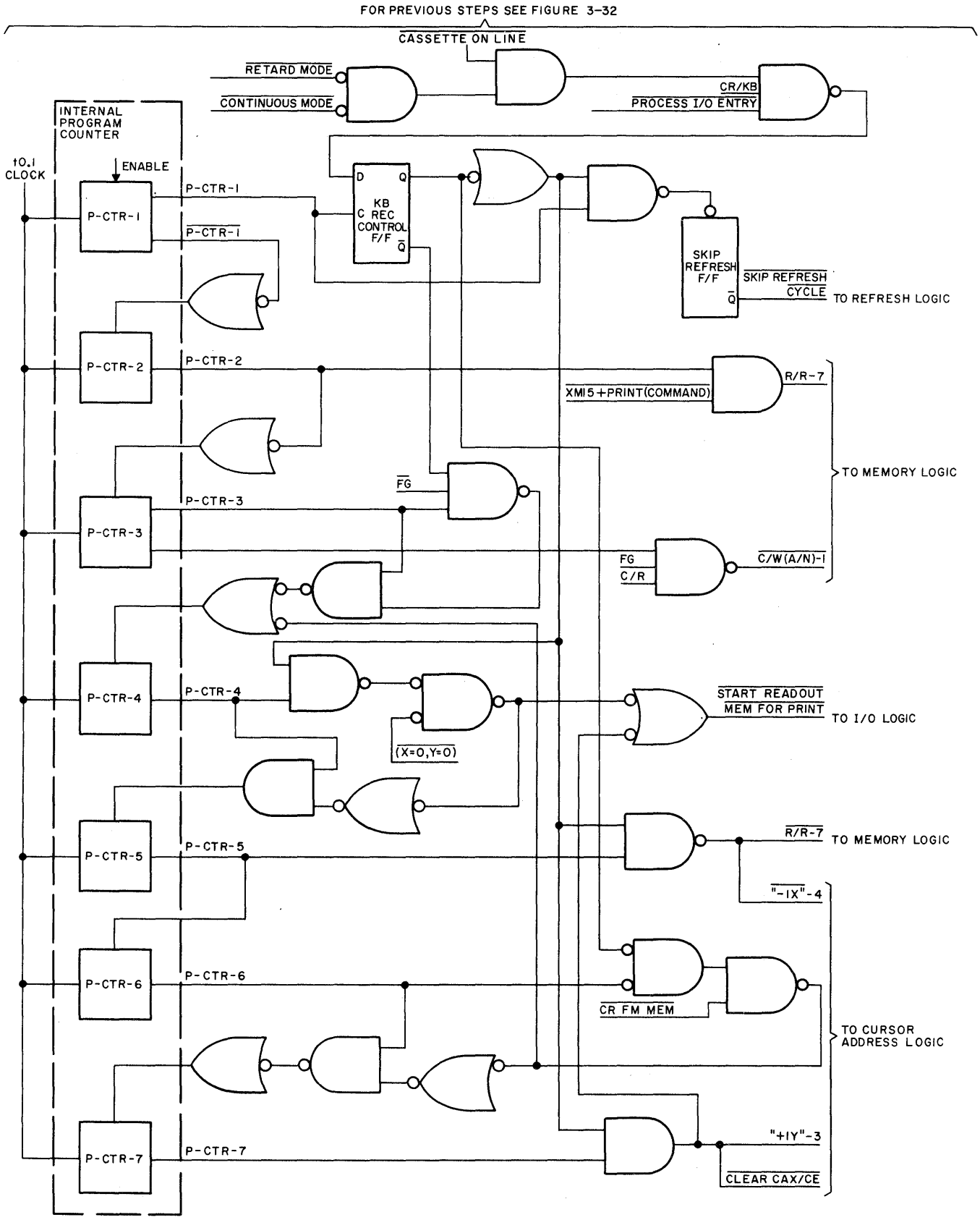


Figure 3-44. RECORD (OFF-LINE CONTINUOUS) Logic Flow Diagram

taken place. If the character sampled if FG, a $\overline{\text{C/W(A/N)-1}}$ signal is developed to enter a C/R character into memory at that location; P-CTR-4 is also enabled and the search for another CR or the (x = 0, y = 0) condition continues.

P-CTR-4 samples for the condition (x = 0, y = 0) and if it is true a $\overline{\text{START READOUT MEM FOR PRINT}}$ signal will be developed and sent to the I/O Logic. P-CTR-5 will also be inhibited. If the (x = 0, y =0) condition is not true, P-CTR-5 is enabled.

P-CTR-5 generates a $\overline{\text{"-1x"-4}}$ to move the cursor one increment toward the (x = 0, y = 0) position and a $\overline{\text{R/R-7}}$ to sample the character in memory at that location. P-CTR-5 also enables P-CTR-6.

P-CTR-6 samples the result of the read/restore operation generated by P-CTR-5 to determine whether a CR has been read out of memory. If a CR has not been read out P-CTR-7 is inhibited and P-CTR-4 is enabled. This continues the search for a CR or (x = 0, y = 0). If a CR has been read out, P-CTR-4 is inhibited and P-CTR-7 is enabled.

P-CTR-7 generates a $\overline{\text{START READOUT MEM FOR PRINT}}$ signal which is sent to the I/O Logic. P-CTR-7 also generates $\overline{\text{"+1Y"-3}}$ and a $\overline{\text{CLEAR CAX/CR}}$ signal to allow recording to begin at the x = 0 position of the next line following the CR character. Once the P-CTR-7 operation has taken place, the Internal Program Counter ceases its operation.

3.7.14 RECEIVE

When the Video Display Terminal is in the Receive mode and a command or a character has entered the I/O Logic, a $\overline{\text{PROCESS I/O ENTRY}}$ signal is developed which enables the Internal Program Counter. Once enabled, the Internal Program Counter functions the same as it does for keyboard-initiated characters or commands and implements them in the same fashion. An additional operation is performed by P-CTR-1 whereby it resets the Process I/O Entry F/F. The following operations may be initiated by the remote CPU.

A/N ENTRY
TRANSMIT
ADDRESS CURSOR (does not use P-CTR)
HOME CURSOR (does not use P-CTR)
DELETE LINE
SET BACKGROUND INTENSITY (does not use P-CTR)

INSERT LINE

CLEAR SCREEN

CLEAR FOREGROUND DATA

PRINT

SET FOREGROUND INTENSITY (does not use P-CTR)

CARRIAGE RETURN

BACKSPACE CURSOR

Section 4

OPERATION

4.1 GENERAL INFORMATION

The VDT can be operated in any one of three modes of operation, as selected by the FULL/HALF/BATCH switch. Batch mode permits the operator to enter up to 1,998 characters into the display memory, and perform editing functions upon the data prior to transmitting the data to the CPU. Half-duplex mode transmits each character to the CPU as the operator presses the key, and simultaneously displays the character on the monitor screen. Full-duplex mode transmits each character directly to the CPU as the operator presses the key without affecting cursor position, and thus without displaying the typed character. Since the VDT has separate input and output registers in full duplex, transmission of a typed character can take place simultaneously with the receipt of a character from the CPU. Typically, characters typed in full-duplex mode are echoed from the CPU to permit validation of operator entries, and thus appear to be displayed as a direct result of pressing the appropriate keys.

The full-duplex mode of communication can be used only when the communication system is capable of simultaneous two-way transmission. The half-duplex and batch modes are used when the communication system is not capable of two-way communication.

The VDT has the capability of displaying characters at two levels of intensity, entitled background and foreground, and signify protected and unprotected data. Background characters are displayed at a lighter intensity than foreground characters. Protected data displayed in the background intensity is data that has been received from the CPU connected to the VDT, and cannot be modified by the operator in batch mode. Unprotected data is displayed in the foreground intensity and can be modified by the operator. The background/foreground feature is used to establish tabular fields, transmission fields, and editorial fields within the display. With both background and foreground characters displayed on the VDT, the operator can make use of the TAB key to move the cursor to the first character position following the next background field. Only foreground data is transmitted from the VDT to the

CPU. The I/C (insert character) and D/C (delete character) keys are used to edit the characters displayed, and expand or contract the foreground characters. The expansion/contraction of foreground characters is delineated by the point of insertion/deletion (the cursor) and the following background character or the end of the display.

4.2 OPERATOR CONTROLS AND KEYBOARD

The operator controls are located on the monitor display panel (behind a spring-loaded panel directly below the display screen) and on the keyboard. The monitor display control panel (figure 4-1) contains the controls that select the operational mode, the transmission controls, and the display contrast control. The keyboard (figure 4-2) contains a typewriter keyboard, a numeric key cluster, cursor and editing keys, and switch indicators. The function of the operator controls is given in tables 4-1 and 4-2.

Figure 4-1. Typical Control Panel Controls

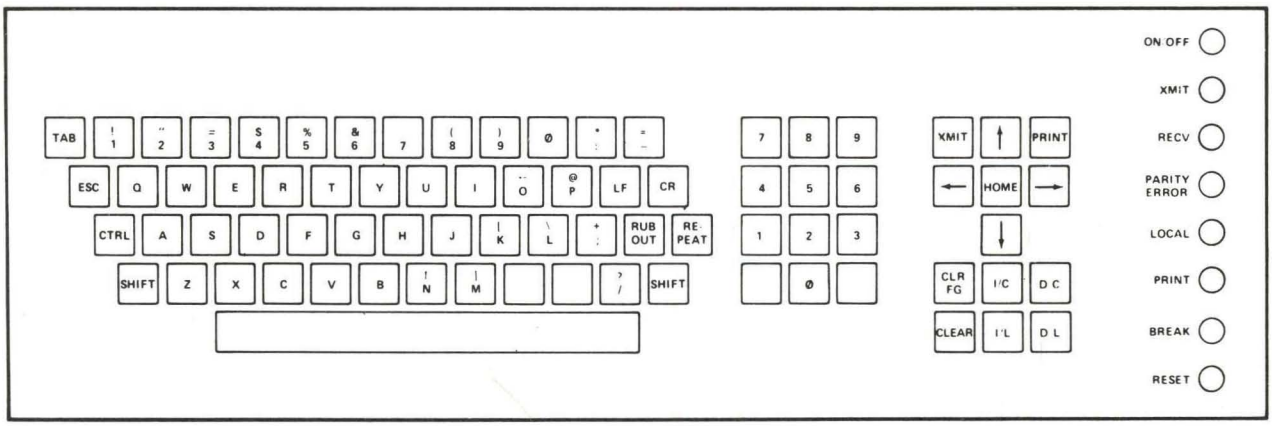


Figure 4-2. Keyboard Keys, Control Switches, and Indicators

Table 4-1. Monitor Display Panel Controls

Control	Function
POWER ON/OFF	Primary control of applied power. When set to ON, the keyboard power ON/OFF switch has no effect. When set to OFF, the keyboard power ON/OFF switch controls the application of power to the VDT.
EOT/CR	Selects end-of-transmission (EOT) code or carriage return (CR) code to indicate the end of message to the CPU. Also controls line turnaround discipline.
PARITY 1, 0, ODD, EVEN	Selectable parity generation and check control to assure compatibility with those systems that require parity. If the system does not use parity checking, the control should be set to 0 or 1.
CONTRAST	Adjusts the intensity of the display.
FULL/HALF/BATCH	Mode control switch. Set to FULL for full-duplex operation; set to HALF for half-duplex operation; and set to BATCH for batch operation.
BAUD	Selects the rate of data transfer between the VDT and the CPU.
CA ON/CA AUTO	Provides selection of CA control for use with data sets that require transmit/receive conditioning. Normally set to CA ON for baud rates under 600, and to CA AUTO for baud rates of 600 or above.

Table 4-2. Keyboard Controls and Indicators

Control/Indicator/Key	Function
TAB key	The TAB key moves the cursor to the next tab stop. Tab stops are set automatically when the display contains both foreground and background fields. The first foreground character following a background field is a tab stop. If there are no tab stops set, the TAB key causes the cursor to move to the lower right-hand corner of the display.
ESC key	The ESC (escape) key is generally used to generate a program interrupt. The use of this key is entirely dependent upon the communications software.

Table 4-2. Keyboard Controls and Indicators (continued)

Control/Indicator/Key	Function
CTRL key	The CTRL (control) key is used in conjunction with the character or symbol keys to generate non-displayed but transmittable character codes. The transmitted character code thus generated is used for function codes, security, etc.
LF key	The LF (line feed) key causes the line feed character to be transmitted when operating in the full- or half-duplex modes, but does not cause the cursor to move and does not cause the line feed character to be stored. When operating in the batch mode, the LF key moves the cursor horizontally and over-writes the foreground character over the cursor with a line feed (background characters are not affected).
CR key	The CR (carriage return) key moves the cursor down and back to the beginning of the next line, thus accomplishing both a carriage return and a line feed. In batch mode of operation, any characters entered on the same line but after the carriage return, cannot be transmitted.
RUBOUT	The RUBOUT key is used only in batch mode, and only for foreground characters. The RUBOUT key deletes the character over the cursor, and enters a character of all 1's into the display memory. The RUBOUT key is generally used to create filler characters when program execution delays are required.
REPEAT key	The REPEAT key can be used in batch mode to repeat any character on the typewriter keyboard. The most useful function of the REPEAT key is for rapid movement of the cursor, which is active in any mode of operation.
SHIFT keys	The SHIFT keys (left or right) permit the operator to type the characters marked in the upper portion of the typing keys. In addition, the SHIFT key is used as an interlock to prevent accidental use of the XMIT, PRINT, CLR/FG, and CLEAR keys. The SHIFT key can be used with any typing key; however, the bit pattern of the typed character is altered to the extent that the letters A through J typed with the SHIFT key pressed result in the letters Q through Z being displayed; and the typed letters Q through Z result in the letters A through J being displayed.
Numeric Key Cluster	The numeric key cluster contains the numeric keys 0 through 9 arranged in the same manner as a standard ten-key keyboard, a comma key (,), and a period key (.). The numeric key cluster displays and transmits the same characters and codes as the identical keys on the typewriter keyboard.

Table 4-2. Keyboard Controls and Indicators (continued)

Control/Indicator/Key	Function
Cursor Control Keys	The five cursor control keys consist of the HOME key, which returns the cursor to the upper left-hand corner of the display, the up-arrow (↑), the down-arrow (↓), the left-arrow (←), and the right-arrow (→), all of which move the cursor one character position or one line position in the indicated direction. With the exception of the HOME key, the cursor stepping keys may be used in conjunction with the REPEAT key for rapid cursor positioning. The cursor can be moved at a rate of 15 character positions or 15 lines per second by using the cursor stepping key and the REPEAT key. With the REPEAT key pressed, the HOME key is deactivated. Positioning of the cursor does not alter the display.
XMIT key	The XMIT key is used only in batch mode of operation, and is interlocked with the SHIFT key. When both the XMIT and SHIFT keys are pressed, the data stored in the VDT memory is transmitted one character at a time at the rate set by the BAUD rate switch. The position of the cursor and any previous transmit symbol determine the data to be transmitted. With the cursor positioned one or more lines down the display from the last transmit symbol, the first foreground character on the line following the last transmit symbol, or the first foreground character in the display if there were no previous transmit symbols, will be transmitted. The cursor will be positioned at the first character position of the line following the transmit symbol when the transmit function is completed. If the cursor was positioned on the last line when the XMIT key is pressed, the entire display is shifted up one line position, and the cursor is positioned at the first character position of the last line, which is now blank.
PRINT key	The PRINT key is used only in batch mode of operation, and is interlocked with the SHIFT key. When both the PRINT key and the SHIFT key are pressed, a batch readout of the contents of the display memory to optional equipment (serial-character hard copy printer or tape cassette unit) is initiated by switching the VDT from the ready or local mode to the print mode. Both foreground and background characters are transmitted.
CLR/FG key	The CLR/FG key is interlocked with the SHIFT key. When both the CLR/FG key and the SHIFT key are pressed, all foreground characters on the display are cleared and the cursor returns to the home position.

Table 4-2. Keyboard Controls and Indicators (continued)

Control/Indicator/Key	Function
CLEAR key	The CLEAR key is interlocked with the SHIFT key. When both the CLEAR key and the SHIFT key are pressed, all characters on the display are cleared and the cursor returns to the home position.
I/C key	The I/C (insert character) key is used only in batch mode of operation, and is used in conjunction with any character or space key on the typewriter keyboard to allow additional characters to be added at any foreground character position on the display. With the I/C key pressed, any character typed will be inserted between the foreground character over the cursor and the character to the left of the cursor. The foreground character over the cursor and all characters to its right will shift right one character position for each key typed. The shift right will continue until a background character or the end of the display is encountered.
D/C key	The D/C (delete character) key is used to delete the foreground character positioned over the cursor. All foreground characters to the right of the deleted character move left one position, and a blank character appears at the farthest right position of the foreground field.
I/L key	The I/L (insert line) key is used to provide a line of blanks between two existing lines for additional entry or formatting. The line directly over the cursor, and all lines below the cursor move down one line position, and the last line is lost. The cursor is relocated to the first character position of the new blank line.
D/L key	The D/L (delete line) key is used to remove an entire line of characters from the display. The line directly above the cursor is deleted and the lines below move up one line. A blank line appears at the bottom of the display, and the cursor is repositioned at the first character position of the line that has moved up to replace the deleted line.
ON/OFF indicator/switch	The ON/OFF indicator/switch is the secondary power switch for the VDT, and is interlocked with the POWER ON/OFF switch located on the monitor display control panel. With the POWER ON/OFF switch set to OFF, the keyboard power ON/OFF indicator/switch controls the application of power to the VDT. With the POWER ON/OFF switch set to ON, the keyboard power ON/OFF indicator/switch has no effect. Regardless of which switch controls the application of power, the keyboard ON/OFF indicator is lit when power is applied.

Table 4-2. Keyboard Controls and Indicators (continued)

Control/Indicator/Key	Function
TRANSMIT indicator	The TRANSMIT indicator is lit when the contents of the VDT memory are being transmitted to the receiving device.
RECV indicator/switch	The RECV indicator/switch is lit when the VDT is in the receive mode, which occurs when the switch is pressed or automatically after a transmission is completed.
PARITY ERROR indicator/switch	The PARITY ERROR indicator/switch lights when a parity error occurs during an exchange of data between the VDT and the CPU. The occurrence of an error is stored until the PARITY ERROR switch is pressed to reset the parity error.
LOCAL indicator/switch	The LOCAL indicator/switch is lit when the keyboard is enabled. The keyboard is not enabled, and the indicator is not lit, when the VDT is in the print, transmit, or receive modes to prevent inadvertent typing. When the LOCAL switch is pressed, the mode is switched to local and interrupts the print, transmit, or receive modes of operation.
PRINT indicator	The PRINT indicator is lit when the contents of display memory are being transferred to the optional serial-character hard copy printer or tape cassette unit.
BREAK indicator/switch	The BREAK indicator/switch is used to request a break in the transmission of data from a CPU to the VDT.
RESET indicator/switch	The RESET indicator/switch interrupts and resets all internal functions of the VDT as long as the switch is pressed. The interrupted functions include synchronization and display refresh, and as a result, the display goes completely blank when the RESET switch is pressed, and reappears when the switch is released.

4.3 POWER ON/OFF

The application of power to the VDT is controlled by the POWER ON/OFF switch located on the display control panel and the power ON/OFF indicator/switch located on the keyboard. With the POWER ON/OFF switch on the display panel set to ON, power is applied to the VDT and the keyboard power indicator lights but the ON/OFF switch has no effect. With the POWER ON/OFF switch on the display panel set to OFF, the application of power is controlled by the alternate action keyboard power

ON/OFF switch. After power is applied to the VDT, clear the display and display memory by pressing and holding the SHIFT key and pressing the CLEAR key. If intermittent operations are being performed at the VDT, it is recommended that power be left on until all VDT operations have been completed.

4.4 BATCH MODE

The batch mode of operation may be used with either half- or full-duplex communication systems. Batch mode enables the operator to take advantage of the editing functions inherent in the VDT.

4.4.1 ENTERING DATA

Data is entered by pressing the keys of the keyboard, and the characters are simultaneously stored in display memory and displayed on the screen. Any number of characters from one to 1,998 may be entered.

4.4.2 EDITING DATA

Five types of editing functions can be performed on the keyboard: retyping, character and line insertions, and character and line deletions. All editing functions are controlled by the position of the cursor, and the cursor position is controlled by the cursor positioning keys. Rapid cursor positioning is accomplished by holding the REPEAT key down while pressing the appropriate cursor positioning keys.

a. Retyping. Any foreground character displayed on the screen can be changed by positioning the cursor under the character to be changed and striking the appropriate key.

b. Character Insertion. The character insertion feature precludes the necessity of retyping an entire line should a character or space be missing. Position the cursor under the character of the desired insertion, press and hold the I/C key, and press the appropriate key to insert a character or space.

c. Character Deletion. The character deletion feature permits one or more characters or spaces to be deleted, and repositions the foreground displayed data. Position the cursor under the character to be deleted and press the D/C key. The character directly over the cursor will be deleted and replaced with the next right

character, with all foreground characters between the cursor position and the next background character left-shifted once each time the D/C key is pressed.

d. Line Insertion. The line insertion feature permits one or more lines to be inserted between existing lines on the display. Position the cursor directly below the line at which the new line of data is to be inserted, and press the I/L key. The line over the cursor and all lines below the cursor line will roll down one line position each time the I/L key is pressed. The last line on the display is lost.

e. Line Deletion. The line deletion feature permits an entire line of data to be deleted. Position the cursor directly below the line of data to be deleted and press the D/L key. The data in the line over the cursor will be deleted and all data lines below the cursor will roll up one line position.

4.4.3 TRANSMITTING DATA

After the data has been entered and edited, it is ready for transmission to the CPU. To transmit the foreground data (background data cannot be transmitted), position the cursor to the right of the last character to be transmitted, press and hold the SHIFT key and press the XMIT key. This will cause a transmit symbol to appear at the cursor position. The cursor will return to the home position or to the beginning of the line below the last prior transmission symbol, and the data will be transmitted at the selected baud rate. The cursor scans each line of data (assuming a multiple line transmission) until it senses a carriage return or end-of-line and will go on to the next line until it returns to the transmission symbol position. A carriage return code, or EOT code as determined by the EOT/CR switch, is then transmitted to the CPU and the cursor is repositioned at the start of the next line, awaiting either a response from the CPU or the next operator entry.

4.5 HALF-DUPLEX AND FULL-DUPLEX MODES

Both the half-duplex mode and the full-duplex mode of operation permit direct communication with the CPU. Each time a key is pressed, the corresponding code is transmitted to the CPU. The difference between the two modes is that in half-duplex the display is activated directly by the keyboard and the transmitted character is displayed on the screen. In full-duplex, the display is not activated by the

keyboard, and the transmitted characters are not displayed unless the CPU is programmed to echo the transmitted character back to the VDT.

4.6 STATUS MODES

The VDT operates in one of five status modes - ready, transmit, receive, print, or local. The selected status mode is indicated by four of the amber indicator/switches on the keyboard.

4.6.1 READY STATUS MODE

The ready status mode is the normal state of the VDT when awaiting either keyboard inputs or incoming data from the CPU. The VDT starts in the ready status mode when power is applied, and returns to the ready status upon completion of transmit, receive, or print functions. Any function can be interrupted and the VDT set to the ready status mode by pressing the RESET indicator/switch. Ready status is indicated when both the RECV and LOCAL indicators are lighted.

4.6.2 TRANSMIT STATUS MODE

The transmit status mode is selected whenever the contents of display memory are being transmitted to the CPU. During the transmit status mode, the typing keys on the keyboard are disabled to prevent garbling by inadvertent typing, and incoming characters from the CPU cannot be received. The TRANSMIT indicator is lighted during transmit status mode.

4.6.3 RECEIVE STATUS MODE

The receive status mode is initiated from the ready status mode when an incoming character is received from the CPU. The RECV indicator/switch is lighted and the keyboard is disabled (except in full-duplex mode of operation) during receive status mode.

4.6.4 PRINT STATUS MODE

The print status mode is initiated from the ready status mode for the purpose of printing the contents of display memory on the optional hard copy printer or tape cassette unit. When the VDT is in the print status mode, the PRINT indicator is lighted and the keyboard is disabled.

4.6.5 LOCAL STATUS MODE

The local status mode is selected by pressing the LOCAL indicator/switch. During local status mode, all communication with the CPU that is not under control of the VDT operator is inhibited, and the LOCAL indicator is lighted.

4.7 PROGRAMMING CONSIDERATIONS

The CPU associated with the VDT can be programmed to cause the VDT to perform a number of distinct functions, listed in table 2-2. Through the use of a combination of ASCII characters in the program, the Class I and Class II remote commands listed in table 2-2 can be issued to the VDT (see paragraph 2.3.8).

Table 4-3 is a listing of the cursor addresses that can be used in conjunction with the address cursor function (command 21 in table 2-2). The display is divided into 74 X-coordinates, numbered 0 through 73, and 27 Y-coordinates, numbered 0 through 26.

Table 4-3. Cursor Address Codes

Bit Pattern $b_7 \ldots\ldots b_1$	Decimal Value	ASCII Character	Key Stroke	Coordinates	
				Col. No. (X)	Line No. (Y)
0000000	0	NUL	$^{cs}@$	0	0
0000001	1	SOH	^{c}A	1	1
0000010	2	STX	^{c}B	2	2
0000011	3	ETX	^{c}C	3	3
0000100	4	EOT	^{c}D	4	4
0000101	5	ENO	^{c}E	5	5
0000110	6	ACK	^{c}F	6	6
0000111	7	BEL	^{c}G	7	7
0001000	8	BS	^{c}H	8	8
0001001	9	HT	^{c}I	9	9
0001010	10	LF	^{c}J	10	10
0001011	11	VT	^{c}K	11	11
0001100	12	FF	^{c}L	12	12
0001101	13	CR	^{c}M	13	13
0001110	14	SO	^{c}N	14	14

Table 4-3. Cursor Address Codes (continued)

Bit Pattern $b_7.....b_1$	Decimal Value	ASCII Character	Key Stroke	Coordinates	
				Col. No. (X)	Line No. (Y)
0001111	15	SI	^{c}O	15	15
0010000	16	DLE	^{c}P	16	16
0010001	17	DC1	^{c}Q	17	17
0010010	18	DC2	^{c}R	18	18
0010011	19	DC3	^{c}S	19	19
0010100	20	DC4	^{c}T	20	20
0010101	21	NAK	^{c}U	21	21
0010110	22	SYN	^{c}V	22	22
0010111	23	ETB	^{c}W	23	23
0011000	24	CAN	^{c}X	24	24
0011001	25	EM	^{c}Y	25	25
0011010	26	SUB	^{c}Z	26	26
0011011	27	ESC	^{cs}K	27	
0011100	28	FS	^{cs}L	28	
0011101	29	GS	^{cs}M	29	
0011110	30	RS	^{cs}N	30	
0011111	31	US	^{cs}O	31	
0100000	32	SP	SP	32	0
0100001	33	!	!	33	1
0100010	34	"	"	34	2
0100011	35	#	#	35	3
0100100	36	$	$	36	4
0100101	37	%	%	37	5
0100110	38	&	&	38	6
0100111	39	'	'	39	7
0101000	40	(	(	40	8
0101001	41	)	)	41	9
0101010	42	*	*	42	10
0101011	43	+	+	43	11
0101100	44	,	,	44	12
0101101	45	-	-	45	13
0101110	46	.	.	46	14

Table 4-3. Cursor Address Codes (continued)

Bit Pattern $b_7 \ldots\ldots b_1$	Decimal Value	ASCII Character	Key Stroke	Coordinates	
				Col. No. (X)	Line No. (Y)
0101111	47	/	/	47	15
0110000	48	0	0	48	16
0110001	49	1	1	49	17
0110010	50	2	2	50	18
0110011	51	3	3	51	19
0110100	52	4	4	52	20
0110101	53	5	5	53	21
0110110	54	6	6	54	22
0110111	55	7	7	55	23
0111000	56	8	8	56	24
0111001	57	9	9	57	25
0111010	58	:	:	58	26
0111011	59	;	;	59	
0111100	60	<	<	60	
0111101	61	=	=	61	
0111110	62	>	>	62	
0111111	63	?	?	63	
1000000	64	@	@	64	0
1000001	65	A	A	65	1
1000010	66	B	B	66	2
1000011	67	C	C	67	3
1000100	68	D	D	68	4
1000101	69	E	E	69	5
1000110	70	F	F	70	6
1000111	71	G	G	71	7
1001000	72	H	H	72	8
1001001	73	I	I	73	9
1001010	74	J	J		10
1001011	75	K	K		11
1001100	76	L	L		12
1001101	77	M	M		13
1001110	78	N	N		14

Table 4-3. Cursor Address Codes (continued)

Bit Pattern $b_7.....b_1$	Decimal Value	ASCII Character	Key Stroke	Coordinates	
				Col. No. (X)	Line No. (Y)
1001111	79	O	O		15
1010000	80	P	P		16
1010001	81	Q	Q		17
1010010	82	R	R		18
1010011	83	S	S		19
1010100	84	T	T		20
1010101	85	U	U		21
1010110	86	V	V		22
1010111	87	W	W		23
1011000	88	X	X		24
1011001	89	Y	Y		25
1011010	90	Z	Z		26
1011011	91	[	[		
1011100	92	\	\		
1011101	93	]	]		
1011110	94	↑	↑		
1011111	95	←	←		
1100000	96	`	c_{SP}	0	0
1100001	97	a	cs_1	1	1
1100010	98	b	cs_2	2	2
1100011	99	c	cs_3	3	3
1100100	100	d	cs_4	4	4
1100101	101	e	cs_5	5	5
1100110	102	f	cs_6	6	6
1100111	103	g	cs_7	7	7
1101000	104	h	cs_8	8	8
1101001	105	i	cs_9	9	9
1101010	106	j	cs:	10	10
1101011	107	k	cs;	11	11
1101100	108	l	c,	12	12
1101101	109	m	c-	13	13
1101110	110	n	c.	14	14

Table 4-3. Cursor Address Codes (continued)

Bit Pattern $b_7.....b_1$	Decimal Value	ASCII Character	Key Stroke	Coordinates	
				Col. No. (X)	Line No. (Y)
1101111	111	o	c/	15	15
1110000	112	p	c0	16	16
1110001	113	q	c1	17	17
1110010	114	r	c2	18	18
1110011	115	s	c3	19	19
1110100	116	t	c4	20	20
1110101	117	u	c5	21	21
1110110	118	v	c6	22	22
1110111	119	w	c7	23	23
1111000	120	x	c8	24	24
1111001	121	y	c9	25	25
1111010	122	z	c:	26	
1111011	123	{	c;	27	
1111100	124	¦	cs,	28	
1111101	125	}	cs-	29	
1111110	126	(Lead in)	cs.	30	
1111111	127	DEL	cs/	31	

As indicated by the above table, cursor addresses can be selected from the lower case ASCII set to avoid conflicts that may be caused by addresses selected from the control set.

It is recommended that X coordinates start with decimal value 96 and proceed through 127 to yield screen addresses of 0 through 31. Addresses 32 through 73 may be taken directly from their decimal equivalents. Similarly, Y coordinates should be selected from values 96 through 122 to yield screen addresses of 0 through 26.

Section 5

MAINTENANCE

5.1 GENERAL

This section provides troubleshooting and checkout procedures, adjustment procedures, and removal and replacement instructions. The unit requires no routine service or adjustment other than external cleaning and cable connection checks (see paragraph 5.6).

The procedures in this section assume that the user is familiar with the operator procedures. Special operating techniques that are useful or required during servicing are described at the appropriate points in this section.

CAUTION

Follow proper turn-on and turn-off procedures to avoid damage to the CRT.

On-premise corrective action is normally limited to replacement of plug-in circuits, and to any adjustments that may be required. When appropriate, visual inspections of the unit should be made to verify that correct internal interconnections are made, that there are no broken push-on pins in the back of the logic rack, and that there is no apparent mechanical damage to the unit.

5.2 TEST EQUIPMENT REQUIRED FOR MAINTENANCE

5.2.1 MULTIMETER

The electrical tests specified in this section are logic level checks, voltage measurements, and continuity checks which can be made with a digital multimeter. The multimeter should be an Eldorado Model 1820A, or equivalent. The primary requirement is that, when used as a dc voltmeter, its sensitivity should be at least 10,000 ohms per volt, and its readings should be accurate to within ±3% of indicated voltage (not of full-scale) for voltages of 1.4 to 2.2 volts, 5.0 volts, and 12.0 to 15.0 volts.

5.2.2 BOARD PULLER

To avoid damage to parts or conductor patterns, a board puller should always be used whenever it is necessary to extract any of the logic rack boards other than the sandwiched pair of memory core boards (B3 and B4). A squeeze-type board extractor is most convenient, but a simple hook-type puller will suffice. Each logic rack board other than the memory sandwich has three puller holes of 1/8-inch diameter spaced approximately 2-13/16 inches between centers along its exposed edge. The wire of a hook-type puller must enter these holes easily, and for insertion clearance between adjacent boards, the inner diameter of the hook (or hooks) should be 1/4 to 3/8 inch. The length of wire from hook or hooks to handling end should be sufficient for finger clearance, and a comfortably shaped handle of wood, metal, or sturdy plastic is recommended in preference to a simple finger-loop, because of the pulling force required.

5.2.3 EXTENDER BOARD

The contacts of the logic rack board receptacles can be probed for logic level checks either by extending the corresponding boards, or by pivoting the logic rack outward to expose the board-to-board wiring. Extending the boards is recommended as safer and more convenient, and two extender boards should be provided for the purpose.

5.2.4 TEST JUMPERS

All internal jumper connections required during troubleshooting are connections between test points which can be made with a set of eight conventional 6- to 10-inch clip-leads, preferably with miniature clips. The external jumper connections are made to and between the male pins (0.032-inch diameter) of modem plug P6 and the female contacts (for 0.032-inch diameter pins) of rear connectors J2, J3 and J4 requiring test leads fitted with appropriate pin adapters to avoid damaging the connectors. A set of six such leads, 3 to 6 inches long, is recommended: three male-to-male, and three female-to-female.

5.2.5 KEYBOARD CONNECTOR EXTENDER

This item is simply a short extender cable with one or both connectors left uncovered (or with the connector-to-connector wiring fanned out to a terminal

block) for electrical access to the individual keyboard lines. The connectors must mate with the keyboard plug and with the corresponding display unit receptacle.

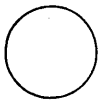
5.3 CHECKOUT AND TROUBLESHOOTING (figure 5-1)

The checkout and troubleshooting diagram (figure 5-1) details, or provides reference to, complete checkout and troubleshooting information for the unit. This troubleshooting diagram, in conjunction with the referenced tables (tables 5-1 through 5-8) and procedures (in the subparagraphs that follow), provides a means for logically analyzing and isolating faults in the equipment.

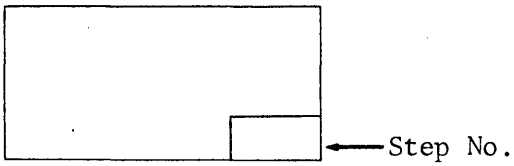
5.3.1 SYMBOLS AND ABBREVIATIONS USED IN TROUBLESHOOTING DIAGRAM AND TABLES

5.3.1.1 Symbols used in Troubleshooting Diagram. Definitions of the symbols used in the troubleshooting diagram are as follows:

TEST IDENTIFICATION CODE
(also used for referencing within the troubleshooting diagram)

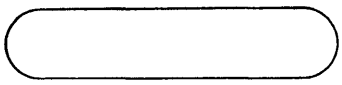


TEST ACTION



SYMPTOM DECISION POINT

CORRECTIVE ACTION
(boards to be replaced)



NOTE: When more than one board is listed, boards are listed in order of probability that they are the cause of the fault. Replace boards one at a time until fault is corrected.

5.3.1.2 Keying Instructions. The keying instructions given in figure 5-1, tables 5-1 through 5-8, and the procedures that follow employ the conventions and abbreviations explained by example in the chart that follows. Alphameric character keys are designated by means of their unshifted keytop labels, and superscript letters are added when a SHIFT key (^{S}) or the CTRL key (^{C}), or both, are to be pressed while the character key is struck. The character designations are enclosed in quotes merely to avoid confusion; the quotemarks themselves are not to be keyed.

Typical Instruction	Keyboard Action
Strike "P"	Strike the P key (press and release it).
Strike "^{C}S"	Strike the S key while pressing the CTRL key.
Strike "^{S}SP"	Strike the space-bar while pressing one of the SHIFT keys.
Strike "^{CS} "	Strike the period (.) key or the decimal point (.) key while pressing both a SHIFT key and the CTRL key.
Repeat "1" for half a line	Press one of the 1 keys while pressing the RPT key, and hold both down until the line is about half-filled with 1's.
Strike ^{S}CLEAR	Strike the CLEAR key while pressing one of the SHIFT keys. (CLEAR is one of 16 keytop labels, including CR, LF, and ESC, for which quotemarks are unnecessary.)
Repeat cursor-right to...	Press and hold down the RPT key, press the right-arrow (→) cursor control key, and hold both down until the specified position has been reached.
Move the cursor to...	Use the HOME key, the TAB key, and/or the arrow-top cursor control keys (with or without the RPT key) in any convenient manner to place the cursor at the specified position.

5.3.2 ANALYSIS OF REPORTED SYMPTOMS

The customer's report of a trouble cannot be expected to pinpoint the part or parts to be replaced, but can often provide enough information to indicate the appropriate procedure in figure 5-1. The procedures are intended to determine whether a fault actually exists and, if so, to localize the fault to one (or a few) of the replaceable items. The procedures are individual portions of a complete functional checkout, and are identified separately by a letter so that only the steps applicable to each type of trouble need be performed. If the type of trouble is not known, the procedures should be performed in the sequence given, because each is based on the assumption that the requirements of the preceding checkout procedures have been met.

In the troubleshooting sub-steps, logic levels are designated as high or low. A high level is a voltage of +3 volts or higher (up to about +4.7 volts), and a low

level is a voltage of +0.4 volt or lower, measured with a multimeter having a sensitivity of 10,000 ohms per volt or higher. When a logic level is expected, voltage of approximately +2 volts represents an open-circuit (defective) integrated circuit output. Pulse or square-wave signals would also appear as open circuits, but the procedures do not specify multimeter measurements of such signals.

5.3.2.1 Turn-On Indications and Display Defects. If the reported symptoms include improper initial indications, defective video, absence of cursor, or no response to the keyboard, begin troubleshooting with the turn-on checks (A of figure 5-1).

5.3.2.2 Typing or Editing Defects. If initial indications, display video, and some typing and/or editing functions are normal, but the response to one or more characters, cursor control, or editing keys is abnormal, begin troubleshooting with the batch typing and editing checks (C of figure 5-1).

5.3.2.3 Printout or Batch Transmit Defects. If operation appears normal except in the print mode or the batch transmit mode, begin troubleshooting with the mode control checks (D of figure 5-1).

5.3.2.4 Communication Defects. If communication through the modem is erratic or impossible, but operation is otherwise normal, begin troubleshooting with the mode control checks (D of figure 5-1).

5.3.2.5 Incorrect Response to Received Characters or Commands. If received characters (including those keyed in full-duplex typing) are not properly displayed, or if received commands do not produce the proper responses, but batch or half-duplex typing and editing appear normal, begin troubleshooting with the half-duplex checks (H of figure 5-1).

5.3.3 TROUBLESHOOTING PRECAUTIONS

5.3.3.1 High Voltages. The only high voltages in the terminal are those within the monitor compartment of the display unit, the 115 vac at the base plate power supply, and the CRT accelerating and deflection yoke voltages. All standard precautions for working on transformer and CRT equipments apply.

5.3.3.2 Avoiding Spot-Burn of the CRT Phosphor. Each time the POWER switch is set to ON, it should be left at ON for at least 30 to 40 seconds (long enough for the CRT cathode to reach operating temperature) before it is set to OFF again. Otherwise there is the danger of a bright long-lasting spot after turn-off that might damage (spot-burn) the CRT phosphor. The TV monitor group does not provide for spot-burn protection under such conditions.

5.3.3.3 Avoiding Power Supply or Circuit Board Damage. To avoid power supply shutdown or actual circuit damage from sudden changes in loading, always make certain that the POWER switch is at OFF before removing or inserting a board or the keyboard cable plug. However, this precaution does not apply to the modem cable or an accessory equipment cable, if any; it is safe to connect or disconnect those cables with the POWER switch at ON. Note that two power switches are provided, one on the keyboard and one on the terminal. The former must be off when the keyboard connector plug is removed to prevent damage to the monitor.

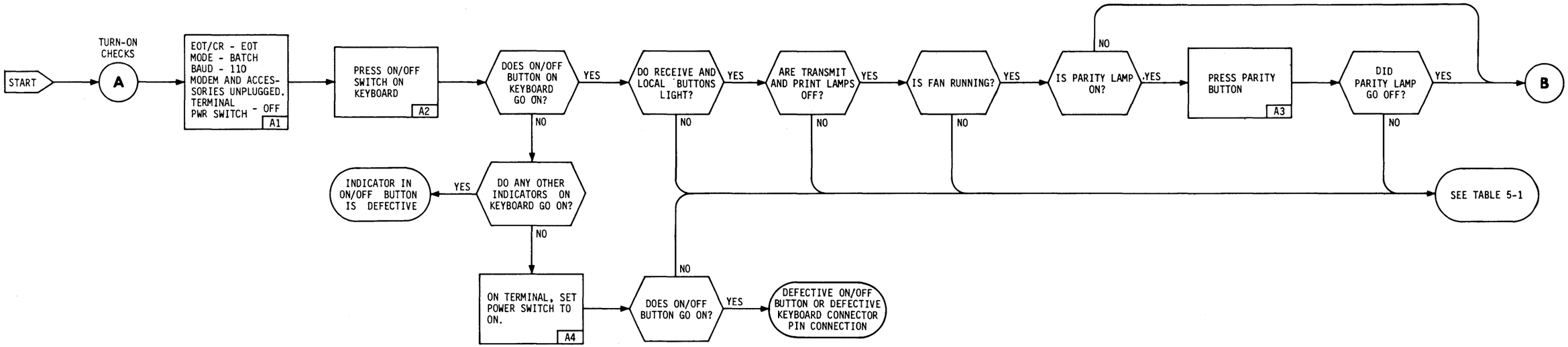
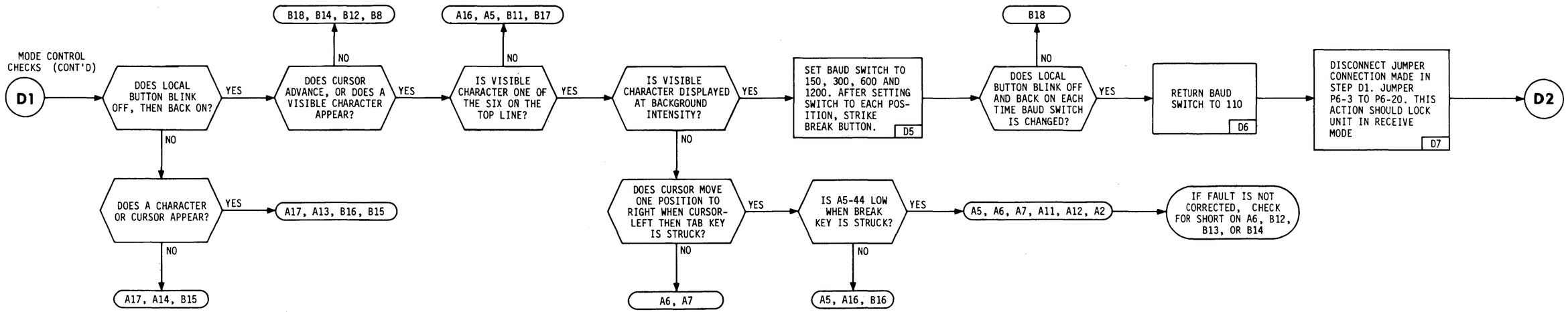


Figure 5-1. Checkout and Troubleshooting Diagram (Sheet 1 of 4)



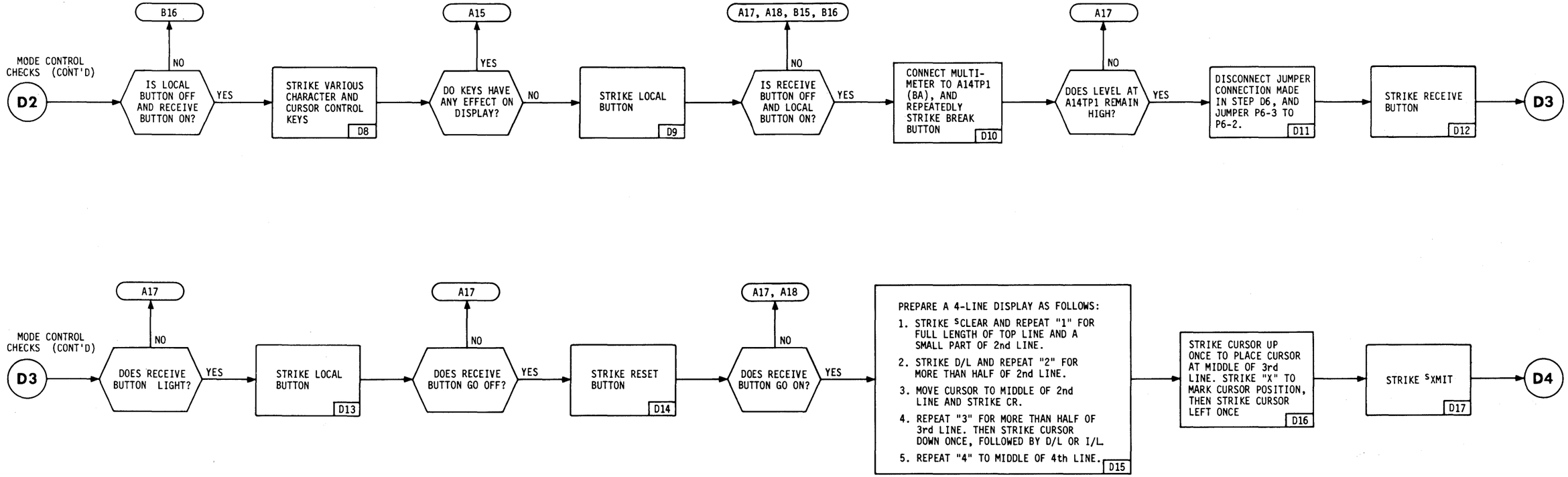


Figure 5-1. Checkout and Troubleshooting Diagram (Sheet 2 of 4)

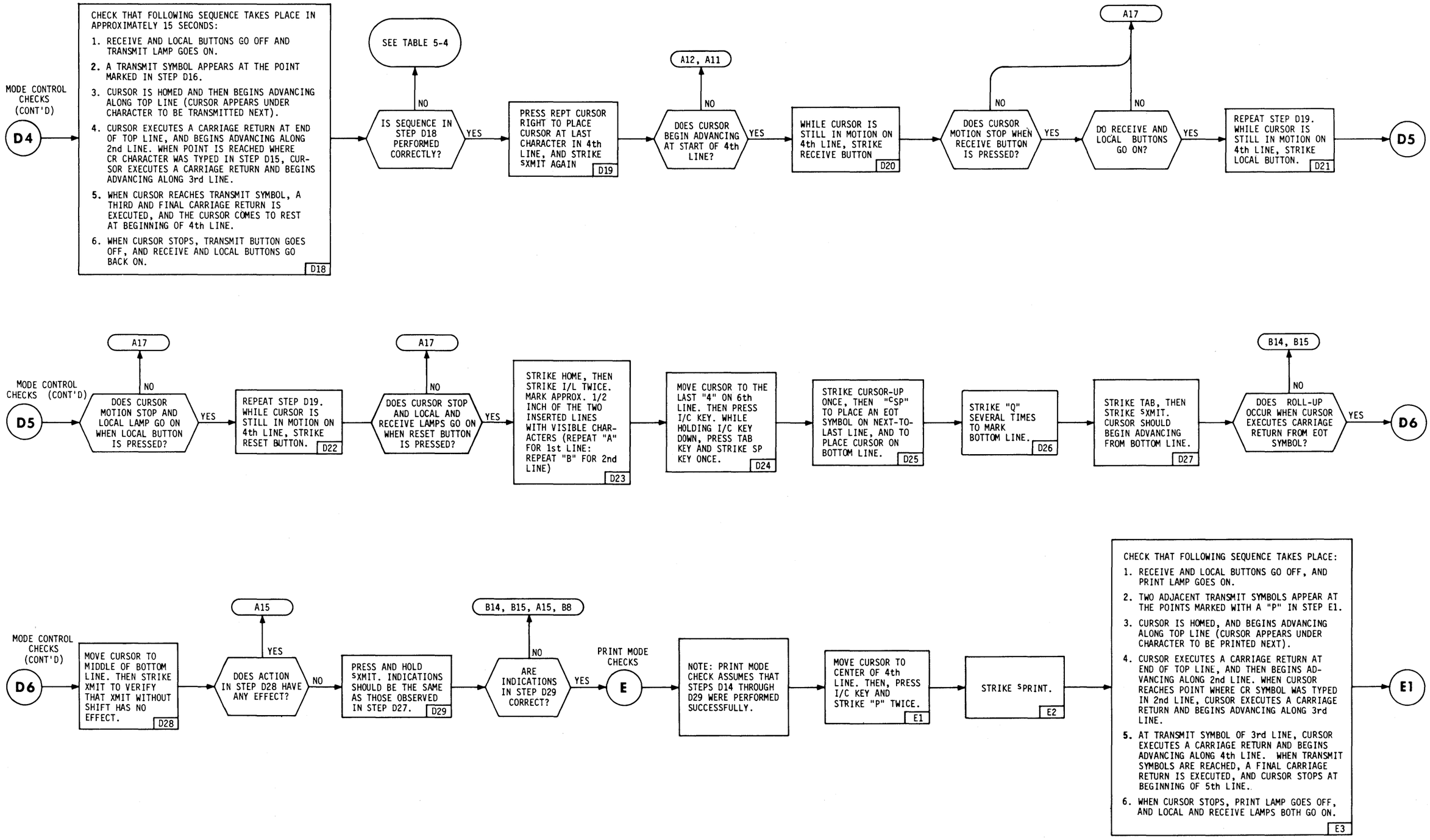


Figure 5-1. Checkout and Troubleshooting Diagram (Sheet 3 of 4)

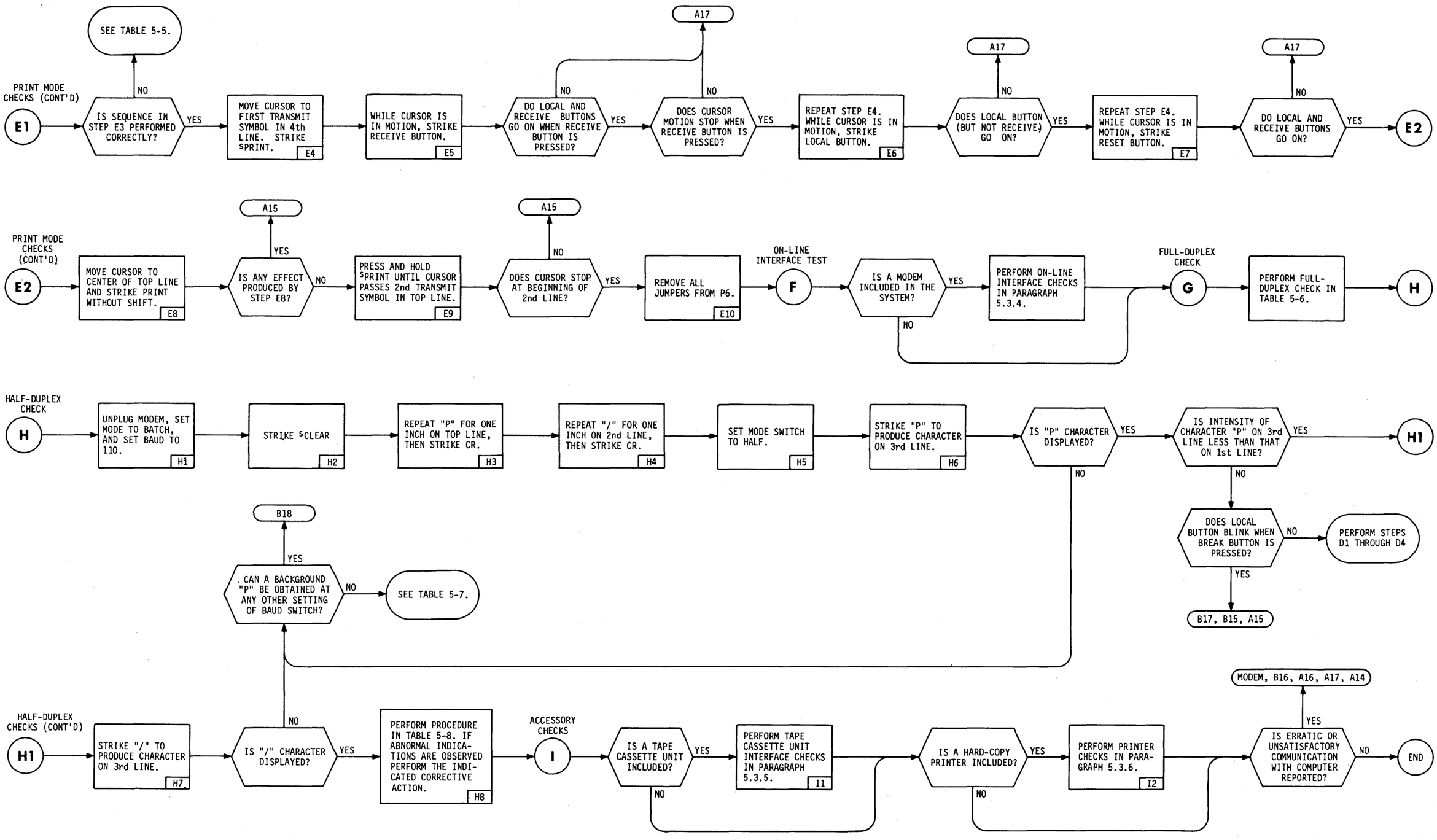


Figure 5-1. Checkout and Troubleshooting Diagram (Sheet 4 of 4)

Table 5-1. Turn-on Check Troubleshooting

Item	Trouble Symptoms	Corrective Action
1	All lamps are off and fan is not running.	1. Check 3.5-amp ac fuse. If fuse is blown, proceed to step 3. If fuse is not blown, proceed to step 2. 2. Check for open circuit in power card, defective POWER switch, or open circuit in internal wiring. 3. Replace fuse (3.5 amp). If fuse blows again, proceed to step 4. 4. Check for short in fan, power supply transformer, and ac relay circuits.
2	Lamps go on, but fan does not run.	Check wiring to fan, and check fan motor.
3	Fan is running, but no lights go on.	1. Press RESET button. If lights still fail to go on, proceed to step 2. 2. After CRT warm-up, check display. If display is completely blank, fault is in power supply. If display contains cursor and/or data, proceed to step 3. 3. Probable cause: B16 or keyboard.
4	RECEIVE button does not light.	Replace A17, A13, or A18.
5	Lamps light, but in wrong combination.	1. Press and release RESET button. If RECEIVE and LOCAL buttons light, replace A13. 2. If either RECEIVE or LOCAL button fails to light when RESET button is pressed, replace B16 or keyboard. Alternate possibilities: A13 or B15 for LOCAL button; A17 or A18 for RECEIVE button.

Table 5-2. Display Check Troubleshooting

Item	Trouble Symptoms	Corrective Action
1	Display remains completely blank.	1. Turn BRIGHTNESS control far enough clockwise to determine whether or not a sweep raster is present. Press and release RESET button to check whether raster is synchronized (not rolling or oscillating). If raster is present and synchronized, proceed to step 2. If raster is abnormal, proceed to step 7. 2. Strike CR key a maximum of five times. If cursor appears, check keyboard input logic for function keys, as directed in paragraphs 5.3.7.2 and 5.3.7.3. If keyboard input logic checks OK, replace B12. Alternate possibilities: B10 or B11. If cursor fails to appear, proceed to step 3. 3. Strike several number or letter keys. If characters appear, replace B10 or B11. If no characters appear, proceed to step 4. 4. Repeat cursor-left for about 3 seconds, then repeat cursor-up for ½ second. Finally, repeat cursor-right for about 3 seconds. If cursor appears at any point in this sequence, replace A15 or A14. Alternate possibilities: A6, B10, or -12-volt power supply. If cursor fails to appear, proceed to step 5. 5. Strike RESET button sharply and repeatedly 6 to 10 times to see whether randomly-timed interruptions of display refresh cycle will insert any character codes into memory, causing characters to appear on CRT. If no characters appear, ground A9TP1 to see whether display fills up with a pattern of thick horizontal lines. If miscellaneous characters appear, or if display fills up with a pattern of thick horizontal lines (27 solid character lines) while A9TP1 is grounded, display refresh logic is cycling, video output logic is able to deliver video, and TV monitor group is able to display video. Replace boards

Table 5-2. Display Check Troubleshooting (continued)

Item	Trouble Symptoms	Corrective Action
1	Display remains completely blank (continued)	A4, A5, B10, B11, B9, B8 and A7 in that sequence, repeating step 2 after each board replacement to check whether cursor appears. If no characters appear and display remains completely blank, proceed to step 6. 6. If entire raster brightens and remains bright while A9TP1 is grounded, replace A8 or A7. If raster brightens when A9TP1 is grounded and then fades to blank, replace A9, A7, or A8. If display remains completely blank when A9TP1 is grounded, troubleshoot display refresh and video output logic as directed in paragraph 5.3.7.4. 7. If sweep raster is not present, or is not synchronized, replace A10, A8, or B13. If replacement does not correct fault, adjust V_{lin} and V_{cf} potentiometers on TV monitor.
2	Data is present, but cannot be synchronized.	Perform step 7 of item 1.
3	Raster and data are synchronized, but cursor is present at more than one position, or is "racing" through display.	Strike RESET button. If cursor settles at one position, replace A13. If cursor still fails to settle, replace B10, B11, or B12. Alternate possibilities: B13 (horizontal movement), B8 or A7 (vertical movement).
4	Display is synchronized and contains data, but focus, brightness, or contrast are unsatisfactory, and cannot be corrected with controls.	Replace TV monitor board.
5	Video is excessively distorted, tilted, or rotated.	CRT yoke requires adjustment or replacement.

Table 5-2. Display Check Troubleshooting (continued)

Item	Trouble Symptoms	Corrective Action
6	Display contains data, and BRIGHTNESS and CONTRAST controls have their normal effect, but characters are broken or non-standard.	Replace A8, A9 or A7.
7	Display contains video noise lines, pulsating video, etc.	Replace A8, A9 or A7.
8	Display consists of thick horizontal lines (solid unblanked character lines).	Replace A8 or A9.
9	Cursor advances normally, but characters do not appear.	Perform step 5 of item 1. If board replacement fails to correct fault, troubleshoot keyboard input logic for character keying, as directed in paragraphs 5.3.7.2 and 5.3.7.3. If keyboard input logic is normal, troubleshoot character generator logic as directed in paragraph 5.3.7.5. If ground of A9TP1 produces any effect other than a solid pattern of thick horizontal lines, replace B9, A9 or A7. If grounding A9TP1 has no effect, replace A7, A4, A5, or check -12-volt power supply.
10	Cursor does not move when character keys are struck.	Troubleshoot keyboard input logic for character keying, as directed in paragraphs 5.3.7.2 and 5.3.7.3. If keyboard input logic is OK, replace A11, B12, A6, or A14.
11	Characters appear and cursor moves, but SCLEAR does not clear display or home cursor.	1. Strike SCLEAR again, and watch for a brief blink of display. If blink occurs, proceed to step 2. If blink does not occur, check keyboard input logic for CLEAR, as directed in paragraphs 5.3.7.2 and 5.3.7.3. If keyboard input logic is OK, replace A13 or A7. 2. Replace A11, A13, A7, A4, or A5. Alternate possibilities: A11, B1, B2, B6 or B7.

Table 5-2. Display Check Troubleshooting (continued)

Item	Trouble Symptoms	Corrective Action
12	Sweep raster cannot be brought to black level, or CONTRAST control is erratic, noisy, or has insufficient range.	Replace TV monitor board.
13	Display contains various characters and spaces, and is not cleared by SCLEAR.	Replace A7 or A13. Alternate possibilities: A11, B1, B2, B6, or B7.
14	Display is filled with a single character, and is not cleared by SCLEAR.	1. Press ON/OFF button to turn off power, wait three seconds, and press ON/OFF button again. Then strike SCLEAR. If clear action is still not performed correctly, check power supply voltages, as directed in paragraph 5.3.7.1. If power supply voltages are OK, proceed to step 2. 2. If displayed character is transmit symbol, replace A12 or B8. Alternate possibilities: A2 or -12-volt power supply. If displayed character is not transmit symbol, interchange A1 and A2, and again strike SCLEAR. If a different character is then displayed, or if display is cleared properly, replace A1 or A2, or both. If interchanging A1 and A2 does not produce a different character, and does not result in a cleared display, troubleshoot character generator logic, as directed in paragraph 5.3.7.5. If character generator logic is satisfactory, replace A4, A5, A13, A11 or A7.

Table 5-3. Batch Typing and Editing Checks

Item	Test Action	Corrective Action if Indication is Abnormal
1	Strike "P" several times in succession, then strike "/" the same number of times. Characters should appear, and cursor should advance each time a key is struck.	1. If character errors occur in alternate position, replace B2, B6, B8 or B9. 2. If wrong character appears, troubleshoot keyboard input logic for defective key, as directed in paragraphs 5.3.7.2 and 5.3.7.3. If keyboard input logic checks OK, replace A1, A2, A5, A9 or A4. 3. If cursor does not advance correctly, replace B10 or B11. 4. If either character appears in more than one position at a time, appears at one or more positions other than the cursor position, or appears only at certain positions, trouble is in memory address logic or memory. Repeat a character through four or more lines and examine error pattern. If pattern is vertical, or if entire line is skipped, replace B8, B9, B10 or B11. If pattern is diagonal, replace B1, B2, B6 or B7. Alternate possibilities: B8 or B3/B4.
2	Strike "SP" and "S/" several times each, and check that characters "@" and "?" appear. Repeat with the other shift key.	1. If correct character appears when one shift key is used, but not when the other is used, replace keyboard. 2. If characters are abnormal for either shift key, perform step 2 or 4 (depending on symptoms) of item 1.
3	Strike "M" and "SSP" keys several times, and check that characters "M" and "0" (zero) appear.	1. Check keyboard input logic for appropriate keys, as directed in paragraphs 5.3.7.2 and 5.3.7.3. 2. If keyboard input logic checks OK, troubleshoot character generator logic, as directed in paragraph 5.3.7.5.
4	Strike CR and check that a carriage return is executed.	1. If CR has no observable effect, perform step 1 of item 3. If fault is not corrected, replace B12 or A13.

Table 5-3. Batch Typing and Editing Checks (continued)

Item	Test Action	Corrective Action if Indication is Abnormal
4	Strike CR and check that a carriage return is executed (continued)	2. If CR moves cursor to any position other than beginning of next line, replace B12 or A13.
5	Strike "CSP" several times, and check that transmit symbol appears. Then strike "CM" and check that a carriage return is executed.	1. If neither transmit symbol nor carriage return is observed, perform step 1 of item 3. 2. If carriage return is executed, but transmit symbol does not appear, replace A12 or B8. Alternate possibilities: A7 or A14.
6	Without pressing SHIFT key, strike CLEAR, then CLR/FG keys, and check that neither key affects the display.	Replace A15.
7	Without using the SHIFT key, strike each character key on the keyboard (including the 12 keys of the numeric pad). Use the CR key for a convenient line arrangement. Check that each key (except SP, LF, ESC, RUBOUT, RPT and CR) produces the correct visual character on the display.	1. If, for any key, neither a character nor a cursor advance is produced, perform step 1 of item 3. 2. If cursor advance occurs but a blank or an incorrect character is produced, strike the key again with the cursor at a different position to see whether the error is associated with a particular position. If the character appears correctly at some positions, proceed to step 3. If character does not appear correctly at any position, proceed to step 2 of item 1. 3. If wrong characters appear only at certain positions, perform step 4 of item 1. If an error pattern appears, perform the corrective actions in step 4 of item 1. If there is no apparent pattern, replace B1, B2, B6, B7 or B3/B4.

Table 5-3. Batch Typing and Editing Checks (continued)

Item	Test Action	Corrective Action if Indication is Abnormal
8	Repeat item 7, except press SHIFT key and strike those keys that have dual (shifted and unshifted) functions on the keytop labels. Check that each key produces the correct visual character on the display.	Perform steps 1 through 3 of item 7.
9	Strike "^{C}LF", "^{C}ESC", and "^{C}RUBOUT" one at a time, and check that characters "J", "[", and "?" appear.	Perform step 1 of item 3.
	NOTE In the remaining steps of this procedure, troubleshoot keyboard input logic as directed in paragraphs 5.3.7.2 and 5.3.7.3 if board replacement fails to correct the fault.	
10	Strike cursor-up, -right, -down, and -left keys one at a time, and verify that each key causes cursor to move one step in proper direction. Then strike HOME, and verify that cursor is homed.	Replace B12, A13, B10, B11.
11	Repeat cursor-right. Cursor should advance along top line and stop at last position in approx. 5 seconds. Then repeat cursor left, and check that cursor stops at home position.	If RPT key has no effect on cursor movement, perform step 1 of item 3. If cursor does not stop at either end of line, replace B12 or B11.
12	Repeat cursor-down, and check that cursor moves down and stops at last line.	Replace B12, B11, B10.

Table 5-3. Batch Typing and Editing Checks (continued)

Item	Test Action	Corrective Action if Indication is Abnormal
13	Move cursor to a line containing characters. Press I/C key, and, while holding this key down, strike "A" several times. Check that a character is inserted each time "A" is struck, causing all other characters in the display to move rightward (characters at end of a line move to the next line). Then, strike D/C and check that the inserted characters are deleted one at a time, causing all other characters to move back to their original positions.	Replace A15, keyboard, B12, B11, or A7. Alternate possibilities: B10, B13.
14	Press cursor-down to move cursor to any blank line other than the bottom line. Repeat "1" and check that character repeat continues with an automatic carriage return from one line to the next. Then, move the cursor down to the bottom line. Repeat "1", and check that character repeat stops at end of bottom line.	Replace B12, A12, B11, B10.
15	Enter a visible character at last position of bottom line. Then, with the cursor under this character, strike CR, and check that character disappears but cursor does not move.	1. If CR moves cursor from last position on bottom line, replace B12 or B8. 2. If CR on bottom line causes roll-up, replace B12, B14, B8, A15.
16	Strike HOME to home cursor. Then strike TAB, and check that cursor moves to last position of bottom line.	Replace A11, A12, B13, A15.

Table 5-3. Batch Typing and Editing Checks (continued)

Item	Test Action	Corrective Action if Indication is Abnormal
17	Strike SCLR/FG and check that display is cleared and cursor is homed.	Replace A13, A15.
18	While holding TAB and RPT, press and hold "P", and check that display is completely filled with the "P" character.	Perform step 3 of item 7.
19	While holding TAB and RPT, press and hold "/", and check that display is completely filled with the "/" character.	Perform step 3 of item 7.
20	Strike HOME, then strike I/L, and check that top line is cleared of "/" characters.	1. If top line is not cleared, replace A12, A13, A6, A11. 2. If one or more of the "/" characters changes to a blank or another character, perform step 3 of item 7.
21	Strike I/L 15 or 20 more times, and check that resulting roll-down clears the upper portion of the display, without altering the remaining "/" characters.	Perform step 3 of item 7.
22	Strike D/L, and check that the remaining "/" characters are rolled up one line without changing to blanks or other characters.	1. If a one-line roll-up is not obtained, replace A12, A13, B12, A7, A6. 2. If one or more of the "/" characters changes to a blank or another character, perform step 3 of item 7.
23	Strike D/L 15 or 20 more times, until topmost line of "/" characters reaches top line of the display, and again check that none of the "/" characters have changed.	Perform step 3 of item 7.

Table 5-4. Troubleshooting in the Transmit Mode

Item	Trouble Symptoms	Corrective Action
1	Transmit sequence takes place in considerably less than 15 seconds.	Replace B18 or A18.
2	Cursor does not appear under next character to be transmitted.	Replace A12.
3	No effect is produced by SXMIT.	1. Strike SXMIT again, and watch for the display to blink. If display blinks, replace B15 or B12. If display does not blink, proceed to step 2. 2. Check keyboard input logic for XMIT, as directed in paragraphs 5.3.7.2 and 5.3.7.3. If keyboard input logic checks OK, replace A11.
4	Transmit symbol appears, but cursor is not homed.	Replace B12.
5	Cursor is homed but a blank or a character other than the transmit symbol appears.	Replace A5 or B15.
6	Display goes blank.	Replace A11.
7	Transmit lamp does not go on.	Replace A15, A17, or A12. Alternate possibilities: B16 or keyboard.
8	Cursor fails to advance from home position.	Replace A17 or B14. Alternate possibilities: B18, B12, B14.
9	Cursor fails to execute carriage return at end of top line.	Replace B12.
10	Cursor fails to execute carriage return at transmit symbol.	Replace B14.

Table 5-4. Troubleshooting in the Transmit Mode (continued)

Item	Trouble Symptoms	Corrective Action
11	Cursor does not stop at beginning of 4th line, and TRANSMIT lamp remains on.	Replace A17.
12	LOCAL button does not go on after cursor comes to rest.	Replace A16, A17, or B19.

Table 5-5. Troubleshooting in the Print Mode

Item	Trouble Symptoms	Corrective Action
1	SPRINT has no effect.	1. Strike SPRINT again, and watch for the display to blink. If blink occurs, replace B15. If blink is not observed, proceed to step 2. 2. Troubleshoot keyboard input logic for PRINT, as directed by paragraphs 5.3.7.2 and 5.3.7.3.
2	Transmit symbols appear, but cursor is not homed.	Replace A13.
3	Cursor is homed, but one or both transmit symbols fail to appear.	Replace A11.
4	PRINT lamp does not light.	Replace A12 or A17. Alternate possibilities: B15 or keyboard.
5	Cursor does not advance from home position.	Replace B14 or B17.
6	Cursor does not execute a carriage return when CR character is reached in 2nd line.	Replace A12 or B14. Alternate possibility: B15.
7	Cursor does not stop at beginning of 5th line, and PRINT lamp stays on.	Replace A17. Alternate possibilities: B14 or B15.

Table 5-6. Full-Duplex Check

Item	Test Action	Corrective Action if Indication is Abnormal
1	Set MODE to BATCH, and set BAUD to 110. Jumper P6-2 to P6-3, and jumper P6-5, P6-6, P6-8 and P6-20 together. Strike SCLEAR, then press RESET button.	None
2	In sequence, type characters A through Z and 1 through 0, then strike CR.	None
3	Set MODE to FULL, and repeat item 2. Note that all characters are repeated at background intensity.	1. If data is not repeated, replace B18, A14, B15, A18, B16, B19. 2. If data is repeated with some characters changed, replace B18, A14, A18. 3. If data is repeated at foreground intensity, replace A16, B16.

Table 5-7. I/O Bit Troubleshooting

NOTE: For character "P", bits 7 and 5 are high, bits 6, 4, 3, 2 and 1 are low. For character "/", bits 7 and 5 are low, bits 6, 4, 3, 2 and 1 are high.

I/O Bit Number	Test Point (measure high or low while striking character key)	Corrective Action if Bit State is Incorrect
7	A16-21	Replace A5, B11, A18, B17, B16.
6	A16-42	Replace B16, B17, B11.
5	A16-11	Replace A5, B11, B16, B17, A18.
4	A16-46	Replace A5, B11, B16, B17, A18.
3	A16-15	Replace A5, B11, B16, B17, A18.
2	A16-45	Replace A5, B11, B16, B17, A18.
1	A16-14	Replace A5, B11, B16, B17, A18.

Table 5-8. Half-Duplex Checks

Item	Test Action	Corrective Action if Indication is Abnormal
1	Strike "^{S}N" and check that up-arrow appears.	Replace A5, A16, A14.
2	Strike cursor-left once, then CR, and check that up-arrow disappears and that a carriage return is executed.	Replace A16, A15, B15, B14.
3	Strike LF, then "^{SC}/" and check that the cursor does not advance.	Replace A16, B14, A14.
4	Strike "^{SC}P" and check that the cursor does not advance.	Replace A16 or B14.
5	With EOT/CR switch set to CR, strike "^{C}D" and check that the cursor does not advance. Then, strike "^{CS}." and "^{C}D" in that order, and check that cursor remains stationary for both characters.	Replace A16 or B15.
6	While holding the CTRL and SHIFT keys down, strike each of the ten number keys "1" through "0"), and the SP key, typing at a slow rate. Check that characters "A" through "I", the transmit symbol, and character "P" appear, in that order.	Replace B17 or A16.
7	Strike "^{C}H" twice and check that cursor moves left two positions to the transmit symbol typed in item 6, without changing or erasing either the "P" or the EOT symbol.	Replace B15, A16, B17.
8	Strike "^{CS}." then CR, and check that a carriage return is executed without erasing the transmit symbol.	Replace B15, A16.
9	Fill at least one line by alternately striking "G" and "8" as rapidly as possible, so that some "G" and "8" characters fail to appear. Check that line contains no characters other than "G" and "8".	Replace A14.

Table 5-8. Half-Duplex Checks (continued)

Item	Test Action	Corrective Action if Indication is Abnormal
10	Set BAUD switch to 150, 300, 600 and 1200. In each position, strike "P", "/", and CR in succession. Check that correct characters, plus carriage return, are produced at each BAUD switch setting.	Replace B18.
11	Set BAUD switch to 300, and retain this setting until directed to change it.	None
12	Strike TAB to place cursor at last position of bottom line. Then strike "Q" and check that the resulting carriage return is accompanied by a one-line roll-up, deleting the top line.	Replace B14, B15. Alternate possibilities: A13, B12, B8.
13	Repeat cursor-right to about middle of bottom line. Then strike CR and check that resulting carriage return is accompanied by a one-line roll-up, deleting the top line.	Replace B14.
14	Mark first position on bottom line with an "X". Then strike "CSB" and check that cursor moves one position rightward and a blank appears.	Replace A16.
15	Strike TAB to place cursor at last position of bottom line. Then strike "CS." followed by "CSB", and check that cursor is homed.	Replace A16, A14, A13.
16	Strike "CSJ" and check that a blank appears, and that cursor moves one position rightward.	Replace A16, B15.
17	Strike "CS." then "CZ", and check that a one-line roll-down occurs.	Replace A16, A12, B15.
18	Strike "CSC" and check that a blank appears, and that cursor moves one position rightward.	Replace A16.
19	Strike "CS." then "CSC", and check that a one-line roll-up occurs.	Replace A16, A15.

Table 5-8. Half-Duplex Checks (continued)

Item	Test Action	Corrective Action if Indication is Abnormal
20	Strike I/L repeatedly until the line filled with "G" and "8" in item 9 reaches the bottom line. Then strike D/L key repeatedly until the line filled with "G" and "8" reaches the top line. Check that the entire line of "G" and "8" characters is still at background intensity.	Memory threshold voltage is unsatisfactory. Refer to paragraph 5.4.3.
21	Strike "CS." then "CSJ", and check that the line of "G" and "8" characters rolls down one line. Set MODE to BATCH. Then repeat "1" for part of top line, strike CR twice, and repeat "2" for the line below the line filled with "G" and "8" characters. Set MODE to HALF, strike "CSM", and check that a blank appears, and that cursor moves one position rightward.	Replace A16.
22	Strike "CS." then "CSM", and check that all of the foreground characters entered in item 21 have disappeared, that the line of "G" and "8" characters remains and is unchanged, and that cursor is homed.	Replace A16, A15.
23	Strike "CSL" and check that a blank appears, and that cursor moves one position rightward.	Replace A16.
24	Strike "CS." then "CSL", and check that line of "G" and "8" characters disappears, and that cursor is homed. All characters should be cleared from screen.	Replace A16, A15, B19.
25	Strike "1" to mark home position, and repeat cursor-right and cursor-down for one second to move cursor away from home. Strike "CSA" then "S/", then "/", and check that a blank, a "?", and a "/" appear in succession, as keyed.	Replace A16.
26	Strike "CS." then "CSA", and check that cursor is homed.	Replace A16, B14, A13.

Table 5-8. Half-Duplex Checks (continued)

Item	Test Action	Corrective Action if Indication is Abnormal
27	Strike "?" then "/", and check that cursor moves right and down to a position approximately one inch from the right-hand end of the middle line of the display.	Replace B14, B11.
28	Strike cursor-right once, then cursor-down once, and mark this position with an "X". Strike "CS." then "CQ", then "SP", then "P". The cursor should appear under the "X" character.	Replace B11.
29	Strike "CS." followed by "/", and observe that cursor does not move but begins blinking.	Replace A16, B17, A6, B10.
30	Strike "CS." followed by "?", and observe that cursor stops blinking.	Replace A16, B17.
31	Strike cursor-right once, then strike "CS0", and check that a blank appears, and that cursor moves one space rightward. Strike "X", and check that it appears as a background character.	Replace A16.
32	Strike "CS." then "CS0", and check that cursor remains stationary.	Replace A16.
33	Strike "X" and check that "X" character appears at foreground intensity.	Replace A16, B17, A5, A7.
34	Strike RESET button, then strike "X" again, and check that the "X" character is displayed at background intensity.	Replace A16.
35	Strike SCLEAR.	
36	Fill the first two inches of top line by striking "A" about 24 times. Strike "CS." followed by "CS0", and fill the next inch of the top line by striking "B" about 12 times. Then strike cursor-down once to place cursor on second line, and strike "CSI". Check that a blank appears, and that cursor moves one position rightward.	Replace A16.

Table 5-8. Half-Duplex Checks (continued)

Item	Test Action	Corrective Action if Indication is Abnormal
37	Strike "CS." then "CSI" and check that cursor remains stationary. Then strike "A" and check that it appears as a background character.	Replace A16.
38	Strike D/L and repeat item 36 to place a two-inch background field (character "C") and an inch of visible foreground data (character "D") on the second line. Strike CR, then "CS.", then "CSI", and repeat item 36 to place a two-inch background field (character "E") and a one-inch foreground field (character "F") on the third line. These actions provide a suitable display for the remainder of this procedure.	
39	With cursor at last character in third line, strike "CSN", and check that a blank appears, and that cursor moves one position rightward.	Replace A16, B15.
40	Strike cursor-left once. Then set BAUD switch to 110.	
41	Strike "CS." then "CSN", and check that PRINT light goes on, and print mode is initiated. Two transmit symbols should appear.	Replace A16, B15.
42	Delete the two transmit symbols produced in item 41 by placing the cursor under the first transmit symbol, then striking SP twice. With the cursor at the last character in the third line, strike "CN" and check that a blank appears, and that cursor moves one position rightward.	Replace A16.
43	Strike cursor-left once to move cursor back into position. Then strike "CS.". When ready press and hold down CTRL key and simultaneously (within about 0.1 second) strike "CN". Check that TRANSMIT light goes on, and that half-duplex transmit mode is initiated. The sequence of actions	1. If half-duplex transmit mode is not initiated, replace B15, A16. 2. If half-duplex transmit mode is initiated, but background characters

Table 5-8. Half-Duplex Checks (continued)

Item	Test Action	Corrective Action if Indication is Abnormal
43 (cont)	should be as described in step D18 of figure 5-1, except that cursor skips over background characters.	are not skipped, replace B14, A17. 3. If EOT symbol does not appear, "^{C}N" was probably not released quickly enough.
44	Set MODE switch to BATCH, and strike HOME to place cursor under first character of top line. Then strike SP, followed by one or more character keys different from the background characters in top line. Note that background field remains unchanged.	Replace B12.
45	Position cursor under a background character and strike CR. Note that resulting carriage return has no effect on the background character.	Replace B12.
46	Strike HOME, then strike TAB once, and check that cursor moves to first foreground character in top line. Strike TAB three times in succession, and note that cursor advances successively to first foreground character in second line, first foreground character in third line, and finally to last position in bottom line.	Replace A11, B13.
47	Strike cursor-up once. Set MODE to HALF. Strike "^{C}S." then "^{C}SI". Strike "Q" and return MODE to BATCH. Strike cursor-up twice. Strike TAB once, and check that cursor has moved to the first character position of bottom line.	Replace B13, A11.
48	Strike TAB several times, and note that cursor does not move.	Replace A11.
49	Strike HOME. While holding down I/C key, strike SP and "X". Check that display remains unchanged except for a brief blink.	Replace A12.

Table 5-8. Half-Duplex Checks (continued)

Item	Test Action	Corrective Action if Indication is Abnormal
50	Strike D/C several times, and check that display remains unchanged except for a brief blink.	Replace A12.
51	Strike TAB. Cursor should move to first foreground character of top line. While holding down I/C, strike "SP" and one or more character keys at least three times. Check that: a. Cursor remains stationary. b. Keyed characters appear in sequence at cursor position. c. Foreground characters on top line move rightward as keyed characters appear. d. Background and foreground characters on 2nd and 3rd line are neither moved nor altered.	Replace A12.
52	Strike D/C three times and check that: a. Cursor remains stationary. b. Character at cursor position is deleted each time D/C is struck. c. All other foreground characters move leftward as deleted characters disappear. d. Background and foreground characters on 2nd and 3rd lines are unaffected.	Replace A12.
53	Strike SCLR/FG, and check that all foreground characters disappear, and that background characters are unaffected.	Replace A13, A15, or keyboard.

5.3.4 ON-LINE INTERFACE TEST

This procedure checks the ability of the terminal to operate on-line with a modem or CPU interface. Proceed as follows:

1. Plug P6 (EIA) into the modem, and turn modem power on.
2. On the terminal, make the appropriate settings for the BAUD and CA switches.
3. On the terminal, make the following control settings:

Control	Setting
EOT/CR	CR
PARITY	1
MODE	BATCH
POWER	ON

4. Obtain dial tone and dial appropriate telephone number.
5. Follow sign-on procedure required and run CPU program available.
6. Enter appropriate control code to ask for program list. Press BREAK button and observe transmission stops. Send a CR and observe CPU program continues. Sign off with appropriate procedure.
7. If sign-on steps failed, check boards B18, A18 or B19, possibly B16. Alternate possibility is faulty modem or noisy line.
8. If program showed data errors, check boards A18, B19 or A17.
9. If BREAK operation failed, check boards A14, A17 or A18.
10. If KB parity error light is intermittently on during transmitting or receiving, check A18 (parity generation/checking circuits).
11. In using a 202C type modem at 1200 baud, EOT can be selected at the CR/EOT switch for an EOT turnaround mode. Should this mode fail, check A18.

5.3.5 TAPE CASSETTE UNIT INTERFACE CHECKS

If the unit has met all requirements of the batch typing and editing checks (D, figure 5-1), and the half-duplex checks (H, figure 5-1), correct functioning of the tape cassette unit interface may be verified by the following procedure. If abnormal indications are observed, replace B17 or B19. Alternate possibilities: B15, B14, A14. If board replacement fails to correct the fault, check the tape cassette interconnections, and the tape cassette unit itself.

1. Connect the cable between J2 on the terminal and the corresponding connector on the tape cassette unit. Make the following control settings:

Control	Setting
BAUD	300
FULL/HALF/BATCH	BATCH
POWER	ON

2. If the left-hand CL indicator on the tape cassette unit is not lighted, strike the left-hand REWIND button. Then, when the left-hand CL button lights, strike the left-hand RESET button, then the RECORD and INTERLOCK buttons together. Note that the CL indicator goes off. Depress the ON LINE button to extinguish the indicator, then press BLOCK/CONT button so that CONT is lighted. The PAGE indicator should be on, and the DUPLICATE indicator should be off.

3. On the keyboard, strike ^{S}CLEAR to clear the display and home the cursor.

4. Type at least ten "1" characters. Then strike CR. Observe that a record of data line is made. Cursor scans data and stops at beginning of next line.

5. Type "2" for at least ten characters, depress CR and observe record made of line 2.

6. Type "3" for at least ten characters, depress CR and observe record made of line 3.

7. Depress RESET on tape cassette, then left-hand REWIND. Now completely press PLAYBACK and observe that VDT displays all data previously recorded.

8. Rewind left-hand tape cassette. Press ON LINE button, select BLOCK mode, then select PLAYBACK. Set mode switch to HALF, enter "^{C}Q" and observe that screen is first cleared, then a row of "1" is received, after which the VDT goes into transmit mode and transfers data.

9. Enter "^{C}Q" and observe that screen clears, receipt of "2"s, and transmit mode.

10. Repeat step 9 and observe same operation for "3"s.

11. Rewind left-hand tape cassette, select CONT, and press left-hand PLAYBACK switch. Observe that results of steps 8, 9 and 10 occur automatically.

12. Rewind left-hand tape cassette, select BLOCK, press ON LINE to off, and press left-hand RECORD/INTERLOCK switches.

13. Set MODE switch to HALF, type about 20 random characters, CR. Strike "^{C}S." then "^{C}S0" and 20 random characters. Observe data in full (foreground) intensity. Press ^{S}PRINT, observe normal PRINT (two transmit symbols) operation, but at much higher cursor speed than 300 baud.

14. Press RESET on tape cassette, rewind left-hand tape cassette, enter SCLEAR on KB, and press left-hand PLAYBACK switch. Observe that all data is displayed in foreground intensity, but that transmit symbols are not displayed from tape cassette.

15. Press RESET on keyboard, enter SCLEAR, rewind left-hand tape cassette, and press left-hand PLAYBACK switch. Observe that first line of data is displayed in background intensity and second line in foreground intensity.

5.3.6 HARD COPY PRINTER CHECKS

If the unit has met all requirements of the batch typing and editing checks (C, figure 5-1), the mode control checks (D, figure 5-1), the print mode checks (E, figure 5-1), and the half-duplex checks (H, figure 5-1), correct functioning of the hard copy printer interface may be checked by the procedure in paragraph 5.3.6.1 (parallel-bit off-line printer interface) or paragraph 5.3.6.2 (teletype printer interface).

5.3.6.1 Parallel-Bit Off-Line Printer Interface Check. To check the interface for this type of printer, proceed as follows. If abnormal indications occur, replace boards B17, B16, B19.

1. Connect the accessory cable between J3 on the VDT and the corresponding connector on the printer.

2. On the VDT, make the following control settings:

Control	Setting
BAUD	300
FULL/HALF/BATCH	HALF
POWER	ON

3. On the printer, press the ON LINE button to light the indicator, then press the CR/LF button.

4. Compose a suitable display (three or more lines) as follows:

    a. Strike SCLEAR to clear display and home cursor.

    b. Type A through Z, 1 through 0, CR, and "1" for at least half a line. Depress CR and repeat "2" for at least half a line. Press SXMIT on keyboard. Observe that data is reproduced and formatted correctly at printer.

c. Set mode switch to HALF, press SCLEAR, enter LF twice at keyboard and observe printer executing two line feeds. Enter characters A through Z, 1 through 0, CR, LF, A through Z, 1 through 0, CR, LF. Observe data is reproduced and formatted correctly at printer. Press ON LINE button so that it is not lighted.

d. Set mode switch to BATCH, press ON LINE so it is not lighted, enter SCLEAR, A through Z, 1 through 0, and CR.

5. Strike SPRINT and observe that a pair of transmit symbols is entered and that the cursor movement through the display is normal for the print mode.

6. Enter CSP, cursor-left, cursor-down, repeat "1" for half a line, and repeat cursor-right until cursor reaches right edge of screen.

7. Enter "2" twice, repeat "3" for half a line, and press SPRINT.

8. Observe that data is reproduced and formatted correctly at printer. (Note transmit symbol is interpreted as CR.)

9. Position cursor under first transmit, enter SPRINT and observe printer double-spaces between record formats correctly.

5.3.6.2 Teletype Printer Interface Check. To check the interface for this type of printer, proceed as follows. If abnormal indications occur, replace boards B16, A17, B17.

1. Connect accessory cable between J3 on VDT and the corresponding connector on the teletype. Turn on the teletype.

2. On the terminal, make the following control settings:

Control	Setting
BAUD	110
FULL/HALF/BATCH	HALF
POWER	ON

3. Perform step 4 of the procedure in paragraph 5.3.6.1 to prepare a suitable display.

4. Strike SPRINT and observe that a pair of transmit symbols is entered and that cursor movement through display is normal for print mode.

5. Check that teletype printout matches display in content and format (except that the teletype printer does not differentiate between background and foreground characters, and does not print the transmit symbols).

6. Turn off teletype printer. Then, unless another hard copy interface check is to follow, strike SCLEAR, and set POWER switch to OFF.

5.3.7 DETAILED TROUBLESHOOTING PROCEDURES

The subparagraphs that follow provide multimeter-and-clip-lead test procedures for the power supply (5.3.7.1), the keyboard (5.3.7.2), the keyboard input logic (5.3.7.3), the display refresh and video output logic (5.3.7.4), and the character generator logic (5.3.7.5). A high level is a voltage of +3 volts or higher, and a low level is a voltage of +0.4 volt or lower. A voltage of approximately +2 volts is an open circuit (defective integrated circuit output).

5.3.7.1 Power Supply. When the power supply is suspected, first visually inspect the CRT screen and keyboard lights for symptoms and clues, then measure the power supply voltage (use digital voltmeter) at the test points using the chassis (ground) as the meter common. If all five voltages agree with the values listed in table 5-9, the trouble is not in the logic power supply.

Table 5-9. Power Supply Voltage Checks

Test Point	Voltage	Use
+5 logic	+5.0 volts ±5%	Used in keyboard and on all logic rack boards except B3 and B4.
+15 memory	+13.0 volts ±5%	Used on A1, A2, B1, B2, B6 and B7.
+15 monitor	+15.0 volts ±5%	Monitor power.
+12 volts	+12.6 volts ±10%	Used on A9, B16, B17.
-12 volts	-12.6 volts ±10%	Used on A9 and B16.
-5 volts	-5.0 volts ±5%	Used on A1 and A2.

5.3.7.1.1 Isolating the Faulty Voltage Supply.

a. Visually Identifying the Faulty Voltage. The faulty voltage supply can usually be quickly identified by inspecting the CRT screen and/or the keyboard lights. If the power on/off light on the keyboard does not turn on, then either the 1/8-amp relay fuse (F8) or the 3.5-amp line fuse (F6) is blown. Check and replace blown fuse. If it is the 3.5-amp fuse that has blown, inspect to see if it has blown

hard (fuse is black inside). If the fuse is blown hard, the cause is very likely to be a shorted bridge rectifier, CR22, CR24, CR25, or CR26 (most likely CR22 or CR25). The bridge rectifiers may be checked with an ohmmeter. (It is not necessary to disconnect the wires to the bridges.) Refer to power supply schematic for internal arrangement of diodes. With the ohmmeter on the RX1 scale, check each internal diode for shorts with forward and reverse polarity. Replace any defective bridge rectifier.

b. CRT Screen. If +5 volts is down, there will be no keyboard lights on (except for the power ON/OFF indicator), no raster, no video and no cursor. If +15 volts (MEM) is down, the screen will be entirely filled with back arrows (←). If the +15 volts (MON) is down, there will be no raster or keyboard lights. If +12 volts is down, the screen will be entirely filled with transmit symbols. If -12 volts is down, the screen will be blank except for the cursor (press HOME to check).

c. Voltage Interrelationships. +15 volts (MON) is needed to power +5 volts. +5 volts, in turn, is needed for +15 volts (MEM). Therefore, if +15 volts (MEM) is down, check +5 volts. If +5 volts is present, +15 volts (MEM) is faulty. If +5 volts is down, check +15 volts (MON). If +15 volts is present, +5 volts is faulty and +15 volts (MEM) may also be faulty. If +15 volts (MON) is down, it is faulty and so also may be +5 volts and +15 volts (MEM), although usually only one voltage will be faulty at a time. +12 volts is independent of all other supplies. If -12 volts is down, it is faulty. The -5 volts is "zenered" off of -12 volts, thus, loss of the -5-volt supply is usually caused by failure of the -12-volt supply.

5.3.7.1.2 Locating the Cause. Once the faulty voltage(s) has been isolated, it should not be too difficult to determine the cause. First, visually inspect the five fuses in the fuse block. If in doubt about the condition of a fuse, check with an ohmmeter. If a fuse is blown, it may be because there is a short circuit in the load and the shutdown circuit did not function properly or there is some other short circuit to ground in the cable harness, loose wire, etc. Check for shorts (with power off) from the fuse terminal to ground with an ohmmeter.

a. Short Circuits. A quick way to determine if there is a short circuit in the load is to measure (power off) the resistance with an ohmmeter from the voltage test point to ground. Do this with both polarities of the ohmmeter. If the

readings obtained are substantially lower than those listed in table 5-10, the short has been located. Note: Measure resistances with regulator board in place, using an ohmmeter on RX1 scale.

Table 5-10. Supply Resistance Measurements

Power Supply (Test Point)	Negative Ground	Positive Ground
+5 volts	21 ohms	11 ohms
+15 volts (MEM)	300 ohms	300 ohms
+15 volts (MON)	26 ohms	5 ohms
+12 volts	1.8 K	600 ohms
-12 volts	600 ohms	1.2 K
-5 volts	600 ohms	1.2 K

Another way to check for an excessive load which causes the supply to shut down or fail to turn on due to under-voltage, is to insert an ammeter of proper value in place of the fuse (positive lead on lower terminal of fuse) and turn on power, observing the ammeter. If the current indication initially jumps up high (for that supply) then goes to zero, there is evidently a short circuit in the load and the protection circuit is functioning properly.

If a short circuit is evident, locating it can be difficult. If a short in the logic cage or cable harness cannot be visually located by removing logic boards one at a time, the VDT should be returned to the shop for additional troubleshooting.

b. Defective Components and/or Connections. If no short is apparent, the power fault was caused by a defective component or connection in the faulty power supply. To determine if the voltage regulators are functioning properly, use a voltmeter to check the voltage on the collector (emitter for -12 volts) of the series pass power transistor of the supply in question. Also, check the voltage at pin 3 of the MC1469R (pin 4 of the MC1463R) voltage regulator in question. These two voltages should approximate the voltages indicated on the schematic diagram. If no voltage is present, trace the circuit through, referring to the schematic diagram, to determine where the break in connection is located. If the proper voltages are present, check the voltage at pin 1 of the MC1469R (pin 5 of the MC1463R). If a voltage is present but there is no output, the fault is either a defective series

pass power transistor or an open connection. If no voltage is present, momentarily short pin 2 of the MC1469R or MC1463R to ground with a jumper lead. (This disables the undervoltage shutdown circuit and will allow the regulator to turn on.) If this procedure still does not produce a voltage at pin 1 of the MC1469R (pin 5 of the MC1463R), then the voltage regulator integrated circuit is defective. If clamping pin 2 to ground momentarily produces the proper voltage, but the voltage collapses as soon as the clamp is removed, then the 2N718 (2N3905 for -12 volts) transistor is defective.

5.3.7.2 Troubleshooting Keyboard. The keyboard can be checked either by substituting a keyboard which is known to be operating satisfactorily, or by using a multimeter to check the logic levels controlled by the individual keys, or buttons in question at the points indicated in table 5-11. If a keyboard extender is available, the checks can be made at the corresponding pins of keyboard receptacle J1. The "pull-ups" for all key lines are internal to the display unit, as indicated. Therefore, if any line is low or open when it should be high, check it again with the keyboard unplugged, and if it is still low or open, replace the board or boards it serves; if not, replace the keyboard.

Table 5-11. Keyboard Function Chart

Key/Button/Lamp	Test Points	Signal	Requirements
Any alphameric (A/N) character key, including "SP", CR, LF, RUBOUT, ESC, and the 12 separately grouped number keys.	J1-A1, A14-4 J1-A2, A14-54 J1-A3, A14-6 J1-A4, A14-5 J1-A5, A14-23 J1-A6, A14-7 J1-A7, A14-10	$\overline{\text{Bit 7/KB}}$ $\overline{\text{Bit 6/KB}}$ $\overline{\text{Bit 5/KB}}$ $\overline{\text{Bit 4/KB}}$ $\overline{\text{Bit 3/KB}}$ $\overline{\text{Bit 2/KB}}$ $\overline{\text{Bit 1/KB}}$	ASCII code bits for unshifted keytop characters. All high until a character key is pressed; then low (for 1's) and high (for 0's) in the character code combination until the key is released. Acceptable only if A/N STROBE is also present. (Pull-ups are on A14.)
	J1-A9, A14-25, A15-35	$\overline{\text{A/N STROBE FM KB}}$	High until any alphameric character key is pressed; then low until the character key is released. (Pull-up is on A15.)

Table 5-11. Keyboard Function Chart (continued)

Key/Button/Lamp	Test Points	Signal	Requirements
Function Keys:			Individual function enables. Each high until the corresponding key is pressed; then low until the key is released. Acceptable only if $\overline{\text{FCN STROBE}}$ is also present. (Pull-ups are on A15.)
RPT	J1-B4, A15-39	$\overline{\text{REPEAT FM KB}}$	
TAB	J1-C5, A15-28	$\overline{\text{TAB FM KB}}$	
CTRL	J1-B1, A15-38	$\overline{\text{CONTROL FM KB}}$	
SHIFT (either)	J1-B3, A15-23	$\overline{\text{SHIFT FM KB}}$	
XMIT	J1-B1, A15-54	$\overline{\text{XMIT FM KB}}$	
↑ (Cursor-up)	J1-B13, A15-57	$\overline{\text{Y-UP FM KB}}$	
PRINT	J1-A13, A15-53	$\overline{\text{PRINT FM KB}}$	
← (Cursor-left)	J1-B12, A15-21	$\overline{\text{X-LEFT FM KB}}$	
HOME	J1-B11, A15-22	$\overline{\text{HOME FM KB}}$	
→ (Cursor-right)	J1-A10, A14-26, A15-29	$\overline{\text{X-RIGHT FM KB}}$	
↓ (Cursor-down)	J1-A4, A15-58	$\overline{\text{Y-DOWN FM KB}}$	
CLR/FG	J1-A12, A15-10	$\overline{\text{CLEAR (PARTIAL) FM KB}}$	
I/C	J1-A8, A14-56, A15-9	$\overline{\text{I/C FM KB}}$	
D/C	J1-B6, A15-40	$\overline{\text{D/C FM KB}}$	
CLEAR	J1-A11, A15-59	$\overline{\text{CLEAR (FULL) FM KB}}$	
I/L	J1-B7, A15-48	$\overline{\text{I/L FM KB}}$	
D/L	J1-B10, A15-20	$\overline{\text{D/L FM KB}}$	
	J1-B2, A15-41	$\overline{\text{FCN STROBE FM KB}}$	High until any function key or keys are pressed; then low until the key or keys are released. (Pull-up is on A15.)
RECEIVE (button S1)	A13-7, A17-34	$\overline{\text{RECEIVE PB}}$	Individual pushbutton enables. Each high until the corresponding button is pressed; then low until the button is released. (Pull-up is on B15 for BREAK, and on A13 for LOCAL, RESET, and RECEIVE.)
LOCAL (button S4)	A13-3, A17-45, A18-16, 19	$\overline{\text{LOCAL PB}}$	
BREAK (button S5)	A17-49, B16-32, A18-23	$\overline{\text{BREAK PB}}$	
RESET (button S6)	A13-6, B12-24/25	$\overline{\text{RESET PB}}$	
PARITY ERROR (button S8)			

Table 5-11. Keyboard Function Chart (continued)

Key/Button/Lamp	Test Points	Signal	Requirements
RECEIVE (lamp L1) TRANSMIT (lamp L2) PRINT (lamp L3) LOCAL (lamp L4)	J2-j, B14-48 J2-k, B16-21 J2-n, B16-54 J2-4, B16-28	$\overline{\text{RECEIVE LAMP}}$ $\overline{\text{TRANSMIT LAMP}}$ $\overline{\text{PRINT LAMP}}$ $\overline{\text{LOCAL LAMP}}$	Individual lamp drive levels. Each low (to light) or high (to extinguish) the corresponding lamp.
(keyboard power ON/OFF switch)		+5 volts	Provides remote power control of ac relay in main power supply.

5.3.7.3 Troubleshooting Keyboard Input Logic. When the keyboard input logic is suspected (having previously replaced keyboard), refer to the subparagraphs that follow for troubleshooting techniques applicable to specific classifications of faults.

5.3.7.3.1 Wrong Character is Produced. When wrong characters are produced, errors can usually be associated with particular ASCII character code bits, and if so, considerable time can be saved by checking only the erroneous bits. First, key enough shifted and unshifted characters to assemble a list of at least two keys that produce wrong characters. Characters with complementary codes, such as "P" and "/," or "SP" and "^{S}O" (left arrow), or "A" and "^{S}." (greater than), etc., are particularly useful for this purpose, because only a few pairs need be keyed. Then use the ASCII character code chart (see Section 1) for a code comparison between characters keyed and those produced, with the aim of identifying one or two bits which could produce the observed characters if locked in the 1 or 0 state. If such bit errors are found, interchange A1 and A2 to see whether error follows the boards, and if so, replace A1 (bits 1 through 4) or A2 (bits 5, 6, and 7), as appropriate. If not, interchange A1 and A2 again and replace A5. If trouble is still present, remove replacement A5 and continue with paragraph 5.3.7.3.2 using the original A5.

5.3.7.3.2 All Character Keying is Abnormal.

Step 1. Extend A5 and, with mode switch at BATCH and LOCAL button lighted, use the multimeter to check ASCII character code bit inputs to "B" data register at

pins listed in table 5-12. First, check that all seven bits are low when no keys are pressed, while "CSP" is pressed, and that all are high while "CS/" is pressed. If so, continue by checking bits for each of the individual characters in question. The bits for each character should correspond to its ASCII character code: high for 1, and low for 0. If the bits are correct, proceed with step 2. If not, replace A14 or keyboard. Alternate possibility (for BIT 5 only): B15.

Table 5-12. Test Points for B Register Inputs

Character Code Bit	Test Point
BIT 7/KB	A5-26
BIT 6/KB	A5-47
BIT 5/KB	A5-34
BIT 4/KB	A5-20
BIT 3/KB	A5-22
BIT 2/KB	A5-11
BIT 1/KB	A5-9

Step 2. Use multimeter to check data bit outputs from "B" data register to memory at pins listed in table 5-13. First strike "SP" and check that all seven data bits are high; then strike "SO" (left-arrow) and check that all seven are low. If so, continue by checking data bits: low for 1 and high for 0. DATA BIT 6 TO MEM is stored high for 1, and low for 0. If all data bits are correct, proceed with step 3. If not (some data bits change correctly but others do not), replace A5. (If the changes for "SP" and/or "SO" are incorrect, check the ASCII character code bit inputs for these characters.) If none of the data bits changes, check again after repeating "SP", then after repeating "SO". If the data bits change only when the RPT key is used, replace A14 or A15. Alternate possibility: A6. If the data bits still do not change, replace A15, A16, or keyboard. Alternate possibility: A6.

Step 3. Strike "SO" and SCLEAR. Check that data bit outputs to memory at pins listed in table 5-13 are all high (corresponding to the "SP" character). If so, continue with step 4. If some data bits are low but others are high, replace A5. Alternate possibilities (for DATA BIT 6 TO MEM only): A13, A7, A11. If all seven

bits are low, check keyboard inputs for the CLEAR key as directed in paragraph 5.3.7.2, and if those inputs are correct, replace A15.

Table 5-13. Test Points for B Register Outputs

Character Code Bit	Test Point
$\overline{\text{DATA BIT 7 TO MEM}}$	A5-43
DATA BIT 6 TO MEM	A5-29
$\overline{\text{DATA BIT 5 TO MEM}}$	A5-27
$\overline{\text{DATA BIT 4 TO MEM}}$	A5-18
$\overline{\text{DATA BIT 3 TO MEM}}$	A5-16
$\overline{\text{DATA BIT 2 TO MEM}}$	A5-6
$\overline{\text{DATA BIT 1 TO MEM}}$	A5-3

Step 4. Use multimeter to check that DATA BIT 8 TO MEM at A5-44 is high. If level is correct, proceed with step 5. If not, replace A5.

Step 5. Use multimeter to check that FOREGROUND at A5-24 is high, and $\overline{\text{FOREGROUND}}$ at A5-23 is low. If levels are incorrect, interchange A1 with A2 and check again. If correct when A1 and A2 are interchanged, replace the board which was originally A2. Otherwise, restore A1 and A2 and replace A7 or A5. Alternate possibilities: A4 or A10.

Step 6. If all requirements of preceding steps have been met, reinstall A5 unless this procedure is to be followed by the character generator check of paragraph 5.3.7.5.

5.3.7.3.3 Incorrect Function Keying. Extend A15 with pin 60 ($\overline{\text{RESTORE KB + SYSTEM RESET}}$) taped off, and check keyboard input logic for any functions in question as directed in the appropriate step that follows. If the specified results are not obtained, check keyboard inputs for the key or keys involved as directed in paragraph 5.3.7.2 before replacing the suggested board or boards.

a. For CLEAR, check that A15-3, TOTAL CLEAR, is high after ^{S}CLEAR is keyed, and low after a character is keyed; and remains low when CLEAR is keyed without SHIFT. If not, replace A15 or A16.

b. For CLR/FG, check that A15-5, PARTIAL CLEAR, is high after SCLR/FG is keyed, and low after a character is keyed; and remains low when CLR/FG is keyed without SHIFT. If not, replace A15 or A16.

c. For PRINT, check that A15-13, PRINT, is high after SPRINT is keyed, and low after a character is keyed; and remains low after PRINT is keyed without SHIFT. If not, replace A15.

d. For XMIT, check that A15-42, XMIT, is high after SXMIT is keyed, and low after a character is keyed; and remains low after XMIT is keyed without SHIFT. If not, replace A15.

e. For SHIFT, check that A15-33, SHIFT/KB, is high while either SHIFT key is pressed, and low when both SHIFT keys are released. If not, replace A15.

f. For CR, check that A15-44, CR/KB ENABLE, is high while the CR key is pressed, and low when the key is released. If not, replace A14. If level is correct, check that A15-24, CR, A15-15, A/N, and A15-56 Y-DOWN + CR, are all high after CR is keyed, and that all are low after any function (such as HOME) is keyed. If not, replace A15 or A16.

g. For I/C, check that A15-14, EXPANSION, is high after a character has been keyed while the I/C key is pressed, and low after the character is keyed alone; and remains low when the I/C key is pressed alone. If not, replace A15.

h. For D/C, check that A15-45, COMPRESSION, is high after D/C has been keyed, and low after a character has been keyed. If not, replace A15.

i. For CTRL, key a character for a high A/N output at A15-15. Then press the CTRL key alone and check that A15-15 remains high. Also check that A15-7, CTRL/KB, is high while the CTRL key is pressed and low when the CTRL key is released. If not, replace A15. (The effect of CTRL/KB on BIT 7/KB at A5-26 (from A14-9) has been checked in paragraph 5.3.7.1.2.)

j. For RPT, check that A15-34, PROCESS REPEAT, is:

(1) Low when RPT key is pressed alone;

(2) High when it is pressed with a character key;

(3) High when it is pressed with each of the four arrow-top cursor control keys in turn; and

(4) Low when character key and each of the cursor control keys are pressed alone.

If any of these levels are incorrect, replace A15.

k. For I/L, check that A15-43, $\overline{I/L}$ is low after I/L has been keyed, and high after a character has been keyed. If not, replace A15.

l. For D/L, check that A15-16, D/L, is high after D/L has been keyed, and low after a character has been keyed. If not, replace A15, A16, or B15.

m. For HOME, check that A15-19, HOME, is high after HOME has been keyed, and low after a character has been keyed. If not, replace A15.

n. For cursor-left, check that A15-55, X-LEFT/KB, and B15-25, X-LEFT, are both high after cursor-left has been keyed, and low after a character has been keyed. If not, replace A15 or B15.

o. For cursor-right, check that A15-26, X-RIGHT, is high after cursor-right has been keyed, and low after a character has been keyed. If not, replace A15.

p. For cursor-up, check that A15-27, Y-UP, is high after cursor-up has been keyed, and low after a character has been keyed. If not, replace A15.

q. For cursor-down, check that A15-56, Y-DOWN + CR, is high after cursor-down has been keyed, and low after a character (other than CR) has been keyed, then high again after CR has been keyed, and finally low again after any other character has been keyed. If not, replace A15 or A16.

r. For TAB, check that A15-30, TAB, is high after TAB has been keyed, and low after a character has been keyed. If not, replace A15.

If all function key checks are normal, the trouble is not in the keyboard input logic. Remove the tape from A15-60 and reinstall A15.

5.3.7.4 Troubleshooting Display Refresh and Video Output Logic. When a synchronized sweep raster is present, but neither cursor nor characters can be obtained, and the display remains complete blank when A9-TP1 is temporarily grounded, then either the display refresh logic is not cycling, the video output logic is unable

to deliver video, or the TV monitor group is unable to display video. Proceed as follows:

Step 1. The skip refresh flip-flops on A11 can prevent the display refresh logic from cycling. Extend A11 and check that SKIP REFRESH CYCLE at A11-11 is low and that $\overline{\text{SKIP REFRESH CYCLE}}$ at A11-9 is high. If so, reinstall A11 and proceed with step 2. If levels are incorrect:

a. If A11-9 is open, replace A11.

b. If A11-9 is low, ground A11-5 (SET SKIP REFRESH FF) and strike a character key. If A11-9 is then low and A11-11 is high, replace A12 or A6.

c. If A11-9 is still low, remove ground from A11-5, tape off A11-9, and check that B9-56 is open. If so, replace A11. If not, the trouble is a short circuit on B8, B9 or B11. Remove the tape from A11-9 and replace those boards one at a time until A11-9 is high.

Step 2. Extend A7 with pin 28 of extender board taped off. Ground A7-53 (CURSOR OUTPUT) and check that sweep raster brightens momentarily then fades to blank. (A ground at A7-53 produces "foreground" brightness.) If so, cursor video can be delivered and TV monitor is able to display video; remove the ground from A7-53 and proceed with step 3. If not, continue as follows:

a. Remove the ground from A7-53 and check dc voltage at A7-53. If high, continue with item b. If low, replace B10; and if still low, replace A7.

b. Check MONITOR VIDEO at A7-60. The dc voltage should be more than 3 volts higher when A7-53 is grounded than when not grounded. If so, replace A8 or B9. If not, replace A7.

Step 3. Ground A7-29 ($\overline{\text{UNDERLINE GATE}}$) and check that sweep raster again brightens momentarily and then fades to blank. (A ground at A7-29 produces "background" brightness.) If so, proceed with step 4. If not, replace A7.

Step 4. Tape off three additional pins of the extender board: pin 56 ($\overline{\text{EOT FM MEM}}$), pin 47 ("A" DATA REG BIT 6), and pin 17 ("A" DATA REG BIT 7). Then ground A7-28 ($\overline{\text{VIDEO DATA}}$) and check that the display consists of the 27 solid character lines. If so, proceed with step 5. If not, ground A5-23 if the solid character lines are present but at background brightness. If the lines then brighten to foreground brightness, refer to paragraph 5.3.7.3.2, steps 3, 4 and 5. If the character lines are not present, or if they remain at background brightness when A5-23 is grounded, replace A7. Alternate possibility: A10. Remove all grounds and taped pins.

Step 5. If the above requirements have been met, video output logic is operating satisfactorily and TV monitor is able to display video. Remove the jumpers and tape from the extender board. If the trouble remains, replace boards in the following order: B9, A8, A10, B13, A5.

5.3.7.5 Troubleshooting Character Generator Logic. When the character generator logic is suspected, proceed to the steps that follow. This procedure assumes that there are, or have been, characters (and/or cursor) in the display, or that the video check of paragraph 5.3.7.4 has provided evidence that the display refresh logic is cycling, that the video output logic is capable of delivering video to TV monitor, and that the TV monitor is capable of displaying video.

Step 1. Remove A5 and insert an extender board in its place, but do not insert A5 into the extender board receptacle. Then proceed with steps 2 through 6, before checking the individual characters in question. For each character, ASCII character code bits are simulated by open circuits (for 1) or clip-lead grounds (for 0) at the extender board pin listed in table 5-14, using pin 1 or pin 31 as the ground point.

Table 5-14. Test Points for Simulating Character Codes

Character Code Bit	Extender Board Pin
"A" DATA REG BIT 7	42
"A" DATA REG BIT 6	30
"A" DATA REG BIT 5	28
"A" DATA REG BIT 4	19
"A" DATA REG BIT 3	17
"A" DATA REG BIT 2	7
"A" DATA REG BIT 1	4

Step 2. Check that, with A5 removed, the display is blank except for cursor, and that all seven bits listed in table 5-14 are open. If display is correct, proceed with step 3. If display is incorrect, proceed as follows:

a. If any of the seven bits is low, the trouble is a short circuit on one of the boards connected to that bit. Replace the corresponding boards one at

a time until the pin is high: for bit 7, replace A7, A12, or A16; for bit 6, replace A7, A9, A12, or A16; for any of the other five bits, replace A9, A12, or A16.

b. If the display is filled with any character other than the transmit symbol, replace A7 or A9.

c. If the display is filled with the transmit symbol, replace A12 or B8. Alternate possibility: B9.

Step 3. Ground bit 7 (extender board pin 41) and check that the display is filled with "?". If display is correct, proceed with step 4. If display is incorrect, proceed as follows:

a. If the display is filled with any character other than "?", replace A9.

b. If the display remains blank, leave extender board pin 42 grounded and ground extender board pin 23 ($\overline{FG}$). If the display is then filled with "?", replace A7; otherwise replace A9 or A7.

Step 4. With extender board pin 42 grounded, ground extender board pin 23 ($\overline{FG}$) and check that the display of "?" increases in intensity (background to foreground). If display is correct, remove both grounds and proceed with step 5. If display is incorrect, proceed as follows:

a. If the display goes blank when extender board pin 23 is grounded, replace A7.

b. If the intensity does not increase when extender board pin 23 is grounded, replace A7.

Step 5. Ground extender board pin 3 and check that the display is filled with "←" (left-arrow); then, with pin 30 grounded, ground each of the extender pins listed below one at a time and check that the display is filled with the corresponding character. If not, replace A9. If indications are correct, proceed to step 6.

Pin 28 for letter "O"

Pin 19 for "W"

Pin 17 for "[" (left-bracket)

Pin 7 for "]" (right-bracket)

Pin 4 for "↑" (up-arrow)

Step 6. Remove ground from extender board pin 3, ground all five pins listed in step 5, and check that the display is filled with the EOT symbol. If not, replace A12 or B8. Alternate possibilities: A7 or A9.

Step 7. For each character in question, other than those checked in steps 4, 5 and 6, simulate the corresponding ASCII character code bits by grounds (for each 0) at the pins listed in table 5-14. If any simulated character code fills the display with the wrong character, or with blanks, replace A9. If the display is filled with the proper character in every case, the trouble is not in the character generator logic; remove the grounding jumpers and either reinstall or replace A5.

5.4 ADJUSTMENTS

5.4.1 TV MONITOR ADJUSTMENTS (figure 5-2)

The TV monitor is adjusted at the factory for best overall video quality, and normally requires adjustment only when critical components or circuits are replaced. When the CRT is replaced, the magnetic deflection yoke, magnetic centering tabs, and shield must be realigned (see paragraph 5.5.4.2). Misalignment of these elements can cause the video display to be noticeably off-center, tilted, distorted, or out of focus.

As shown in figure 5-2, there are six adjustments on the TV monitor board located above the CRT. These adjustments control display focus and brightness, display size, vertical sync, and vertical linearity. To verify the correct settings of these adjustments, proceed as follows:

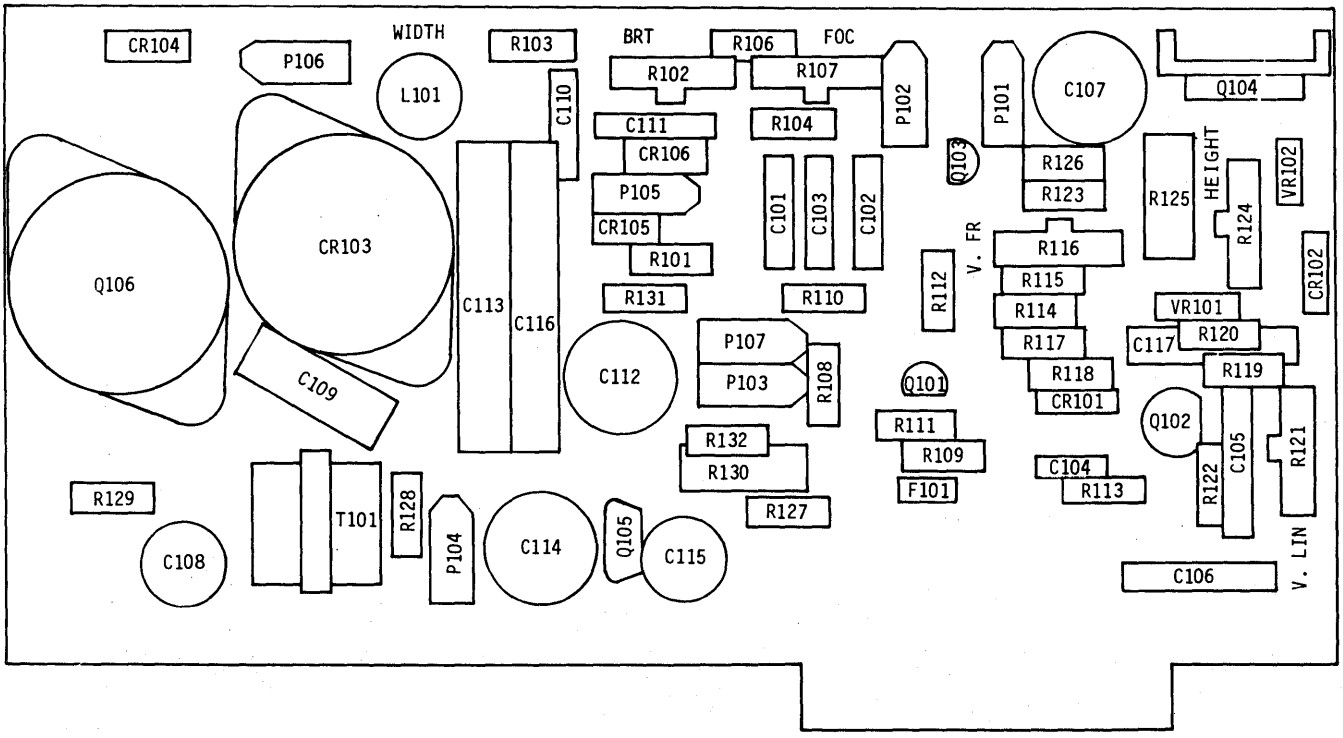


Figure 5-2. Monitor Control Board Adjustments

1. Turn on the terminal and allow three minutes for warm-up.
2. Strike ^{S}CLEAR to clear the display and home the cursor.
3. Enter a full screen of "#" symbols by simultaneously pressing and holding the REPEAT, TAB, SHIFT, and "#" keys.
4. Set the CONTRAST control on the front of the VDT to a mid-range setting. Then adjust BRT potentiometer R102 for best character contrast, such that the raster is just extinguished, and there is good foreground/background contrast without noticeable character blooming.
5. Adjust FOC control R107 for best overall focus of video data for the area between the center and upper left of the display.
6. Adjust WIDTH control L101 for a display width of 9 inches.
7. Adjust HEIGHT control R124 for a display height of 6 inches.
8. Adjust V LIN control R121 for best vertical linearity.
9. Connect a test jumper from B13TP8 (vertical drive) to ground.
10. Adjust V.FR control R116 until picture rolls up slowly.
11. Disconnect the jumper connection made in step 8. The raster should be locked in without any rolling effect. Repeat steps 7 and 8, and readjust if necessary.

5.4.2 POWER SUPPLY ADJUSTMENTS (figure 5-3)

The power supply voltages are adjusted at the factory, and normally require readjustment only when the power supply board, or a component on this board, is replaced. The eight potentiometer adjustments are shown in figure 5-3. Proceed as follows:

Step 1. Set POWER switch to ON, and allow a three-minute warm-up period.

Step 2. Referring to table 5-15, check each of the power supply voltages with a digital voltmeter, and adjust the corresponding voltage potentiometer if voltage is out of tolerance.

Step 3. Connect the digital voltmeter to the +5 Logic test point, and turn current limiting potentiometer R22 counterclockwise until a slight dip in the voltage occurs. Then turn the potentiometer clockwise until the voltage stabilizes, and continue turning clockwise one full turn from this point. Recheck that +5 Logic voltage is still within the tolerance given in table 5-15, and readjust R58 if necessary.

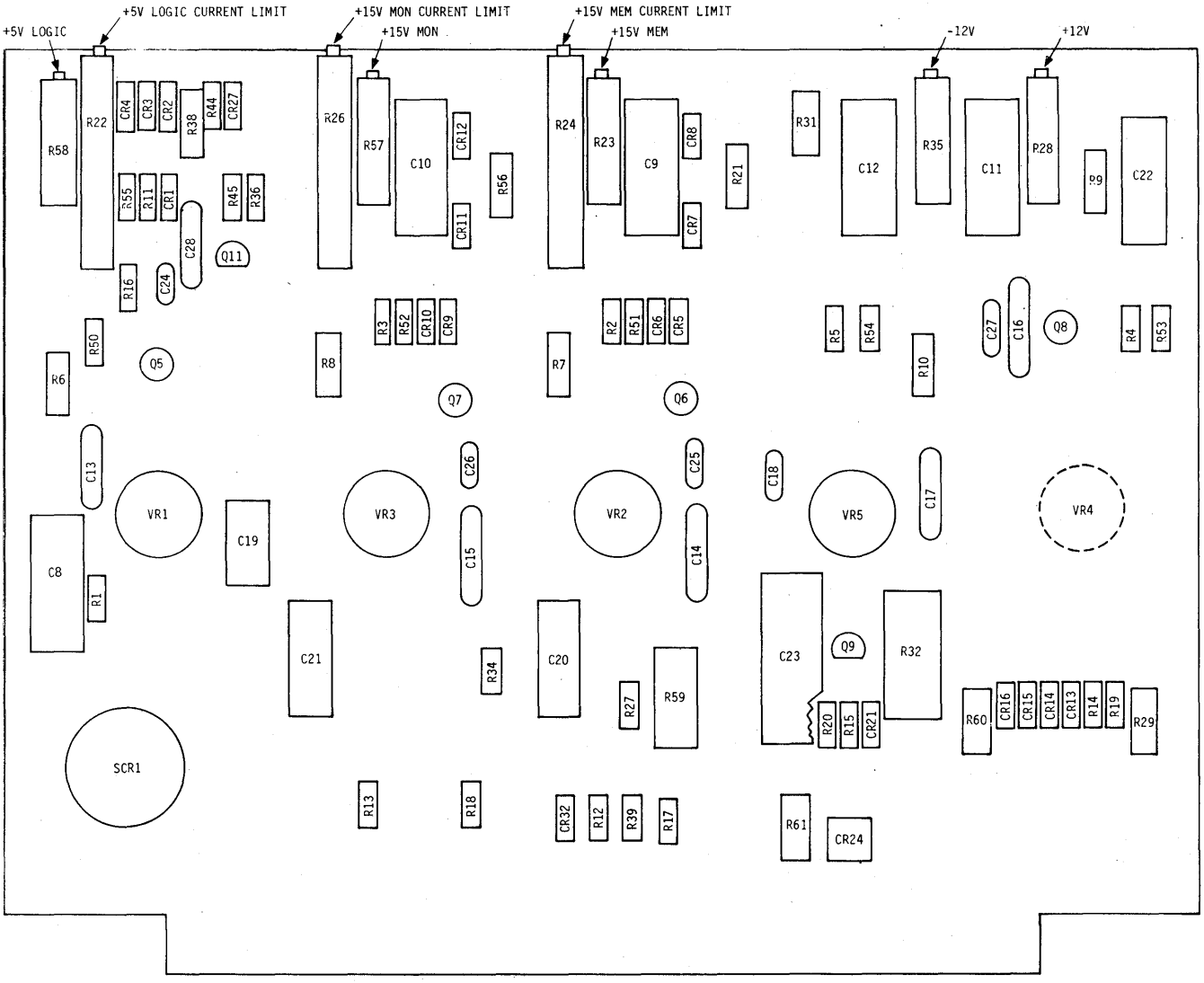


Figure 5-3. Power Supply Adjustments

Table 5-15. Power Supply Voltage Adjustments

Test Point	Voltage	Voltage Adjustment
+5 Logic	+ 5.0 ±0.25	R58
+15 MEM	+13.0 ±0.67	R23
+15 MON	+15.0 ±0.75	R57
+12 Volts	+12.6 ±1.26	R28
-12 Volts	-12.6 ±1.26	R35
- 5 Volts	- 5.0 ±0.25	None (adjust with R35)

Step 4. To set the current limits for the +15 MEM and +15 MON supplies, refer to table 5-16, and proceed as follows:

a. Set POWER switch to OFF, and remove the fuse indicated in table 5-16 for the supply being adjusted. On the power supply board, ground the collector (case) of the shutdown transistor indicated in table 5-16.

b. With the meter set to the 10-amp scale, connect the + lead of the meter to the top of the fuse holder, and leave the other lead disconnected. Also, connect a lead to the output test point for the supply being adjusted, and leave the other end of this lead disconnected. Be sure that the ends of the two unconnected leads are insulated, and that neither lead is making contact with the unit.

c. Set POWER switch to ON. Ground the lead connected to the output test point, and, at the same instant, connect the loose multimeter lead to the other side of the fuse holder. DO NOT GROUND THE OUTPUT TEST POINT FOR MORE THAN 5 SECONDS. While the output test point is grounded, read the current (in amperes) on the meter.

d. If current read in step c is not the value indicated in table 5-16, shift the setting of the current limiting pot slightly, and repeat step c. Continue this process until the correct current value is obtained. When correct current reading is obtained, turn power off, replace the fuse, and disconnect all test leads.

Table 5-16. Current Limiting Adjustments

Voltage	Shutdown Transistor	Fuse	Current Limiting Potentiometer	Current Limit (amps)
+15 MEM	Q6	F2	R24	5.6
+15 MON	Q7	F3	R26	2.6

5.4.3 MEMORY THRESHOLD ADJUSTMENT

Memory threshold adjustment A4R4 is adjusted at the factory, and should normally not require readjustment except when board A4 is replaced, or when directed in the checkout and troubleshooting procedures. For this procedure, a digital voltmeter is required, since the memory threshold voltage must be set to a tolerance of ±0.05 volt. To perform the adjustment, proceed as follows:

Step 1. Set POWER switch to ON, and set MODE switch to BATCH. Allow a three-minute warm-up period.

Step 2. Without extending board A1, connect the digital voltmeter between pin XA1-60 (+) and XA1-30 (-).

NOTE

Do not permit the memory threshold voltage to swing outside the range from +1.4 volts to +2.2 volts during the steps that follow.

Step 3. Momentarily press the TAB, REPEAT, and "SO" keys as a group to write "↵" into every memory location. If the memory threshold voltage is satisfactory for "↵", every "↵" will be retained in memory and the display will be filled with "↵". If not, the "↵" in some or all locations will be dropped, and different characters or blanks will be displayed at the corresponding positions. Adjust A4R4 for a reading of 1.85 ±0.05vdc on the digital voltmeter.

Step 4. Press HOME and repeatedly strike I/L until three lines of data remain, noting that no characters change during the line insertion. Then repeatedly strike D/L until three lines of data remain at top of display, noting no characters have changed.

Step 5. Repeat steps 3 and 4 for characters "J", "K", "L", "N", "O", "Q", and "W". Adjust potentiometer on A4R4 for a reading within the threshold range given in step 3 such that all characters are displayed correctly.

5.5 REMOVAL AND REPLACEMENT (table 5-17)

The subparagraphs that follow contain removal and replacement procedures for items where the procedure is not obvious. The removal and replacement procedures are listed in table 5-17.

Table 5-17. Removal and Replacement Procedures

Assembly or Part	Paragraph No.
Display Unit Housing	5.5.1
Logic Rack	5.5.2
TV Monitor Cage/Fan Assembly	5.5.3
CRT and Shield	5.5.4
CRT Deflection and Control Board (Monitor Board)	5.5.5
Keyboard	5.5.6

WARNING

All removal and replacement actions should be performed with the POWER switch set to OFF, and with the power plug disconnected.

5.5.1 DISPLAY UNIT HOUSING

5.5.1.1 Removal. The housing is attached to the chassis by eight screws along the bottom edges of left and right sides only. To remove the housing, remove and retain the four screws on each side, close the control cover and then carefully lift the housing straight upward high enough to clear the tops of the CRT mask, monitor cage/fan assembly and logic rack.

5.5.1.2 Installation. At the rear of the display unit, make certain that the power cord and modem cable are clear of the chassis and that the internal cabling is pushed inward far enough so that it will not be caught by the edges of the housing. Then install the housing as follows:

Step 1. Raise and center the housing above the chassis and, with the control cover closed, begin lowering the housing into position. While lowering, guide the housing to prevent its front edge or the control cover hardware from scratching or catching on the CRT mask; and in the final few inches make certain that the control cover hardware and rear housing cutout are properly aligned with the front control panel and the rear cabling panel, respectively.

Step 2. Align screw holes along bottom left and right edges of housing with corresponding tapped holes in chassis, and secure the housing with the eight screws retained during removal.

5.5.2 LOGIC RACK

The logic rack is not a replaceable assembly, and removal should not be attempted. The rack-to-chassis wiring is cabled vertically along the inner rear edge so that rack can be pivoted outward along that edge for access to the pins and wiring of board receptacles, but such access is not usually required during servicing. When a board pin must be probed, the recommended method is to extend the board and probe pin on the extender board. All board test points are accessible at the exposed side of the rack, and the boards are removed and inserted from that side.

5.5.2.1 Board Removal. The boards in the rack are supported along their sides in guide slots formed by the edges of the ventilating holes in the vertical rack members and are secured in the rack by receptacle contact pressure. To remove any board other than the memory sandwich (covered in paragraph 5.5.2.4), always use a board extractor or a hook-type board puller for the initial pull necessary to free the board from its receptacle. Never pull a board by its test points or with any finger or tool grip on its conductor pattern or parts. Also, when using a hook-type puller, insert the hook or hooks through the bottom side so that the board will not fall off when free, and take care to limit the pulling distance to avoid "throwing" the board.

5.5.2.2 Board Insertion. Proper circuit board orientation is component-side-up in the front half of the rack (the "A" row) and solder-side-up in the rear half ("B" row). This orientation is enforced by an off-center keying blade in each receptacle and a corresponding keying notch in the pin edge of each board. However, the off-centering is the same for all boards, and there is no provision against inadvertent installation of boards in the wrong positions. Therefore, before installing a board, always make certain that it is the proper board for the intended position. To install the board, simply position the board in its guide slots with the proper side up, slide it straight inward, and then press it fully into its receptacle.

5.5.2.3 Use of Extender Boards. The recommended extender boards have the same orientation keying provisions as the circuit boards, and are installed in the same way. In this case, the orientation is enforced so that the test point numbering on the extender boards (low numbers 1 to 30 on one side, high numbers 31 to 60 on the other) will always match the pin numbering of the circuits. Proper orientation is low-numbers-up in the "A" row and high-numbers-up in the "B" row. Insert the circuit board into the extender board receptacle only after (not before) installing the extender board. Any attempt to install the circuit board and extender board as an assembly can bend the circuit board and/or tilt it in the extender board receptacle, thus damaging either the circuit board or the extender board receptacle.

5.5.2.4 Replacement of Memory Core Boards. The memory sandwich formed by the permanent assembly of memory core boards B3 and B4 will not operate normally when extended and should not be removed for any purpose other than replacement. When replacement is necessary, the pulling forces during removal, and the insertion

pressures during installation, must be substantially equal on both boards to avoid distorting and damaging the assembly. The boards have pulling holes in their outer corners, but it is difficult to apply equal pulling forces at all four holes unless a special four-hook puller is hand-made for the purpose. A safe alternate method is to work the assembly free from both receptacles by hand: first remove boards from above and below the assembly to gain hand clearance; then grasp the assembly by hand, applying pressure only on and around the assembly screw-heads (internal board spacers), and pull outward and alternately side-to-side until both boards are free from their receptacles. When installing the replacement assembly, press equally on both boards to insert them fully into their receptacles.

5.5.2.5 Pivoting the Logic Rack. The logic rack is secured to the chassis by four screws -- two through each end of its bottom plate -- and is spaced from the adjacent side of the monitor compartment by a bumper screw near the top. The rack-to-chassis wiring is cabled vertically along the inner rear edge of the rack, and the rack can be pivoted outward on the mounting screw nearest to the cabled wiring. However, this should not be attempted unless it is absolutely necessary to gain access to the receptacle pins and wiring, because when secured by only a single screw (loosened for pivoting), the rack is vulnerable to damage from carelessly applied probing or board pulling and insertion forces. The procedure is as follows:

1. Remove the outer rear mounting screw, and loosen the inner rear screw slightly (a quarter to half a turn) to serve as a pivot.

2. Position the unit so that the rack will not overhang the edge of the desk or work table when pivoted outward.

3. Loosen two top bracket screws, but do not remove.

4. At front of rack, remove the two front mounting screws and swing the rack outward only far enough for the desired access. Block up the front end of the bottom plate to support the overhanging weight. Note that the logic rack near the connector area is not protected if it is swung too far inward, unless two top bracket screws are secure.

5. As soon as access to the receptacle pins and wiring is no longer necessary, slowly pivot the rack back into position and install, but do not fully tighten, the two front mounting screws. The two top rack brackets must first be aligned.

6. Align the rear end of the rack and the bracket assembly, and install the outer rear mounting screw. Then tighten all four mounting screws, plus two top bracket screws.

5.5.3 TV MONITOR CAGE/FAN ASSEMBLY

The TV monitor cage including the fan housing, can be removed from the system base plate and the logic rack (spacer brackets) fairly easily. The cage is flexible enough so that its feet can be spread slightly to facilitate lifting the cage up and over the power supply.

5.5.3.1 Removal Procedure.

1. With power shut down, remove ac plug from outlet. Disconnect two fan wires from fan lugs. Note that one fan wire is "hot" when the ac plug is plugged in.

2. Loosen two logic rack bracket screws at top. Loosen logic rack mounting screws (four). Remove two front and right rear screws only. Swing logic rack outward only so far that right-hand TV monitor cage mounting screws are accessible.

3. Remove six cage mounting screws.

4. Remove 10-pin edge connector from top monitor PC board.

5. Carefully grasp bottom portion of cage at left and right sides, pulling up and slightly outward so that cage will clear power supply harness and components. Do not place cage (including CRT) on any surface with the CRT facing down. The tube surface can be easily scratched or damaged.

5.5.3.2 Installation Procedure.

1. Reinstall monitor cage/fan assembly by carefully lowering it (with slight outward pull) over baseplate power supply harness, etc. Position cage feet so that screw holes are lined up with base plate. Install six mounting screws, align two top logic rack brackets, and install logic rack screws (three). Tighten all screws when aligned.

2. Reconnect two fan wires to fan lugs. Install 10-pin edge connector to monitor PC board.

3. Install ac power plug.

5.5.4 CRT AND SHIELD

The CRT or shield is removed and installed from the front of the display unit. The CRT is supported and secured at the front by four screws, one at each corner of its face flange, accessible when the CRT mask is removed. The deflection yoke is clamped in position on the neck of the CRT and connected to the deflection and control board by separate horizontal and vertical plug-in connections. The

Mu-metal magnetic shield is supported and padded against the bell of the CRT by four foam-polyurethane padding strips, and is held forward against the bell by the combination of two tension springs hooked on the side-plates of the monitor compartment and forward pressure from the fan motor support at the rear. Note exactly how yoke plugs are mounted and yoke is rotated. In some units the yoke plug pins have been reversed and yoke rotated for best display linearity/focus.

5.5.4.1 Removal.

1. Unplug the keyboard, remove its cable from underneath the display unit, and lay it aside. Then move the display unit back or turn it to one side so that the CRT can be placed face down immediately in front of the display unit. Disconnect CRT connector at rear of CRT tube and two sets of yoke plugs from monitor PC board. Be sure that these yoke wires are loosened from CRT harness.

2. Loosen or remove and retain the four screws which secure the CRT mask at its left and right sides, two screws on each side. Then pull the mask straight off and place it face down in front of the display unit as a cradle for the CRT after step 6, below.

3. Unclip the high-voltage wire from the CRT anode terminal and momentarily ground the terminal.

4. Unhook and retain the two Mu-metal shield tension springs. (Note how they are hooked to cage.)

5. Remove and retain two screws from diagonally opposite corners of the CRT face flange. (In early production models, note that two vertical support brackets under CRT have to be loosened.)

6. Hold up the weight of the CRT by the bottom edge of the face flange and remove and retain the two screws from the other two corners of the flange. Then, handling the CRT only by its face flange and bell, carefully withdraw it from the display unit: straight outward at first, until the neck and socket are clear of the Mu-metal shield; then downward, tilting it to an angle which maintains slack in deflection yoke wiring, until the bottom edge of the face flange is resting in the previously positioned mask.

7. Keeping the CRT at an angle which provides wiring slack, loosen yoke wires for best slack. Then stand the CRT face down in the mask and move both aside to a safe area.

8. Remove the Mu-metal shield from the top of the power supply assembly and lay it aside.

9. Note the position of the deflection yoke with respect to the sides of the CRT bell for reference during installation. Then loosen the deflection yoke clamp screw and slide the yoke off the neck of the CRT. Note exact position of metallized sleeve under yoke which is required for horizontal linearity control.

5.5.4.2 Installation and Alignment. All steps of the following procedure are to be performed when the CRT or the deflection yoke is being replaced. However, if the CRT has been removed merely for the removal of the shield and adjustment, replacement of yoke, perform only steps 1, 3, 11, 12, 13 and 14.

1. Stand the CRT face down in the CRT mask immediately in front of the display unit, with the anode terminal facing away from the display unit.

2. Slide the deflection yoke on the neck of the CRT down against the bell, rotating it to the position noted during removal (with "top" on the same side as the anode terminal). Then lightly tighten the clamp screw: enough to prevent radial motion of the yoke, but not enough to prevent manual rotation. The final clamped position will be determined during the tilt and centering adjustments of step 7. Check that metallized horizontal linearity sleeve is in proper position, as during removal.

3. Handling the CRT only by its face flange and bell, tilt it toward the display unit to an angle at which the socket can be installed and the deflection yoke plugs can be guided toward the bottom rear of the monitor board. For the deflection yoke, insert vertical connector P102 (green and yellow wires) and horizontal connector P106 (red and blue wires) into the correspondingly labeled receptacles on the deflection and control board, with each plug "pointed" in the direction stencilled around its receptacle unless plugs were previously installed backward as described in paragraph 5.5.4.

NOTE

The Mu-metal CRT shield is not installed until after the tilt and centering of the display have been checked and any necessary corrections made.

4. Lift the CRT by its face flange and carefully position it in the display unit. Align the holes at the corners of the face flange, and loosely install (at two diagonally opposite corners) two of the screws retained during removal.

5. Clip the high-voltage wire into the anode terminal and connect the CRT connector.

6. Plug in ac plug and keyboard, set mode switch to BATCH, and set the POWER switch to ON. Then, when the CRT has warmed up enough to produce a display, fill the display with a visible character and check the tilt and centering of the display. To fill the display, press tha TAB and RPT keys together with a character key. Allow CRT to warm up for at least one minute.

7. First correct the off-centering, if any, by rotating one or both of the two centering magnets in the deflection yoke, disregarding any additional tilt that may result. Then correct the tilt, if any, by rotating the deflection yoke, keeping it firmly against the CRT bell. Repeat the centering and

tilt adjustments alternately until the display is tilt-free, centered within ±1/8-inch from side to side, and centered within ±1/8-inch from top to bottom. Then tighten the deflection yoke clamp screw enough to prevent rotation.

8. Adjust the CRT focus and vertical frequency (V.FR), as directed in paragraph 5.4.1. If these adjustments affect the apparent centering of the display, loosen the deflection yoke clamp screw and repeat the centering and tilt corrections of step 7. Then set the POWER switch to OFF.

9. Unclip the high-voltage wire from the CRT anode terminal, and momentarily ground the anode terminal.

10. Hold up the weight of the CRT by the bottom edge of the face flange and remove the two corner screws loosely installed in step 4. Then, handling the CRT only by its face flange and bell, withdraw and lower it to rest in the mask, holding it at an angle which does not strain the socket and deflection yoke wiring. Guide the yoke wires through the shield toward the rear.

11. Position the Mu-metal CRT shield on the top of the logic power supply assembly, funnel end forward, with the four foam-polyurethane padding strips at the top, bottom, left, and right, to contact the corresponding surfaces of the CRT bell.

12. Lift the CRT by its face flange, align the neck and socket to enter the Mu-metal shield without touching, and start the CRT into position, watching to make certain that the padded funnel end of the Mu-metal shield contacts the bell of the CRT properly before pressing the CRT face flange into position. Align the holes at the four corners of the face flange and install the four screws retained during removal. For early production models fitted with vertical CRT support brackets and spacers, align brackets during CRT installation, and tighten all screws securely.

13. Clip the high-voltage wire into the CRT anode terminal, and hook in the two Mu-metal shield tension springs retained during removal. Install two yoke plugs onto PC board as required and harness all CRT wires to avoid fan blades.

14. Position the CRT mask on its mounting brackets, hold the mask firmly against the face of the CRT, and install and tighten the four attaching screws retained during removal.

5.5.5 CRT DEFLECTION AND CONTROL BOARD

The CRT deflection and control board (monitor board) is secured via four nylon pins and is interconnected by a set of molex plugs.

5.5.5.1 Removal.

1. Disconnect all molex plugs from PC board. Record all plug numbers and color codes.
2. Disconnect 10-pin edge connector from board.
3. Squeèze board up and over four nylon mounting plugs.
4. Remove CRT connector.

5.5.5.2 Installation.

1. Place PC board over four nylon pins with connector edge facing rear. Press down and pop in. Connect CRT connector.
2. Connect all plugs to board per schematic.
3. Connect 10-pin edge connector (check key).

5.5.6 KEYBOARD

The top housing (cover) hooks under the bottom housing at the front and is secured by four screws at the rear. When these screws are removed, the top housing can be tilted up slightly at the rear, pulled forward to unhook at the front, and then lifted off. The keyboard subassembly is secured by screws and must be aligned so that none of the keys will jam in any of the top housing cutouts after the top housing is installed and its three attaching screws fully tightened.

5.6 CLEANING AND CHECKING CONNECTIONS

The VDT is no more difficult to care for than a conventional electric typewriter. Keep it clean, keep its connectors plugged in, and keep its immediate area clear of stacked papers, books, and other items that might interfere with free air circulation.

5.6.1 CLEANING

Wipe or brush off dust, lint, smudges, etc., whenever they become noticeable, being careful not to wipe or brush them into the keyboard through the spaces between and around the keys. For wiping, use a soft clean dust cloth or a facial tissue, and be careful not to scratch the display face plate by hard rubbing. For stubborn

accumulations, dampen the cloth or tissue slightly, but only with plain water or a cleaning fluid suitable (safe) for use on plastic and painted surfaces, such as one of the commercially available anti-static cleaning fluids. The cloth or tissue should not be wet, just damp, so that it will pick up dust or dirt.

The occasional light cleaning described above will not always prevent the gradual accumulation of a coating of varnish-like dirt, particularly on the keytops, and a more thorough cleaning will usually be necessary about once a month. For this cleaning, wipe all surfaces with the anti-static cleaning fluid, and use a succession of soft cloths or facial tissues. As each cloth or tissue becomes dirty, discard it and continue with a clean one. Use enough fluid to moisten each cloth or tissue thoroughly, so that very little wiping pressure will be needed, but not enough to allow dripping under the wiping pressure. After cleaning each surface, wipe off the fluid and dissolved dirt with a dry cloth or tissue. Pay particular attention to the display unit face plate, where any residual dirt film will later be visible in front of the display, and to the keytops on the keyboard, which receive the largest deposits of dirt.

5.6.2 CHECKING CONNECTIONS

The performance of the unit depends on four plug-and-socket connections (more than four if the optional accessories are provided) as follows:

a. The keyboard cable plug in its socket on the rear of the display unit;
b. The modem cable plug in the corresponding socket on the modem;
c. The power cord plug in a grounded wall outlet;
d. The modem power cord plug in a wall outlet; and
e. The plugs on the ends of the optional accessory cable, if provided - one plug in the accessory socket on the rear of the display unit, and the others in the sockets on the hard copier (if provided) and the tape cassette unit (if provided).

These connections should be checked whenever the unit seems to be operating abnormally, whenever it has been moved for cleaning, and whenever it has been moved and reinstalled in a new location. Make certain that each plug is fully inserted straight into its socket. Also make certain that the cables are not caught under the feet or edges of any units and that they are draped neatly, with no tangles, sharp bends, or kinks that might strain their internal wires or outer insulation.