

## INDEX

|  | <u>Page No.</u> |   |    |
|--|-----------------|---|----|
| <b><u>INTRODUCTION AND OPERATION – SECTION I</u></b>                     |                 |   |    |
| Communication Options . . . . .  | 18              | Push by One Replace – Cursor Over ES . . . . .  | 28 |
| Communication Procedures . . . . .                                       | 15              | Push by One Replace –(One Memory Position Character<br>Replaced by a Two Memory Position Character) . . . . . | 31 |
| Functional Description . . . . .   | 1               | Push by Two Replace – Cursor Over ES . . . . .  | 34 |
| General Description . . . . .  | 1               | Put and Get . . . . .   | 24 |
| Glossary of Terms . . . . .  | 20              | Reposition Cursor . . . . .   | 93 |
| Operation . . . . .  | 3               | Rope Character Generator . . . . .  | 10 |
| Physical Characteristics . . . . .                                       | 1               | Rope Timing . . . . .   | 16 |
| System Interface . . . . .   | 19              | Select . . . . .  | 66 |
|  |                 | Selective Send . . . . .  | 82 |
|  |                 | Sequential Select . . . . .   | 89 |
|  |                 | Single Variable Tab Option . . . . .  | 97 |
|  |                 | Synchronous Option . . . . .  | 98 |
|  |                 | Tab Delete . . . . .  | 61 |
|  |                 | Tab Insert . . . . .  | 61 |
|  |                 | VAR Tab . . . . .   | 58 |
|  |                 | Variable Tab Operation . . . . .  | 57 |
|  |                 | VTAB Set . . . . .  | 57 |
| <b><u>FUNCTIONAL DETAIL – SECTION II</u></b>                             |                 | <b><u>CIRCUIT DETAIL – SECTION III</u></b>  |    |
| Backline . . . . .   | 50              | Deflection Amplifier . . . . .  | 23 |
| Backspace . . . . .  | 53              | Digital-to-Analog Section . . . . .   | 20 |
| Basic Machine Timing . . . . .   | 3               | Logical Elements . . . . .  | 1  |
| Broadcast Select . . . . .   | 84              | Memory Section . . . . .  | 5  |
| Character Delete (One Memory Position Character) . . . . .               | 45              | Symbol Generator Section . . . . .  | 14 |
| Character Generation . . . . .   | 9               |   |    |
| Character Insert . . . . .   | 50              | <b><u>ADJUSTMENTS – SECTION IV</u></b>  |    |
| Clear . . . . .  | 54              | Control Unit Adjustments . . . . .  | 1  |
| Contention Mode . . . . .  | 93              | Monitor Adjustments . . . . .   | 6  |
| Control Storage Register . . . . .                                       | 17              | <b><u>MAINTENANCE PROCEDURES – SECTION V</u></b>  |    |
| Core Memory Addressing . . . . .   | 6               | Fault Isolation Procedure . . . . .   | 23 |
| Cursor Display . . . . .   | 15              | <b><u>INSTALLATION PROCEDURES – SECTION VI</u></b>  |    |
| Display Sequence & Rope Timing . . . . .                                 | 12              | Interunit Cabling . . . . .   | 1  |
| Edit & Processor Memory Sequences . . . . .                              | 25              | Special Wiring and Cabling . . . . .  | 5  |
| Edit Sequence (One Position) . . . . .                                   | 25              | <b><u>RELIABILITY IMPROVEMENT NOTICES – SECTION VII</u></b>   |    |
| Edit Sequence (Two Position) . . . . .                                   | 27              | Index – RIN/EI-EWI-ECN . . . . .  |    |
| Erase . . . . .  | 63              | Introduction . . . . .  |    |
| Fast Select . . . . .  | 84              | RIN File . . . . .  |    |
| Fixed Tab . . . . .  | 55              | Unit Travel Log . . . . .   |    |
| Flyback (Retrace) . . . . .  | 16              | <b><u>OPTIONAL FEATURES – SECTION VIII</u></b>  |    |
| Forms . . . . .  | 63              | AIC Flip/Flops . . . . .  | 3  |
| Forward Space . . . . .  | 63              | Functional Description . . . . .  | 1  |
| Forward Space – Shift . . . . .  | 63              | Functional Detail . . . . .   | 2  |
| Frame Send . . . . .   | 80              | General . . . . .   | 1  |
| General Description of B9353 System . . . . .                            | 1               | Glossary of Signal Names . . . . .  | 3  |
| Group Select . . . . .   | 88              |   |    |
| Home . . . . .   | 53              |   |    |
| Keyboard . . . . .   | 17              |   |    |
| Line Delete . . . . .  | 48              |   |    |
| Line Insert . . . . .  | 50              |   |    |
| Memory Timing & Operation . . . . .                                      | 9               |   |    |
| New Line . . . . .   | 61              |   |    |
| Point to Point Receive . . . . .   | 92              |   |    |
| Point to Point Transmit . . . . .  | 93              |   |    |
| Polling . . . . .  | 75              |   |    |
| Power on Clear . . . . .   | 54              |   |    |
| Pull by One Replace . . . . .  | 38              |   |    |
| Pull by 2 Replace Character Delete (2 Memory<br>Position Char) . . . . . | 47              |   |    |

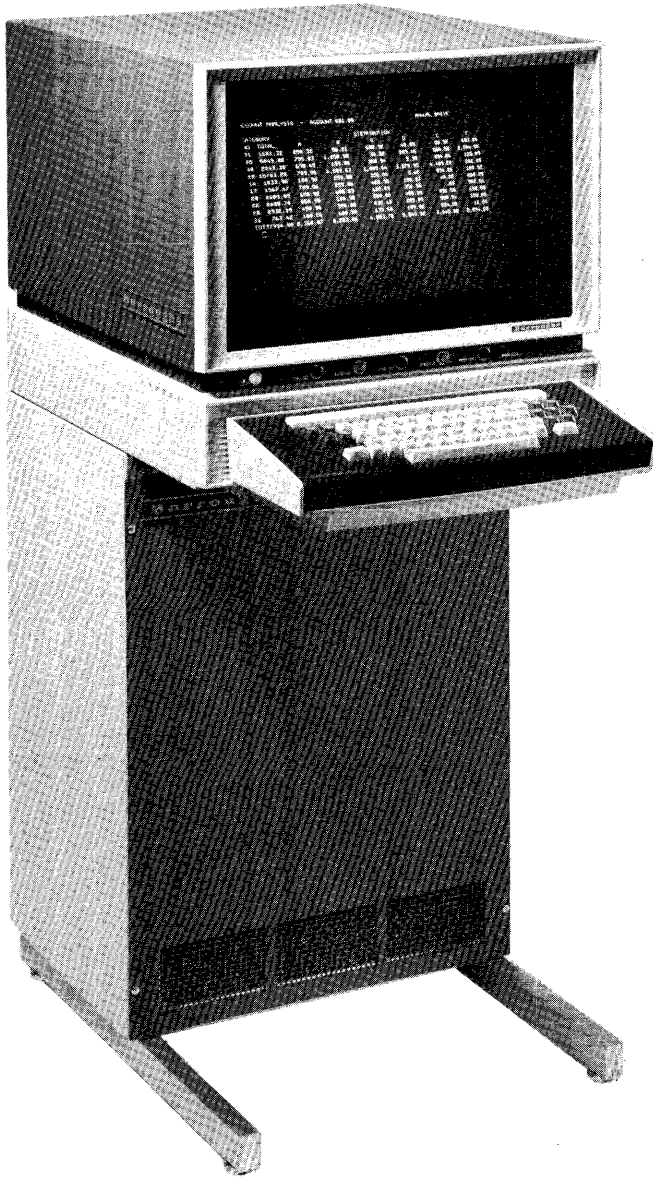
Introduction and OperationGENERAL DESCRIPTION

Fig. I-1

The Burroughs Input and Display Set incorporates a keyboard for data entry and a cathode ray tube (CRT) with a character generator for data display. The display has a repertoire of 66 alphanumeric and special symbols, plus space and non-destructive cursor. The content of the internal storage is sampled, converted to stroke code, and displayed 50 to 60 times per second to produce a display presentation with 2000 character positions arranged as 25 lines of 80 characters each. The memory may be partitioned and the associated logic time-shared by up to four separate, functionally independent CRT displays, each with

an independent keyboard. Figure I-1 illustrates a single monitor in the "Free Standing" configuration.

The data content of the internal storage is generated or altered by a keyboard or incoming message or both. All or part of the content of storage may be transmitted as an outgoing message upon initiation by a keyboard. A multi-wire direct system interface is standard. This interface is also compatible with several Bell System data sets utilizing either 2-wire or 4-wire private lines, or DATA-phone® for communications circuit or with a 2-wire direct circuit.

DATA-phone service is a data communication transmission service supplied by the Bell System. This service utilizes the switched telephone network and includes the use of the following features of this network:

DDD(Direct Distance Dialing)  
 WATS (Wide Area Telephone Service)  
 PBX (Private Branch Exchange)  
 Automatic alternate routing.

PHYSICAL CHARACTERISTICSMONITOR UNIT

The monitor unit houses the 17-inch CRT, all CRT power, CRT driving circuits and CRT controls and status indicators. The brightness control, the only CRT control accessible to the operator, is located on the lower right side. An AC power switch is located on the left front of the monitor.

KEYBOARD UNIT

The keyboard is designed to comply with the touch and geometry common to conventional office typewriters. The keyboard is electrically interlocked so that no more than one key is operative at a time. The keyboard unit also includes control keys above the keyboard proper. The slope top case is designed to be either mounted with the control unit and monitor unit for the free-standing composite assembly or as a separate assembly. An option can be provided to permit remote keyboard operation up to 10 feet from the monitor.

CONTROL UNIT

The control unit houses the logic circuits, memory and power supplies. The logic circuits are separated into replaceable circuit cards. An AC power switch is located on the top of the control unit. The unit is intended to rest on the floor.

FUNCTIONAL DESCRIPTION

This system contains a magnetic core memory with 1024 six-bit bytes. The entire memory may be associated with a single monitor (CRT) and keyboard or the memory may be divided into four equal sections with each section

® DATA-phone is registered service mark of AT&T Co.

Introduction and Operation

TABLE I-1 PHYSICAL CHARACTERISTICS

|              |                                  |  |   |                   |
|--------------|----------------------------------|--|---|-------------------|
| Power:       | (Control Unit)                   | 105 to 130 volts AC, 60 Hz, 400 watts  |   |                   |
|              | (Monitor)                        | 105 to 130 volts AC, 60 Hz, 300 watts  |   |                   |
| Weight:      |                                  | Monitor Unit – 100 pounds  |   |                   |
|              |                                  | Keyboard Unit – 5.5 pounds   |   |                   |
|              |                                  | Control Unit – 140 pounds (including power supplies)   |   |                   |
| Dimensions:  |                                  | Height<br>(inches)   | Width<br>(inches)   | Depth<br>(inches) |
|              | Monitor                          | 18   | 19.2  | 22                |
|              | Keyboard                         | 3.5  | 17.5  | 6.6               |
|              | Control                          | 27   | 19.2  | 14.5              |
| Cooling:     |                                  | Quiet, low-speed fans in Monitor and Control Unit  |   |                   |
| Temperature: |                                  | +59 to +100 <sup>o</sup> F (15 to 37.7 <sup>o</sup> C)   |   |                   |
| Cabling:     |                                  | Standard   | Optional  |                   |
|              | Keyboard to Monitor              | 4.5 ft., connector<br>in middle.   | Keyboard to Monitor<br>10 ft. extension                             |                   |
|              | Monitor to Control               | 4 ft. assy. pair   | Monitor to Control<br>20, 50, 100, 300 ft.<br>extensions (in pairs) |                   |
|              | Control to Modem                 | Control to Modem<br>All interface cables are optional and must be ordered.<br>20, 50, 100, and 300 ft. cables available. |   |                   |
|              | Control to Modem Expander        | In addition to interface cables above there are also<br>500 and 1000 ft. cables available.                               |   |                   |
|              | Control to System Direct Connect | Two wire direct connect option   |   |                   |
|              |                                  |  | Control to Printer<br>20, 50, 100, 300 ft.<br>cables available      |                   |

NOTE: For additional cabling information, see Section VI Installation Procedures.

## Introduction and Operation

or group of sections associated with a separate monitor and keyboard. Each section or group of sections associated with a particular keyboard is loaded by that keyboard or by an incoming message addressed to the associated monitor. The content of each section or group of sections of the memory is displayed on the associated monitor at a flicker-free refresh rate. All or part of the display content of a section or group of sections of the memory is transmitted as an outgoing message upon command from the associated keyboard.

### MEMORY

For each monitor-keyboard connected to the Input Display System, six memory bytes are reserved for internal functions and the remainder of the memory assigned to the screen is available for display data. The number of bytes available for display data on a monitor is dependent upon the number of sections of memory assigned to the monitor as follows:

| Number of Sections of Memory Assigned to Monitor | Total Number of Memory Bytes Assigned to Monitor | Number of Display Data Memory Bytes Assigned to Monitor |
|--|--|---|
| 1  | 256  | 250   |
| 2  | 512  | 506   |
| 3  | 768  | 762   |
| 4  | 1024   | 1018  |

The display data consists of both graphics and format effectors.

The first display data memory byte corresponds to the upper left screen position. If this character is a graphic, the next memory byte corresponds to the next screen position on the line. Each graphic character in memory assigns the next memory byte to the next screen position across each line and automatically from the end of one line to the beginning of the next line. A format effector in memory assigns the next memory byte to a screen position dependent upon the format effector. For example, the format effector character NL (new line) assigns the next memory byte to the first position of the next line; therefore, the screen positions after the NL screen position do not have an associated byte in memory. These screen positions are referred to as unassigned positions. Two memory bytes are required for eight of the graphic characters, \$, %, &, ≤, x, ] , ≠, and ←. One memory byte is required for each screen for the end-of-screen position. This byte is one of the six mentioned above.

### DISPLAY FORMAT

The memory data content is displayed on a 17-inch CRT. Table I-2 lists the CRT display characteristics. The tube is masked to correspond to the horizontal and vertical extremities of the 80-character by 25-line format, which

yields a 12-inch horizontal by 9-inch vertical, rectangular display presentation.

Corresponding to each of the graphic codes stored in memory, the appropriate graphic character is generated by using up to 12 straight-line strokes to form the character. A cursor presentation is also provided. It appears at the character position where the next keyboard or Input/Output operation is to take place. The character is formed on the screen as a set of horizontal brackets,  $\square$ , that surround the character position, such that the character to be edited remains viewable. Chart I-1 shows the font of the Input Display graphic character set.

### CHARACTER SET AND CODE

Burroughs Input Display System uses a modified limited subset of the USASCII, USASCII X3.4. Figures I-2 and I-3 designate the characters to each seven-bit character code for receive and transmit respectively.

With the exception of five character substitutions, namely the characters  $\geq$ ,  $\leq$ , x, ≠ and ← in lieu of the characters !, ', /, ^ and —, respectively, the figures agree with the proposed USASCII. These figures differ only in the inclusion of certain characters.

#### Undesignated Codes

Received character codes undesignated in Figure I-2 are undefined in the Input Display System. The character codes undesignated in Figure I-3 are not generated by the Input Display System.

### OPERATION

Operation may be considered to consist of four phases: Composition, Send, Wait, and Receive. The modes are independent for each keyboard.

#### COMPOSITION PHASE

In the composition phase, the operator may compose a message by using the keyboard shown diagrammatically in Figure I-4. The operator has complete freedom to edit and compose information anywhere on the screen (except with forms mode option).

The composition phase is ended by pressing the SEND or PRINT keys, either of which evoke the Send phase.

During composition phase, the KEYBOARD LOCK indicator is OFF. Depressing the KEYBOARD key reverts the terminal to the composition phase. The keyboard consists of graphic, cursor movement, and control keys. Certain of the keys work in conjunction with the SHIFT key.

A graphic key causes the character depicted by the keytop designation to appear at the position of the cursor. If the cursor position already displays a graphic character, it is replaced by the new character. The cursor moves forward

Introduction and Operation

TABLE I-2 DISPLAY CHARACTERISTICS

Cathode Ray Tube (CRT)

|                       |                           |
|-----------------------|---------------------------|
| Deflection Technique: | Electromagnetic           |
| Overall Tube Size:    | 17 in. Rectangular        |
| Phosphor Type:        | P4 Aluminized             |
| Viewable Tube Area:   | 9 in. high by 12 in. wide |
| Brightness:           | 50 foot-lamberts (min.)   |

Deflection Circuitry

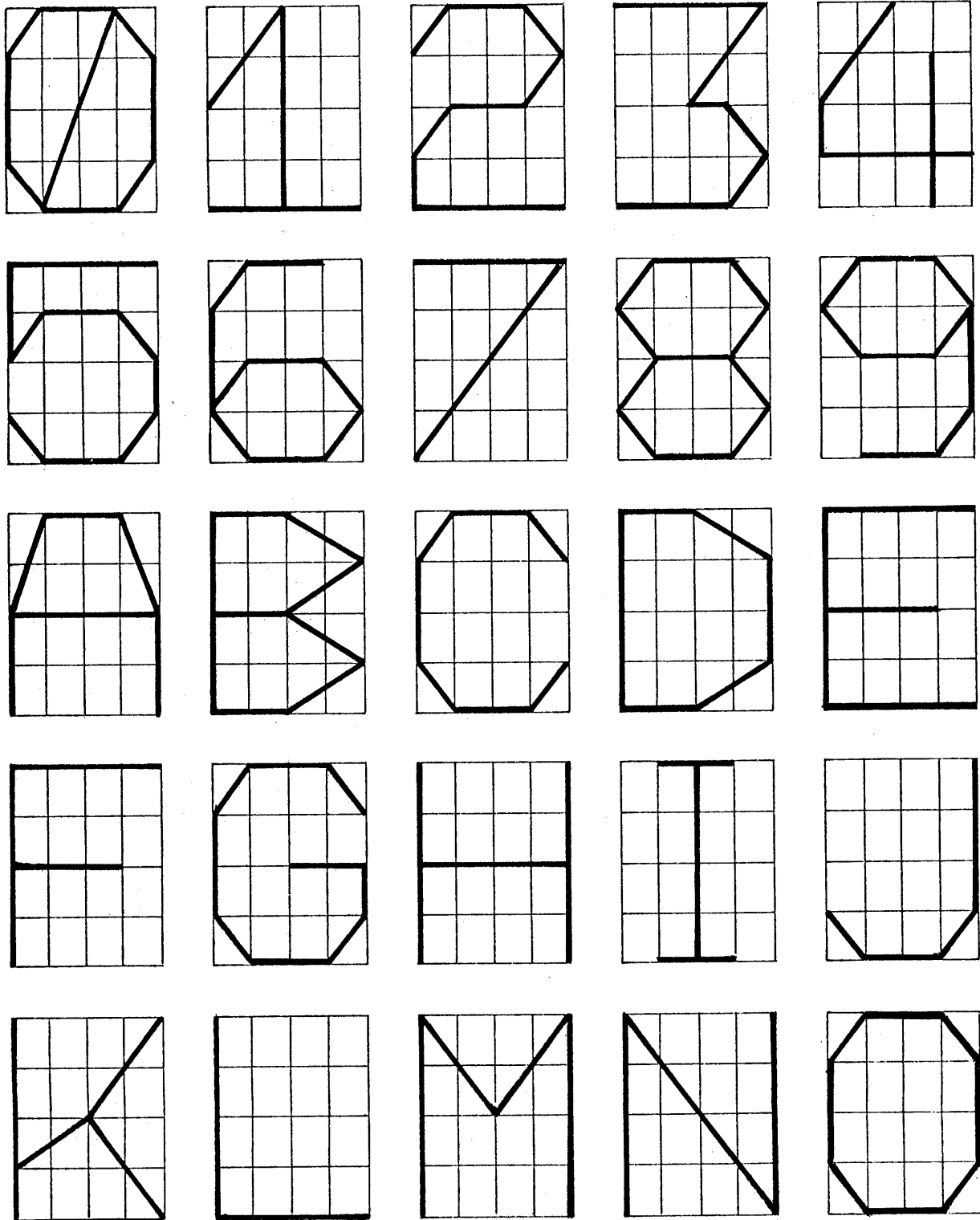
|  |  |
|--|--|
| Writing Rate:                            | 100,000 inches/sec (max.)                  |
| Flyback Time:                            | 42.3 microseconds                          |
| Character to Character Positioning Time: | 2.7 microseconds                           |
| Small Signal Response:                   | 1 MHz                                      |
| Refresh Rate:                            | 50 to 60 Hz synchronized to the power line |

Characters

|                                 |   |
|---------------------------------|---|
| Repertoire:                     | 63 graphic characters, blank, cursor, group separator (Δ), record separator (◀), unit separator (▶) |
| Method of Generation:           | Stroke type   |
| Max. strokes per character:     | 12  |
| Drawing Time:                   | 0.9 microsec/stroke   |
| Character Size (nominal):       | 0.150 in. high by 0.115 in. wide  |
| Character Spacing:              | 7 characters/in.  |
| Line Spacing:                   | 3.33 lines/in.  |
| Stroke Brightness Compensation: | Two levels – compensated with residual variation of 1.66:1  |

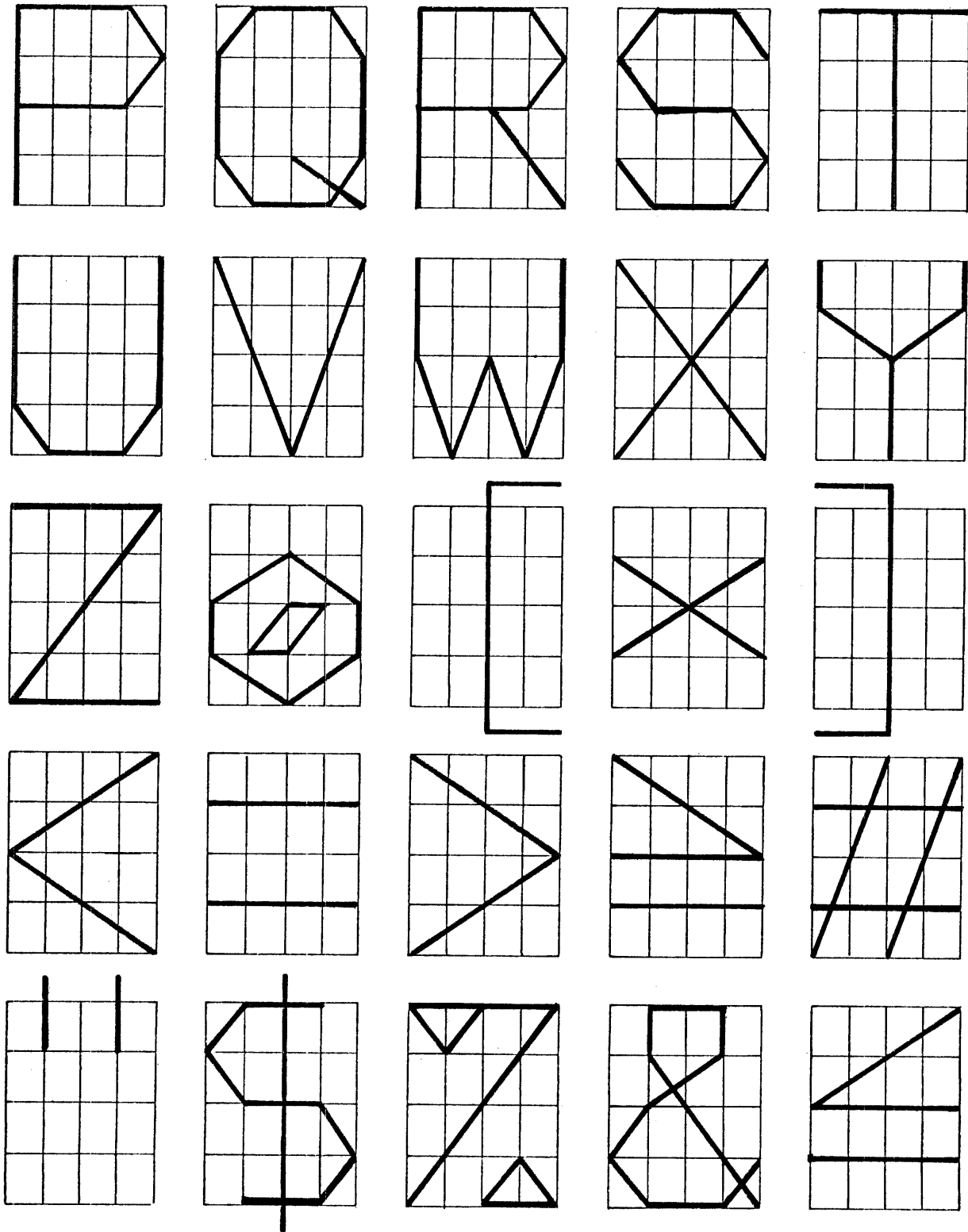
Introduction and Operation

CHART 1



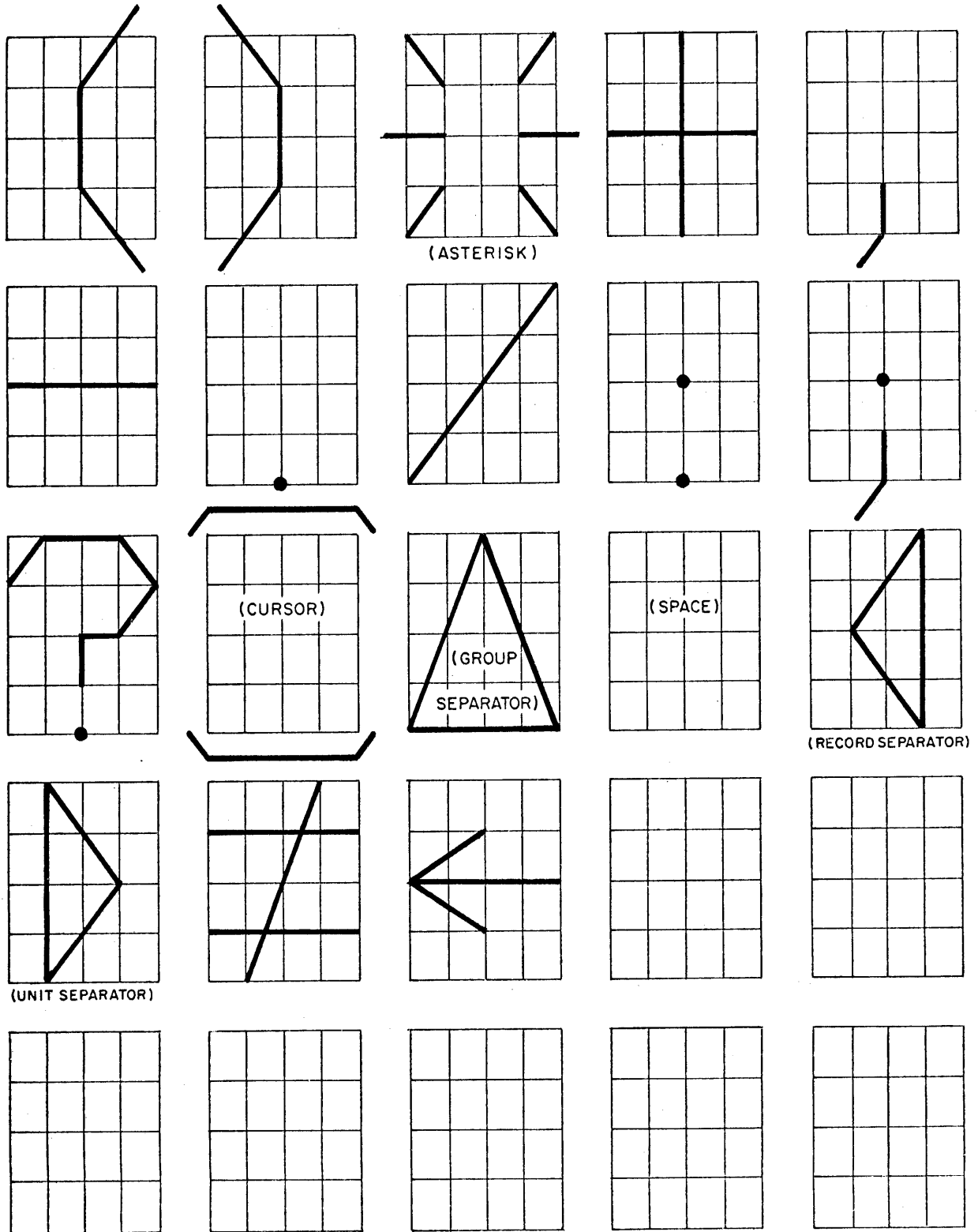
Introduction and Operation

CHART 1 (CONTD.)



Introduction and Operation

CHART 1 (CONTD.)





Introduction and Operation

| BITS           |                |                |                |                 | 0 0     | 0 0 1 | 0 1 0 | 0 1 1 | 1 0 0 | 1 0 1 | 1 1 0 | 1 1 1 |
|----------------|----------------|----------------|----------------|-----------------|---------|-------|-------|-------|-------|-------|-------|-------|
| b <sub>4</sub> | b <sub>3</sub> | b <sub>2</sub> | b <sub>1</sub> | COLUMN<br>ROW ↓ | 0       | 1     | 2     | 3     | 4     | 5     | 6     | 7     |
| 0              | 0              | 0              | 0              | 0               | NUL     | DLE   | SP    | 0     | @     | P     | \     | p/POL |
| 0              | 0              | 0              | 1              | 1               | SOH     |       | ∑     | 1     | A     | Q     | a     | q/SEL |
| 0              | 0              | 1              | 0              | 2               | STX     |       | "     | 2     | B     | R     | b     | r/SEQ |
| 0              | 0              | 1              | 1              | 3               | ETX     |       | #     | 3     | C     | S     | c     | s/FSL |
| 0              | 1              | 0              | 0              | 4               | EOT     |       | \$    | 4     | D     | T     | d     | t/BSL |
| 0              | 1              | 0              | 1              | 5               | ENQ     | NAK   | %     | 5     | E     | U     | e     | u     |
| 0              | 1              | 1              | 0              | 6               | ACK     | SYN   | &     | 6     | F     | V     | f     | v     |
| 0              | 1              | 1              | 1              | 7               | BEL/CON | ETB   | ≤     | 7     | G     | W     | g     | w     |
| 1              | 0              | 0              | 0              | 8               |         |       | (     | 8     | H     | X     | h     | x     |
| 1              | 0              | 0              | 1              | 9               | HT      |       | )     | 9     | I     | Y     | i     | y     |
| 1              | 0              | 1              | 0              | 10              | LF      |       | *     | :     | J     | Z     | j     | z     |
| 1              | 0              | 1              | 1              | 11              | VT      | ESC   | +     | ;     | K     | [     | k     | {     |
| 1              | 1              | 0              | 0              | 12              | FF      |       | ,     | <     | L     | ]     | l     |       |
| 1              | 1              | 0              | 1              | 13              | CR      | GS    | -     | =     | M     | ^     | m     | }     |
| 1              | 1              | 1              | 0              | 14              |         | RS    | .     | >     | N     | _     | n     | ~     |
| 1              | 1              | 1              | 1              | 15              |         | US    | /     | ?     | O     | ←     | o     | DEL   |

- |     |   |     |                                |
|-----|---|-----|--------------------------------|
| NUL | NUL (CC)                                | SYN | SYNCHRONOUS IDLE (CC)          |
| SOH | START OF HEADER (CC)                    | ESC | ESCAPE (CC)                    |
| STX | START OF TEXT (CC)                      | GS  | GROUP SEPARATOR (IS)           |
| ETX | END OF TEXT (CC)                        | GSL | GROUP SELECT** (BCS)           |
| HT  | HORIZONTAL TABULATION (FE) – FIXED TAB  | DEL | DELETE*                        |
| LF  | LINE FEED (FE)                          | CR  | CARRIAGE RETURN (FE)           |
| VT  | VERTICAL TABULATION (FE) – VARIABLE TAB | US  | UNIT SEPARATOR (IS)            |
| FF  | FORM FEED (FE)                          | ENQ | ENQUIRY (CC)                   |
| RS  | RECORD SEPARATOR (IS)                   | ETB | END OF TRANSMISSION BLOCK (CC) |
| EOT | END OF TRANSMISSION (CC)                | DLE | DATA LINK ESCAPE (CC)          |
| NAK | NEGATIVE ACKNOWLEDGE (CC)               | FSL | FAST SELECT (BCS)              |
| ACK | ACKNOWLEDGE (CC)                        | BSL | BROADCAST (BCS)                |
| BEL | ATTENTION SIGNAL (CC)                   | CON | CONTENTION (BCS)               |
| POL | POLL (BCS)                              | SEQ | SEQUENTIAL (BCS)               |
| SEL | SELECT (BCS)                            |     |                                |
- (CC) COMMUNICATION CONTROL, (FE) FORMAT EFFECTOR, (IS) INFORMATION SEPARATOR  
 (BCS) BURROUGHS COMMUNICATION STANDARD

\* IN THE STRICT SENSE, DEL IS NOT A CONTROL CHARACTER.  
 \*\* ANY CHARACTER IN COLUMN 2-6.

Fig. I-2 RECEIVE CHARACTER CODES

one position. When a graphic key is operated with the cursor in the last position of the line, the cursor moves to the first position of the next line down, except that the cursor does not move from the last position of the last line.

Graphic keys with dual keytop designations work in conjunction with the SHIFT key. With the SHIFT key not depressed, the character depicted by the lower keytop designation appears at the cursor position. With the SHIFT

key depressed the character depicted by the upper keytop designation appears at the position of the cursor. Keys with a single keytop designation are unaffected by the SHIFT key.

Keys are included to move the cursor forward or backward a single position, or to the first position of the next line down, the same line, the next line above, or the top line. Some of these keys also have erase capability when

Introduction and Operation

| BITS           |                |                |                |                | 0              | 0              | 0      | 0   | 1   | 1   | 1  | 1 |   |   |   |     |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|--------|-----|-----|-----|----|---|---|---|---|-----|
|                |                |                |                |                | 0              | 0              | 1      | 1   | 0   | 0   | 1  | 1 |   |   |   |     |
| b <sub>7</sub> | b <sub>6</sub> | b <sub>5</sub> | b <sub>4</sub> | b <sub>3</sub> | b <sub>2</sub> | b <sub>1</sub> | COLUMN | ROW | 0   | 1   | 2  | 3 | 4 | 5 | 6 | 7   |
| 0              | 0              | 0              | 0              | 0              | 0              | 0              |        |     |     |     | SP | 0 | @ | P |   | POL |
| 0              | 0              | 0              | 0              | 1              | 1              | 1              |        |     | SOH |     | ≥  | 1 | A | Q |   |     |
| 0              | 0              | 0              | 1              | 0              | 2              | 2              |        |     | STX |     | "  | 2 | B | R |   |     |
| 0              | 0              | 1              | 1              | 3              | 3              | 3              |        |     | ETX |     | #  | 3 | C | S |   |     |
| 0              | 1              | 0              | 0              | 4              | 4              | 4              |        |     | EOT |     | \$ | 4 | D | T |   |     |
| 0              | 1              | 0              | 1              | 5              | 5              | 5              |        |     | ENQ | NAK | %  | 5 | E | U |   |     |
| 0              | 1              | 1              | 0              | 6              | 6              | 6              |        |     | ACK | SYN | &  | 6 | F | V |   |     |
| 0              | 1              | 1              | 1              | 7              | 7              | 7              |        |     |     |     | ≤  | 7 | G | W |   |     |
| 1              | 0              | 0              | 0              | 8              | 8              | 8              |        |     |     |     | (  | 8 | H | X |   |     |
| 1              | 0              | 0              | 1              | 9              | 9              | 9              |        |     | HT  |     | )  | 9 | I | Y |   |     |
| 1              | 0              | 1              | 0              | 10             | 10             | 10             |        |     | LF  |     | *  | : | J | Z |   |     |
| 1              | 0              | 1              | 1              | 11             | 11             | 11             |        |     | VT  |     | +  | ; | K | [ |   |     |
| 1              | 1              | 0              | 0              | 12             | 12             | 12             |        |     |     |     | ,  | < | L | X |   |     |
| 1              | 1              | 0              | 1              | 13             | 13             | 13             |        |     | CR  | GS  | -  | = | M | ] |   |     |
| 1              | 1              | 1              | 0              | 14             | 14             | 14             |        |     |     | RS  | .  | > | N | # |   |     |
| 1              | 1              | 1              | 1              | 15             | 15             | 15             |        |     |     | US  | /  | ? | O | ← |   |     |

- |       |  |     |                           |
|-------|--|-----|---------------------------|
| NUL   | NULL   | ACK | ACKNOWLEDGE (CC)          |
| SOH   | START OF HEADER (CC)   | NAK | NEGATIVE ACKNOWLEDGE (CC) |
| STX   | START OF TEXT (CC)   | POL | POLL (BCS)                |
| ETX   | END OF TEXT (CC)   | SYN | SYNCHRONOUS IDLE (CC)     |
| ENQ   | ENQUIRY (CC)   | GS  | GROUP SEPARATOR (IS)      |
| HT    | HORIZONTAL TABULATION (FE) – FIXED TAB                                   | CR  | CARRIAGE RETURN (FE)      |
| VT    | VERTICAL TABULATION (FE) – VARIABLE TAB                                  | US  | UNIT SEPARATOR (IS)       |
| LF    | LINE FEED (FE)   | RS  | RECORD SEPARATOR (IS)     |
| EOT   | END TRANSMISSION (CC)  |     |                           |
| (CC)  | COMMUNICATION CONTROL, (FE) FORMAT EFFECTOR, (IS) INFORMATION SEPARATOR, |     |                           |
| (BCS) | BURROUGHS COMMUNICATION STANDARD   |     |                           |

Fig. I-3 TRANSMIT CHARACTER CODES

used in conjunction with the SHIFT key. In addition, horizontal tabulation is provided. Fixed tab stops are located in the first position and every eighth position thereafter, i.e., positions 1, 9, 17, 25, 33, 41, 49, 57, 65, and 73.

A cursor movement operation which moves the cursor to a screen position to which a memory byte is assigned does not alter the content of memory. A cursor movement operation from an assigned position to an unassigned position alters the content of the memory to make the new cursor position an assigned position, e.g., an NL (new line) character is inserted in memory when the NEW LINE key is operated if the first position of the next line is unassigned. The following keys are referred to as cursor movement keys:

Space bar:— The space bar moves the cursor forward one position. From the last position of a line the cursor moves

to the first position of the next line down; except the cursor will not move from the last position of the last line.

With the SHIFT key depressed, the space bar also causes the SP (space) character to replace the character at the cursor position before the cursor is moved. (In this regard, the space bar may be considered a graphic.)

Backspace:— The BACKSPACE key moves the cursor backward one position, unless the cursor is in the first position of the first line in which case the key is inoperative. When the cursor position is at the first position of any other line, the BACKSPACE key will cause the cursor to be positioned to the last character position of the line above.

New Line:— The NEW LINE key moves the cursor to the first position of the next line down. The NEW LINE key, when the cursor is in the last line, is inoperative.

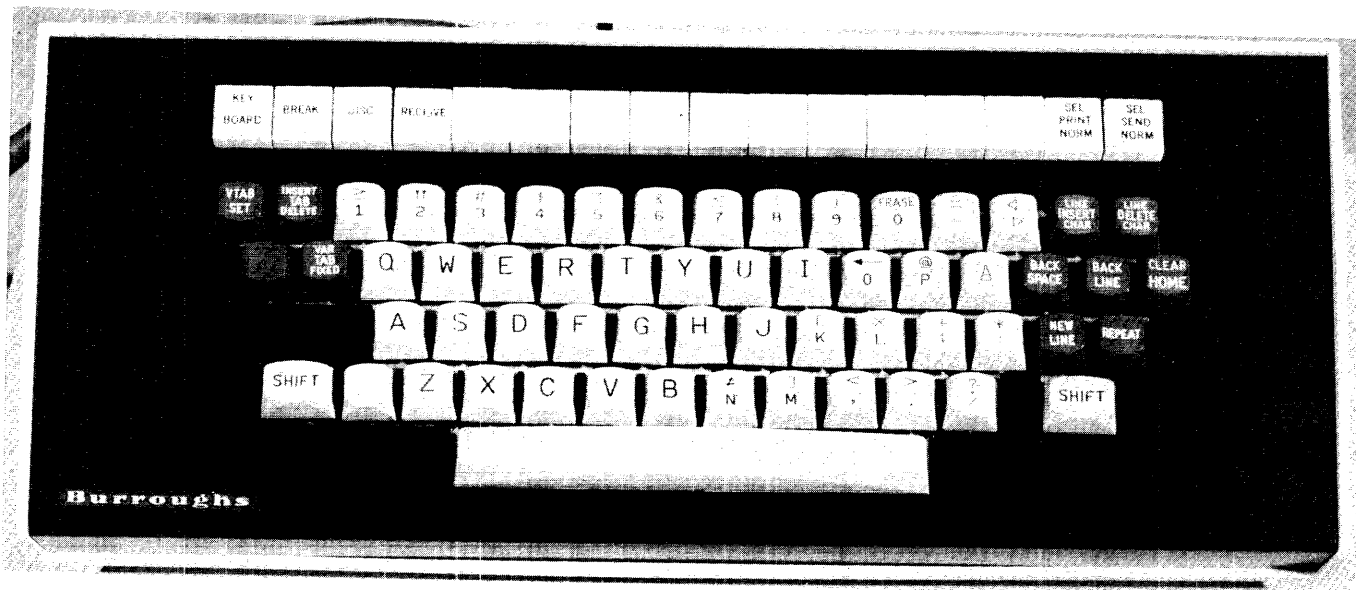
Introduction and Operation

Fig. I-4 INPUT DISPLAY KEYBOARD

**Back Line:**— The BACK LINE key moves the cursor to the first position of the same line if it is not positioned at the first position of a line. When it is positioned at the first position of a line, instead, the cursor is moved to the first position of the line above. The BACK LINE key is inoperative when the cursor is in the first position of the top line.

**Clear/Home:**— The CLEAR/HOME key moves the cursor to the first position of the top line. With the SHIFT key also depressed, the CLEAR/HOME key also clears the screen.

**Tab:**— With the SHIFT key not depressed, the TAB key moves the cursor to the next tab stop (fixed or variable). If there is no tab stop to right of the cursor, the cursor will move to the first tab stop on the next line except when the cursor is in the last line. The TAB key is used in conjunction with a depressed SHIFT key for the Single Variable Tab option.

**Tab Insert/Delete:**— With the SHIFT key depressed, the TAB INSERT causes a fixed tab character to be inserted at the cursor position. The cursor does not move. If the line contains more than 72 character positions, the tab insert function is inhibited. If the cursor is positioned over a tab character, the TAB DELETE (SHIFT key not depressed) causes deletion of the tab character and as a result, the character at the next tab stop (which may not be displayed) is moved to the cursor position and following characters move left a corresponding number of positions or a full tab stop.

Included as part of the keyboard assembly are six control keys, the function of which are as follows:

**Repeat:**— The REPEAT key causes repetition, at the rate of 10 per second, of the function defined by a graphic or format effector key that is operated concurrently with the REPEAT key. Either key may be operated first, and repetition stops when either key is released.

**Send:**— With the SHIFT key not depressed, the SEND key invokes Frame Send to the System Interface. With the SHIFT key depressed, the SEND key invokes Selective Send to the System Interface.

**Keyboard:**— The KEYBOARD key initiates the composition phase.

**Break:**— The BREAK key initiates the receive break function.

**Disconnect:**— The DISCONNECT key is used only in conjunction with the Data-Set interface and the switched telephone network to initiate the disconnect or hang-up function.

**Receive:**— The RECEIVE key evokes the Wait phase. Additionally, there are six indicators as follows:

**Power:**— The POWER indicator is turned ON when the monitor power on switch is activated.

Introduction and Operation

**Error:**— The INPUT ERROR indicator is turned ON when an input error is detected. The INPUT ERROR indicator is turned OFF by START of a receive message.

**Retransmit:**— The RETRANSMIT indicator is turned ON when the system response to a transmitted message from the terminal is a NAK, or an invalid character, or the system gives no response at all within a specified time. The RETRANSMIT indicator is turned OFF by depressing the SEND key.

**Receive Alarm:**— The RECEIVE ALARM indicator is turned ON when the system attempts to send a message to a terminal which is not in Wait phase. The RECEIVE ALARM indicator is turned OFF when the terminal is placed in the Wait phase.

**Keyboard Lock:**— The KEYBOARD LOCK indicator is turned ON whenever the SEND, PRINT, RECEIVE keys are activated, or a BSL, GSL, or SEQ message is received. It is turned OFF by either the KEYBOARD switch or whenever the terminal ends the receive phase or terminates the print operation.

#### SEND PHASE

The Send phase may be evoked by the keyboard operator to send a display data message to the System or a print message (option). The KEYBOARD LOCK indicator is ON.

Upon completion of sending a message the TERMINAL goes to Wait phase.

There are two types of display data send operations, Selective Send and Frame Send. The text generated by Selective Send is the content of memory corresponding to the display data from the cursor position to, but not including, the first following GS ( $\Delta$ ) character or to and including the last data character; whichever occurs first. In Frame Send, the text is the content of memory corresponding to all the display data on the screen.

The operator normally either completes his composition by writing GS ( $\Delta$ ), then positions the cursor to the beginning of his message before evoking Selective Send, or he evokes Frame Send. In either case, following the Send operation, the cursor is left in its original position on the screen and the receipt of an ACK response positions the cursor to either end of screen, or, in the case of Selective Send, over the GS ( $\Delta$ ) character.

Each stored NL (new line) character is transmitted as CR LF.

#### WAIT PHASE

The KEYBOARD LOCK indicator is ON during the Wait phase. A Receive message terminates the Wait phase and evokes the Receive phase. If the operator terminates

the Wait phase with the KEYBOARD key, the keyboard returns to composition phase.

#### RECEIVE PHASE

The Receive phase is evoked by the start of a receive message addressed to the associated screen. The keyboard is inoperative. The INPUT ERROR indicator is reset at the beginning of the receive message.

The terminal examines the input data bits as characters and places the terminal in Receive phase when STX is detected. The characters following are processed until ETX or ETB is detected. If circuit CF goes OFF before ETX is detected, the error is set and the ERROR indicator is turned ON.

The System may send a message at any time. However, the System usually responds only to a message from the operator. When the System sends a message to the terminal for another reason, that message is termed as an unsolicited message and will cause a receive alarm indicator to be lit. This feature is to allow the system to inform the operator that it has traffic to send. The operator may place the terminal in the Wait phase by depressing the RECEIVE key. Broadcast, Group Select and Sequential Select messages (options) may be unsolicited messages which automatically lock the keyboard and evoke Receive phase.

During the Receive phase, the System has control of the keyboard-monitor. The display data in the message may be preceded by the FF (form feed) character which clears the screen and causes subsequent message display data to be loaded starting in the first display data memory byte (which corresponds to the first display position). Otherwise, the graphic data and format effectors load starting in the memory location corresponding to the screen position of the cursor. At the end of an error-free message, the cursor is left positioned after the last display character received. In the case of an input error, the cursor is left in the original position before reception of the message.

Received data that exceeds the capacity of either the screen or the memory is discarded. The input error is set and the INPUT ERROR indicator is turned ON to indicate such loss of data. The response to a message containing an overflow error is NAK.

During the Receive phase, message integrity is checked by examining each character individually for correct character parity. If the message does not pass this test, the input error is set and the INPUT ERROR indicator is turned ON, and a ? is substituted for each character in error.

The text characters that are received by the terminal are termed as Graphics, Format Effectors, and Control Characters.

Each received graphic character and some Format Effectors are stored in the display memory.

The BREAK key is provided to signal the System

## Introduction and Operation

TABLE I-3 RECEIVE CHARACTER FUNCTIONS

| CHARACTER  | FUNCTION  |
|--|---|
| NUL<br>SYN<br>DEL  | USED TO ACCOMPLISH TIME FILL FOR THE TERMINAL AND TIME AND MEDIA FILL FOR THE SYSTEM. ONLY SYN IS EXCLUDED FROM BLOCK CHECK SUMMATION.  |
| STX  | INITIATES ACCEPTANCE OF CHARACTERS AND BLOCK CHECK SUM IF TERMINAL HAS BEEN SELECTED.   |
| ETX<br>ETB   | TERMINATES ACCEPTANCE OF CHARACTERS.  |
| ESC  | INITIATES A SPECIAL CONTROL SEQUENCE. TABLE I-4 APPLIES DURING SPECIAL CONTROL SEQUENCES.   |
| ≥ " = \$ % & ≤<br>( ) * + , -<br>/ Ø 1 2 3 4 5<br>6 7 8 9 : ; <<br>= > ? @ A B C<br>D E F G H I J<br>K L M N O P Q<br>R S T U V W X<br>Y Z [ x ] ≠ ←<br>GS (Δ) RS (◀) US (▶) | LOADS CHARACTER INTO MEMORY EXCEPT DURING COMMUNICATION SELECTION (MAY BE USED FOR GROUP SELECTION EXCEPT FOR GS, RS, AND US).  |
| SP   | LOADS SP (SPACE) CHARACTER INTO MEMORY.   |
| \ a b c d e f<br>g h i j k l m<br>n o p q r s t<br>u v w x y z /<br>  } ~  | LOADS CORRESPONDING COLUMN 4 OR 5 (UPPER CASE) CHARACTER INTO MEMORY EXCEPT DURING COMMUNICATION SELECTION (MAY BE USED FOR GROUP SELECTION EXCEPT FOR CHARACTER CODES FROM p THROUGH ~). |
| LF   | LOADS NL (NEW LINE) CHARACTER INTO MEMORY.  |
| CR   | NO FUNCTION IS PERFORMED.   |
| FF   | CLEARs SCREEN AND CAUSES FOLLOWING CHARACTERS TO BE LOADED STARTING AT THE BEGINNING OF MEMORY.   |
| HT   | LOADS FIXED TAB CHARACTER, TF, INTO MEMORY.   |
| VT   | LOADS VARIABLE TAB CHARACTER, TV, INTO MEMORY.  |
| SOH  | INITIATES BLOCK CHECK SUM.  |
| EOT  | INITIATES IDLE OR PASSIVE CONDITION.  |
| ENQ  | INITIATES A RESPONSE.   |
| ACK  | POSITIVE ACKNOWLEDGEMENT.   |
| NAK  | NEGATIVE ACKNOWLEDGEMENT.   |
| DLE  | USED IN CONJUNCTION WITH EOT TO INDICATE DISCONNECT.  |
| POL  | USED DURING COMMUNICATION SELECTION TO INDICATE POLL.   |
| SEL  | USED DURING COMMUNICATION SELECTION TO INDICATE SELECTION.  |
| SEQ  | USED DURING COMMUNICATION SELECTION TO INDICATE SEQUENTIAL SELECTION.   |
| FSL  | USED DURING COMMUNICATION SELECTION TO INDICATE FAST SELECTION.   |
| BSL  | USED DURING COMMUNICATION SELECTION TO INDICATE BROADCAST SELECTION.  |
| BEL  | USED DURING COMMUNICATION SELECTION TO INDICATE CONTENTION.   |
| UNDESIGNATED IN TABLE I-3  | NO FUNCTION IS PERFORMED.   |

(message source) to terminate transmission to the terminal. The method of transmission is dependent upon the type of System interface.

A summary of the receive character functions is shown in Table I-3.

## Introduction and Operation

Receipt of any graphic character, i.e., Figure I-2 columns 2 through 5, causes storage of that character in memory. Note the SP (space) is considered a graphic. The group separator, GS, is also treated as a graphic and has no control significance in Receive phase. Characters from columns 6 and 7, except for the last four characters in column 7 including DEL, are translated to the equivalent character in columns 4 and 5 by changing b6 to 0. Thus, received lower case characters are displayed (and may be later transmitted) as upper case characters.

Format Effectors are used to define the display format on graphic characters. The function of the format effectors is as follows:

CR (carriage return). Receipt of the CR character has no effect. It is equivalent to a NUL character, except for Block Check summation.

LF (line feed). Receipt of the LF character causes the NL (new line) character to be stored in memory, and as a result, the next stored character is displayed at the first position of the next line down.

FF (form feed). Receipt of the FF character clears the screen and causes the following characters to be loaded starting in the first memory locations.

HT (fixed tab). Receipt of the HT character causes the fixed tab character, TF, to be stored in memory; and as a result, the next stored character is displayed at the next fixed tab stop.

The following are Control Characters and their respective functions when encountered in a received message:

NUL, SYN, or DEL. These characters may be used to accomplish time fill for the terminal and time and media fill for the System. No function is performed when these characters are received. Only SYN is excluded from the block check summation.

ESC (control sequence). Receipt of ESC indicates the beginning of a special control sequence. The special control sequences following ESC's cause the actions indicated in Table I-4. A message may contain any number of control sequences. Special control sequence characters which are not defined terminate the control sequence, and no function is performed.

#### SYSTEM OPTIONS

The basic set consists of three units (monitor, keyboard, and control) which are put together to form a free-standing assembly.

Although the normal display system operates from a 60 Hertz power source, a 50 Hertz system will be available for International use.

| Character | Function                       |
|-----------|--------------------------------|
| :         | Reposition Cursor (Option)     |
| ;         | Set Variable Tab Stop (Option) |
| Others    | Terminate ESC sequence         |

TABLE I-4 SPECIAL RECEIVE CONTROL SEQUENCE CHARACTER FUNCTION

One of the features of the display system is the ability of the memory to be partitioned as described previously to accommodate the following combinations of keyboard-monitors and displayed characters:

- 2 keyboard-monitors (Control Unit II) with 506 characters each,
- 2 keyboard-monitors (Control Unit II) with 250 and 762 characters,
- 3 keyboard-monitors (Control Unit III) with 250, 250 and 506 char.,
- 4 keyboard-monitors (Control Unit IV) with 250 characters each.

When the keyboard-monitor combination is in the free-standing configuration, the monitor plugs into the control unit with nominal 4-foot cables. However, when configured as a multiple device, the keyboard-monitors can be operated with up to 300 foot separation between the monitors and the control unit. The available options are 20, 50, 100, and 300 feet. The keyboard normally plugs into the monitor with the connection made under the monitor assembly. An optional 10 foot extension cable is available which will allow the keyboard to be operated away from the monitor.

An optional printer interface allows the connection of a Teletype Model 33 TCB page printer to the display control unit. The memory content associated with any monitor/keyboard may be printed at 10 characters per second upon command of the associated keyboard PRINT key or by the presence of B3 in the AD2 character. With the SHIFT key not depressed, the PRINT key evokes Frame Send to the Printer Interface. With the PRINT key depressed, the PRINT key evokes Selective Send to the Printer Interface. The message format of message to the printer is as follows:

```
C C L A C C L
R R F D R R F      text      (from memory)
> 2
```

The two least significant bits of the AD2 character indicates the monitor from which the text data is obtained. It is printed as 0, 1, 2, or 3. Within text, each NL character from memory is translated to CR, CR, LF. In addition, CR, CR,

---

 Introduction and Operation
 

---

LF is inserted following each character in position 80. Tabulation should be avoided in messages to be printed because the printer does not have the tabulation function. When a receive message is addressed to a screen which is in the process of being printed, the terminal will respond with a NAK. The printer can be located up to a maximum distance of 300 feet from the control unit. Optional cables available are in lengths of 20, 50, 100, and 300 feet.

A second option is available when using the printer interface which enables a portion of the Input Display System memory (250, 506, or 762 characters) to be used as a buffer between communications inputs and the printer. The portion of memory so assigned is unavailable for display purposes, and the Input Display System will function exactly as if a monitor-keyboard unit was assigned to the printer buffer area of memory. Multiple buffer areas can be assigned. Messages placed in the buffer area by the system (via the use of the AD2 address specifying the buffer area) may be printed via a print command from the system.

#### DISPLAY OPTIONS

Insert and Delete Options provide additional keyboard editing capability whereby characters and lines may be inserted into or deleted from existing screen data. In the case of inserting data, the existing data, starting at the insertion position, is moved to the right and/or down to make room for the data being inserted.

When the INSERT key is operated, the character under the cursor and following characters move one position to the right, a space character is inserted at the cursor position and the cursor remains at the original position. The movement of characters to make room for the insertion involves all characters from the one under the cursor to the first character that precedes an unused screen location, i.e., at the end of a partially filled line or at the end of a tab column. When a character is moved from the last position of a line, it moves to the first position of the following line. Characters may be pushed from the bottom of the screen or memory by an insert operation.

With the SHIFT key depressed, the INSERT key moves the character under the cursor to the first position of the next line down and following characters to following positions. The cursor is not moved.

The DELETE key deletes the character under the cursor provided the character is a graphic or space. The character to the right of the cursor is moved to the cursor position and the following characters through the end of a tab column or the end of the line are moved one position to the left. Character DELETE has no effect when the cursor is positioned over the Format Effector characters, TF (fixed tab), TV (variable tab), NL (new line), or RS (record separator).

With the SHIFT key depressed, the DELETE key causes erasing of all the characters from the cursor to the

end of the line or RS character, provided the cursor is not positioned over the NL (new line) or RS character. An NL or RS character is placed at the cursor position and the cursor is not moved. If the cursor is positioned over an NL character, the NL character is deleted and as a result, characters from the line below fill the cursor line starting at the cursor position. Any excess characters remain on their line, left-justified. The leftward and upward movement of characters affects all lines following the deleted NL character through the next NL character. If the cursor is positioned over an RS character, the Line Delete function has no effect.

A Single Variable Tab option is available. This option provides a single tab stop which may be set either by a message or from the keyboard. This tab stop may be used for tabulation either by a message or from the keyboard. The single variable tab stop position may be altered by the message character sequence:

```

E   P
S ; O
C   S
  
```

where POS is defined as a horizontal screen position. Time fill characters must follow POS for at least 17 milliseconds when contained in data.

The single variable tab stop position may also be altered by the keyboard to the horizontal position of the cursor using the VAR TAB SET key. In order to avoid superimposition of characters, the variable tab stop should always be set greater than the maximum number of characters from the beginning of any line to the first variable tab character. With the SHIFT key depressed, the TAB key causes the cursor to move to the next single variable tab stop, either on the same line or at the first character of the next line. There is an automatic variable tab stop designated at the first position of each line, in addition to the variable stop, which may be positioned anywhere on the line. The VT character in a message causes the following character to be displayed at the next variable tab stop position.

In the time-shared mode of operation when data for one monitor contains an unusually high number of variable tab (VT) and new line (NL) characters, the time required to display all of the Monitor Data may exceed the time available for the display. If this situation occurs, the display will be terminated when the monitor display time runs out (all following characters for this monitor will not be displayed). The INPUT ERROR indicator of the associated keyboard will be lit, and the System will respond with a NAK.

This option is not to be used when multiple screens are addressed by means of BSL, GSL or SEQ.

A Reposition Cursor option is available that is used to reposition the cursor by means of the message character sequence:

Introduction and Operation

E P L  
S : O I  
C S N

where POS is defined as a horizontal position on a line and LIN is defined as a screen line. USASCII characters 2/0 through 6/15 are used to define horizontal screen positions 1 through 80 respectively. USASCII characters 2/0 through 3/8 are used to define lines 1 through 25 respectively.

If the cursor is repositioned to an unassigned (in memory) screen position the cursor will be moved to the position corresponding to the end of screen character. Following the reposition sequence, at least 33.4 milliseconds of fill characters (NUL or DEL) must be included in a message.

This option is not permitted with multiple screen selection BSL, GSL or SEQ.

A Forms option is available which allows a message to be displayed that inhibits operator access to predefined areas of the screen. This option operates in either one of two modes: unalterable or alterable.

When in the unalterable mode, the RS character (◀) is displayed in character position 1 of line 1. This causes the terminal to prevent the operator from entering data in any character position except those between the US and RS characters. In this mode, the INSERT, DELETE and TAB keys are inhibited to prevent the operator from expanding or contracting any preassigned operating area.

If the cursor is moved to any position between the RS and US characters, the terminal automatically positions the cursor to the first character position following the next US character. For example, if the cursor is initially placed over the character position following a US character and the BACKSPACE key is depressed, the cursor will move back one position (over the US character) and then automatically be repositioned to its original position.

The function of the TAB key in this mode, is to allow the operator to change from one operating area to the next by positioning the cursor to the character position following the next US character. Subsequently, if no other US characters occur between this cursor position and the end of the screen, then the cursor will be positioned to the character following the first US on the screen (i.e., wrap-around).

When in the alterable mode, the US character (▶) is displayed in the first character position of line 1. The terminal functions basically as in the unalterable mode except that the operator is allowed to expand or contract the operator areas of the screen. Each operator area is treated as if it were a separate screen and were completely expandable. If the cursor is positioned to the RS character and data is entered, the RS character and the data is shifted one character to the right and the operator data placed in front of the RS character. If the NEW LINE key is depressed —

causing the cursor to be placed in a non-operator area — the terminal will automatically insert a new line character and assign an additional line to the operator area.

The INSERT and DELETE keys are enabled with the alterable mode. The character delete function is modified to inhibit the deletion of RS characters. The line delete with an alterable form on the screen will cause deletion from the cursor to the end of the line or to the first RS character.

The TAB key operates in the same manner as in the unalterable mode; the variable tab is still inhibited. Note that even though the tab functions are inhibited, the tab characters may be placed by the processor in either the operating or non-operating areas for formatting purposes.

In order to generate a form by means of the keyboard, a Compose Forms mode is used. This mode is enabled by a switch on the Control Unit which then allows the operator access to the entire screen. Forms controls are disabled to permit the operator to type RS and US characters, thereby generating a form to be used for a given application.

Available for stand alone Terminal Systems only is the Paging option. This option provides the terminal operator with some control over groups of messages coming from the System. Each message from the System should be no longer than 240 characters or 3 lines when using this option. With this option, ACK messages are automatically transmitted to the System following each received message. The ACK will always be sent unless the previous message included data which fell in the last 256 characters of available memory or the last 3 lines of the screen. In this case, the RECEIVE ALARM indicator will be lit and NAK is sent as the response to all messages until operator intervention. The PAGE FULL indicator will be lit, and will remain lit as long as data remains in the last 256 characters of available memory or the last 3 lines of the screen. With the RECEIVE ALARM indicator lit, the terminal will respond to all messages addressed to the screen with a NAK. The operator may take either of two actions:

1. Delete characters and/or lines from the existing text on the screen until the PAGE FULL indicator is extinguished. Depressing the Receive key will allow the System to send new messages and extinguish the RECEIVE ALARM indicator.
2. Depress the RECEIVE key while the PAGE FULL indicator is lit. This will clear the screen, reposition the cursor to the top of the screen, reset the RECEIVE ALARM and PAGE FULL indicators, and allow the System to send new messages.

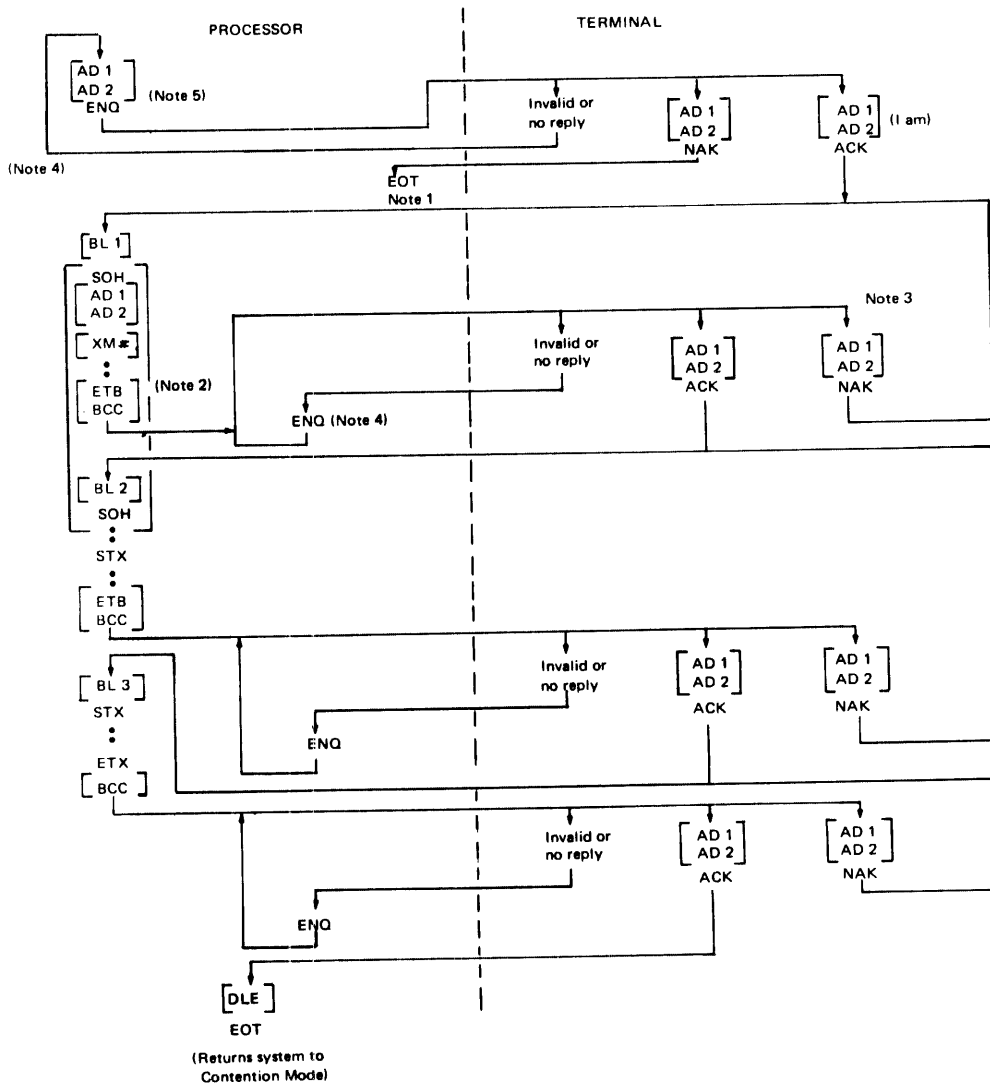
COMMUNICATIONS PROCEDURES

The basic Input Display System is compatible with the proposed Burroughs Standard Data Communications



Introduction and Operation

TABLE I-5



Note 1 – If the terminal is not ready to receive, as indicated by transmission of NAK, the central computer will normally retry the selection at the terminal's proper sequence; however, for some installations it may be desired to repeat a selection sequence immediately.

Note 2 – Provision is made for transmission of blocks of header and/or text. In this case a block is terminated with ETB-BCC. When one block has been acknowledged, the header or text is resumed after transmission of BL # followed by SOH or STX. Response to blocks is the same as response to a message ending in ETX-BCC.

Note 3 – If character parity or block check are not validated by the terminal, it will send NAK. In this case the central computer will retransmit the message "p" times ("p" may be equal to zero). If the terminal still does not acknowledge the message, the central computer will terminate the sequence with EOT.

Note 4 – If the central computer does not receive a response (ACK or NAK) to its message, it may time out and retransmit the block "n" times, where "n" may equal zero. If still no response is received, the computer will terminate the sequence with EOT, after recording the error. The central computer will retain the message for transmission on the next selection sequence to this terminal.

Note 5 – In certain systems implementations where downstream communication is not permitted and transmission can only be to the central processor, the identification characters in a transmission may represent the terminal address ("you are") for selection verification purposes.



Introduction and Operation

COMMUNICATION OPTIONS

Polling and Select

Polling and Select option is available for multi-point operation. This option permits the terminal to transmit data only in response to a poll message, unless the terminal is in the contention mode. The terminal is selected and will accept data if it has received a select message. Included in this option is the poll selection decode, the contention selection decode and the select selection decode. In addition, an EOT character will be transmitted as a no traffic response to a poll message. This option is included with the Fast Select option.

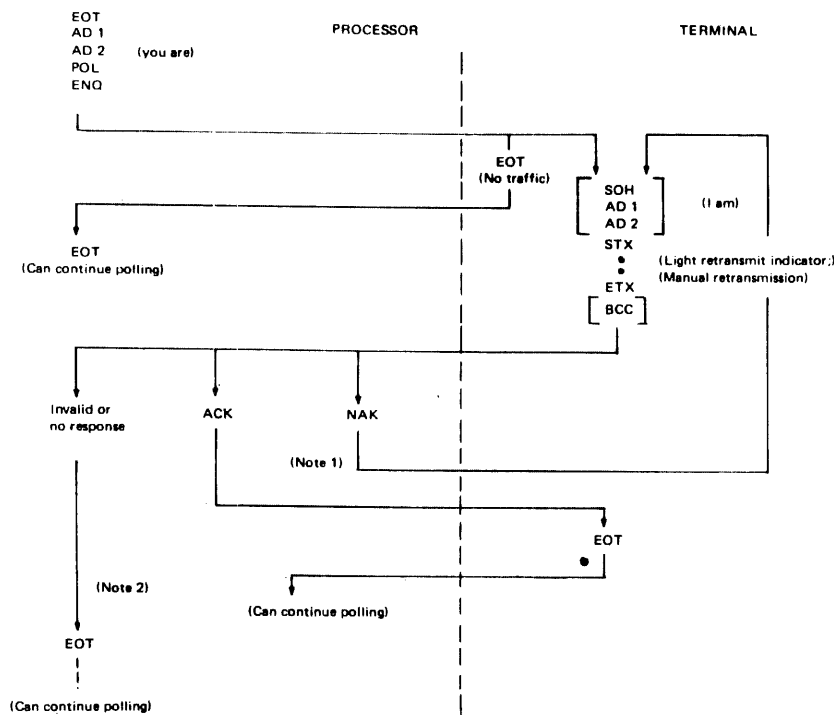
A Selection Addressing option, part of Polling and Select, allows the terminal to respond to ENQ only when ENQ is contained in a message whose first two characters are the correct address for the terminal. This option, used by itself, will allow a time-shared terminal to operate on a point-to-point line. The terminal will interpret the three least significant bits of AD2 as the terminal or printer

address.

Also included as part of the Polling and Select option is the Identification Address option which will allow the terminal to precede all transmitted messages with the terminal address. If the message is a data message preceded by STX, the terminal will precede the STX with the sequence SOH AD1 AD2.

The basic design of the terminal is such that the terminal ignores all special selection codes (i.e., FSL, BSL, GSL and SEQ) even though they are contained in a message received by the terminal. (Refer to the flow diagram in Table I-7.) With the Fast Select (FSL) option incorporated in the terminal, receipt of an FSL character will cause the terminal to check for a STX and accept the data that follows. This means that the terminal does not respond with an ACK to indicate it is ready for a message, but accepts it immediately, thus the term Fast Select. If the terminal is not RECEIVE ready, no response will be generated, the data following the FSL will be ignored, and the RECEIVE ALARM light for the addressed screen will be lit.

TABLE I-7



Note 1 - If the central computer receives a message for which character parity, block check, or terminal address test fails, NAK will be transmitted, calling for a repeat of the transmission. This can be repeated "n" times (to be defined for each particular system and "n" may be equal to zero), at which time, if the test fails, an error will be recorded at the central computer and EOT will be transmitted, terminating the sequence. The terminal may transmit the same message at the discretion of the operator.

Note 2 - If the terminal does not receive ACK, NAK, or EOT, it will retain its message and remain quiet. The central computer will time-out and transmit EOT or DLE EOT, terminating the sequence. In this case the message will be retransmitted when next polled. See section on Error Recovery in Burroughs Standard 1284-9006 for a definition of time-out conditions.

Introduction and Operation

There are three multiple selection decode options: Broadcast Select (BSL), Group Select (GSL), and Sequential Select (SEQ).

**Broadcast Select**

When the Broadcast Select option is incorporated along with the Selection Decode option, the terminal will respond to a received BSL code as the third character following an EOT by putting all incorporated screens into the RECEIVE mode and loading the data following a STX onto all screens. The starting address for loading is derived from the cursor position of screen zero. Those screens with cursor positions before that of screen zero will not be loaded since an end-of-screen character will precede all loaded data in their memory blocks. To insure that all screens are successfully loaded during a BSL operation, it is advisable to transmit a Form Feed (FF) character following the STX.

**Group Select**

The Group Select Option enables the fast selection of any pre-selected subset of terminal screens. Operation is similar to that of Broadcast Select, except that instead of all screens being sensitive to the message following, only those screens with their corresponding Select Flip-flop set will be sensitive. The Group Select option requires incorporation of the Broadcast Select option.

**Sequential Select**

The Sequential Select option can be added if the Group Select option is incorporated. This option permits screens to be successively selected to receive by receiving a SEQ message containing their address. This option allows the setting of the appropriate Select Flip-flop when a particular screen's address is received followed by an SEQ or SEL. Otherwise, operation is identical to Group Select except that the screen whose address preceded the SEL will generate a response after the following ENQ. This screen will also generate a response following receipt of an ETB or ETX.

**Block Check**

With the Block Check option installed, all characters from the SOH character to ETX or ETB, or from the STX character to ETX or ETB will be used to generate or verify the Block Check Character. The STX character is included in the block check summation only when it follows SOH in a message block.

When the Block Check option is not installed,

messages received containing a Block Check Character are accepted; however, the BCC is discarded.

SYSTEM INTERFACE

The standard interface in the Input Display System terminal is a multi-wire synchronous interface which is compatible with RS232 EIA. This interface can be connected to the following Bell System DATA-phone Data Sets: 201A and 201B. Data transfer rates up to 2400 baud can be achieved in the synchronous mode. Asynchronous operation is optionally available for either direct connection to a system over two wires or with the following Bell System DATA-phone Data Sets: 103A/F and 202C/D. Data transfer rates up to 38.4K bits can be achieved in the direct connect operation. Transfer rates for modem operation is currently limited to 1200 baud.

Each character transmitted or received in the asynchronous mode consists of ten nominally equal time intervals, representing a start bit, seven data bits (least significant bit first), a parity bit and a stop bit. Polarity of the start bit is a 'space' or '0'. Stop bit polarity is a 'mark' or '1'. Parity is even for asynchronous operation.

During synchronous operation, each character transmitted or received consists of eight nominally equal time intervals. The first seven bits represent the 7-bit character code and are handled least significant bit first. The eighth bit is the parity bit and is selected to make the sum of the 'marking' bits odd.

In order to obtain synchronism between the System and the terminal, at least four SYN characters are required preceding all transmissions. The Input Display System terminal automatically sends four SYN characters preceding all transmissions to the System. Maintaining synchronism, once a message has been started, requires SYN characters as 'time fill' except between ETX or ETB and the following BCC. SYN characters are not included in the BCC summation in the Input Display System terminal.

When using 4-wire, full duplex, the BREAK key may be used to terminate transmissions by causing circuit BB to be turned off for a period of 25 ms. This function should be disabled when not operating 4-wire.

Operation utilizing switched lines can make use of the automatic disconnect feature in the control unit. This feature allows the processor to cause the terminal to perform a local disconnect of the switched circuit. When the terminal detects the character sequence DLE, EOT, the terminal will respond by turning the CD line to the modem OFF for a period sufficient to cause the modem to disconnect. The CD line is then returned to the ON state.

## Introduction and Operation

## GLOSSARY OF TERMS

| <u>Mnemonic</u> | <u>Page No.</u> | <u>Definition</u>                      | <u>Mnemonic</u>  | <u>Page No.</u> | <u>Definition</u>                                  |
|-----------------|-----------------|--|------------------|-----------------|--|
| -x              | 227             | -x Deflection                          | BELD             | 313             | Bell decoded                                       |
| -y              | 227             | -y Deflection                          | BEV              | 231             | Translator bit B – Even                            |
| -2x             | 227             | -2x Deflection                         | BITTME           | 210             | Bit time   |
| -2y             | 227             | -2y Deflection                         | BLCI             | 205             | Back line cursor inhibit                           |
| +x              | 227             | +x Deflection                          | BLMCY            | 116             | Back line memory cycle                             |
| +y              | 227             | +y Deflection                          | BLMR             | 123             | Back line memory request                           |
| +2x             | 227             | +2x Deflection                         | BLS              | 205             | Back line storage                                  |
| +2y             | 227             | +2y Deflection                         | BOD              | 231             | Translator bit B – Odd                             |
| +5.95V          | 132             | +5.95 Volt Power Supply                | BRC              | 109             | Buffer register clock                              |
| +8V             | 132             | +8 Volt Power Supply                   | BRKn             | J3n             | Break – from keyboard n, n = 0, 1, 2, 3            |
| ACKD            | 313             | ACK Decoded                            | BRn              | 120             | Buffer requester bit n, n = 1 thru 6               |
| ACMA            | 116             | Allow cursor memory address            | BSL              | 322             | Broadcast select                                   |
| ACURL           | 203             | Allow cursor load                      | BSLD             | 322             | Broadcast select decoded                           |
| ADRDO           | 327             | Screen address 0 decoded               | BSCP             | 205             | Backspace complete pulse                           |
| ADRDOE          | 322             | Address screen 0 enable                | CAD              | 208             | Request to send                                    |
| ADRD1           | 326             | Screen address 1 decoded               | CBR              | 311             | Clear to send                                      |
| ADRD1E          | 322             | Address screen 1 enable                | CBS              | 312             | Control bit set                                    |
| ADRD2           | 325             | Screen address 2 decoded               | CCA              | 217             | Count cursor address                               |
| ADRD2E          | 322             | Address screen 2 enable                | CCAB2            | 205             | Count cursor address by 2                          |
| ADRD3           | 324             | Screen address 3 decoded               | CCAD             | 205             | Count cursor address down                          |
| ADRD3E          | 322             | Address screen 3 enable                | CCAU             | 110             | Count cursor address up                            |
| ADR1            | 320             | Screen address bit 1                   | CCF              | 211             | Character not loaded                               |
| ADR1T           | 320             | Screen address bit 1                   | CCR              | 311             | Data set ready                                     |
| ADR2            | 320             | Screen address bit 2                   | CCn              | 319             | Character count n, n = 0 thru 4                    |
| ADR2T           | 320             | Screen address bit 2                   | CDIn             | 217             | Cursor data input bit n, n = 1 thru 10             |
| ADR3            | 331             | Address bit 3 – Print                  | CDPOS            | 121             | Cursor data position                               |
| ADS             | 209             | Asynchronous data set                  | CEV              | 231             | Translator bit C – Even                            |
| ADn             | 320             | Terminal address ADn decoded, n = 1, 2 | CFR              | 311             | Data carrier detected                              |
| AD2B1           | 320             | Screen identification address bit 1    | CHB8             | 121             | Count Horizontal by 8                              |
| AD2B2           | 320             | Screen identification address bit 2    | CHIN             | 108             | Character insert                                   |
| AD2B3           | 320             | Screen identification address bit 3    | CHPC             | 121             | Count horizontal position counter                  |
| AD2E            | 319             | Address 2 Enable                       | CHPn             | 202             | Complement horizontal position bit n, n = 1 thru 7 |
| AEDP            | 124             | Allow end display pulse                | CLKa             | 221             | Clock a, a = A thru H                              |
| AEV             | 231             | Translator bit A – Even                | CLKIOB           | 213             | Clock I/O buffer                                   |
| AKBS            | 215             | Allow keyboard strobe                  | CLS              | 114             | Clear storage                                      |
| ALS1            | 112             | Allow screen 1                         | CMA              | 121             | Count memory address                               |
| ALS2            | 112             | Allow screen 2                         | CMA <sub>n</sub> | 217             | Cursor memory address bit n, n = 1 thru 10         |
| ALS3            | 112             | Allow screen 3                         | CMPUL            | 109             | Count memory pull                                  |
| AMA             | 116             | Allow memory address                   | ÇOD              | 231             | Translator bit C – Odd                             |
| AMF             | 220             | Allow Memory Full                      | CODA             | 115             | Cursor over data                                   |
| AOD             | 231             | Translator bit A – Odd                 | COM              | 128             | Common ground                                      |
| APMA            | 116             | Allow processor memory address         | CON              | 318             | Contention   |
| BAD             | 208             | Transmit data                          | COTI2C           | 228             | Cursor over Tab, insert 2 characters               |
| BBR             | 311             | Receive data                           | CP               | 317             | Character present                                  |
| BDTn            | 227             | Buffered display time – n = 0, 1, 2, 3 |                  |                 |  |

Introduction and Operation

## GLOSSARY OF TERMS (CONTD.)

| <u>Mnemonic</u> | <u>Page No.</u> | <u>Definition</u>  | <u>Mnemonic</u> | <u>Page No.</u> | <u>Definition</u>                                |
|-----------------|-----------------|--|-----------------|-----------------|--|
| CPMA            | 111             | Count processor memory address   | DDMR            | 122             | Delayed display memory request                   |
| CSB1            | 115             | Control storage bit 1, when set, indicates Alterable Forms mode                                | DDR             | 311             | Receive timing                                   |
| CSB2            | 115             | Control storage bit 2, when set, indicates Cursor position is less than Variable Tab setting   | DEND            | 117             | Display end                                      |
| CSB3            | 115             | Control storage bit 3, when set, indicates Memory Full *<br>DMCY or L25 * CDPOS *<br>* HP ≠ 80 | DEV             | 231             | Translator bit D – Even                          |
| CSB4            | 115             | Control storage bit 4, when set, indicates DMCY * L25 * EOL                                    | DIP             | 112             | Display in process                               |
| CSB5            | 115             | Control storage bit 5, when set, indicates Unalterable Forms mode                              | DIS             | 112             | Discharge  |
| CSB6            | 115             | Control storage bit 6, spare – not used  | DISC            | 222             | Discharge  |
| CSB7            | 115             | Control storage bit 7, when set, indicates ESCS * NFLBT3 * MAE                                 | DISCD           | 122             | Discharge delayed                                |
| CSB8            | 115             | Control storage bit 8, when set, indicates HP less than or equal to 72                         | DLNC            | 223             | Delete next character                            |
| CSB9            | 115             | Control storage bit 9, when set, indicates Cursor over NL                                      | DMCY            | 116,121         | Display memory cycle                             |
| CSB10           | 115             | Control storage bit 10, when set, indicates Cursor over ESC                                    | DMR             | 222             | Display memory request                           |
| CSB11           | 115             | Control storage bit 11, when set, indicates Cursor over TAB                                    | DOD             | 231             | Translator bit D – Odd                           |
| CSB12           | 115             | Control storage bit 12, when set, indicates Cursor over ES                                     | DOVFEN          | 327             | Display overflow enable                          |
| CSCLK           | 219             | Count special clock  | DRn             | 228             | Driver n, n = 0 thru 7                           |
| CTM             | 224             | Transfer cursor memory address to memory address   | DRnT            | 126,127         | Driver n inverted, n = 0 thru 7                  |
| CTS             | 211             | Clear to Send  | DSCP            | 101             | Reposition storage clock pulse                   |
| CTTS            | 214             | Change tab to space  | DSRC            | 109             | Display register clock                           |
| CURBEL          | 203             | Cursor before end-of-line  | DSRn            | 119             | Display register bit n, n = 1 thru 6             |
| CVPC            | 121             | Count vertical position counter  | DTAB            | 110             | Display tab                                      |
| C1              | 208             | C1   | DISTAB          | 103,205         | Display tab                                      |
| DB              | 311             | Transmit timing  | DTn             | 112             | Display time n, n = 0 thru 3                     |
| DBR             | 311             | Transmit timing  | ECB             | 310             | Encoded control bit                              |
| DCCLR           | 317             | DC Clear – Data Comm Clear   | ECn             | 310             | Encoded bit n, n = 1 thru 6                      |
| DCT4            | 113             | Display cursor T4  | ECnS            | 312             | Encoded bit n storage, n = 1 thru 6              |
| DCURB           | 121             | Display cursor buffered  | EDEL            | 109             | End delete                                       |
| DCUR            | 121             | Display cursor   | EDP             | 203             | End display pulse                                |
| DD              | 311             | Receive timing   | EEV             | 231             | Translator bit E – Even                          |
| DDAV            | 310             | Data available   | ELG1            | 312             | Encoded least significant group 1                |
|                 |                 |  | ELn             | 312             | Encoded least significant bit n, n = 1 thru 5, 7 |
|                 |                 |  | EMCY            | 116             | Edit memory cycle                                |
|                 |                 |  | EMR             | 109             | Edit memory request                              |
|                 |                 |  | EMn             | 312             | Encoded most significant bit n, n = 2, 3, 6, 7   |
|                 |                 |  | ENBIT3          | 316             | Enable bit 3                                     |
|                 |                 |  | ENDBCC          | 212             | End block check character                        |
|                 |                 |  | ENDBLK          | 316             | End block  |
|                 |                 |  | ENDPRTB1        | 331             | End print  |
|                 |                 |  | ENDSS           | 322             | End special select                               |
|                 |                 |  | ENQD            | 313             | Enquiry Decoded                                  |
|                 |                 |  | ENQME           | 316             | Enable question mark                             |
|                 |                 |  | EOC             | 229             | End-of-character                                 |
|                 |                 |  | EOD             | 231             | Translator bit E – Odd                           |
|                 |                 |  | EOL             | 121             | End-of-line                                      |
|                 |                 |  | EOTD            | 313             | EOT decoded                                      |
|                 |                 |  | EOTEA           | 315             | EOT enable                                       |

Introduction and Operation

## GLOSSARY OF TERMS (CONTD.)

| <u>Mnemonic</u> | <u>Page No.</u> | <u>Definition</u>   | <u>Mnemonic</u> | <u>Page No.</u> | <u>Definition</u>                         |
|-----------------|-----------------|---|-----------------|-----------------|---|
| ERR             | 317             | Error   | IBS             | 114             | Input backspace                           |
| ES              | 109             | End-of-screen   | ICC             | 223             | Insert count cursor                       |
| ESC             | 216             | Escape  | ICDL            | 114             | Input character delete                    |
| ESCD            | 313             | Escape decoded  | ICMA            | 224             | Insert count memory address               |
| ESCP            | 114             | Escape present  | IDGI            | 220             | Input data gate inhibit                   |
| ESCR            | 331             | Escape request  | IDNS            | 223             | Insert and delete normal strobe           |
| ESCS            | 121             | Escape set  | IDn             | 114             | Input data bit n, n = 1 thru 6            |
| ESCSTB          | 223             | Escape strobe   | IES             | 228             | Inhibit end-of-screen                     |
| ESNL            | 214             | End-of-screen or new line                                   | IFTB            | 312             | Input fixed tab                           |
| ESUD            | 110             | End-of-screen update decode                                 | IFT8            | 220             | Inhibit fixed tab 8                       |
| ETXD            | 313             | ETX decoded   | IHT             | 107             | Inhibit tab                               |
| FDC             | 107             | First display cycle   | IINS            | 114             | Input insert                              |
| FES             | 115             | Form end-of-screen  | ILDL            | 114             | Input line delete                         |
| FEV             | 231             | Translator bit F - Even                                     | ILDn            | 114             | Input load delayed n, n = 0 thru 3        |
| FFD             | 313             | Form feed decoded   | INHn            | 128             | Inhibit n, n = 1 thru 6                   |
| FFEED           | 329             | Form feed   | INL             | 312             | Input NL                                  |
| FFSPMA          | 316             | Form feed set PMA   | INSI            | 108             | Insert inhibit                            |
| FLB             | 121             | Flyback   | IOBn            | 213             | Input output buffer bit n, n = 1 thru 7   |
| FOD             | 231             | Translator bit F - Odd                                      | IOC             | 327             | Input output clear                        |
| FORMS           | 107             | Compose forms switch (input to Forms option)                | IOCLK           | 211             | Input output register clock               |
| FRMn            | 112             | Frame n, n = 0 thru 3                                       | IODC            | 312             | Input output decode                       |
| FSC12           | 107             | Forms count 1 to 2  | IOOn            | 211             | Input output register bit n, n = 1 thru 7 |
| FSMI            | 107             | Forms mode inhibit  | IRQ             | 318             | Input request                             |
| FT              | 216             | Fixed tab   | IRQDIS          | 313             | Input request disable                     |
| GEMCY           | 224             | Generate edit memory cycle                                  | IRS             | 107             | Input record separator                    |
| GENES           | 316             | Generate end-of-screen                                      | ISP             | 312             | Input space                               |
| GES             | 224             | Generate end-of-screen                                      | ITDL            | 114             | Input tab delete                          |
| GESC            | 109             | Generate escape   | ITIN            | 114             | Input tab insert                          |
| GET             | 116             | Get control data  | ITOF            | 114             | Input top-of-form (Home)                  |
| GETn            | 113             | Get screen n, n = 0 thru 3                                  | IUS             | 107             | Input unit separator                      |
| GIZMR           | 203             | Generate initialize memory request                          | IVTB            | 312             | Input variable tab                        |
| GIZMRE          | 322             | Generate initialize memory request                          | IZMCY           | 116             | Initialize memory cycle                   |
| GRD202          | 202             | Ground from card location 202                               | IZMR            | 113             | Initialize memory request                 |
| GS              | 216             | Group separator   | IIC             | 223             | Insert 1 character                        |
| GTT3            | 217             | Get T3 time   | I1CT1           | 224             | Insert 1 character time 1                 |
| GOFRAM          | 327             | Generate 0 frame  | I1CT4           | 224             | Insert 1 character time 4                 |
| HP≤VS           | 103             | Horizontal position equal to or less than variable tab stop | I2CT1           | 224             | Insert 2 characters time 1                |
| HP=VS           | 103             | Horizontal position equals variable tab stop                | I2CT2           | 224             | Insert 2 characters time 2                |
| HPE80           | 202             | Horizontal position equals 80                               | I2CT4           | 224             | Insert 2 characters time 4                |
| HPO72           | 202             | Horizontal position over 72                                 | KBDB            | 215             | Keyboard busy                             |
| HPn             | 202             | Horizontal position register bit n, n = 1 thru 7            | KBLC            | 223             | Keyboard load complete                    |
| HP12            | 112             | Horizontal position bits 1 and 2                            | KBLCA           | 220             | Keyboard load complete buffer A           |
| IA/N            | 114             | Input alphanumeric  | KCI             | 312             | Keyboard Character Insert                 |
| IBL             | 114             | Input backline  | KRIH            | 223             | Keyboard reset inhibit                    |
|                 |                 |   | LA              | 316             | Load allow                                |
|                 |                 |   | LAST            | 322             | Last screen loaded                        |
|                 |                 |   | LBR             | 116             | Load buffer register                      |

Introduction and Operation

## GLOSSARY OF TERMS (CONTD.)

| <u>Mnemonic</u>   | <u>Page No.</u> | <u>Definition</u>                                   | <u>Mnemonic</u>   | <u>Page No.</u> | <u>Definition</u>                              |
|-------------------|-----------------|---|-------------------|-----------------|--|
| LCMA1             | 123             | Load cursor memory address 1                        | NL                | 216             | New line                                       |
| LCMA2             | 123             | Load cursor memory address 2                        | NLC               | 205             | New line character                             |
| LCR               | 108             | Load carriage return                                | NLEMR             | 108             | New line edit memory request                   |
| LCS1              | 123             | Load control storage 1                              | NLRP              | 110             | New line reset pulse                           |
| LCS2              | 123             | Load control storage 2                              | NLT               | 110             | New line transfer                              |
| LDCD              | 202             | Last data character displayed on-line               | NOESCSQ           | 211             | No escape sequence                             |
| LDCHAR            | 210             | Load character                                      | NORCV             | 211             | No receive                                     |
| LDCTRL            | 316             | Load control character                              | NSTB              | 223             | Normal strobe                                  |
| LDSC              | 215             | Load display cycle                                  | N21               | 313             | Not bit 2 and not bit 1                        |
| LDSR              | 116             | Load data storage register                          | N43               | 313             | Not bit 4 and not bit 3                        |
| LESM              | 117             | Load end-of-screen in memory                        | N76               | 313             | Not bit 7 and not bit 6                        |
| LFD               | 313             | Line feed decoded                                   | ODTBn             | 314             | Output data translator bit n, n = 1 thru 7     |
| LIIN              | 205             | Line insert   | ONE               | 321             | One  |
| LMIR              | 116             | Load memory information register                    | ORQ               | 316             | Output request                                 |
| LPOR              | 331             | Load printer output register                        | OUTCMP            | 211             | Output complete                                |
| LPULL             | 108             | Load pull   | PADn              | 333             | Print address n, n = 0 thru 3                  |
| LQM               | 112             | Last quarter of memory                              | PAGE              | 106             | Page full (paging option)                      |
| L1                | 202             | Line 1  | PAR               | 208             | Parity   |
| L25               | 202             | Line 25   | PARERR            | 213             | Parity error                                   |
| MAE               | 118             | Memory address equal                                | PARIN             | 211             | Parity insert                                  |
| MA <sub>n</sub>   | 218             | Memory address bit n, n = 1 thru 10                 | PAn               | 331             | Print address bit n, n = 1, 2                  |
| MCAn              | 125             | Memory control address n, n = 0 thru 4              | PCLK              | 331             | Print clock                                    |
| MCIP              | 122             | Memory cycle in process                             | PCRE              | 331             | Print carriage return                          |
| MCSB <sub>n</sub> | 123             | Most significant control storage bit n, n = 1, 2, 3 | PCn               | 315             | Processing condition n, n = 0 thru 3           |
| MDG <sub>n</sub>  | 119,120         | Memory data gate bit n, n = 1 thru 6                | PC1OR3            | 315             | Processing condition 1 or 3                    |
| MEMF              | 123             | Memory full   | PDB               | 310             | Printer Data Bit                               |
| MEMF+1            | 123             | Memory full +1                                      | PED               | 317             | Processor error detected                       |
| MEMT              | 336             | Memory test switch                                  | PFL               | 106             | Page full                                      |
| MIB <sub>n</sub>  | 226             | Memory input bit n, n = 1 thru 6                    | PFUL <sub>n</sub> | 214             | Page Full screen n, n = 0-3 (lights indicator) |
| MIN <sub>n</sub>  | 213             | Memory input n, n = 1 thru 6                        | PGOP              | 106             | Paging option                                  |
| MIR <sub>n</sub>  | 118             | Memory input register bit n, n = 1 thru 6           | PIP               | 109             | Push in process                                |
| MMCA              | 117             | Memory most significant count allow                 | PLFE              | 331             | Print line feed                                |
| MOVF              | 219             | Memory overflow                                     | PMACLK            | 219             | Processor memory address register clock        |
| MR                | 122             | Memory read   | PMA <sub>n</sub>  | 219             | Processor memory address bit n, n = 1 thru 10  |
| MSS               | 122             | Memory safety single shot                           | PMCY              | 116             | Processor memory cycle                         |
| MT <sub>n</sub>   | 122             | Memory time n, n = 1 thru 5                         |                   |                 |  |
| MT <sub>n</sub> E | 122             | Memory time n end, n = 1, 3, 4, 5                   |                   |                 |  |
| MVCUR             | 314             | Move cursor   |                   |                 |  |
| NAKD              | 313             | Nak decoded   |                   |                 |  |
| NCCXMT            | 210             | Data set ready transmit                             |                   |                 |  |
| NFLBT3            | 220             | Not flyback and MT3                                 |                   |                 |  |



Introduction and Operation

## GLOSSARY OF TERMS (CONTD.)

| <u>Mnemonic</u> | <u>Page No.</u> | <u>Definition</u>                      | <u>Mnemonic</u> | <u>Page No.</u> | <u>Definition</u>                                   |
|-----------------|-----------------|--|-----------------|-----------------|---|
| PMCY4           | 117             | Processor memory cycle and T4          | RCVMSG          | 315             | Receive message                                     |
| PMR             | 111             | Processor memory request               | RCVNMSG         | 315             | Receive normal message                              |
| POCa            | 221             | Power on clear a, a = A thru D         | RCVRSP          | 315             | Receive response                                    |
| POLA            | 321             | Poll allow                             | RCVRSPC         | 315             | Receive response -C-                                |
| POLE            | 319             | Poll enable                            | RCn             | 228             | Receive bit n, n = 0 thru 8                         |
| PORVD           | 335             | Printer output register void           | RCnT            | 126-128         | Receive bit n transfer, n = 0 thru 8                |
| POT             | 335             | Printer interface output transistor    | RD              | 122             | Read  |
| PRCYn           | 111             | Processor cycle n, n = 1, 2            | RDP             | 122             | Read pulse  |
| PRDA            | 329             | Print data available for memory        | RECS            | 215             | Reset encoder storage                               |
| PREIO           | 210             | Present I/O register                   | RECXMT          | 210             | Receive transmit                                    |
| PRNT            | 331             | Print                                  | RERR            | 316             | Reset error   |
| PRQ             | 331             | Print request                          | RESET           | 317             | Reset I/O   |
| PRQn            | 333             | Print request n, n = 0 thru 3          | RESPD           | 314             | Respond   |
| PSC             | 109             | Push complete - not                    | RHPC            | 121             | Reset horizontal position counter                   |
| PSC             | 220             | Push complete                          | RKBEP           | 223             | Reset Keyboard - End-of-Page                        |
| PSS             | 335             | Print special select                   | RLAA            | 313             | Reset load allow                                    |
| PSn             | 329             | Print screen n, n = 0 thru 3           | RMA             | 118             | Reset memory address                                |
| PT4             | 109             | Push T4                                | RMAAn           | 132             | Reset memory A-n, n = 1, 2                          |
| PULn            | 109             | Pull n, n = 1, 2                       | RMAAnD          | 132             | Reset memory A-n-D, n = 1, 2                        |
| PUTI            | 113             | Put inhibit                            | RMA10           | 220             | Reset memory address bit 10                         |
| PUTn            | 113             | Put control data screen n, n = 1, 2, 3 | RMA9            | 220             | Reset memory address bit 9                          |
| Qn              | 124             | Quarter n, n = 0 thru 3                | RMA90           | 112             | Reset memory address bits 9 and 10                  |
| RBLKCHK         | 210             | Reset block check                      | RMB-n           | 132             | Reset memory B-n, n = 1, 2                          |
| RCA             | 113             | Replace cycle allow                    | RMB-nD          | 132             | Reset memory B-n-D, n = 1, 2                        |
| RCLR            | 106             | Receive clear                          | RPAR            | 211             | Reset parity  |
| RCMA            | 224             | Reset cursor memory address            | RPCE            | 102             | Reposition cursor equal                             |
| RCMAP           | 224             | Reset cursor memory address pulse      | RPCTM           | 203             | Reposition cursor transfer memory address to cursor |
| RCP             | 317             | Reset character present                | RPCUR           | 101             | Reposition cursor                                   |
| RCPA            | 313             | Reset character present A              | RPCn            | 314, 318        | Reset PCn, n = 1, 2                                 |
| RCR-n           | 232             | Read current -n-, n = 1, 2             | RPIP            | 101             | Reposition in process                               |
| RCUn            | 117             | Receive cursor update n, n = 1, 2      | RPMA            | 117             | Reset processor memory address                      |
| RCVENQ          | 315             | Receive ENQ (PCO • CP)                 | RPMACK          | 117             | Reset processor memory address register clock       |

Introduction and Operation

## GLOSSARY OF TERMS (CONTD.)

| <u>Mnemonic</u> | <u>Page No.</u> | <u>Definition</u>                          | <u>Mnemonic</u> | <u>Page No.</u> | <u>Definition</u>                        |
|-----------------|-----------------|--|-----------------|-----------------|--|
| RPTMSG          | 317             | Repeat message                             | SCSn            | *               | Set control storage bit n, n = 1 thru 12 |
| RQCMP           | 111             | Request complete from memory - char in MIR | SCTM            | 110             | Set cursor to memory                     |
| RRDS            | 222             | Reset rope decode storage                  | SCTTS           | 117             | Set change tab to space                  |
| RRPS            | 101             | Reset reposition cursor storage            | SCONa           | 318,321         | Set contention a, a = A, B               |
| RS              | 216             | Record separator                           | SDCY            | 336             | Single display cycle                     |
| RSEND           | 315             | Reset send                                 | SDEN            | 117             | Set display end                          |
| RSENDA          | 318             | Reset send -A-                             | SDMCY           | 116             | Set display memory cycle                 |
| RSENDB          | 315             | Reset send -B-                             | SDP             | 203             | Start display pulse                      |
| RSOUS           | 107             | Record separator or unit separator         | SDS             | 209             | Synchronous data set                     |
| RTBS            | 105             | Reset tab storage                          | SEC             | 222             | Set end-of-character                     |
| RTIMER          | 318             | Reset timer                                | SECS            | 122             | Set end-of-character storage             |
| RVTS            | 105             | Reset variable tab storage                 | SELD            | 321             | Select decoded                           |
| RWAIT           | 318             | Reset wait                                 | SELP            | 333             | Selective print                          |
| RXMIT           | 210             | Reset xmit                                 | SELS            | 327             | Selective send                           |
| R1CT1           | 224             | Replace 1 character time 1                 | SEND            | 314             | Send                                     |
| R1CT4           | 224             | Replace 1 character time 4                 | SENDa           | 327             | Send a, a = G, K                         |
| R1I2T1          | 224             | Replace 1, insert 2 characters time 1      | SENDSS          | 322             | Set end special select                   |
| R1I2T4          | 224             | Replace 1, insert 2 characters time 4      | SENDn           | 327-324         | Send screen n, n = 0, 1, 2               |
| R2CT1           | 224             | Replace 2 characters time 1                | SEnN            | 129,230         | Translator Sense lines n, n = 1 thru 12  |
| R2CT4           | 224             | Replace 2 characters time 4                | SEQD            | 207             | Sequence decoded                         |
| S-X             | 229             | Set -X storage                             | SEQR            | 207             | Sequence received                        |
| S-Y             | 229             | Set -Y storage                             | SESC            | 121             | Set escape                               |
| S-2X            | 229             | Set -2X storage                            | SFT             | 312             | Shift                                    |
| S-2Y            | 229             | Set -2Y storage                            | SGENES          | 316             | Set generate end-of-screen               |
| S+X             | 229             | Set +X storage                             | SGETn           | 113             | Set get n, n = 0 thru 3                  |
| S+Y             | 229             | Set +Y storage                             | SHIFT           | 310             | Shift bit keyboard                       |
| S+2X            | 229             | Set +2X storage                            | SInC            | 223             | Set insert n characters, n = 1, 2        |
| S+2Y            | 229             | Set +2Y storage                            | SLA             | 318             | Set LA                                   |
| SAD2            | 320             | Set AD2                                    | SLADIS          | 319             | Set LA disable                           |
| SAn             | 130             | Sense amplifier n, n = 1 thru 6            | SLCT            | 321             | Select                                   |
| SBLR            | 220             | Set backline request                       | SLCTD           | 321             | Select decoded                           |
| SBRK            | 327             | Set break                                  | SLIO            | 210,211         | Shift left I/O                           |
| SCDEC           | 117             | Allow special character decode             | SLIOA           | 210             | Shift left I/O allow                     |
| SCHAR           | 316             | Special character                          | SLIOE           | 210             | Shift left I/O enable                    |
| SCHARD          | 313             | Special character decode                   | SMC             | 116             | Set memory cycle                         |
| SCLK            | 331             | Start bit clock                            | SMCLND          | 313             | Semi-colon decoded                       |
|                 |                 |  | SMCY            | 336             | Start memory cycle (maint. sw.)          |
|                 |                 |  | SMMC            | 116             | Set most significant memory count        |

\*103,107,214,220

## Introduction and Operation

## GLOSSARY OF TERMS (CONTD.)

| <u>Mnemonic</u>   | <u>Page No.</u> | <u>Definition</u>   | <u>Mnemonic</u> | <u>Page No.</u> | <u>Definition</u>                                  |
|-------------------|-----------------|---|-----------------|-----------------|--|
| SNLK              | 110             | Set new line character  | START           | 209             | Start  |
| SODB              | 208             | Set output data bit   | STD             | 115             | Start display                                      |
| SOF               | 209             | Sync off  | STDP            | 112             | Start display pulse                                |
| SOHD              | 313             | Start-of-header decoded   | STMCA           | 117             | Set transfer CMAR to memory                        |
| SPCn              | 315             | Set PCn, n = 1,2  | STMRA           | 316             | Start timer A                                      |
| SPC1EN            | 318             | Set PC1 enable (SELTD • RCVENQ)   | STP             | 331             | Start pulse  |
| SPC2A             | 323             | Set SPC2A   | STRTRPT         | 317             | Start retransmit                                   |
| SPMA <sub>n</sub> | 219             | Set processor memory address bit n, n = 9, 10   | STXD            | 313             | STX decoded  |
| SPMCY             | 111             | Set processor memory cycle  | STXE            | 314             | STX encoded  |
| SPREIO            | 210             | Set preset I/O  | STXEA           | 319             | STX encode allow                                   |
| SPRQ              | 331             | Set print request   | SVT             | 103             | Set variable tab                                   |
| SPULL             | 205             | Set pull  | SWAIT           | 310             | Set wait   |
| SPOC              | 122             | Special power-on clear, enabled when 'Clear Allow' switch on Control Unit is 'On' and any 'Break' key is depressed on keyboard. | SXMIT           | 210             | Set xmit   |
| SRC               | 214             | Strobe reposition cursor  | SYMD            | 222             | Symbol decode                                      |
| SRCVALM           | 318             | Set receive alarm   | SYND            | 211             | Sync decode  |
| SRM               | 318             | Start receive mode  | SYNn            | 201             | Sync pulse n, n = 1, 2                             |
| SRPC              | 318             | Shift right PC  | SZn             | 229             | Set Zn storage, n = 1, 2                           |
| SRPOR             | 329             | Shift right printer output register   | SnCC            | 203             | Screen n cursor count, n = 1, 2, 3                 |
| SRSA              | 222             | Set rope storage sense amplifier  | Sn              | 209             | Sync n, n = 2, 3                                   |
| SRXMIT            | 318             | Set re-xmit   | S2CDL           | 108             | Set 2 character delete                             |
| SRnC              | 223             | Set replace n characters, n = 1, 2  | TAB             | 216             | Tab decoded  |
| SR1I2             | 223             | Set replace 1, insert 2 characters  | TABR            | 220             | Tab reset storage de code                          |
| SSA               | 122             | Strobe sense amplifier  | TBRn            | 112             | Tab reset screen n, n = 0 thru 3                   |
| SSCDL             | 108             | Set single character delete   | TCC             | 110             | Tab count control                                  |
| SSCLK             | 322             | Special select clock  | TCE             | 209             | Transmit clear enable                              |
| SSELSND           | 327             | Set selective send  | TCFM            | 216             | Transfer cursor from memory                        |
| SSENDa            | 318,321,327     | Set send -a-, a = A, B, K   | TCMA            | 203             | Transfer cursor address to memory address register |
| SSL               | 215             | Second stroke level   | TDSR            | 109             | Transfer data storage register to buffer register  |
| SSLF              | 322             | Special select  | TDSn            | 336             | Test display screen n, n = 0 thru 3                |
| SSMA              | 121             | Set SMA   | TGNL            | 105             | Tab generate new line                              |
| SSSL              | 322             | Set special select  | TIDBR           | 109             | Transfer input data gates to buffer register       |
| SSSn              | 323             | Set special select screen n, n = 0 thru 3   | TIDD            | 109             | Transfer input data gates to data storage register |
| SSTOP             | 215             | Set stop  | TIMERA          | 310             | Timer on   |
|                   |                 |   | TIOBIO          | 210             | Transfer I/O buffer to I/O register                |
|                   |                 |   | TIOBIOA         | 318             | Transfer I/O buffer to I/O -A-                     |

Introduction and Operation

## GLOSSARY OF TERMS (CONTD.)

| <u>Mnemonic</u> | <u>Page No.</u> | <u>Definition</u>                                       | <u>Mnemonic</u> | <u>Page No.</u> | <u>Definition</u>                             |
|-----------------|-----------------|---|-----------------|-----------------|---|
| TIOBMIN         | 316             | Transfer I/O buffer to memory input                     | VTIOB6          | 213             | Variable tab memory cycle and IOB6            |
| TIOIOB          | 210             | Transfer I/O register to I/O buffer                     | VTMCY           | 105             | Variable tab memory cycle                     |
| TKBD            | 336             | Test keyboard   | WAITK           | 327             | Wait -K-                                      |
| TMIR            | 109             | Transfer memory input register to data storage register | WCR-n           | 232             | Write current -n-, n = 1, 2                   |
| TMIRPOR         | 331             | Transfer memory input register to printer output reg.   | WP              | 122             | Write pulse                                   |
| TODTIOB         | 316             | Transfer output data translator to I/O buffer           | WTP             | 122             | Write pulse                                   |
| TPAPOR          | 331             | Transfer printer address to printer output register     | Wn              | 222             | Rope write n, n = 0 thru 5                    |
| TPMCY           | 111             | Transfer processor memory cycle                         | XAMn            | 234             | X address bit n, n = 1 thru 10                |
| TRCSn           | 218             | Transfer to control storage cycle n, n = 1, 2           | XAn             | 125             | X address n, n = 1 thru 10                    |
| TRD             | 121             | Transfer rope decode                                    | XDISC           | 227             | X discharge                                   |
| TRES            | 222             | Transfer even strokes                                   | XMCn            | 133             | X memory common n, n = 1 thru 4               |
| TROS            | 222             | Transfer odd strokes                                    | XMIT            | 210             | Transmit                                      |
| TRPD            | 101             | Transfer reposition cursor data                         | XMITC           | 315             | Transmit - C- Set Xmit                        |
| TSD             | 222             | Transfer stroke decode                                  | XMSG            | 315             | Transmit message                              |
| TSTOP           | 113             | Tab stop  | XMSGRSP         | 315             | Transmit message response                     |
| TSYNIO          | 210             | Transfer sync to I/O register                           | XMTA            | 117             | Transmit allow                                |
| TVS             | 103             | Transfer variable tab storage                           | XMTBRK          | 208             | Transmit break                                |
| TVTHP           | 105             | Transfer variable tab to horizontal position reg.       | XMTSLS          | 209             | Transmit set load sync                        |
| TVTMR           | 105             | Transfer variable tab from memory register              | XMn             | 235             | X memory address line n, n = 0 thru 31        |
| TWDI            | 311             | Two wire direct interface                               | XRBC            | 212             | Transmit receive block check character        |
| TWO             | 321             | Two   | XRQRSPA         | 315             | Transmit request response -A-                 |
| US              | 216             | Unit separator  | XRQRSP4         | 315             | Transmit request response -4-                 |
| VPn             | 202             | Vertical position counter bit n, n = 1 thru 5           | YAMn            | 233             | Y address bit n, n = 1 thru 10                |
| VSn             | 104             | Variable tab storage bit n, n = 1 thru 7                | YAn             | 124             | Y address n, n = 1 thru 10                    |
| VT              | 216             | Variable tab  | YDISC           | 227             | Y discharge                                   |
| VTFL            | 105             | Variable tab function load                              | YMCn            | 133             | Y memory common n, n = 1 thru 4               |
|                 |                 |   | YMn             | 134             | Y memory line n, n = 0 thru 31                |
|                 |                 |   | Znm             | 227             | Brightness n screen m, n = 1, 2; m = 0 thru 3 |
|                 |                 |   | nFRAM           | 327-324         | n frame, n = 0 thru 3                         |
|                 |                 |   | nALS            | 112             | n allow screen, n = 0 thru 3                  |
|                 |                 |   | 10CPS           | 201             | 10 cycles per second                          |
|                 |                 |   | 2CL             | 109             | 2 character load                              |

Introduction and Operation

GLOSSARY OF TERMS (CONTD.)

| Mnemonic | Page No. | Definition                              | C.U. conn. | Mon. conn. | Key. conn. | Description  |
|----------|----------|---|------------|------------|------------|--|
| 2N1      | 313      | Bit 2 and not bit 1                     | X          | X          | X          | KYBRDn Keyboard key from keyboard                  |
| 21       | 313      | Bit 2 and bit 1                         | Y          | Y          | -          | PFULn Lights Page Full indicator                   |
| 218GRD   | 218      | Ground from card location 218           | Z          | Z          | Z          | Ground   |
| 3CLK     | 221      | 3.3333 MHz clock                        | b          | b          | b          |  |
| 3N4      | 313      | Bit 3 and not bit 4                     | c          | c          | c          |  |
| 5N76     | 313      | Bit 5 and not bits 7 and 6              | u          | u          | -          | ALSn Indicates to C.U. that Monitor n is connected |
| 765      | 313      | Bits 7 and 6 and 5                      | -          | -          | u          | +3.6VDC From monitor to Keyboard                   |
| 765N43   | 313      | Bits 7 and 6 and 5 and not bits 4 and 3 | -          | -          | a          |  |

Note: n = 0 through 3.

INTERFACE SIGNALS FROM CONTROL UNIT TO MONITOR/KEYBOARD

| C.U. conn. | Mon. conn. | Key. conn. | Description                             |
|------------|------------|------------|---|
| J3n        | J3         | J2         |   |
| A          | A          | A          | EC1n Entry character bit 1              |
| B          | B          | B          | EC2n Entry character bit 2              |
| C          | C          | C          | EC3n Entry character bit 3              |
| D          | D          | D          | EC4n Entry character bit 4              |
| E          | E          | E          | EC5n Entry character bit 5              |
| F          | F          | F          | EC6n Entry character bit 6              |
| H          | H          | H          | CBn Control bit                         |
| J          | J          | J          | nLCP Load complete to keyboard          |
| K          | K          | K          | SHIFTn Shift, from keyboard             |
| L          | L          | L          | DAVn Data available from keyboard       |
| M          | M          | M          | LOCKn Lock to keyboard                  |
| N          | N          | N          | FRAMn Frame time (nFRAM in keyboard)    |
| R          | R          | -          | RCVALRMn Lights Receive Alarm indicator |
| S          | S          | S          | BRKn Break from keyboard                |
| T          | T          | -          | RXMITn Lights Retransmit indicator      |
| V          | V          | -          | ERRn Lights Error indicator             |
| W          | W          | W          | 10CPS 'Repeat' clock to keyboard        |

COMMUNICATIONS INTERFACE

| Input Display Conn. | RS-232 Conn. | Circuit | Description                             |
|---------------------|--------------|---------|---|
| A                   | 1            | AA      | Protective ground                       |
| B                   | 2            | BA      | Transmitted Data                        |
| C                   | 3            | BB      | Received Data                           |
| D                   | 4            | CA      | Request to Send                         |
| E                   | 5            | CB      | Clear to Send                           |
| F                   | 6            | CC      | Data Set Ready                          |
| H                   | 7            | AB      | Signal Ground                           |
| J                   | 8            | CF      | Data Carrier Detector                   |
| K                   | 9            | -       | Reserved for testing                    |
| L                   | 10           | -       | Reserved for testing                    |
| M                   | 11           | RSS     | Rate Select - SA - SUPERVISORY REV DATA |
| N                   | 12           | -       | Unassigned - SB - SUPERV. SEND DATA     |
| P                   | 13           | -       | Unassigned                              |
| R                   | 14           | -       | Unassigned                              |
| S                   | 15           | DB      | Transmit Signal Timing Element          |
| T                   | 16           | -       | Spare                                   |
| U                   | 17           | DD      | Receive Signal Timing Element           |
| V                   | 18           | -       | Spare                                   |
| W                   | 19           | -       | Spare                                   |
| X                   | 20           | CD      | Data Terminal Ready                     |
| Y                   | 21           | -       | Unassigned - RS RATE SELECT             |
| Z                   | 22           | CE      | Ring Indicator (Not used)               |
| a                   | 23           | -       | Unassigned                              |
| b                   | 24           | DA      | Transmitted Timing Element (not used)   |
| c                   | 25           | -       | Unassigned                              |
| d                   | -            | -       |   |

Note: Two-wire Direct Interface uses pins B and H for circuits BA and AB respectively.

Functional Detail

GENERAL DESCRIPTION OF B9353 SYSTEM

The "Burroughs Input & Display System" (B9353) consists of a control unit, and from one to four monitors and their associated keyboards. (Fig. II-2) A read only printer is also available which, while physically cabled to the control unit, is available to all of the monitors.

All of the memory and logic is contained in the control unit, while the monitors do little more than display the data. Each monitor utilize a 17" cathode ray tube, with 2000 screen positions available for display. The screen layout is 80 characters per line, and 25 lines per screen. 68 different types of characters can be displayed, although memory can store up to 72 different types of characters.

The B9353 uses a Non Spatial Memory as opposed to a Spatial Memory. A Spatial Memory, by definition has a memory position for every screen position. A Spatial Memory therefore must be at least as large as the number of positions available on the screen. (Fig. II-1).

A Non Spatial Memory, by contrast does not have to meet this requirement. As a result, a Non Spatial Memory can in most cases accomplish the same results as a Spatial

Memory using far less actual memory. The B9353 memory consists of 1024 six bit characters. The entire memory may be associated with one monitor and keyboard, or may be divided into four equal sections with each section or group of sections associated with a particular monitor & keyboard. (Fig. II-3 shows how memory is divided with various numbers of monitors.)

Even though memory is shared by all monitors, each monitor's portion of memory is completely separate from the others. One monitor and keyboard cannot effect what another monitor is displaying. It is apparent that with a 1024 character memory, and a 2000 position screen, it is not possible to fill the screen with data. This then is one example of how a Non Spatial Memory differs from a Spatial Memory.

In a 6 bit memory, up to 64 possible character configurations can be stored. It was mentioned earlier however that the B9353 can store up to 72 possible characters, therefore, 8 of the 72 characters have a bit configuration that is identical with 8 other characters. (Fig. II-1A)

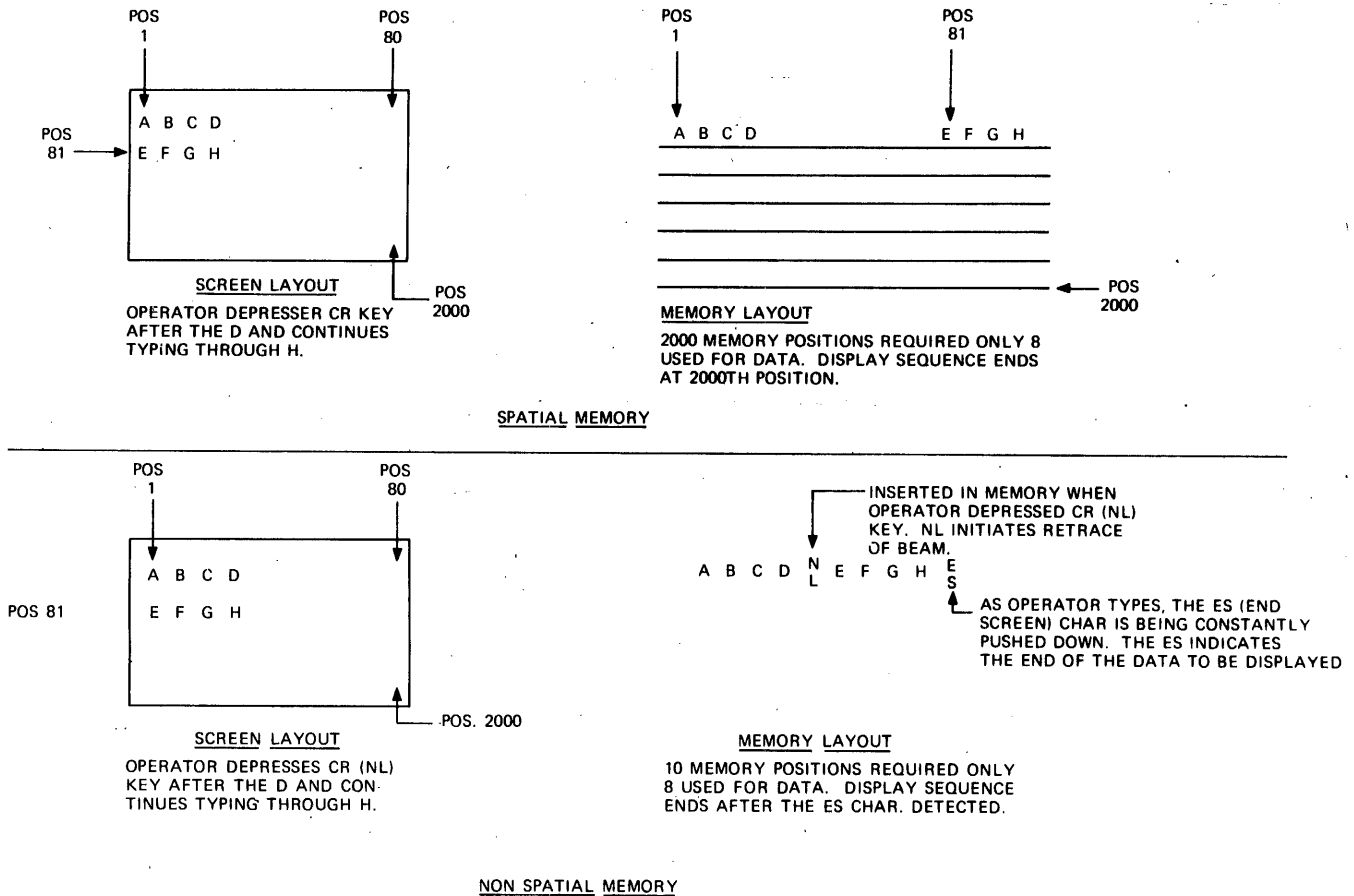


Fig. II-1

Functional Detail

To differentiate, the characters x, ], ≠, ←, ≤, &, %, \$, are written into memory with a character called ESC (Escape) preceding them. The Y goes into memory as ESC Y, thereby occupying two memory positions. When the Display sequence begins, the ESC being sensed first, will cause the ] to be displayed, instead of the Y.

EXAMPLE: Space = \$  
 ≥ = %  
 " = &  
 # = ≤  
 X = X (Multiply)  
 Y = ]  
 Z = ≠  
 [ = ←

| BIT 6 |       | BIT 5 |       | 0        | 0 | 1 | 1   |
|-------|-------|-------|-------|----------|---|---|-----|
|       |       | BIT 4 |       | 0        | 1 | 0 | 1   |
| BIT 4 | BIT 3 | BIT 2 | BIT 1 |          |   |   |     |
| 0     | 0     | 0     | 0     | SPACE-\$ | 0 | @ | P   |
| 0     | 0     | 0     | 1     | ≥ %      | 1 | A | Q   |
| 0     | 0     | 1     | 0     | " &      | 2 | B | R   |
| 0     | 0     | 1     | 1     | # ≤      | 3 | C | S   |
| 0     | 1     | 0     | 0     | ESC      | 4 | D | T   |
| 0     | 1     | 0     | 1     | Δ        | 5 | E | U   |
| 0     | 1     | 1     | 0     | ◁        | 6 | F | V   |
| 0     | 1     | 1     | 1     | ▷        | 7 | G | W   |
| 1     | 0     | 0     | 0     | (        | 8 | H | X X |
| 1     | 0     | 0     | 1     | )        | 9 | I | Y ] |
| 1     | 0     | 1     | 0     | *        | : | J | Z ≠ |
| 1     | 0     | 1     | 1     | +        | ; | K | [ ← |
| 1     | 1     | 0     | 0     | ,        | < | L | ES  |
| 1     | 1     | 0     | 1     | -        | = | M | FT  |
| 1     | 1     | 1     | 0     | .        | > | N | NL  |
| 1     | 1     | 1     | 1     | /        | ? | O | VT  |

Fig. II-1A MEMORY DECODE CHART

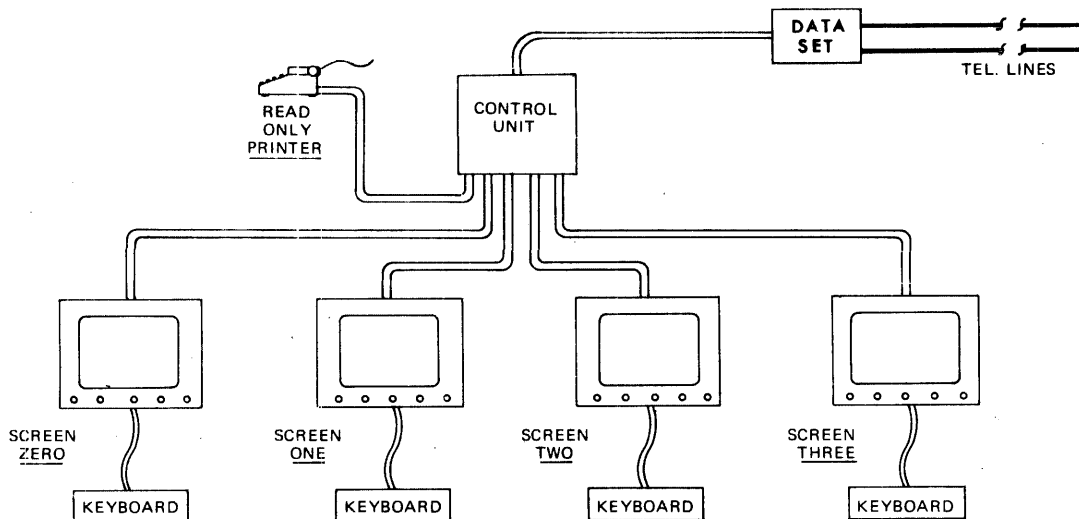


Fig. II-2 MAXIMUM B9353 SYSTEM

Functional Detail

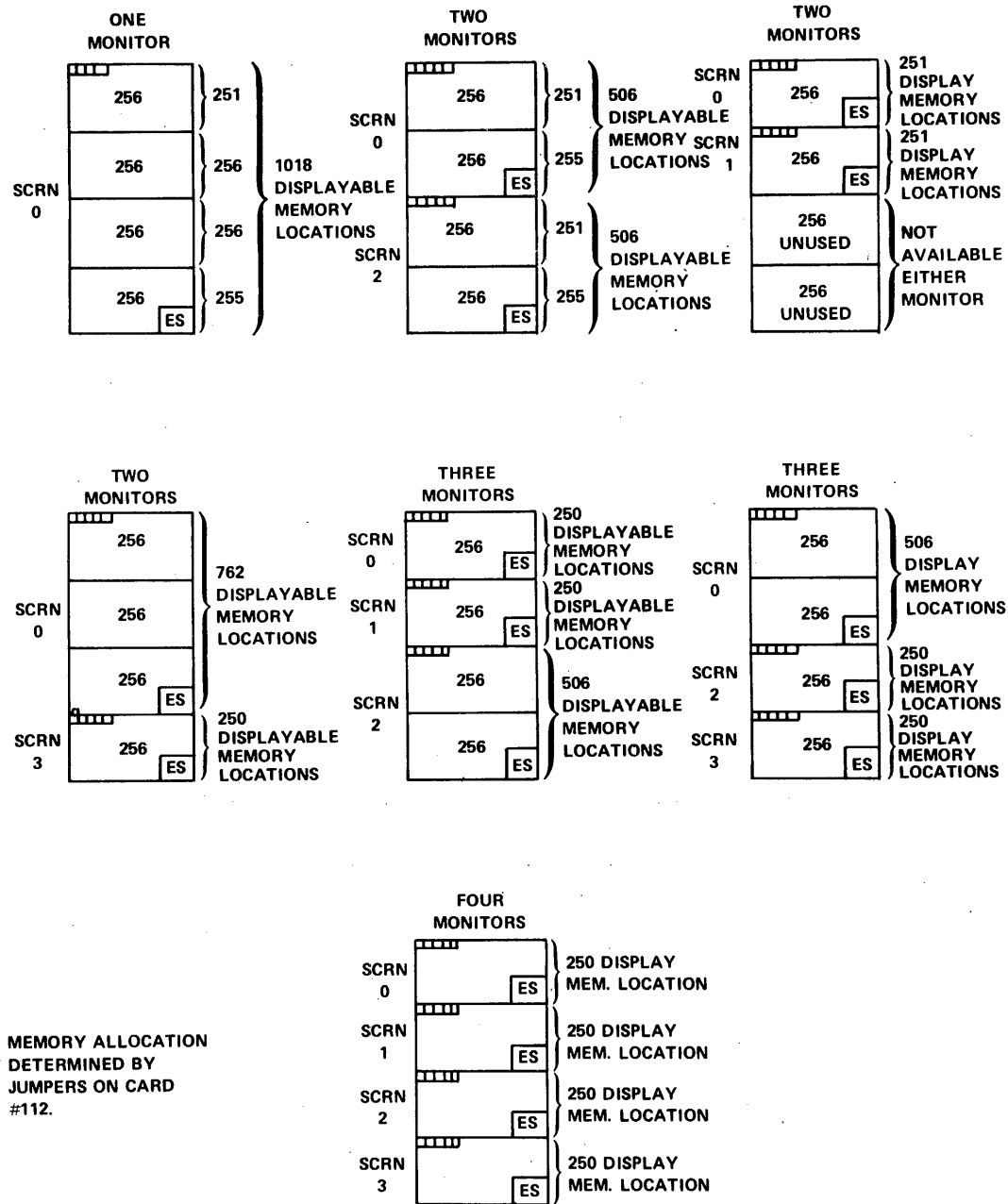


Fig. II-3 MEMORY ALLOCATION

BASIC MACHINE TIMING

GENERAL DESCRIPTION

The Control Unit, being a shared device, must through the use of timing logic set up priority as to when a monitor and associated keyboard has access to it, and for what period of time. Machine Timing determines this. Timing is referenced to the 60Hz line frequency, which, with its 16.6 millsec. repetition rate is divided into 4 equal

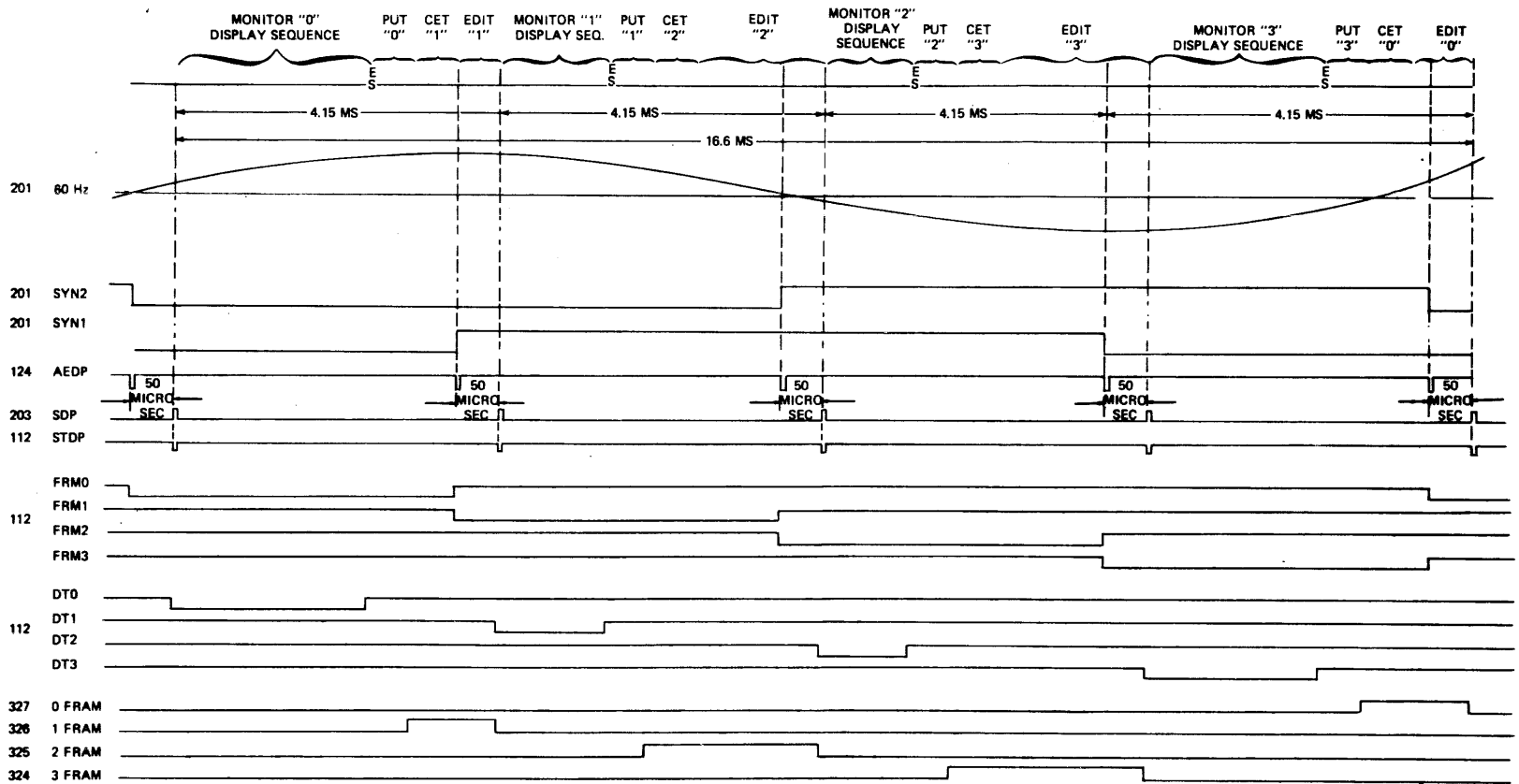
times of 4.15 millsec. each (Fig. II-4).

Assuming a 4 monitor system, and in the time periods allotted to monitor "O" the following actions take place in one of these 4.15 millsec. periods.

1. **DISPLAY SEQUENCE** — Begins at the 6th memory location allotted to monitor "O" and continues until an ES character is detected. During this time, memory is read out, and all displayable characters are stroked onto the screen.



Functional Detail



THESE LEVELS ALLOW THE VARIOUS KEYBOARDS TO SEND DATA TO THE "CONTROL UNIT"

Fig. II-4 BASIC MACHINE TIMING SHOWING SEQUENCE OF EVENTS FOR A "4" MONITOR SYSTEM

Functional Detail

2. PUT SEQUENCE – The contents of the “Cursor Memory Address” reg, the Control Storage Reg, and the Variable Tab Reg. are placed into the 5 reserved memory locations allotted to monitor “O”.
3. GET SEQUENCE – The contents of the 5 reserved memory locations allotted to monitor “1” are read out, and in the order read, placed into the “Cursor Memory Address” reg., the Control Storage Reg, and the Variable Tab Reg.
4. EDIT OR PROCESSOR SEQUENCE – New data for monitor “1”s display sequence is placed into memory. If the Edit or Processor sequence is not completed by the time STDP (Start Display Pulse) signals the start of the next display sequence, that display sequence will simply be delayed until such time as the Edit or Processor sequence is completed.

If the system should be a 2 monitor system, with memory divided equally, both monitors will have 8.3 milliseconds to complete the various sequences. The division of memory, and as a result of this, the division of the 4.15 milli time periods is a result of the jumpers shown on print 112.

DETAILED DESCRIPTION (Fig. II-4)

Basic Machine Timing

The division of the 60Hz line frequency into 4 equal time periods of 4.15 milsec. each is accomplished through the use of two differential amplifiers (print 201). The output of these amplifiers, SYN1 and SYN2, in conjunction with flip-flops on print 124 causes an AEDP (Allow End Display Pulse) to be generated every 4.15 milliseconds.

“AEDP” through gating on print 203 develops EDP (End Display Pulse) and, by delaying AEDP by 50 microseconds, SDP (Set Display Pulse) is generated. (The 50 microseconds between AEDP and SDP is reserved for the edit sequence.) “EDP” clocks a ring counter (print 112) which as it counts, generates “Frame” pulses (FRMO → FRM3). The output of the ring counter, gated with “SDP” and pre-determined “Screen allocation connectors” will set the appropriate “Display Time” flip-flops (DTO → DT3). (The state of the DTO thru DT3 flip-flops determines which “display sequence” is about to begin and which portion of memory is to be displayed.) In addition, a “Start Display Pulse” (STDP) is generated which initiates the actual Display Sequence.

Each section of memory that is allotted to a particular monitor has 6 positions that are unavailable for storing working data. The first 2 locations allotted to any monitor are for storing the cursor address, the second 2 locations are for control information, and the last location is for the variable tab address. Depending on which quarter or quarters of memory being used, the reserve addresses are:

|             |             |         |             |             |
|-------------|-------------|---------|-------------|-------------|
| 0000 → 0004 | 1st quarter | Cursor  | Control     | Variable    |
| 0256 → 0260 | 2nd quarter | Address | Information | Tab Address |
| 0512 → 0516 | 3rd quarter | X X     | X X         | X           |
| 0768 → 0772 | 4th quarter |         |             |             |

The sixth position that is unavailable for storing working data is for the “ES” (End Screen) character. The “ES” character does not occupy a fixed location in memory, but instead is shifted down as new data is placed into memory. The “ES” character is initially placed into memory on the power up sequence. The ES character, when detected during a display sequence terminates that display sequence, thereby saving time by not having the control unit cycle through the rest of memory.

The Control Unit is shared by all monitors, each monitor having use of it for a pre-determined period of time. Each monitor displays a cursor symbol □ to indicate to the operator where the next character is to be placed on the screen. The Cursor Memory Address Register in the control unit is the temporary storage area of this cursor address, when the particular monitor has use of the Control Unit. This register counts up or down as the cursor position is changed on the screen, and indicates to the Control Unit where the cursor is located with respect to memory. This register is always preset to a count of 5 upon the completion of every “PUT” sequence. Any change in cursor address would occur during the EDIT, BACKLINE, or DISPLAY sequences.

A Control Storage Register is the temporary storage area of control information in the Control Unit, and the contents of this register indicate to the Control Unit such information as:

|                                       |                            |
|---------------------------------------|----------------------------|
| Cursor over ES                        | Cursor over tab            |
| Cursor over data                      | Cursor over NL             |
| Cursor over ESC                       | Cursor before H.P. 72      |
| Forms mode alter-able and unalberable | Cursor following ESC       |
| Cursor on line 25                     | Cursor at L25 HP 80        |
| Memory full                           | Cursor less than Var. Tab. |

These conditions require certain specific functions within the Control Unit. The Control Storage Register indicates what conditions are present for that particular monitor, and the Control Unit acts accordingly. The register is cleared at the start of each “DISPLAY SEQUENCE” and new information is placed into it during the “Display Sequence”.

A Variable Tab register is the temporary storage area in the Control Unit for the address of the variable tab in a variable tab operation. As that period of time that a monitor has use of the Control Unit ends, the contents of the Cursor Memory Address reg, the Control Storage Reg, and the Variable Tab Reg are put or written into the 5

### Functional Detail

reserved locations of memory assigned to that monitor. This is known as a "PUT" memory sequence. When the period of time arrives that a monitor again has use of the control unit, a "GET" sequence is performed to get the information from the 5 reserved areas of memory for that particular monitor and place it in the various registers.

Every 16.6ms (60 Hz) the info on the monitors is rescanned or refreshed. This is accomplished by initiating a "Display Memory Sequence". There are 5 types of memory sequences. In order of priority, they are:

- |               |            |
|---------------|------------|
| 1. Initialize | 4. Edit    |
| 2. Processor  | 5. Display |
| 3. Back line  |            |

Initialize deals with the 5 reserved areas of memory allotted to each monitor Processor (Put & Get) deals with data being sent to the line or coming from the line (Data Comm).

Backline is a function of the back line operation.

Edit is placing info into memory from the keyboard.

Display is the memory sequence required to display the data every 16.6 ms. Anything that is to be displayed must be in memory. (The cursor is the exception.) A display memory sequence triggers display cycles from the 6th position allotted to that monitor (1st 5 positions reserved) until an "ES" character is detected, at which time the display memory sequence is terminated. The time of the Display Sequence depends upon the amount of memory allotted and how much of the actual memory is being used.

With four monitors, the 16.6 ms must be divided equally between all monitors. Each monitor has only 4.15 ms to display its data. This presents no problem because each monitor has only one quarter of the total memory to display. The time in between the "Display Sequences" is when new information from the keyboard or the Data Comm network is placed into memory.

### CORE MEMORY ADDRESSING

Core memory is a 6 bit, 1024 character memory. It is physically laid out on 6 planes, with a 32x32 core matrix on each plane. Memory is addressable from any of 3 registers:

1. Memory Address Register
2. Cursor Memory Address Register
3. Processor Memory Address Register

Data can be written into memory from any of 8 registers:

1. Display Register
2. Buffer Register
3. Memory Information Register
4. Memory Address Register
5. Cursor Memory Address Register
6. Processor Memory Address Register
7. Control Storage Register
8. Variable Tab Register

All outside data (keyboard or line) will be placed in memory through the use of the Display or Buffer registers. The Memory Information Register (MIR) is used for recirculating data in memory. All remaining registers are used in various internal functions. Memory is addressed through the use of Address Switching Cores, of which two types are used. (Fig. II-6) One type, called a 2x2, will select 8 of the 32 lines through memory. To handle all 32 lines requires four 2x2 switching cores. (There are 4 for the X lines and 4 for the Y lines.) Each 2x2 effectively divides the memory into quarters.

The second type of Address Switching Core is called a 4x2. Each 4x2 has 4 of the lines through memory connected to it. One line from each 2x2. Consequently to handle all 32 lines requires eight 4x2's. (There are 8 for the X lines and 8 for the Y lines.) To summarize briefly, each 2x2 selects 8 lines through memory, and each 4x2 selects 1 of the 8 lines connected to each 2x2. To select any location of memory requires selecting one 2x2 and one 4x2 for the X line, and one 2x2 and one 4x2 for the Y line. It requires two address terms (print 124 and 125) to select each Address Switching Core, or 8 address terms are required to address any location of memory.

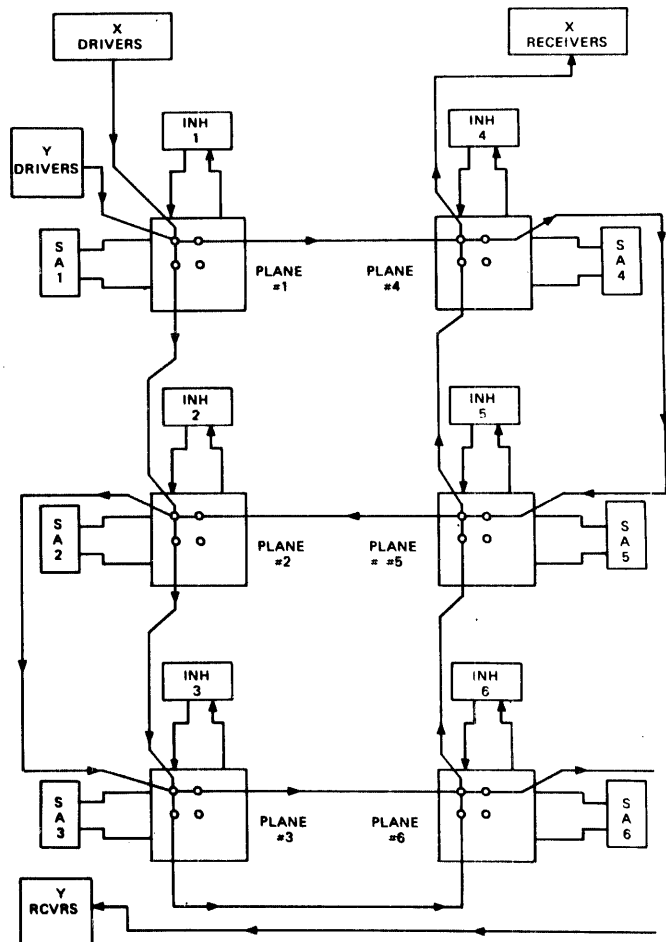


Fig. II-5 SIMPLIFIED MEMORY LAYOUT

Functional Detail

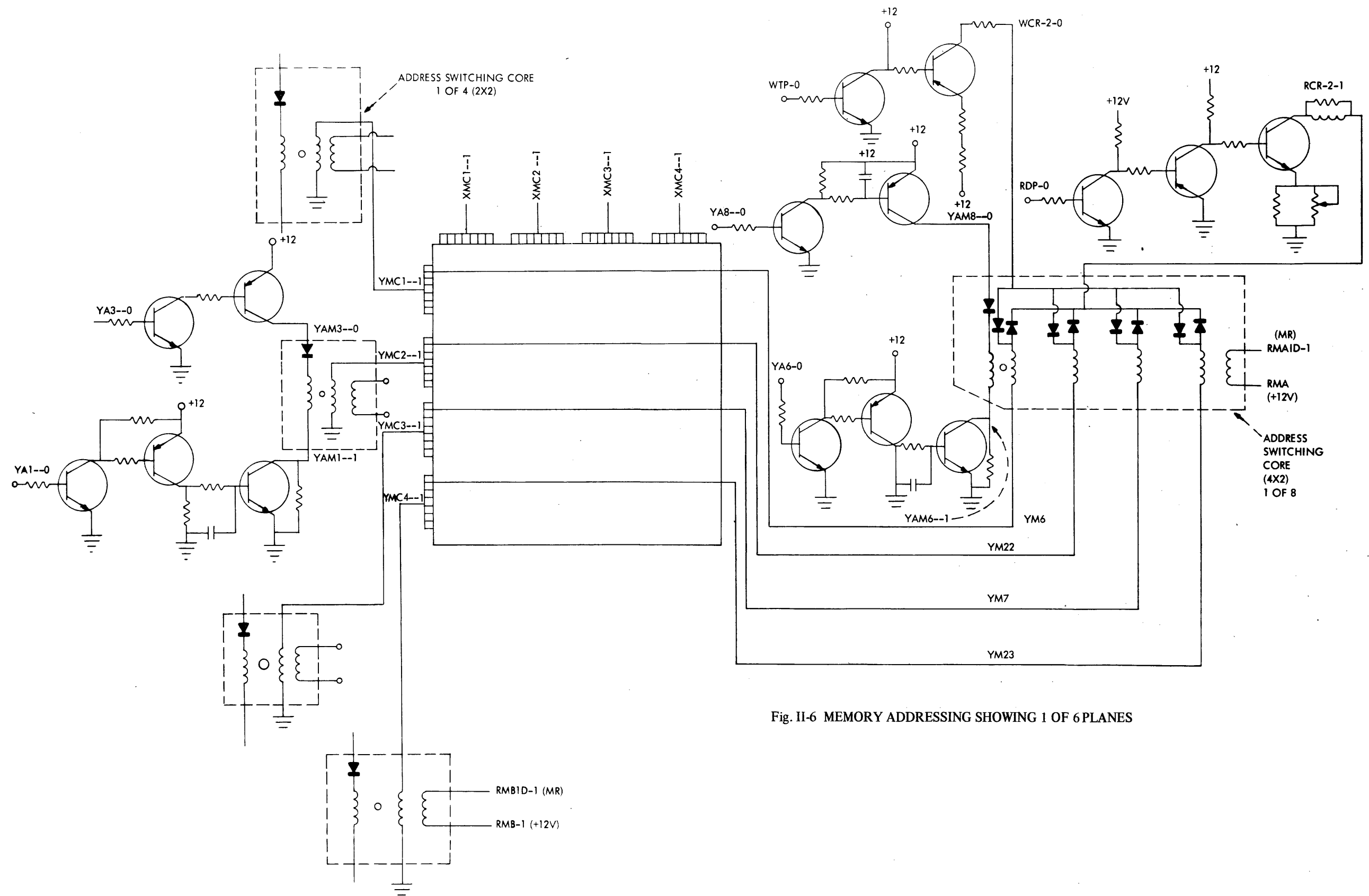


Fig. II-6 MEMORY ADDRESSING SHOWING 1 OF 6 PLANES

Functional Detail

MEMORY TIMING & OPERATION – FIG. II-6, II-7  
PRINT 122

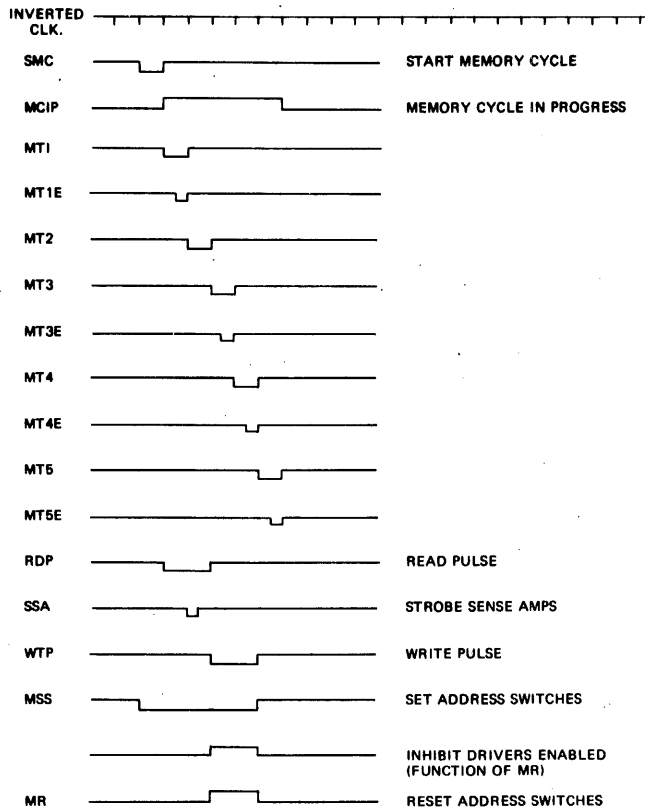


Fig. II-7 MEMORY TIMING

A memory cycle consists of 5 memory times MT1→MT5. The memory time pulses are the outputs of a shift register which is triggered by a Start Memory Cycle (SMC) pulse. The Read portion of the memory cycle occurs during MT1 & MT2, and the Write portion during MT3 & MT4.

The Address Switch Cores are selected at the same time that the read cycle begins (MT1). The selecting of the Address Switch Cores sets the cores in those 2x2's and 4x2's selected. This setting or switching of the cores, along with the read pulse that occurs at the same time, drives current through one X line and Y line, causing the contents of that memory location to be read out into the sense amps. At MT2 time (approximately the center of the read pulse) a Strobe Sense Amp (SSA) pulse occurs to strobe the sense amps into the MIR register. The Read Pulse (RDP) is felt on all 4x2's, however only one 4x2 is selected, and it requires the setting of the switch core in the 4x2 and the Read Pulse to develop enough current to read out a particular memory location.

At MT3 & MT4 the write pulse (WTP) occurs, and as with the RDP, the WTP alone cannot produce sufficient current to change the state of a memory core. It requires

the WTP along with the resetting of the previously selected Address Switch Cores. At the same time WTP occurs (MT3) a Master Reset pulse occurs (RMA1D for the 4x2's, and RMB1D for the 2x2's). This reset pulse is felt on all Address Switch cores, but only those Address Cores that were set earlier will be effected (The others are already reset). This resetting of the Address Cores along with the WTP will produce sufficient current to write info back into memory.

At the same time that the WTP occurs, the appropriate inhibit drivers are enabled by the MIB1 → MIB6 terms. Those enabled Inhibit Drivers will drive inhibit current through the planes where the memory cores are not to be set. Inhibit current cancels the effect of the Write Current. Which inhibit drivers are enabled is a direct result of what info is to be written into memory.

CHARACTER GENERATION

Each of the characters displayed on the B9353 is produced by moving the CRT beam in a series of straight lines called strokes. A character can be made up of no more than 12 strokes. The strokes can be either single or double length, and can be in any of 16 possible directions. To best illustrate the possible directions and length of any stroke refer to Fig. II-8. As shown, the amount and direction of movement is dependent upon the values of X and Y. It is also evident that because the B9353 uses only whole values of X and Y, some of the strokes are restricted to being double length strokes. Each stroke is always made with reference to Rest Position.

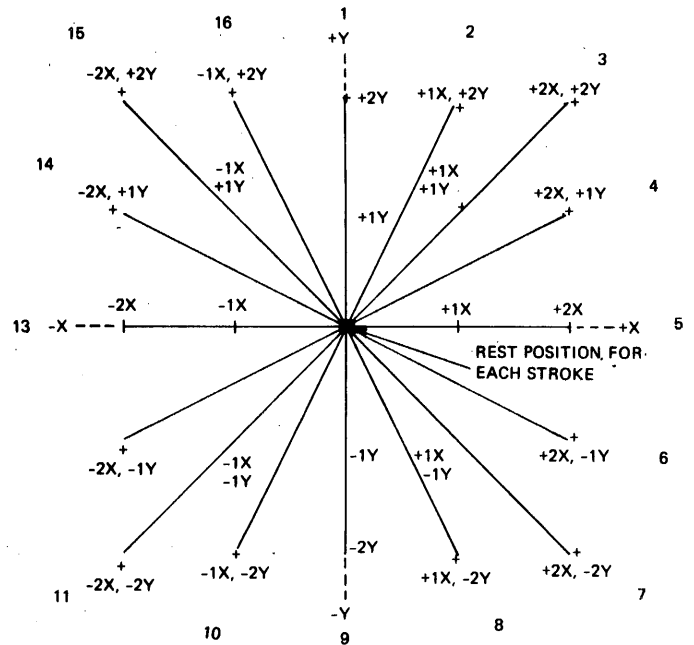


Fig. II-8 SYMBOL GENERATOR STROKE SEGMENTS

Functional Detail

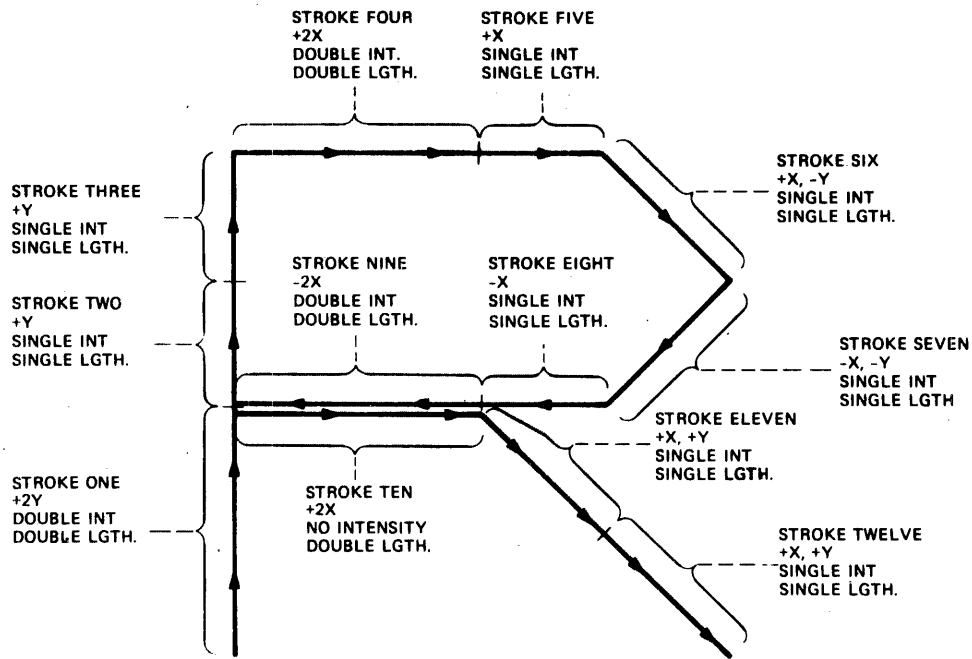


Fig. II-9 STROKE PATTERN FOR CHAR. R

The beam deflection time is the same for both single and double length strokes. As a result of this, whenever the beam is moved a double length, the beams intensity must also be doubled. There are three possible intensities: normal (single stroke), double (double stroke), and blanked (single or double stroke). Fig. II-9 shows the strokes needed to display the character R.

ROPE CHARACTER GENERATOR

The Rope Character Generator is a 12 x 6 matrix of 72 non-switching type cores. These cores differ from switching type cores in that the characteristics are similar to those of a transformer. The drive or set winding functions like the primary, and the sense winding like the secondary. An inhibit line is also used, which when energized cancels the effects of the drive line.

The core matrix is divided into the 12 possible strokes needed to develop a character. The first 6 cores of row 1 make up stroke 1. The second 6 cores of row 1 make up stroke 2. The first 6 cores of row 2 make up stroke 3, etc. Using Fig. II-11 as reference, it can be seen that the char R has a line, which is a Set or Drive line and is strung through the entire rope matrix. It is also evident that this line only passes through certain cores and bypasses others.

If rows 2,3,4,5,& 6 had inhibit current flowing through their cores, the effects of sending a pulse of current through the drive line would be cancelled, and the sense windings in these rows would feel nothing. The sense windings in row 1 however, due to the fact there was no

inhibit current, would feel an induced voltage from those cores that had the drive line passing through them. Only row 1 which consists of strokes 1 & 2 would feel the effects of the drive pulse.

Assigning a binary value of 1,2,4,8 & 16 to the first 5 cores in each stroke, it becomes evident that for stroke 1, the sense windings have a combined binary value of 26, and for stroke 2, the sense lines would have a value of 18. These sense voltages are amplified and clocked into a stroke storage register which stores the odd and even strokes for any particular row.

|               |                   |               |
|---------------|-------------------|---------------|
| 1 = -X -Y     | 13 = +X -2Y       | 25 = -X +2Y   |
| 2 = -Y        | 14 = Not Used     | 26 = +2Y      |
| 3 = -2X -2Y   | 15 = +2X -2Y      | 27 = -2X +2Y  |
| 4 = -X        | 16 = End of Char. | 28 = Not Used |
| 5 = +X -Y     | 17 = -X +Y        | 29 = +X +2Y   |
| 6 = -2X       | 18 = +Y           | 30 = Not Used |
| 7 = -2X -Y    | 19 = -2X +Y       | 31 = +2X +2Y  |
| 8 = Not Used  | 20 = +X           |               |
| 9 = -X -Y     | 21 = +X +Y        |               |
| 10 = -2Y      | 22 = +2X          |               |
| 11 = -2X -2Y  | 23 = +2X +Y       |               |
| 12 = Not Used | 24 = Not Used     |               |

Fig. II-10 BINARY VALUES OF STROKE STORAGE & RESULTING DEFLECTION

Functional Detail

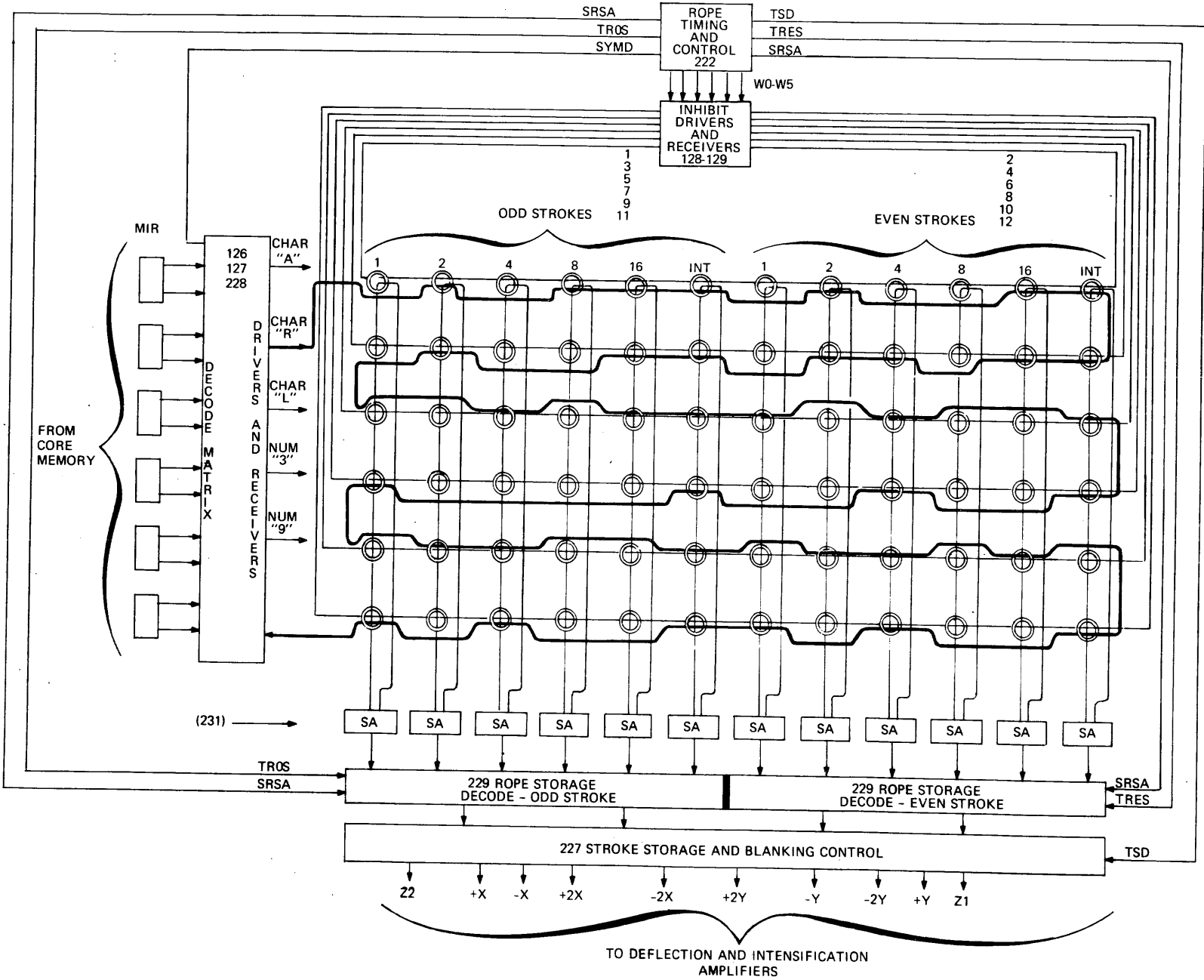


Fig. II-11 ROPE BLOCK DIAGRAM

Functional Detail

The output of the Stroke Storage register is gated through a decode matrix which converts the binary 26 for stroke 1 into +2Y and the 18 for stroke 2 into +y. (Fig. II-10.) These decode outputs are then sent to both move and intensify the beam. The char R drive line would now be pulsed a second time, however, the inhibit current would have been changed to pass through rows 1,3,4,5, & 6. Row 2, then, will now feel the effects of the drive winding and strokes 3 & 4 will be placed into the Stroke Storage Reg. to be decoded. This action continues with the inhibit current being changed and the drive winding being pulsed until the number of strokes required for the character have been completely stored & decoded.

The sixth core for each stroke indicates whether a stroke is to be intensified. If the drive winding passes through it, the stroke is to be intensified.

An End of Character (Core #5 – binary 16) is wired in for all characters having 9 strokes or less. The EOC terminates the stroke timing for that char. and allows the sequence for the next char. to be displayed to begin. If a char. consists of 4 strokes, an EOC (Binary 16) is wired into the 5th stroke position. This eliminates unnecessary timing

and allows the next char. sequence to begin earlier. If a char. consists of 10 strokes, the stroke timing is allowed to continue on as if there were 12 strokes.

DISPLAY SEQUENCE & ROPE TIMING

The Display Sequence is initiated by the SDP (Start Display pulse -203). All addressing is accomplished by the Memory Add Reg (218), which at the start of the display sequence is at a preset address of 005, 261, 517, or 773. The preset address is arrived at by clearing the MA1 through MA8 flip-flops, and placing the MA9 and MA10 flip-flops in a state determined by the SnCC levels (203). The SnCC levels are in turn a function of the particular Get sequence that has just occurred.

The display sequence also triggers the Rope Timing logic for displaying and stroking the characters, and counts the H.P. (horizontal position) and V.P. (vertical position) counters for positioning the beam horizontally from char. to char., and vertically from line to line. These counters are in a cleared state at the beginning of the Display Sequence which is the equivalent of position 1 line 1 on the screen.

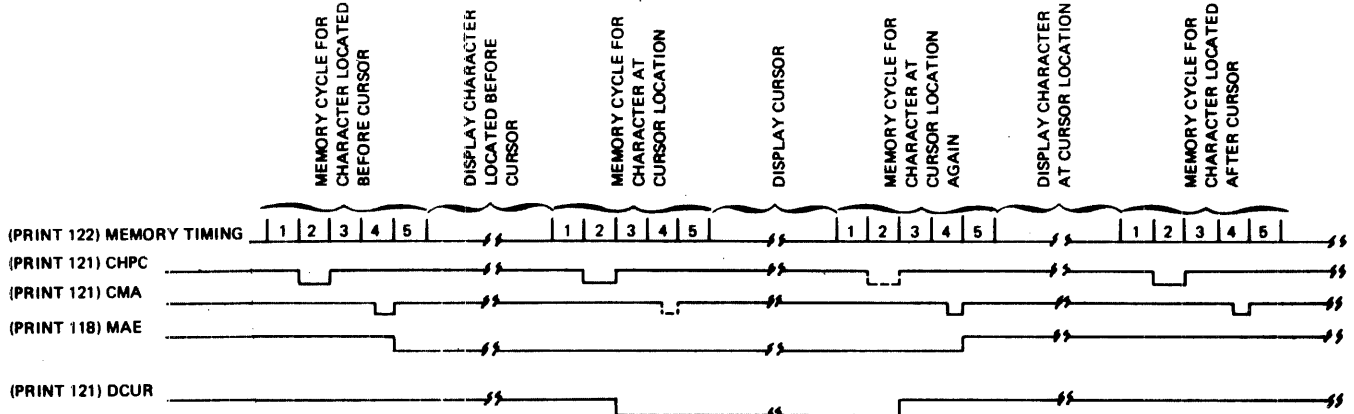


Fig. II-11A CURSOR DISPLAY TIMING



Functional Detail**DETAILED DESCRIPTION (FIG. II-12, II-13) (PRINT 222)**

The Display Sequence is initiated by STD (Start Display – a function of SDP) setting the DMR (Display Memory Request -222) FF. DMR is used on 122 to develop DDMR which sets the DMCY (116) FF and triggers the Memory Timing (122) with SMC (Start Memory Cycle). SMC also conditions the Rope Timing (222) by setting the WO FF. (See Rope Timing explanation.) Because the display cycle has begun, the output of the WO FF resets the DMR FF(222). This FF will not be set again until the Rope Timing for the present memory cycle is complete. This means that the time between the end of one cycle and the start of the next is not constant, but depends on the number of strokes in the character.

With the DMCY FF set, AMA (allow memory addressings) is developed, causing all addressing to be by the Memory Add Reg, and LMIR is developed causing everything that was read into MIR to be gated onto the inhibit lines to be written back into memory. With the display sequence inhibited, the following occurs:

MT1 & MT2 – first char read into the MIR.

**MT2**

- A. CHPC (Count Horiz Position counter -121) resets the HPEO -202 FF, allowing the H.P. counter to be counted at future MT2 times. The output of the H.P. and V.P. counters go directly to the deflection logic.

MT3 & MT4 – Contents of MIR written back into memory

MT4 CMA (121) counts the "Memory Add Reg" in preparation for the next memory cycle.

**MT5E**

Rope timing (222) triggered to begin stroking the char onto the screen at completion of stroking the char onto the screen, the DMR FF is again set to initiate another display memory cycle.

**DISPLAY OF 2 MEMORY POSITION CHAR**

If the next char read from memory is the ESC portion char the action is as follows:

SMC sets the WO FF (222) in turn resetting the DMR FF so that additional memory cycles cannot occur

MT1 & MT2 – ESC read into MIR.

MT2 A. CHPC (121) counts the H.P. counter, moving the CRT beam to the next char. position on the screen.

- B. TRD clocks the ESC into the rope decode reg (225).

MT3 & MT4 – Because LMIR (116) is present the ESC in MIR is written back into memory.

MT3 A. The ESCS FF (121) is set, disabling the CHPC logic, thereby preventing a CHPC on the next memory cycle.

- B. The DMR FF (222) is set, initiating a memory cycle for the char. portion of the 2 Memory Position char, upon completion of this cycle

MT4 CMA (121) counts the Memory Add Reg in preparation for the next memory cycle.

With the DMR FF set, as soon as this cycle is complete another cycle is immediately initiated. As before, SMC sets the WO FF, in turn resetting DMR FF.

MT1 & MT2 – Char portion of 2 memory position char read into MIR.

MT2 A. Because the ESCS FF was set earlier, a CHPC does not occur and as a result the CRT beam remains at the same char position on the screen.

- B. TRD clocks the contents of MIR into the Rope Decode Reg conditioning the DRn and RCn levels

- C. Because the ESCS FF is set, the RAE FF (228) is now set, conditioning the RC8 level. RC8 will enable the corresponding receiver CKT, which has connected to it the rope lines that are for the 2 Memory Position char. (Print 129).

MT3 & MT4 – Contents of MIR written back into memory.

MT4 A. CMA counts the Memory Add Reg in preparation for the next memory cycle.

MT5E – Rope timing (222) is triggered to begin stroking the char. onto the screen.

Upon completion of Rope Timing, which is also completion of the char. on the screen, the level SECS (222) resets the ESCS FF (121).

**CURSOR DISPLAY (FIG. II-11A)**

When the memory address register is equal to the Cursor Memory Address Register, the level MAE is generated. This level will allow the cursor character to be addressed in the rope. It will also disable the counting of the Memory Address Register and the Horizontal Position Counter. After the cursor is displayed, the character at the cursor position is displayed.

**DETAILED DESCRIPTION Memory Cycle For Character Before Cursor**

MT1 & 2 – Character is read into MIR.

MT2 A. CHPC counts the H.P. counter, moving the beam to the next character position on the screen.

- B. TRD clocks the contents of MIR into the Rope Decode Register (228).

MT3 & MT4 – Contents of MIR is written back into memory:

MT4 A. CMA counts Memory Address Register.

- B. MAE (Memory Address Equal – 118) comes true.

MT5E Rope Timing is triggered.

Functional Detail

## Memory Cycle for Cursor

MT1 & MT2 – Character is read into MIR from memory.

- MT2 A. CHPC counts the H.P. counter as before.  
 B. TRD clocks the contents of MIR into the Rope Decode Register.  
 C. With MAE true, DCUR is set. Disabling CMA and CHPC.

MT3 & MT4 – Contents of MIR is written back into memory.

MT4 Because DCVR is set CMA is disabled and the Memory Address Register is not upcounted.

MT5E Rope timing is triggered to display the cursor. Note that the driver DRO and receiver RC1 are enabled with DCUR and SYMD.

## Memory Cycle For Character At Cursor Position

MT1 & MT2 – Character is read into MIR from memory.

- MT2 A. Because DCUR is set, CHPC is disabled.  
 B. TRD clocks MIR into the Rope Decode Register.  
 C. DCUR is reset.

MT3 & MT4 – Contents of MIR is written back into memory.

MT4 With DCUR reset, CMA is enabled to count the Memory Address Register.

MT5E Rope Timing is triggered to display the character at the cursor position.

ROPE TIMING (FIG. II-13 PRINT 222)

Rope timing controls the entire character stroke sequence. The timing logic determines when the Drive or Set Line is pulsed, when and which inhibit drivers are enabled, when strobe for sense amps should occur, etc. A complete Rope Timing operation begins at the end of each display memory cycle. It is initiated by the triggering of the Rope Control (RCO-RC5) flip-flops by MT5E. These flip-flops are actually a ring counter, and once triggered will continue cycling until either an EOC has been sensed, or 12 strokes have been completed. At RCO time of each cycle a Symbol Drive pulse (SYMD) is generated to drive current through the Character Drive Line.

At RC1 time of each cycle a Strobe Rope Sense Amp pulse (SRSA) is developed to strobe the output of the sense amps (231) into the rope register (229). Also, at RC1 time of each cycle, the Write Register (WO – W5) is advanced by one. The WO-FF is set at the beginning of each memory cycle by SMC (Start Mem Cycle). This register determines which rows of the rope will have (inhibit) current. WO indicates inhibit current in row 2,3,4,5, & 6. W1 indicates inhibit current in row 1,3,4,5, & 6, etc. Only that row that has no inhibit current will feel the effects of the drive pulse on its sense line.

At RC1 and RC4 time the TRES/TROS (Transfer even or odd stroke) flip-flop has its state changed. This flip-flop determines which stroke in the “rope register” is to be gated to the decoder, the even or odd. The levels out of the decoder are the intensification and deflection signals to intensify and move the CRT beam. At RC2 and RC5 time, TSD (Transfer Stroke Decode) clocks the contents of the Rope Storage Register into the Stroke Storage Register.

The timing for any one character stroke sequence is terminated by either detecting an EOC (Binary 16) in the rope or by completing all 12 strokes. Completing 12 strokes is indicated by the setting of the L1S and L2S flip-flops. Either condition will cause the Rope Control (RCO – RC5) and the Write (WO – W5) flip-flops to be unconditionally reset, ending all rope timing. Also, the DMR (Display Memory Request) Flip-flop is set requesting a display memory cycle for the next character in memory.

FLYBACK (RETRACE)

## GENERAL DESCRIPTION (PRINT 121)

Upon detecting a NL char, or an H.P. count of 80 (HPE80), flyback is initiated. At this time the H.P. counter is reset, the V.P. counter is either counted by 1 or reset, depending on whether NL or HPE80 occurred on line 25 or not. When flyback is initiated, a delay network consisting of flip-flops FB1, FB2 and FB3 is triggered (121). During the delay, memory cycles and rope timing continue, but the delay network inhibits the memory count (CMA) and rope decode logic. As a result, memory is not advanced, thereby losing no information and even though rope timing is active, nothing is displayed.

## DETAILED DESCRIPTION – NL OR HPE80 NOT ON LINE 25

Detecting either NL or HPE80 generates CVPC (count vert. position counter -121) moving the CRT beam down 1 line. At the same time, FB1 (flyback 1) Flip-flop is set. With FB1 set, CMA is disabled, preventing the “memory add reg” from being upcounted, CHPC is disabled inhibiting counting of the H.P. counter, and RHPC (Reset H.P. Counter) clears the counter causing the CRT beam to sweep back across the screen. FB1 also generates FLB which disables the Rope Address & Decode, (228) preventing anything from being stroked onto the screen during flyback. Because the rope timing (222) is not disabled the timing continues, and when SEC (set end char storage -222), gated with FB1, sets FB2, continuing to disable CMA, CHPC, and Rope Address Decode logic. FB2 causes FB3 to be set, and if a NL initiated the flyback, the next memory cycle read NL out, and NL, gated with FB3, generates CMA. (This is necessary because the next memory cycle is initiated after flyback is complete and should be on the next memory location.)

Functional Detail

Upon completing rope timing again, SECS gated with FB3, clears all FBnFF, allowing the display sequence to continue.

**DETAILED DESCRIPTION – NL OR HPE80 ON LINE 25**  
NL or HPE80 on line 25 (L25-202) is identical to NL or HPE80 not on L25 with these differences.

1. DEND (display end) is generated (117) and used on 112 to reset the DTN (display time) Flip-flop. This results in DIP (display in progress) being dropped preventing further display memory cycles.
2. DEND conditions the set of the PUTnFF (113).
3. LESM (load ES to memory -117) generated. LESM immediately gates the ES char onto the memory inhibit lines. As a result, when the write portion of this display cycle that read out the NL char, or the char corresponding to HPE80 occurs, the ES is written into memory instead.

CONTROL STORAGE REGISTER – PRINT 115

During the display sequence, the control storage register is set to its various conditions. These conditions are:

- CSB1 Forms Alterable. Set by detecting a US character in position 1 line 1 of the screen.
- CSB2 Cursor Before Var Tab. Set when DCUR occurs, when the HP counter is at a lower count than the Var. Tab. Reg.
- CSB3 Memory Full. Set when the Memory Add. Reg. has been counted beyond that amount of memory allotted, or set with the cursor at position 80 of line 25.
- CSB4 Last Line End of Line. Set when NL or HPE80 detected on line 25.
- CSB5 Forms Unalterable. Set by detecting an RS char. in position 1, line 1 of the screen.
- CSB6 Cursor Over RS in Alterable Forms. Set when CDPOS (a function of DCUR) occurs at same time RS is in MIR in Alterable Forms mode.
- CSB7 Cursor Immediately Follows a 2 Memory Position Char. Set when MAE (Memory Add. Reg. equals Cursor Add Reg.) occurs on the memory location after the char portion of a 2 memory position char.
- CSB8 Cursor Beyond H.P. Count of 72. Set when CDPOS occurs with HPO72 (H.P. over 72 – screen position of 73 or more).
- CSB9 Cursor Over NL char. Set when CDPOS occurs with NL character in MIR.
- CSB10 Cursor Over ESC portion of a 2 Memory Position Char. Set when CDPOS occurs with the ESC configuration in MIR.

CSB11 Cursor Over FT or VT Char. Set when CDPOS with Tab (FT or VT in MIR).

CSB12 Cursor Over ES or Cursor Over RS in Alterable Forms Mode. Set when CDPOS occurs with ES in MIR. Set when CDPOS occurs with RS in MIR and unit in alterable forms mode.

**CODA** – (Cursor Over Data) true only with all of the following conditions:

CSB9 Cursor not over NL.

CSB10 Cursor not over ESC portion of a 2 memory position char.

CSB11 Cursor not over FT or VT.

CSB12 Alterable forms mode.

KEYBOARD – PRINTS 401 & 402 FIG. II-15

The keyboard is virtually an independent unit. It receives power from the control unit, but operates under its own 100KHZ clock, and develops its own unique character code which is decoded by the control unit into its own internal code. The keyboard utilizes 62 keys, with each key being attached to a microswitch. The generation of the keyboard code is by way of a continuously cycling binary counter, composed of Flip-flops CB, EC6, EC5, EC4, EC3, EC2, EC1, and an associated gating matrix. Each key has a binary value assigned to it in the counter. Each key is then connected to the counter and matrix in such a manner that when a key is depressed, and the counter arrives at the count assigned to that key, an output is produced from the matrix. This output is used to count a Sample Counter. For a keyboard character to be considered valid, the matrix must be sampled 3 times during the time that the key is depressed. The third sample will remove the 100KHZ clock to the binary counter, stopping it with the keyboard character in it. At this time, the keyboard idles until the control unit calls on it. The keyboard responds to the control unit indicating that data is available on the binary counter output lines. Until such time as the control unit accepts the character, the keyboard is locked. By requiring 3 samples from the matrix before stopping the counter, the keyboard considers only the 3rd sample as valid. Though the 1st and 2nd samples may have been valid, they might also have been in error due to mechanical considerations such as contact bounce, etc.

DETAILED DESCRIPTION – PRINTS 401 & 402

The circuit consisting of Flip-flops SC2, SC1 and gates H1-E and H2-6, compose the Sample Counter. This counter, being counted by one with each output from the code matrix will, at a count of 3, disable gate H4-I thereby removing the 100KHZ clock from the binary counter and stopping it with the keyboard character in it. Flip-flop G3-I & H is set when the Sample Counter attains a 3 count and

Functional Detail

serves to disable the input to the Sample Counter. As long as the Sample counter remains at a 3 count, the clock to the binary counter is inhibited, and the keyboard is locked.

At the keyboard time that corresponds to the keyboard number a nFRAM pulse from the control unit gates the contents of the binary register onto the output lines going to the control unit. If this is a valid character, such as the sample counter being at a 3 count, the keyboard simultaneously responds with DAV (Data Available). This signal allows the control unit to strobe the data into its keyboard storage register. The keyboard is released for further keyboard functions when LCP (load complete) is received from the control unit, and the operator releases the key. LCP unconditionally clears the sample counter and the releasing of the key by the operator allows Flip-flop G3-I and H to be reset, enabling the input to the Sample Counter again.

The output of gate H3-F, added to the output of gate H4-I, makes it mandatory that both the Sample Counter be cleared and the key released before the keyboard can be used again. (Exception being the repeat operation.)

**REPEAT KEY DEPRESSED**

The repeat operation differs from the normal keyboard operation only that Flip-flop G3-I & H is reset, not by releasing the key, but by the 10 Hz pulse output of the Repeat Circuit.

If for any reason the keyboard does not receive LCP (Load Complete) from the control unit, the Sample Counter can be cleared and the keyboard released when the key labeled Keyboard is depressed. This brings up the term KYBRD which unconditionally clears the Sample Counter.

**SHIFT KEY DEPRESSED**

The Shift Flip-flop is set when the Shift key is depressed and the gating matrix produces an output.

At the Frame time corresponding to the keyboard number, the Shift level is sent to the Control Unit where, in a decode network, it is combined with the keyboard character to produce the desired Shift character.

Any key that is depressed, with the exception of the Shift and Repeat keys, sends a 7 bit keyboard character to the control unit. In the control unit, the 7 bit keyboard character is changed into a 6 bit character to be placed into core memory by way of the ID1-ID6 lines, or into one of the specific format characters (backspace, insert, delete, etc.).

(Fig. II-14, Print 312, 114)

If the format character is to be placed in memory, it will be gated onto the ID1-ID6 lines, otherwise, the gating of the format character onto the ID1-ID6 lines is disabled.

If the keyboard character is one of the eight characters requiring two memory positions, an ESCP level is generated at this time, to be used in related logic.

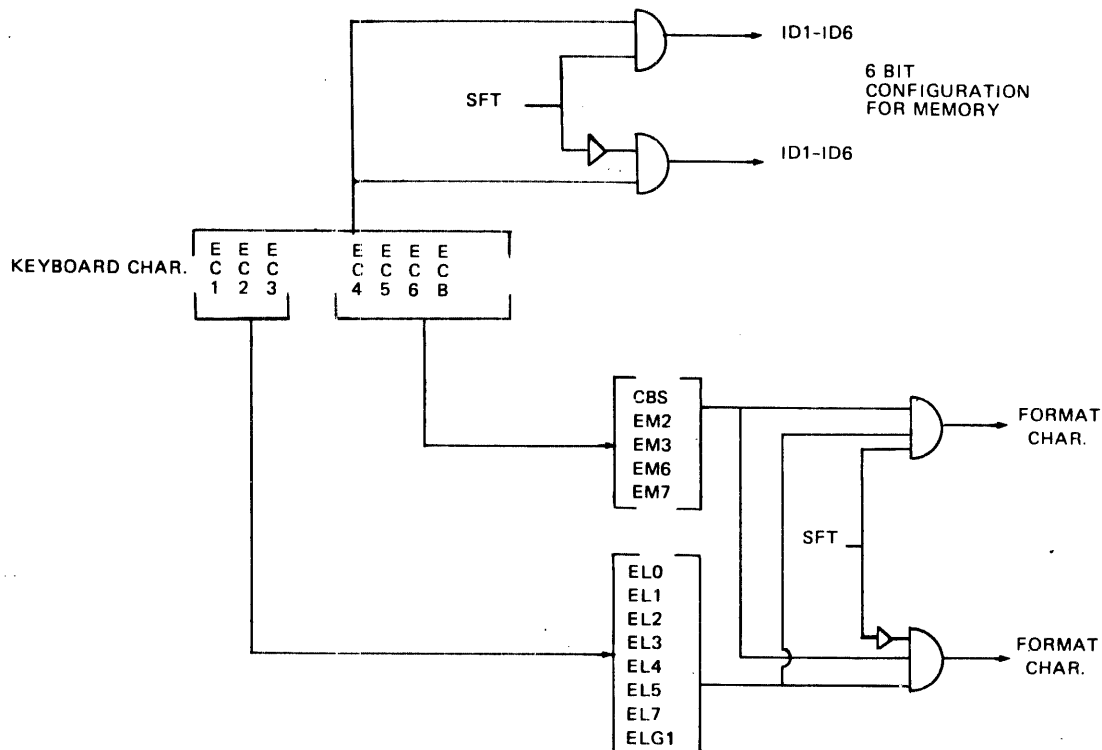


Fig. II-14

Functional Detail

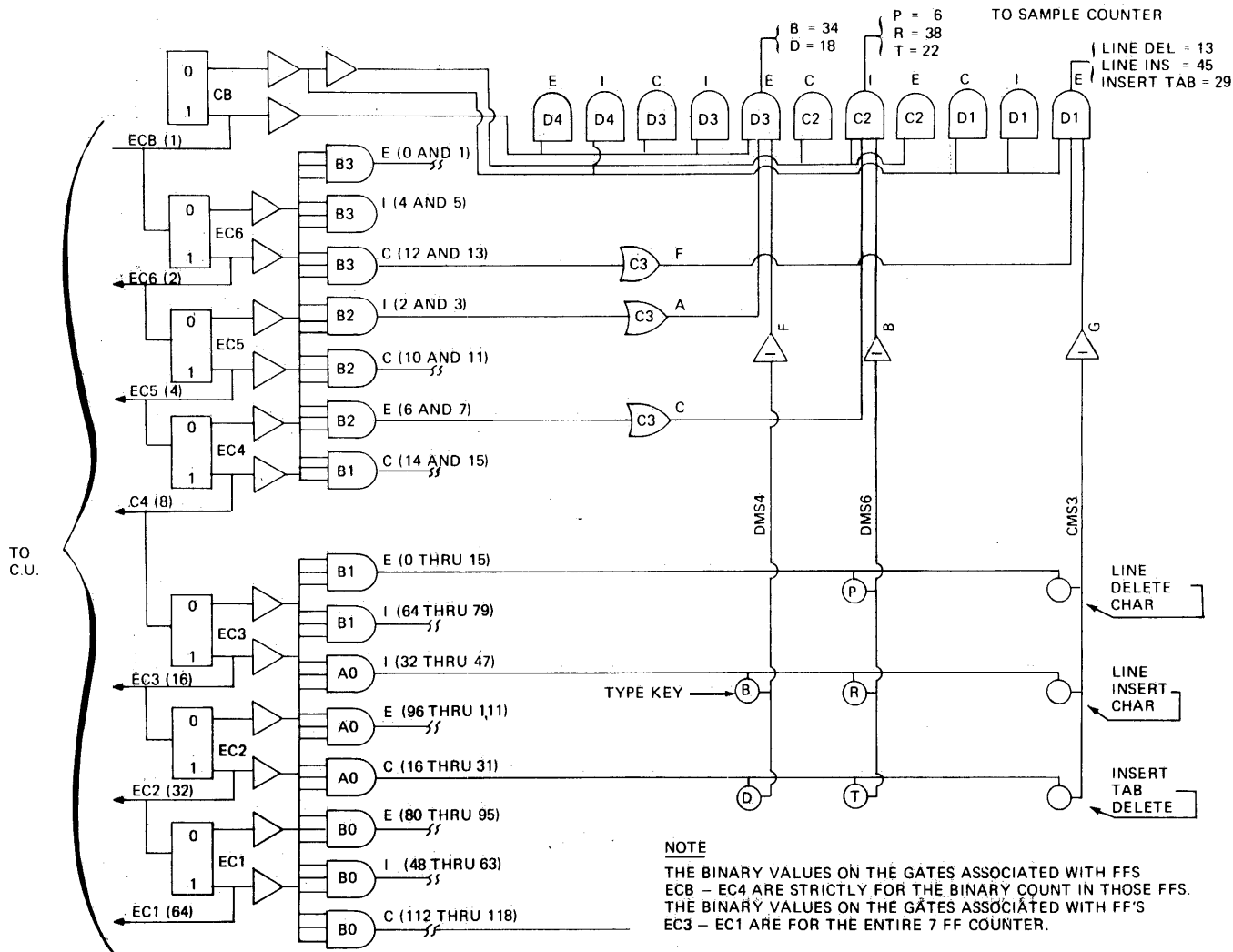


Fig. II-15 KEYBOARD CODE GENERATOR

Functional Detail

| EC3 |     |     |     | 0                      | 0            | 0                      | 0                    | 1                       | 1                   | 1           | 1          |
|-----|-----|-----|-----|------------------------|--------------|------------------------|----------------------|-------------------------|---------------------|-------------|------------|
| EC2 |     |     |     | 0                      | 0            | 1                      | 1                    | 0                       | 0                   | 1           | 1          |
| EC1 |     |     |     | 0                      | 1            | 0                      | 1                    | 0                       | 1                   | 0           | 1          |
| ECB | EC6 | EC5 | EC4 | X                      | X            | X                      | X                    | X                       | X                   | X           | X          |
| 0   | 0   | 0   | 0   | SPACE                  |              |                        |                      |                         |                     |             |            |
| 0   | 0   | 0   | 1   |                        |              |                        |                      |                         |                     |             |            |
| 0   | 0   | 1   | 0   | ERASE<br>Ø             | ≥<br>1       | "<br>2                 | #<br>3               | \$<br>4                 | %<br>5              | &<br>6      | ≤<br>7     |
| 0   | 0   | 1   | 1   | (<br>8                 | )<br>9       | *<br>:                 | +<br>;               | <<br>.                  | =<br>-              | ><br>.      | ?<br>/     |
| 0   | 1   | 0   | 0   |                        | A            | B                      | C                    | D                       | E                   | F           | G          |
| 0   | 1   | 0   | 1   | H                      | I            | J                      | K                    | L                       | M                   | N           | O          |
| 0   | 1   | 1   | 0   | @<br>P                 | Q            | R                      | S                    | T                       | U                   | V           | W          |
| 0   | 1   | 1   | 1   | X                      | Y            | Z                      | [<br>K               | x<br>L                  | ]<br>M              | ≠<br>N      | ←<br>O     |
| 1   | 0   | 1   | 0   | SEL<br>SEND<br>NORM    | RCV          | DISK                   | SEL<br>PRINT<br>NORM |                         |                     |             |            |
| 1   | 0   | 1   | 1   | LINE<br>DELETE<br>CHAR | BACK<br>LINE | LINE<br>INSERT<br>CHAR | BACK<br>SPACE        | INSERT<br>TAB<br>DELETE | Δ<br>GS             |             | Δ RS<br>US |
| 1   | 1   | 1   | 1   |                        | VTAB<br>SET  |                        |                      | CLEAR<br>HOME           | VAR<br>TAB<br>FIXED | NEW<br>LINE |            |
|     |     |     |     |                        |              |                        |                      |                         |                     |             |            |
|     |     |     |     |                        |              |                        |                      |                         |                     |             |            |
|     |     |     |     |                        |              |                        |                      |                         |                     |             |            |
|     |     |     |     |                        |              |                        |                      |                         |                     |             |            |
|     |     |     |     |                        |              |                        |                      |                         |                     |             |            |

✓ Fig. II-15A KEYBOARD DECODE CHART

Functional Detail

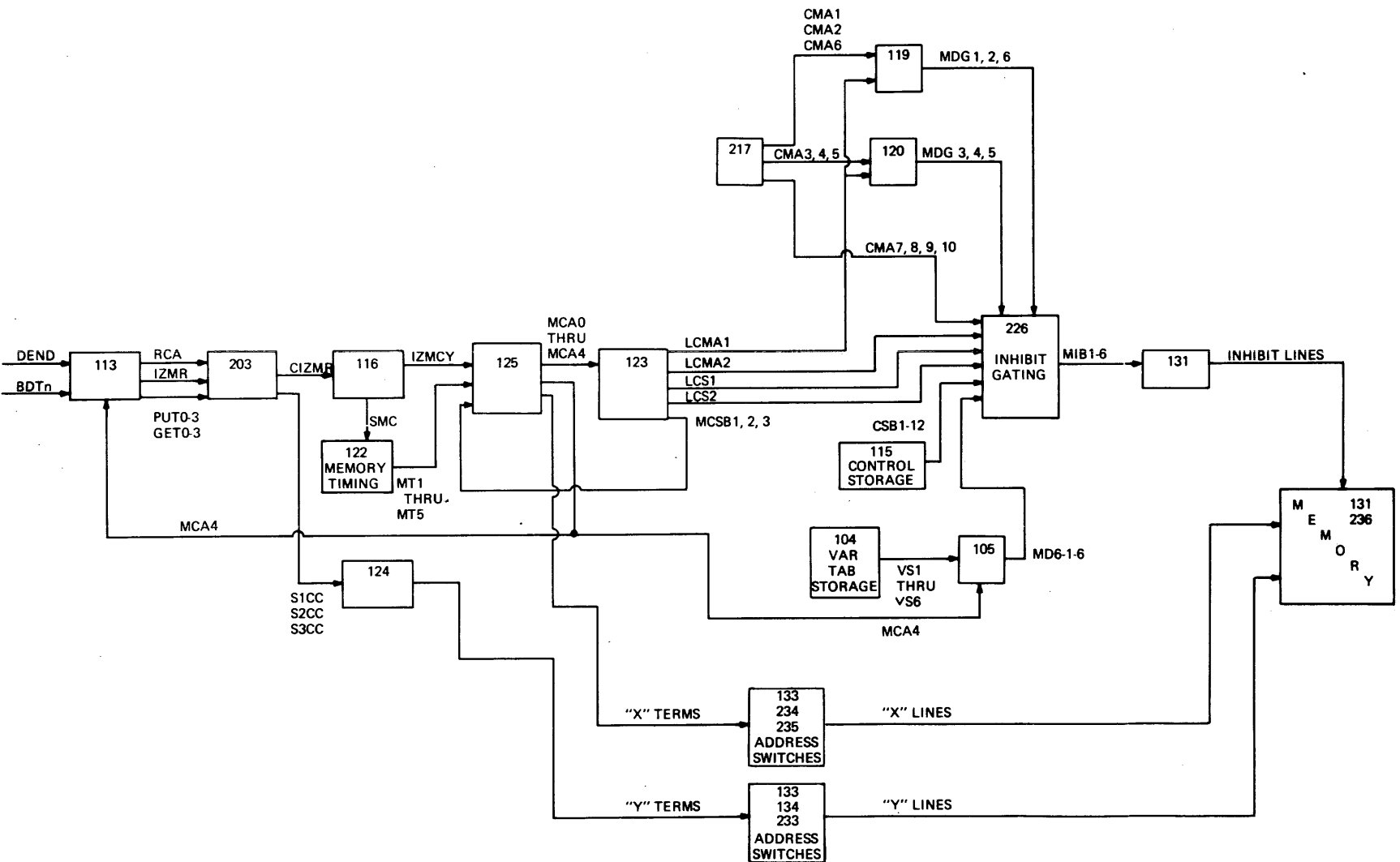


Fig. II-16 PUT SEQUENCE BLOCK DIAGRAM

Functional Detail

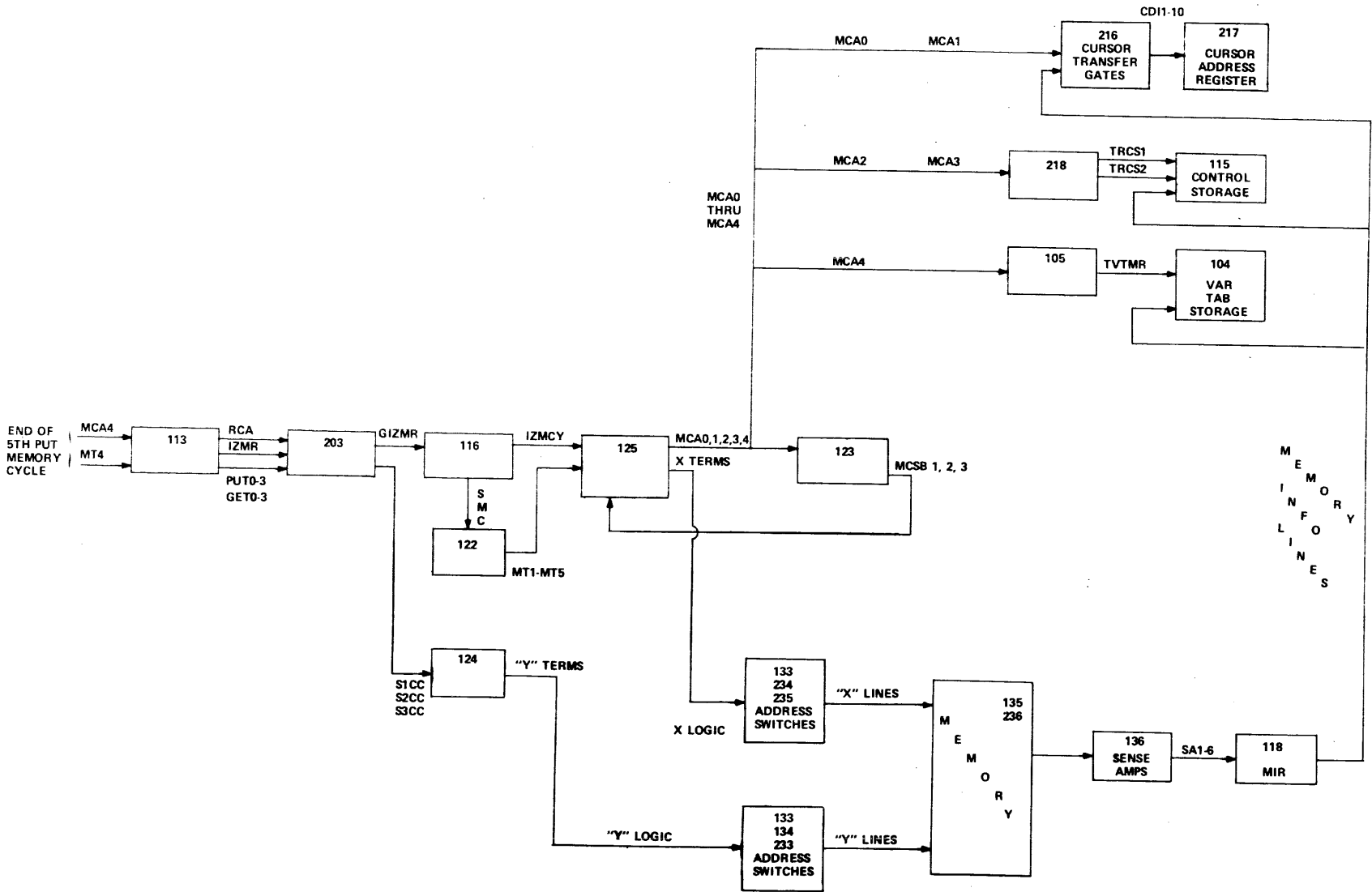


Fig. II-17 GET SEQUENCE BLOCK DIAGRAM



Functional Detail

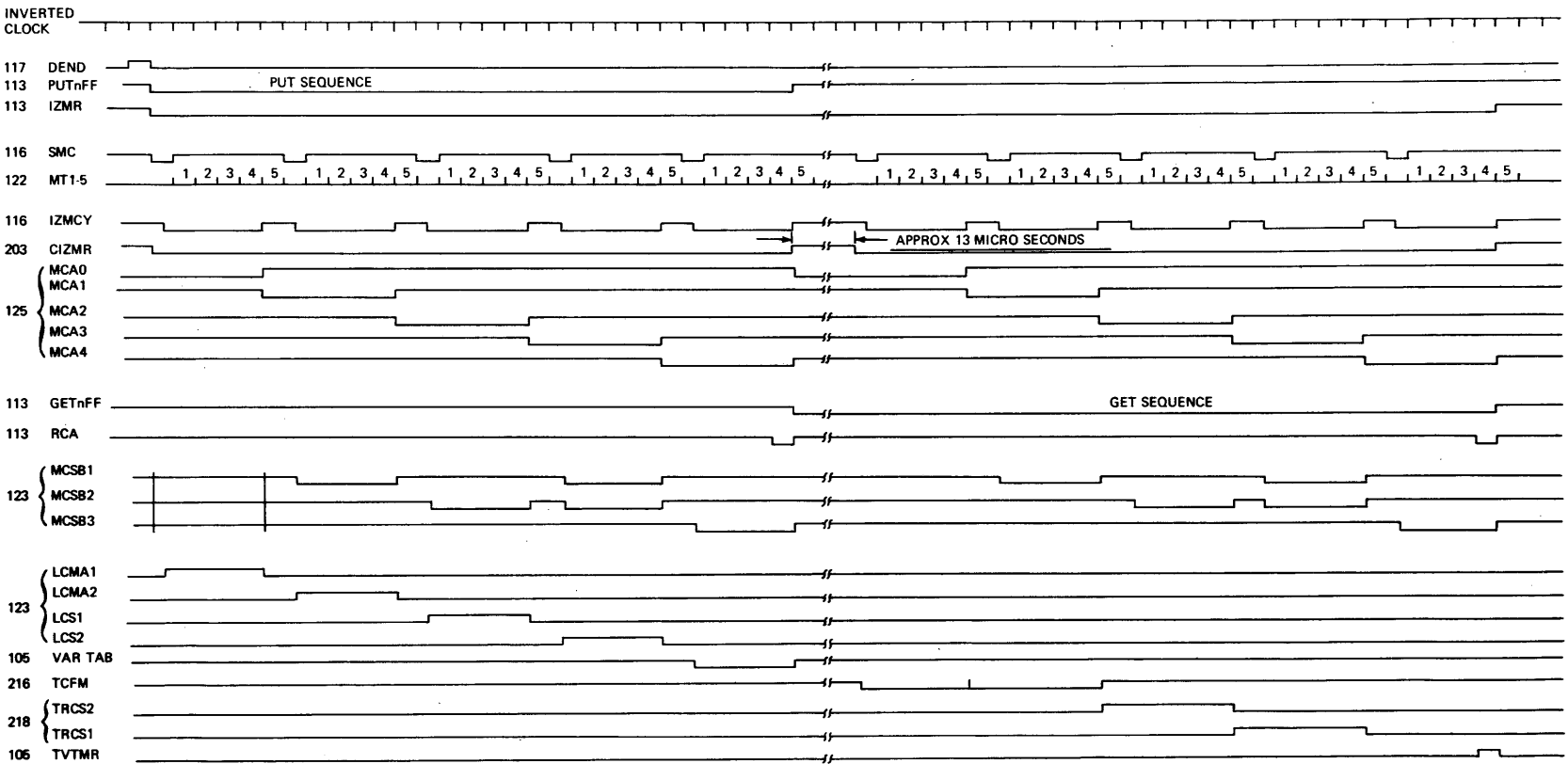


Fig. II-18 PUT GET TIMING DIAGRAM

### Functional Detail

## PUT AND GET

### GENERAL DESCRIPTION

A PUT & GET sequence is always initiated at the end of a Display sequence. The PUT sequence, which occurs first, consists of 5 memory cycles, the purpose being to Put the information contained in the Cursor Memory Address register, the Control Storage register and the Variable Tab register, into the 5 reserved locations of memory assigned to that monitor. These reserved locations of memory serve as a temporary storage area until the control unit again has need of that information.

As the Put sequence ends, the Get sequence is initiated. It also consists of 5 memory cycles, in this case, the memory cycles are to Get the Cursor Address, the Control Information, and the Variable Tab Address from the 5 reserved locations of memory assigned to the Next monitor and place this information in the appropriate registers. (This information had been Put into memory by an earlier Put sequence.)

### DETAILED DESCRIPTION — PUT SEQUENCE FIG. II-16, II-18

For purpose of explanation, assume that a Put sequence for monitor O is being initiated.

The Initialize memory sequence (Put & Get) can be initiated by:

1. Detecting an ES character during a display sequence.
2. Detecting a NL character on line 15 during a display sequence.
3. Going beyond allotted memory during a display sequence (MEMF + 1).

Any of these conditions develops the term DEND, setting the Put O Flip-flop. IZMR (Initialize Memory Request), results from setting Put o, and is true for an entire Put and Get sequence. (10 memory cycles) IZMR, in turn develops GIZMR (Generate Initialize Memory Request) setting the IZMCY FF on print 116 and initiating the memory timing. As long as GIZMR remains true, continuous memory cycles will be developed. At MT4 of every memory cycle, a counter (MCA0 thru MCA4-print 125) will be advanced. This counter indicates which of the 5 memory cycles is in progress. The output of this counter thru gating on print 123 develops MCSB1, MCSB2, & MCSB3 (Memory Char. Select). As these terms vary with the counting of MCA0 — MCA4, different X memory addressing lines are selected. As an example,

$MCA0 = \overline{MCSB1} * \overline{MCSB2} * \overline{MCSB3} = X$  address levels for location 0000. (Print 125)

$MCA1 = MCSB1 * \overline{MCSB2} * \overline{MCSB3} = X$  address levels for location 0001 etc.

The selecting of the Y address line results from the fact that with Put O Flip-flop set, the terms S1CC, S2CC, &

S3CC (Screen Character Count) are all false. These terms all being false bring up the Y address levels for monitor O (Print 124). The Put O Flip-flop remains set for the 5 Put cycles, this means the same Y address line is selected for all 5 cycles. The X lines, as noted earlier, change with each memory cycle.

The MCA0 — MCA4 counter, in addition to controlling the X address lines also controls the gating of the various registers to memory. MCA0 = LCMA1 (Load Cursor Memory Address 1) gates CMA1 — CMA6 FFS to the MDGn (Memory Data Gate) lines, and by way of the inhibit gates & drivers to memory.

MCA1 = LCMA2 gates CMA7-10 to the inhibit gates, (bypassing MDGn lines) to memory MCA2 = LCS1 (load control storage) gates CSB1-CSB6 to the inhibit gates, to memory MCA3 = LCS2, gates CSB7-CSB10 to the inhibit gates, to memory MCA4 gates VS1-VS6 (Variable Tab Storage) to the inhibit gates, to memory. The variable tab address occupies only one position of memory. The variable tab register, however, is a 7 bit register, and as a result, only VS1-VS6 can be stored in memory. Each monitor compensates for this by having a separate, hardware VS7 flip-flop assigned to it.

At MT4 of MCA4, the Put O Flip-flop is reset, the Get 1 Flip-flop set holding IZMR true, and a pulse called RCA (Replace cycle allow) generated. RCA, through a delay circuit (Print 203) causes GIZMR to go false, (preventing any further memory cycles) and approximately 13 microseconds later, go true again.

### DETAILED DESCRIPTION — GET SEQUENCE FIG. II-17, II-18

The Get sequence is identical to the Put sequence in many respects. It is initiated by GIZMR going true 13 microseconds after Put ends. GIZMR sets the IZMCY Flip-flop on 116, starting memory timing, and causing the MCA0-MCA4 counter to be counted with every MT4 pulse. As with the Put sequence, as MCA0-MCA4 is counted, the terms MCSB1, 2, & 3 change state and different X address lines are selected. The selecting of the Y address line is due to the Get 1 Flip-flop producing S1CC, S2CC & S3CC (Print 203). These terms, applied to the gating logic on print 124, cause the Y address levels for monitor #1 to be developed, in turn selecting the proper Y line.

The MCA0 — MCA4 counter controls the gating of MIR to the various registers.

MCA0 — Gate contents of MIR1 — MIR6 to CDI1 — CDI6 to CMA1 — CMA6 (Print 217)

MCA1 — gate contents of MIR1 — MIR4 to CDI1 — CDI4 to CMA7 — CMA10

Functional Detail

MCA2 = TRCS2 (Print 218), gates contents of  
MIR1 – MIR6 to CSB1-CSB6

(Print 115)

MCA3 = TRCS1 gates contents of MIR1 –  
MIR6 to CSB7-CSB12.

MCA4 = TVTMR (Print 105) gates contents of  
MIR1 – MIR6 to VS1-VS6

(Print 104)

At MT4 of MCA4 the Get 1 FF is reset, causing IZMR to go false, in turn causing GIZMR to go false ending the complete Put & Get sequence. A second RCA pulse is also generated at this time, but the use is in other operations and has no effect on Put or Get. An RCA pulse is generated at the end of each Put sequence and each Get sequence, and is used elsewhere to indicate when a Put or a Get sequence is ended.

EDIT MEMORY SEQUENCES

There are 5 basic types of Edit Memory Sequences

1. Equal Replace (Cursor not located over ES)
 

The character replacing, and the character being replaced occupy the same number of memory positions.
2. Push by 1 Replace
  - A. The ES character is replaced by a 1 memory position character. The ES is then written into the following memory location, having effectively been pushed down by one position.
  - B. A 1 memory position character is replaced by a 2 memory position character. All characters in memory following the replaced character (up to and including the ES) are then shifted or pushed down by one position.
  - C. Character insert: The character in the memory location addressed by the Cursor Add. Reg. is replaced by a space character. The character formerly in this location and everything in memory following is then shifted down by one memory location.
3. Push by 2 Replace –
 

The ES character is replaced by a 2 memory position character. The ES is then written to the right of the new character, having effectively been shifted or pushed down by 2 positions.
4. Pull by 1 Replace –
 

A 2 memory position character is replaced by a 1 memory position character. Everything in memory following the re-

placed character is then shifted back or pulled by 1 memory position.

5. Pull by 2 Replace – A 2 memory position character is deleted. The character in the memory location addressed by the Cursor Add. Reg is replaced by the character(s) in the following memory location(s), and all following characters in memory are then shifted or pulled back by 2 memory locations.

EDIT SEQUENCEEQUAL REPLACE – ONE POSITION CHAR REPLACES A ONE POSITION CHAR

Edit indicates that the action being performed is a keyboard action. In this explanation one memory cycle is being performed, and it is replacing a single memory position data character (As opposed to a NL or ES character) with a single memory position data character from the keyboard.

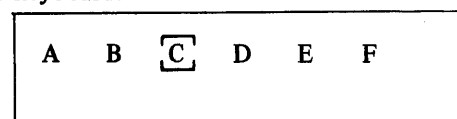


Fig. II-19

The location of the character being replaced is determined by the contents of the Cursor Address Register.

Detailed Description – Figs. II-20, II-21

At the Frame time corresponding to the keyboard, that has the key depressed, the keyboard sends DAV (Data Available) to the Control Unit (Print 215). Simultaneously, the keyboard character is gated to the Keyboard Storage Register (Print 312) in the Control Unit

It is not until the Get sequence associated with the keyboard/monitor is completed that the keyboard character is actually strobed into the Keyboard Storage Register. This is necessary, because not until the Get sequence is completed does the control unit know where the keyboard character is to be placed in memory. At this time, AKBS (Allow keyboard strobe) strobes the keyboard character into the keyboard register, LCP (Load Complete Print 215) releases the keyboard, and KBDB (Keyboard busy) prevents other keyboards from getting access to the Control Unit until this keyboard character has been placed into memory.

KBDB, in conjunction with ESCP being false (Escape – true if the keyboard char is one of the 8 char that require 2 memory locations), is used to develop NSTB (Normal strobe – not an escape operation) on print 223. NSTB develops SR1C (Set replace 1 character), setting the R1C Flip-flop on print 224. This Flip-flop is only set when a character in memory (Single or double memory position) is

Functional Detail

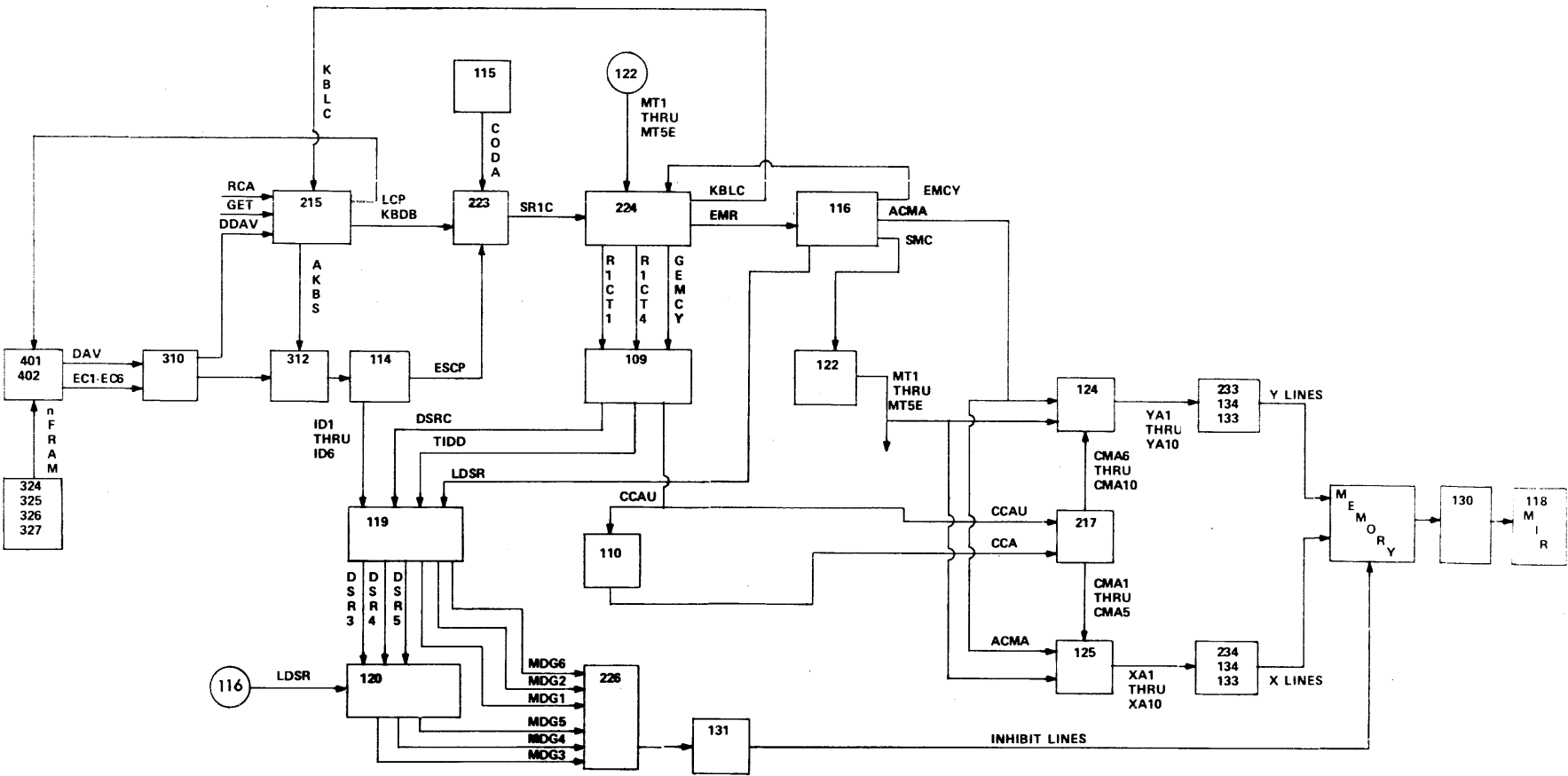


Fig. II-20 EQUAL REPLACE (ONE MEMORY POSITION REPLACE ONE MEMORY POSITION)

Functional Detail

being replaced by a Single Memory Position Character. The setting of R1CFF develops EMR (Edit memory request) initiating an EMCY (Edit memory cycle, print 116) which in turn generates ACMA (Allow Cursor Memory Address). ACMA indicates that memory addressing is determined by the cursor address register.

The keyboard character meanwhile is gated from the Keyboard Storage Register to the decode logic (Print 114), where it is gated onto the ID1-ID6 (Input decode) line in the bit configuration that is to go to memory. The following actions now take place:

1. MT1, gated with R1CFF which, on print 109, generates TIDD (Transfer ID1-ID6 to Display Reg) and DSRC (Display register clock). These term gate and clock the ID1-ID6 lines into the display register (Print 119). LDSR (Load Display Register to Memory), which is a function of EMCY, gates the display register contents onto the lines going to the Memory Inhibit Drivers & thus to memory.

2. MT4, gated with R1CFF produces R1CT4 which on print 109 & 110 generates CCA (Count cursor address) and CCAU (Count Cursor Address Up) to advance the cursor address register one count.
3. MT4 resets R1CFF, removing EMR which prevents any further memory cycle.
4. MT4E resets EMCY Flip-flop which, with EMR removed, cannot be set again.
5. MT4 generates KBLC (Keyboard Load Complete), which resets the KRBD (Keyboard Busy Flip-flop on print 215), allowing other keyboards to have access to the Control Unit.

EDIT SEQUENCE

EQUAL REPLACE — TWO POSITION CHAR REPLACES A TWO POSITION CHAR (FIG. II-22)

The two character replace two character operation is similar to the one character replace one operation, except that two memory cycles are performed. The first memory cycle deals with the ESC portion of the character and the second cycle deals with the character itself. The detailed differences are:

1. The cursor is positioned over the ESC portion of the character in memory, generating the term CSB10.
2. The keyboard character is a two memory position character, causing term ESCSTB to be true.
3. ESCSTB sets the R2CFF in turn initiating an EMCY which along with R2CFF develops terms R2CT1 and R2CT4.
4. R2CT1 generates and clocks ESC into the display reg. with the terms GESC, TIDD, and DSRC. LDSR gates the ESC character from the display register to the inhibit lines to memory. At MT3 & MT4 time the ESC portion of the character is written to memory.
5. R2CT4 sets the RP2S Flip-flop on print 109 which generates EMR to trigger a second memory cycles as soon as the first one is completed. R2CT4 also advances the cursor register by one in preparation for this second memory cycle (CCAU).
6. At MT1 of the second memory cycle, TIDD and DSRC cause the actual keyboard character to be clocked into the display register.

NOTE: During the first memory cycle, the term GESC blocked the keyboard character from getting to the Display Register. With the term GESC now removed, the keyboard character now is felt at the input to the display register.

7. At MT3 and MT4 time, the second portion of the character is written in memory.
8. At MT4 the RP2S Flip-flop is reset so additional memory cycles cannot be generated. Also, at this time, the cursor address register is counted by one in preparation for future operations. R2CFF is reset.

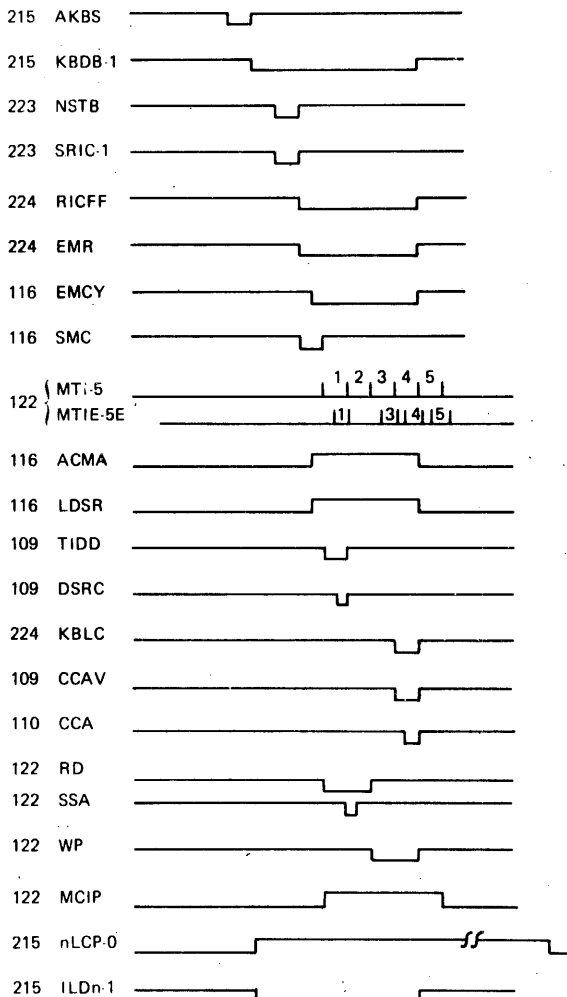
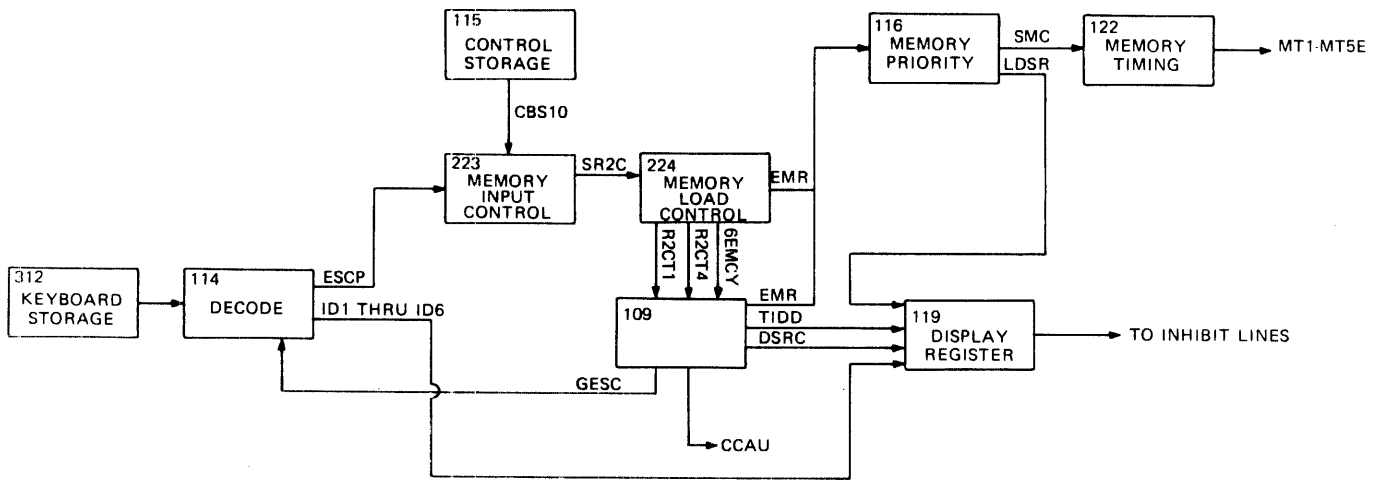


Fig. II-21 EDIT ONE CHAR REPLACE ONE CURSOR OVER DATA

Functional Detail



NOTE: THIS BLOCK DIAGRAM ONLY SHOWS THE DIFFERENCES BETWEEN THE EQUAL REPLACE (1 MEMORY POSITION CHAR) AND THE EQUAL REPLACE (2 MEMORY POSITION CHAR).

Fig. II-22 EQUAL REPLACE – (2 MEMORY POSITION CHAR. REPLACE 2)

PUSH BY ONE REPLACE – CURSOR OVER ES

GENERAL DESCRIPTION – FIG. II-23

The Push by one replace with the cursor over the ES character operates in the following manner. The keyboard character, a One Memory Position character is stored in the Display Register and two memory cycles are initiated. The first cycle addressing is done by the Cursor Address Reg. The read portion of the first cycle takes the ES character and stores it in the MIR register. The write portion writes the contents of the Display Reg (Keyboard Character) into that memory location formerly occupied by the ES character and the ES character is transferred from the MIR register to the Display Register. The read portion of the second cycle reads and discards the contents of the Display Register into this same location. Until both memory cycles are completed, no further keyboard information can be sent to the control unit from any keyboard.

DETAILED EXPLANATION (FIG. II-23, II-24, II-25 & ASSOCIATED PRINTS)

Refer to the Equal Replace explanation for that portion of the operation dealing with the transfer of information from the Keyboard to the Control Unit.

The clocking of the keyboard character into the Keyboard Storage Register causes the KBDB Flip-flop to be set, preventing other keyboards from sending data to the control unit and, in turn, developing NSTB and SI1C (Set Insert One Char). SI1C sets the I1CFF which causes EMR to be generated, initiating the first of the two memory cycles.

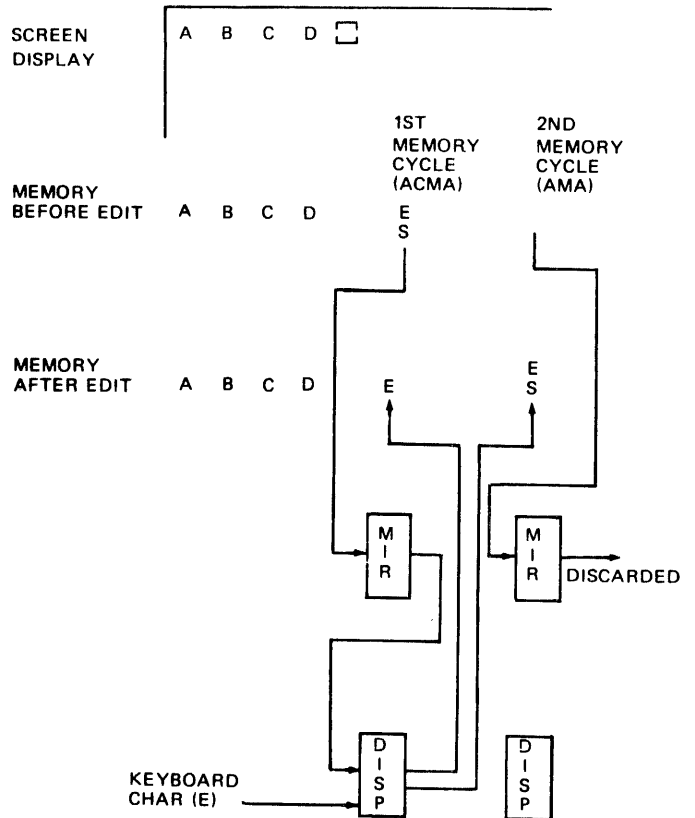


Fig. II-23

Functional Detail

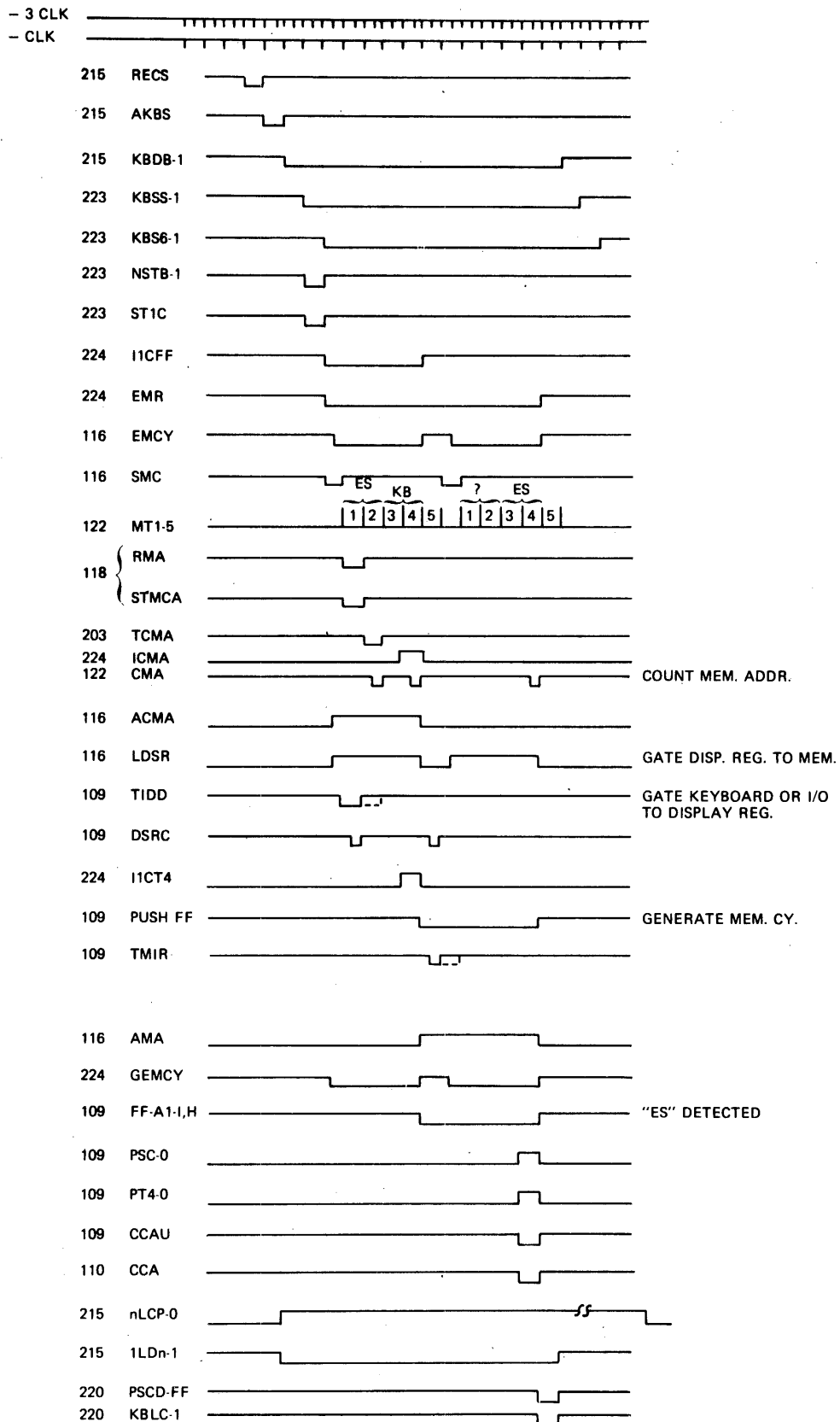


Fig. II-24 PUSH BY ONE REPLACE (CURSOR OVER ES)





Functional Detail

The following actions now take place:

1. EMCY Flip-flop is set allowing memory addressing for this first cycle to be controlled by the Cursor Add Register (ACMA).
2. At MT1 and MT2, the ES character is read into the MIR register.  
At MT1 -
  - A. The Memory Add Register is reset (RMA - 118) (I1CT1).
  - B. TIDD and DSRC gate and clock the keyboard character from the Keyboard Storage Register into the Display Register. (I1CT1 - 109-119)
  - C. LDSR gates the contents of the Display Register to the inhibit lines. (116)
3. At MT2 - Contents of Cursor Add. Register gated and clocked into the Memory Add Register. (TCMA & CMA - 203, 122)
4. At MT3 and MT4 the contents of the Display Register written into memory.  
At MT4 -
  - A. The Memory Add register is counted by one in preparation for the second memory cycle (CMA).
  - B. The Push Flip-flop is set PIP-0 (I1CT4) developing AMA to Enable Memory Add Register addressing, and dropping ACMA to disable Cursor Add Register addressing.  
Push Flip-flop also holds EMR true so a second memory cycle and will be initiated as soon as the first is completed.
  - C. Flip-flop A1 pin H is set to indicate only one memory cycle remains. (Print 109). This is due to the fact that the ES character is in the MIR register.
5. At MT5 - The ES character, which is in the MIR register is gated and clocked into the Display Reg (TMIR & DSRC) LDSR immediately gates Display Register to the inhibit lines.
6. Second memory cycle is initiated because EMR is held true. Addressing is under control of the Memory Address Register (AMA).
7. At MT1 and MT2 the contents of the following memory location are read into the MIR register. This data has no value, so it will be left in the MIR register.
8. At MT3 & MT4 the contents of the Display Register are written into memory (ES character).
9. At MT4 -
  - A. PT4 generates CMA, counting the Memory Add register up. This is redundant logic in this particular operation.
  - B. Push Flip-flop is reset, causing EMR to be dropped preventing further memory cycles.

- C. LDSR and AMA are dropped.
- D. CCAU and CCA count the Cursor Add Register up so it is positioned over the ES character.
- E. PSC - 0 sets the PSCD Flip-flop which generates KBLC (Keyboard Load Completed) KBLC then resets the KBDB (keyboard busy) Flip-flop, allowing the keyboards to again send data to the control unit.
- F. LCP Flip-flop is reset at a much later point in time by the next EDP (End Display Pulse), which occurs at the end of the next display sequence, resets the LCP Flip-flop.

PUSH BY ONE REPLACE - (ONE MEMORY POSITION CHARACTER REPLACED BY A TWO MEMORY POSITION CHARACTER.)

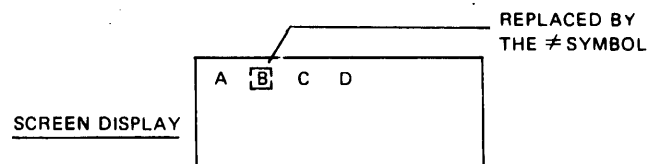


Fig. II-26

GENERAL DESCRIPTION (FIG. II-26, II-27)

This operation requires numerous memory cycles, due to the fact that everything in memory to the right of the character being replaced (up to and including the ES character) must be shifted down one position. The initial operation places the keyboard character into the keyboard storage register and causes the first of the memory cycle to be initiated. The addressing for the first two memory cycles is done by the Cursor Add Register, all further addressing is done by the Memory Add Register.

1st Memory Cycle

ESC clocked into the Display Register  
Character B read into MIR register and discarded  
ESC written into location formerly occupied by character B.

2nd Memory Cycle

Character C read into MIR register  
The ≠ symbol placed into Display Register.  
The ≠ symbol written into character C location from the Display Register  
Character C transferred from MIR to Display Register.

3rd Memory Cycle

Character D read into MIR  
Character C written into Character D location from the Display Register  
Character D transferred from MIR to Display Register.

Functional Detail

4th Memory Cycle

- Character ES read into MIR
- Character D written into ES location
- Character ES transferred from MIR to Display Register.

5th Memory Cycle

- Undetermined character read into MIR and discarded.
- ES written from Display Register into location formerly occupied by the undetermined character.

DETAILED DESCRIPTION (FIG. II-27, II-28, II-29 & ASSOCIATED PRINTS)

Because the keyboard character is a Two Memory Position character, ESCSTB (Escape Strobe) is generated. ESCSTB and CODA (Cursor Over Data) causes the R1I2CFF (replace one, insert two characters Flip-flop) to be set, in turn initiating the first of the edit memory cycles.

1ST MEM. CYCLE (CURSOR REGISTER ADDRESSING)

MT1 and MT2 — Char. B Read Into The MIR And Discarded.

MT1 — R1I2T1 —

- A. TIDD, DSRC, & GESC (Generate escape) generates, gates and clocks the ESC into the Display Register.

MT3 and MT4 — ESC Written Into Memory.

MT4 — R1I2T4 —

- A. R1I2C Flip-flop is reset
- B. I1CFF (Insert on character) is set
- C. CCAU and CCA counts cursor address register in preparation for next memory cycle

2ND MEMORY CYCLE (CURSOR REGISTER ADDRESSING)

MT1 and MT2 — Character C Read Into The MIR

MT1 — I1CT1 —

- A. TIDD and DSRC gates and clocks the keyboard char (≠) into the Display Register.
- B. TCMA (Transfer cursor to memory add) gates and clocks cursor register to memory add register.

MT3 and MT4 — Contents of Display Reg. Written Into Memory.

MT4 — I1CT4 —

- A. I1CFF is reset
- B. Push Flip-flop set generating EMR for additional memory cycles (109).
- C. PIP (push in progress) disables ACMA (Allow cursor addressing) and enable AMA (Allow memory addressing) (Print 116).
- D. ICMA causes the Memory Add Register to be counted in preparation for next memory cycle.

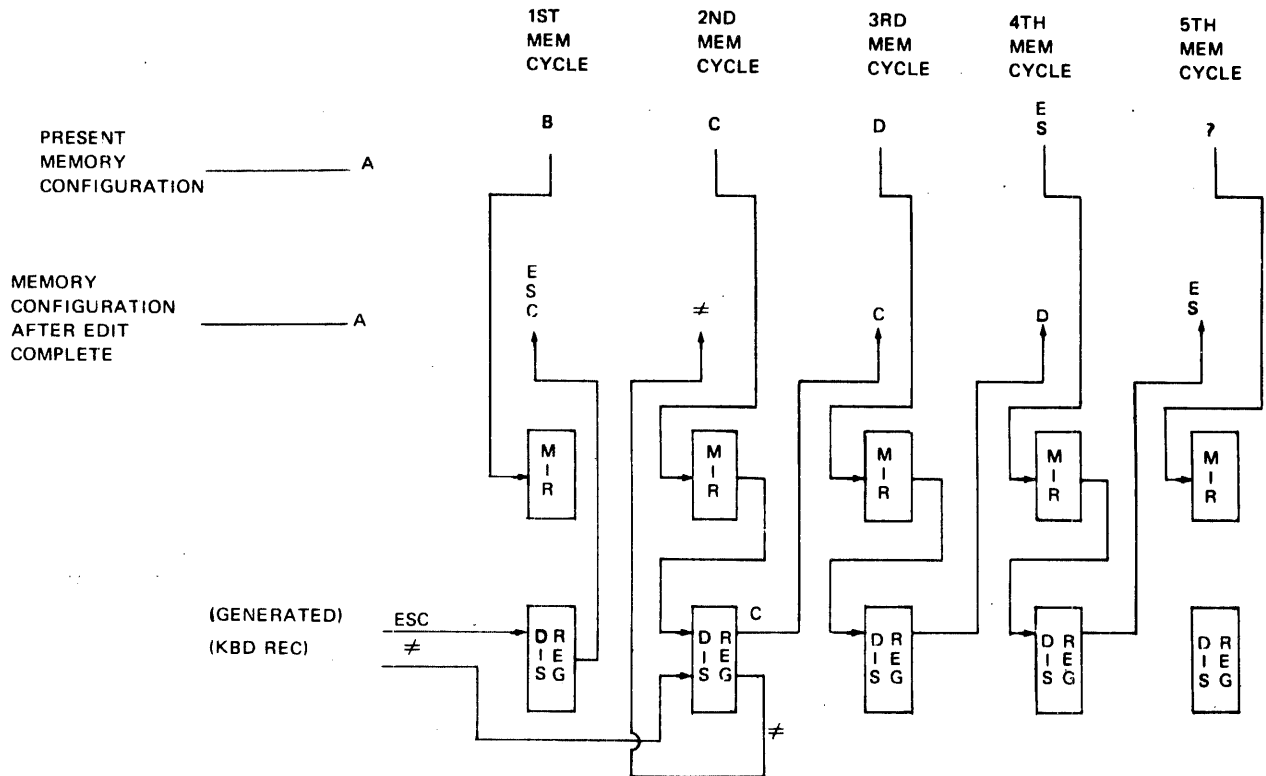


Fig. II-27

Functional Detail

MT5E – Contents of MIR (Character C) Gated and Clocked Into The Display Register By TMIR, and DSRC.

3RD MEMORY CYCLE (MEMORY REG. ADDRESSING)

MT1 and MT2 – Character D Read Into The MIR

MT3 and MT4 – Character C Written Into Memory From The Display Register

MT4 – PT4 counts Memory Add register by one

MT5E – Character D Gated and Clocked from MIR Into the Display Register

4TH MEMORY CYCLE (MEMORY REG. ADDRESSING)

MT1 and MT2 – Character ES Read Into the MIR

MT3 and MT4 – Character D Written Into Memory From the Display Register

MT4 –

- A. Flip-flop A1 pin H (109) set, indicating only

one memory cycle remains.

- B. PT4 counts Memory Add Register by one.

MT5E –

Character ES gated and clocked into the Display Register from the MIR by TMIR and DSRC.

5TH MEMORY CYCLE (MEMORY REG. ADDRESSING)

MT1 and MT2 – Undetermined Char. Read Into MIR.

MT3 and MT4 – ES Written Into Memory From the Display Register.

MT4 –

- A. Push Flip-flop reset
- B. CCAU and CCA counts the Cursor Reg. by one so it is positioned over the char. following the # symbol.
- C. PSC-0 (109) develops KBLC (220) to reset the KBDB (Keyboard busy Flip-flop), so that other keyboards can send information to the control unit.

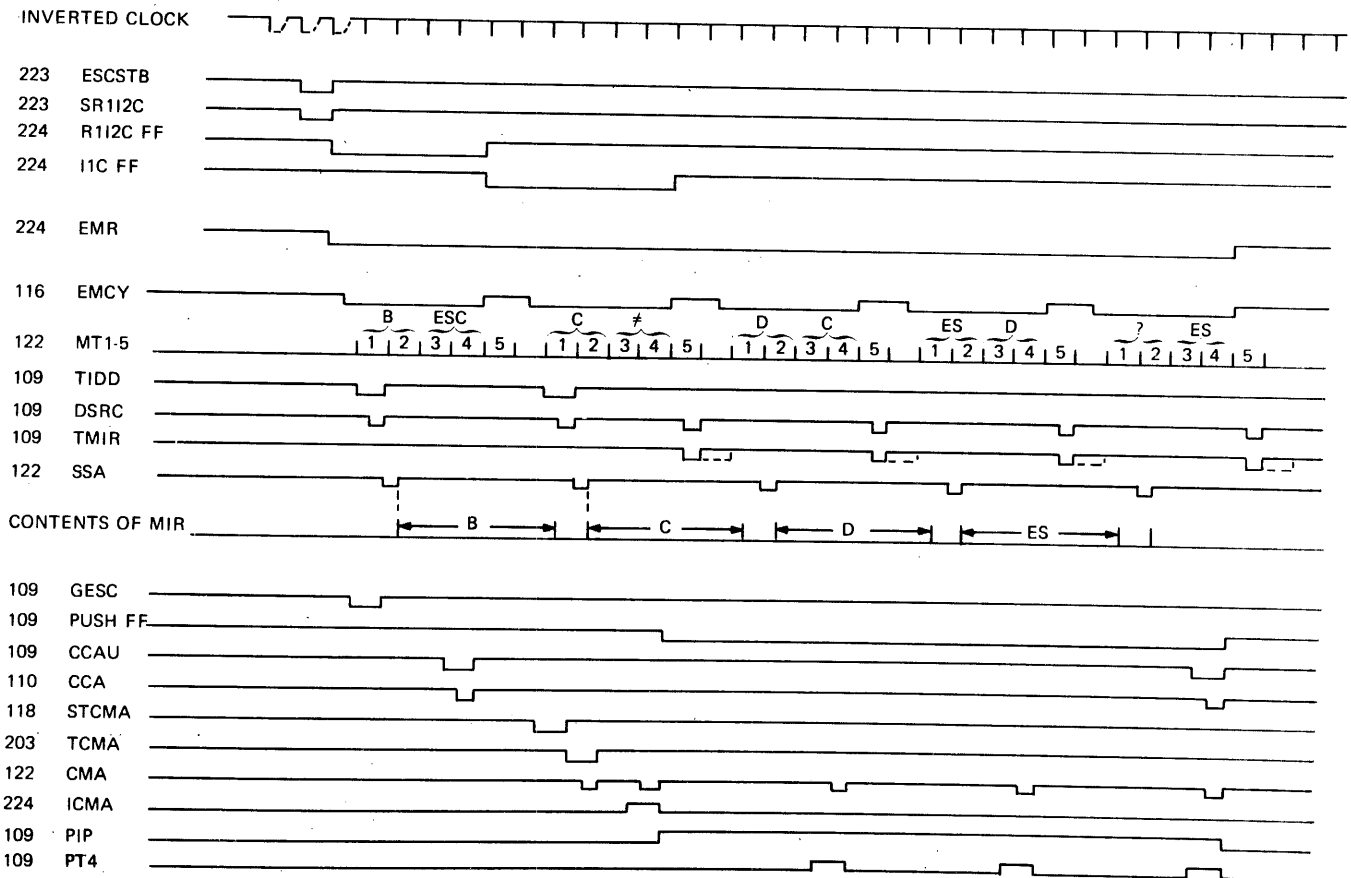


Fig. II-28 PUSH BY ONE REPLACE (ONE MEMORY POSITION CHAR. REPLACED BY A TWO MEMORY CHAR.)

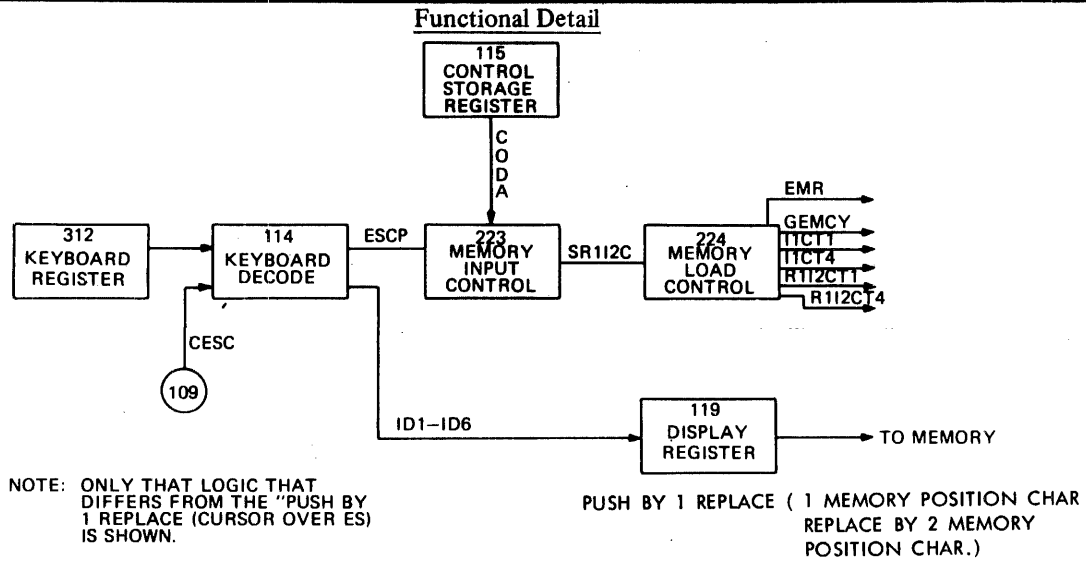


Fig. II-29 PUSH BY 1 REPLACE (1 MEMORY POSITION CHAR BY 2 MEMORY POSITION CHAR)

PUSH BY TWO REPLACE — (CURSOR OVER ES)

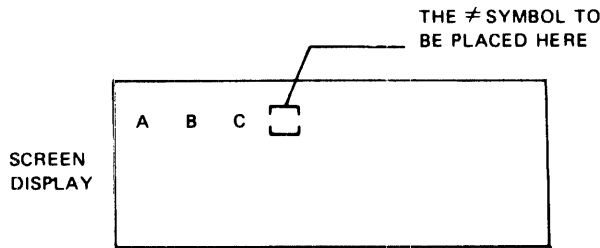


Fig. II-30

GENERAL DESCRIPTION — FIG. II-30, II-31

The Push by 2 Replace replaces the ES character with a 2 memory position character. This operation requires 3 memory cycles, with the addressing for the first cycle done by the Cursor Add Register, and the addressing for the remaining 2 cycles done by the Memory Add Register.

1ST MEM. CYCLE (CURSOR REGISTER ADDRESSING)

1. The ES character is read from memory into MIR. The ESC character is generated and clocked into the buffer reg.
2. The keyboard character (≠) is clocked into the display register.

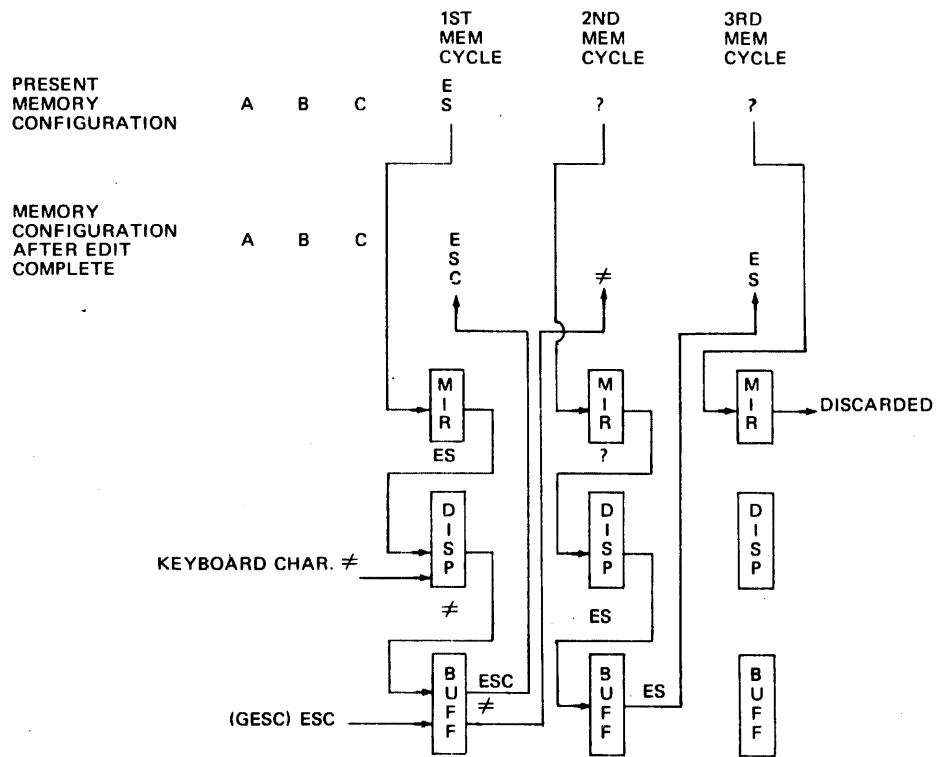


Fig. II-31

Functional Detail

3. The ESC character is written into the memory location formerly occupied by ES.
4. The ES character transferred from MIR to Display Register and the keyboard character in turn transferred from the Display Register to the Buffer Register.

At completion of the first memory cycle

Memory = ESC character  
 Display Register = ES character  
 Buffer Register = keyboard character (≠)

### 2ND MEMORY CYCLE (MEMORY REGISTER ADDRESSING)

1. Undetermined character read into MIR
2. The ≠ symbol is written into memory from the Buffer Register
3. The ES character is transferred from the Display Register to the Buffer Register
4. Contents of MIR transferred to the Display Register (undetermined character)

At completion of the second memory cycle

Memory = Keyboard character (≠)  
 Display Register = Undetermined character  
 Buffer Register = ES character

### 3RD MEMORY CYCLE (MEMORY REG. ADDRESSING)

1. Undetermined character read into MIR
2. The ES character is written into memory from the Buffer Register

Detailed Description (Fig. II-32 and II-33 and Related Prints)

Because the keyboard character is a 2 Memory Position character ESCSTB is generated (223). ESCSTB, and the fact that the cursor is over the ES character, causes the term SI2C (Set insert 2 character) to be developed, in turn setting the I2CFF on 224. With the I2CFF set, the first of the 3 edit memory cycles is initiated by EMR. EMR also sets the EMCY Flip-flop generating ACMA (Allow cursor memory addressing) for the first memory cycle.

#### 1ST MEMORY CYCLE

MT1 and MT2 – ES character read from memory into the MIR.

MT1 – I2CT1 –

- A. ESC generated, gated and clocked into the Buffer Register by GESC, TIDBR, and BRC
- B. 2CL Flip-flop is set, in turn developing LBR. LBR gates the contents of the Buffer Register onto the inhibit lines.
- C. RMA presets (clears) the Memory Address Register.
- D. STCMA (set transfer cursor to Memory Add) conditions TCMA Flip-flop

MT2 – I2CT2 –

- A. Flip-flop A1 pin M (109) set to hold the reset input of the push Flip-flop (109) disabled, so

that the Push Flip-flop cannot be hit by set and reset simultaneously at MT4.

- B. The keyboard character (≠) gated and clocked into the Display Register by TIDD, and DSRC.
- C. The contents of the Cursor Add. Register are gated and clocked into the Memory Add. Register by TCMA and CMA

MT3 and MT4 –

The contents of the Buffer Register (ESC character) are written into memory.

MT4 – I2CT4 –

- A. The Push Flip-flop is set holding EMR for additional memory cycles and developing the term PIP (push in progress). PIP disables ACMA and enables AMA, causing upcoming addressing to be a function of the Memory Address Register.
- B. I2CFF is reset.
- C. The Memory Address Register is counted by one in preparation for upcoming memory cycles.  
 (ICMA – Print 224, and CMA – Print 122)

MT5 –

Flip-flop A1 Pin I (Print 109) is set, to remember that the ES was detected and to condition the reset input of the Push Flip-flop.

MT5E –

The keyboard character in the Display Register is gated and clocked into the Buffer Register with TDSR and BRC.

Simultaneously, the ES character in MIR is gated and clocked into the Display Register by TMIR and DSRC.

#### 2ND MEMORY CYCLE

MT1 and MT2 –

Undetermined character read into the MIR.

MT3 and MT4 –

The contents of the Buffer Register are written into memory (the ≠) char.)

MT4 –

PT4 (109) generates CMA to count the Memory Address Register in preparation for next memory cycle.

MT5 –

Flip-flop A1 pin M (109) is reset, removing the disabling logic from the reset input of the Push Flip-flop.

MT5E –

TDSR and BRC gates and clocks the ES character from the Display Register into the Buffer Register. Simultaneously, the undetermined character in MIR is gated and clocked into the Display Register with TMIR and DSRC – This action is unnecessary at this point, but it causes no problems so it is allowed to occur.

Functional Detail

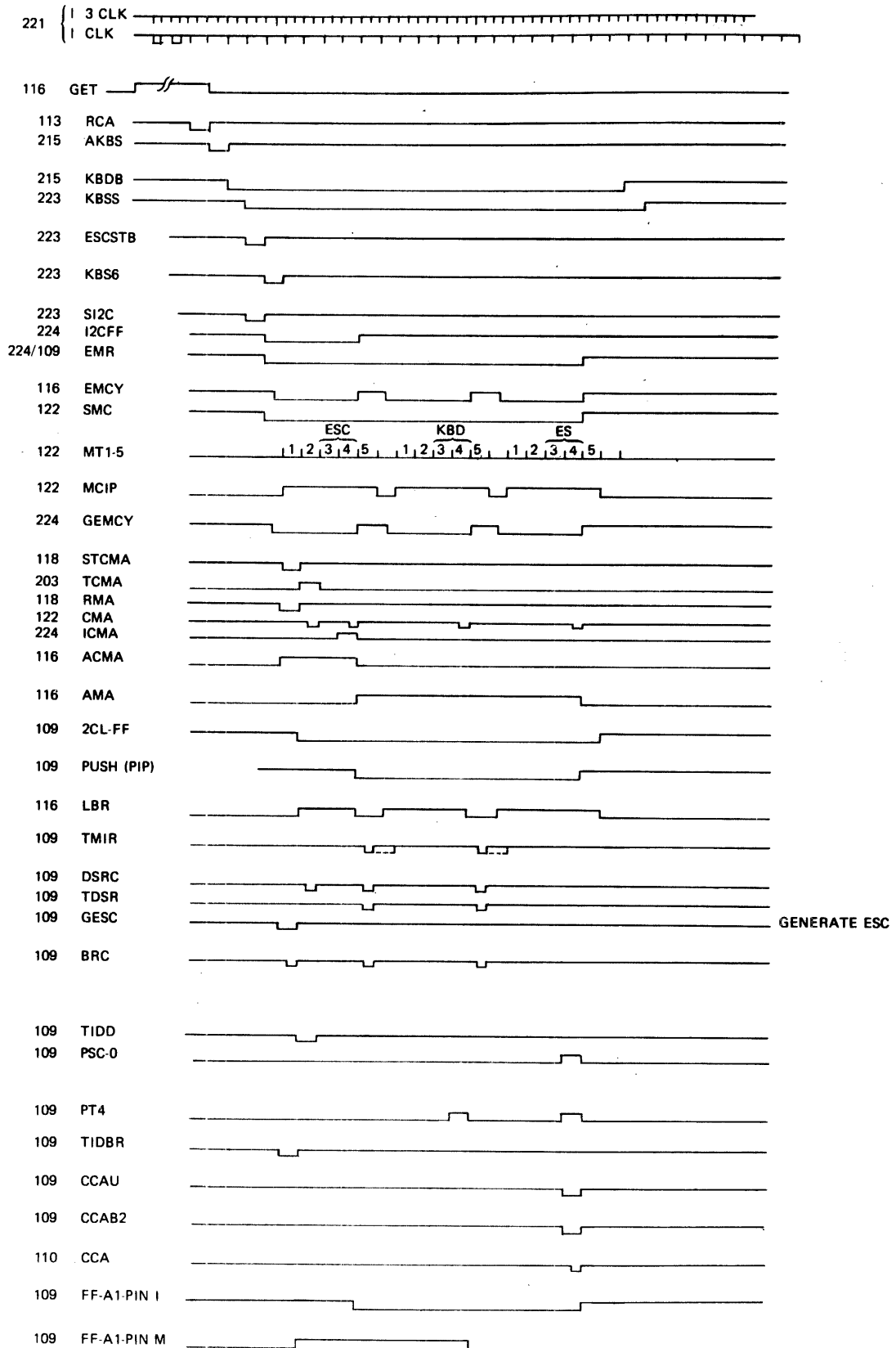


Fig. II-32 PUSH BY TWO REPLACE (CURSOR OVER ES)

Functional Detail

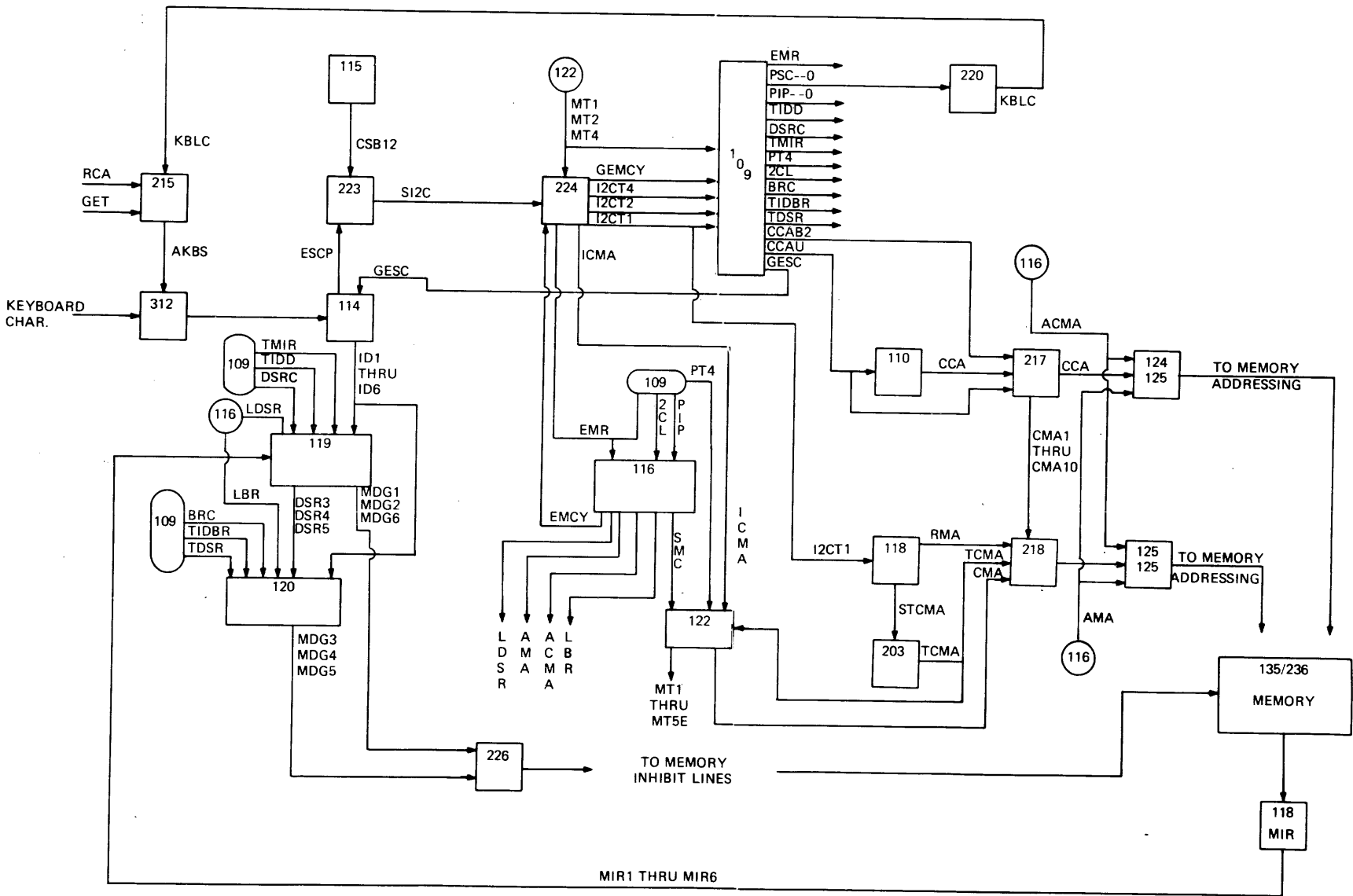


Fig. II-33 PUSH BY TWO REPLACE (CURSOR OVER ES)

Functional Detail**3RD MEMORY CYCLE**

MT1 and MT2 –

Undetermined character read into MIR

MT3 and MT4 –

The ES character in the Buffer Register is written into memory

MT4 –

- A. PT4 generates CMA to count the Memory Address Register in preparation for future operations.
- B. CCAU (Count Cursor Add. Up) and CCAB2 (Count Cursor Add. by 2) and CCA (Count Cursor Address) generated (109) to count the cursor register twice, so that as the operation is completed, the cursor is positioned over the ES character.
- C. Push Flip-flop reset, prohibiting further memory cycles.  
Push Flip-flop being reset disables the following terms:  
TMIR  
TDSR  
BRC  
DSRC
- D. PSC--0 (109) causes the PSCD Flip-flop on print 220 to be set, generating KBLC (keyboard Load Complete) and in turn causing the KBDB Flip-flop (Keyboard Busy) on print 215 to be reset. This allows other keyboards use of the Control Unit.

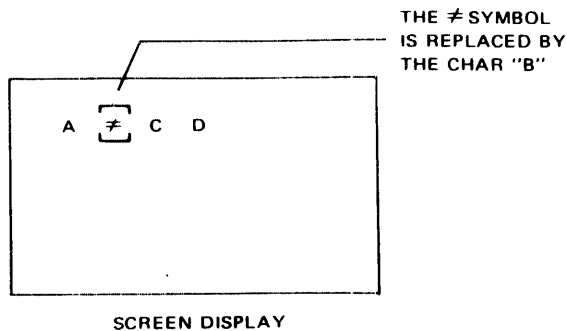
PULL BY ONE REPLACE

Fig. II-34

**GENERAL DESCRIPTION (FIG. II-34, II-35)**

The Pull By 1 Replace, replaces a 2 memory position character with a 1 memory position character, thereby causing everything in memory to the right of the replaced character to be shifted or pulled back by 1 memory position. In the example being used, this requires 7 memory

cycles. The first two memory cycles have the addressing done by the Cursor Add Register. For the remaining memory cycles, the addressing is alternated between the Cursor and Memory address registers.

**1ST MEMORY CYCLE (CURSOR REGISTER ADDRESSING)**

1. The ESC is read into MIR and discarded.
2. The keyboard character B is placed into the Display Register and written into the memory position formerly occupied by the ESC.

**2ND MEMORY CYCLE (CURSOR REGISTER ADDRESSING)**

1. The character C is read into the MIR register and, both written back into the same location and placed into the Display Register.

**3RD MEMORY CYCLE (MEMORY REGISTER ADDRESSING)**

1. The ≠ character is read into MIR and discarded.
2. The character C is written from the Display Register into the memory location formerly occupied by the ≠ character.

**4TH MEMORY CYCLE (CURSOR REGISTER ADDRESSING)**

1. The character D is read into the MIR register and simultaneously written back into the same location and placed into the Display Register.

**5TH MEMORY CYCLE (MEMORY REGISTER ADDRESSING)**

1. The C character is read into MIR and discarded.
2. The D character is written from the Display Register into the location previously occupied by the C character.

**6TH MEMORY CYCLE (CURSOR REGISTER ADDRESSING)**

1. The ES character is read into MIR, written back into memory and placed into the Display Register.

**7TH MEMORY CYCLE (MEMORY REGISTER ADDRESSING)**

1. The character D is read into MIR and discarded.
2. The ES character is written from the Display Register into the location formerly occupied by the D character.

Because this operation requires counting of the Cursor Add Register as well as the Memory Add Register, the completion of the Edit sequence finds the following:

1. The Memory add register is pointing one position beyond the ES character.



Functional Detail

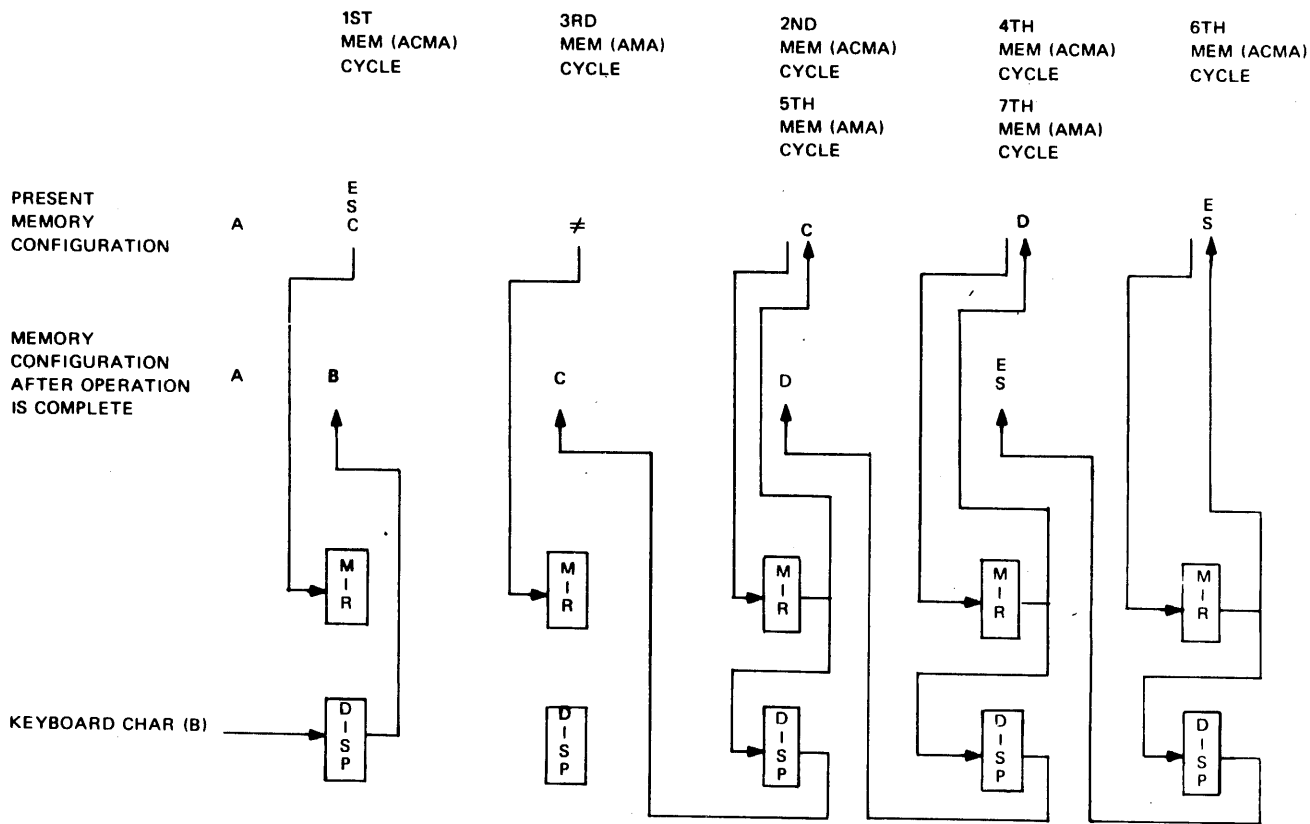
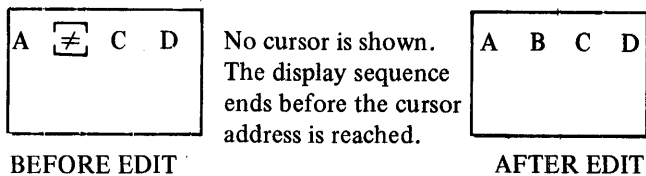
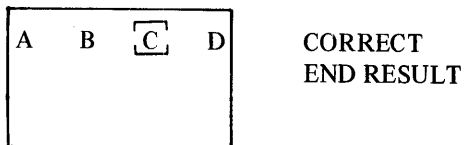


Fig. II-35

- The Cursor Add Register is pointing 2 positions beyond the ES character. If the operation was considered complete at this point, the result would be as shown.



What has happened is that the cursor ends up 2 positions beyond the ES character instead of simply being shifted one memory position to be positioned over the character C, as shown.



To handle this situation the following occurs:

- All keyboards are inhibited from the beginning of the edit sequence through two display sequences for this

monitor. This is necessary because the first display sequence sets the Control Storage Register to an unknown state, instead of the desired Cursor over Data (The character C). The 2nd display sequence sets the control storage to the desired state.

- Upon completion of the first display sequence, the Put sequence is inhibited. This insures that the original cursor address that is in the reserved area of memory is not destroyed by having the incorrect address written over it. (This first display sequence actually has no cursor display, because the display sequence ends before the cursor address is reached). This display sequence also sets the control storage register to a completely unknown state.
- The next Get sequence for this monitor reads the original cursor address from reserved core into the Cursor Add Register and upcounts it by one. Therefore, the Cursor Add Register is now at the proper memory location, over the character C.
- The second display sequence for this monitor now occurs, displays the cursor at the proper location and places the proper information into the Control Storage Register (CODA).
- Having completed the 2nd display sequence, the inhibiting logic for the keyboards is removed, and the keyboards again have access to the Control Unit.

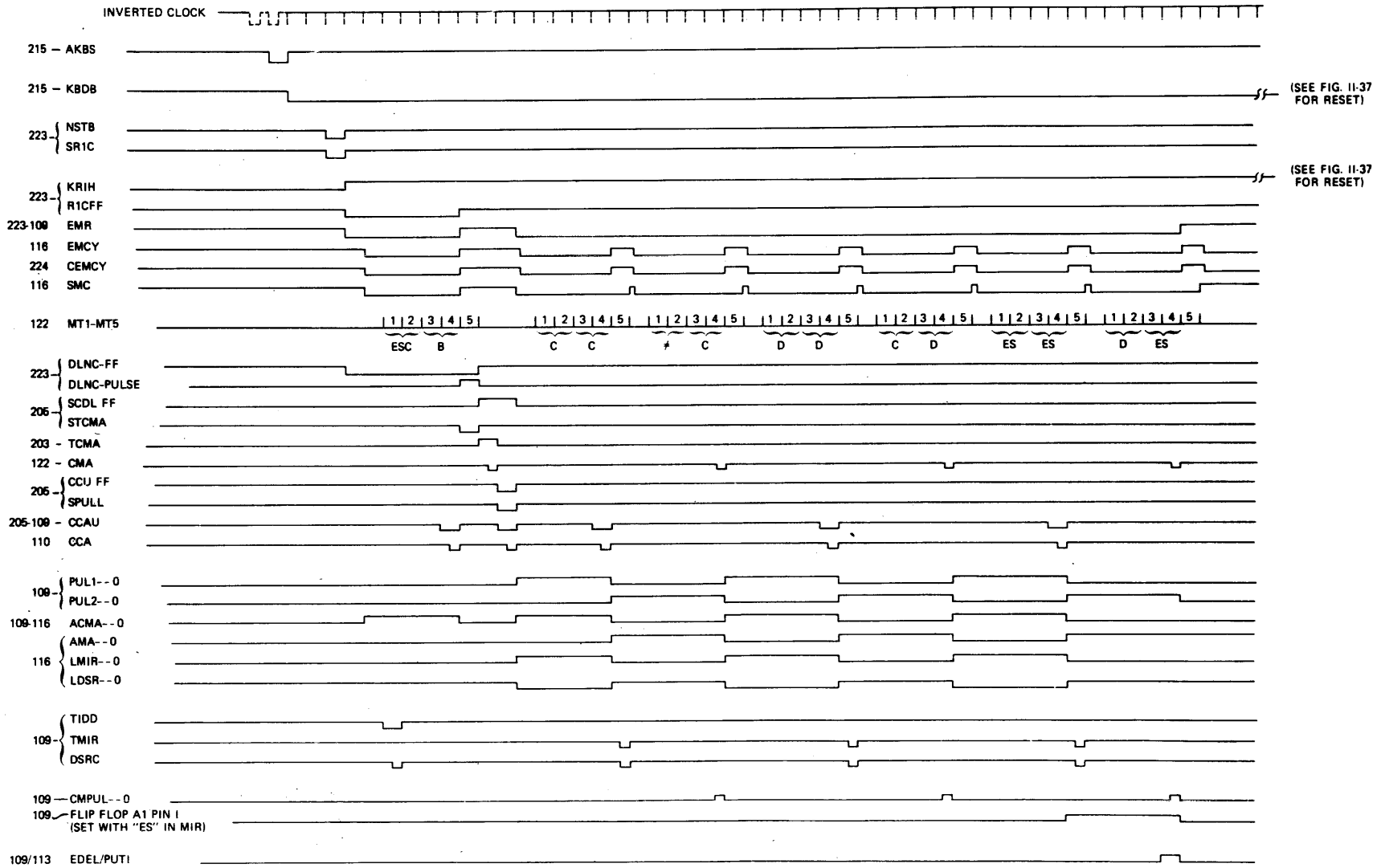
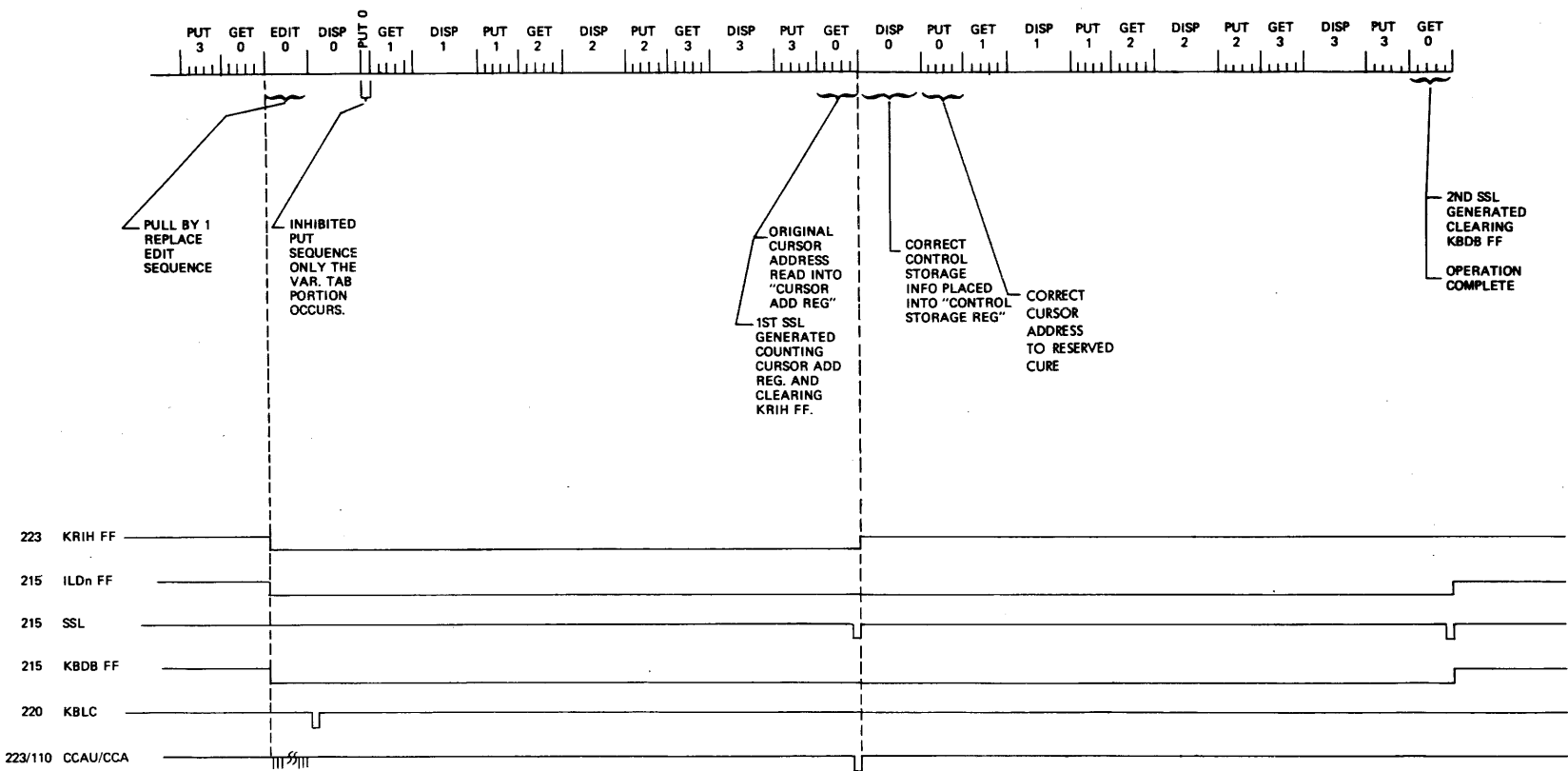


Fig. II-36 PULL BY ONE REPLACE

Functional Detail



BECAUSE THIS TIMING CHART COVERS A LONG TIME PERIOD,  
THE TIMING IS ONLY APPROXIMATE.

Fig. II-37

Functional Detail

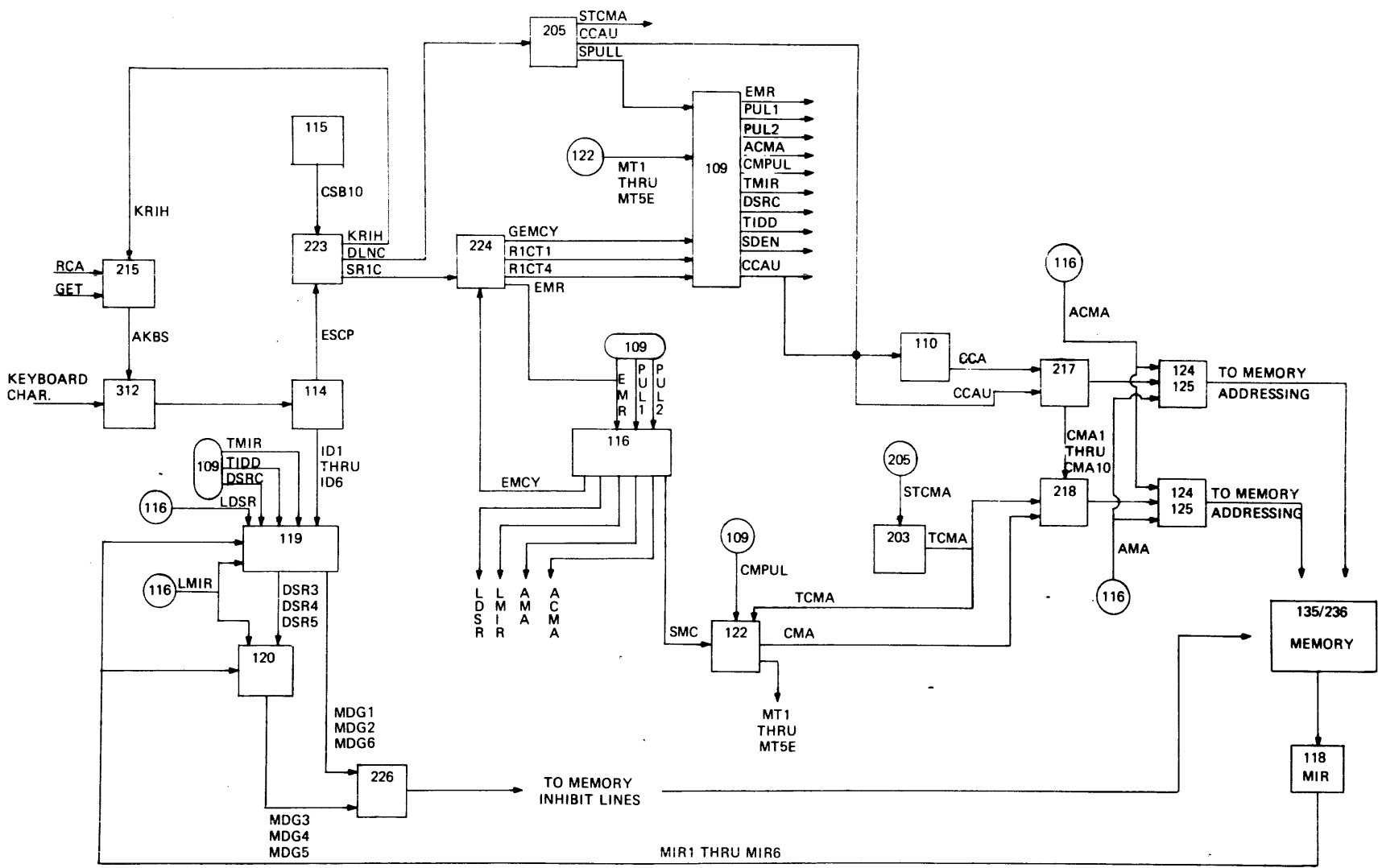


Fig. II-38 PULL BY ONE REPLACE

Functional Detail

## DETAILED DESCRIPTION

## (FIG. II-36, II-37, II-38 &amp; RELATED PRINTS)

At the time that AKBS (Allow Keyboard Strobe) is developed to strobe in the keyboard character, and the appropriate LCP Flip-flop is set (215), an appropriate ILN (215) Flip-flop is also set. This flip-flop does not get reset until the KBDB Flip-flop is reset and is part of the logic necessary for resetting KBDB Flip-flop. Having strobed the keyboard character into the keyboard storage register, NSTB is generated because the character to be placed in memory is a 1 memory position character. This, and the fact that the cursor is over ESC causes SR1C to be developed, setting the following flip-flops.

R1CFF - Develops EMR, initiating the first memory cycle and in turn setting the EMCY Flip-flop.

KRIH - Keyboard Reset Inhibit. This flip-flop remains set until 2 display sequences for this monitor have occurred. While KRIH is set, the KBDB (Keyboard Busy) Flip-flop cannot be reset, and all keyboards are denied access to the Control Unit.

DLNC - Delete Next Character Flip-flop is set. Though not a delete operation, this flip-flop is used at MT5 of the first memory cycle. Because the EMCY Flip-flop is set, ACMA is generated, allowing the Cursor Add Register to handle the addressing at this time.

## 1ST MEMORY CYCLE

MT1 & MT2 -

ESC read from memory into the MIR and discarded

R1CT1 - MT1 -

TIDD and DSRC gate and clock the keyboard character B into the Display Register.

MT3 & MT4 -

The keyboard char (B) written from the Display Register into the memory position previously occupied by the ESC.

R1CT4 - MT4 -

- A. R1CFF is reset
- B. CCAU & CCA counts the Cursor Add Register by one, causing it to point at the location now occupied by the ≠ symbol.

MT5 -

- A. DLNC-0 (223) generates STMCA (Set logic for TMCA FF - 205)

There is a 4 clock period between the end of the 1st memory cycle and the beginning of the 2nd memory cycle. During this time the following occurs:

1. Cursor Add Register and Memory Add Register are set so the Cursor Add Register is one count ahead of the Memory Add Register.
  - (a) TMCA and CMA gates and clocks the contents

of the Cursor Add Register into the Memory Add Register, meaning that now both registers are pointing at the ≠ symbol

- (b) The CCU Flip-flop (Count Cursor Up - 205) is set generating CCAU and CCA to upcount the Cursor Add Register to now be one count ahead of the Memory Add Register. This means that the Memory Register is pointing at the ≠ symbol and the Cursor Register is now pointing at the character C.
2. The PUL1 Flip-flop (109) is set by SPULL from the CCU Flip-flop (205)
  - (a) PUL1 generates EMR to initiate the 2nd of the memory cycles. EMR in turn sets the EMCY Flip-flop, generating ACMA, so the 2nd memory cycle addressing is by the Cursor Register.
  - (b) PUL1 also generates LMIR (Load MIR to memory), thereby causing the 2nd memory cycle to read the contents of memory into MIR and then write the same information back into memory. However the information in MIR would, in addition to being placed back into memory, also be placed into the Display Register for further use.

## 2ND MEMORY CYCLE (ACMA)

MT1 & MT2 -

The character C is read into MIR.

MT3 & MT4 -

Because of LMIR, the character C is written back into the same location of memory.

MT4 -

CCAU and CCA count the cursor add register in preparation for the 4th memory cycle.

MT5 -

- A. The PUL1 Flip-flop is reset, dropping LMIR, and attempting to drop EMR
- B. The PUL2 Flip-flop is set, developing LDSR to gate the future contents of the display register to the memory inhibit lines.
- C. PUL2 holds EMR and both generates AMA and drops ACMA (116) thereby indicating that the 3rd memory cycle addressing is done by the memory add register.

MT5 - R -

TMIR and DSRC gate and clock the char C from MIR into the Display Register.

## 3RD MEMORY CYCLE (AMA)

MT1 & MT2 -

The ≠ symbol is read into MIR and discarded

Functional Detail

## MT3 &amp; MT4 —

The character C in the display is written into the memory location occupied by the # symbol

## MT4E —

CMPUL (109) generates CMA to count the Memory Add Register by one in preparation for the 5th memory cycle.

## MT5 —

- A. PUL1 Flip-flop set holding EMR, and again generating LMIR
- B. PUL2 Flip-flop rest, attempting to drop EMR, and causing LDSR and AMA to be dropped. This means that ACMA again will be brought up with the setting of EMCY Flip-flop again.

## 4TH MEMORY CYCLE (ACMA)

## MT1 &amp; MT2 —

The character D read into the MIR.

## MT3 &amp; MT4 —

The character D written back into the same location from MIR.

## MT4 —

- A. CCAU & CCA count the Cursor Register in preparation for the 6th memory cycle

## MT5 —

- A. PUL1 Flip-flop is reset, dropping LMIR and attempting to drop EMR
- B. PUL2 Flip-flop is set, developing LDSR
- C. PUL2 holds EMR and generator AMA, so the 5th memory cycle addressing is by the Memory Add Register.

## MT5E —

TMIR & DSRC gate and clock the character D from MIR into the Display Register

## 5TH MEMORY CYCLE (AMA)

## MT1 &amp; MT2 —

The character C is read into MIR and discarded

## MT3 &amp; MT4 —

The character D is written into the memory position formerly occupied by the character C from the Display Register

## MT4E —

CMPUL generates CMA counting the Memory Add Register in preparation for the 7th memory cycle

## MT5 —

- A. PUL1 Flip-flop is set, holding EMR & again generating LMIR
- B. PUL2 Flip-flop is reset dropping LDSR and AMA and attempting to drop EMR, ACMA is brought up again as soon as EMCY is set by EMR.

## 6TH MEMORY CYCLE (ACMA)

## MT1 &amp; MT2 —

The ES character is read into MIR.

## MT3 &amp; MT4 —

The ES character is written back into the same location because of LMIR

## MT4 —

- A. CCAU & CCA counts the Cursor Add Register resulting in the cursor being two position beyond where the ES is to be written into memory at the next cycle.
- B. FFA1 pins H & I is set, disabling the set input to the PUL1 Flip-flop. This means that only 1 more memory cycle remains.

## MT5 —

- A. The PUL1 Flip-flop is reset dropping LMIR and attempting to drop EMR
- B. The PUL2 Flip-flop is set generating LDSR
- C. The PUL2 Flip-flop holds for the 7th memory cycle addressing

## MT5E -

TMIR & DSRC gate & clock the ES char into the Display Register.

## 7TH MEMORY CYCLE (AMA)

## MT1 &amp; MT2 —

The character D is read into the MIR and discarded.

## MT3 &amp; MT4 —

The ES character is written from the Display Register into the memory location previously occupied by the character D.

## MT4 —

- A. EDEL (109) generates PUTI (113) to set the counter on print 125 to a count of MCA4.
- B. SDEN — (109) begins initiate of the Display Sequence.

## MT4E —

CMPUL generates CMA to upcount the memory add register to 1 position beyond where the ES character is finally written.

## MT5 —

- A. Because the set input to PUL1 is disabled, it cannot be set at this time.
- B. The PUL2 Flip-flop is reset, dropping EMR and resulting in no additional memory cycles.
- C. Flip-flop A1 pins H & I is reset.

The display sequence is now initiated. Because the Cursor Add Register is at an address other than that desired (over the character B), the Put sequence associated with this display sequence is inhibited. This insures that the original cursor address in reserved core is not destroyed by

Functional Detail

having the incorrect address written over it. MCA4, which was developed during the 7th edit cycle, provides the logic for this. When the Put cycle is initialized, MCA4 being true, brings up the addressing for only the Variable Tab portion of the put cycle. MCA4 being true indicates also that this is the 5th of the 5 put cycles. The control unit has, in effect, been told that it has already completed the first 4 put cycles, and is now working on the 5th. Thus, the Cursor Address, and the Control Storage portion of the put sequence, have been completely eliminated and the information in core remains unchanged. MCA4 is cleared after the variable tab portion is complete.

When the next get sequence for this monitor occurs, the original cursor address and control storage information is read into the registers.

The following then happens:

1. GETn gated with its corresponding ILDn Flip-flop (set at the beginning of the operation) generates SSL (215).
2. SSL, generates CCAU and CCA to count the Cursor Add Register by 1, thereby causing the cursor register to be at the correct count, (over the character C).
3. SSL also sets Flip-flop D0 – output pin N., and clears the KRIH Flip-flop.

At this time, the 2nd display sequence occurs, and now the correct control storage information will be placed into the Control Storage Register (CODA – cursor over character C). Also, because MCA4 has been cleared, the

following PUT sequence will be a normal one and at this time the now corrected cursor address is placed into reserved core.

When another GET sequence for this monitor occurs, the following happens to complete the operation:

1. GETn, gated again with its corresponding ILDnFF generates SSL again.
2. SSL, gated with the output of Flip-flop D0 – output pin N (set at last get sequence for this monitor) resets the KBDB Flip-flop and the ILDn Flip-flop. KBDB being reset now allows all keyboards access, and the operation is complete.

CHARACTER DELETE (ONE MEMORY POSITION CHARACTER)

GENERAL DESCRIPTION (FIG. II-39, II-40, II-41)

The Character Delete operation is, with minor variations, almost identical to the Pull by one replace operation. For this reason, only a general explanation is given during which time, the differences are covered.

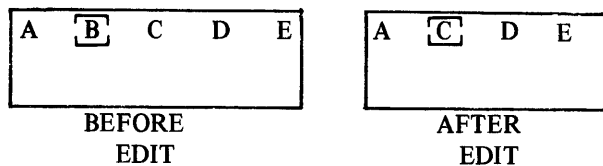
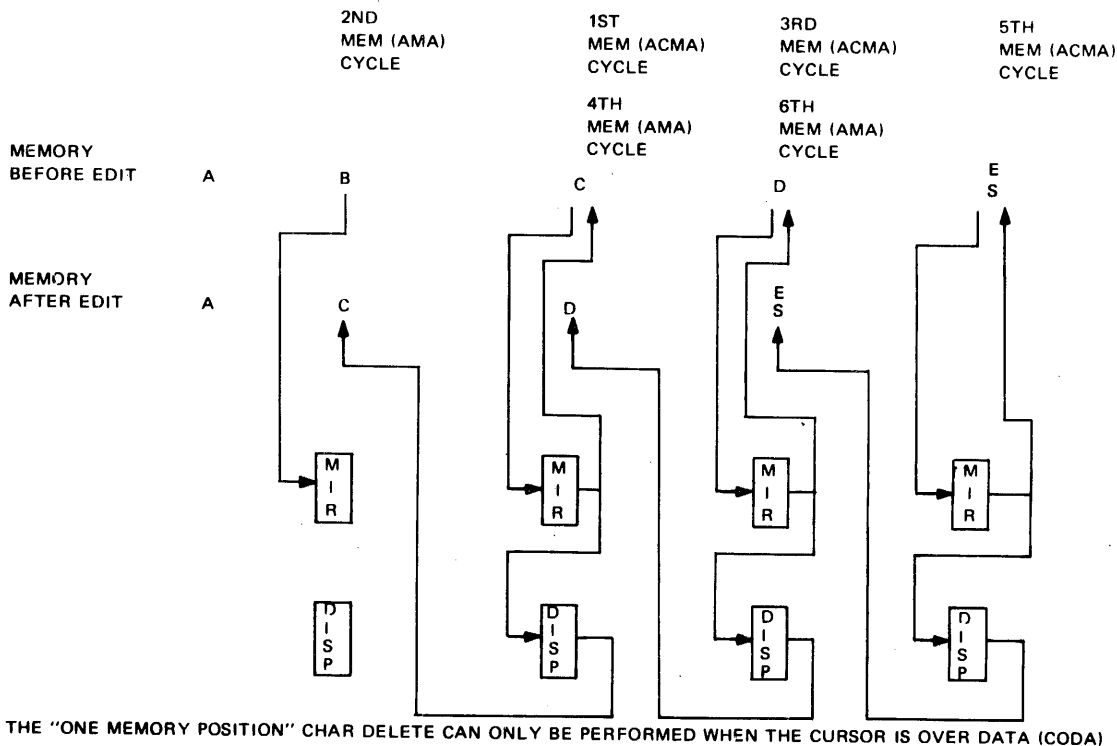


Fig. II-39



THE "ONE MEMORY POSITION" CHAR DELETE CAN ONLY BE PERFORMED WHEN THE CURSOR IS OVER DATA (CODA)

Fig. II-40





Functional Detail

Prior to the 1st Memory Cycle, the contents of the Cursor Add Register (which is pointing at the character B) is gated and clocked into the Memory Add Register. Addressing is alternated between the two registers, with the Cursor Register having the first cycle. The Cursor Register through the complete operation will always be one count ahead of the Memory Add Register.

**1ST MEMORY CYCLE (ACMA)**

The character C is read from memory into MIR, written back into the same location and placed into the Display Register. The Cursor Register is counted by one in preparation for the 3rd memory cycle.

**2ND MEMORY CYCLE (AMA)**

The character B is read into MIR and discarded, and the character C written into its place from the Display Register. The Memory Add Register is counted by one in preparation for the 4th memory cycle.

**3RD MEMORY CYCLE**

The character D is read into MIR and both written back into the same location and placed into the Display Register. The Cursor Add Register is counted in preparation for the 5th cycle.

**4TH MEMORY CYCLE (AMA)**

The character C is read from memory and into MIR and discarded. The character D in the Display Register is written into memory in place of the C character. The Memory Add Register is counted in preparation for the 6th memory cycle.

**5TH MEMORY CYCLE (ACMA)**

The ES character is read into MIR, written back into the same location and placed into the Display Register. The Cursor Register is then counted by one.

**6TH MEMORY CYCLE (AMA)**

The character D is read from memory and discarded and the character ES is written into its place from the Display Register. The Memory Add Register is then counted by one and inhibiting logic for the associated PUT sequence is also generated at this time. Upon completion of the EDIT sequence, the Cursor Add Register is pointing two positions beyond the new ES location and the Memory Add Register one position beyond. To get the Cursor Add Register back to its original count (The cursor position should not change in a character delete operation), the associated put sequence, which occurs immediately after the associated display sequence, is inhibited. This means that the contents of the Cursor Register cannot be written into reserved core, and the original cursor address which is in reserved core, is not destroyed. The next GET sequence for this monitor reads out the original cursor address from core, places it into the Cursor Add Register and the cursor is then back in its original location. This differs slightly from the Pull by 1 Replace. In that operation, the original cursor address is read from core during the GET sequence, and also counted by 1. The Pull by 1 Replace finishes with the cursor 1 position beyond its original location.

**PULL BY 2 REPLACE CHARACTER DELETE (2 MEMORY POSITION CHAR)**

This operation is identical to the Character Delete for a 1 memory position character with one exception. In this operation the Cursor Add Register is always two positions ahead of the Memory Add Register. (Fig.II-42, II-43)

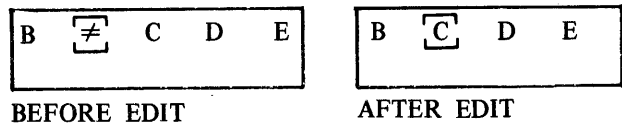


Fig. II-42

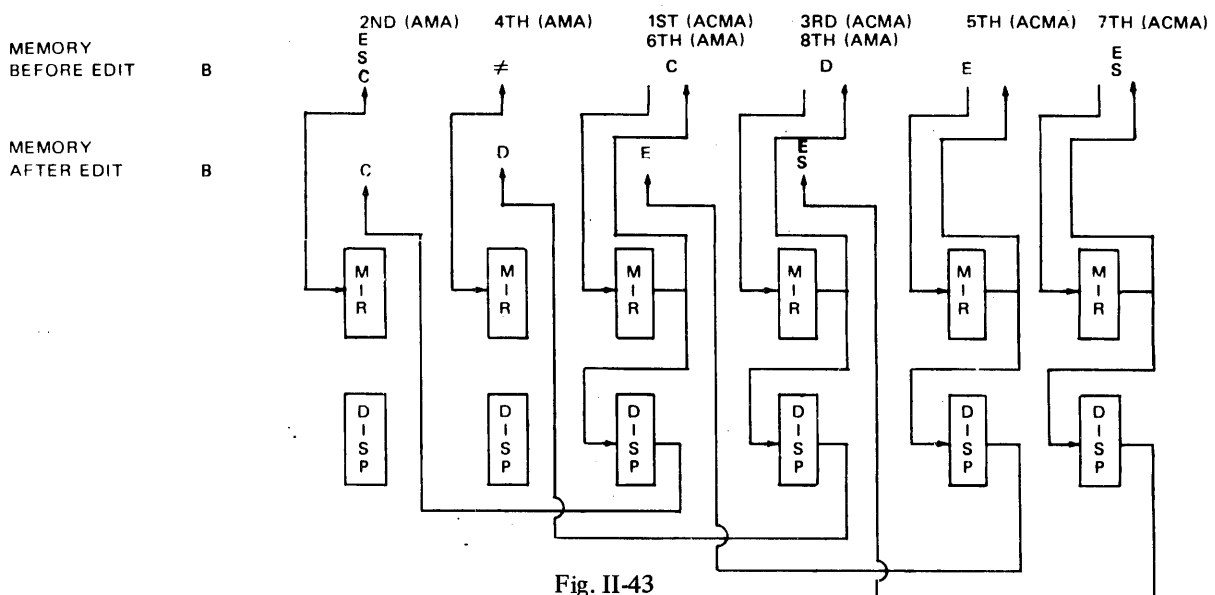


Fig. II-43



Functional Detail

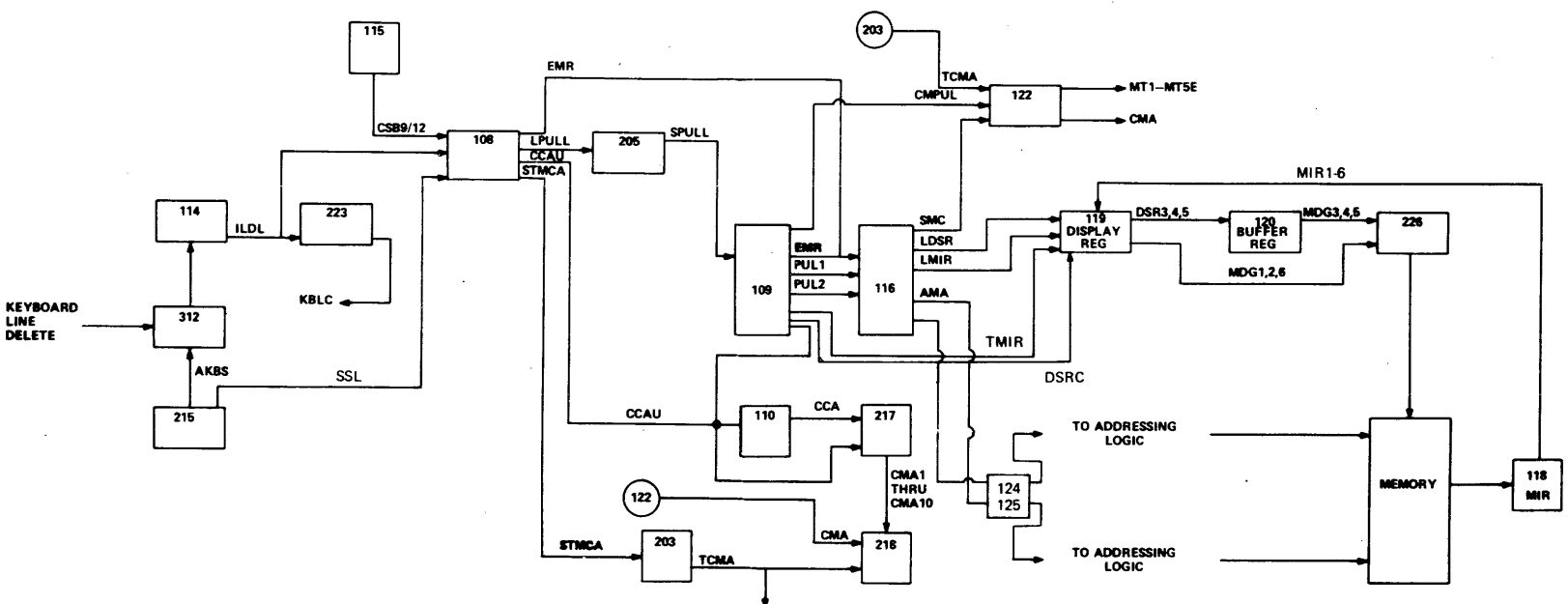


Fig. II-45 LINE DELETE

Functional Detail

everything between the cursor and the NL character. The sequence now becomes a basic Pull operation which continues until the ES is detected. The Put sequence, following the first Display sequence, is inhibited, as in the basic Pull operation, and it is not until the following Get sequence for this monitor that the cursor add. register is placed at its original count. This differs slightly from the basic Pull, where the cursor add register would end up 1 count above the original count.

CHARACTER INSERTGENERAL DESCRIPTION

Character Insert is a type of Push by 1 Replace. It differs only in that the cursor position does not change. The character in memory at the cursor address is replaced with a space character, and the previous character and all following characters are pushed down by one position. For this reason, only that portion that differs will be explained here.

DETAILED DESCRIPTION (PRINT 108, 110, 114, 223)

With the Insert key alone depressed, an insert character is placed into the keyboard storage register where, through the decoding network, it goes onto the ID1-6 lines (114) as a space character. Simultaneously, the level IINS (Input Insert) is developed (114) and used on 108 to generate CHIN (character insert). CHIN, gated with CSB3 (Cursor at Pos. 80 of L25 or Memory Full) and IDNS (Insert/Delete normal strobe 223), generate SIIC to set the IICFF and initiate a push edit sequence. The CSB3 term is used to insure that a push cannot occur with memory full or the cursor at the end of last line. This prevents information from being pushed out the end of memory. CHIN also generates ICC (Inhibit Cursor Count). This logic, used on 110, prevents the Cursor Add Register from being upcounted. For a detailed description of the remainder of the operation, see the Push By 1 Replace explanation.

LINE INSERTGENERAL DESCRIPTION

Line Insert is identical to the Character Insert, except that rather than replacing the character at the cursor address with a space, it is replaced with a NL character.

DETAILED DESCRIPTION – (PRINTS 110, 114, 205, 223, 312)

With the Insert key and the shift key depressed, an Insert New Line character configuration is placed into the keyboard storage register. Decoding takes place and the NL char is placed on the ID1-6 lines (114). Simultaneously, the level INL (Insert New Line -312) is developed. INL, gated

with NSTB, sets the NLC (New Line Char -205) Flip-flop, generating LI1N (Line Insert). LI1N, gated with CSB3 on 223, generates SIIC to set the IICFF and initiate a Push Edit sequence. LI1N also generates ICC (Inhibit Cursor Count) to set the ICC Flip-flop on 110. This flip-flop being set prevents the cursor add register from being upcounted. The remainder of the operation is a basic Push By 1 Replace.

BACKLINEGENERAL DESCRIPTION

Backline is used to return the cursor to position 1 of the same line. If the cursor is already at position 1, depressing the Backline key will move the cursor to position 1 of the preceding line. Except for the first sequence, the Backline sequences occur during the time allotted to the display sequence. A Backline sequence occurs prior to each line on the screen being displayed, and stores that lines beginning address in the memory location reserved for the cursor address. Therefore, when the cursor is found on a particular line, the beginning address for that line is already in reserved core, and any additional backline sequences are inhibited. This operation is carried out in the following manner. Assume the screen contains the following:

|   |   |          |   |   |
|---|---|----------|---|---|
| A | B | <u>C</u> | D | E |
| F | G | H        | I | J |
| K | L |          |   |   |

Depressing the Backline key causes a Backline sequence to be initiated. A Backline sequence consists of 2 memory cycles, during which time the contents of the Memory Add Register are written into that portion of memory reserved for the cursor address. Because the Memory Add Register is always cleared or preset to a count of 5 following every Edit or Display sequence, the initial Backline sequence places the count of 5 into the memory locations reserved for cursor add. Because the cursor is found before a NL or horizontal position count of 80 is detected, the operation is terminated, and the count of 5 left in the reserved core. Being in reserved core means that the new address cannot be placed into the Cursor Add Register until a GET sequence occurs. (Fig. II-46). As a result, the first Display sequence following the Backline sequence displays the cursor in the original location, and not until the next GET sequence for this monitor occurs will the display sequence then display the cursor in position 1 of the screen. The PUT sequence following the display sequence is inhibited to prevent overwriting the cursor address in memory.

Functional Detail

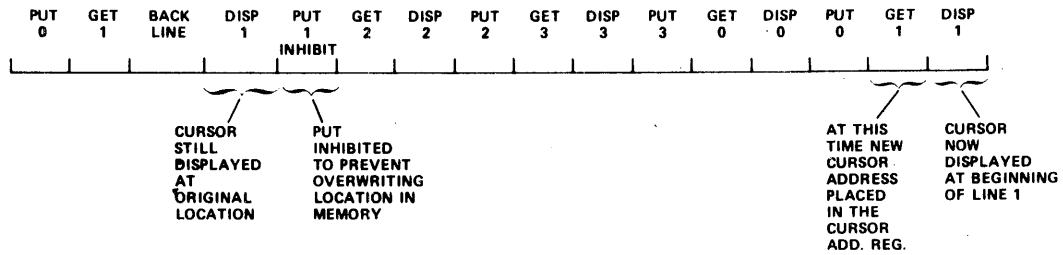


Fig. II-46 BACKLINE TIMING

If the screen contains the following:

|   |   |   |   |
|---|---|---|---|
| A | B | C | D |
| E | F | G | H |
| I | J | K |   |
| L | M | N |   |

Fig. II-47

The operation is similar to that described earlier, except that a Backline sequence is initiated (Interrupting the Display sequence) everytime a NL or horizontal position count of 80 is sensed during the Display sequence. (See Fig. II-47A).

Each Backline sequence places the memory address following the NL (or horizontal count of 80) into reserved core. Once the cursor has been located (DCUR), no further Backline sequences will occur.

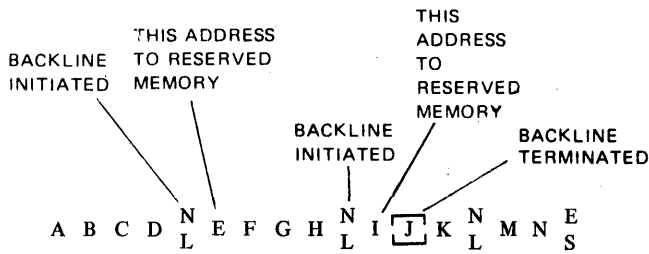


Fig. II-47A

**DETAILED DESCRIPTION (FIG. II-46 AND RELATED PRINTS)**

The detailed description will be for that screen display shown in Fig. II-47. Depressing the Backline key cause IBL (Insert Backline Level) to be generated (114). IBL disables KBLC (223), preventing other keyboards from having access. IBL and NSTB causes the BLS (Backline Search -205) Flip-flop to be set. This flip-flop remains set until the cursor position is found, at which time it is reset, terminating the operation. In addition to setting the BLS Flip-flop, the term SSMA is generated, setting the SMA

Flip-flop (220) and developing SBLR (Set Backline Request). SBLR is used to set Flip-flop EO – pins I and H (123). With this flip-flop set, a BLMR (Backline Memory Request) is initiated and the contents of the Memory Add Register MA1 thru MA6 are gated to the MDG1-6 lines (123). Because neither AMA (Allow Memory Register Addressing) or ACMA (Allow Cursor Register Addressing) are used at this time, the addressing logic is that which is used with the Put Get sequences. As a result, this first Backline Memory cycle finds MCSB1, MCSB2 and MCSB3 (123) false. Being false, the X addressing lines for the first of the 2 cursor locations in reserved core is brought up. The Y address lines are determined by which monitor the Backline is being performed on. BLMCY, gated with the appropriate ILDN Flip-flop (215), produce the SnCC terms (220) and these terms in turn activate the appropriate Y lines.

At MT4 of the first memory cycle, Flip-flop EO-pins I and H is reset, and Flip-flop EO-pins M and N is set. (123) This flip-flop holds BLMR so that a 2nd memory cycle is initiated, and gates Memory Add Register MA7 thru MA10 to the MDG 1-6 lines. Also at this time, Flip-flop EO-pins M and N changes the state of the MCSB lines to MCSB1, MCSB2 and MCSB3 so that the 2nd of the 3 cursor locations in reserved core is addressed. At MT4 of the 2nd cycle, BLMR is dropped, the BLMCY Flip-flop is reset (116) and the Display sequence is initiated (The cursor address in reserved core is not pointing at position 1 of the 1st line). The display sequence is initiated with the Memory Add Register preset to a count of 5 (Display sequence addressing done by the Memory Add Register). Upon detecting a NL character in MIR, or finding the Horizontal Position Counter at 80, the EOL Flip-flop (121) is set. With the next CMA pulse (This next CMA counts the Memory Add Register to one count beyond the NL character or to one count beyond where the HP=80 is detected), SSMA is generated, initiating a Backline sequence. Because BLMCY has a higher priority (116) than DMCY, the display sequence is interrupted. The Backline sequence now places the contents of the Memory Add Register into the location reserved for the cursor address. At this time, the BLMCY Flip-flop (116) is reset and the DMCY again is set to let the Display sequence continue. (At this point, the cursor address in memory is pointing at position 1 of the 2nd line).

Functional Detail

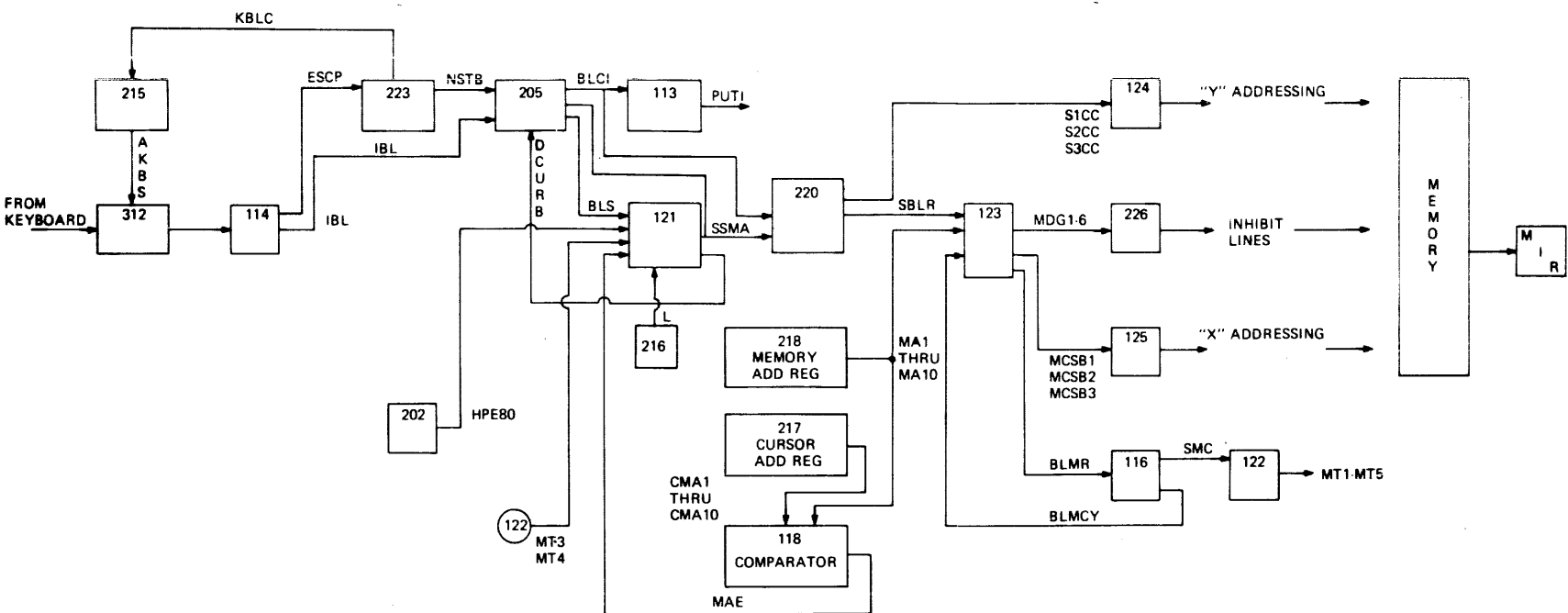


Fig. II-48 BACKLINE

Functional Detail

Because the cursor is not on the 2nd line, a Backline sequence is again initiated when a NL or HP=80 is detected. (This Backline causes the cursor address in memory to be pointing at position 1 of the 3rd line).

Because the cursor is on the 3rd line, when the Memory Add Register equals the Cursor Add Register, DCURB (Display Cursor) is generated, resetting the BLS Flip-flop (205). With the BLS Flip-flop reset, Backline is terminated and NL or HP=80 cannot initiate a Backline sequence. In addition to resetting the BLS Flip-flop, DCURB also develops BLCI (Backline inhibit -205). BLCI in turn, develops PUTI (Put Inhibit -113). This is necessary because a Put sequence would overwrite the present contents of the reserved cursor locations which now contains the address for position 1 line 3. BLCI also sets Flip-flop DO pin N' (220), which, gated with the following SSL pulse, generates KBLC to clear the KBDB Flip-flop (215) and allow other keyboards access. The SSL pulse does not occur until the following Get Sequence for this monitor occurs. This get sequence sets the proper cursor address into the Cursor Add Register.

BACKSPACEGENERAL DESCRIPTION (FIG. II-49)

A backspace moves the cursor one screen position to the left. If the cursor is at the first position of a line, the

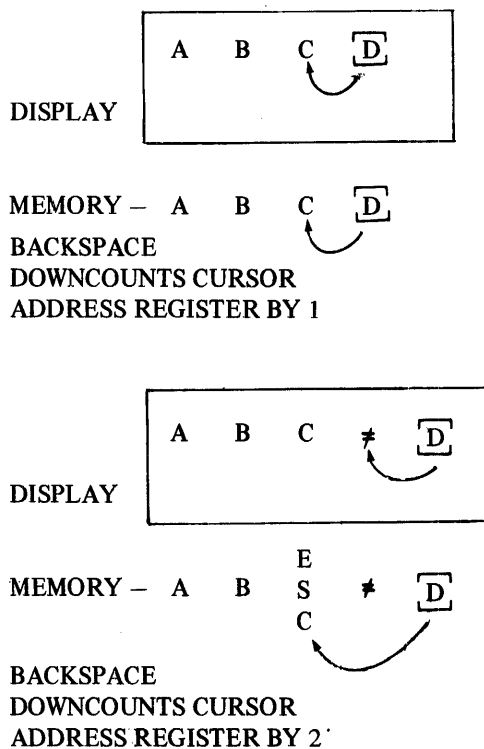


Fig. II-49

backspace moves it to the last position of the previous line. In memory, however, the cursor address may be downcounted by 1 or 2, depending on whether the cursor is located after an ESC character.

DETAILED DESCRIPTION (PRINTS 110, 114, 205, 217, 223)

Backspace causes IBS (Input Backspace -114) to be generated. IBS is gated with NSTB and CSB7 on print 205. IF CSB7 is false (cursor not following ESC), CCAD (count cursor address down) is generated, counting the Cursor Address Register down by 1. If CSB7 is true (cursor following ESC), CCAD and CCAB2 (count cursor address by 2) is generated, counting the Cursor Add Register down by 2. CCAD is used directly on 217 (Cursor Add Register) and also on 223 to develop CCA and KBLC. KBLC clears KBDB, allowing other keyboards use of Control Unit again. CCAB2 is used directly on 217 to cause the Cursor Register to be downcounted by 2.

A backspace cannot be performed if the cursor is at position 1 line 1 of the screen. The inhibiting logic is generated if MAE (Memory Add Register = Cursor Add Register) is false at the time IBS and NSTB occur (205). Because the Memory Add Register stands by at a preset count of 5, 261, 517 or 773, depending on the monitor in use, and because these counts are the equivalent of position 1 line 1 on their corresponding screens, then, if MAE goes false, it indicates the Cursor Add Register is also at the same count or on position 1 of line 1 of the screen. MAE being false, prevents the countdown logic from being developed and causes BSCP to be developed. BSCP on 110 generates KBLC, clearing the KBDB Flip-flop and allows the other keyboards to have use of the control unit again.

HOMEGENERAL DESCRIPTION

The Home key, when depressed, returns the cursor to position 1 line 1 of the screen. It does this by setting the Cursor Add Register to a count of either 5, 261, 517, or 773, depending on which monitor is being homed, and how the memory is divided.

DETAILED DESCRIPTION (PRINTS 114, 217, 218, 224)

The Home key causes the term ITOF (Input Top of Form) to be developed (114). ITOF generates RCMA (Reset Cursor Memory Add Register -224), which resets or clears the Cursor Add Register to a count of 5 (Normal Cleared State). ITOF also sets the CTM (Transfer Memory Add Register to Cursor Add Register -224) Flip-flop which then both gates and clocks the contents of the Memory Add Register (218) to the Cursor Add Register (217). The Memory Add Register stands by in a cleared state of 5, 261, 517, or 773, depending on which monitor the control unit

Functional Detail

is working with. Therefore, depending on which monitor is being homed, determines what count is being transferred to the Cursor Add Register from the Memory Add Register. With a 4 monitor system, 5, 261, 517, or 773 represents position 1 line 1 on the corresponding screens.

CLEARGENERAL DESCRIPTION

A screen is cleared by writing an ES character into the memory position corresponding to position 1 line 1 of the screen. As a result, the Display Sequence detects the ES character in its first memory cycle and immediately is terminated with nothing being displayed on the screen but the cursor.

DETAILED DESCRIPTION -- PRINTS 109, 114, 116, 119, 215, 224

The Clear key causes CLS (Clear Screen -114) to be generated. CLS, gated with NSTB on 224, produces RCMA (Reset Cursor Memory Address) and CTM (Memory Add Register). RCMA clears the Cursor Add Register to a count of 5, and CTM gates and clocks the contents of the Memory Add Register into the Cursor Register. The Memory Add Register contains the count corresponding to position 1 line 1 of the screen being cleared. Simultaneously, CLS and NSTB set the GES (Generate End Screen) Flip-flop and the R1C Flip-flop on 224. The R1C Flip-flop when set, generates EMR (Edit Memory Request), which initiates a 1 memory cycle edit sequence at the location determined by the Cursor Add Register. KBLC (224) is also developed, resetting the KBDB Flip-flop (215), allowing other key-boards use of the control unit again.

The output of the GES Flip-flop meanwhile forces an ES configuration onto the ID1-6 lines (114). At MT1 (R1CT1-224), TIDD and DSRC (109), gate and clock the ES character from the ID1-6 lines into the Display Register (119). Because LDSR (Load Display Register to Memory -116) stands by true, when the write portion of the memory cycle occurs (MT3 and MT4), the ES character is written into memory from the Display Register. At MT4 time, the R1CFF is reset, dropping EMR and ending the operation.

POWER ON CLEARGENERAL DESCRIPTION

Power On Clear, is a level that is generated during the initial power up. (Power on Clear can also be generated as a maintenance aid by the use of the Clear Allow switch on the Control Unit along with the Break key on any keyboard).

Power On Clear Accomplishes Two Things:

1. Causes control and register flip-flops to come up in a cleared and/or preset state.
2. Causes ES (End Screen) characters to be written into the first displayable location of memory allotted to the various screens, i.e., 0005, 0261, 0517, 0773.

DETAILED DESCRIPTION (FIG. II-49A, PRINTS 221, 218, 123, 122, 116)

|  |                    |  |
|--|--------------------|--|
| SMC<br>AMA   | POCB               | Generate Memory Cycles   |
| CMA  | POCB<br>MT4        | Count Memory Address Register  |
| $\overline{\text{MDG1}}$<br>$\overline{\text{MDG2}}$<br>MDG3<br>MDG4<br>MDG5<br>MDG6 | POCB<br>MA1<br>MA3 | Place ES Configuration on MDG1 -- MDG6 Lines, which in turn go to the inhibit lines. |

Fig. II-49A

Power On Clear is generated on 221 by the slow charge of capacitor C1 and C2 through resistor R22.

When power is turned on, all supply voltages are instantly present. Because of the slow charge of C1 and C2, transistor Q8 comes up in a cut off state and remains cut off until C1 and C2 have charged sufficiently to allow it to conduct. (Approximately 200 milli-seconds).

Because Q8 is cut off, Flip-flop Z2 is set, causing Power On Clear to be generated through transistors Q9 and Q10, Q11 and Q12.

When C1 and C2 have charged sufficiently to allow Q8 to conduct, Flip-flop Z2 will be reset, terminating Power On Clear.

The writing of ES characters in memory is accomplished by having POCB (Power On Clear B) generate SMC (Start Memory Cycle -116). This means continuous memory cycles occur during the time Power On Clear is present.

POCB also generates AMA (Allow Memory Addressing -116), causing all addressing to be a function of the Memory Address Register (218). At MT4 of each memory cycle, CMA (Count Memory Address -122) is generated to upcount the Memory Address Register. Each time MA1 and MA3 Flip-flops are set, an ES configuration is placed on the MDG1-MDG6 lines (123) to be written into memory on the next memory cycle.



Functional Detail

This means ES characters will be written into locations 0005, 0261, 0517, 0773 and many other locations as well. However, it is only the first ES character detected for any screen that has any effect. The entire operation ceases when Power On Clear is terminated.

FIXED TAB

NOTE: A Fixed Tab function causes the cursor to be positioned to the next screen position that corresponds to a mod 8 count of the horizontal position counter.

EXAMPLE #1 – A Mod 8 Count Of The H.P. Counter Is Reached Before An ES or NL Character Is Detected.

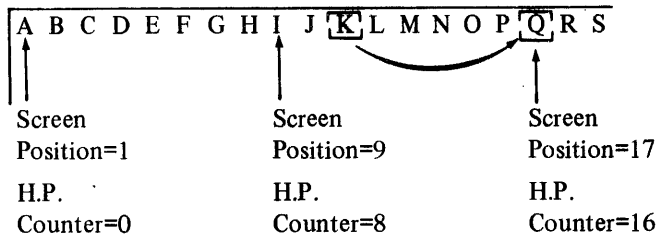


Fig. II-50

GENERAL DESCRIPTION OF EXAMPLE #1 (FIG. II-50)

Other than the keyboard function of placing the FT (Fixed Tab) character in the keyboard storage register, the entire operation takes place during the Display Sequence associated with the keyboard. Because the Fixed Tab key is depressed, a check is made for a mod 8 count in the H.P. counter as it counts during the display sequence. The check begins at the cursor location (over the character K) and continues until the H.P. counter reaches a mod 8 count (at the character Q).

At that time, the contents of the Memory Add Register are transferred to the Cursor Add Register and because the H.P. counter and the Memory add register have been counting together, when the next Display Sequence for this monitor takes place, the cursor will be displayed over the character Q.

DETAILED DESCRIPTION EX. #1 PRINT 110 FIG. II-56

IFTB (Input Fixed Tab -312) is generated by the fixed tab key. IFTB, gated with DCT4, (Display Cursor MT4), sets the FTC (Fixed Tab Compare) Flip-flop, providing the H.P. counter is not at a mod 8 count (T STOP-0). With the FTC Flip-flop set, the next time the H.P. counter reaches a mod 8 count, SCTM (Set Memory Register to Cursor Register) is developed. SCTM in turn generates RCMA (Reset Cursor Memory Add -224), clearing

the Cursor Add Register. SCTM also sets the CTM Flip-flop 224. CTM then gates and clocks the contents of the Memory Add Register to the Cursor Add Register.

EXAMPLE #2 --  
ES Detected Before H.P. Counter Reaches a Mod 8 Count. (Fig. II-51)

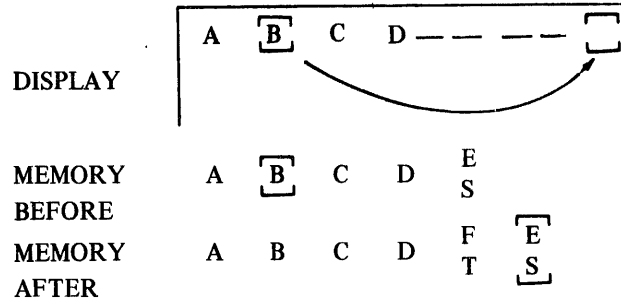


Fig. II-51

As in Example #1, beginning at the cursor location, and during the Display Sequence, a check is made of the H.P. counter for a mod 8 count. Because the ES character is detected before the mod 8 count is reached, the address of the ES character is placed in the Cursor Add Register and conditions set for initiating an Edit sequence at the proper time. The Edit sequence performs a basic Push By 1 Replace, replacing the ES character with the FT character, pushing the ES down and placing the Cursor Add Register to the ES locator. The next Display Sequence, upon detecting the FT character, immediately jumps the H.P. counter to the next mod 8 count which is where it is positioned when the cursor is displayed.

DETAILED DESCRIPTION EXAMPLE #2 – FIG. II-57, PRINT 110-202-121

At the DCT4 (Display Cursor MT4) time of the Display Sequence the FTC Flip-flop is set. Sensing of the ES character generates SCTM, which causes the Memory Add Register contents (ES location) to be placed into the Cursor Add Register. The ES character also cause the I1CS (Insert 1 character set) Flip-flop to be set. This Flip-flop is set to insure that an Edit Sequence is performed after the next GET Sequence for this monitor has occurred. At that time SSL, gated with I1CS Flip-flop, generates SI1C (Set Insert 1 Character) to set the I1CFF (224), thereby initiating a Push By 1 Replace edit sequence. Having performed the Push By 1 Replace edit sequence and written the FT character into memory from the Display Register (119), it is not until the next display sequence that the cursor is actually seen at the new position on the screen. The sensing of the FT character during the display sequence generates CHB8 (Count H.P. Counter by 8 – Print 121),

Functional Detail

CHB8 immediately causes the H.P. counter (202) to jump to the next mod 8 count instead of just advancing by 1. The result is that the CRT beam is also rapidly positioned to that corresponding screen position at which time the cursor is displayed.

EXAMPLE #3 -  
 NL Detected Before H.P. Counter Reaches A Mod 8 Count (Fig. II-52)

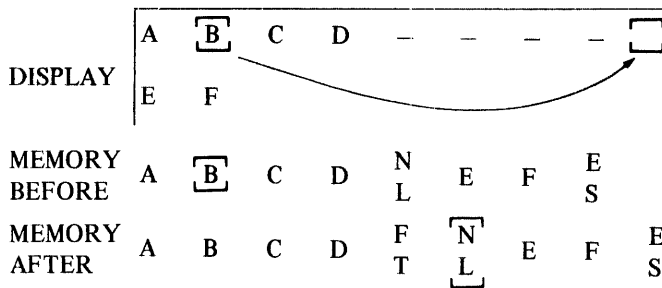


Fig. II-52

DETAILED DESCRIPTION EXAMPLE #3 (FIG. II-57)

This example is identical to example #2 except that it is the NL character being detected before a mod 8 H.P. count that initiates the Push By 1 Replace edit sequence.

EXAMPLE #4 -  
 Fixed Tab Performed When Cursor Located Beyond HP of 72 and the ES Character Is Before HP of 79. (Fig. II-53)

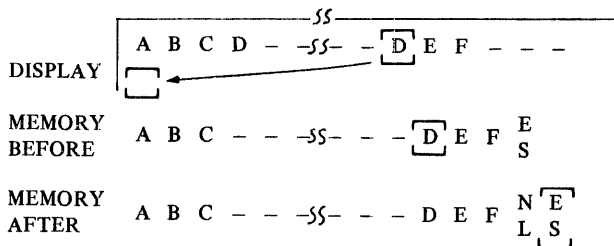


Fig. II-53

The operation is similar to the examples #2 or #3 in that a Push By 1 Replace sequence is performed. It differs in that the Push By 1 Replace replaces the ES with a NL instead of a FT (Fixed Tab). This is necessary because if a FT (Fixed Tab) was inserted, the next display sequence would attempt to count the H.P. counter by 8 which would place the counter at a count exceeding 79.

DETAILED DESCRIPTION - EXAMPLE # 4 FIG. II-58, PRINTS 110, 205, 224

Because Display Cursor (DCT4) occurs when the H.P. count is over 72 (HP072), and IFTB is present, the level

SNLK (Set New Line Character) is generated. SNLK sets the NLC FF on 205, generating NLC and LI1N. LI1N forces the NL configuration onto the ID1 - 6 lines (114) and NLC returns to 110 to be gated with SSL (215). SSL occurs with the next GET sequence for this monitor, and gated with NLC, sets the MCEL Flip-flop. The output of the MCEL Flip-flop is gated so that when the ES character is detected during the following Display sequence, the term ESUD (End Screen Update) is generated, setting the I1CS (Insert 1 Character Set) Flip-flop. With I1CS Flip-flop set, the next GET sequence generates SSC which, gated with the I1CS Flip-flop, produces SI1C to set the I1CFF (224) and generate a Push By 1 Replace Edit sequence. This edit sequence replaces the ES character with the NL character on the ID1-6 lines and pushes the ES down one position.

Example #5 -  
 Fixed tab performed with cursor beyond HP of 72 and NL or HP count of 79 reached.

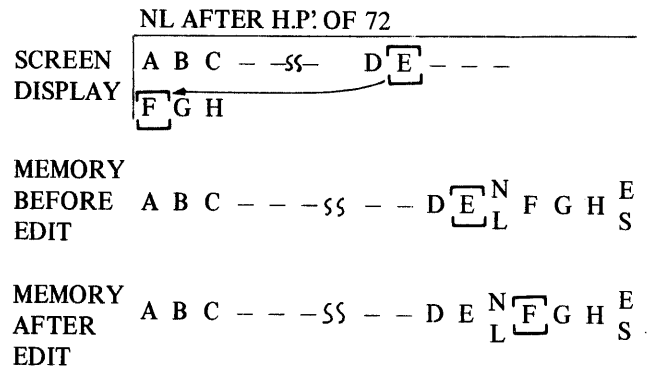


Fig. II-54

NO NL AFTER H.P. OF 72, AND A H.P. OF 79 IS REACHED

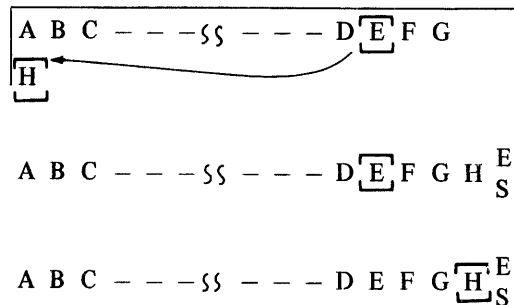


Fig. II-55

With either of these configurations, the memory remains the same, only the cursor add register is changed. When either a NL character or a H.P. count of 79 is reached during the display sequence, the contents of the following Memory Add Register locations are transferred to the Cursor Add Register.

Functional Detail

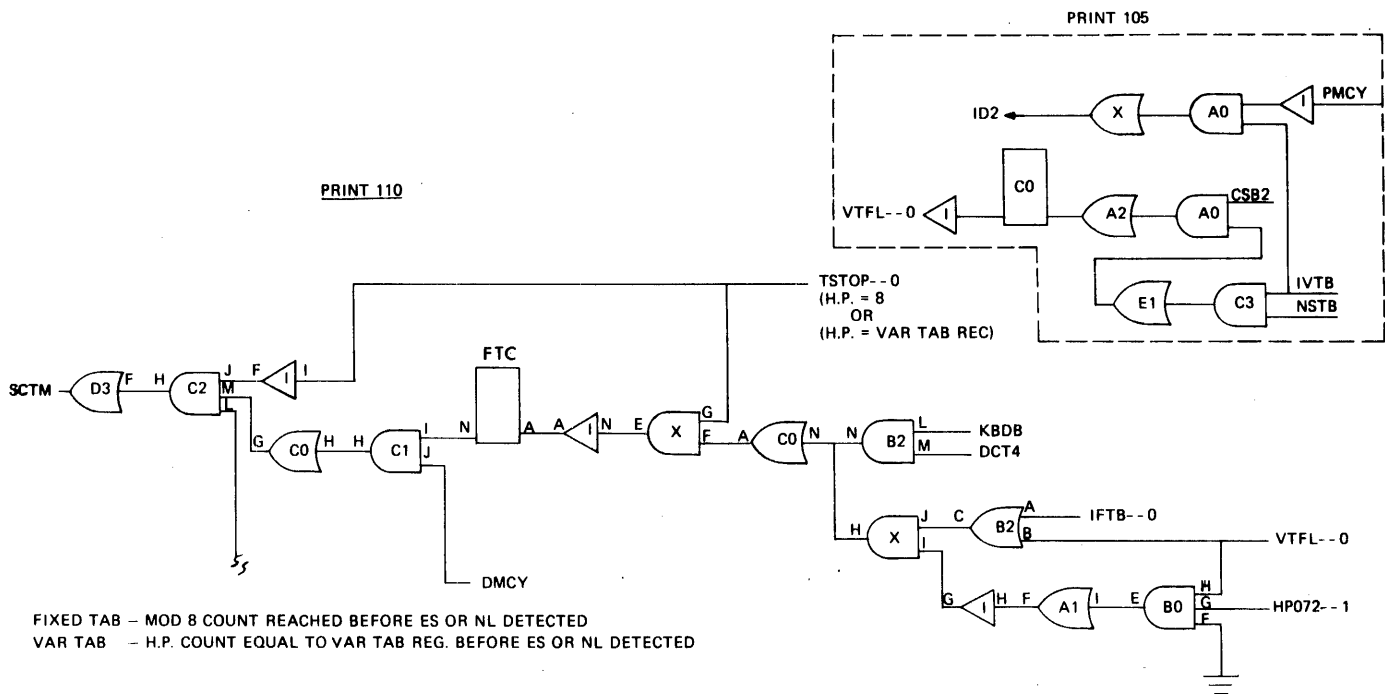


Fig. II-56

DETAILED DESCRIPTION EXAMPLE #5 FIG. II-58  
 PRINT 110, 203, 205, 224

Display cursor (DCT4) occurring with HP072 and IFTB generates SNLK, setting the NLC Flip-flop (205). NLC returns to 110 to be gated with SSL and I1CS Flip-flop, (SSL occurs with the next GET sequence for this monitor) in turn setting the MCEL Flip-flop. Because the MCEL Flip-flop is set, if a NL character is detected (EOL) or H.P. of 79 reached (LDCD) during the display sequence, the level NLT is generated at MT4 time. NLT is used on 203 to generate CTM (Memory Add Register to Cursor Add Register). Because NLT must set 2 flip-flops on 203 before developing CTM, the Memory Add Register will have been upcounted one more count before the CTM occurs to gate and clock the Memory Add Register to the Cursor Add Register. The result is that the Cursor Add Register ends up one count beyond the NL character and/or one count beyond the memory address corresponding to HP of 79. In either case, this places the cursor at position 1 of the next line.

VARIABLE TAB OPERATION

GENERAL DESCRIPTION PRINTS 104, 105

The Variable Tab Storage Register consists of 10 flip-flops. Six of these (V1-VS6) have as inputs H.P.1 – H.P.6 and/or MIR1 – MIR6. The MIR inputs are used during the GET sequence to transfer the variable tab address from the reserved location of memory for that particular monitor, to the Variable Tab Register

(VS1-VS6). The H.P. inputs are for placing new variable tab addresses into the register. These 6 flip-flops can, as a result, contain a maximum count of 64, which is equivalent to positions 65 on the screen. The remaining 4 flip-flops are divided with one flip-flop assigned to each of the four possible monitors a system can have. These Flip-Flops VS17, VS27, VS37, VS47 are necessary because a variable tab address can go beyond an H.P. count of 64 (H.P. 7 is set). Therefore the single reserved memory location reserved for the variable tab address consists of 6 bits, it can store a maximum count of only 64 (Position 65 on the screen) there, when the Put sequence occurs, it is only the contents of the VS1 – VS6 Flip-flops that are stored. The VS7 Flip-flops are not affected by the Put/Get sequence and are only changed when the Variable Tab Address is changed to some address that would also require a change in the VS7 Flip-flop.

VTAB SET

The VTAB SET key causes the count in the H.P. counter that corresponds to the cursor position to be placed in the Variable Tab Register.

DETAILED DESCRIPTION (PRINT 105)

Flip-flop SVTBS (Set Var Tab Storage -105) is set as a result of Var Tab set key being depressed. At DCUR (Display Cursor) time, RTBS (Reset Tab Storage) resets the appropriate VS7 Flip-flop, and RVTS (Reset Variable Tab Storage) resets the VS1 – VS6 Flip-flops. One clock later,

### Functional Detail

Flip-flop CO pin N is set producing TVTHP (To Var Tab From H.P.) to gate the contents of the H.P. counter to the Variable Tab Register (105).

#### VAR TAB

##### GENERAL DESCRIPTION

Depressing the Var Tab key initiates a search, during the Display Sequence, beginning at the cursor location and ending when the H.P. counter is equal to the Variable Tab Register. At that time, the contents of the Memory Add Register are transferred to the Cursor Add Register thereby causing future Display Sequences to display the cursor at the variable tab location. If, during the search, a NL/ES character is detected, before the H.P. becomes equal to the Variable Tab Register, the contents of the Memory Add Register (which contains the NL/ES address), are transferred to the Cursor Add Register and a Push By 1 Replace performed to replace the NL/ES with a VT character and push the NL/ES down 1 position. The Push By 1 Replace then counts the Cursor Add Register equal to the new NL/ES location. During the next Display Sequence, upon detecting the VT character, the H.P. counter is immediately set to the count in the Variable Tab Register, which instantly moves the CRT beam to the corresponding screen location. It is at this position that the cursor is then displayed. If the cursor is located equal to or beyond the Variable Tab Register when the VTAB Key is depressed, the search will be initiated as normal, but because now either NL, or HP = 80 is detected, the Memory Add Register corresponding to the following H.P. count (a zero count) will be placed into the Cursor Add Register, resulting in the cursor being moved to position 1 of the next line. If an ES is detected, a Push By 1 Replace is initiated, replacing the ES with NL and pushing the ES down 1 position, at which time the Cursor Add Register is made equal to the ES location.

##### DETAILED DESCRIPTION – H.P. COUNTER COUNTED EQUAL TO VARIABLE TAB REGISTER – FIG. II-56 PRINT 105, 110

This operation is identical to the Fixed Tab Mod 8 Reached Before NL or ES Detected in all respects but the following:

1. Nor gates B2 output pin C (110) is enabled by VTFL-0 (Variable Tab Function Load) instead of IFTB (Input Fixed Tab). VTFL-0 is developed on 105 by IVTB (Input Variable Tab) and CSB2 (Cursor Position Less Than Variable Tab count).
2. TSTOP (110) is developed by H.P. = Variable Tab Register rather than H.P. = 8.

For further explanation, see the related fixed tab detailed description.

##### DETAILED DESCRIPTION – ES/NL DETECTED BEFORE H.P. COUNTERS COUNTED EQUAL TO VARIABLE TAB REGISTER FIG. II-57 PRINT 103, 105, 110, 121, 202

This operation is identical to the Fixed Tab ES Detected Before Mod 8 Count Reached with the following exception:

1. Nor gate B2 output Pin C enabled by VTFL-0 (Variable Tab Function Load -105)
2. The Push By 1 Replace that is initiated replaces the ES character with the VT character (From the display reg) rather than the FT character.
3. When the following Display Sequence begins, the sensing of the V.T. character causes the contents of the variable tab register to be clocked into the H.P. counter (103 and 202) thereby moving the CRT beam to the variable tab location. Simultaneously, sensing of the VT character triggers the flyback logic on 121. This is necessary because, depending on the cursor position and the variable tab position, the CRT beam might have to move the entire width of the screen, during which time, numerous display memory cycles occur. While the beam is moving, the Display Sequence memory cycles continue, but the Flyback logic prevents the memory add register from being upcounted until the flyback time has been completed, thereby insuring that information in memory has not been passed over during the time the beam is moving.

The remainder of the operation is identical to the related fixed tab operation.

##### DETAILED DESCRIPTION – CURSOR LOCATED BEYOND THE VAR TAB REG AND NL, OR H.P. 79 DETECTED. FIG. II-58 PRINT 105, 110, 205.

This operation is identical to the Fixed Tab with Cursor beyond H.P. 72 and NL or H.P. of 79 Reached with the following exception:

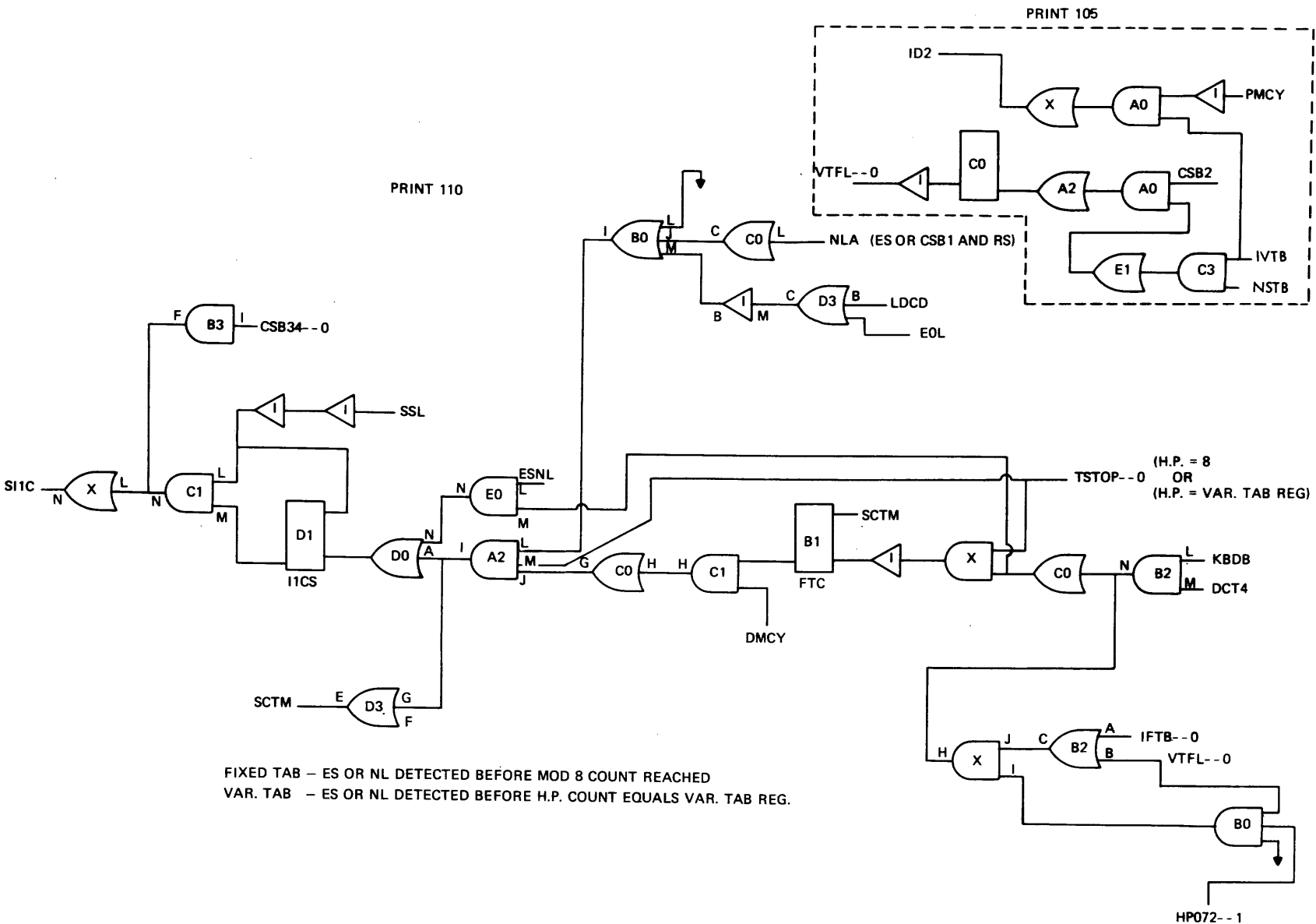
The NLC Flip-flop (205) is set by TGNL (105). TGNL is a function of IVTB and CSB2/ (Cursor Position Higher Than Variable Tab. Register). The remainder of the operation is identical.

##### DETAILED DESCRIPTION – CURSOR LOCATED BEYOND THE VARIABLE TAB REGISTER AND AN ES IS DETECTED. FIG. II-58, PRINT 105, 110, 205

This operation is identical to the Fixed Tab With Cursor Beyond H.P. 72 and ES Detected with the exception that the NLC Flip-Flop (205) is set by TGNL (105), a function of IVTB and CSB2/.

The remainder of the operation is identical.

Functional Detail



FIXED TAB - ES OR NL DETECTED BEFORE MOD 8 COUNT REACHED  
 VAR. TAB - ES OR NL DETECTED BEFORE H.P. COUNT EQUALS VAR. TAB REG.

Fig. II-57

Functional Detail

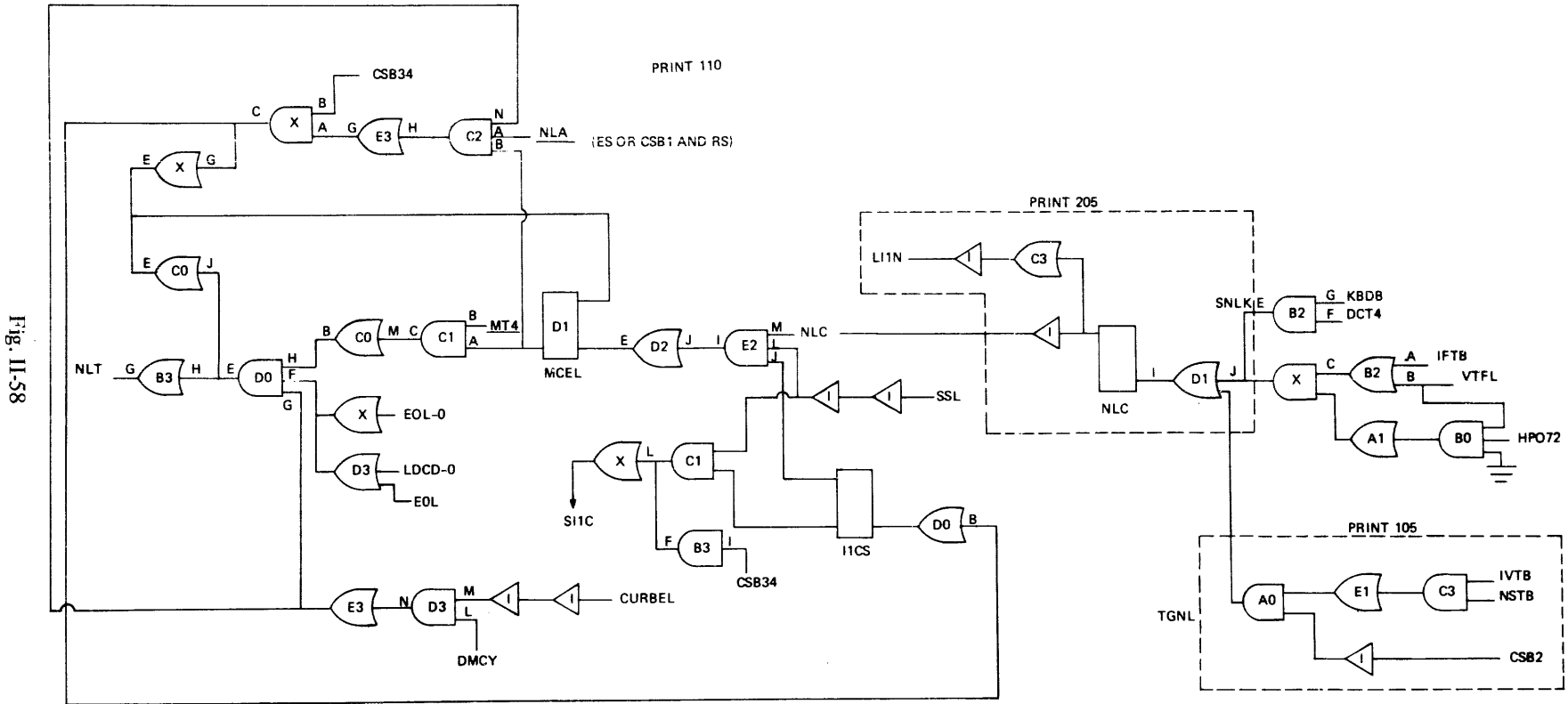


Fig. II-58

- FIXED TAB { CURSOR BEYOND H.P. OF 72 AND ES DETECTED  
AND/OR
- { CURSOR BEYOND H.P. OF 72 AND NL OR H.P. 79 REACHED
- VAR. TAB { CURSOR BEYOND VAR TAB REG. AND ES DETECTED
- { CURSOR BEYOND VAR TAB REG. AND NL OR H.P. 79 DETECTED

Functional Detail

TAB INSERT

GENERAL DESCRIPITON

Tab Insert causes a Push By 1 Replace sequence to be performed. The character in memory at the cursor address is replaced by a F.T. (Fixed Tab) character and the character formerly at that location and everything following, up to and including ES, is pushed down by 1 position. The cursor is not upcounted at the end of the push operation however, the effect as seen on the screen is

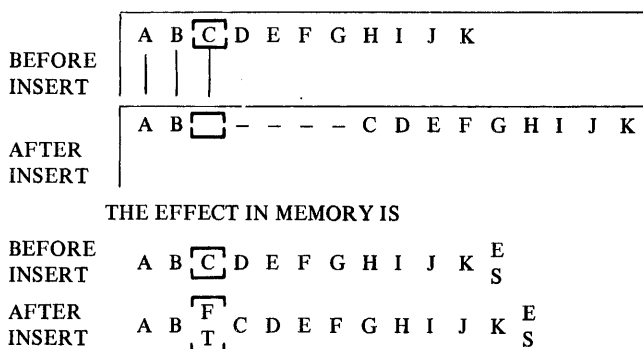


Fig. II-59

If the cursor is beyond a H.P. count of 72 (HP072), or the memory is full, the tab insert is inhibited.

DETAILED DESCRIPTION - PRINT 114, 108, 223, 224

Because the Tab Insert key is depressed, a F.T. character is forced onto the ID1-6 lines (114) and ITIN (Input Tab Insert) generated. ITIN, gated with CSB8/ (cursor not beyond H.P. 72) and CSB3/ (Memory not full) generates INSI (108) which in turn generates LIIN (205). LIIN is used on 223 to develop SI1C to set the I1CCF (224), initiating a Push By 1 Replace edit sequence. LIIN also develops ICC (Inhibit Cursor Count) which sets the ICC Flip-flop on 110. With the flip-flop set, when the Push By 1 Replace is completed, the Cursor Add Register cannot be upcounted. When the display sequence begins, and the F.T. is detected, the H.P. counter is immediately set to the next mod 8 count, (See fixed tab explanation), instantly moving the CRT beam to the corresponding screen location. As the display sequence continues, and as the characters are read from memory, they are placed on the screen beginning at this new location.

TAB DELETE

Tab delete functions only when the cursor is positioned over a F.T. or V.T. character other than recognizing only F.T. or V.T. characters, the operation is identical to a

Pull By 1 Replace sequence. The tab character at the cursor location is read out and discarded, and everything following in memory is pulled back 1 position. The effect as seen on the screen is:

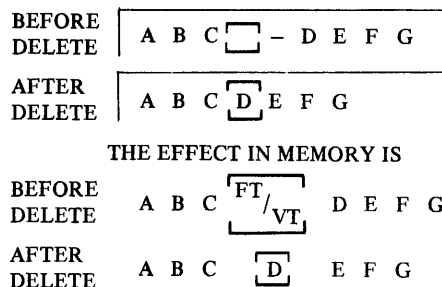


Fig. II-60

DETAILED DESCRIPTION - PRINT 114, 110, 205

ITDL (Input Tab Delete 114) the decode of the Tab Delete character develops DTAB-0 (110) if CSB11 (cursor over FT or VT char) is true. DTAB is used on 205 to set the SCDL (Single Character Delete Flip-flop) and initiate a Pull By 1 Replace sequence. The operation from this point on is a basic Pull By 1 Replace operation. If the cursor is not over a FT/VT character, KBLC is developed (110) immediately, terminating the operation.

NEW LINE

Depressing the NL key repositions the cursor to position 1 of the next line. If, however, the cursor is on line 25 or is over anything other than ES or RS (See Forms Mode) or, memory is full, the NL key has no effect. The operation is basically a search that takes place during the display sequence, beginning at the cursor location and ending with the detection of NL, HPE80, ES, or RS. Detecting NL or HPE80 causes the contents of the Memory Add Register for the following location to be transferred to the Cursor Add Register, which is equivalent to position 1 of the next line. Detecting ES or RS (in alterable forms mode) causes a Push By 1 Replace to be initiated at the proper time, pushing the ES or RS character down and placing the NL character into that location previously occupied by ES or RS.

DETAILED DESCRIPTION (FIG. II-61 PRINTS 110, 203, 205, 215, 217, 218, 224, 312)

Depressing the N.L. key causes INL (Input New Line) to be developed (312). INL causes the NLC Flip-flop to be set (205), generating NLC and LIIN. Nothing more happens until the next GET sequence for this monitor has occurred, at which time, SSL is generated (215). SSL, gated with

Functional Detail

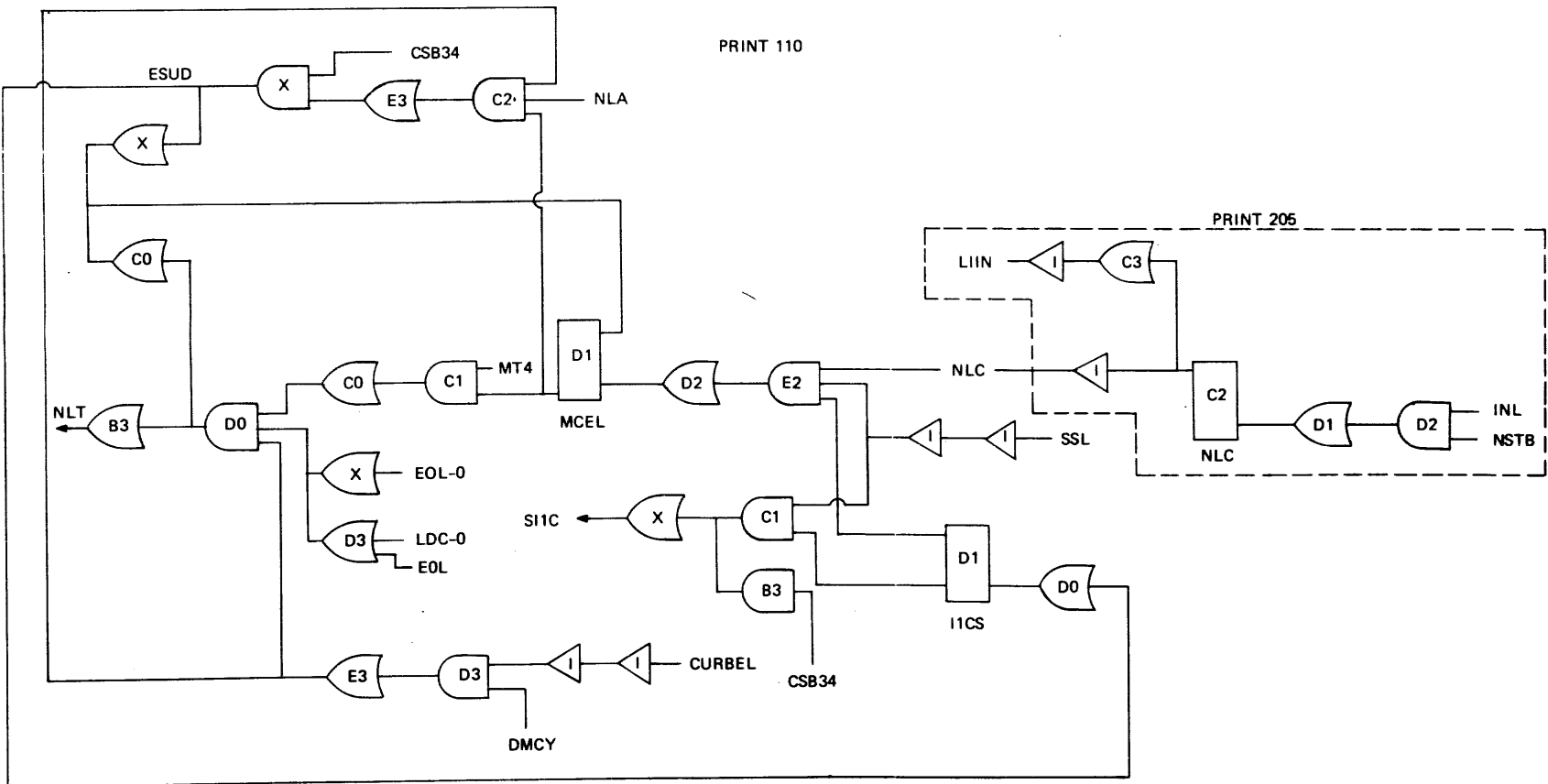


Fig. II-61 NEW LINE DIAGRAM



Functional Detail

NLC, sets the MCEL Flip-flop (110). With the MCEL Flip-flop set, the EOL (NL character), LDCD (HEP80), and NLA (CSB1 & RS or ES) levels are monitored during the next display sequence, beginning with Cursor Compare (CURBEL). Detecting either EOL or LDCD causes NLT to be developed at MT4 time. NLT develops CTM (Memory Add Register to Cursor Add Register -203). Because of the CTM timing, the Memory Add Register is counted up one more time before CTM occurs. CTM then gates and clocks the contents of the Memory Add Register into the Cursor Add Register. The result is that the Cursor Add Register contains the address for the memory location following the NL character, or the character corresponding to HPE80. This is the equivalent of position 1 of the next line.

FORWARD SPACE

Depressing the space bar results in the following:

1. If the cursor is over the NL or ES character (CSB9 or CSB12), a Push By 1 Replace sequence is initiated to push the NL or ES down by 1 position and inserts the Space character in its place.
2. If the cursor is over the ESC portion of a 2 Memory Position Character, the only result is the cursor add register is counted by 2.
3. If the cursor is over any other character, the Cursor Add Register is counted by 1.

DETAILED DESCRIPTION – PRINT 223, 224, 110

With the space bar depressed, the space character in the Keyboard Storage Register (312) and ISP (Input Space) is developed. If the cursor is over either NL or ES, (CSB9 or CSB12) gate AO output I (223) is disabled, resulting in SI1C being developed, setting the I1CFF (224) and initiating a Push By 1 Replace edit sequence.

If the cursor is over the ESC portion of a 2 Memory Position Character, gates AO output I, AO output E, and C1 output H cause CCAB2 and CCAU (CCAU generates CCA on 110) to be developed counting the Cursor Add Register by 2.

If the cursor is over any other character than those already mentioned, gates A0 Output H and C1 Output H

cause CCAU to be developed counting the Cursor Add Register by 1.

FORWARD SPACE – SHIFT, PRINT 223, 224, 110

Depressing of the space bar and shift key results in an operation that is similar to Forward Space without shift, except that in all cases, a Space character is written into memory in place of the character at the cursor location. This because gate A0 output I is held disabled by SFT-0, (Shift). As a result, any one of 3 types of Edit sequences can occur:

1. If the cursor is over ES (CSB12) a Push By 1 Replace edit sequence is initiated to push the ES down and place the Space character into its previous location.
2. If the cursor is over the ESC portion of a 2 Memory Position Character (CSB10), a Pull By 1 Replace edit sequence is initiated to replace the 2 Memory Position Character with the 1 Memory Position Character (Space).
3. If the cursor is over any other character, an Equal Replace edit sequence is initiated to replace the character at the cursor location with the Space character because each of these edit sequences are covered in a different part of section II. No additional explanation is required.

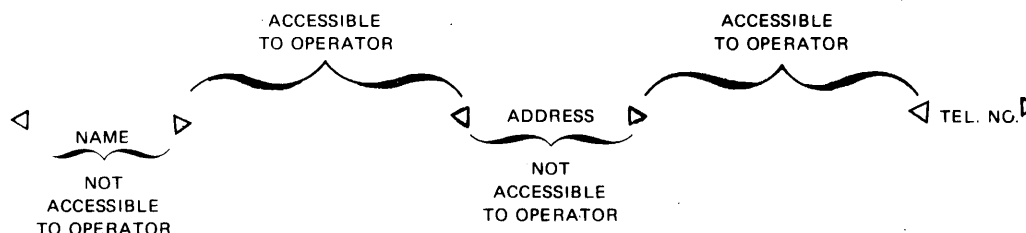
ERASE

The operation resulting from depressing the Erase key and the Shift key is the same in all respects to depressing the Space bar and the Shift key. See Forward Space – Shift for a detailed explanation.

FORMSGENERAL DESCRIPTION

The Forms Option is used in those applications where a previously set up form is sent to the B9353 to be typed on by the operator. As with any form, there are areas where the operator is expected to type in data, and other areas where the operator is not to enter.

EXAMPLE:



### Functional Detail

Those areas not accessible to the operator are between the RS (Record Separator<␣>) symbol and the US (Unit Separator>) symbol. The operator cannot move the cursor into these areas.

In most applications, the form to be used in any particular operation is first set up at the remote device, which in this case is the B9353. To do this, requires that the operator be able to place the necessary RS and US symbols on the screen, and be able to enter the area between them to type in the necessary form information. This is accomplished through the use of a Forms Compose switch located on the control unit. With this switch in the ON position, the RS/US key is active, allowing those characters to be placed on the screen. The logic which prevents the cursor from being placed between these symbols is disabled during this time. Thus, an entire form can be set up and sent to the master station. Having sent the form, the Forms Compare switch is turned off, and when that form is sent back in future operations, because the Forms Compare switch is off, the operator will only have access to those areas not between the RS and US symbols.

The B9353 uses two types of Forms, Unalterable and Alterable. Unalterable form means that those areas of the form accessible to the operator are unalterable in size. Unalterable forms is identified in the control unit by an RS symbol in screen position one of the particular monitor displaying the form. If an attempt is made to position the cursor over the RS symbol, the cursor is immediately advanced to one location beyond the next US symbol.. Attempting to backspace over the US symbol results in the cursor remaining at its original location.

Alterable form means that those areas of the form accessible to the operation are alterable in size to fit the data being typed in. Alterable forms is identified in the Control Unit by a US symbol in screen position one of the particular monitor displaying the form.

If an attempt is made to position the cursor over the RS symbol, the RS symbol, and everything after it on the screen is shifted down by one position. As a result, that part of the form used by the operator expands to handle the extra data being typed in. Should the operator want to advance to the next accessible area (following the next US symbol), merely requires depressing the Fixed Tab key. This advances the cursor to one position beyond the next US symbol.

Attempting to backspace over the US symbol results in the cursor remaining at its location.

#### DETAILED DESCRIPTION – (PRINT 107)

##### Use of Forms Compose Switch

Placing a form on the screen from the keyboard requires switching the Forms Compose switch on (Print

1725 8112 – Control Unit Cabinet Power). This causes the term Forms – 1 to be true conditioning gates E3-I and E3-C on 107. Because these gates are conditioned, if the RS/US key is depressed, the levels IRS (Input RS) or IUS (Input US) are developed. These levels are used on 114 to place the RS or US bit configuration on the ID1-6 lines to then be placed in memory, and the operation proceeds as in any edit sequence. The level FORMS – 1 is also used on 107 to disable that logic which prevents the operator from moving the cursor between the RS and US symbols.

##### Unalterable Forms

At the beginning of each monitors display sequence, the STD pulse sets the FDC (First Display Cycle) Flip-flop on 107. This Flip-flop is reset at MT3 time of the first display cycle and is not set again until the next STD pulse occurs. If an RS symbol is detected while the FDC Flip-flop is set, it means the RS symbol is in screen position one. The level SCS5 is then generated, setting the Control Storage 5 Flip-flop, to indicated Unalterable Forms for that particular monitor.

If the cursor is positioned over the RS symbol during the edit sequence the following occurs during the following Display Sequence.

1. Gate B1-C is enabled (RS character in MIR, during Cursor display time).
2. Gate X – C is enabled (Forms Compose switch off, and MT3 of a display memory cycle).

The output of these gates, gated with CSB5 (Unalterable Forms), causes the FTSI Flip-flop to be set. Because display memory cycles continues occurring, and the Memory Add Register is being counted up, in time, the following US symbol is detected in MIR. This enables gate D1-H at the output of the FTSI Flip-flop, in turn causing the SCTM Flip-flop to be set, generating CTM (Transfer Memory Add Register to Cursor Add Register). CTM causes the address of the US character to be gated and clocked into the Cursor Add Register. Because the cursor should end up one location beyond the US symbol, the output of the SCTM Flip-flop causes the FCC Flip-flop to be set, generating CCAU (CCAU generates CCA-110), causing the Cursor Add Register to be given on additional count. This causes the cursor to be positioned one location beyond the US symbol. Should an attempt be made to backspace the cursor into a form (Over the US symbol), the following occurs (See Figure II-62):

1. The RSS Flip-flop is set when the RS symbol is detected during the display sequence following the edit sequence where the backspace was initiated.
2. Because the backspace edit sequence actually does count the cursor add register to where it is at the same address as the US symbol, DCUR (Display Cursor) occurs with the US character in MIR. The combination of DCUR, US symbol in MIR, and the

Functional Detail

- RSS Flip-flop set therefore causes the FTSI Flip-flop to be set at MT3 of the US symbol.
3. The setting of the FTSI Flip-flop conditions gate D1 pin J but because of the timing, gate D1 input pin I goes false at the same time that D1 input pin J goes true. However, because the cursor is actually displayed over the US character during this first display sequence, the Memory Add Register is not advanced, and the US symbol is read from memory twice (See Display Sequence and Rope Timing). It is during MT3 of the second memory cycle for the US symbol, therefore that gate D1 output pin H is enabled.
  4. The output of gate D1-4 causes the SCTM Flip-flop to be set, generating CTM to gate and clock the contents of the Memory Add Register into the Cursor Add Register. (Both registers contain the same address at this time).
  5. The output of the SCTM Flip-flop causes the FCC Flip-flop to be set, generating CCAU (CCAU in turn generates CCA — 110) to count the Cursor Add Register one additional count, effectively placing it back in its original location to the right of the US symbol.

The following keyboard functions are disabled with an Unalterable Form on the screen:

- IVTB (Input Variable Tab)
- IINS (Character and Line Insert)
- ICDL (Input Character Delete)
- ILLD (Input Line Delete)

Any of these levels being decoded with CSB5 set immediately causes KBLC (Keyboard Load Complete — 107) to be developed, thereby effectively terminating the action before it has even been allowed to begin.

**Alterable Forms**

As in Unalterable Forms, the FDC (First Display Cycle) is set by the STD pulse, and reset at MT3 of the first display memory cycle. If a US symbol is detected during the time the FDC Flip-flop is set, it means the US symbol is in screen position one. The result is that the level SCS1 (Set Control Storage 1) is generated, setting the CSB1 Flip-flop, indicating Alterable Forms for that particular monitor.

If during a subsequent edit sequence the cursor should be positioned over the RS symbol, the following display sequence will cause the level FSC12 (Forms Set Control Storage 12) to be generated. CSB12 Flip-flop is set whenever the cursor is over an RS symbol in Alterable Forms mode, or over an ES character. Because CSB1 is set (Alterable Forms mode), the logic for moving the cursor to one position beyond the US symbol is disabled (Gate BO output pin N is disabled). As a result, any future edit sequence which attempts to move the cursor forward into the Forms area is handled in a manner similar to a Cursor over ES operation. If the character to be placed in memory is a one memory position character, SI1C (223) is generated, placing the keyboard character into memory in place of the RS character and everything after it, up to and including the ES character to be pushed down one memory

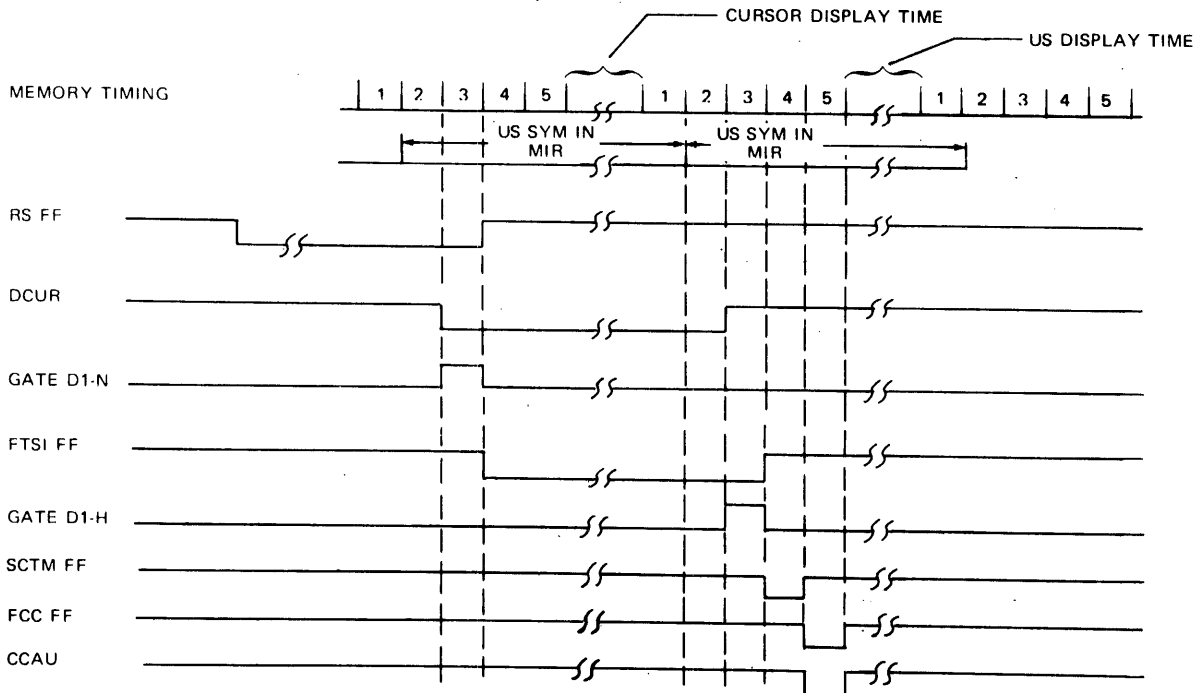


Fig. II-62

Functional Detail

position. If the incoming character is a two memory position character, SI2C is generated pushing the RS and everything after it down two positions.

Backspacing into a form (over the US symbol) is handled in the same manner as in Unalterable Forms.

Depressing the Fixed Tab key while in Alterable Forms moves the cursor to one position beyond the next US symbol. This is accomplished by having the decode of the Fixed Tab character (312) set the TABS Flip-flop (107). The TABS Flip-flop in turn sets the RSS Flip-flop. When DCUR is generated during the following display sequence, the FTSI Flip-flop is set. Upon detecting the next US symbol, FTSI gated with the US symbol in MIR sets the SCTM Flip-flop, generating CTM, to gate and clock the Memory Add Register contents into the Cursor Add Register.

The SCTM Flip-flop in turn sets the FCC Flip-flop causing the Cursor Add Register to be counted one additional count, placing the cursor one position beyond the US symbol.

SELECT

The I/O logic operates under the control of the PC (Processing Condition) Flip-flops. The state of these Flip-flops determines the actions the Control Unit is to perform.

To clarify the detailed description, it is divided into the three following categories:

1.     E A A    E  
   Receive O D D q N and send ACK or NAK.  
          T 1 2    Q
  
2.     S  
   Receive T T E X T  
          X
  
3.     E B  
   Receive T C and send ACK or NAK.  
          X C

As data is received from the line, it is assembled or stacked serially into the I/O Register. From the I/O register it is paralleled into the I/O buffer where it is converted from a 7 bit ASCII code to a 6 bit internal code for placement into memory. The output of the I/O buffer is

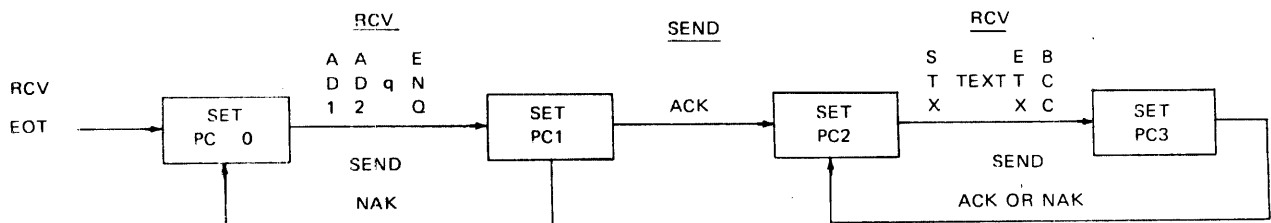
also sampled by a decoder network where it is determined if the character in the buffer register is a Control Character (STX, EOT, ETX, etc.) or possibly a Format Character (LF, CR, BS, etc.). If the character is determined to be an STX character, everything after it, up to, but not including the ETX is written into memory.

**DETAILED DESCRIPTION**

E A A    E  
Receive O D D q N and respond with ACK or NAK.  
T 1 2    Q Figure II-64, 65, 66, 67 and 70 and related prints.

The following description relies heavily on Fig. II-64. Assuming this is an asynchronous device, the detection of the start bit on the BB line removes the unconditional reset of the binary count on print 209, allowing it to begin counting. This counter is used to generate the pulses necessary to strobe the incoming data into the I/O register. Through the use of jumpers, the counters output can be varied to allow the Control Unit to communicate at bit rates of from 150 BPS up to 38,400 BPS. The output of the counter is a pulse called BITTME. This pulse is basically a square wave, whose repetition rate is equal to the bit rate of the incoming data. The counter runs only long enough to assemble one character in the I/O register at which time the unconditional reset is again applied.

The BITTME pulses are used on Print 210 to develop SLIO (Shift left I/O) pulses and IOCLK (I/O clock) pulses (see Fig. II-66 and 67). These pulses strobe the incoming data into Flip-flop C1, and from C1 into I/07 - I/01, and Flip-flop CC (Complete Character). These Flip-flops stand by in a set state when containing no data. When the start bit of a character is detected, I/07 is reset thereby effectively containing the flag. As the remaining bits of the character are assembled, the flag bit is shifted through the I/O register into Flip-flop CC, resetting it, and generating CCF (Complete Character Finish). CCF generates TIOIOB (Transfer I/O Register to I/O Buffer) on Print 210, which in turn gates and clocks the output of the I/O Register into the I/O Buffer. TIOIOB simultaneously causes the CP (Character Present) Flip-flop on Print 317 to be set, indicating that the I/O buffer now contains a character. The output of the I/O buffer is sampled by the I/O decoder (Print 313) and assuming this is the EOT character the level



Functional Detail

EOTD is developed, generating DCCLR (Data Com Clear) on 317. DCCLR unconditionally clears the PC Flip-Flops (Print 315) to a count of PC = 0. With the PC Flip-flops equaling zero, the Control Unit is ready to receive the rest of the incoming Select message.

The next character received is AD1, the first of the two address characters. This character, when seated in the I/O buffer, is sampled by the Selection Address card (Print 320). This card contains the wired-in address for this particular unit. If comparison exists between the AD1 character in the I/O buffer and the wired in AD1 address, Flip-flop AD1 is set, conditioning the set input of Flip-flop AD2. When the AD2 character is received, and comparison is found, the AD2 Flip-flop is set.

At this point a brief explanation of the AD1 and AD2 characters becomes necessary. The AD1 character deals strictly with the first character of the station address, and can be alpha or numeric. The AD2, however, is determined by the station address, the screen address, and whether or not the printer is called on.

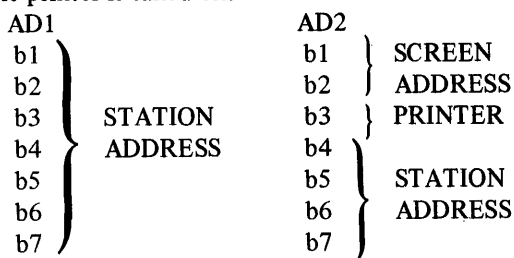


Fig. II-63

As shown in Fig. II-63, bits 1, 2 and 3 for the AD2 character are variable for a particular station. It is only bits 4, 5, 6 and 7 that are wired in on card 320, that comparison is made for. As a result, the AD2 character can have eight possible combinations (0-7) and still call on the correct station address.

At the time that the AD2 Flip-flop is set, the appropriate ADR1, ADR2 Flip-flops are also set, as determined by bits 1 and 2 of the AD2 character.

Setting the AD2 Flip-flop attempts to make SPC1EN (Set PC1 enable) go false, but because SPC1EN is anded with SPC1EN, logic for the q and ENQ characters, SPC1EN cannot go false at this time.

Upon receiving the q and ENQ characters, the level SPC1EN goes false, setting the PC1 Flip-flop on 315. With PC1 set, the Control Unit is ready to respond. If card 319 (Identification Address) is in the unit, the response is A A A A A N  
D D C or D D A. Without card 319 the response is simply 1 2 K 1 2 K  
ACK or NAK.

PC1 causes XMITC (Transmit Character) to be developed, in turn setting the XMIT Flip-flop. The output of the XMIT Flip-flop generates CAD (Request to Send) to

the data set, and conditions the input of the bit timer (Print 209). PC1 also causes XRQRSP4 (Transmit Request Response 4) to be generated. If card 319 is not present, XRQRSP4 goes false immediately. If card 319 is present, XRQRSP4 goes false only after AD1 and AD2 have been generated. XRQRSP4 going false, gated with WAITK (true if unit in receive mode), causes the bit configuration for ACK or NAK to be placed on the Output Data Translator lines (ODTB1 – ODTB7). The response on the ODTB1 – 7 lines is gated and clocked into the I/O Buffer by TODTIOB (Transfer Output Data Translator to I/O Buffer). TODTIOB also sets the CP (Character Present) Flip-flop on 317, indicating the I/O Buffer now contains a character.

The Control Unit now idles, waiting for CBR (Clear to Send) to be received from the data set. CBR, gated with the output of the XMIT Flip-flop, allows the bit timer to begin developing BITTME (Bit Time Pulses). Because CP indicates a character is in the I/O buffer, the level TIOBIO (Transfer I/O Buffer to I/O Register) is developed to gate and clock the character in the I/O buffer into the I/O register. (See Fig. II-66 and 67.)

Shortly after, SLIO and IOCLK pulses begin shifting the character from the I/O register onto the BA line to the data set. TIOBIO also generates RCP (Reset Character Present Flip-flop) and XRQRSPA (Transmit Request Response A).

If the NAK character is sent, XRQRSPA, gated with NAKD from the I/O decoder, generates RPC1 (Print 318) resetting PC1 and placing PC's = 0, and RCVAlM, sounding the receive alarm. This occurs while the NAK is being transmitted. If the ACK is sent XRQRSPA, gated with ACKD, generates SRPC (Shift Right PC), setting PC2 and resetting PC1. Also, if ACKD is decoded, it means the control unit is ready to receive data after the ACK is transmitted, so SRM (Start Receive Mode) is developed. SRM will allow memory cycles to be developed for storing data when the STX is detected in the coming message.

If the Control Unit contains card 319 (Identification Address), the XMITC term releases the unconditional reset from the character counter Flip-flops CC1 – CC4. This counter, as it counts from CC0, (all reset) to CC4 (it is upcounted by TODTIOB each time I/O buffer is loaded) normally generates SOH (CC0), AD1 (CC1), AD2 (CC2) STX (CC3) TEXT (CC4) for use in a polling operation.

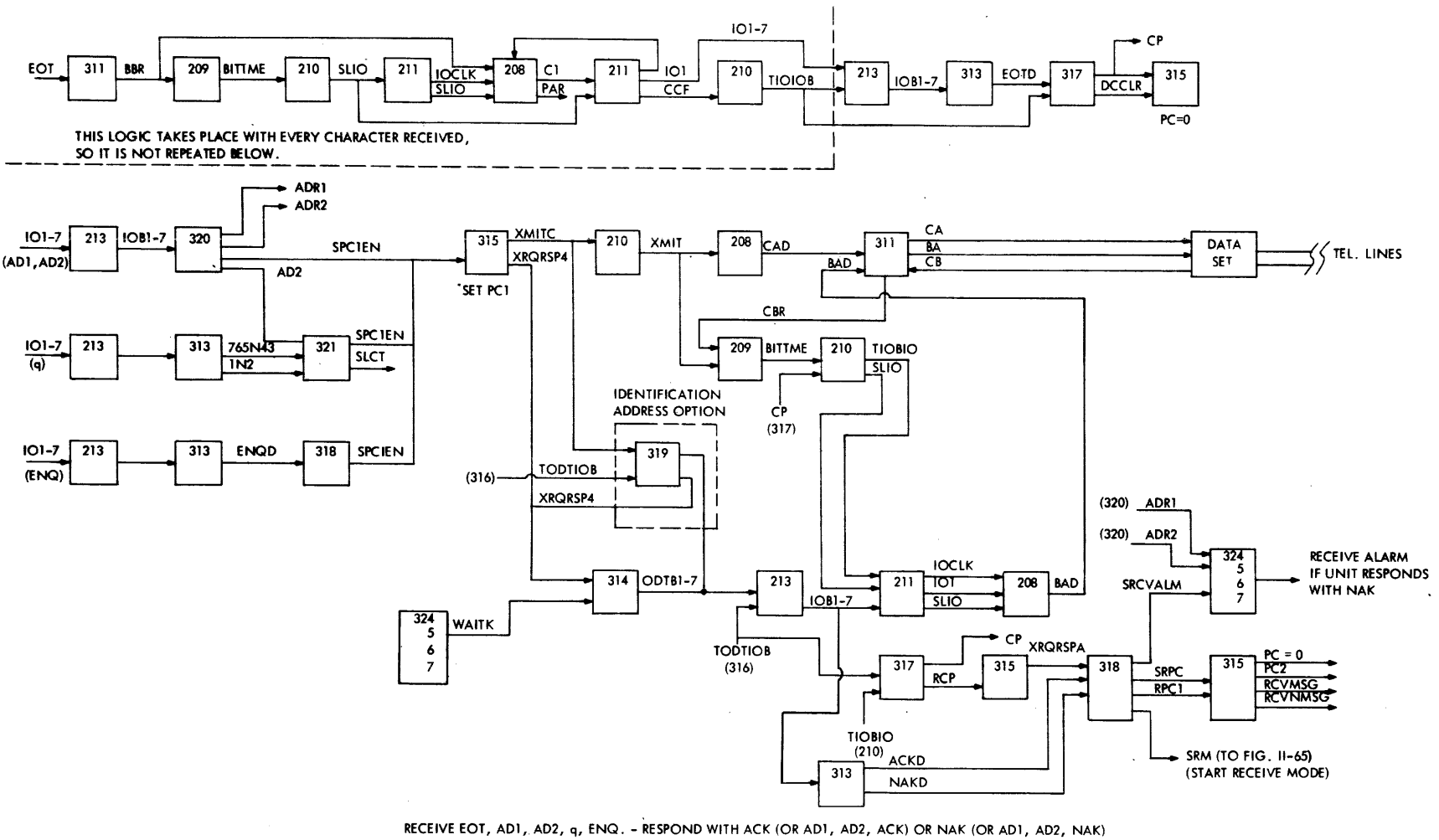
A A A A A N

Because this response is to be D D C or D D A, SOH and 1 2 K 1 2 K

STX are not generated at CC0, because XMSG and SEND are not true (Only for Transmit), gate B2-1 is not enabled, thereby not generating SOH. At CC1 and CC2, because XMITC is true, the wired in AD1 and AD2 characters are placed on the ODTB1-7 lines to be gated and clocked into the I/O buffer. At CC3, because SEND is not true, STXEA (STX enable) is not generated. At CC4, the level XRQRSP4 goes false, allowing the ACK or NAK to be placed on the

Functional Detail

Fig. II-64 RECEIVE EOT, AD1, AD2, q, ENQ. - RESPOND WITH ACK (OR AD1, AD2, ACK) OR NAK (OR AD1, AD2, NAK)



Functional Detail

ODTB1-7 lines from Print 314.

|           |                              |
|-----------|------------------------------|
| S         | S A A S                      |
| Receive T | O D D T Text. Fig. II-65 and |
| X         | H 1 2 X related prints       |

The incoming data is in either of the formats shown in Fig. II-65 and related prints. In either case, it is only the data following the STX character that is placed in memory. Developing of the BCC (Block Check Character) is on everything following SOH or STX, whichever occurs first. Upon detecting the STX character, two memory cycles are initiated. These memory cycles read out the cursor address from the reserved core assigned to the particular screen being addressed. This cursor address is then placed into the PMA (Processor Memory Address, Register - Print 219). All future addressing is then handled by the PMA register.

Assuming the SOH character is the first character received, the level SOHD from Print 313 removes the unconditional reset from the Block Check Register (Print 212), allowing BCC to be developed on everything following the STX character, up to and including the ETX character. Having received the AD1 and AD2 characters, the STX is received and on Print 313 generates STXD. STXD gated with RCVNMSG (Receive Normal Message - function of CP and PC2) sets the LA (Load Allow) Flip-flop on Print 316. LA in turn causes IRQ (Input Request) to be generated on 318 which in turn generates SPMCY (Set Processor Memory Cycle) on Print 111. SPMCY sets the PMCY Flip-flop on 116, initiating the first of the memory cycles to read out reserved core. Flip-flops PRCY1 and PRCY2 (Processor Cycle 1 and 2) on Print 111 serve to handle the X addressing logic and to gate the info read from memory into the proper portions of the PMA register. PRCY1 and the RCV Flip-flop (Print 111) were initially set when SRM was generated earlier.

The Y addressing is a function of the ADR1 and ADR2 Flip-flops (Print 320). The state of these Flip-flops being determined by bits 1 and 2 of the AD2 character in the initial select message.

PRCY1 and PRCY2 also develop MMCA (Memory Most Significant Count Allow - Print 117). MMCA disables APMA (Allow Processor Memory Addressing - Print 116), allowing addressing for these first two cycles to be strictly a function of the PRCY1, PRCY2, ADR1 and ADR2 Flip-flops.

At MT4 of the first cycle, RQCMP (Print 111) is developed, PRCY1 is reset and PRCY2 is set generating SPMCY again to initiate the second of the two memory cycles. At MT4 of the second cycle, PRCY2 is reset, dropping MMCA and allowing all future processor memory cycles to be addressed by the PMA register. RQCMP is also developed again. RQCMP occurs at the end of all processor memory cycles. RQCMP, among other things, resets the CP Flip-flop (Print 317), indicating that the I/O buffer is empty. Because the CP Flip-flop is reset, when the initial

two memory cycles are complete, IRQ is not generated and the STX character which is still in the I/O buffer is not written into memory. (The I/O buffer does not get cleared after each operation, the next character is simply placed over it).

As characters following the STX are received and placed in the I/O buffer, RCVNMSG generates IRQ initiating the memory cycle and TIOBMIN (Transfer I/O Buffer to Memory Input - Print 316) gates the character from the I/O buffer to MIN1 - MIN6, and in turn, by way of Print 111 to the MDG1 - MDG6 lines to be written into memory. It is when the character is gated from the I/O buffer to MIN1 - MIN6 that it is converted from the 7 bit ASCII code to the 6 bit internal code.

If, when the character is received it has incorrect parity, the level PARERR is generated (Print 213), in turn generating ENQME (Enable Question Mark - Print 316). ENQME disables TIOBMIN on Print 316 preventing the character in the I/O buffer from being placed on MIN1 - MIN6. ENQME also forces MIN1 - MIN6 to the bit configuration for a question mark, placing it in memory in place of the character in the I/O buffer.

Should the character received be a character that requires two memory positions, the level SCHARD (Special Character Decode - Print 313) is developed. SCHARD disables TIOBMIN and generates ENBIT3, causing MIN3 to go true, writing ESC into memory (the ESC CHAR is bit 3 on in memory). At MT4 time, RQCMP resets the CP Flip-flop and sets the SCHAR Flip-flop (Print 316). With SCHAR Flip-flop set, TIOBMIN is enabled again and SCHAR causes CP to be set. This generates IRQ, allowing the two memory position character in the I/O buffer to be written into memory along side the ESC character. In the process of doing so, however, the level SCHAR disables MIN3 (print 213) resulting in the character being written into memory with bit 3 off. At MT4 of this second memory cycle, RQCMP resets the SCHAR Flip-flop. If, when an IRQ is generated, an ES is detected during the read portion of the memory cycle (this would be the case with every incoming character if the cursor is over an ES character when the control unit is selected), the following happens:

1. At MT4 of the memory cycle, having written the data character to memory, RQCMP resets CP and sets the GENES Flip-flop on Print 316. The GENES Flip-flop being set disables TIOBMIN.
2. The term GENES generates IRQ on Print 318, initiating another memory cycle.
3. GENES forces the ES configuration onto the MIN1 - MIN6 lines and into the memory location after the last data character.
4. Because the GENES Flip-flop is set, CPMA (Count Processor Memory Address - Print 316) is disabled resulting in the PMA register remaining at the same address count. As a result, the next incoming char-

Functional Detail

acter is written over the ES character, and because the ES is detected during the read portion of the cycle, the same operation repeats. In effect, with the cursor positioned over an ES, each incoming character requires two memory cycles.

E B

Receive T C and send ACK or NAK. Fig. II-68 and X C related prints.

Upon receiving the ETX character, logic is developed for comparing the next character to be received with the character in the BCC Register. If the two compare, and if there have been no vertical parity errors in the message, or

A A A

memory has not been exceeded, an ACK or D D C

1 2 K

response is generated and transmitted and the cursor is updated to one position beyond the last data character. Neither the ETX or the BCC is written into memory.

If comparison does not occur, or there has been a vertical parity error in the message, or memory has been

A A N

exceeded, a NAK or D D A is generated and transmitted,

1 2 K

and the cursor remains at its original position.

Detecting the ETX character on 313 generates ETXD, RLAA (Reset Load Allow) and IRQDIS (Input Request Disable – developed by term N764 on Print 313). RLAA sets the BLKCHK Flip-flop on Print 316, causing NOESCSQ (No Escape Sequence) to go true. Being true, NOESCSQ disables the set input of the CLR Flip-flop (Print 317). This Flip-flop is set only if an EOT is received, and generates DCCLR (Data Com Clear) clearing much of the I/O logic. Because there is the possibility that the BCC character could resemble an EOT, NOESCSQ disables the set input of the Flip-flop.

RLAA resets the LA Flip-flop on Print 316 preventing a memory cycle on the BCC character and anything following. IRQDIS is used on Print 318 to disable IRQ for the ETX character, therefore preventing a memory cycle from being generated for it. ETXD is used to set the XRBC (Transmit or Receive BCC) Flip-flop on Print 212, generating XRBC. RLAA also develops ENDBLK (End, prepare for BCC) on Print 316 which sets the ENDS (End Select) Flip-flop on Print 117, initiating a 16 to 32 millisecond delay. The length of this delay is a result of the FRMO pulses, and can be from 16 to 32 milliseconds, depending on when the ENDS Flip-flop is set in relation to the FRMO pulse. During this delay two things occur. The BCC character is received and compared with the character in the BCC register on Print 212. Also a Display sequence occurs, and if the last incoming data character went into the last position of memory assigned to the particular monitor (this is where the ES character should normally be), the ES character in that position is destroyed. As a

result, when the Display sequence occurs, and no ES character is located, the level EDP occurs while DIP is true (Print 117) setting the LESM (Load End Screen to Memory) Flip-flop. This Flip-flop causes the ES character to be written into memory and also causes the ERR Flip-flop on Print 317 to be set. With the ERR Flip-flop set, a NAK response is generated on Print 314. The ERR Flip-flop is also set if the incoming BCC character did not match the BCC being developed by the B9353. In which case PED (Parity Error Detected – Print 212) is generated setting the ERR Flip-flop.

The ERR Flip-flop can be set if a vertical parity error occurs anywhere in the incoming message. A vertical parity error causes the level PAR (parity) to be generated on Print 208, in turn causing the ERR Flip-flop to be set. The ERR Flip-flop is reset whenever STX of the retransmitted message is detected.

At the end of the 16-32 millisecond delay, and while the level ACURL (Allow Cursor Load – The 13 micro seconds between any Put and Get sequence – Print 203) is present, the level SRPC (Shift Right PC) is generated on Print 210, causing PC3 to be set on Print 315. With PC3 set, XMITC is generated, in turn generating XMIT on Print 210. XMIT causes CAD (Request to Send – Print 208) to be generated. XMIT also conditions the bit timer on Print 209. Upon detecting CBR (Clear to Send – Print 311) the bit timer is allowed to count developing BITTME. BITTME causes TIOBIO (Transfer I/O Buffer to I/O Register – Print 210) to be generated, transferring the ACK or NAK response (or AD1, AD2, ACK or NAK) from the I/O Buffer (Print 213) to the I/O Register (Print 211). BITTME also causes the response to be shifted onto the line. TIOBIO also generates TIOBIOA on Print 318 when ACK or NAK is detected. TIOBIOA causes PC3 to be reset on Print 315, and PC2 to be set. If the response is NAK, nothing further happens.

If the response is ACK, however, it is when PC3 is set that the level MVCUR is generated. MVCUR sets the RCU1 flip-flop (Print 117) generating PMR to initiate the first of two memory cycles to write the contents of the PMA register into the reserved core assigned to that particular monitor. It is for this reason that PC3 is set when ACURL indicates that the 13 micro seconds between any Put and Get sequence is present. Because the Processor memory cycle is second highest in priority, it can interrupt any Display sequence. Should the two Processor memory cycles for writing the PMA register contents into reserved core occur during that particular monitor's Display sequence, the following Put sequence would simply overwrite it. By not letting PC3 be set, and as a result not initiating the two Processor memory cycles until after Put has occurred (the 13 micro seconds between any Put and Get) it insures that the data placed into reserved core will not be destroyed. The X addressing is determined by RCU1 and RCU2 (RCU1 is reset and RCU2 set at MT4 of the first memory



Functional Detail

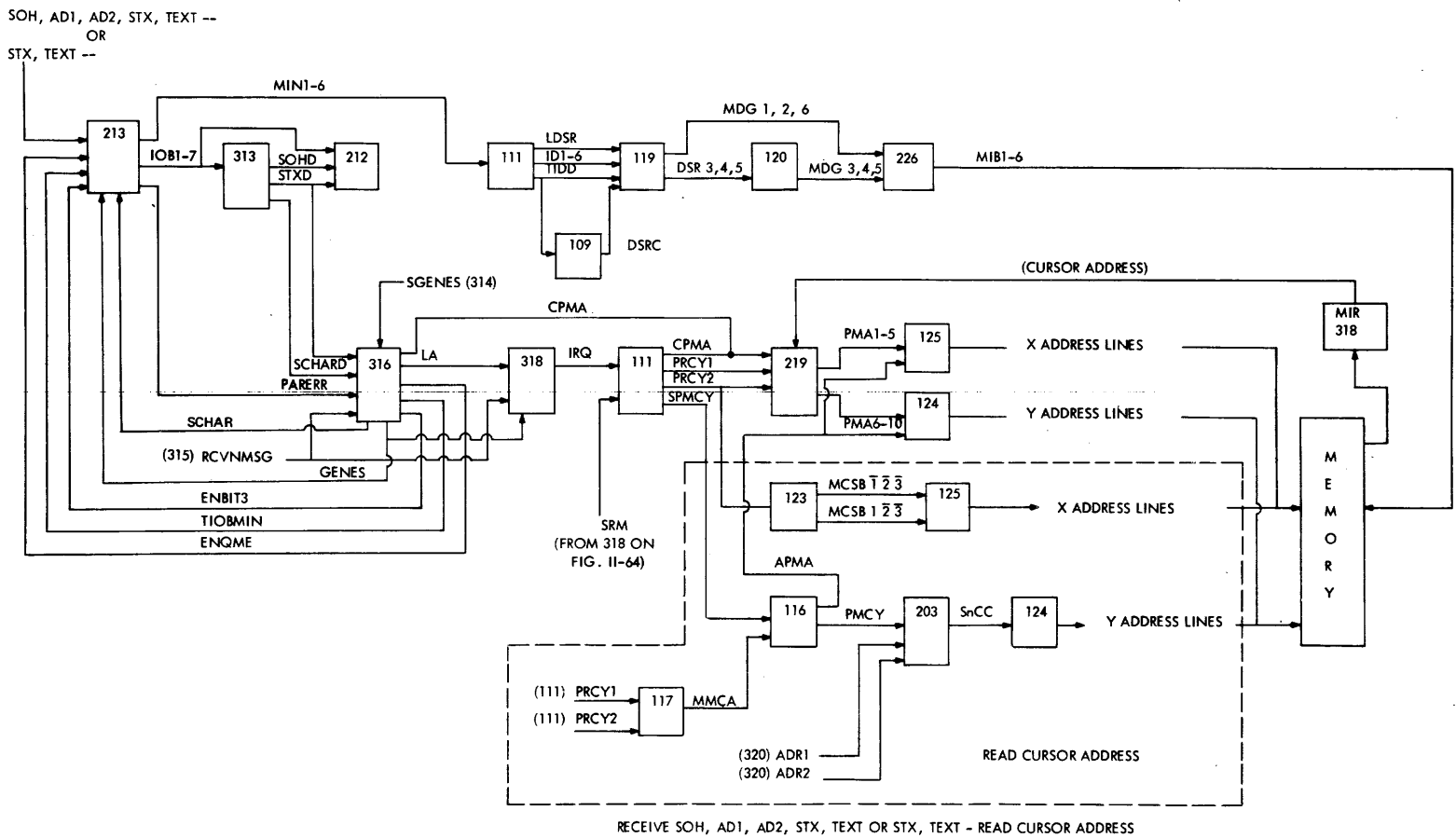


Fig. II-65 RECEIVE SOH, AD1, AD2, STX, TEXT OR STX, TEXT. - READ CURSOR ADDRESS

Functional Detail

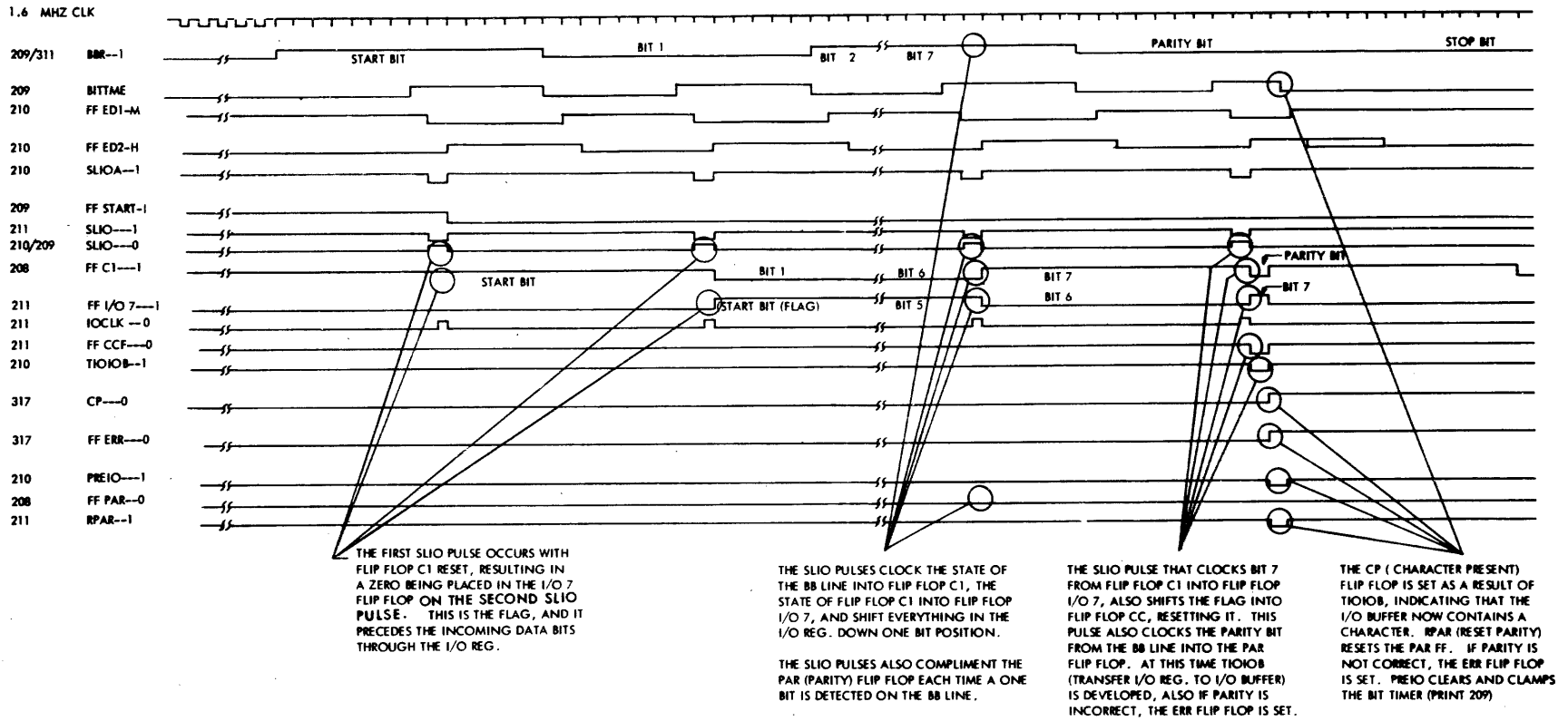


Fig. II-66 I/O REG SHIFT TIMING — RECEIVE

Functional Detail

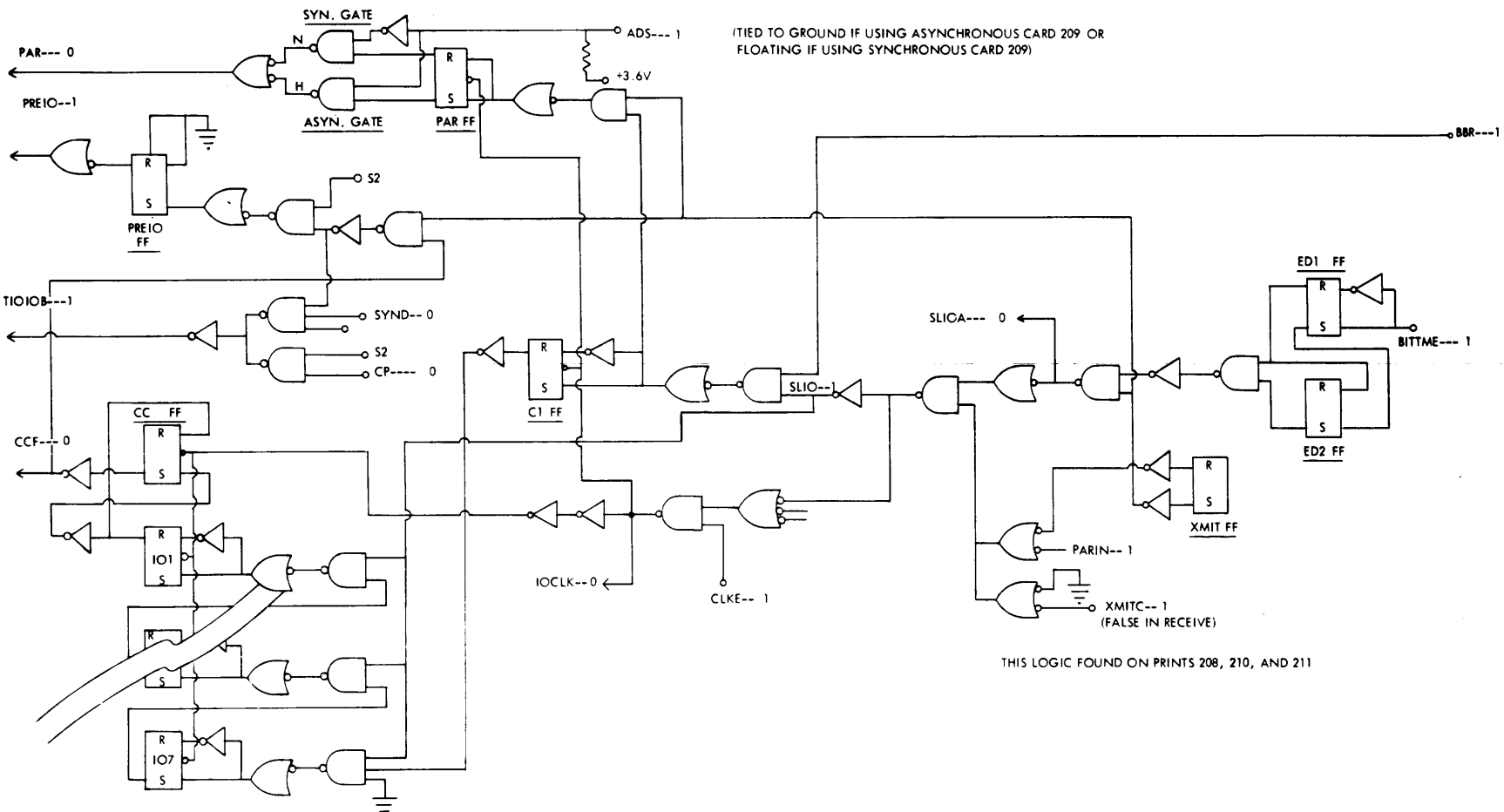


Fig. II-67



Functional Detail

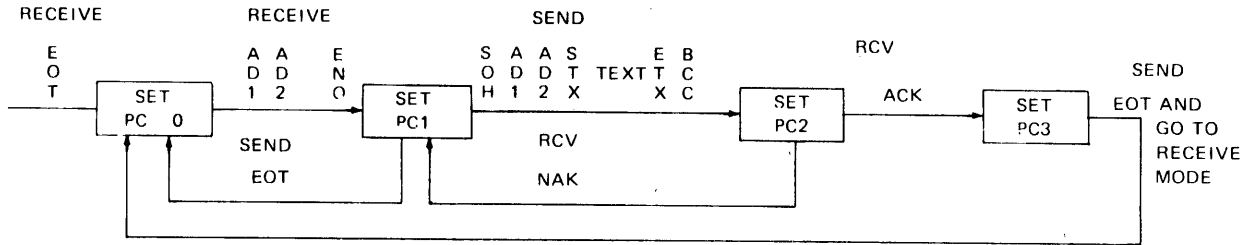


Fig. II-69, 70, 71 and related prints.

cycle). The Y addressing is a function of ADR1 and ADR2, which is determined by the 1 and 2 bits of the AD2 character in the original Select sequence. RCU1 is also used to gate the contents of PMA 1,2,6 and 3,4,5 onto the memory inhibit lines during the first memory cycle. RCU2 is used to gate the contents of PMA 7,8,9 and 10 onto the inhibit lines during the second memory cycle. The level MMCA is generated to inhibit APMA (Allow Processor Memory Addressing) on Print 116 thereby preventing the PMA register from effecting the addressing logic. MMCA is also used on Print 203 to allow ADR1 and ADR2 to develop the Y addressing.

The handling of the five character Polling sequence is virtually identical to the handling of the Select sequenc, except that it is a p (Poll) character instead of the q (Select) character that is detected. (See Select for a detailed explanation.) When the entire five character sequence has been received SPC1EN causes PC1 to be set on Print 315, in turn generating XMITC for the response. XMITC is used on Print 210 to generate XMIT. XMIT in turn causes CAD (Request to Send) to be generated on Print 208. CAD causes the carrier to be turned on in the data set and conditions the binary counter on Print 209. This counter will be allowed to count, producing BITTME pulses when CBR (Clear to Send) is detected (Print 311). The p character, when detected, generates the terms 765N43 and N21 (Print 313). These terms cause SSENDDB to be developed, setting the SEND Flip-flop on Print 315. The term SEND from the SEND Flip-flop is gated with the level SENDK from the particular keyboard/monitor being polled (Print 321). Because the unit is not ready to send, the level SENDK is false, causing the level ODTB3 and TODTIOB to be generated. These terms cause a 3 bit (the configuration for an EOT character) to be gated and clocked into the I/O buffer (Print 213). Because SENDK is false, the level RPC1 is also generated (Print 321), resetting the PC1 Flip-flop on 315. When CBR allows the binary count on Print 209 to count, the first BITTME pulse causes TIOBIO to transfer the EOT from the I/O buffer (Print 213) to the I/O register (Print 211). The BITTME pulses also generate SLIO (Shift Left I/O) pulses on 210. These pulses cause the contents of the I/O register to be shifted into the ODB Flip-flop on Print 208. The output of the ODB Flip-flop is the term BAD, which goes to the interface and on out to the line. As the EOT character is being shifted out, the PAR Flip-flop is being complimented with each one bit, so that proper vertical parity is inserted in the bit 8 position. Asynchronous transmission requires even parity.

POLLING

In a Polling Operation, the Processing Condition Flip-flops control the Control Units actions as shown above.

To clarify the detailed description, it is divided into the three following catagories.

1. Receive E A A E and send O  
O D D p N and send O  
T 1 2 Q T
2. Receive E A A E and send O  
O D D p N and send O  
T 1 2 Q
- a. S E B  
T Text T C  
X X C
- or
- b. S A A S E B  
O D D T Text T C  
H 1 2 X X C
3. Receive A E  
C response, upcount cursor and send O  
K T
- N
- Receive A response and retransmit  
K

- Receive E A A E S E B  
O D D p N and send T Text T C  
T 1 2 Q X X C
- or
- Receive E A A E S A A S E B  
O D D p N and send O D D T Text T C  
T 1 2 Q H 1 2 X X C

Fig. II-72 and related prints.

DETAILED DESCRIPTION

- Receive E A A E  
O D D p N and respond with O  
T 1 2 Q T

Functional Detail

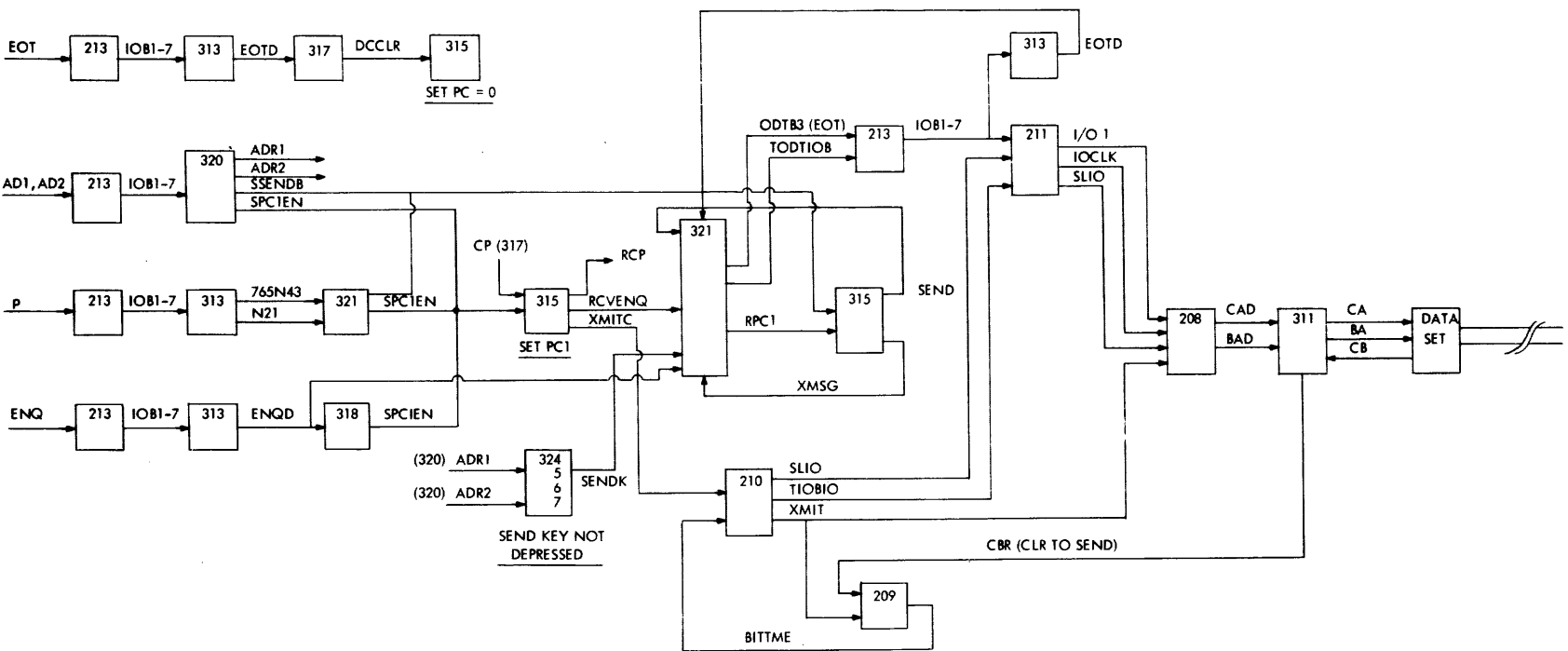


Fig. II-69 SELECTIVE/FRAME SEND - RECEIVE EOT, AD1, AD2, q, ENQ, - SEND EOT.

Functional Detail

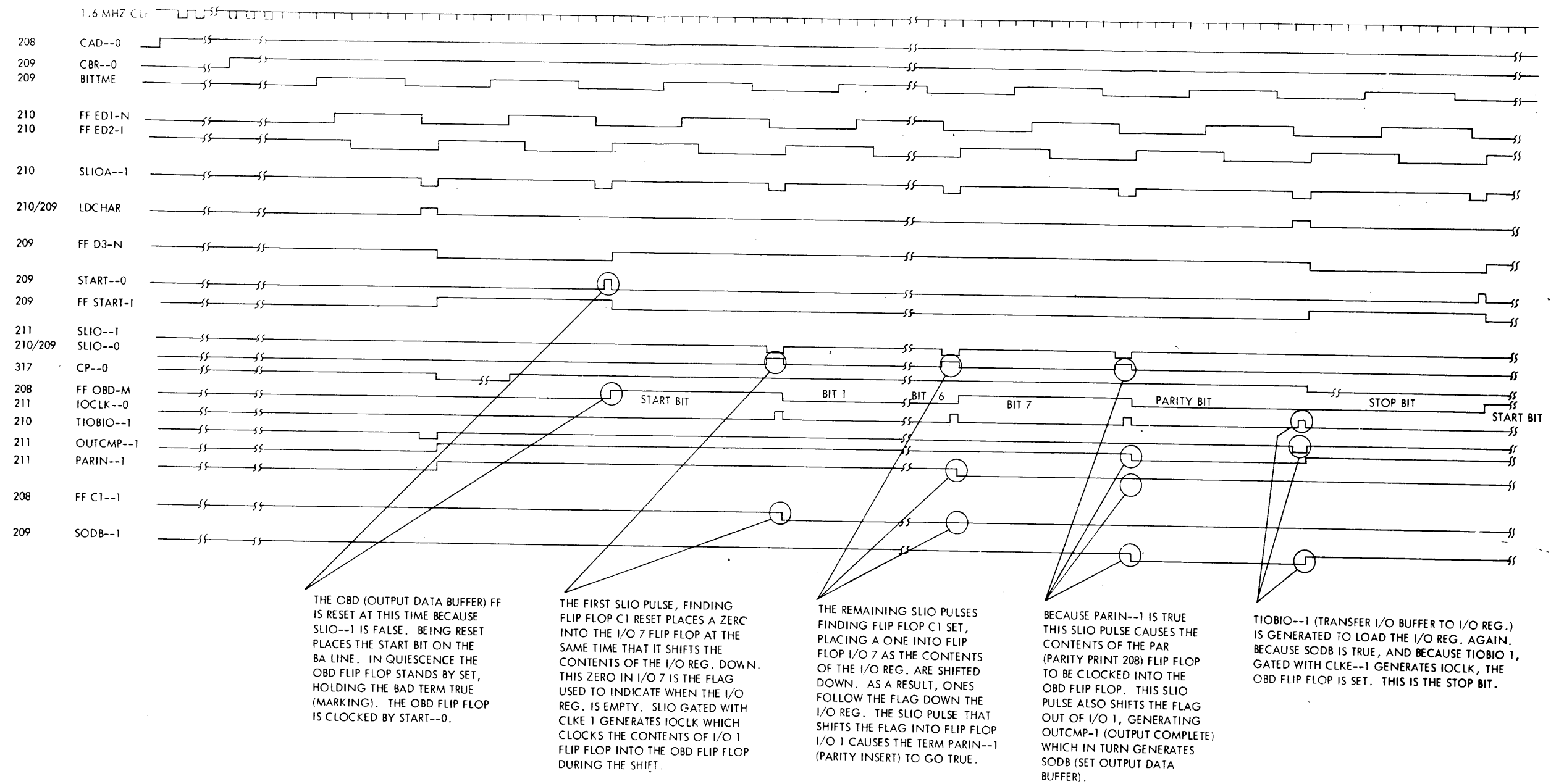
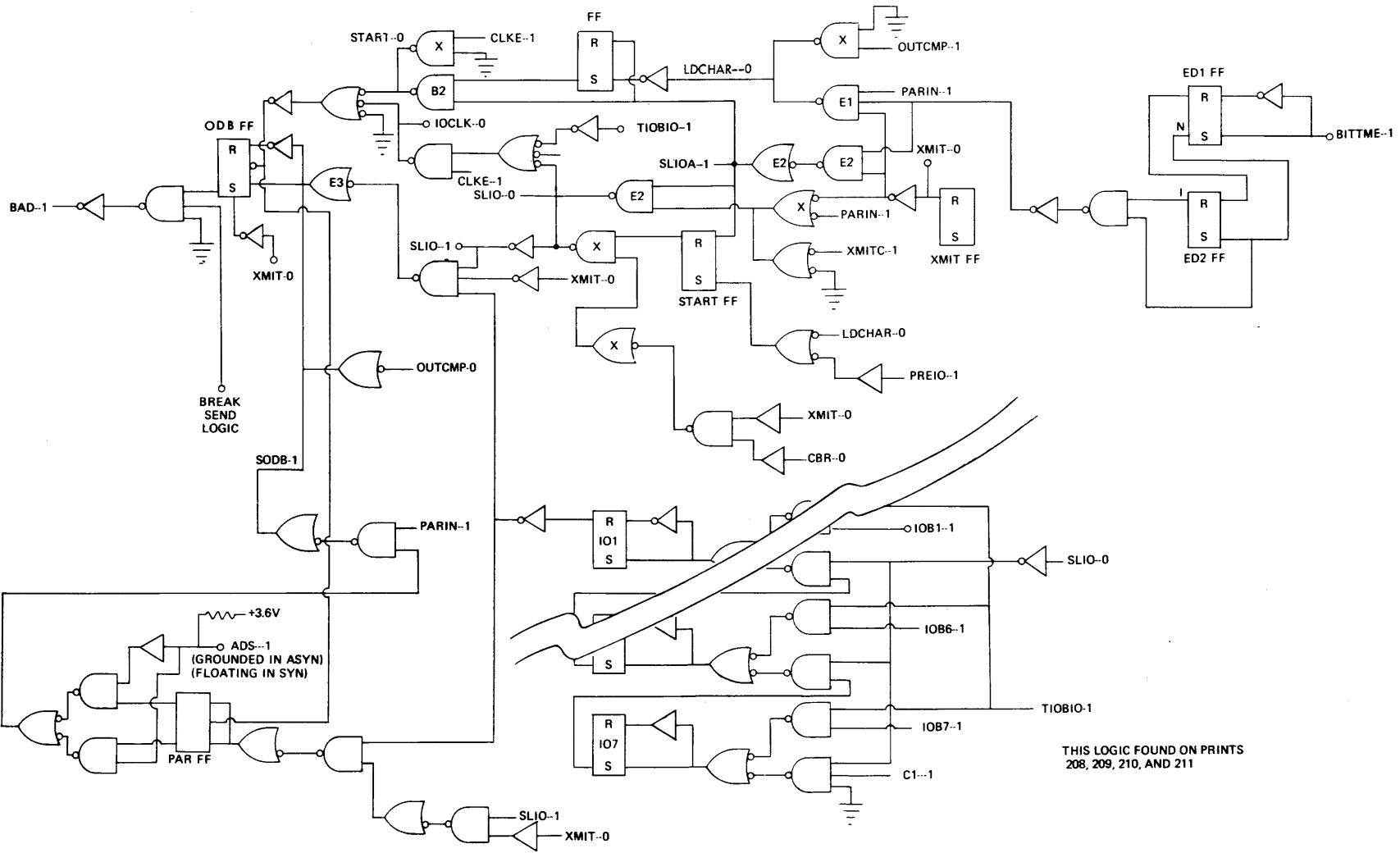


Fig. II-70 I/O REG SHIFT TIMING-TRANSMIT

Functional Detail



THIS LOGIC FOUND ON PRINTS  
208, 209, 210, AND 211

Fig. II-71



### Functional Detail

#### FRAME SEND

When the five character Polling sequence is received, the detection of the p (765N43, and N21 – Print 313) causes SSENDB to be developed, in turn setting the Send Flip-flop, generating SEND. (Print 315). The ENQ character, when detected develops SPC1EN, setting the PC1 Flip-flop (Print 315) generating XMITC. XMITC generates XMIT (Print 210) which conditions the bit timer (Print 209) and causes CAD (Request to Send) to be developed on Print 208. When CBR (Clear to Send) is detected, the bit timer is started, producing BITTME pulses. These pulses are used on 210 to develop SLIO (Shift Left I/O register) pulses, to shift any character that is in the I/O register (Print 211) onto the line. (The first BITTME pulse after CBR also generates TIOBIO, transferring any character in the I/O buffer into the I/O register. TIOBIO is generated in all other instances by CP (Character Present) indicating that the I/O buffer is empty.)

The level SEND, gated with SPC1EN and SENDK from the polled keyboard/monitor, causes SLA (Set Load Allow – Print 321) to be generated, in turn setting the LA Flip-flop on Print 316. This Flip-flop being set, allows the level STXE to be generated on Print 314 to take data from memory to be transmitted.

The setting of the SEND Flip-flop also generates XMSG (Transmit Message – Print 315). If the control unit has the Identification Option (Print/Card 319), XMITC releases the unconditional reset of the Character Counter, and XMSG, SEND, SENDK, and the fact that the counter is at a count of zero (CC0), causes SOH to be generated and placed on the ODTB1-7 lines. XMSG and CP also generated TODTIOB (Print 316) to transfer the contents of the ODTB1-7 lines to the I/O buffer (Print 213). Each time TODTIOB is generated, indicating the character on the ODTB1-7 lines has been transferred to the I/O buffer, the Character Counter is upcounted. At counts of CC1 and CC2, the wired in AD1 and AD2 characters on Print 319 are placed on the ODTB1-7 lines. At a count of CC3, the level STXEA (STXE Encode Allow) is generated. STXEA is used on print 314 to place the STX configuration onto the ODTB1-7 lines, and generate the term STXE (STX Enable). STXE generates RPMA (Reset Processor Memory Address) and LMIR (Load MIR) on Print 111.

RPMA clears the PMA Register (Print 219) and LMIR allows the data that is read out of memory to be recirculated back into memory. STXE, gated with ADR1 and ADR2 also causes the corresponding SPMA9 and SPMA10 levels to be generated. This sets the PMA9, PMA10 Flip-flops to the state necessary to address the section of memory determined by ADR1 and ADR2. (If the control unit did not have the Identification Option, STXE would be developed when LA is developed, resulting in no SOH, AD1, or AD2 characters being developed.) At a

character count of CC4, CC4 gated with XMSG causes the level ORQ (Output Request – Print 316) to be developed. (Without the Identification Option, CC4 is floating, and because of the resistor tie to ground on print 314, always appears true.)

QRQ causes SPMCY (Set Processor Memory Cycle – Print 111) to be generated, setting the PMCY Flip-flop on print 116. PMCY being set, generates APMA (Allow Processor Memory Addressing), causing addressing to be determined by the PMA Register (Print 219). PMCY also activates memory timing (Print 122). As data is read from memory, it is strobed into the MIR Register (Print 118). The output of the MIR Register is in turn placed onto the ODTB1-7 lines (Print 314) to be transferred to the I/O buffer. It is on this print that the 6 bit internal code is converted to the 7 bit ASCII code that is transmitted. The output of the MIR Flip-flops are also sampled on print 216 for specific characters.

The output of the I/O buffer, in addition to going to the I/O Register, is also sampled by the I/O Decoder (Print 313) and the Block Check Register (Print 212). When either the SOH, or the STX character is detected, the levels SOHD or STXD causes the BCCE (Block Check Character Enable) Flip-flop to be set on print 212. Being set, this allows block check to be developed on all characters following SOH, or STX, whichever is detected first.

When the ES character is detected in the MIR Register, the level ES is generated (Print 216) and causes the ETX configuration to be placed on the ODTB1-7 lines (Print 314). With the ETX character in the I/O Buffer (Print 213), the level ETXD from the I/O Decoder (Print 313) conditions the BCC Register (Print 212) so that when the ETX is transferred to the I/O Register (Print 211) the BCC character is transferred from the BCC Register to the I/O Buffer.

At this time ENDBCC is generated, developing SRPC (Shift Right PC) to cause PC2 to be set. With PC2 set, an ACK or NAK response is looked for after the BCC character has been transmitted. Also, with PC2 set, XMSG and XMITC are dropped, resulting in no additional memory cycles.

Should the ESC portion of a two memory position character be detected in MIR by the logic on print 216, the level ESC causes the SCHAR (Special Character – Print 316) Flip-flop to be set when RQCMP indicates the end of that memory cycle (MT4)time. ESC also prevents the setting of the CP Flip-flop (Print 317). Because CP is not set, TIOBIO is not generated (210) meaning the ESC character does not get placed in the I/O Register and is not transmitted. Disabling CP also prevents it from being felt at the input of the BCC register, consequently there is no BCC character developed on the ESC. Because CP indicates there is no character in the I/O Buffer (regardless of the fact the ESC character is in the I/O Buffer), ORQ is held active, resulting in a second memory cycle being initiated as soon

Functional Detail

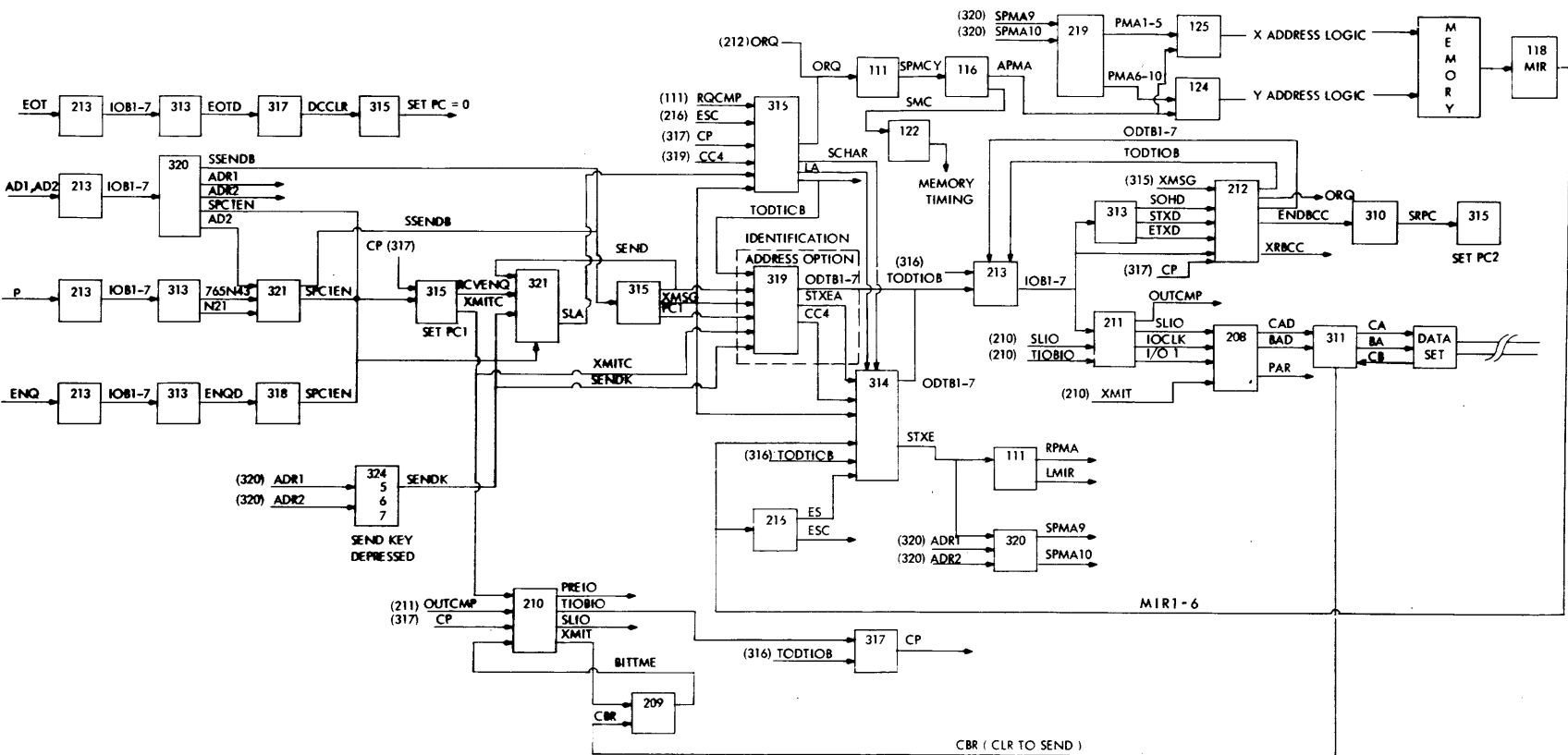


Fig. II-72 FRAME SEND-RECEIVE EOT, AD1, AD2, p, ENQ — SEND — SOH, AD1, AD2, STX, TEXT, ETX, BCC, OR STX, TEXT, ETX, BCC.

Functional Detail

as the one for the ESC is complete. This memory cycle reads out the character associated with the ESC character, and it is then placed on the ODTB1-7 lines. However, because the SCHAR Flip-flop is still set when TODTIOB occurs, ODTB3 is forced true (Print 314), causing that character placed in the I/O Buffer to have its 3 bit turned on.

**SELECTIVE SEND: FIGURE II-73 AND RELATED PRINTS**

If, when the SEND key is depressed the shift key is also depressed, a slightly different action takes place than described earlier. In Selective Send, the transmission takes place from the cursor location, meaning the cursor location must first be placed in the PMA register. When the STX character is generated, the level STXD, gated with SELS (Selective Send) on print 111 generates SPMCY, initiating the first of two memory cycles. These memory cycles read out the reserved core locations containing the cursor locations for the particular monitor being polled. (See Select for a detailed explanation of this type of operation.) This data, as it is read out is placed in the PMA register (Print 219) and when the information in memory is transmitted, it is read from memory from the starting location determined by the PMA register.

If, during a Selective Send operation, the RS symbol of a form is detected, the RS symbol, the US symbol and the data between the symbols is prevented from being transmitted. (In a Frame Send operation, the entire form, including the RS and US symbols is transmitted.) The data is read from memory and placed in the I/O Buffer (Print 213) but is not allowed to get to the I/O Register, and as a result, cannot be transmitted.

Detecting the RS symbol causes the FSMI Flip-flop

(Print 107) to be set. FSMI is used on print 317 to disable the setting of the CP Flip-flop. Because the CP Flip-flop cannot be set, XMSG (Print 315) remains true, and as a result, ORQ (Print 316) remains true, and memory cycles continue, with the data being placed in the I/O Buffer. Also, because the CP Flip-flop can not be set, the level TIOBIO (Print 210) is not generated and the data in the I/O Buffer is not transferred to the I/O Register.

Detecting the US symbol resets the FSMI Flip-flop and the Selective Send operation continues in the normal manner.

|   |    |
|---|----|
| A   | E  |
| Receive C response, upcount cursor and send O |    |
| K   | T  |
|   | or |
| N   |    |
| Receive A response and retransmit             |    |
| K   |    |

Refer to Figure II-74 and related prints.

Decoding of ACK in the I/O decoder (313) and RCVRSP (Receive Response – Print 313) develops ENDBLK, setting the ENDS Flip-flop (117). At the next FRMO time the XMT Flip-flop is set and at the second FRMO time, the level XMTA is generated. XMTA, ACURL (Allow Cursor Level – Print 203) XRBCC and NORCV generate SRPC setting PC3 for the EOT response. ACURL occurs at the end of the Put sequence. There is a period of approximately 13 microseconds between the end of the Put sequence and the beginning of the Get sequence and it is during this time that the cursor is updated and the EOT generated. From the time the ACK is detected, till the second FRMO pulse occurs to generate XMTA and in turn cause the setting of PC3 to generate, EOT can be a minimum of 16 milliseconds to a maximum of 32 milliseconds, depending on which screen is being used.

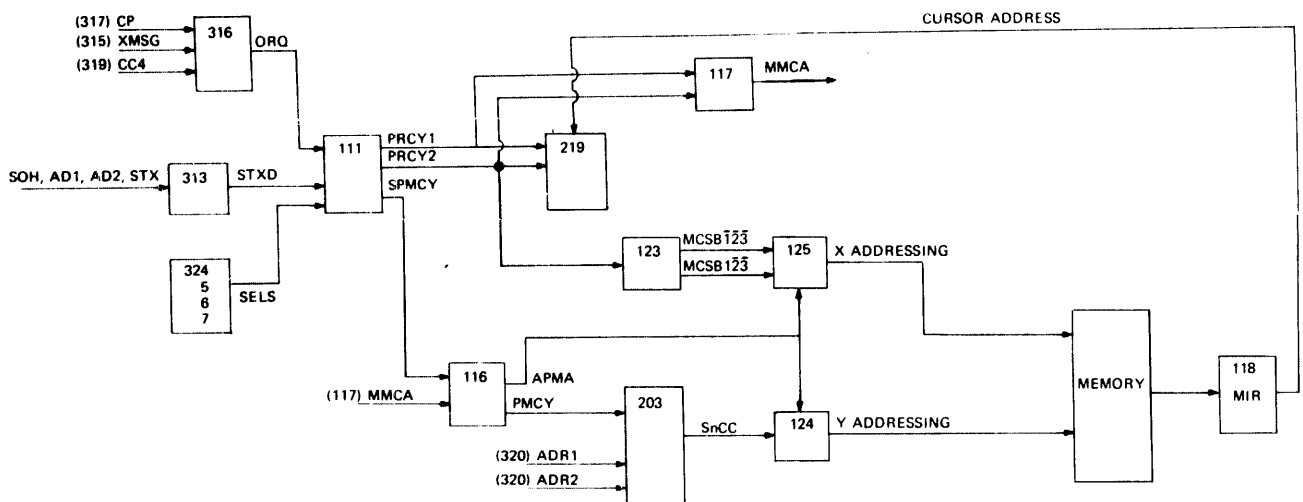
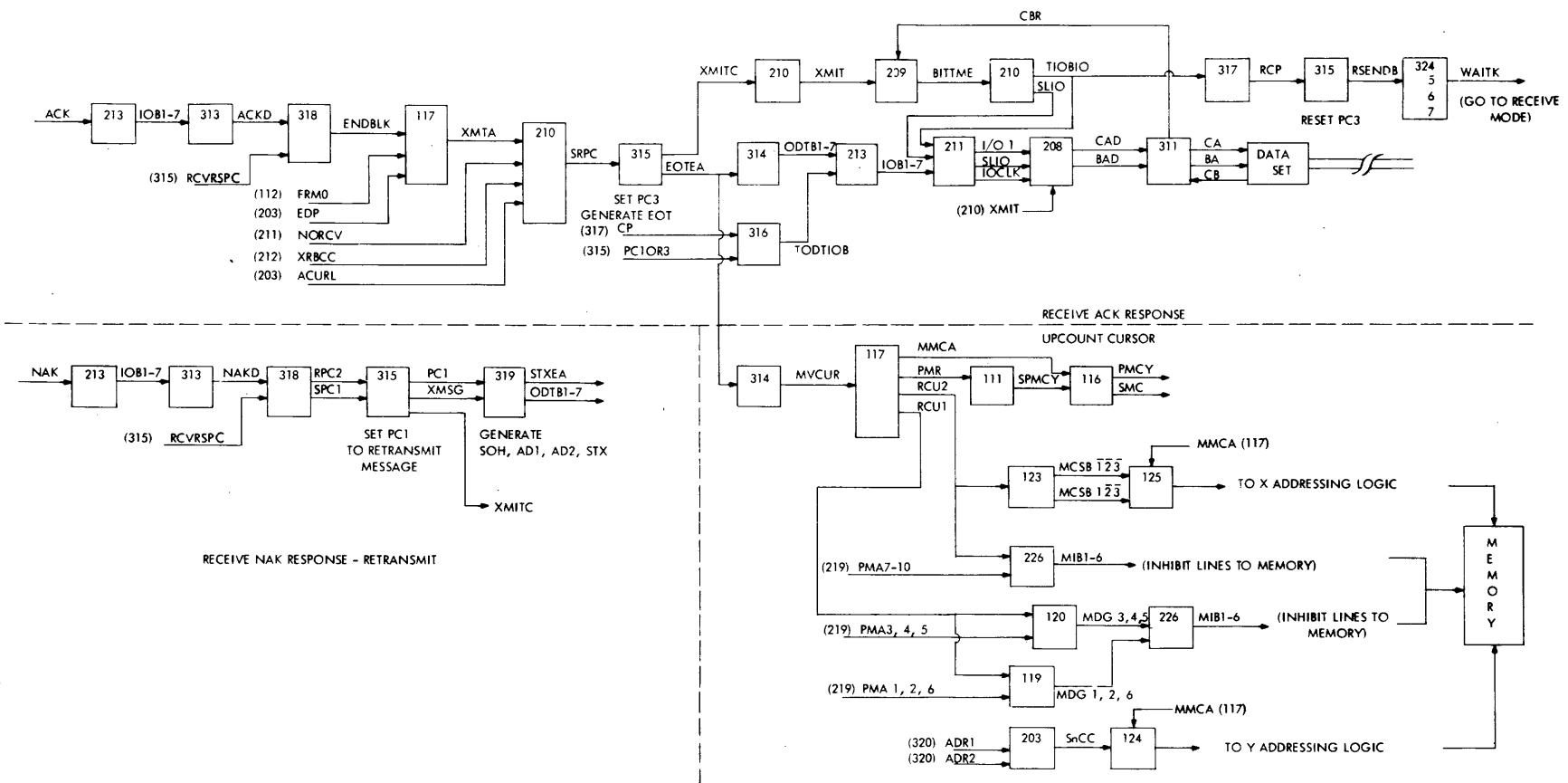


Fig. II-73 SELECTIVE SEND – POLL RECEIVED – SEND SOH, AD1, AD2, STX – LOCATE CURSOR

Functional Detail



SELECTIVE/FRAME SEND - RECEIVE ACK RESPONSE - SEND EOT - UPCOUNT CURSOR  
 SELECTIVE/FRAME SEND - RECEIVE NAK RESPONSE - RETRANSMIT MESSAGE

Fig. II-74

Functional Detail

The setting of PC3 causes EOTEA to be developed to place an EOT character in the I/O Buffer. PC3 also develops XMITC to condition the Bit Timer (209) for transmitting the EOT. When TIOBIO is developed (Print 210) to transfer the EOT to the I/O register, it also generates RCP (Print 317) which in turn generates RSEMDB (Print 315). RSEMDB forces the unit into the receive mode. EOTEA causes the cursor to be positioned to the end of the message on the screen by developing MVCUR (Move Cursor – Print 314) which initiates 2 additional memory cycles. During these memory cycles, the contents of the PMA register (219) are written into the two reserved cursor memory locations for this particular screen. (See Select for a detailed explanation of the MVCUR operation.)

Decoding of the NAK character on 313 causes RPC2 and SPC1 to be developed (318), in turn, resetting PC2 and setting PC1. PC1 set generates XMITC and XMSG triggering the character counter (319) and the operation repeats with the entire message being retransmitted.

FAST SELECT

Fast Select differs from the normal Select in that an ACK response is not required from the master station to send the message. The Fast Select message sequence is:

```
E A A F S      E B
O D D S T Text T C
T 1 2 L X      X C
```

└─ Lower case s

The master station only looks for an ACK, NAK, or no response at the end of the entire message. Refer to Prints 321, 317, 315, 313.

Fast Select is identical to normal Select in most respects, so only those differences are covered. Upon receiving the EOT character, the term DCCLR (317) is developed, clearing the PC counter (315). Having received the AD1 and AD2 characters, the TWO Flip-flop (321) is set, indicating that two characters have arrived following the EOT character and the third character should be looked at.

The third character, being the FSL (s) character, causes the terms 765N43 and 21 to be generated (313). These terms are used on 321 to generate either RCVAlM (Receive Alarm) if the monitor being selected is not in receive mode (WAITK is false), or SPC2 if the monitor selected is in receive mode (WAITK is true). SPC2 causes the PC2 Flip-flop on 315 to be set placing the unit in the state necessary to receive STX, TEXT, ETX and BCC. The operation from this point on is the same as a normal Select sequence.

BROADCAST SELECT

The Broadcast Select message is used to broadcast data to all terminals on a line. The terminals, upon determining that a Broadcast message is being transmitted, go into the receive mode automatically. One terminal is designated to respond for all other terminals. The Broadcast Select message is as follows:

```
E A A B S      E B
O D D S T Text T C
T 1 2 L X      X C
```

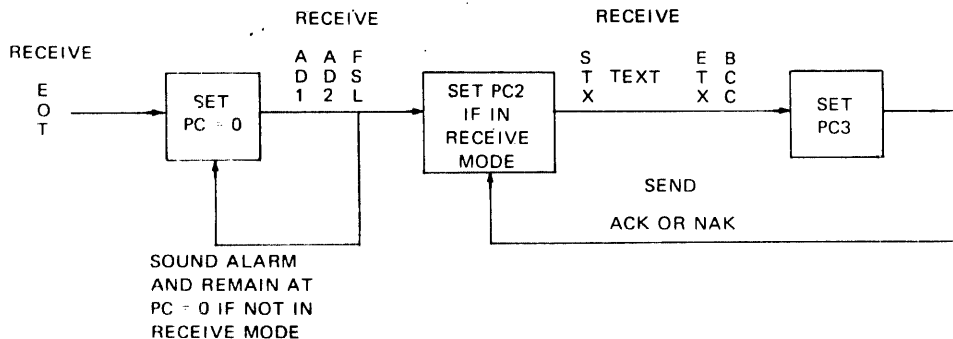
└─ Lower case t  
└─ Terminal that is designated to respond

Assuming the B9353 is a 4 monitor system, as each data character is received, eight memory cycles are performed. Four memory cycles are to write the data character into the four different quarters of memory and four memory cycles are to write the ES character into the location following the data character in the four different quarters of memory. The starting address for writing data into memory is derived from the cursor location of monitor zero.

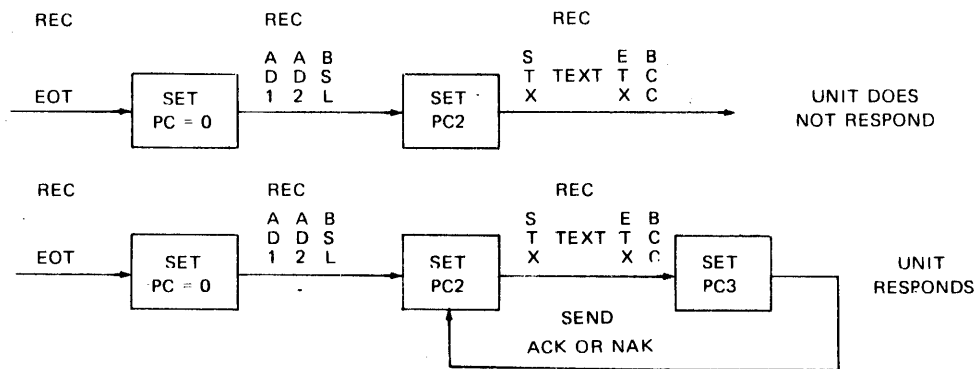
Those monitors that have an ES character before the cursor location of monitor zero will not display anything since the Display sequence will detect the ES before the data that is written into memory.

Refer to Fig. II-75, 76 and related prints.

Receiving EOT generates DCCLR (Data Comm Clear – 317) clearing the PC counter (315) and the following



## Functional Detail



Flip-flops on 322, ENDSS (End Special Select), SSLF (Special Select), BSL (Broadcast Select) and by way of the SSLF Flip-flop, the SC1 and SC2 (Screen Count) Flip-flops. With this counter cleared, the screen address terms ADR1 and ADR2 are both false, effectively causing screen zero to be addressed.

The AD1 and AD2 characters are compared with the wired in address (320) and if comparison exists, the term AD2 (320) causes SLCT to be generated (321). SLCT, if true, indicates that this unit is to respond with ACK or NAK at the end of the message. The BSL character causes the following to happen:

1. SPC2 (322) generated, setting PC2 (315) which places the unit in the condition where STX, TEXT, ETX, & BCC are expected.
2. The BSL Flip-flop (322) is set, developing the term BSL.
3. The term BSLD (BSL Detected) is generated, and assuming the Control Unit is wired for four monitors (ALS1, ALS2, ALS3), causes the Special Select Flip-flops to be set (SS0, SS1, SS2 and SS3) on print 323. These Flip-flops in turn generate ADRDOE, ADRD1E, ADRD2E and ADRD3E to be used on print 322. BSLD also generates RSENB which is gated with ADRDOE on 327, ADRD1E on 326, ADRD2E on 325 and ADRD3E on 324 to set the related WAIT Flip-flops. This action forces all monitors into Receive mode.
4. The SSLF (Special Select Flip-flop – 322) is set, developing SSLF which is used on print 320 to disable the ADR1 and ADR2 terms. This means that only the ADR1 and ADR2 on 322 are used. SSLF also disables CPMA (Count Processor Memory Address – 314) so that the PMA register (219) cannot be counted in the regular manner.
5. Develops SRM (Start Receive Mode) setting FLip-flops PRCY1 and RCV on print 111. When the STX character is received, the LA Flip-flop is set (Print 316), generating IRQ (Print 318). IRQ then initiates the first of the two memory cycles to read out the reserved memory locations that contain the cursor

address for screen zero. The ADR1 and ADR2 terms being false, develop the Y addressing for screen zero and PRCY1 and PRCY2 develop the X addressing. This information when read out is placed into the PMA register for later use.

## REFER TO 322

Because Flip-flops SS0, SS1, SS2 and SS3 were set, ADRDOE, 1E, 2E and 3E disable gates D2-I, D2-C, D2-E and E1-C (322). Because all of the gates are disabled, IRQ is allowed to go false (though IRQ is initiated on 322 it cannot go false until the logic on 318 allows it), initiating the memory cycle and writing the Data character into the section of memory for screen zero. At MT4 of the first memory cycle, RQCMP (111) enables gate EO-E counting the SC counter to 1. This same pulse also generates SPMA9 and CSCLK to set the PMA9 Flip-flop. CSCLK only clocks PMA9 and PMA10. PMA9 and PMA10 address the different quarters of memory, consequently, by setting PMA9 and leaving PMA1–PMA8 unchanged, the PMA register is now addressing the same relative address, only now in the second quarter of memory. Because CP is still present, and gates D2-I, D2-C, D2-E and E1-C are still disabled, IRQ is held, and a second cycle is initiated to write the Data character into the second quarter of memory. At MT4 of the second cycle, the SC counter is counted to 2, and simultaneously SPMA9 and CSCLK are again generated. Setting PMA10 (the second SPMA9 resets PMA9 & sets PMA10) and causing the same relative address of the third quarter of memory to be addressed. IRQ still being present, initiates a third memory cycle to write the Data character into the third quarter of memory. At MT4 of the third cycle, the SC counter is counted to 3 and SPMA9 and CSCLK causes PMA9 and PMA10 to be set, addressing the fourth quarter of memory. Again IRQ initiates the memory cycle and the Data character is written into the fourth quarter of memory. At MT4 of the fourth memory cycle, SPMA9 and CSCLK causes PMA9 and PMA10 to be reset, thereby addressing the first quarter of memory again. Because the SC counter is at a count of 3, RCP (Reset CP) is generated to reset the CP Flip-flop, causing the IRQ to

Functional Detail

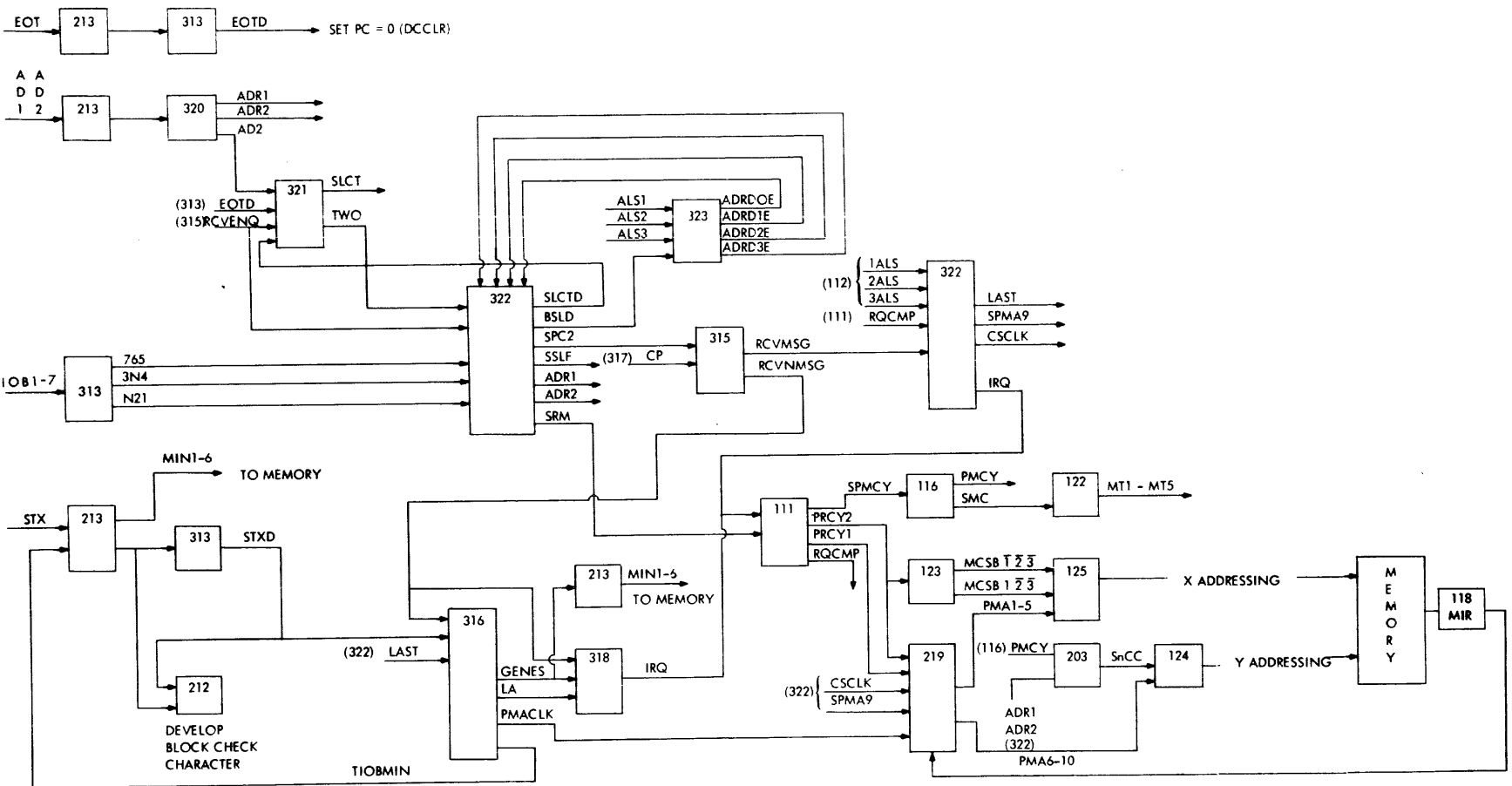


Fig. II-75 BROADCAST SELECT-RECEIVE

E A A B S  
O D D S T  
7 1 2 L X  
TEXT

Functional Detail

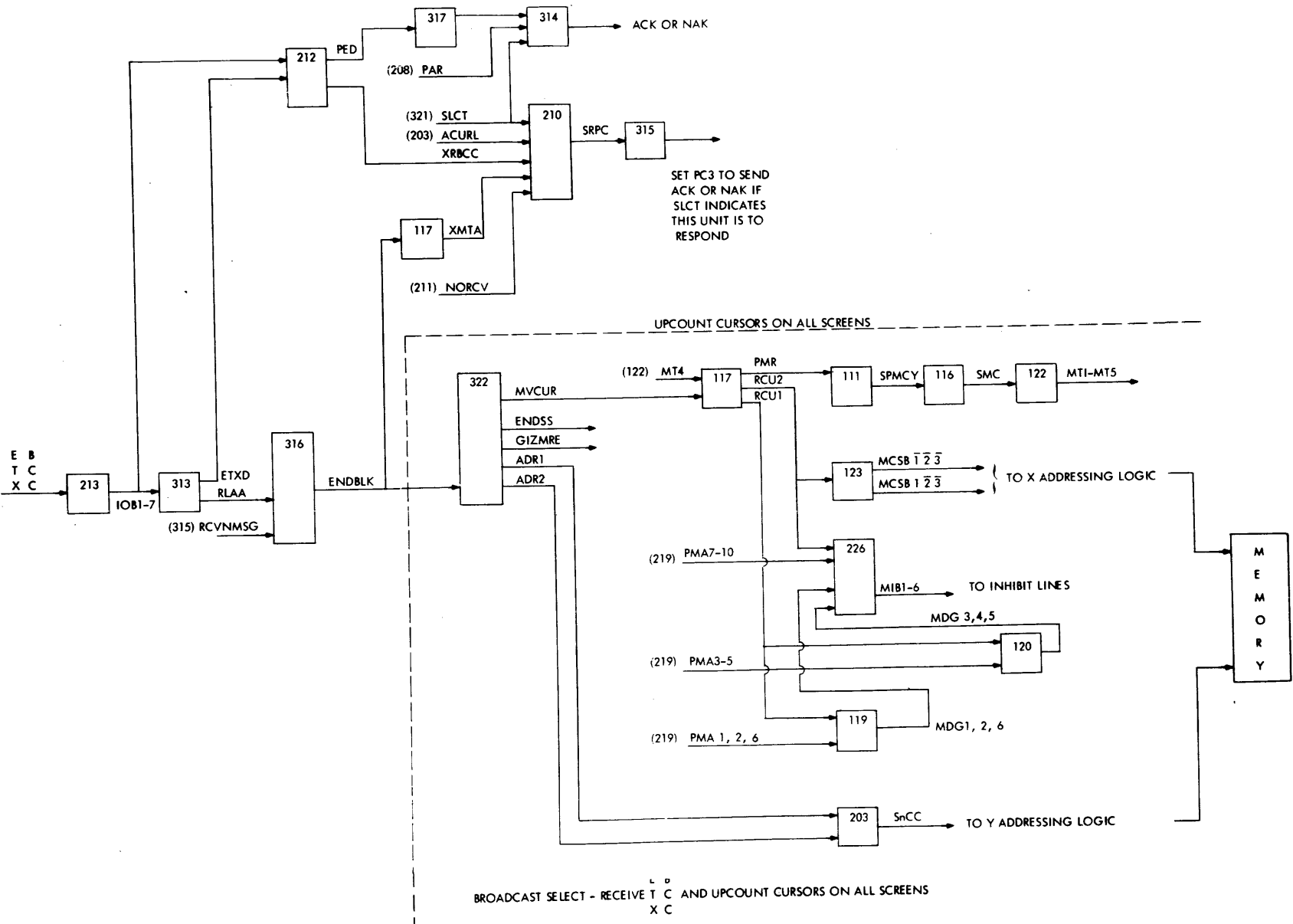


Fig. II-76 BROADCAST SELECT-RECEIVE EB AND UPCOUNT CURSORS ON ALL SCREENS

TC XC



Functional Detail

drop (318). At this time, the term LAST is developed, setting the LAST+1 Flip-flop (316) and generating PMACLK. PMACLK clocks the PMA1 – PMA8 Flip-flops, causing them to be upcounted by 1 to address the next address in memory. Due to the fact the SC counter is at a count of 3, the pulse that occurs at MT4, also counts the SC counter back to zero.

The term LAST also sets the GENES (Generate End Screen Flip-flop – 316). GENES in turn generates IRQ again to initiate another sequence of memory cycles. These memory cycles are to write the ES character into the memory location following the last Data character in the different sections of memory (ES placed on the MINI-6 lines by GENES – 213.) At the end of the last of these memory cycles, the term LAST is again generated, this time, resetting the GENES FF, preventing additional memory cycles from occurring.

Because the GENES Flip-flop is set during this second series of memory cycles, the term PMACLK is disabled, preventing PMA1-8 from being upcounted. As a result, when the next Data character comes in from the master station, it is written over the ES characters.

The other portions of the operation, such as developing BCC, parity checking, ESC characters, etc. is the same as a normal select (see Fig. II-64). Upon detecting the ETX character (see Fig. II-71), RLAA (Reset Load Allow – 313) is developed, resetting the LA Flip-flop (316) and developing ENDBLK (End Block). ENDBLK sets the ENDSS (End Special Select – 322) Flip-flop, setting up the conditions for updating the cursors on all screens. ENDSS causes the GIZMRE (Generate Initialize Memory Request – 322) Flip-flop to be set at the next ACURL pulse. With GIZMRE set, Put Get sequences cannot be initiated until all of the cursors have been updated. The setting of GIZMRE generates MVCUR (move cursor), in turn developing SPMCY (117) and causing 2 memory cycles for each monitor to be performed. X addressing is a function of the state of RCU2, one of a two Flip-flop counter. (RCU2 is reset for the 1st cycle and set for the 2nd.) Y addressing is a

function of the output of the SC counter (322), which is in a cleared state at the beginning of cursor update, and is upcounted at the end of each 2nd cycle. The SC counter as it is upcounted every 2nd cycle, effectively changes the Y lines to address a different quarter of memory. The first of each of the 2 memory cycles transfers PMA1–6 to memory and the 2nd of each two cycles transfers PMA7–10 to memory. At MT4, with the SC at a count of 3, ENDSS (322) is reset, resetting GIZMRE ending the operation.

If the SLCT Flip-flop (321) is set, in addition to performing everything previously mentioned, an ACK or NAK response would be generated. This is done in the same manner as a normal select, so it is not covered here.

GROUP SELECT

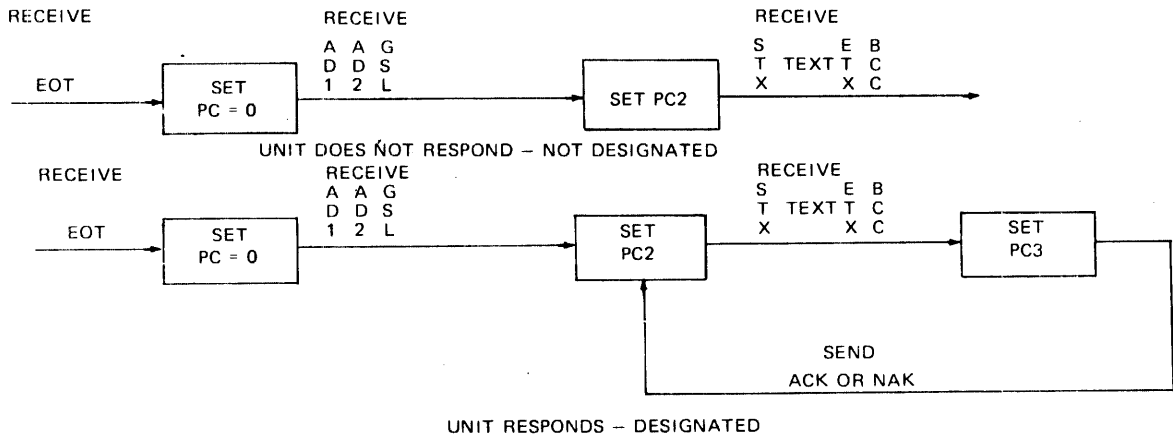
The Group Select message is used to send data to specified groups of terminals on a line, with a specific terminal within that group designated to respond for the entire group. The Group Select message is as follows:

```

E A A G S           E B
O D D S T TEXT T C
T 1 2 L X           X C
    
```

Any character in columns 2 thru 6 of the ASCII 1967 code card  
 Address of the terminal that is to respond for the group

The Group Select character (GSL) is wired in on print 323 and enables one or more monitors to be a part of the group, regardless of the Control Units wired in address. The starting address for writing data into memory is derived from the cursor location of the lowest number monitor in the group. Those monitors within the group that have an ES character before the cursor location of the lowest number monitor will not display anything, since the Display Sequence for those monitors will detect the ES



UNIT RESPONDS – DESIGNATED

Functional Detail

character before the data that is written into memory. Refer to Fig. II-77 and related prints.

Group Select is identical to Broadcast in most respects. Receiving EOT generates DCCLR (Data Com Clear – 317) clearing the PC counter (315) and the following Flip-flops on 322. ENDSS (End Special Select) Flip-flop SSLF (Special Select) Flip-flop and, as a result of clearing SSLF, SC1 and SC2 (Screen Count Flip-flop are cleared.

The AD1 and AD2 characters are received and compared with the wired in address (320). If comparison exists, AD2 causes SLCT to be generated (321) indicating that this terminal is the terminal designated to respond with ACK or NAK. The GSL character causes the following to happen:

1. Assuming a four monitor system, and screen 2 and 3 wired for the GSL character, the SS2 and SS3 Flip-flops are set, generating ADR2, and ADRD2E and ADRD3E.
2. SPC2A (323) generated, setting PC2 (315) placing the control unit in the receive mode to receive STX, TEXT, ETX and BCC.
3. SSL is generated, setting the SSLF (Special Select Flip-flop –322) whose output is used on 320 to disable ADR1 and ADR2. This means that only the ADR1 and ADR2 terms on 323 are used. SSLF also disables CPMA (Count Processor Memory Address – 314), preventing the PMA (219) register from being counted in the normal manner.
4. Develops SRM (Start Receive Mode) setting PRCY1 and initiating the first of two memory cycles to read out the reserved memory location containing the cursor address of screen two. ADR1 being false and ADR2 being true develops the X addressing logic for screen two. PRCY1 and PRCY2 develops the Y addressing logic. This information when read out of memory is placed into the PMA register for later use.

Upon receiving the STX character, the LA (Load Allow) Flip-flop is set (Print 316). When the next character is received and CP indicates it is in the I/O Buffer, IRQ is generated initiating the memory cycle to store the character at the address called out by the PMA register.

Refer to 322 and 323

Because Flip-flops SS2 and SS3 are set, ADRD2E and ADRD3E are true, disabling gates D2-C and D2-I on 322. Also, because ADRD0E is false, and the SC1 and SC2 Flip-flops are reset, gate E1-C is enabled, in turn enabling gate E0-H, preventing IRQ from being generated at this time.

Gate E0-H enables gate E0-C upcounting the SC1 and SC2 Flip-flops to one. This same pulse that set SC1 Flip-flop also generates SPMA9 and CSCLK setting the PMA9 Flip-flop on 219, addressing the second quarter of memory. Because SC1 Flip-flop is set and ADRD1E is false,

gate D2-E is enabled. This in turn enables gate E0-H, again preventing IRQ and again causing SPMA9 to be generated, resetting PMA9 and setting PMA10, causing the third quarter of memory to be addressed. The SC2 Flip-flop is also set at this time.

Because the SC2 Flip-flop is set, and ADRD2E is true, gate D2C is disabled, in turn disabling gate E0-H allowing IRQ to be generated writing the Data character into the quarter of memory for screen two. At MT4 of this memory cycle, RQCMP (111) is generated, enabling gate E0-E (322) generating SPMA9 again (causes both PMA9 and PMA10 to be set) to cause the fourth quarter of memory to be addressed. Also SC1 Flip-flop is set (SC1 and SC2 now set). Because SC1 and SC2 are set, and ADRD3E is true, IRQ is allowed to go false, initiating another memory cycle to write the same Data character into the fourth quarter of memory (screen three).

At MT4 of this second cycle, the term LAST is developed, setting the LAST +1 Flip-flop (316), generating PMACLK and clocking the PMA1–8 Flip-flops. Also at this time the SC1 and SC2 Flip-flops are cleared. LAST also generates GENES (Generate End Screen) in turn generating IRQ (318) to initiate another sequence of memory cycles, to write the ES character into the location of memory following the last Data character in the quarter of memory allotted to monitors two and three.

At MT4 of the second memory cycle, having written the ES character, the GENES Flip-flop is reset. Because GENES was set, PMACLK was disabled, preventing PMA1–8 from being upcounted. The remainder of the operation is the same as a Broadcast Select sequence.

SEQUENTIAL SELECT

Sequential Select allows data to be transmitted to numerous terminals in just one transmission. Sequential differs from Broadcast or Group Select in that only specific terminals are selected to receive the message, rather than all terminals or groups of terminals. Sequential also differs from Broadcast and Group select in that an ACK response must be received before data can be transmitted. The message sequence is as follows:

```

E A A S A A S A A S A A S A A S E
O D E D D E D D E D D E D D E N
T 1 2 Q 1 2 Q 1 2 Q 1 2 L Q
    
```

Lower case q indicates that this monitor/terminal is to respond for all.

Lower case r

Address if specific terminal/monitor. If more than one monitor on a terminal is to be sequentially selected, the next AD1, AD2, SEQ string should address the next monitor.

Functional Detail

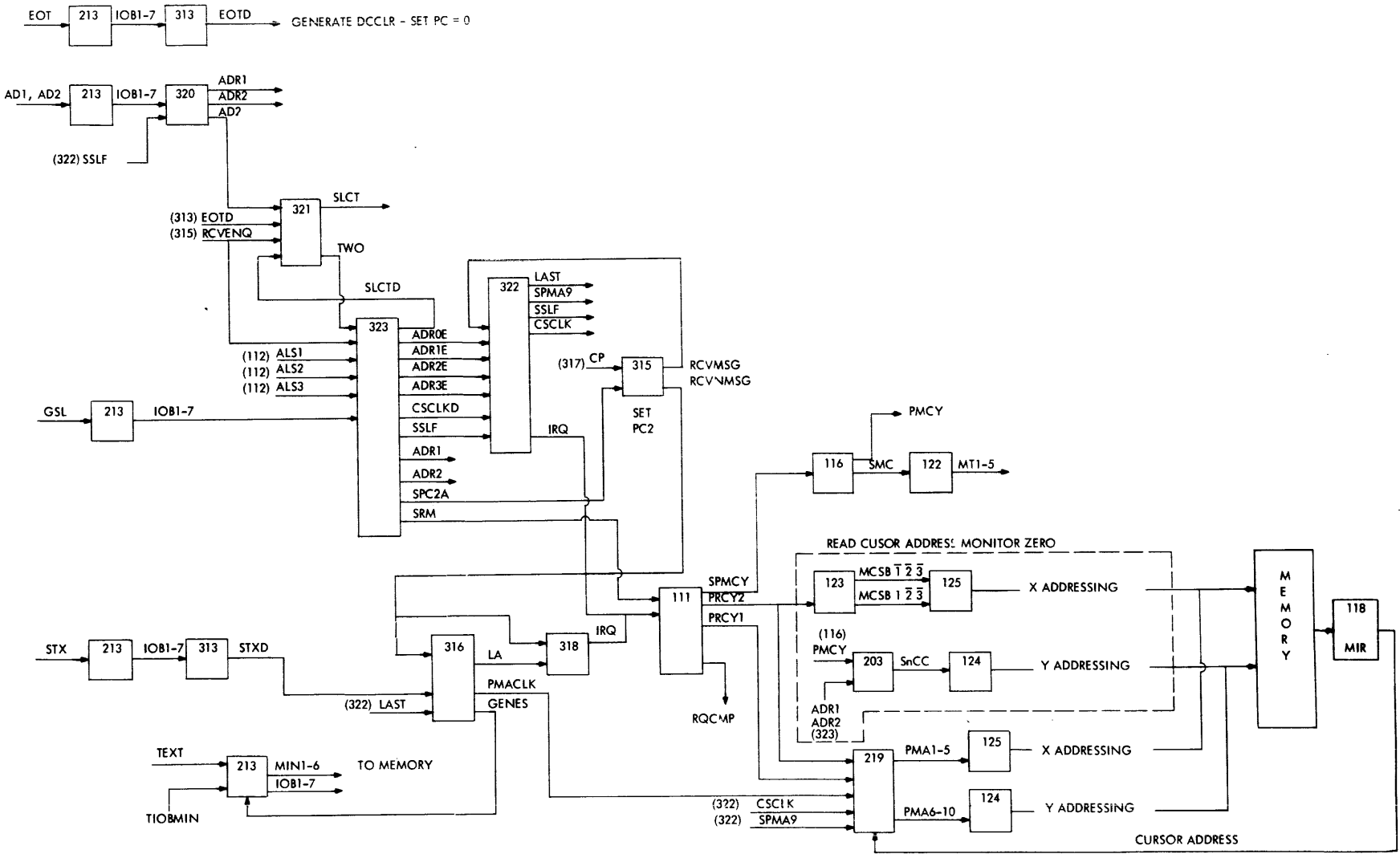
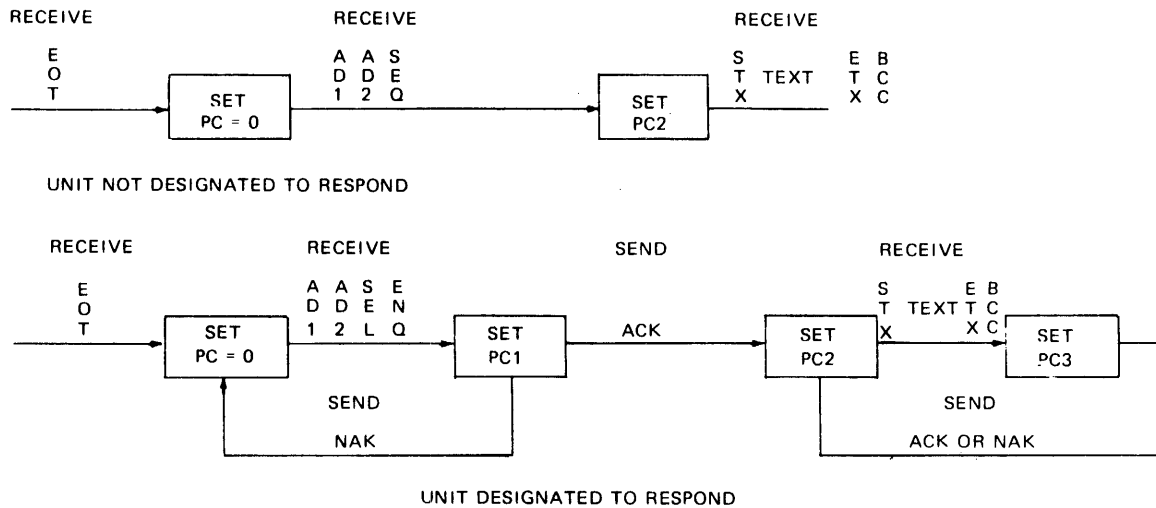


Fig. II-77 GROUP SELECT

Functional Detail



Refer to Prints 207, 323.

This operation varies little from Broadcast or Group select except in the way the selection of the monitors occur.

Assume the sequence EOT, AD1, AD2, SEQ is received and compares with the wired in terminal address on 320. With the AD2 Flip-flop (320) set, if the next character is the SEQ character, the terms 765N43 and 2N1 are generated, setting the SEQR Flip-flop (207). Because the AD2 Flip-flop is set, the state of the ADR1 and ADR2 Flip-flops (contain the screen address) causes the appropriate SSS0, SSS1, SSS2 and SSS3 terms to be generated, setting the related SSSn Flip-flop (323) and in turn generating SSSL (323). SSSL is used on 322 to set the SSLF (Special Select Flip-flop).

Because the character following the AD2 character was not the q or the ENQ, the AD2 Flip-flop (320) is reset, and if another screen on the same control unit is to be sequentially selected, the AD1, AD2, SEQ is repeated, only this time the AD2 bit configuration calls on a different screen (ADR1T, and ADR2T – 320) and sets a second SSSn Flip-flop.

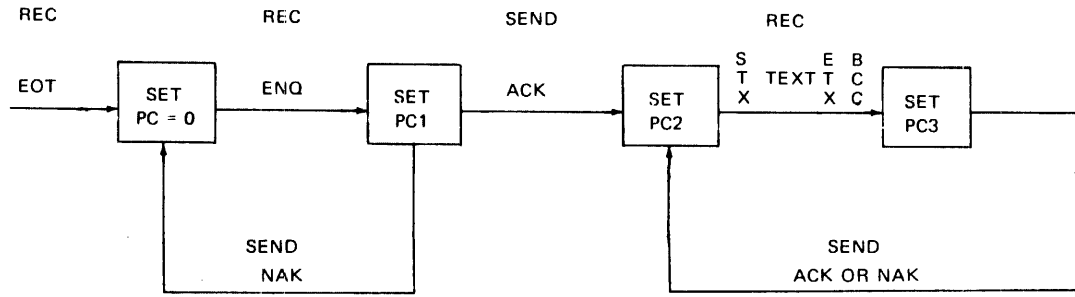
Because the message sequence is AD1, AD2, SEQ it indicates that this particular Control Unit is not to respond. As a result, the PC counter (315) is set directly to PC2 upon detecting the ENQ in the last Selection string (323).

ENQD, gated with SSLF – 0 being false, and SLCT – 1 being true (this Control Unit and monitor not selected to respond), causes SPC2 to be generated, setting the PC2 Flip-flop on 315. Also at this time SRM (Start Receive Mode – 323) is developed, initiating the two memory cycles necessary to locate the cursor. If more than one screen on a Control Unit is selected, the cursor address of the lowest number screen is used. The operation from this point on is identical to Broadcast or Group Select.

If the message sequence received is AD1, AD2, q, ENQ it indicates that this Control Unit, and its designated monitor is to respond for all.

The AD2 Flip-flop, when set, attempts to generate SPC1EN (SPC1EN cannot be generated until q and ENQ are received). Because the character after the AD2 character is the Select character (q), the term SLCTD is generated (321), satisfying another condition for SPC1EN and preventing the AD2 Flip-flop from being reset. Upon receiving the ENQ character, ENQD is developed (313), satisfying the final condition necessary for SPC1EN (318). ENQD also prevents AD2 Flip-flop from being reset at this time. Because SPC1EN is generated, the PC1 Flip-flop is set (315), providing the condition necessary for sending ACK or NAK. (See Select Sequence.) Having sent ACK, the PC2 Flip-flop is set and the operation continues as in Broadcast or Group select.

Functional Detail



POINT TO POINT RECEIVE

Refer to Fig. II-78 and related prints

Point to point differs from Multipoint in that only one control unit with only one monitor is used. This eliminates the need for any addressing.

Upon receiving an ENQ from the master station, PC1 is set for one character ACK or NAK response (315). If the

unit is in receive mode, an ACK is generated (314) and transmitted and the control goes to PC2 to wait for the message. The remainder of the operation is identical to multipoint select (Fig. II-64).

If the unit is not in receive mode, WAITK is false (324), an NAK is generated (314) and transmitted, the control unit returns to PC=0, and the alarm indicator is lit.

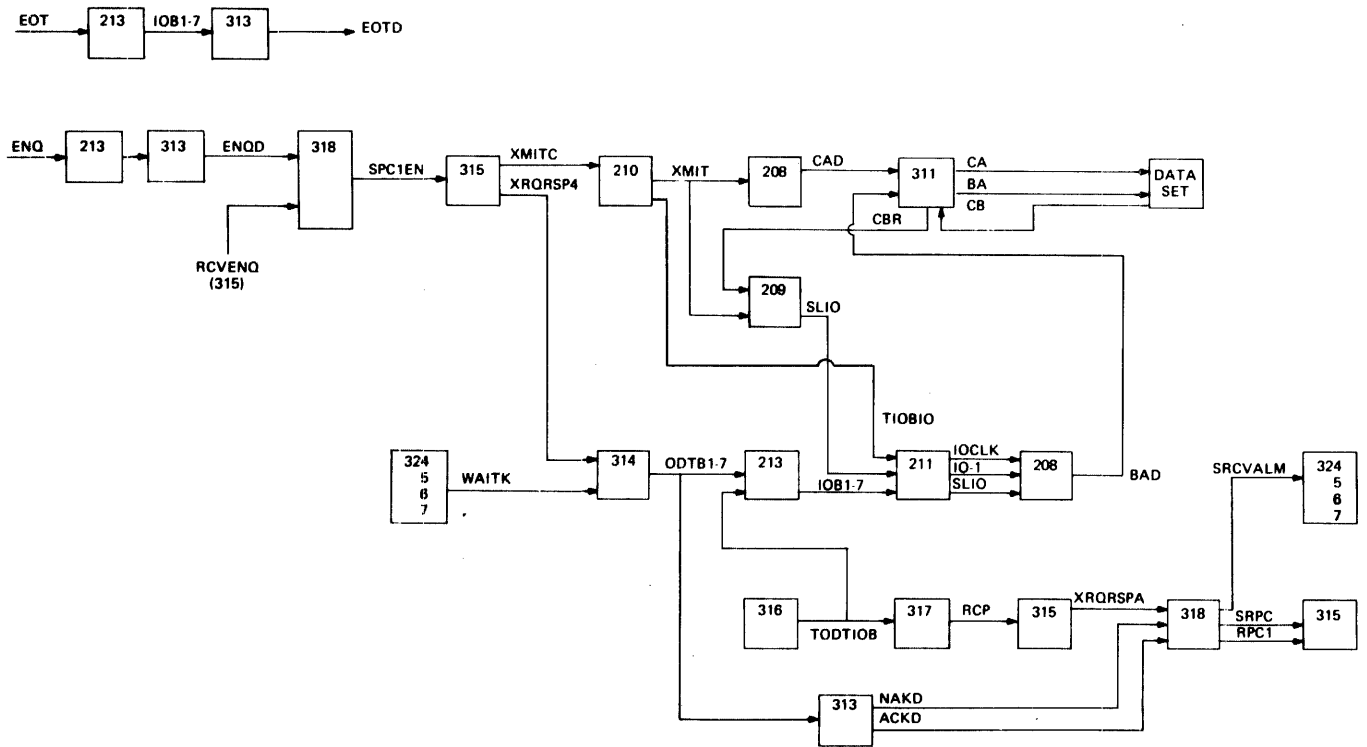
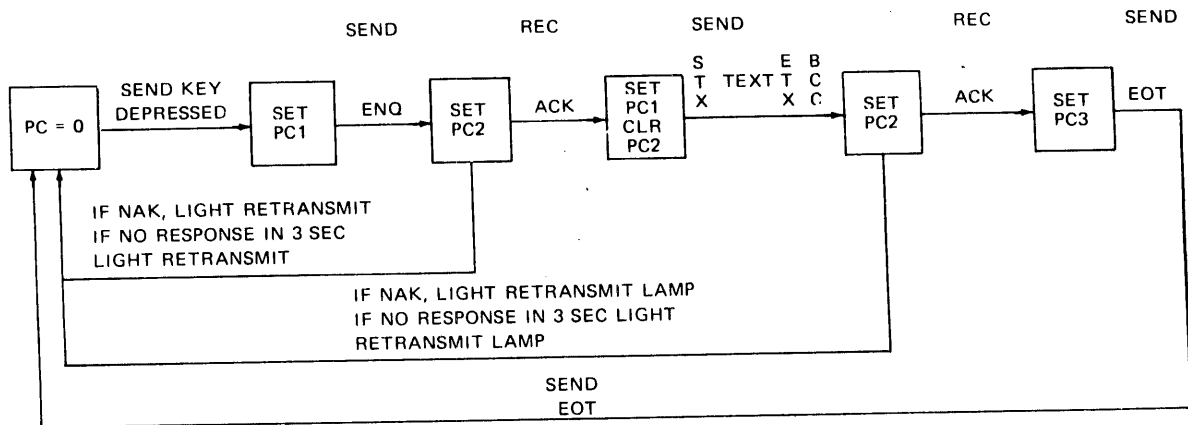


Fig. II-78 POINT TO POINT RECEIVE

Functional Detail



POINT TO POINT TRANSMIT

Refer to Fig. II-79 and related prints.

With the Send key depressed, PC1 is set (315), allowing ENQ to be generated and transmitted. Simultaneously, the 3 second Retransmit timer is triggered and PC2 is set for the ACK or NAK response. If the response is NAK, or if there is no response, allowing the 3 sec timer to time out, PC2 is reset, the retransmit lamp is lighted and the operation is complete. If the response is ACK, PC2 is reset and PC1 set and the STX character is generated (314) and retransmitted. ORQ (Output Request – 316) generates SPMCY (111) and the text of the message is read from memory and transmitted. After sending the complete message, PC2 is again set for the ACK or NAK response and the 3-second retransmit timer (310) is triggered. If the response is NAK, or if there is no response allowing the retransmit timer to time out, PC2 is reset, and the retransmit lamp is lighted. If the response is ACK, PC3 is set for an EOT response (see Polling Operation, Fig. II-69).

CONTENTION MODE

Contention mode is similar to point to point in that the terminal (B9353) has some control in initiating an operation.

In point to point only one control unit and one display is used, but in contention being a multipoint network, numerous control units and displays are used. The Master Station places the multipoint network in a contention state by sending O U U O. In this state, if a terminal wants to transmit, the send key is depressed and

the sequence D D p N is transmitted. Upon receiving that sequence, the Master Station will send a poll message to that terminal and the terminal responds in the normal

S A A S E B  
 manner with O D D T TEXT T C. The receiving of  
 H 1 2 X X C  
 the poll sequence takes the entire network out of contention mode. Where upon the Master Station will have to reinstate it if desired.

DETAILED DESCRIPTION

(Fig. II-80 and related prints)

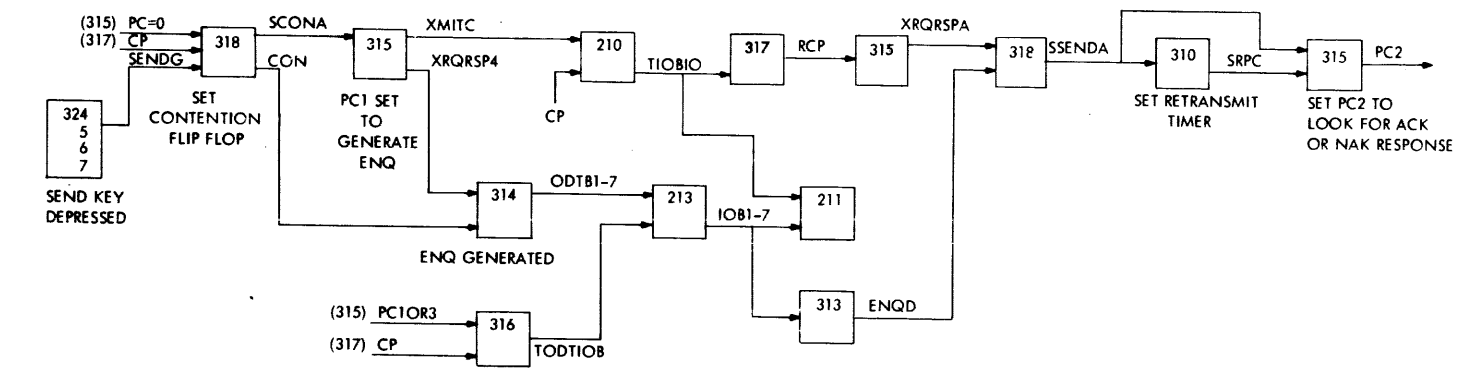
Upon receiving the O U U O sequence, the control unit will accept the CON character as being meant to place the unit in contention mode. Two characters after EOTD – 321, CON is received, and SCONB is generated, setting the Contention Flip-flop (318) and the control now idles. When the Send key is depressed, SENDG (324,5,6,7) causes SCONA to be developed, setting PC1 (315). With PC1 set, XMITC is generated, setting the XMIT Flip-flop (210) and triggering the Bit Time (209). Simultaneously, the character counter (319) is triggered and as it counts, AD1, and AD2 and P (Poll Character) are generated and placed on the ODTB1-7 lines to be sent to the I/O Buffer (213). Having enenerated the P character, XQRSP40 allows the ENQ to be generated (314) and placed on the ODTB1-7

A A E  
 lines. With the D D p N sequence generated, the control goes to PC2 to wait the poll sequence from the master station. The poll sequence will immediately set the PCC to zero and the operation proceeds as a normal polling operation (see Fig. II-67).

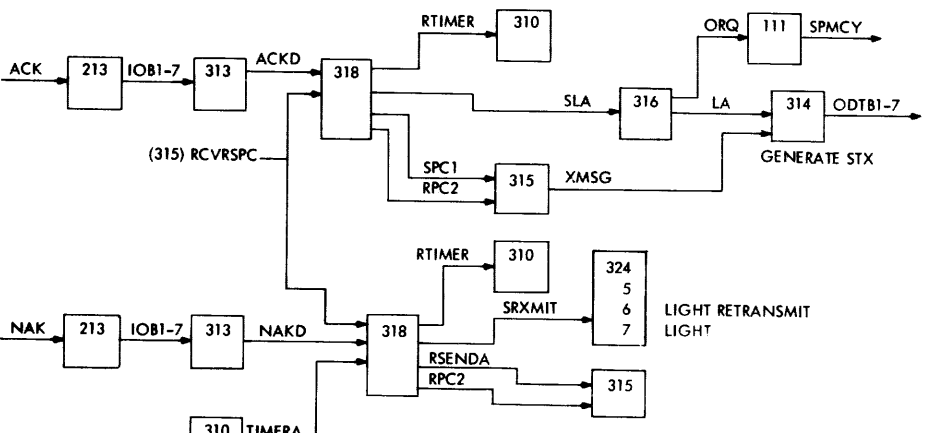
REPOSITION CURSOR

The cursor can be repositioned from the master control to any position within the data on the screen, if anywhere in the message after STX has been received, the

Functional Detail

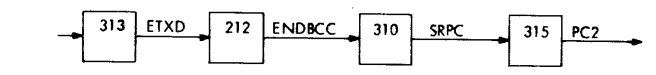


SEND ENQ - START RETRANSMIT TIMER - LOOK FOR ACK OR NAK RESPONSE

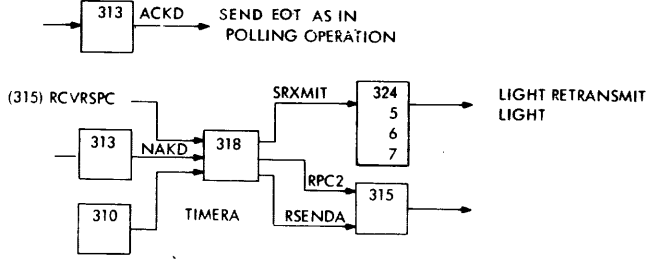


TIMERA GENERATED IF NO RESPONSE WITHIN 3 SECONDS.

RECEIVE ACK - SEND STX - TEXT  
 RECEIVE NAK - LIGHT RETRANSMIT LIGHT  
 NO RESPONSE - LIGHT RETRANSMIT LIGHT



SEND ETX, BCC - SET RETRANSMIT TIMER - LOOK FOR ACK OR NAK RESPONSE



RECEIVE ACK - SEND EOT AS PER POLLING OPERATION  
 RECEIVE NAK - LIGHT RETRANSMIT LIGHT  
 NO RESPONSE - LIGHT RETRANSMIT LIGHT

Fig. II-79 POINT TO POINT TRANSMIT

Functional Detail

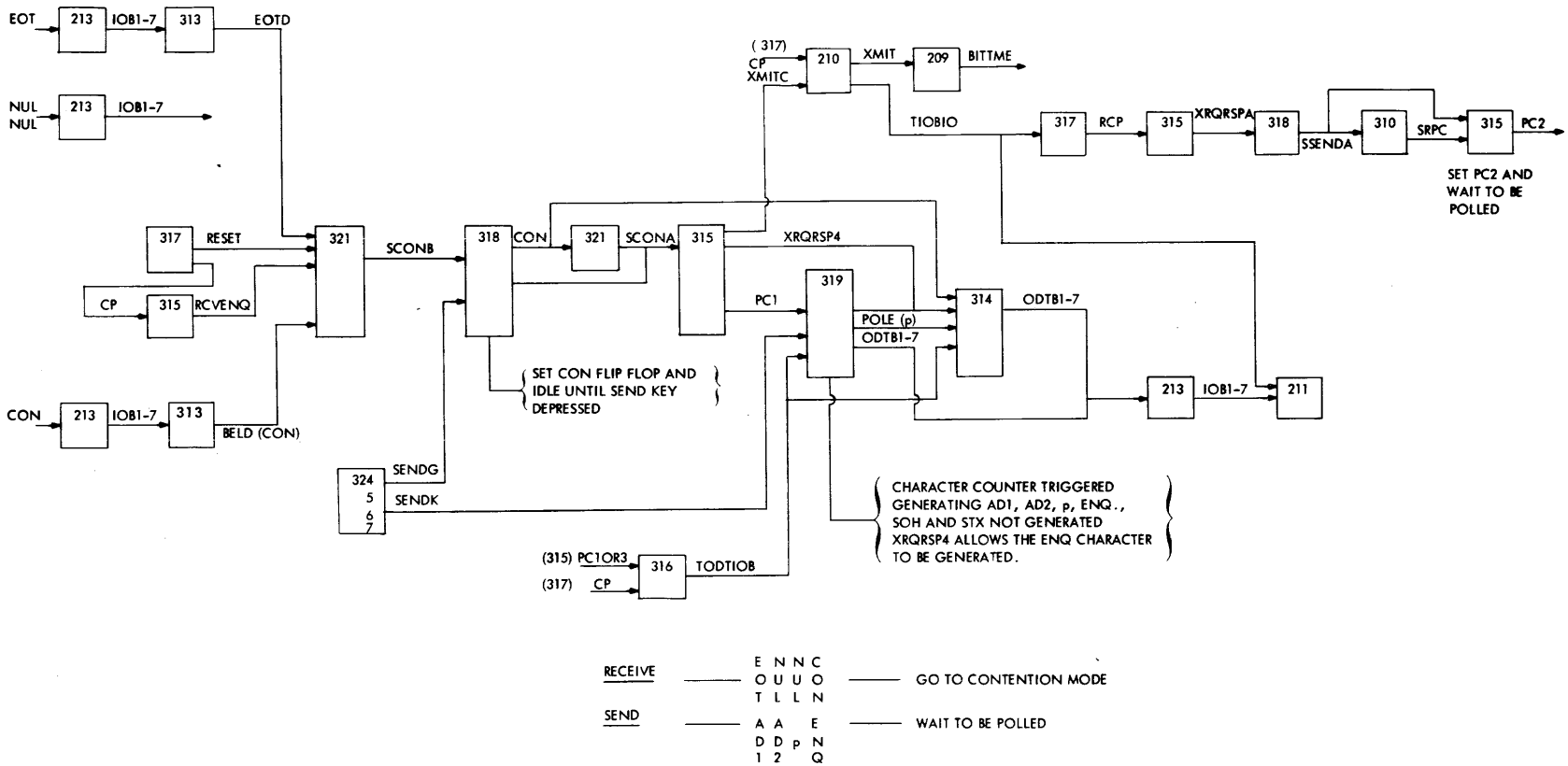


Fig. II-80



## Functional Detail

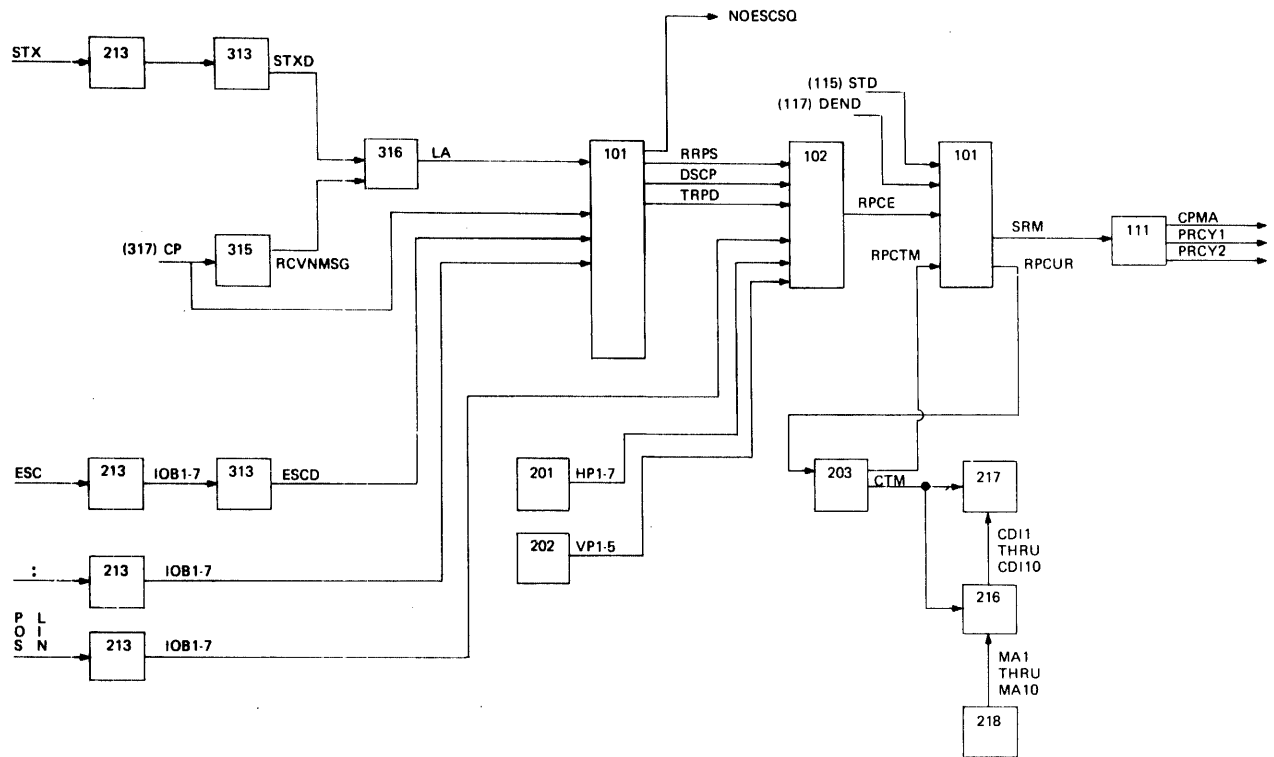


Fig. II-81 REPOSITION CURSOR

E P L  
 sequence S : O I is received. ESC indicates that a special  
 C S N  
 control sequence is to be initiated. The character  
  
 : indicates the special sequence is a Reposition Cursor  
 P  
 function. O is a character corresponding to the horizontal  
 S L  
 location to which the cursor is to be moved, and I is a  
 N  
 character corresponding to the vertical location to which  
 the cursor is to be moved.

The horizontal and vertical positions are placed into  
 the Reposition Cursor Storage Register (101). DSCP clocks  
 the POS character from IOB1-7 into Reposition Cursor  
 Storage 1 thru 7, and TRPD clocks the LIN character from  
 IOB1-5 into Reposition Cursor Storage 8 thru 12. When the  
 Display Sequence associated with that particular screen  
 begins, the contents of the Reposition Cursor Storage  
 Register are compared to the H.P. and V.P. counters as they  
 are counted up. When comparison exists, RPCE (Reposition  
 Cursor Equal — 102) causes RPCUR (Reposition Cursor —  
 101) to be developed, generating CTM. CTM gates and  
 clocks the contents of the Memory Add Register — 218,

which has been counting up as the H.P. and V.P. counters  
 are counted, to the Cursor Add Register (217).

## DETAILED DESCRIPTION

(Fig. II-81 and related prints)

The detection of the ESC character causes the level  
 NOESCSQ to be made true and this level is held true as:

P L  
 O I are received. Preventing memory cycles from being  
 S N  
 initiated on these characters, NOESCSQ causes  
 RCVNMSG-315 to be disabled, in turn preventing IRQ-318  
 from being developed.

When the Display Sequence is finished, a Put Get  
 sequence is initiated in the normal manner to place the  
 contents of the Cursor Add Register into reserved core.  
 (See Put Get sequence.) Also, at the end of the Display  
 Sequence, DEND is generated and develops SRM (Start  
 Receive Mode — 101) to read out the cursor address from  
 reserved core. This occurs after the Put Get sequence is  
 completed. Put Get has highest priority, so after it is  
 finished, SRM can initiate the PMCY to read the cursor  
 address and place it into the PMA register (219). From this  
 point on, the operation continues in the normal manner as  
 if no interruption had occurred.

Functional Detail

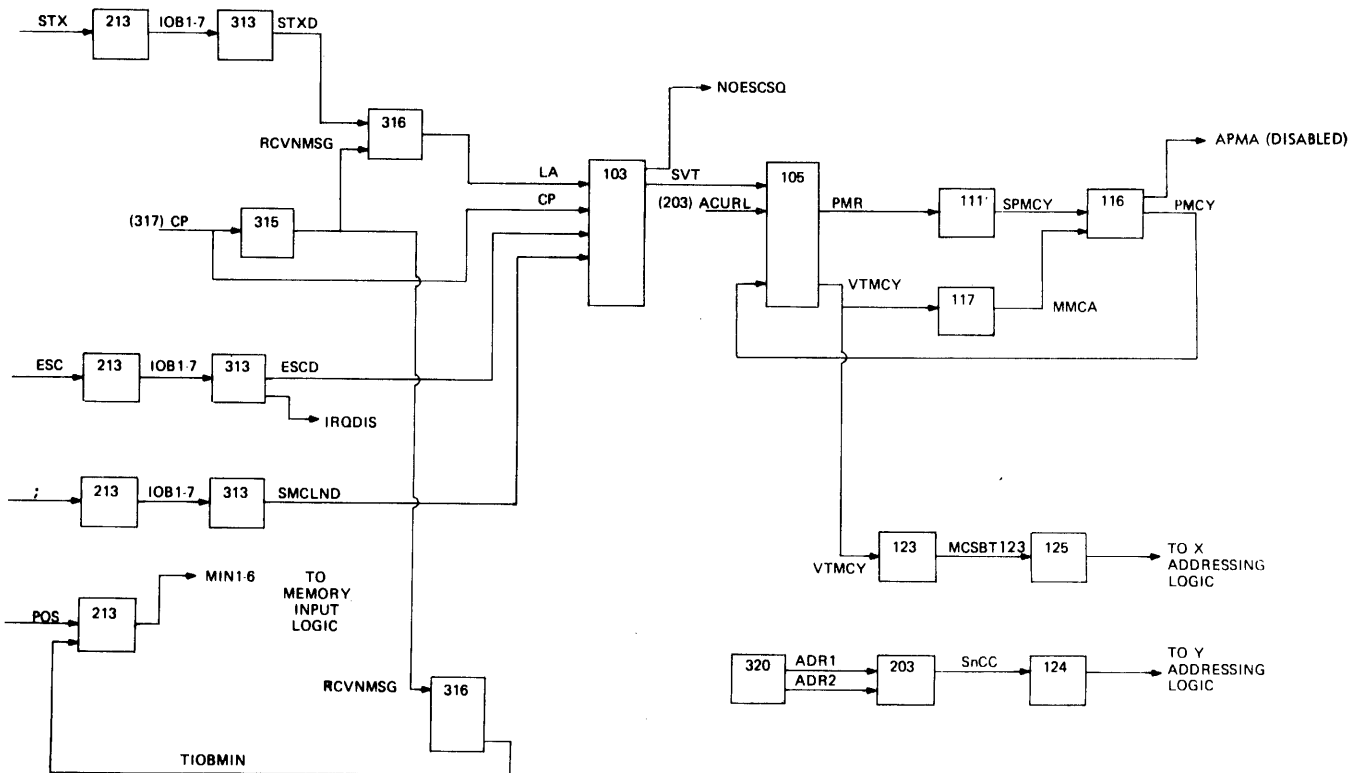


Fig. II-82 SINGLE VARIABLE TAB SET

SINGLE VARIABLE TAB OPTION

A single variable tab stop can be inserted from the master station if anywhere in the text the sequence **E P S : O C S** is received. ESC indicates that a special control sequence is to be initiated. Character ; indicates that it is a variable tab operation, and O is a character corresponding to the horizontal location of the tab on the screen.

When the variable tab sequence is received, it is not until the next Put sequence is complete that the variable tab information is used. At that time, the variable tab location in reserved core is addressed, and the variable tab information is written into that location.

There should be a period of approximately 17 ms between receipt of the variable tab information and any additional text information during which time nothing is received. Any information received during this time will terminate the variable tab operation.

DETAILED DESCRIPTION

(See Fig. II-82) and related prints)

Upon detecting the ESC and ; characters, the level

SVT (Set Variable Tab – 103) is developed. ESC is prevented from being written into memory by IRQDIS (313), and ; is prevented by NOESCSQ (103). When the next Put sequence is complete, ACURL (Allow Cursor Level – 203) gated with SVT generates SVT. (There is a period of approximately 13 microseconds between Put and Get, during which time the variable tab information is written into memory.) SVT causes PMR to be developed, initiating a Processor Memory Cycle. PMCY in turn develops VTMCY (105), which develops the X addressing necessary to address the reserved location assigned to the variable tab address. The Y addressing is a function of the ADR1 and ADR2 terms which would have been received in the initial select sequence. The O character which is in the

I/O Buffer at this time is now gated onto the MINI-6 lines and written into the reserved variable tab location in memory. The VTMCY level causes bit 6 of the POS character in the I/O Buffer to be complimented when placed on the MINI-MIN6 lines. Thus, if bit 6 is a zero, it goes into MIN6 as a one, and vice versa. As a result if the POS character is a Space, it goes into memory as 000000 instead of 100000. The operation now continues in the normal manner as if no interruption for inserting a variable tab address had occurred.

Functional DetailSYNCHRONOUS OPTION

## RECEIVE

For the control unit to be considered in sync for a receive operation, two consecutive sync characters must be received. The first character is sampled bit by bit as it is assembled in the I/O Register until a SYNC configuration is detected. If the first sync character is legitimate, the second sync character should arrive 8 bit times later. Therefore, the second character is sampled only when it has been completely assembled in the I/O Register.

## DETAILED DESCRIPTION

(Fig. II-83) and related prints)

In synchronous mode, all bit time is generated by the Data Set. Consequently, as data arrives from the remote station, the receiving Data Set provides pulses for storing the data into the I/O Register.

DDR (Receive Timing – 311) is the bit timing pulse from the Data Set. DDR occurs in the center of the data bit on the BB line. DDR causes SLIO (Shift Left I/O – 211) and IOCLK (211) to be generated, both of which occur in the center of the bit to strobe the contents of the BBR line into the C1 Flip-flop (208). The output of the C1 Flip-flop is in turn clocked into the I/O Register (211). As each bit is clocked into the I/O Register, the level SYND (Sync Detect – 211) is sampled. When SYND goes false, this indicates that a SYNC configuration is in the I/O Register. At this time, SPREIO (Set Preset I/O – 210) is generated presetting the I/O Register to all ones and presetting the C1 Flip-flop (208). By presetting the C1 Flip-flop, a zero is placed into the I/O 7 Flip-flop of the I/O Register ahead of the next incoming data bit. This zero acts like a flag to

indicate when the I/O Register has a complete bit character in it. SPREIO also causes Flip-flop S1 to be set (209) indicating that the first SYNC configuration has been detected. Because S1 is set, gate C3-I is conditioned and the level SOF goes false. SOF being false, allows SPREIO to be generated only when a character is completely assembled in the I/O Register rather than any time a SYNC configuration is detected. When the next character is completely assembled in the I/O register, the level NCCXMT (210) causes SPREIO to be generated and one of two things can now happen. If the character in the register is a SYNC, character SYND goes false, disabling gate C3-I (209) and simultaneously SPREIO causes S1 to be reset and S2 set, indicating 2 consecutive SYNC characters. If the character in the I/O Register is not a SYNC, character SYND is true, gate C2-I is enabled, resetting Flip-flop S1 and preventing Flip-flop S2 from being set thereby requiring two more consecutive SYNC characters.

The setting of Flip-flop S2 allows TIOIOB (Transfer I/O register to I/O Buffer – 210) to be developed anytime a character is assembled in the I/O register, and the I/O Buffer is empty (CP=0). Anytime a SYNC character should be assembled after S2 has been set, the level SYND goes false, preventing TIOIOB from being shifted into the I/O Buffer, meaning it cannot be placed in memory nor have a BCC developed on it.

The remainder of the operation is identical to any asynchronous operation except that when a complete message has been received, the level RBLKCHK (Reset Block Check – 210) which is developed after the BCC character has been received causes unconditional reset to be felt on Flip-flops S1, S2 and S3.

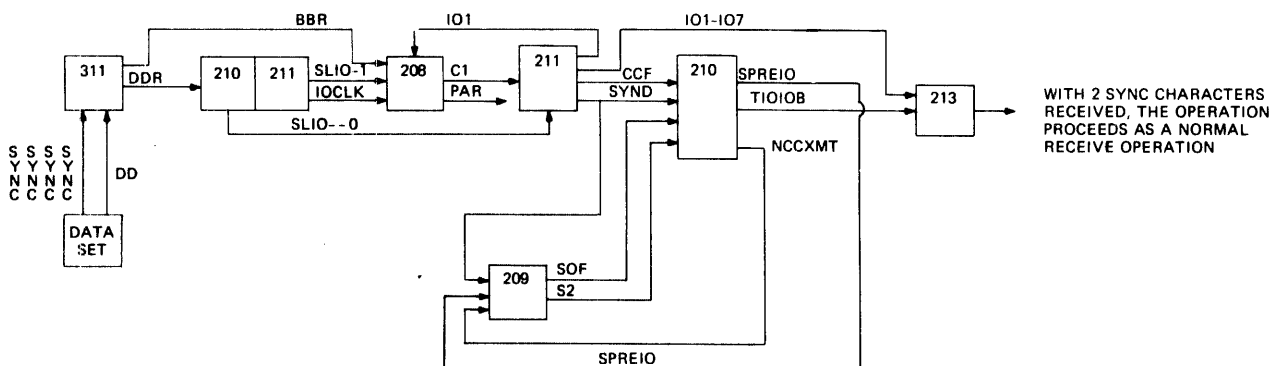


Fig. II-83 RECEIVE 2 CONSECUTIVE SYNC CHARACTERS

## Functional Detail

## TRANSMIT

In any synchronous transmission, 4 sync characters are transmitted before the data. These characters are hardware generated in the control unit.

## DETAILED DESCRIPTION

(See Fig. II-84 and related prints)

When something is to be transmitted, the PC1 Flip-flop (315) is set generating XMITC. XMITC in turn generates SXMIT (210) and sets the XMIT Flip-flop (210). The output of the XMIT Flip-flop is used on 208 to generate CAD (Request to Send) turning the carrier on in the data set. The data set produces pulses that occur at bit rate (DBR). After a delay of approximately 250 msec, the data set also develops CBR (Clear to Send). CBR, along with the DBR pulses, sets the TCE (Transmit Clear Enable Flip-flop — 209). The TCE level and the DBR pulses are used on 211 to generate CTS pulses. The CTS pulses develop SLIO pulses (210) which, gated with clock pulses (CLKE) on 211, develop IOCLK. SLIO shifts the character down in the I/O register (211) and IOCLK clocks the contents of IO1 into the OBD Flip-flop (208) and onto the BAD line. The generation of the 4 sync characters is accomplished as follows.

SXMIT clears the SYNC counter (S1, S2, S3, — 209) and causes XMTSLS (Transmit Set Load Sync — 209) to be generated. XMTSLS causes TSYNIO (Transfer Sync to I/O register — 210) to be developed. TSYNIO causes the SYNC bit configuration to be clocked in the I/O register. When the SYNC character is shifted out of the I/O register, the level PARIN (Parity Insert — 211) develops when the I/O register contains all 1's, indicating it is empty. 1's are forced in as the character is shifted out by C1 standing by true. PARIN, in addition to placing the contents of the PAR Flip-flop (208) into the ODB Flip-flop and consequently onto the BAD line, causes LDCHAR (Load Character — 210) to be developed. LDCHAR causes sync counter S1 to be set, and generates XMTSLS to cause another SYNC character to be forced into the I/O register. This operation continues with the Sync counter being counted as each

SYNC is generated and transmitted. After the fourth SYNC has been developed SYNC counter Flip-flop S3 is set, preventing XMTSLS from being generated again. S3 also is used to develop TIOBIO (210) allowing any character that has been developed in the meantime (ACK, NAK, SOH, etc.) to be placed in the I/O register to be transmitted.

At the end of any transmission, the level XMITC (315) is dropped, causing RXMIT (210) to be generated. RXMIT causes the Sync counter (209) to be unconditionally reset, requiring the generating of 4 sync characters again before any transmission of data can occur.

## PRINT OPTION

A print operation can be initiated from the Master Station or from the keyboard. If from the Master Station, all that is required is that bit 3 of the AD2 character in the select sequence be on. Because the print command requires bit 3 of the AD2 character, the control unit must have a selection address card (320). Having this card in the unit means that even if the control unit is on a point to point network, it must be handled as a multipoint network with the unit being polled and selected. When initiated from the Master Station, all printing begins from screen position 1 of the screen selected until the ES is detected (Essentially a Frame Print). The printing does not begin until the control unit sends ACK for a message received with no errors, and gets the EOT in response. All printing takes place during the display sequence(s) associated with the selected screen. As the message is being printed, the cursor advances across the screen at the printer rate. During the printing operation all keyboards are locked, and if the unit is polled the response is EOT and if selected the response is NAK.

When initiated from the keyboard, two types of print operations can be performed, Frame and Selective print. Frame or normal print requires depressing the print key alone. Printing begins at screen position One and continues until ES is detected. Selective print requires depressing the Shift and Print keys. Printing then begins from the cursor location, and continues until either GS (Group Separator) or ES is detected.

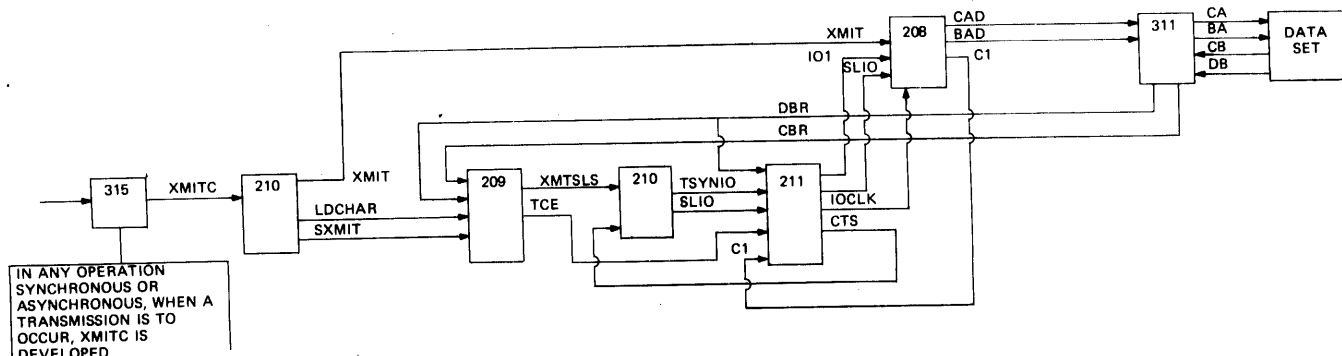


Fig. II-84 TRANSMIT 4 CONSECUTIVE SYNC CHARACTERS

All print operations, whether from keyboard or  
 C C L A C C L  
 remote send R R F D R R F to the printer before any  
 2

of the text is sent. The AD2 character is the address of the screen that is being used with the printer. Detection of NL or HPE80 causes C C L to be generated and sent to the  
 R R F

printer. Detection of ESC causes a one character lull while waiting for the character associated with the ESC. When the character associated with ESC arrives, bit 3 in the POR (Printer Output Register) is turned on, thus causing the proper character to be sent to the printer.

#### DETAILED DESCRIPTION

(See Fig. II-85 and related prints)

If the print is initiated from the Master Station, SAD2 (the character in the I/O buffer is the second address character) and IOB3 (Print request bit) set ADR3 Flip-flop (331). When the control unit has received the complete message with no errors, responded with ACK, and gotten an EOT back in response, SPRQ (Set Print Request – 331) is generated. At the FRAM time corresponding to the address of the screen that is to be printed (ADR1 and ADR2 levels) the corresponding print request flip-flop is set (PRQ0 thru PRQ3 – 333). If the print command is from the keyboard, IODC and EL3 (Decode of Print) will set the related Print Request Flip-flop at the FRAM time that occurs with the print key that was depressed. The output of the PRQ Flip-flops develop related PS0-PS3 levels (329) which are used on 331 to set the PA (Print Address) Flip-flops. These flip-flops retain the address of the screen that is to be printed. At the time that the PA Flip-flops are set, CR1 is also set, causing the following to happen.

1. STP – Start pulse generated, setting Flip-flop C1-I and H (329). Flip-flops C1-I and H, C1-N and E1-I and H are used only in a frame print operation. These flip-flops are used to develop RCMAP (Reset Cursor Memory Address Register) and CTM (a clock for the Cursor Address Register) to preset or clear the CMA register so that printing will begin at screen position 1. Printing does not occur until after the necessary CR and LF characters have been generated however.
2. If PORVD (Printer Output Register Void (empty) indicates the POR is empty, PCRE (Printer Carriage Return Enable (331) is developed. PCRE places the CR character at the inputs to the POR.
3. PCRE develops LPOR, immediately setting PORS2 (2nd stop bit flip-flop – 335) and allowing one PCLK (print Clock – 331) to be developed with the next clock pulse.

This next PCLK pulse causes PORS1 (1st stop bit flip-flop) to set, and clocks the CR character into the POR.

Because the POR now contains a character, PORVD (335) is disabled, in turn causing LPOR to be disabled (331) preventing any additional PCLK pulses to be developed by LPOR. The PCLK pulses to be generated now must occur at the rate at which the printer operates.

This is accomplished through the use of a binary counter (329) which is continually counting, producing an SRPOR (Shift Right POR) pulse at the required bit rate (110 BPS – 9.09 ms). At the time that CR1 Flip-flop was set, the PRNT Flip-flop (331) was also set, PRNT allows the contents of the POR to be gated to the POT Flip-flop (339) to be sent to the printer. When the character in POR has been shifted out, PORVD causes CR2 to be set and the process repeats. When CR2 has been shifted out LF is set, repeating the process. When the LF character has been shifted out, the Data Flip-flop (331) is set causing TPAPOR (Transfer Print Address to POR) to gate the contents of the PA Flip-flop to the POR and cause it to be shifted out. Setting the Data Flip-flop conditions CR1 again so that when PORVD indicates an empty condition CR1 is again set, and the entire CR CR LF sequence is again generated. The Data Flip-flop also conditions the PAD (Print Addressed Data – 331) Flip-flop which is set at the same time as CR1. With PAD set, when the CR CR LF sequence is complete, PRQ is generated (Print Request) instead of TPAPOR. PRQ (Print Request) sets Flip-flop E1-M (329) disabling SRPOR preventing any shift right pulses. At this point, having printed CR CR LF AD2 CR CR LF, one of two things can happen:

1. If the operation is a Selective Print operation, nothing more happens until the display sequence associated with the screen to be printed begins. When cursor compare occurs during the display sequence (CDPOS), the level PRDA (Print Data – 329) is generated, resetting Flip-flop E1-M, enabling SRPOR again for shift right pulses. Also, PRDA causes TMIRPOR (Transfer MIR to POR – 331) to be generated. TMIRPOR gates the contents of MIR into the POR register to be shifted out and printed. PRDA also generates CCAU upcounting the CMA register.
2. If the operation is a Frame Print operation, Flip-flop C1-I and H would have been set earlier, and Flip-flop C1-N is set with next STD pulse. If this is the STD pulse for the screen that is to be printed, RCMAP (Reset Cursor Memory Address register) is generated, to preset or clear the CMA register to a count of 5. (Equivalent of position 1 on the screen.) One clock later Flip-flop E1-I and H is set generating CTM, which does the clocking for clearing the CMA register.

Because Flip-flop E1-I and H is set, indicating the CMA register is cleared, Flip-flop C1-I and H is reset, allowing PRDA to be generated when the display sequence begins and cursor compare (CDPOS) occurs. (Because CMA is cleared, CDPOS occurs on the first memory cycles, which

Functional Detail

is equivalent to screen position one). PRDA, as before generates TMIRPOR gating MIR to POR and PRDA also resets Flip-flop E1-M and develops CCAU to upcount the CMA register enabling SRPOR, generating shift right pulses. Detecting either ES or if in selection print CS causes ENDPRT to be developed, resetting the PAD Flip-flop, ending the operation. Detection of NL causes TMIRPOR to be disabled, and causes CR CR LF to be developed in the manner described earlier.

Detection of ESC causes TMIRPOR to be disabled, and sets Flip-flop D2-N generating ESCR, (331) forcing the 3 bit into the POR register.

BREAK

The Break option can be used on leased or switched line. It is a method whereby the B9353 can signal the master station to stop transmitting. It also allows the B9353 to be signalled by the master station (during the time the B9353 is transmitting) to stop transmitting. The B9353 alerts the operator that a break has been detected, by lighting the retransmit light. A Break signal is transmitted by causing the BA (Transmit Data) line, which normally stands by in a marking state, to go into a space state for approximately 250 milliseconds. A Break signal is detected when the BB (Receive Data) line, which normally stands by in a marking state goes to a spacing state for approximately 250 milliseconds.

DETAILED DESCRIPTIONTransmitting a Break

(Print 208, 327, 402 and Fig. II-86)

Depressing the Break key on any keyboard causes the corresponding BRKn-0 term to go false (402). This level, through the related keyboard print (324, 5, 6 or 7) generates SBRK-1. SBRK is used on 208 and only has an effect if the control unit is the terminal that has been selected (SLCT), is in receive mode, and is in that PC state of receive mode (PC2 is set) where data is actually being received (RCVMSG). With these conditions present, Flip-flop B2-I and H is set, generating CAD-0 (Request to Send) turning the carrier on. When Clear to Send is received from the data set (CBR – received approximately 200 ms after CA sent to the data set) the delay circuit consisting of buffer AO, capacitor C4, resistor R6, and inverters A1 is triggered causing the output of inverter A1 to go false for approximately 250 milliseconds. This false disables gate BO, causing the term BAD-1 to go false (spacing or break condition) for approximately 250 milliseconds. At the end of the delay, BAD-1 goes true again and Flip-flop B2 is reset.

Receiving a Break

(Print 208 Fig. II-87)

The Break signal can only be sensed if the Control

Unit is in transmit mode, and in that PC state of transmit (PC1 Set) that indicates that data is actually being transmitted. With these conditions met, should BBR (receive data) line go from a true (marking) to a false (spacing) the delay circuit consisting of buffer A0, capacitor C3, resistor R3, and inverters A1 is triggered, and Flip-flop CO-N is set. If the BBR line remains false until the delay circuit times out, gate BO is enabled, setting Flip-flop CO-H, in turn generating XMTBRK (Break detected during transmit – resets Send and PC Flip-flops on 315) effectively terminating the transmission, and SRXMIT (SET Retransmit – Sets retransmit flip-flop on 324, 5, 6 or 7, lighting the retransmit light). Should the BBR line go true before the delay circuit has timed out, Flip-flop CO-N is immediately reset, preventing gate BO from being enabled. Because the Break signal is not present long enough (may have been noise rather than a break) the data being transmitted is not interrupted and the transmission is completed in the normal manner.

DISCONNECTPRINT 208

The disconnect option is used only on switched (dial) lines, and allows the line to be disconnected (hung up) either by way of the keyboard or by receiving the character sequence DLE, EOT from the master station (auto disconnect – available by jumpering E3 to E4 and inhibited by jumpering E3 to E5). The character sequence DLE, EOT causes the B9353 to disconnect the line.

DETAILED DESCRIPTION

The Flip-flop DISCON stands by in a reset state, causing the term CDD-0 (data terminal ready) to be false. This indicates to the data set that the terminal (the B9353) is in a ready condition. The reset input to the DISCON Flip-flop is CCR-0 (data set ready) and stands by false indicating that the data set is in a ready condition. Depressing the disconnect key on the keyboard causes the terms IODC and EL2 to go true, setting the disconnect flip-flop and in turn causing the level CDD-0 (data terminal ready) to go true. The data set, upon detecting CDD going true, immediately disconnects or hangs up the line. Simultaneously, the data set causes CCR (data set ready) to go true indicating that having disconnected the line, it is now in a not ready condition. CCR going true resets the DISCON Flip-flop, allowing CDD to go false again. Having disconnected the line, the data set cannot go into a ready state again unless one of the following conditions occur:

1. The data set is strapped for auto answer and a ring signal is detected.
2. The data set is not strapped for auto answer, a ring is heard and the operator places the data set in a ready condition.

## Functional Detail

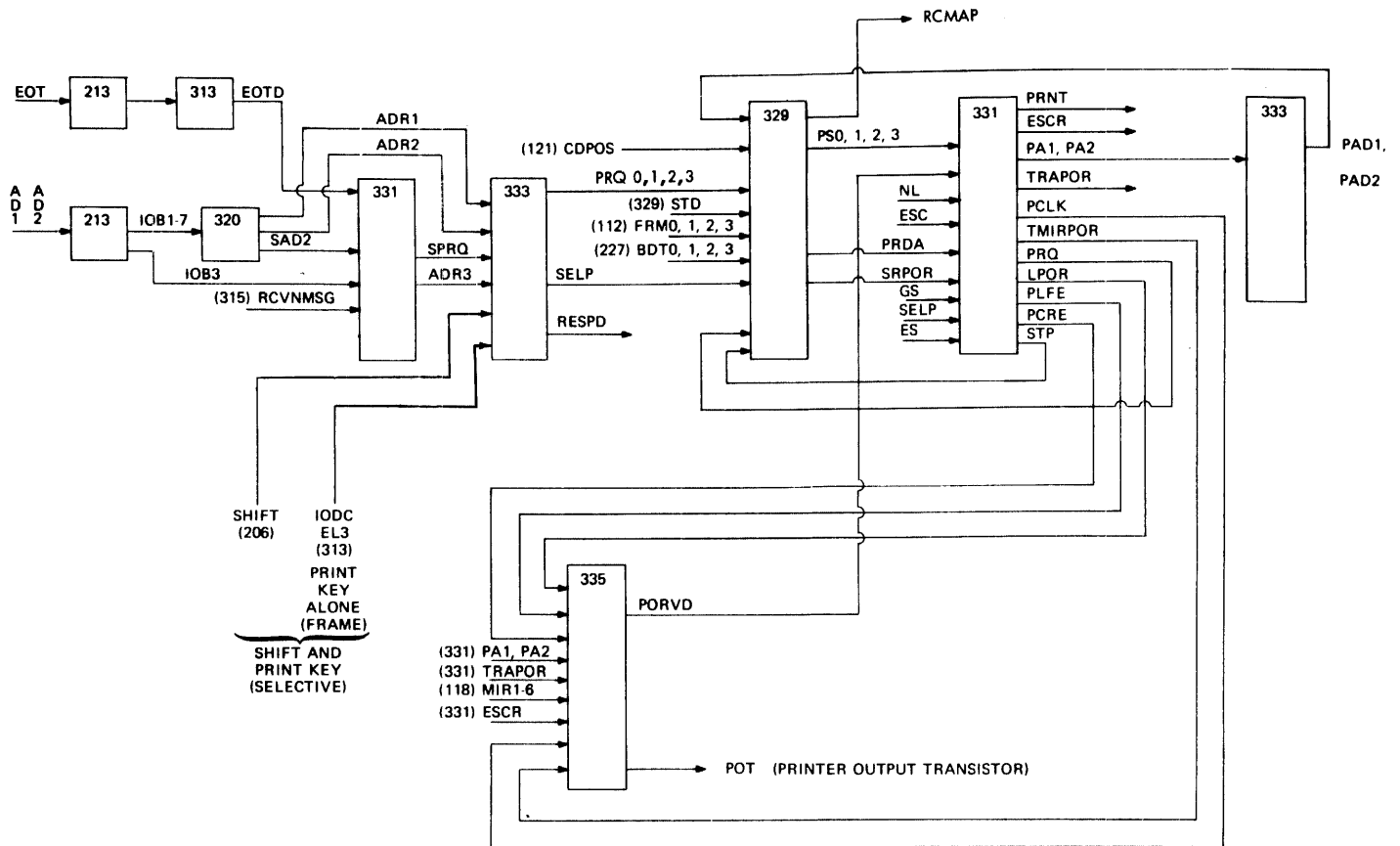


Fig. II-85 PRINT OPTION RECEIVE PRINT COMMAND FROM REMOTE  
OR  
INITIATE FRAME OR SELECTIVE PRINT FROM KEYBOARD.

3. The operator dials up a number and having made connection places the data set in a ready state.

Disconnecting the line by way of the master station requires receiving the character sequence DLE and EOT. Receiving the DLE character sets the DLE Flip-flop, conditioning the set input of the DISCON Flip-flop. The EOT, being the next character received causes the DISCON

Flip-flop to be set, disconnecting the line. Because the DLE Flip-flop is set only from the time the DLE is detected in the I/O buffer until immediately after the following character is in the buffer, (CP causes DLE to be reset), the characters must be received in the sequence DLE, EOT in order for disconnect to occur.

Functional Detail

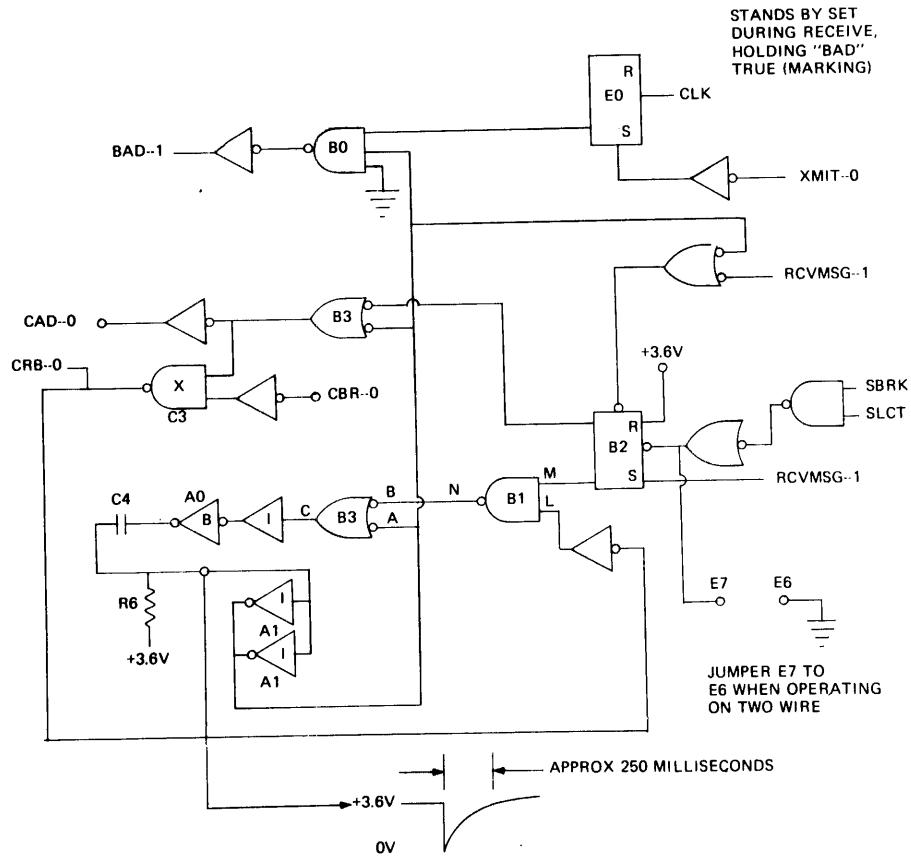


Fig. II-86 TRANSMIT BREAK SIGNAL

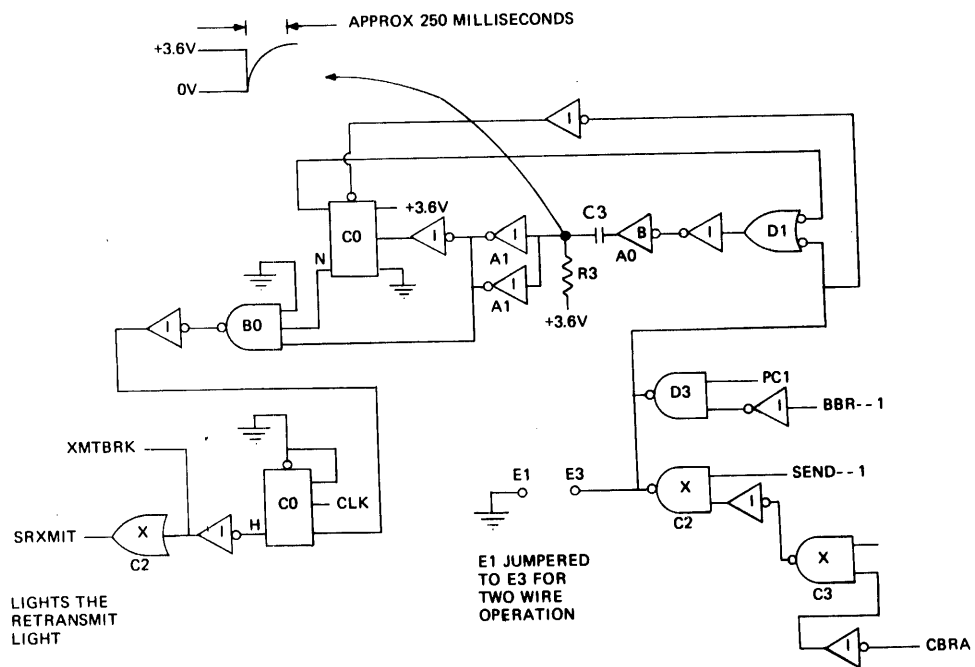


Fig. II-87 RECEIVE BREAK SIGNAL





Circuit Detail

LOGICAL ELEMENTS

Resistor Transistor Micro-Logic (RTuL) integrated circuits are used throughout the Input and Display system. Using NAND/NOR logic, a logical ONE is ground potential and a logical ZERO is from +1 to +3.6 volts, depending on the load. All elements are in-line units with 14 pins identified as shown in Figure III-1. The ground connection for all units is pin D while the Vcc (+3.6vdc) for all units except the Expander Gate is pin K. The Expander Gate does not require Vcc. For debugging purposes, the outputs of all elements, except the Dual Buffer, may be grounded, but may not be connected to a voltage. The buffer output may not be connected to ground or a voltage.

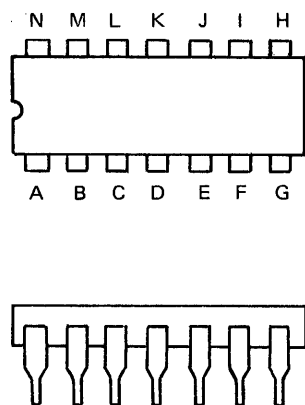


Fig. III-1

QUAD TWO-INPUT GATE

The quadruple two-input gate consists of 12 resistors and 8 transistors arranged to form four NAND/NOR gates, each with two inputs. Figure III-2 depicts circuit one using two transistors and three resistors. Circuits two, three and four are identical to circuit one. The input signals are applied to the bases of the two transistors through the 450 ohm resistors. When utilizing the circuit as a NAND gate, TRUE signals would be applied to both bases. This action causes both transistors to be cut-off resulting in the collector voltage going to approximately 1.5 volts. When operating as a NOR gate, a FALSE signal would be applied to either base causing that transistor to conduct. The collector then goes to ground producing a TRUE output. Refer to the truth table, Figure III-3 A, for the output signals of the NAND/NOR gate.

QUAD TWO-INPUT EXPANDER GATE

The quadruple two-input expander gate is identical to the Quad Two-Input Gate with one exception. The collectors are not connected to the Vcc supply of +3.6 volts. When a logic equation requires a NAND or NOR circuit involving three or more inputs, an expander gate may be

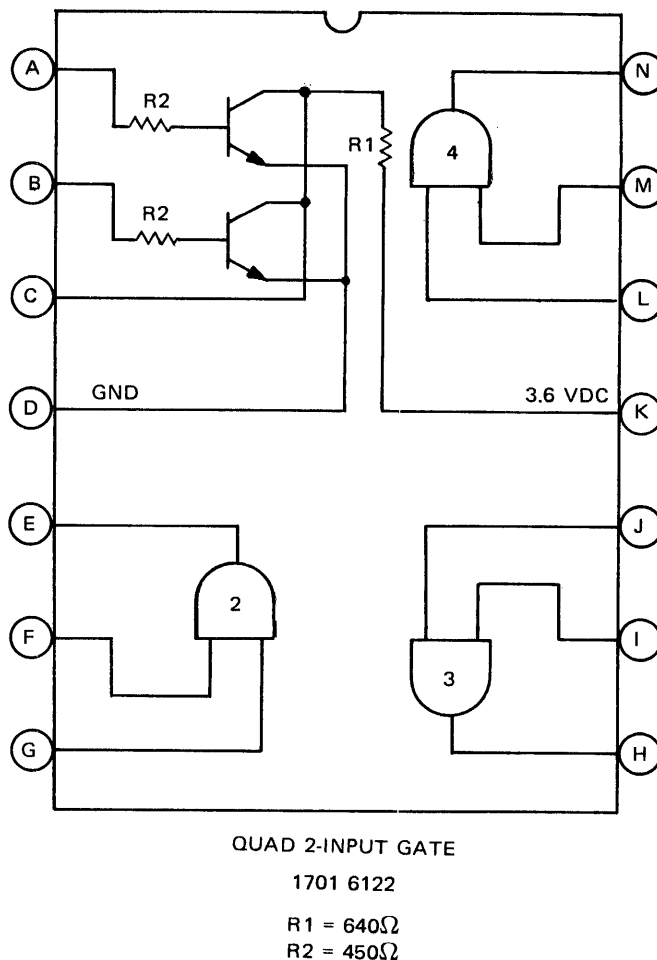


Fig. III-2 QUAD 2-INPUT GATE

| INPUT |   | OUTPUT |
|-------|---|--------|
| 1     | 2 |        |
| F     | F | T      |
| F     | T | T      |
| T     | F | T      |
| T     | T | F      |

A

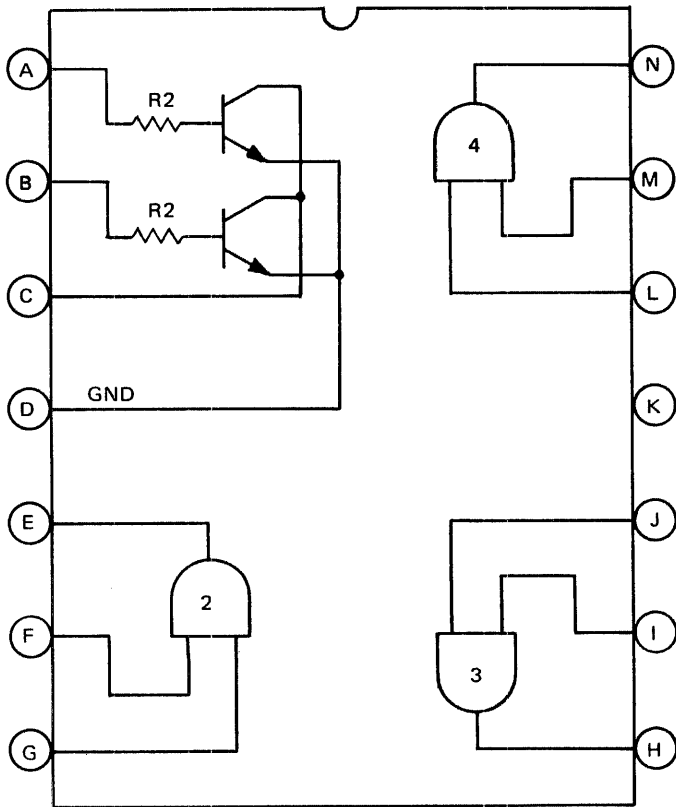
| INPUT |   |   | OUTPUT |
|-------|---|---|--------|
| 1     | 2 | 3 |        |
| F     | F | F | T      |
| F     | F | T | T      |
| F     | T | F | T      |
| F     | T | T | T      |
| T     | F | F | T      |
| T     | F | T | T      |
| T     | T | F | T      |
| T     | T | T | F      |

B

Fig. III-3

used in conjunction with a quad two-input gate or a triple three-input gate. Refer to Figure III-4. The collector output of the expander is connected to the output of the quad two or triple three-input gate forming a four or five input-gate.

Circuit Detail



QUAD 2-INPUT EXPANDER  
1706 8859  
R2 = 450Ω

Fig. III-4 QUAD 2-INPUT EXPANDER

Figure III-5 illustrates three examples of the use of the quad two-input expander element. In examples A and B, the output of the expander is connected to the output of the quad two-input and triple three-input gates respectively. Example C depicts an expander gate used as a two-input gate. All that is required is that the output be connected to +3.6 volts through a 680 ohm resistor.

TRIPLE THREE-INPUT GATE

The triple three-input gate is illustrated in Figure III-6 and consists of nine transistors and twelve resistors arranged to form three NAND/NOR gates, each with three inputs. When used as a NAND gate, all three inputs must be TRUE in order to produce a FALSE output. As a NOR gate, any FALSE in will produce a TRUE out. Refer to the truth table, Figure III-3 B.

HEX INVERTER

Consisting of six transistors and twelve resistors, the hex inverter is arranged to form six single input inverter circuits. Refer to Figure III-7. As with the previously described gate circuits, a TRUE input signal will produce a FALSE output. Conversely, a FALSE input signal will produce a TRUE output.

DUAL J-K FLIP-FLOP

The dual J-K Flip-flop element consists of twenty-eight transistors and thirty-six resistors arranged to form two flip-flops. Each circuit can be set, reset, cleared or preset. Refer to Figure III-8. A negative clock pulse, when applied to pin B will cause the outputs at M and N to assume the levels indicated in the table, provided the Reset/Preset input is at ground. When the Reset input is positive, the '0' output of the flip-flop will go TRUE and the '1' output will go FALSE. When the reset (R) input is

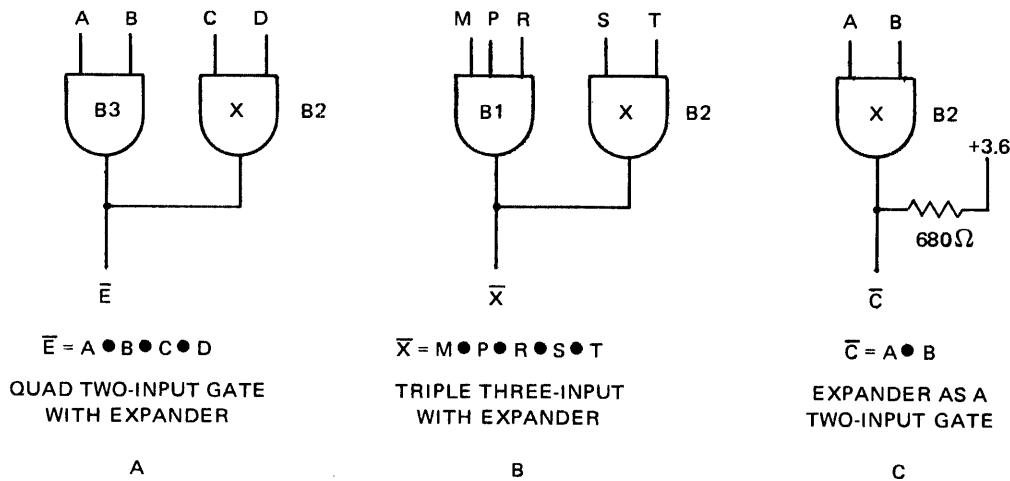
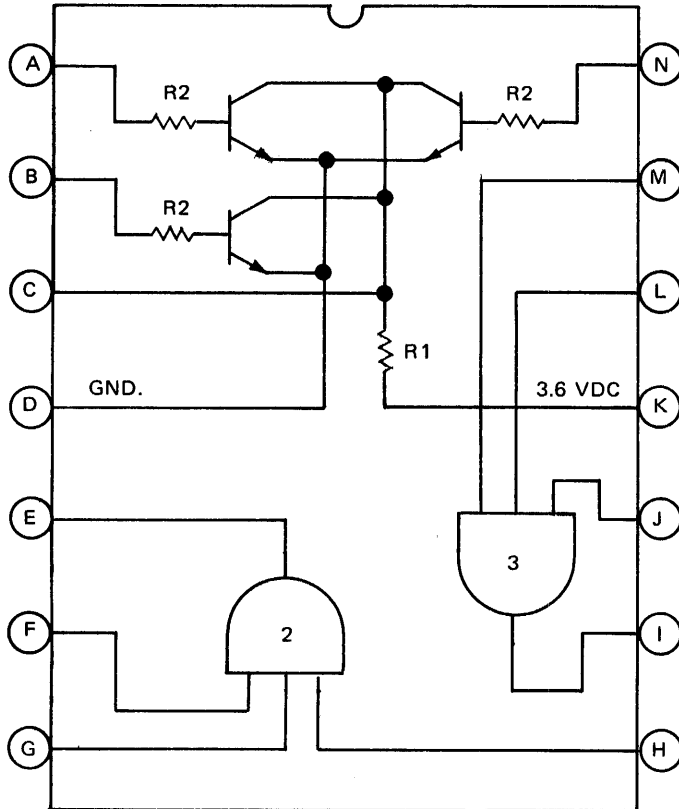


Fig. III-5 EXAMPLES OF EXPANDER GATE

Circuit Detail

used as a preset (P) input, all the input and output signals are considered as reversed. In this case, a positive signal applied to the Preset will cause the '1' output to go TRUE and the '0' output will go FALSE.



TRIPLE 3-INPUT GATE

1701 6155

R1 = 640Ω

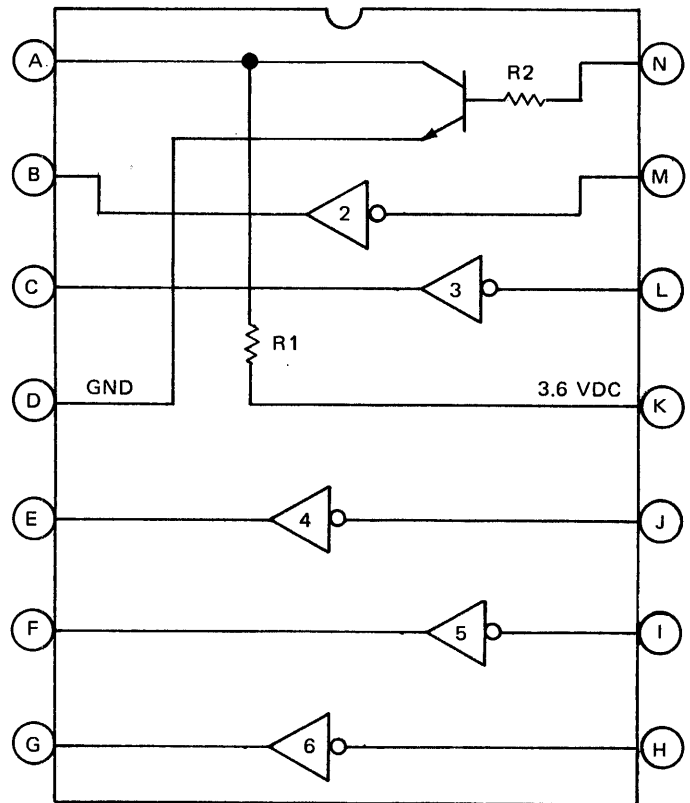
R2 = 450Ω

Fig. III-6 TRIPLE 3-INPUT GATE

DUAL BUFFER

The dual buffer element consists of six transistors and twelve resistors arranged to form two buffers. Refer to Figure III-9. Each buffer can be operated as an inverter, an inverter with an emitter-follower for drive or with another buffer as a single shot. The inverter alone would have pins M or F as inputs with the output at pins B and J respectively. As with the gating elements, a TRUE level in will cause transistor Q1 to be cut-off, allowing the collector voltage to approach +3.6 volts. A FALSE level in will cause Q1 to conduct, the collector then going to ground.

The most common application of the dual buffer element is the inverter with the emitter-follower. Operation of the inverter is as previously described above; however, the output is applied to the base of Q2. When the input at M is TRUE, Q1 is cut-off, and Q2 conducts, thereby



HEX INVERTER

1701 6130

R1 = 640Ω

R2 = 450Ω

Fig. III-7 HEX INVERTER

providing greater drive capacity than possible with the inverter alone. Transistor Q3 functions in the same manner as Q1. With a TRUE input, Q3 is cut-off and essentially out of the circuit. When the input goes FALSE, Q3 will conduct, pulling the output at C to ground.

Operation as a single-shot involves the use of two buffers connected as shown in Figure III-10. When the input to the first buffer goes FALSE, the TRUE output signal is applied to the coupling capacitor C. During the time that the capacitor is discharging, a TRUE signal is applied to the input of the second buffer, the output then going FALSE. The time duration of this FALSE level is determined by the RC network at the input of the second buffer and is calculated as follows:

$$T = .8 RC \text{ using input pins M or F}$$

or

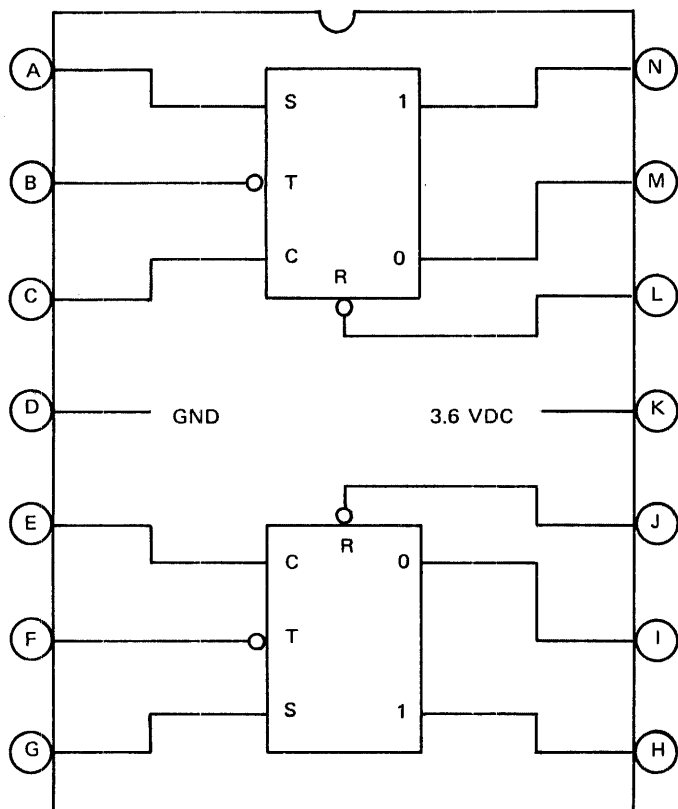
$$T = .8 (R+1k)C \text{ using input pins M or F with R at L or I,}$$

Where T is in seconds

R is in ohms

C is in farads

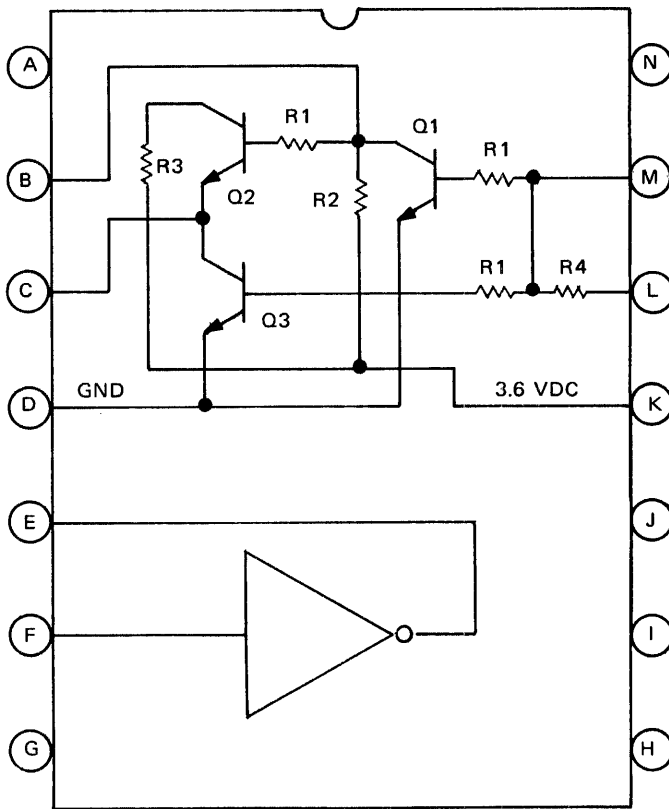
Circuit Detail



DUAL J-K FLIP-FLOP  
1701 6148

| R or P | INPUT |       | OUTPUT     |       |
|--------|-------|-------|------------|-------|
|        | C     | S     | I          | O     |
| GND    | TRUE  | TRUE  | COMPLEMENT |       |
| GND    | FALSE | TRUE  | TRUE       | FALSE |
| GND    | TRUE  | FALSE | FALSE      | TRUE  |
| GND    | FALSE | FALSE | NO CHANGE  |       |

Fig. III-8 DUAL J-K FLIP-FLOP



DUAL BUFFER  
1701 6163

- R1 = 450Ω
- R2 = 640Ω
- R3 = 100Ω
- R4 = 1000Ω

OUTPUTS B AND C MAY NOT BE USED SIMULTANEOUSLY

OUTPUTS J AND E MAY NOT BE USED SIMULTANEOUSLY

Fig. III-9 DUAL BUFFER

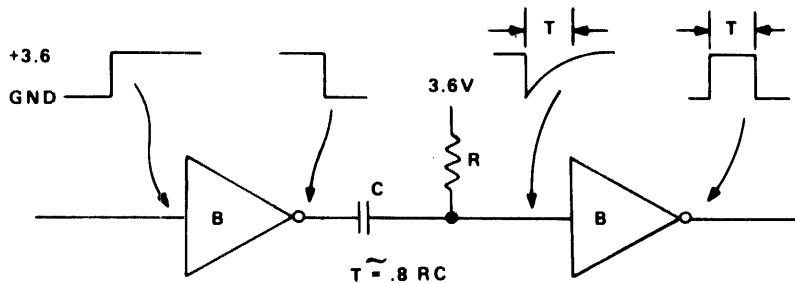


Fig. III-10 DUAL BUFFER AS A SINGLE-SHOT

Circuit Detail

MEMORY SECTION

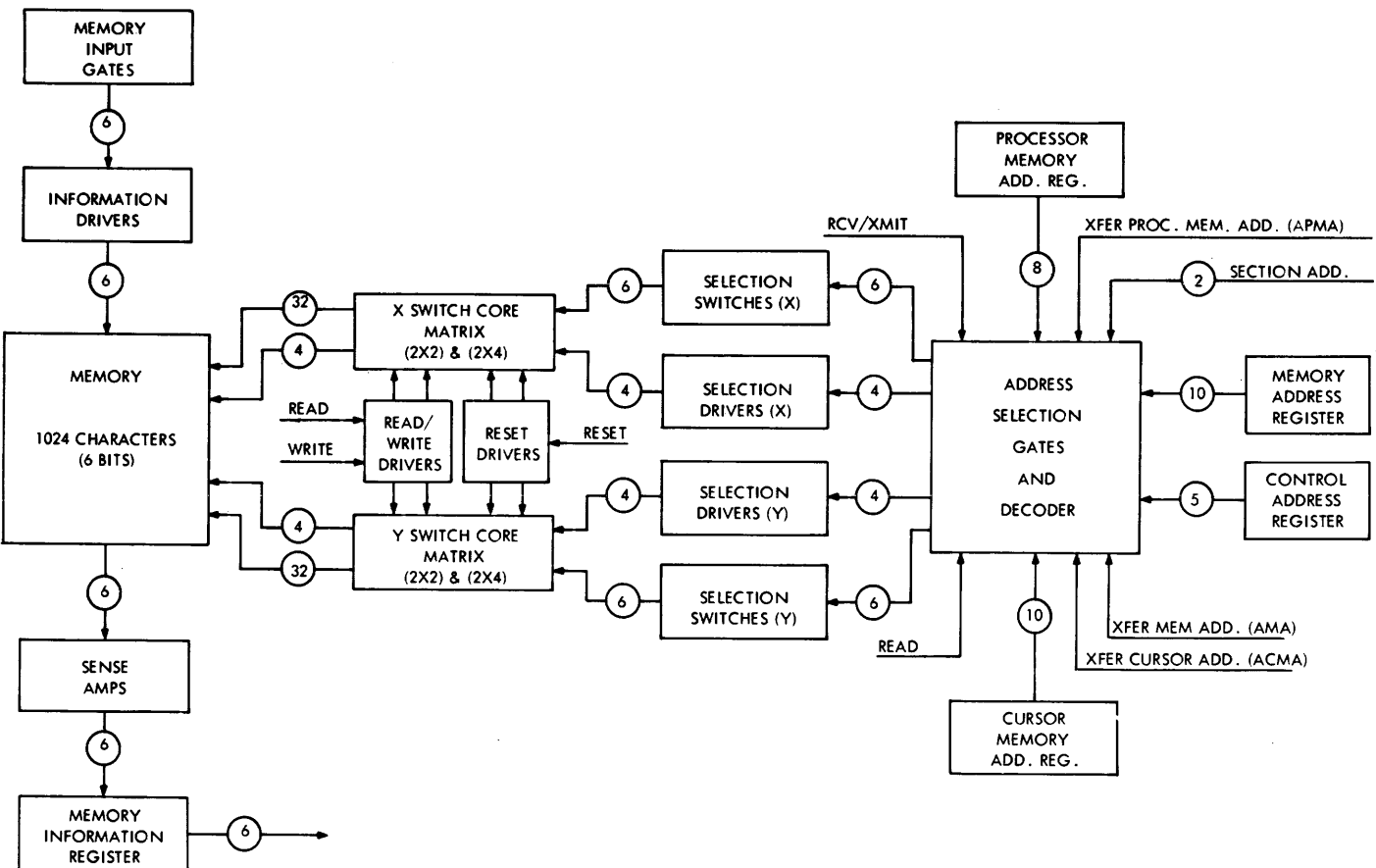


Fig. III-11 BLOCK DIAGRAM - MEMORY SECTION

The memory section of the Input and Display Control Unit consists of the ferrite core assembly (stack), a switch core matrix, selection drivers and switches, switch core reset drivers, information (inhibit) drivers, sense

amplifiers, read and write current regulators, memory address registers, memory information register and memory input gates. Refer to the block diagram, Figure III-11. To select a particular memory location, the desired

### Circuit Detail

address is gated to the address decoder. The output from the decoder enables selection switches and drivers. These switches and drivers then set switch cores in the switch core matrices. The address lines are enabled by the setting of the switch cores. During a read operation, half current will flow in one direction through the address lines. During a write operation, half current will flow in the opposite direction through the address lines. Information to be stored is fed to the information drivers via the memory input gates. Information read from memory, as detected by the sense amplifiers is strobed into the memory information register.

#### FERRITE CORE MEMORY

Ferrite cores, 50 millimeters in diameter, are assembled in six mats of 32 x 32 cores each. The ferrite cores produce a hysteresis loop that is essentially rectangular and are therefore unaffected by small switch currents that may pass through the X and Y lines.

Each core is threaded as shown in Figure III-12. The address lines, X and Y, are threaded at right angles to each other. When both lines have half current passing through them, the core is said to be selected. During the write portion of a memory cycle, an opposing half current may be present in the inhibit line which parallels the X line. This opposing current would then inhibit setting the core to the ONE state, resulting in a ZERO being stored for that bit of the selected address. The sense winding will have an emf induced into it during the read portion of a memory cycle only if the core was previously in the ONE state. When the core is selected, it will be reset to the ZERO state. The reset action then induces the emf into the sense winding. Cores in the ZERO state do not reset, thus an emf is not induced in the sense winding.

#### SWITCH CORE MATRIX

In order to select a given character location in memory, the desired address is decoded to produce signals which in turn will activate the switch core matrix. The switch core matrix then enables one X line and one Y line. The intersection of these two lines is the desired location. Both the X and Y matrices operate in the same manner; therefore only the X matrix will be described.

Consisting of a 4 x 2 and a 2 x 2 matrix, one switch core from each will be enabled by using six switch circuits and four driver circuits. Refer to Figure III-13. One switch and one driver will select a switch core in the 4 x 2 matrix. Another switch and another driver will select a switch core in the 2 x 2 matrix.

When a core is selected, current will flow from the switch through the core to the driver. As the current flows in the primary winding about the core, the core is said to be set to the ONE state. This results in a voltage being induced into the secondary winding or windings. The combined emf generated in the two switch cores becomes the collector voltage for the read current regulator. A constant current then flows through only one of the 32 memory address lines. This half current, in coincidence with the half current from the Y matrix will select the desired character location.

At write time, the switch cores that had been set during read time will be reset to the ZERO state. The resetting action produces an emf of opposite polarity in the secondary windings of the selected memory address lines. This combined emf becomes the collector voltage for the write current regulator.

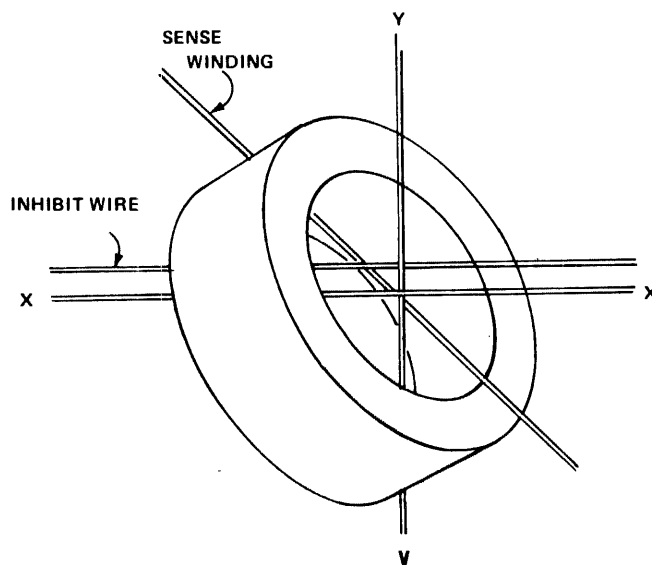


Fig. III-12

Circuit Detail

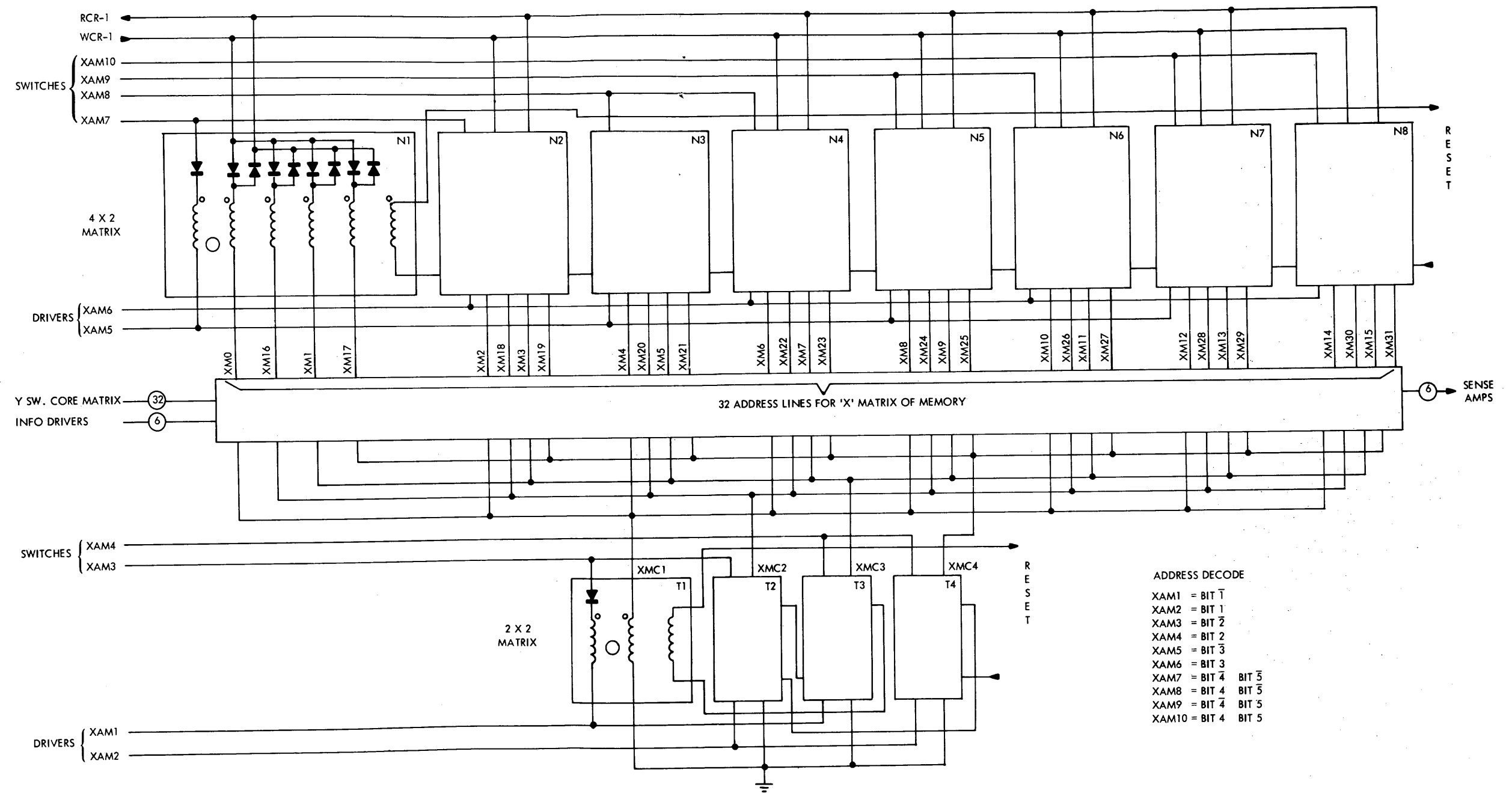


Fig. III-13 'X' SWITCH CORE MATRIX BIDS MEMORY SYSTEM



Circuit Detail

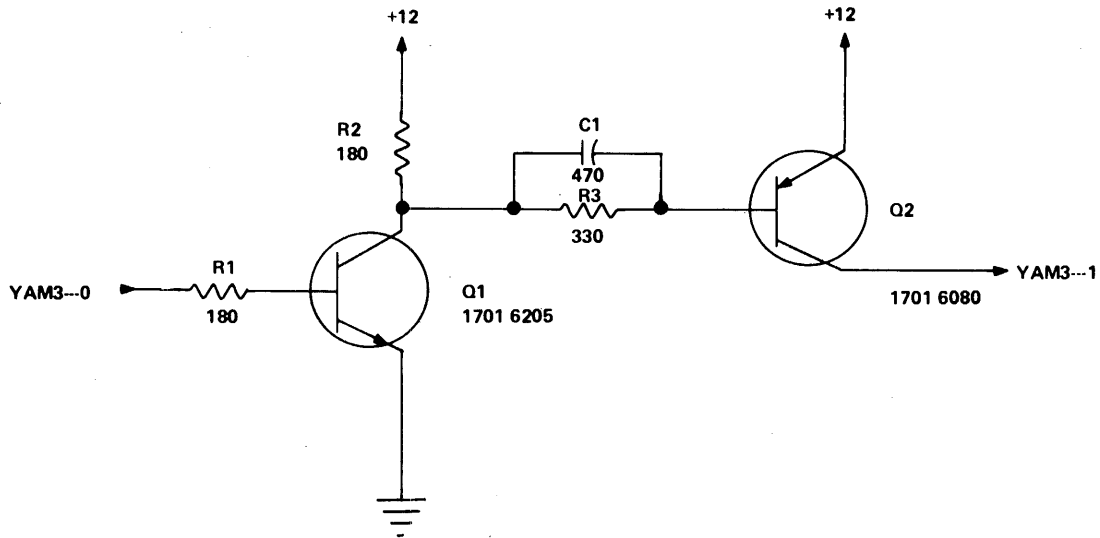


Fig. III-14 SELECTION SWITCH

**SELECTION SWITCH**

There are six selection switches for the X switch core matrix and six for the Y matrix. The memory address decoder provides FALSE signal levels to the selection switches depending on the desired address. Refer to Figure III-14. With a FALSE level at the input, transistor Q1 is turned on, which in turn will allow transistor Q2 to conduct. Conduction will take place when the associated selection driver is turned on.

**SELECTION DRIVER**

There are four selection drivers for the X matrix and four drivers for the Y matrix. Refer to Figure III-15. A FALSE level from the address decoder will turn on Q3 which then turns on emitter-follower Q4. Q5 is then allowed to conduct to saturation. The selection driver and the selection switch are operated at the same time. This then allows current to flow from the +12 volt supply, through the selected switch, through the switch core, to the selected driver and to ground.

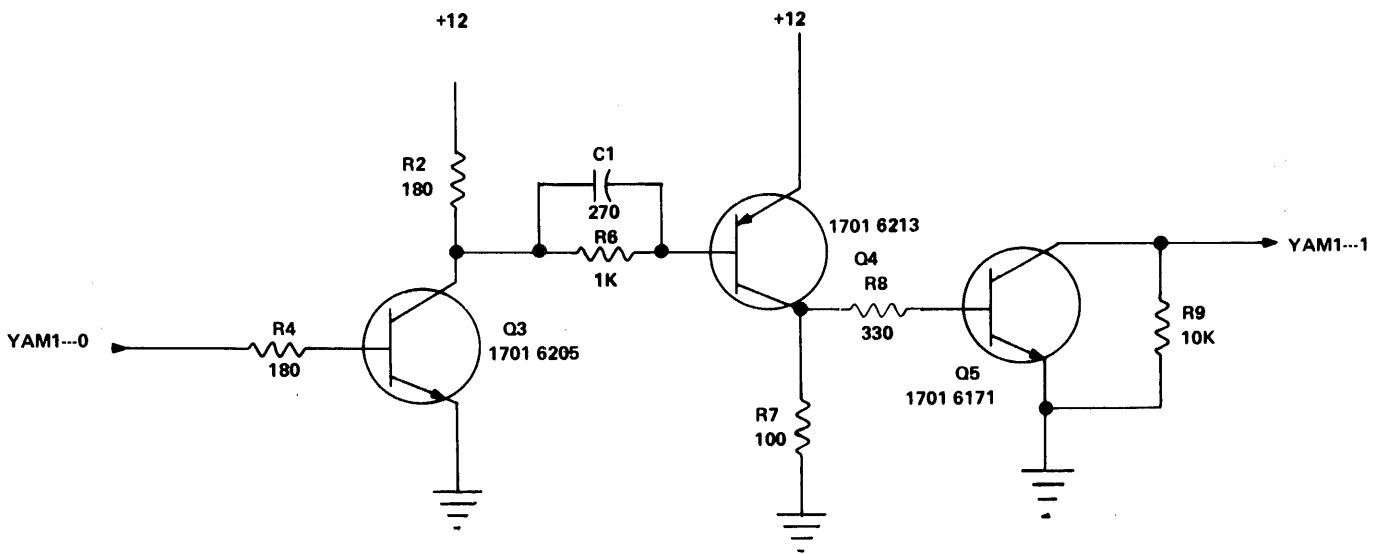


Fig. III-15 SELECTION DRIVER

## Circuit Detail

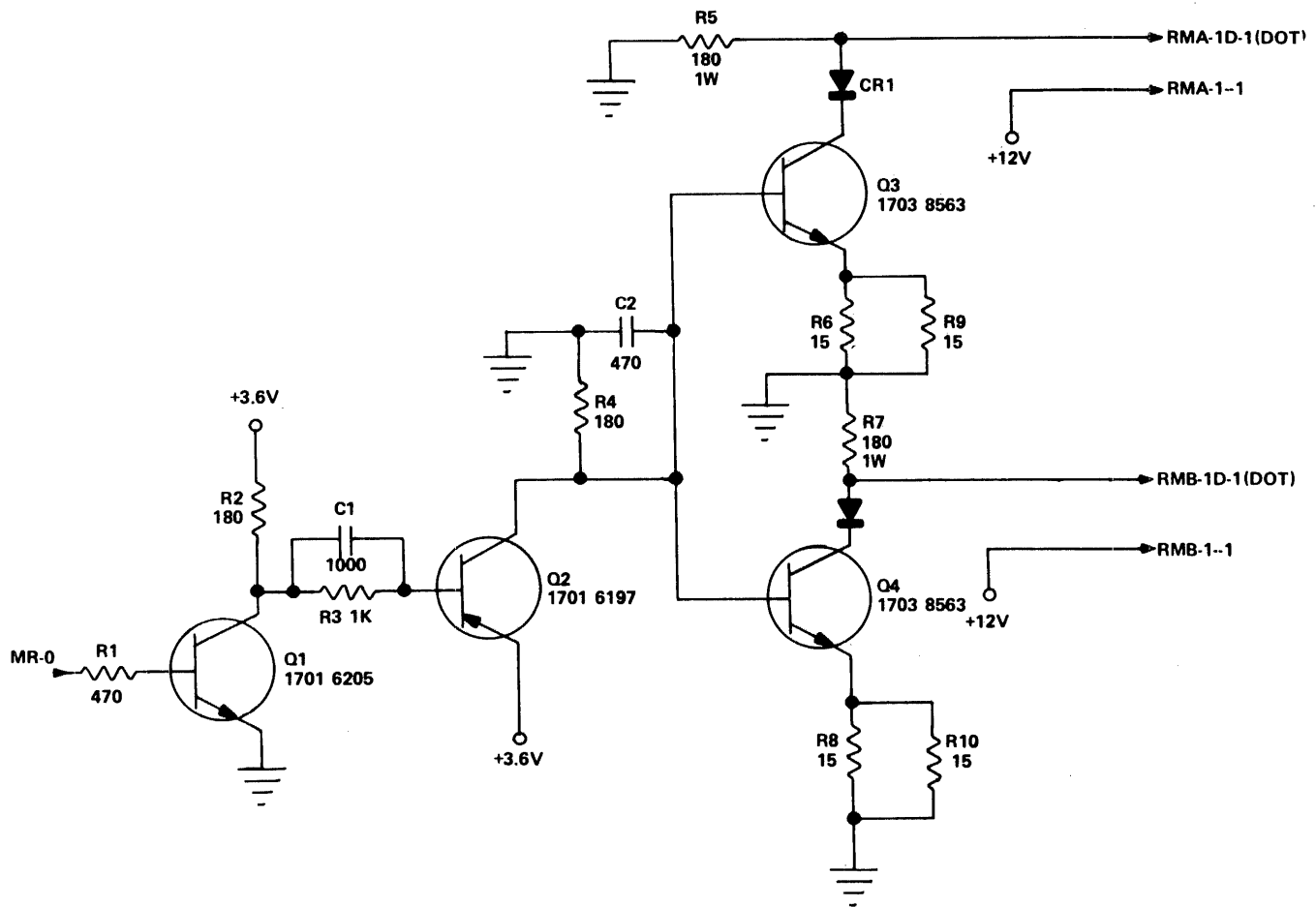


Fig. III-16 SWITCH CORE RESET DRIVER

## SWITCH CORE RESET DRIVER

The reset drivers are used to reset the switch cores to their original ZERO states after having been switched to ONE's during the read cycle. Refer to Figure III-16. When MR----0 goes FALSE for a write cycle, transistor Q1 is turned on, which in turn activates Q2. A potential of approximately +3.6 volts is then applied to the bases of Q3 and Q4. These transistors then conduct to provide the reset current required to reset the 4 x 2 and 2 x 2 switch core matrices.

## READ CURRENT REGULATOR

There are two read current regulators; one for X and one for Y. A read signal is applied to both regulators

simultaneously. Both regulators are identical, therefore only one will be described. Refer to Figure III-17. When RDP-0 is FALSE, Q3 will be turned on causing the collector to go to ground. This in turn causes Q4 to conduct allowing its collector to approach +12 volts. The series diodes CR7, CR8, CR9 and CR10 are used to determine the base voltage of Q5. The voltage developed across the diodes is approximately 2.8 volts. Q5 is then turned on to provide the read current required by the X switch core matrix. Diode CR11 prevents Q5 from saturating. Potentiometer R13 provides adjustment of the read current. Inductor L2 and resistor R15 determine the rise time of the output.

Circuit Detail

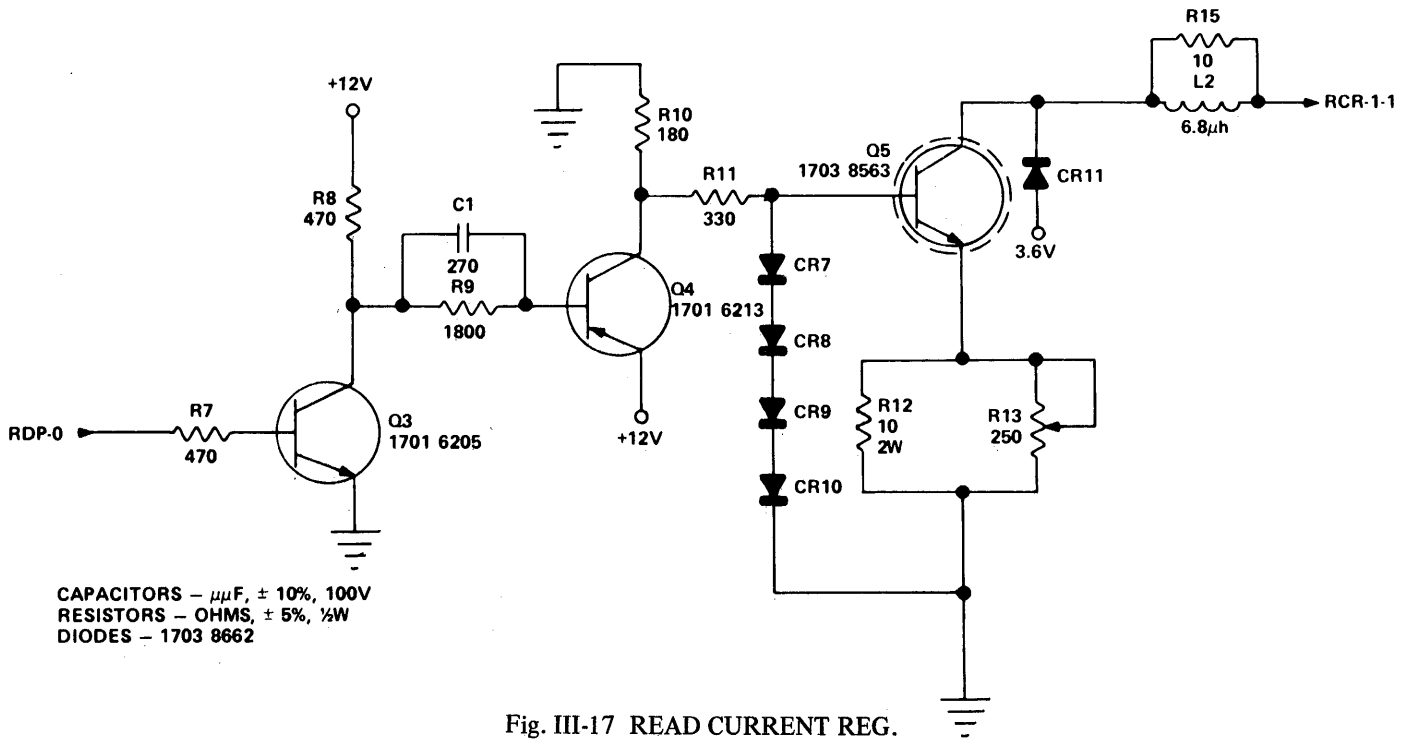


Fig. III-17 READ CURRENT REG.

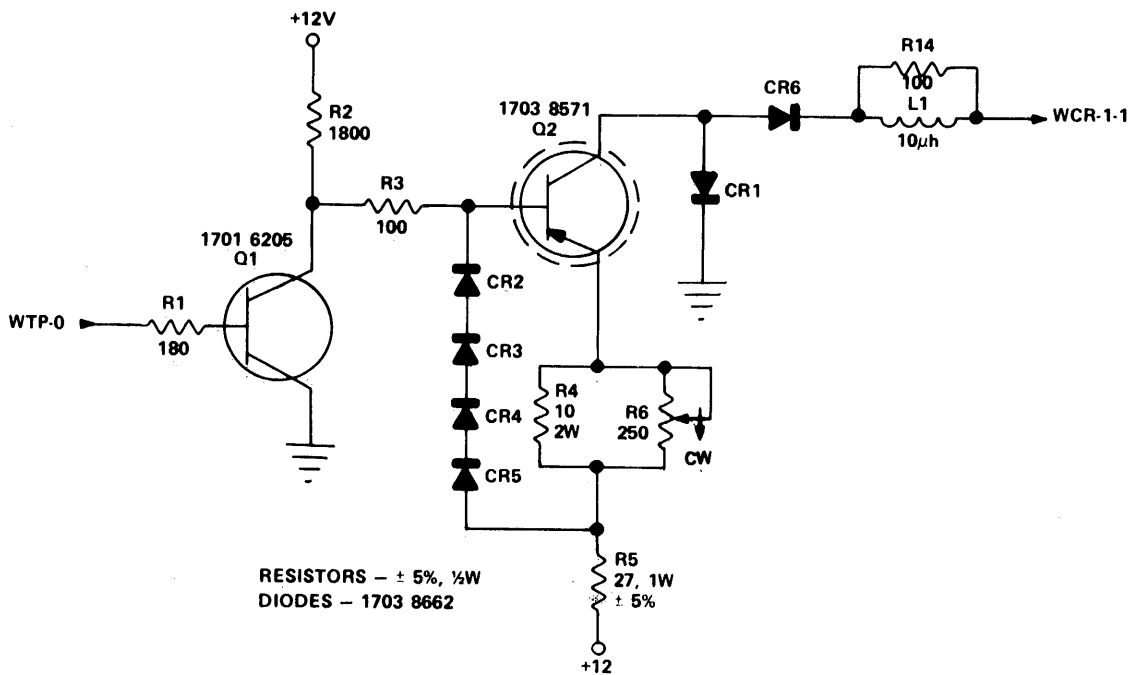


Fig. III-18 WRITE CURRENT REG.

**WRITE CURRENT REGULATOR**

As with the read current regulators, there are also two write current regulators. The write signal is also applied to both regulators simultaneously. Refer to Figure III-18. When WTP-0 is FALSE, Q1 is turned on, allowing the

collector to go to ground. This then permits Q2 to conduct and provides the necessary write current. CR1 prevents saturation. Potentiometer R6 provides the current adjustment. Output signal rise time is determined by the inductance L1 and resistor R14.

Circuit Detail

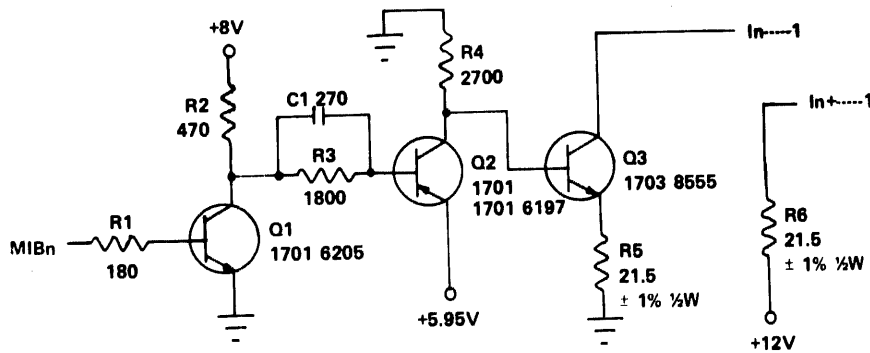


Fig. III-19 INFORMATION DRIVER

**INFORMATION DRIVER**

Data input to the information, or inhibit drivers is from Memory Input Bits 1 through 6. Refer to Figure III-19. The selection of an information driver will cause transistor Q1 to conduct which will then permit Q2 and finally Q3 to conduct. The collector current of transistor Q3 is the inhibit current and is limited by resistor R6. This current is approximately equal to, and opposite in polarity to the one half write current in the ZERO bits of the character being written into memory. The inhibit wire

parallels the X addressing line through the memory core. There are six inhibit lines; one for each bit of the character.

**SENSE AMPLIFIER**

During the read portion of a memory cycle, memory cores that had been set to the ONE's state by a previous write operation are detected by the sense windings. These signals, one for each ONE bit in the character read, are coupled through a sense transformer. Refer to Figure III-20. The center-tapped secondary applies the signal to

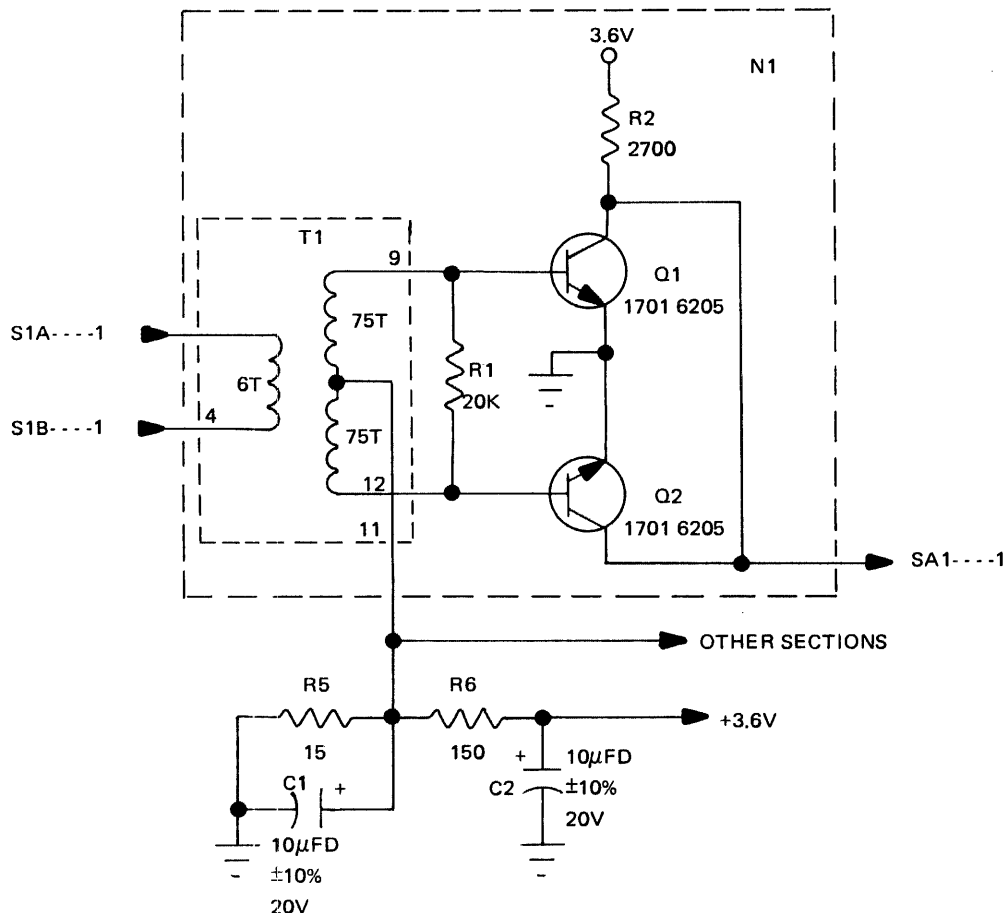


Fig. III-20 TRANSLATOR SENSE AMP.

Circuit Detail

the bases of transistors Q1 and Q2 to provide bi-polar operation. Feeding the center-tap of all six sense amplifiers is a bias voltage developed by the voltage divider network R5 and R6. The output signal is connected directly to the set input of the Memory Information Register Flip-flops. The strobe level, SSA, occurring 600 nanoseconds after the start of the read portion, strobes the information into the flip-flops. There are no adjustments to the sense amplifier.

INFORMATION DRIVER POWER SUPPLY

The information (inhibit) driver requires two voltages that are not available as normal logic voltages. These voltages, +8 and +5.95, are supplied by the information driver power supply. Refer to Figure III-21. Transistor Q5 is a conventional emitter-follower with the base voltage being controlled by a series of ten diodes for temperature compensation. Adjustment of the supply to 5.95 volts is provided by potentiometer R11. The +8 volt supply is obtained from the +12 volt supply through zener diode CR13. Load resistor R15 keeps CR13 conducting when the information drivers are not conducting.

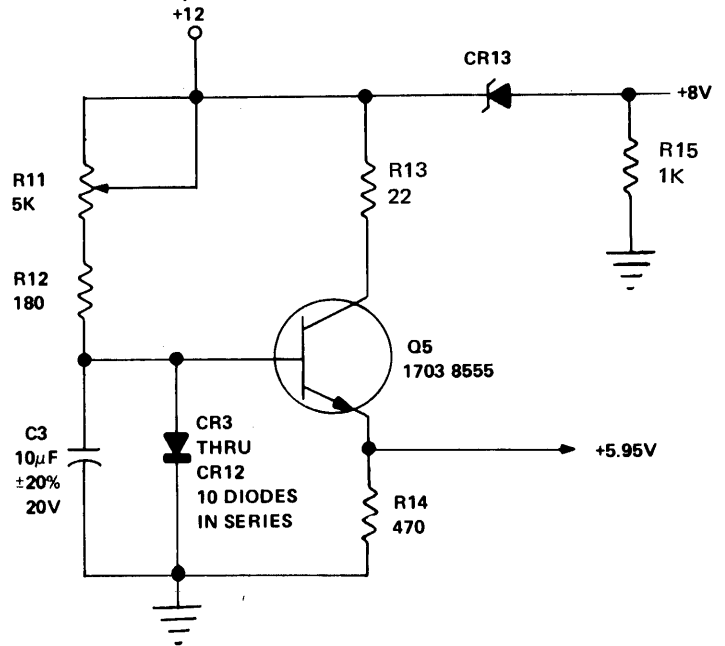


Fig. III-21 INFORMATION DRIVER POWER SUPPLY

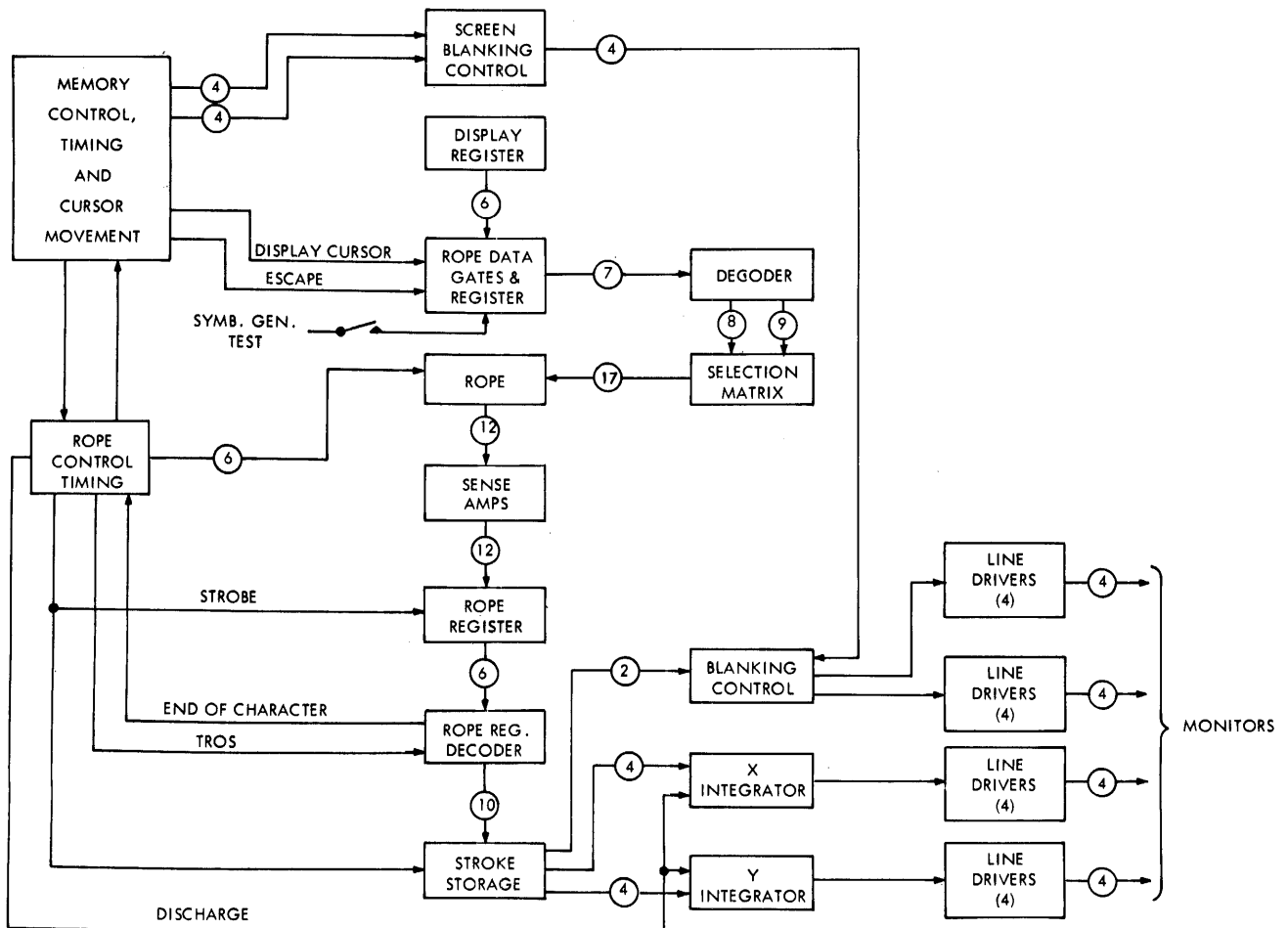


Fig. III-22 SYMBOL GEN. BLOCK

Circuit Detail

SYMBOL GENERATOR SECTION

A simplified block diagram of the symbol generator section is shown in Figure III-22. The symbol generator consists of the memory information register (as input), translator (Rope) address register, a decoder and selection matrix, the core translator or Rope, sense amplifiers, a twelve bit register to store the stroke information, a six bit decoder, a ten bit stroke storage register, a blanking control circuit and two bi-directional integrators.

The operation required to produce a character on the CRT begins with a display memory cycle. At the completion of the memory cycle, the character to be displayed is present in the memory information register. Assuming that this is not an escape character, nor is the cursor to be displayed, the display information is applied to the decoder network. Output from the decoder will be a FALSE level from one of eight gates labeled DRn-0 (where n = 0 thru 7) and a FALSE level from one of nine gates labeled RCn-0 (where n = 0 thru 8). These two levels are applied to a corresponding driver (Figure III-23) and receiver (Figure

III-24) which, in turn, feed the selection matrix. The selection matrix, an eight by nine diode array, will then enable one of the possible 72 lines in the core translator. Cores in the translator are then set according to the stroke pattern for the character to be displayed. At the end of the set cycle, the sense amplifier information is strobed into the 12 bit Rope register. The information is then decoded, 6 bits at a time, to produce 10 bits of stroke data. This data is stored in the ten bit stroke storage register. Two bits of this register store blanking information, which is then gated with screen control and determines whether the stroke about to be made is of single, double or blank intensity. The remaining 8 bits contain the directional information for the stroke. Four bits for vertical movement and four bits for horizontal movement. Combinations of the two will produce angular stroke information. This data is applied to the bi-directional integrators. The output from the integrators is then fed to the line drivers which feed the monitor deflection amplifiers. Intensity or blanking information is also fed to line drivers which feed the monitor 'Z' amplifier.

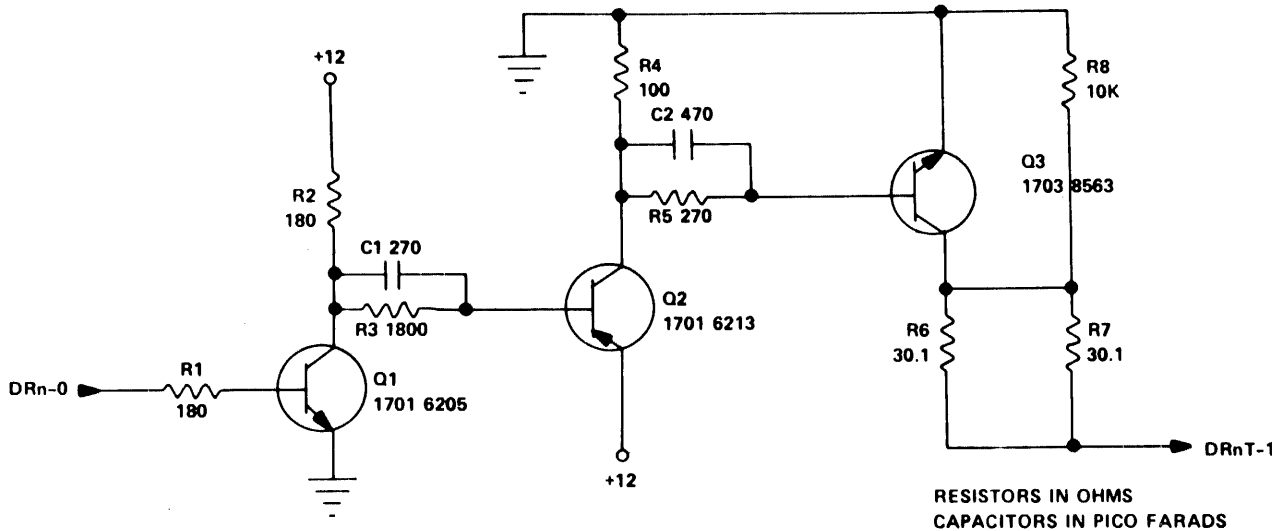


Fig. III-23 TRANSLATOR DRIVER

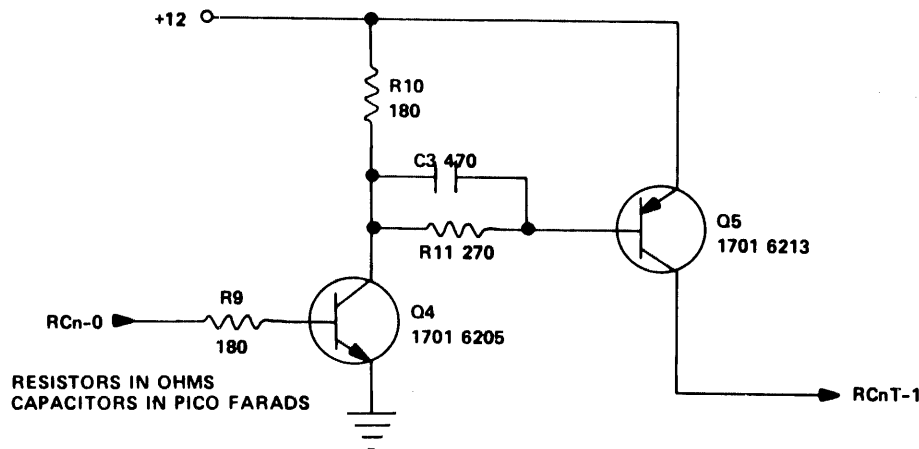


Fig. III-24 TRANSLATOR RECEIVER

## Circuit Detail

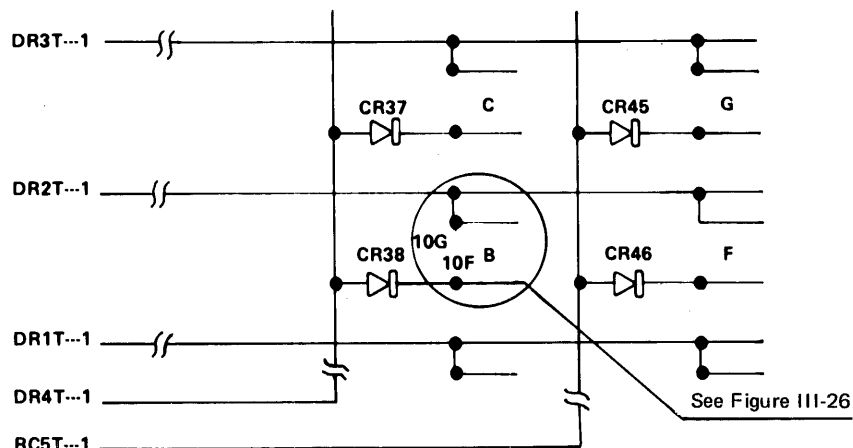


Fig. III-25 A PORTION OF THE ROPE SELECTION MATRIX

## DECODER AND SELECTION MATRIX

Information to be decoded is received from the Rope address register. Gated into the decoder are levels to indicate a test mode (TSG-0), display the cursor (DCUR-1) decode the symbol (SYMD-0) and the fly-back (FLB-0) level.

The test symbol generator level, when FALSE, will force the New Line (NL) character to be addressed in the Rope. This is a maintenance aid to allow the field engineer to adjust the bi-directional integrators, which will be discussed later.

During each display cycle, there will come the time to display the cursor. When this occurs, the level DCUR-1 will be TRUE. This causes the cursor to be addressed prior to addressing the character specified by the memory information register. Following the action of displaying the cursor, another display memory cycle occurs, returning the character to the memory information register which will then be addressed and displayed.

The decoding network enables one of eight drivers and one of nine receivers. The drivers and receivers then become inputs to the selection matrix. The selection matrix consists of 72 diodes arranged in an 8 x 9 array. Refer to Figure III-25. With driver DR2T---1 enabled and receiver RC4T---1 enabled, current will flow from the driver, through the set line connected to pins 10G and 10F, through CR38 and back to the receiver. This set line will then set the tape-wound cores necessary to generate the strokes for the character 'B'.

When the character to be displayed is an escape characters (←, ≠, ], , x, ≤, \$, %, \$) which require two character positions in memory, a flip-flop will be set to remember the fact that an escape character is to be displayed. After the first of the two characters is read from memory, it is decoded from the MIR, the flip-flop is set and a second memory cycle initiated. When the second character is

decoded, it is combined with the previously set flip-flop to enable the RC8T level that is used for selecting the eight escape characters. At the same time, one of the eight drivers will be enabled to select the desired escape character.

## ROPE

Consisting of 72 tape-wound cores arranged in a 12 x 6 matrix, mounted on a single plane, the core translator, or Rope, is used as a read only memory capable of presenting to the Rope register, the data required to generate the 72 characters. Passing through the tape-wound cores are set lines, inhibit lines and sense lines. For each of the 72 characters there is a set line. Each set line will pass only through those cores that are required for a given character. Because tape-wound cores are relatively slow in operation, information for two strokes is set at any given time. This set current is in conjunction with an inhibit current passing through cores designated for other strokes. There are six inhibit lines used in the core translator. Not all characters require twelve strokes. Characters requiring nine or less strokes are ended by using an End-of-Character decode of the information following the final stroke of the character. This action terminates the display cycle for this character. Therefore, inhibit current may or may not pass through all six inhibit lines, depending on the number of strokes required for the character. As mentioned earlier, two strokes are set at a time. Each stroke requires six bits of information, therefore 12 sense amplifiers are required to sense the 12 bits for the two strokes.

Figure III-26 illustrates the manner in which the cores are wound for the character 'B'. The squares represent the cores. The squares containing an X have the set wire passing through the corresponding core. Squares not containing an X have the set wire bypassing the corresponding core. Stroke one information is contained in row one, columns one through six. Stroke two information is contained in

Circuit Detail

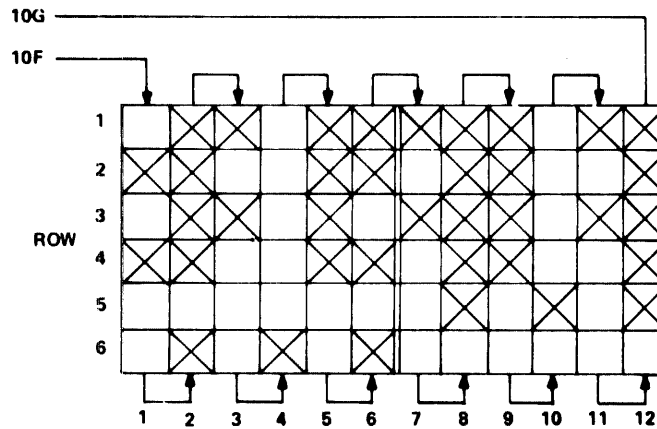


Fig. III-26 SET WIRE WINDING FOR CHARACTER 'B'

row one, columns seven through twelve. Row six, columns one through six then contains stroke eleven information. Inhibit wires are wound such that inhibit wire one passes through all cores except those in row one. Inhibit wire two passes through all cores except those in row two. The remaining inhibit wires are wired in a like manner. Sense windings are wired by the column such that sense winding for bit one represents the cores in column one. Bit twelve

then represents the cores in column twelve.

Tape-wound cores operate much like a transformer in that the set current, when not opposed by the inhibit current, will induce an emf into the sense windings. At the proper time, 1.7 microseconds after the start of the set current, a strobe pulse, 100 nanoseconds in duration, occurs which will strobe the sense amplifier output into the Rope register. Refer to the timing diagram in Figure III-27.

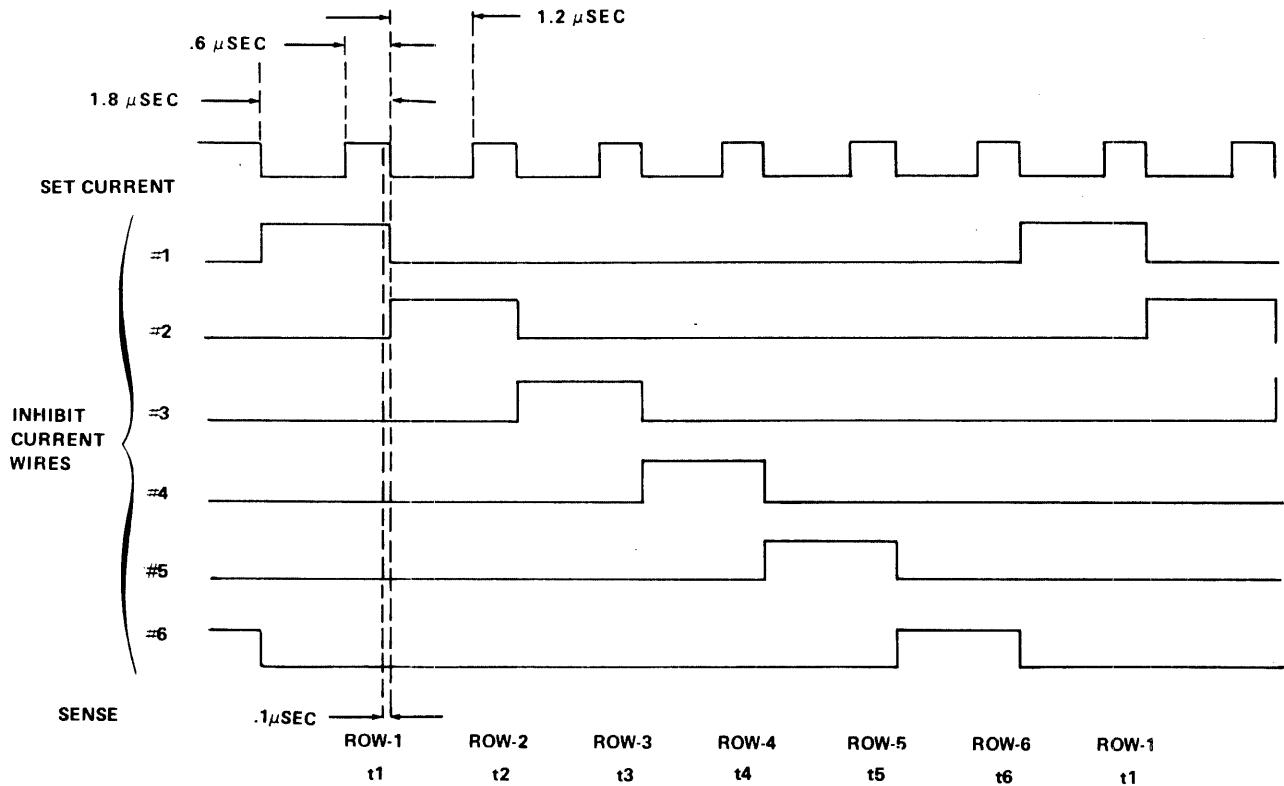


Fig. III-27



Circuit Detail

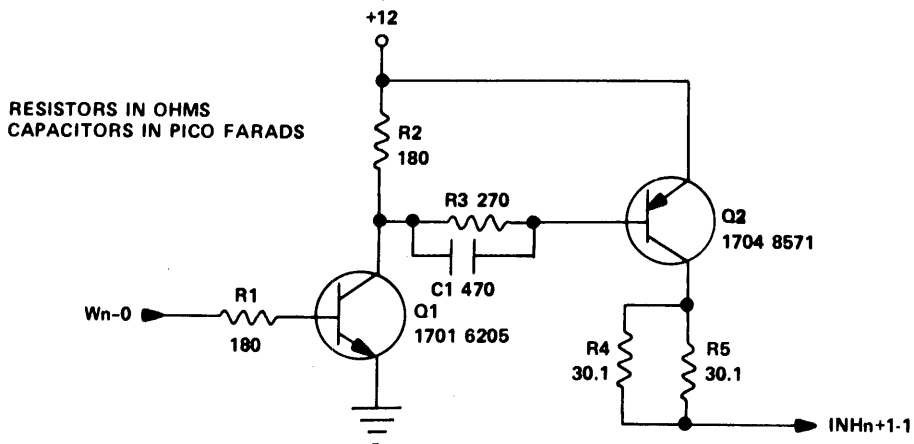


Fig. III-28 TRANSLATOR INHIBIT DRIVER

TRANSLATOR INHIBIT DRIVER

In order to set cores in the translator two strokes at a time, six driver circuits are used to inhibit the setting of the undesired cores. Refer to Figure III-28. When the input to the driver,  $W_n-0$ , is FALSE, Q1 will be turned on allowing the collector to go to ground. This action causes Q2 to conduct through the corresponding inhibit line in the Rope. This current is in opposition to the set current, therefore inhibiting the set of those particular cores. At any one 'set' time, one of the six inhibit drivers will be enabled.

TRANSLATOR SENSE AMPLIFIER

Each of the twelve sense amplifiers consists of a sense transformer and two transistor amplifiers. Refer to Figure III-29. The sense signal is applied over twisted-pair lines to the primary winding of the sense transformer. The center-tapped secondary applies the signal to the bases of transistors Q1 and Q2 to provide bi-polar operation. The

collectors are tied together to form the output level. The output levels are designated AOD through FOD for ODD stroke information and AEV through FEV for EVEN stroke information. These levels are applied to the set input of the Rope storage register and are clocked with SRSA (Stroke Rope Sense Amp.) level.

ROPE STORAGE REGISTER AND DECODER

The twelve bits of information from the translator sense amplifier are set into the Rope storage register. The odd stroke information from the register is gated with TROS during odd stroke time, while the even stroke output from the register is gated with TRES during even stroke time. During each stroke time, six bits are decoded and used to generate set levels for the stroke storage register. Also decoded with this logic is the End-of-Character decode for characters that require nine or less strokes.

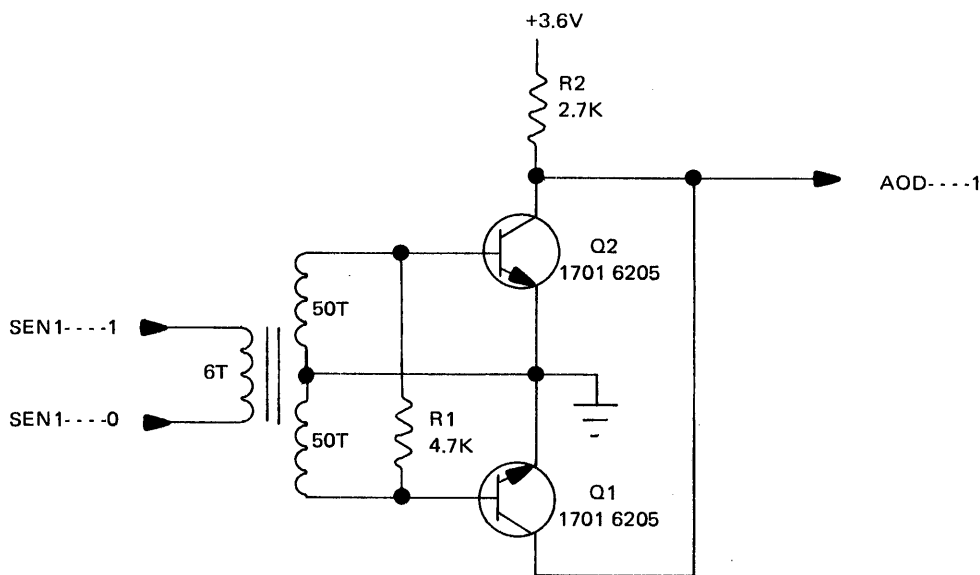


Fig. III-29 TRANSLATOR SENSE AMP.

Circuit Detail

|    | 5 | 4 | 3 | 2 | 1 | X         | Y  |
|----|---|---|---|---|---|-----------|----|
| 0  | 0 | 0 | 0 | 0 | 0 | NO MOTION |    |
| 1  | 0 | 0 | 0 | 0 | 1 | -1        | -1 |
| 2  | 0 | 0 | 0 | 1 | 0 |           | -1 |
| 3  | 0 | 0 | 0 | 1 | 1 | -2        | -1 |
| 4  | 0 | 0 | 1 | 0 | 0 | -1        |    |
| 5  | 0 | 0 | 1 | 0 | 1 | +1        | -1 |
| 6  | 0 | 0 | 1 | 1 | 0 | -2        |    |
| 7  | 0 | 0 | 1 | 1 | 1 | +2        | -1 |
| 8  | 0 | 1 | 0 | 0 | 0 | NOT USED  |    |
| 9  | 0 | 1 | 0 | 0 | 1 | -1        | -2 |
| 10 | 0 | 1 | 0 | 1 | 0 |           | -2 |
| 11 | 0 | 1 | 0 | 1 | 1 | -2        | -2 |
| 12 | 0 | 1 | 1 | 0 | 0 | NOT USED  |    |
| 13 | 0 | 1 | 1 | 0 | 1 | +1        | -2 |
| 14 | 0 | 1 | 1 | 1 | 0 | NOT USED  |    |
| 15 | 0 | 1 | 1 | 1 | 1 | +2        | -2 |

|    | 5 | 4 | 3 | 2 | 1 | X            | Y  |
|----|---|---|---|---|---|--------------|----|
| 16 | 1 | 0 | 0 | 0 | 0 | END OF CHAR. |    |
| 17 | 1 | 0 | 0 | 0 | 1 | -1           | +1 |
| 18 | 1 | 0 | 0 | 1 | 0 |              | +1 |
| 19 | 1 | 0 | 0 | 1 | 1 | -2           | +1 |
| 20 | 1 | 0 | 1 | 0 | 0 | +1           |    |
| 21 | 1 | 0 | 1 | 0 | 1 | +1           | +1 |
| 22 | 1 | 0 | 1 | 1 | 0 | +2           |    |
| 23 | 1 | 0 | 1 | 1 | 1 | +2           | +1 |
| 24 | 1 | 1 | 0 | 0 | 0 | NOT USED     |    |
| 25 | 1 | 1 | 0 | 0 | 1 | -1           | +2 |
| 26 | 1 | 1 | 0 | 1 | 0 |              | +2 |
| 27 | 1 | 1 | 0 | 1 | 1 | -2           | +2 |
| 28 | 1 | 1 | 1 | 0 | 0 | NOT USED     |    |
| 29 | 1 | 1 | 1 | 0 | 1 | +1           | +2 |
| 30 | 1 | 1 | 1 | 1 | 0 | NOT USED     |    |
| 31 | 1 | 1 | 1 | 1 | 1 | +2           | +2 |

Table III-30 ROPE STORAGE DECODE

STROKE STORAGE AND BLANKING CONTROL

Consisting of ten flip-flops, the stroke storage register receives set levels from the Rope storage decoder and is clocked with the level TSD (Transfer Storage Decode). Stored in this register is the directional and blanking information for one stroke of the character to be displayed. Directional information is then fed to the bi-directional integrators. Blanking information is gated with display time controls to allow only the selected screen to display the

stroke. It should be noted that while the directional information is fed to all screens simultaneously, only one screen will actually display the stroke.

Figure III-30 tabulates the directional information as it is decoded. Figure III-31 illustrates the twenty-four (24) possible strokes that can be made from any one point. Figure III-32 illustrates the stroke pattern for the character 'B'. All characters start at the lower left corner of the character position on the display line.

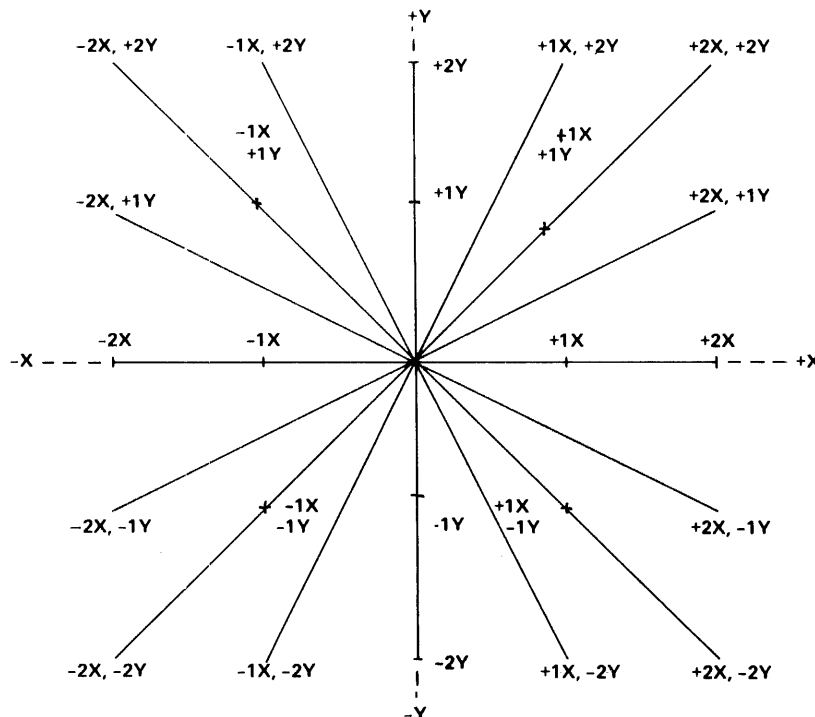


Fig. III-31 SYMBOL GENERATOR STROKE SEGMENTS

## Circuit Detail

The information required for the first stroke of the character 'B' can be obtained from Figure III-26. Of the first five bits, bits 2, 3, and 5 are set. Applying this information to the table, it can be seen that a double horizontal stroke to the right will result. With bit 6 set, the stroke will not be blanked. The second stroke can now be drawn. For this stroke, bits 1, 2, 3 and 5 are set. From the table, a double horizontal with a single vertical stroke is found. This results in an angular stroke being produced. Again, bit 6 is set, allowing the stroke to be displayed. Note that when stroke 5 is drawn, (information from row 3, columns 1 to 6) bit 6 is not set. This causes the stroke to be blanked and not displayed. Note also that stroke 9 is not a stroke at all, there being no bits set. This is done to ensure a sharp corner for the character.

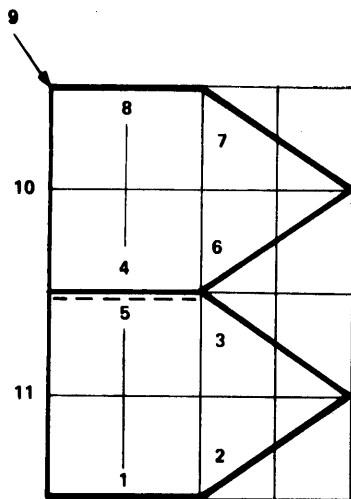


Fig. III-32 STROKE PATTERN FOR CHARACTER 'B'

## BI-DIRECTIONAL INTEGRATOR

The bi-directional integrators accept stroke information from the stroke storage register. An additional input, DISC, is used at the completion of a character to discharge the integrating capacitor, C3. Refer to Figure III-33. At any one stroke time, only one of the four directional signals will be enabled. The amplifier circuit for a single, positive stroke, Q1 and Q5, will produce a voltage of about +2.5 volts to charge C3. A double, positive stroke is amplified by Q2 and Q6 and will produce a charging voltage that can be adjusted to twice that of the single stroke. Q3 and Q7 are used to amplify the negative, single stroke. This charging voltage is also adjustable and is set equal to the positive, single stroke. Q4 and Q8 amplify the negative, double stroke, and this too is adjustable. Capacitor C3 is then charged by the positive stroke information and discharged by the negative stroke information. The resulting signal is fed to the 'Darlington' pair of transistors Q9/Q10. Output from this circuit is then fed to the driver transistors Q13, Q14, Q15, and Q16.

At the completion of a character, the charge on C3 may not be reduced to zero, therefore, a discharge circuit is provided to ensure that the next character to be displayed starts at the proper point. This is accomplished at 'End-of-Character' time, which will cause the level DISC--1 to be generated. When this level goes TRUE at the integrator card input, Q11 will be turned on which in turn will allow Q12 to conduct. This action will discharge C3 and bias it to +3.6 volts less the drop across the transistor.

Figure III-34 illustrates the waveshapes that would be generated for both the X and Y symbol data for the asterisk. Also shown are the intensity signals. Examination of the X axis waveshape reveals that at the end of the twelfth stroke, the output is not at zero. At 'End-of-Character' time, DISC--1 will be generated, which will return the X axis to the reference point. The discharge level is also fed to the Y axis integrator but has no effect since the Y axis is at the reference point.

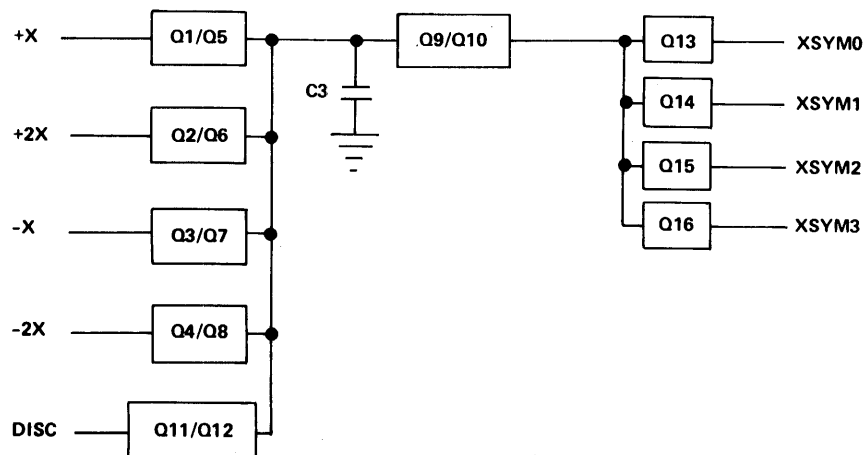


Fig. III-33 BI-DIRECTIONAL INTEGRATOR BLOCK DIAGRAM

Circuit Detail

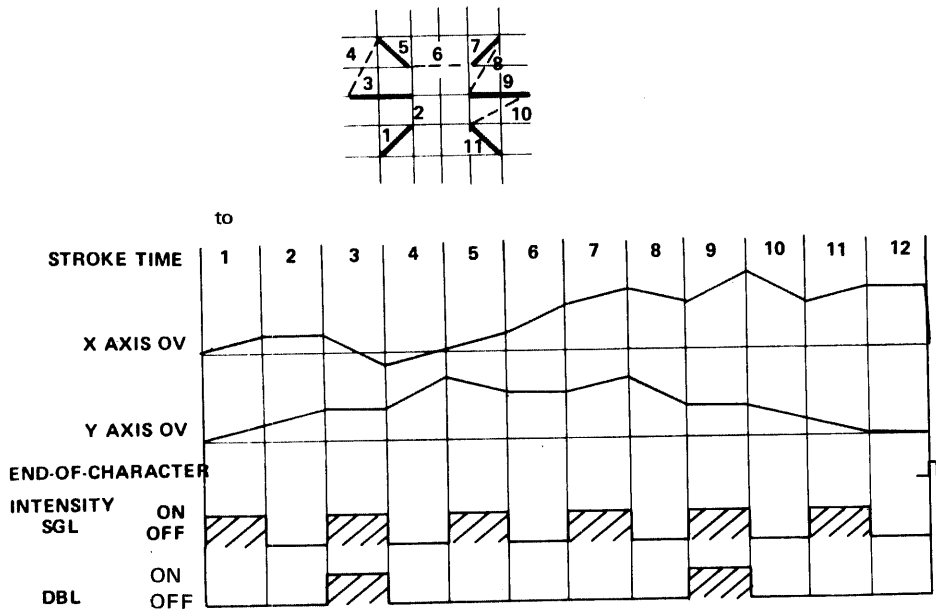


Fig. III-34 CONSTRUCTION OF AN ASTERISK

DIGITAL-TO-ANALOG SECTION

While the symbol generator circuit produces the desired symbols, the positioning of these symbols on the CRT is controlled by the X and Y Digital-to-Analog converters. The analog outputs are applied to the summing junction of the X and Y deflection amplifiers in the Monitor.

There are a total of 2000 available coordinate positions on the CRT. Each of the 25 lines contains a maximum of 80 characters. Display information starts in the upper left corner of the CRT and proceeds to the right

sequentially. When the line is filled, the next character is positioned to the first location of the next line down.

Twelve bits of information control the positioning of the CRT beam. Seven of these bits originate at the horizontal position counter and five at the vertical position counter. This information is provided as input to the X and Y Digital-to-Analog converters, respectively. These circuits convert the digital inputs to analog positioning voltages. The analog outputs are step functions having amplitudes which are proportional to the binary weights of the input levels. Refer to Figure III-35 for typical examples of the X and Y Digital-to-Analog output waveforms.

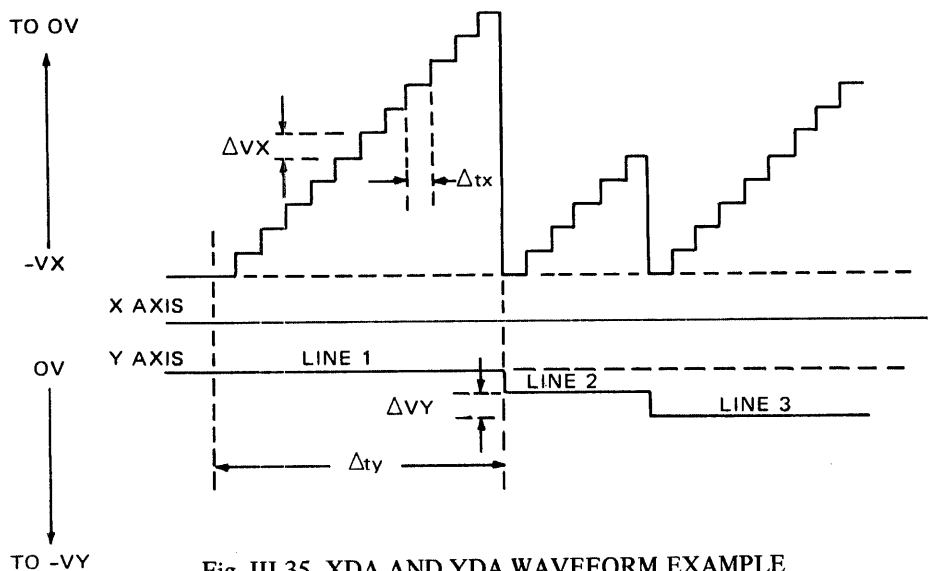


Fig. III-35 XDA AND YDA WAVEFORM EXAMPLE

Circuit Detail

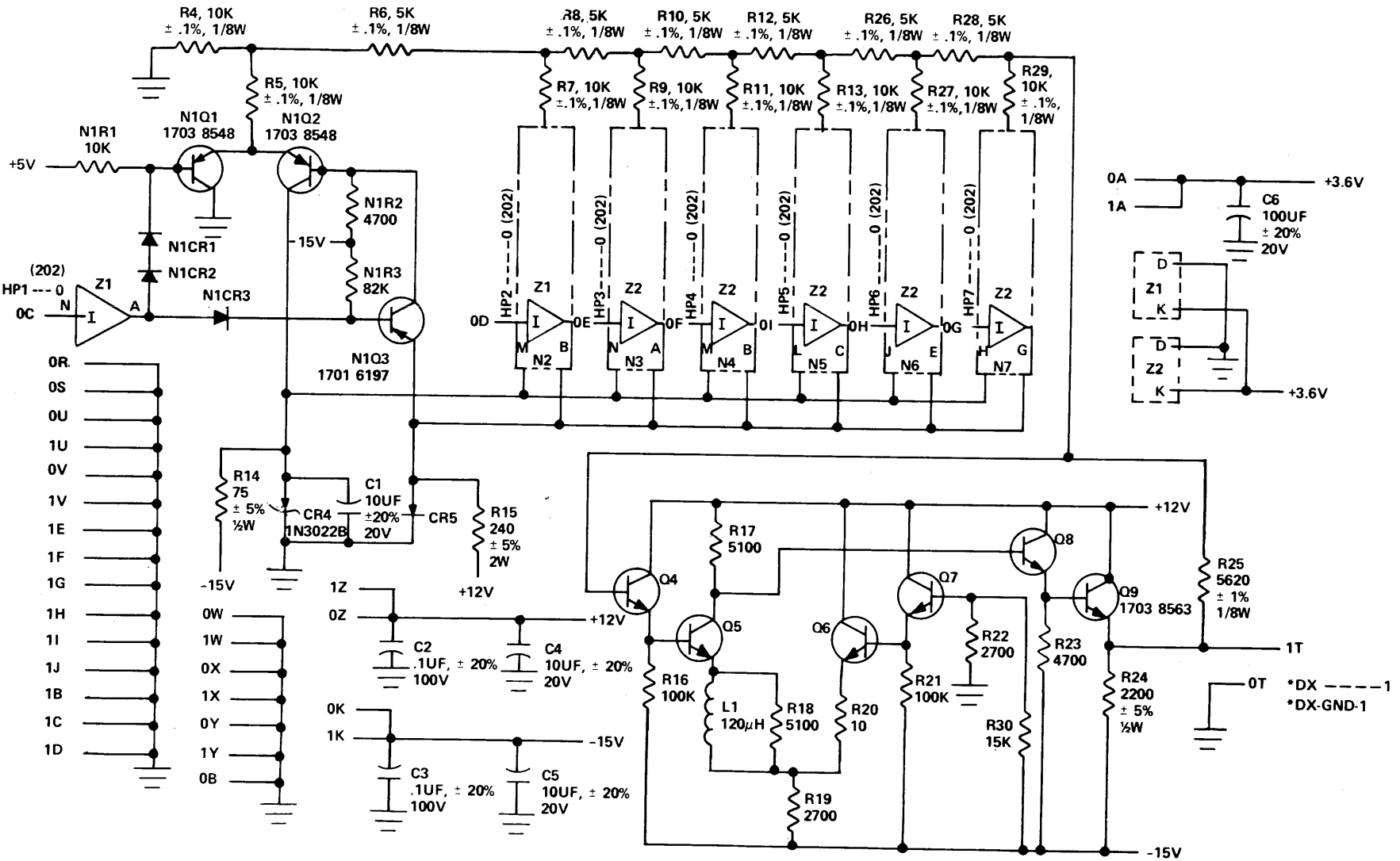


Fig. III-36 X AXIS DIGITAL-TO-ANALOG CIRCUIT SCHEMATIC

## Circuit Detail

The X axis waveform is a positive going staircase function. A seven bit input of 1111111 will produce voltage  $-v_x$  at the output of the XDA converter and positions the CRT beam to the left side of the display screen. As the input is down-counted, the XDA output is incremented in sequential steps. The voltage change per step ( $\Delta V_X$ ) is proportional to the horizontal distance between character positions on the screen. The time plateau ( $\Delta t_x$ ) at each step is the time during which the character to be displayed is drawn on the CRT. This time will vary, depending on the number of strokes required for the character.

The Y axis waveform is a negative going staircase function. A five bit input of 00000 will produce 0 volts at the output of the YDA converter and positions the CRT beam to the top line of the display screen. As the input is up-counted, the YDA output is decremented in sequential steps. The voltage change per step ( $\Delta V_Y$ ) is proportional to the vertical distance between display lines. The time plateau ( $\Delta t_y$ ) at each step is the time required to draw the specified number of symbols along the respective display line.

The two converters are essentially identical, the XDA having two additional voltage switch circuits. Reference voltage for the XDA is supplied by the zener diode CR4 on the XDA card at location 301. Zener diode CR4 on YDA card at location 306 provides the reference voltage for the YDA.

Following is the description of the CDA circuit. Refer to Figure III-36. The ladder network consists of resistors R4 through R13 and R26 through R29. Each ladder branch resistor, the odd numbered resistors, is connected to a voltage switch. The junction of the emitters of transistors Q1 and Q2 reflect the status of the voltage switch. When Q1 is conducting, the switch is at ground; when Q2 is conducting, the switch output is close to the reference voltage.

When the input level at pin OC is at ground (TRUE), pin A of the inverter will be positive. This level is then applied to the base of Q1 through the level shifting diodes CR1 and CR2. Bias current for these diodes is provided by resistor R1. Transistor Q1 is operated in an inverse beta mode with the collector at ground, therefore is cut off. The positive output of the inverter is also applied to the base of Q3 through level shifting diode CR3. Resistor R3 supplies bias current for CR3. Transistor Q3 is reversed biased and therefore cut off. The collector of Q3 then rises toward -15 volts through resistor R2, but is clamped at the reference voltage when Q2 becomes forward biased. Transistor Q2 will then saturate, and its emitter voltage will come within .2 volts of the negative reference voltage.

A FALSE level at the input pin will cause pin A of the inverter to approach ground. This level, when applied to the base of Q1 will cause conduction, providing a switch output of ground. Transistor Q3 will also conduct causing Q2 to be reversed biased, thereby cutting off Q2.

The inputs to the XDA from the horizontal position counter are weighted in binary fashion such that bit 1 (HP1) has a binary weight of 1, while bit 7 (HP7) has a binary weight of 64. The voltage switch controlled by bit 1 will produce the least amount of change, namely 1/128 of the reference voltage. Bit 7 will produce the greatest amount of change, 1/2 of the reference voltage. Refer to Figure III-37 for an equivalent ladder network. As each switch is operated, a proportionate change will result at the output.

The combined output of all the switches is then fed to the differential amplifier circuit consisting of transistors Q4, Q5, Q6 and Q7. Emitter follower circuit Q8 and Q9 provide drive and isolation to the Line Driver cards for each Monitor.

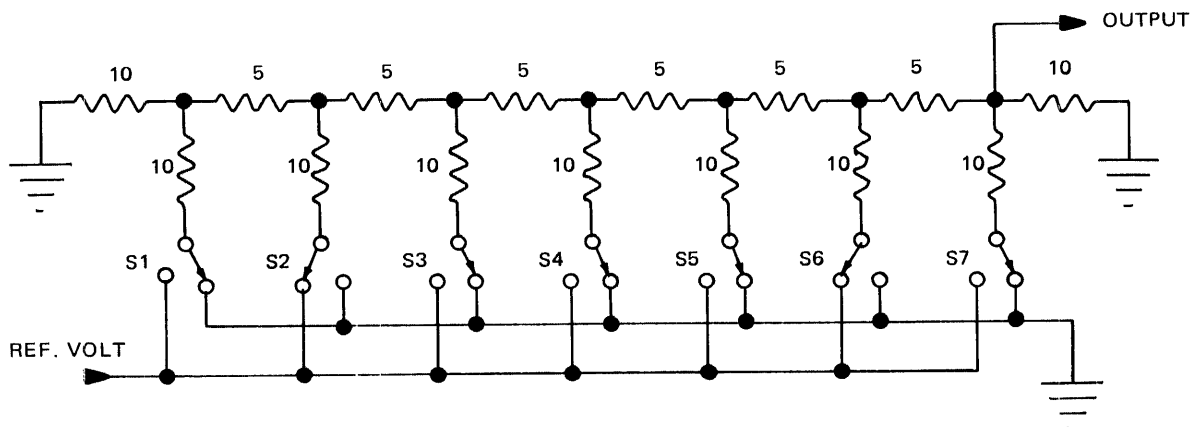


Fig. III-37 EQUIVALENT LADDER NETWORK

Circuit DetailDEFLECTION AMPLIFIER

The X and Y deflection amplifiers receive the respective Digital/Analog converter and symbol excursion inputs and convert them to the current outputs required to drive the horizontal and vertical coils of the CRT deflection yoke. A simplified diagram is illustrated in Figure III-38. The circuit is essentially a closed loop operational amplifier which derives its degenerative feedback by sampling the current in the output branch. The closed loop input voltage to output current ratio is nominally 1:1; i.e., a change of one volt at the input of the amplifier causes a corresponding change of one ampere in the deflection yoke current. The gain of the amplifier is adjusted by modifying the feedback impedance.

The D/A output step function and the symbol generator output waveform are algebraically added with the feedback and centering voltages at the input summing junction. The centering input is adjustable, positive DC voltage that defines the quiescent beam position on the CRT. The feedback input provides controlled degeneration, thereby providing closed loop stability. The summed input voltage is applied to one side of the first differential stage, and the amplified signal is applied differentially to the second differential stage. A single output from the second differential stage is applied to the inverter driver stage. The inverter driver stage provides the required bi-directional current drive for the two transistors of the complementary

output stage. By applying complementary symmetry, the output stage provides bi-directional current to the deflection yoke, thereby allowing symmetrical bi-directional positioning of the CRT beam. The output yoke current is sampled by means of a series 1 ohm sampling resistor, and the voltage thereby developed is returned to the input summing junction as degeneration.

Figure III-39 illustrates the deflection circuit in more detail. Connected to the input terminal E2 is the D/A step function signal which is then fed through series resistors R48 and R52 to the summing junction at the base of transistor Q1.

Symbol information is applied as input to terminal E3 and is attenuated by the voltage divider network R3, R5 and potentiometer R8, which provides for symbol gain adjustment. The signal is then fed to the summing junction through R4.

The centering control, potentiometer R7, provides a positive offset voltage to the summing junction through the series resistor R6. The voltage is derived from the +15 volt supply and is used to center displayed information on the CRT screen.

Degenerative feedback is through resistors R1 and R2. Potentiometer R2 controls the overall gain of the amplifier. Considering gain adjustment from the standpoint of the CRT display, potentiometer R2 adjusts the overall display size, or specifically, the distance between symbol positioning points. Once the overall gain has been adjusted,

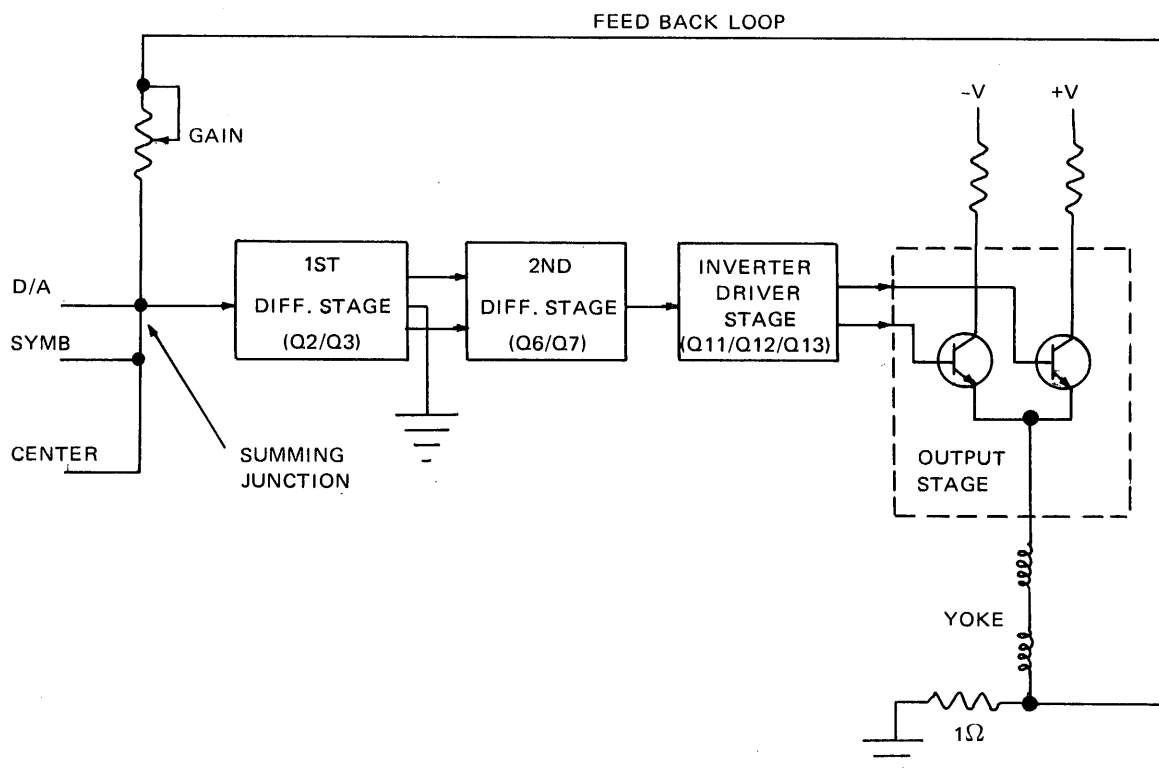


Fig. III-38 DEFLECTION AMPLIFIER BLOCK

Circuit Detail

the symbol gain adjustment controls the size of the symbols displayed, without altering the symbol positioning points.

The voltage at the input summing junction is fed through the emitter follower Q1 to the base of the first differential transistor Q2. Transistors Q2 and Q3 comprise the first differential stage of the amplifier. The base of the complementary differential transistor, Q3, is referred to signal ground through the emitter follower Q4. If the input voltage at the summing junction is at ground potential, both collectors of the first stage are nominally -11 VDC. As the summing junction goes positive, the collector of Q2 goes negative while the collector of Q3 goes positive by an equal amount. This differential stage provides stability in the deflection amplifier. Resistor R16, capacitor C1 and resistor R18 and capacitor C3 are supply voltage decoupling networks for the first stage.

The inverted signal and the in-phase signal at the collectors of Q2 and Q3 are fed to the second differential stage emitter followers Q5 and Q9 respectively. The differential stage, Q6 and Q7 provide further amplification of the applied signal. The amplified voltage waveform at the collector of Q7 is virtually in phase with the summing junction input signal.

Potentiometer R26 provides current adjustment in the second differential stage. It is adjusted to obtain the

quiescent operating point of Q7. When the summing junction input to the amplifier is referred to ground, R26 is adjusted so that the voltage drop across R24 at the collector of Q7 is such that the open loop voltage across the output sampling resistor is zero.

The in-phase voltage waveform at the collector of Q7 is fed through emitter follower Q10 to the base of the inverter-driver Q11. Operating as a class 'A' amplifier, as determined by R30 and R32, the output of Q11 is connected to output terminal E7 through emitter follower Q12. The output of Q11 is also fed through the level shifting diodes CR1, CR2, and CR3 to the emitter follower Q13. The emitter output of Q13 is then fed to output terminal E6. The output terminals E6 and E7 are in turn connected to the bases of output driver transistors (Figure III-40) which provide the required deflection yoke current. Diodes CR1, CR2, and CR3 prevent cross-over distortions at the transition point between positive and negative current in the deflection yoke.

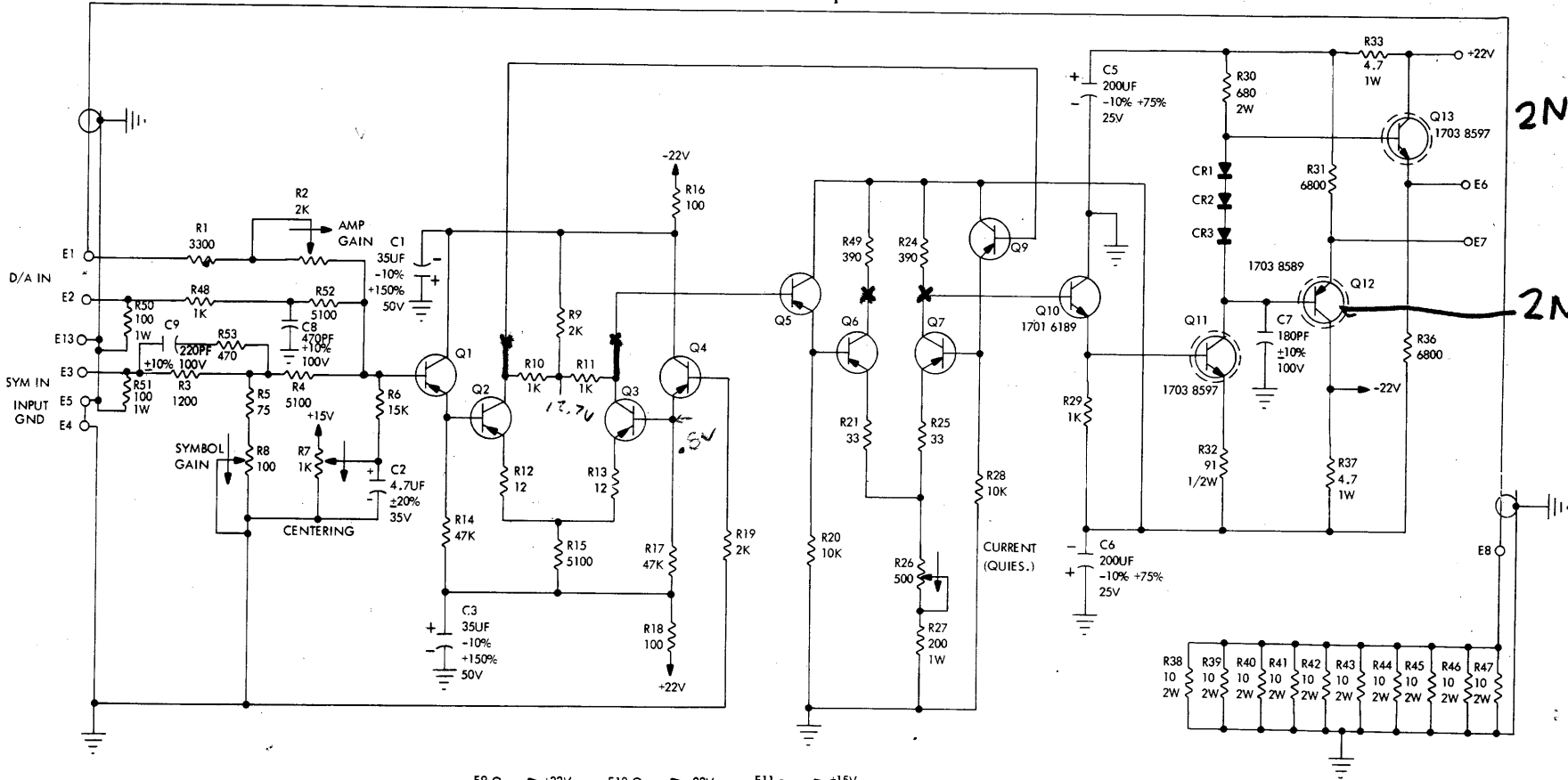
The emitters of the output transistors are connected in parallel with respect to each other and in series with the 1 ohm output sampling register. The output waveform is taken from the sampling node at terminal E8 and returned to the summing junction through the series feedback resistors.



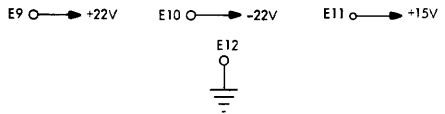
Q1 4265 936  
 Q2 27  
 Q3 29  
 Q4  
 Q5  
 Q6

310  
 4264 936A

Fig. III-39 DEFLECTION AMPLIFIER - DA



- 1. UNLESS OTHERWISE SPECIFIED:  
 ALL RESISTOR VALUES ARE IN OHMS  
 ±5%, 1/4W  
 ALL DIODES ARE 1701 6106  
 ALL TRANSISTORS ARE 1701 6197



2N698

2N4013

Circuit Detail

Circuit Detail

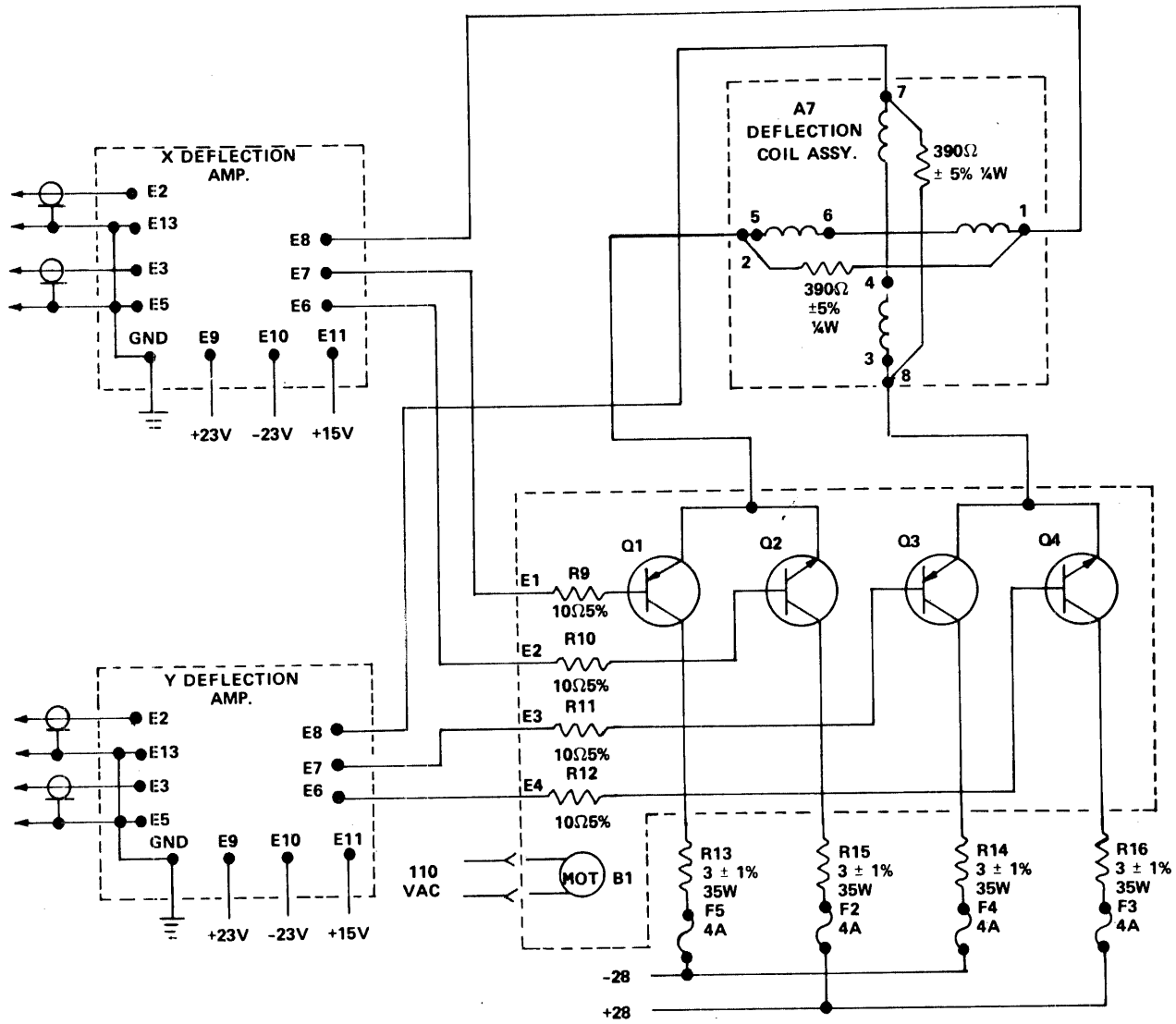


Fig. III-40 MONITOR ASSEMBLY INTERCONNECTIONS

Adjustments

CONTROL UNIT

**SYMBOL GENERATOR (BI-DIRECTIONAL INTEGRATOR)**

The symbol generator consists of two Bi-directional Integrator cards located at positions 307 and 308. These cards control the horizontal and vertical positioning of the CRT beam about the character position. Three potentiometers on each card control the charging current being applied to capacitor C3 of each circuit. The charge on C3 for both X and Y is buffered and fed to the deflection amplifiers in the monitor. To produce the correct length stroke, the charge on C3 must be the proper value. Potentiometers R11, R12, and R13 are used to provide this value.

**Adjustment**

1. The oscilloscope setup is as follows:  
 Sync - EXT +  
 Time base - 2.0 usec/cm  
 Vertical - .05 v/cm

2. Apply power to the Control Unit and a Monitor and fill the memory with information from the keyboard.
3. Connect probe A of the scope to the junction of R19 and C3 (Point A in Figure IV-1). Connect the scope ground lead to point B. Connect the Sync lead to DISC---1 at backplane pin FB9G.
4. At the rear of the Control Unit, place the SYM GEN switch (on the power supply gate) to ON. The screen will go blank if the intensity is normal. The NL character is now being forced into the rope.
5. Adjust the TRIGGER level of the scope to display a presentation similar to Figure IV-2. Adjust the vertical position so that the reference line is 1 cm. below the center line.
6. The small ramp represents two single, positive strokes followed by two single, negative strokes. The single

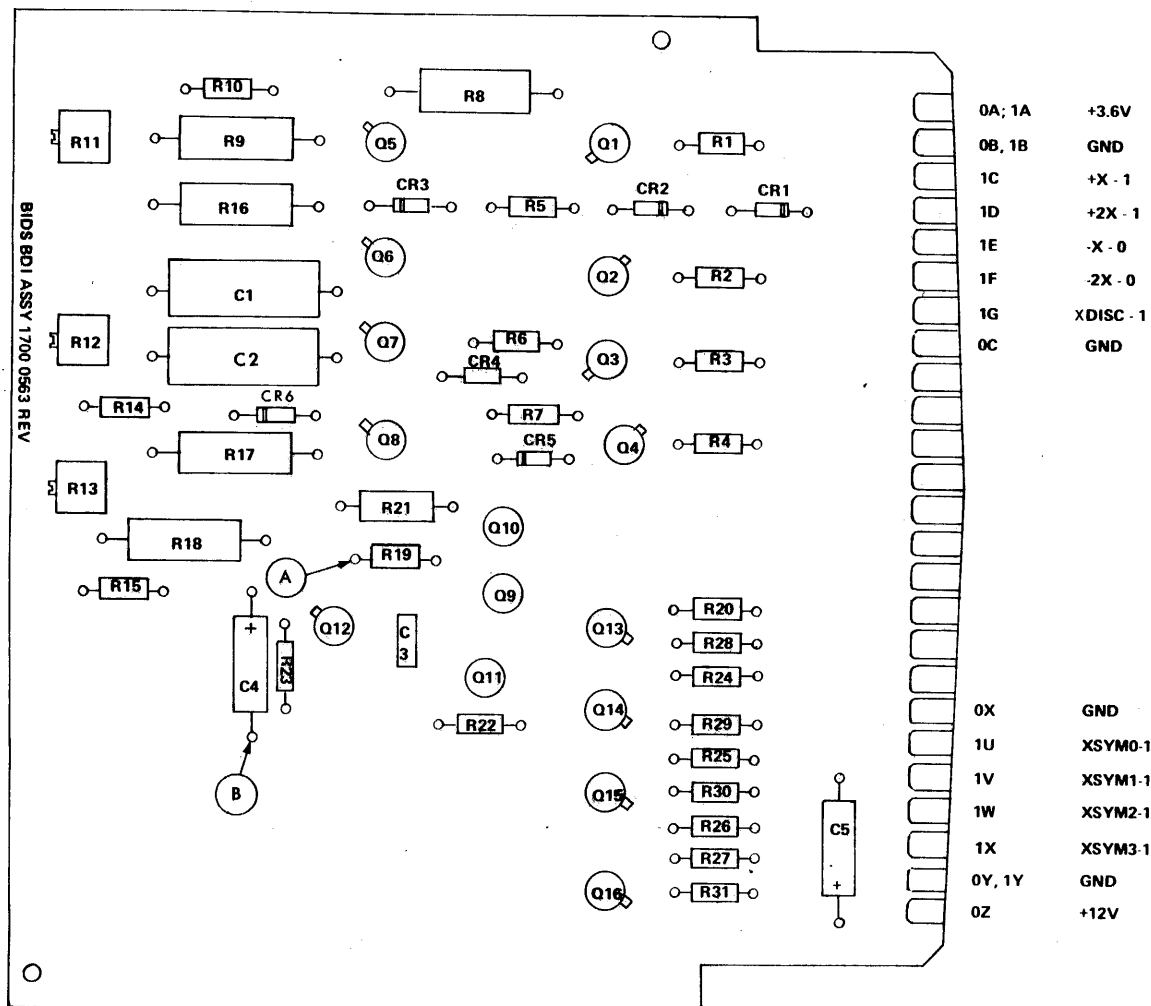


Fig. IV-1

For Library Binder 110

Adjustments

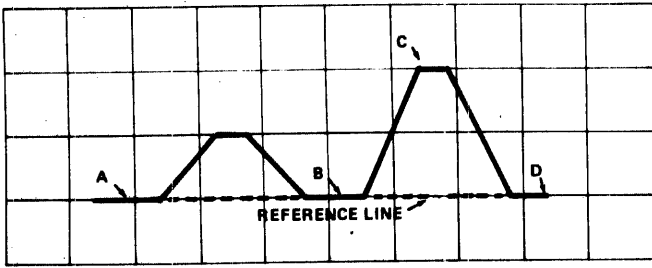


Fig. IV-2

positive stroke is not adjustable and forms the reference for the remaining strokes. Using the Variable adjustment control on the oscilloscope pre-amp, adjust the leading edge of the first ramp so that the amplitude is 1 cm. in height. Adjust R12 (center pot.) of the card to position line B to the reference line. R11 (upper pot.) is now adjusted to position line C 2

cm. above the reference line. R13 (lower pot.) is now adjusted to position line D at the reference line. The presentation should now appear as in Figure IV-2.

7. The above procedure is then repeated for the remaining integrator card.

MASTER CLOCK

The master clock consists of a crystal controlled oscillator operating at 3.3333 MHz. Most of the Input Display System logic operates at half this frequency. This is accomplished by complementing a flip-flop at the 3.3333 MHz rate and using only the output from the 1 side of the flip-flop. Certain portions of the logic require the 3.3333 MHz rate. In both cases, the clock output to the System is buffered to provide the necessary drive requirements.

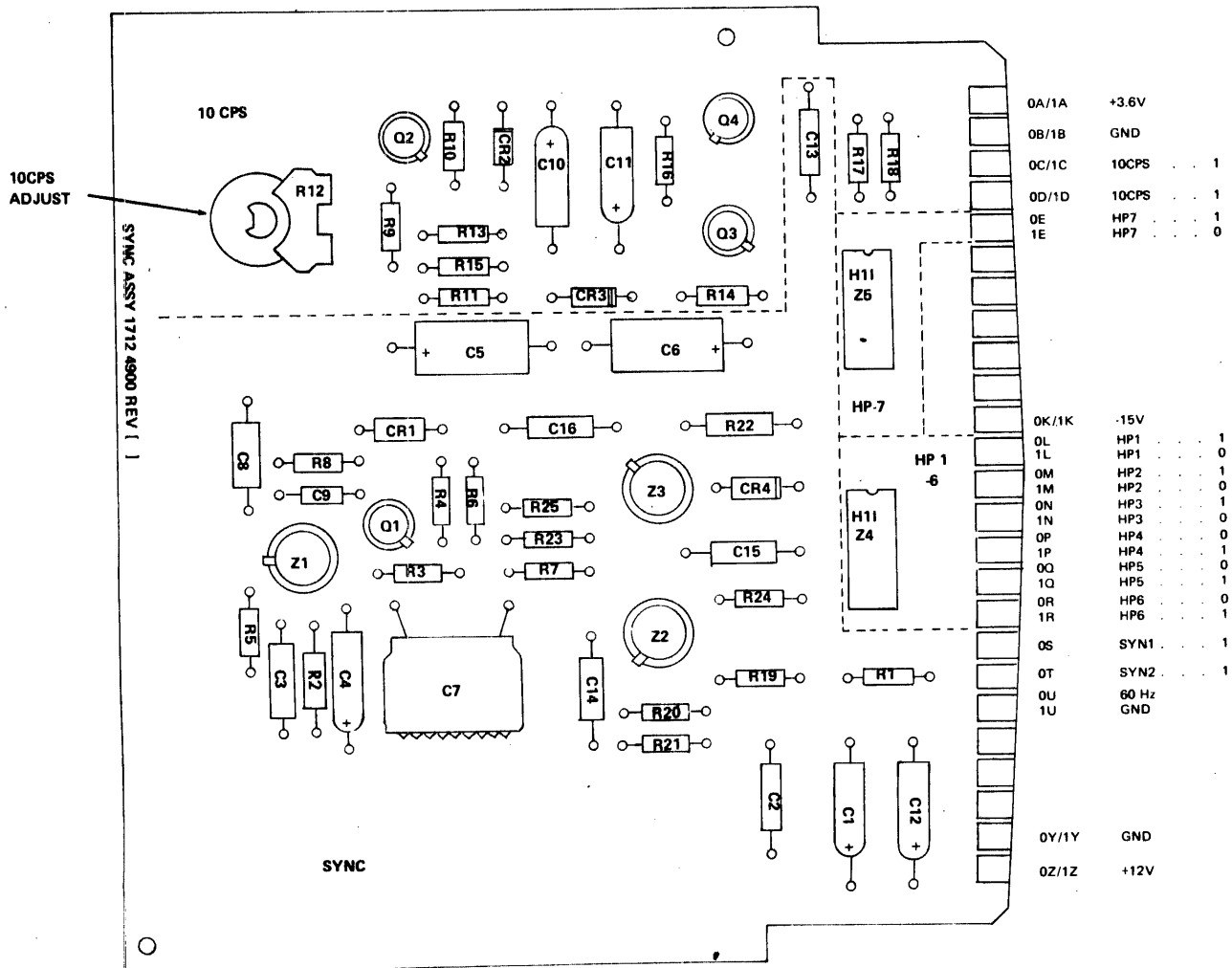


Fig. IV-3

Adjustments

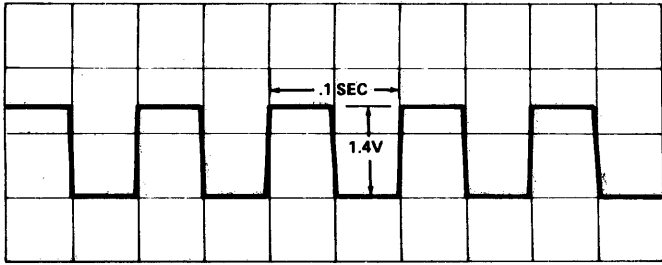


Fig. IV-4

Adjustment

An adjustment has been provided to control the delay between the leading edge of the main clock, CLKA (1.6666Mhz), and the leading edge of the 3CLK (3.3333Mhz), and is accomplished in the following manner:

1. Connect Trace A to CLKA--1 at backplane pin DE1I.
2. Connect Trace B to 3CLK--1 at backplane pin DE1J.
3. Sync positive on the A trace.
4. Adjust R32 so that the positive going edge of 3CLK precedes the positive going edge of CLKA by 15 nanoseconds.

Refer to Figure V-6 for component layout.

10CPS CLOCK

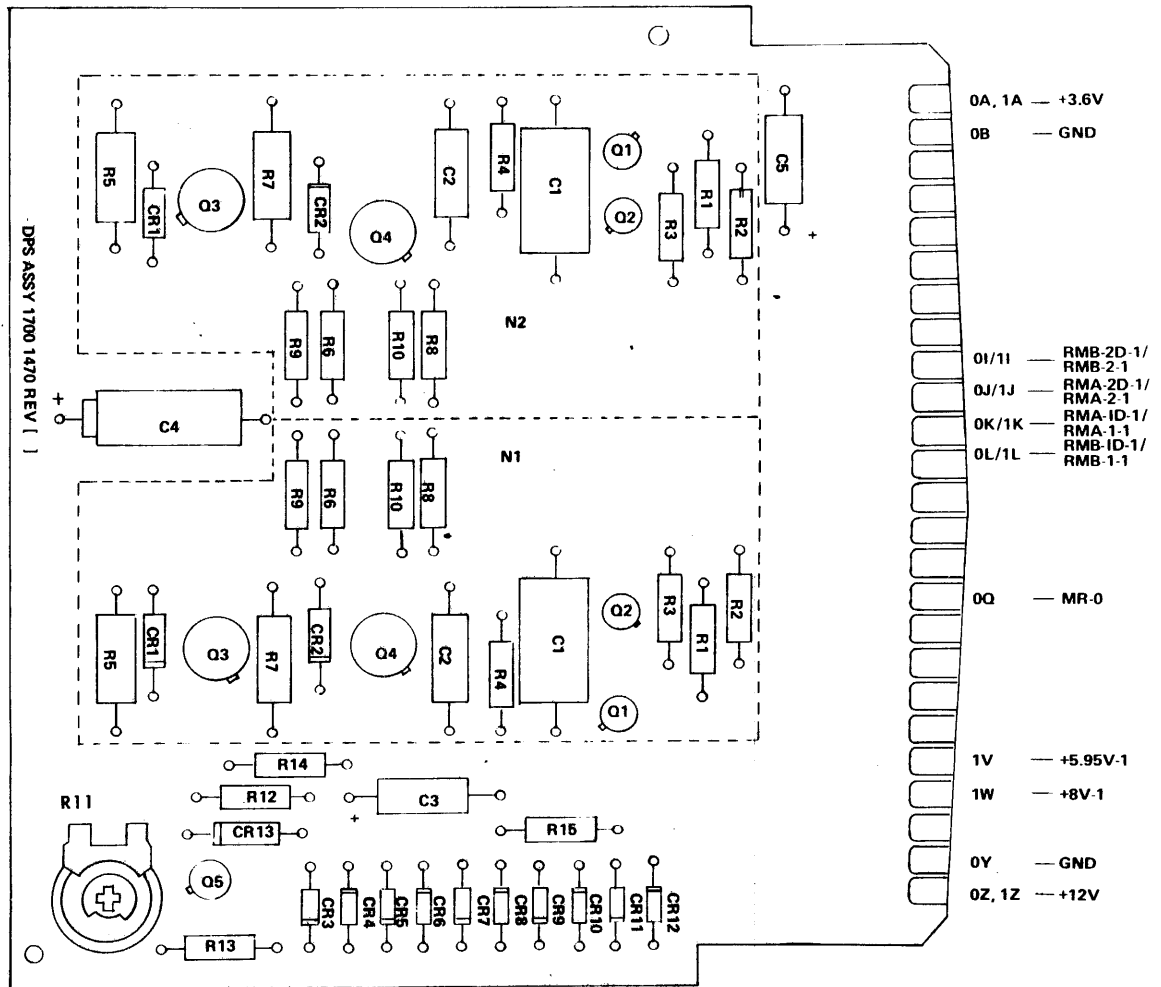
The 10CPS clock is used for the keyboard repeat function and consists of a free running multivibrator circuit on the card located at position 201.

Adjustment

1. Set up the oscilloscope as follows:  
 Sync - Int +  
 Time base - 50 ms/cm  
 Vertical - .1 v/cm
2. Extend the card at location 201.
3. Connect probe A to pin DAOD in the Control Unit and apply power.
4. Presentation should be approximately as shown in Figure IV-4.
5. Potentiometer R12, Figure IV-3, should be adjusted to correct timing.

INFORMATION DRIVER POWER SUPPLY

The information driver card requires two voltages that are not available from the regular supplies. Adjustment



✓ Fig. IV-5

Adjustments

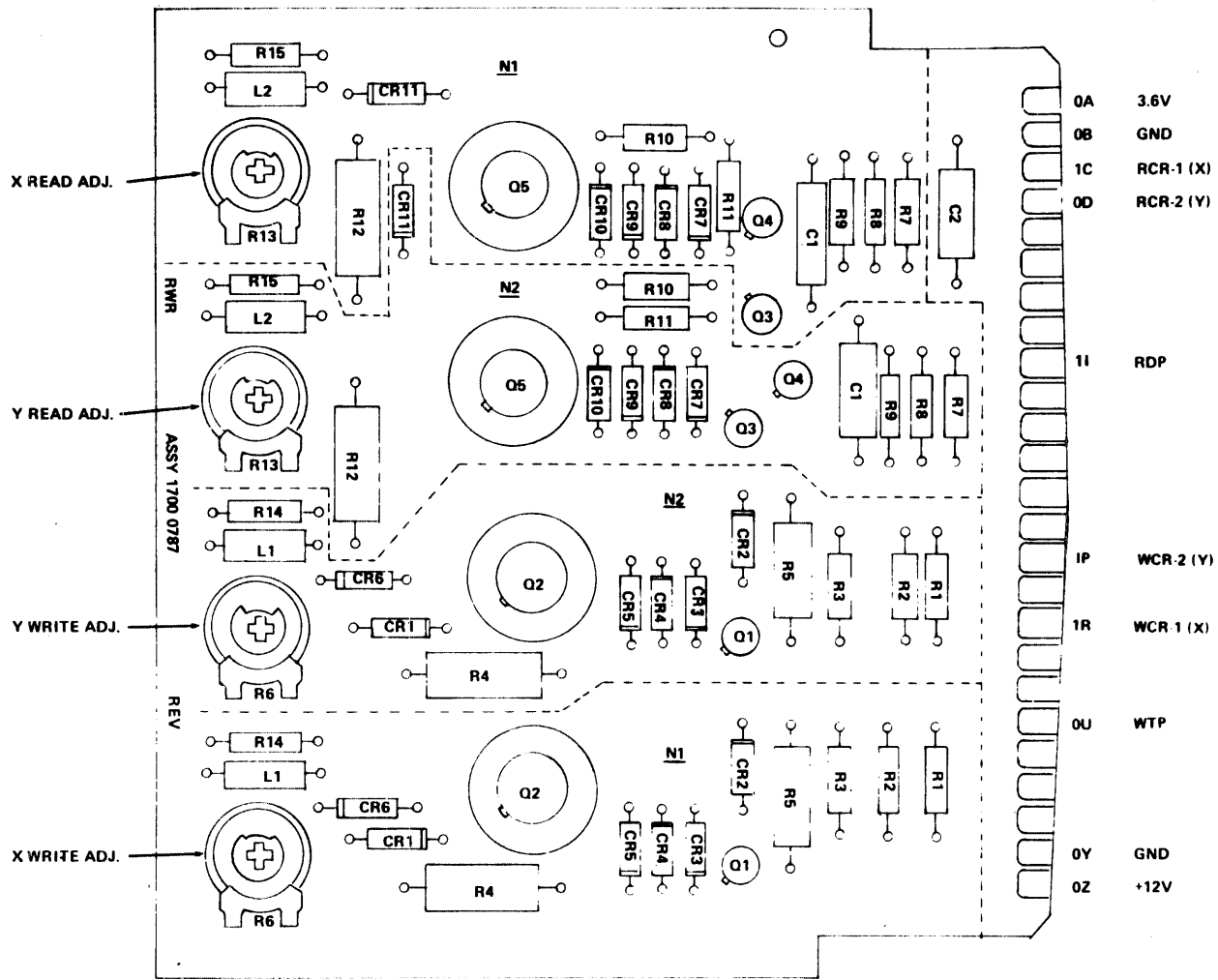


Fig. IV-6

can be made to the +5.95 volt supply on DPS assembly location 132 but not to the +8 volt supply.

Adjustment

1. With a voltmeter connected from pin BH2V to ground and set to a range of 6 volts, the supply should read +5.95 volts.
2. R11, Figure IV-5, should be adjusted if required.

MEMORY CURRENT

In order for the memory system to operate correctly, both X and Y read and write currents must be set within prescribed limits. Adjustments for the four currents are on the Read Write Regulator card at location 232. Figure IV-6 illustrates the card and identifies the potentiometers.

Adjustment

1. Set up the oscilloscope as follows:  
Sync - Int +

Time base - .5 usec/cm

Vertical Sufficient to provide adequate display.

2. Position current probe around desired memory current line at the backplane.  
Read X - DG9C to DH8Y  
Read Y - DG8D to BH8Y  
Write X - DG9R to DH8X  
Write Y - DG9P to BH8X

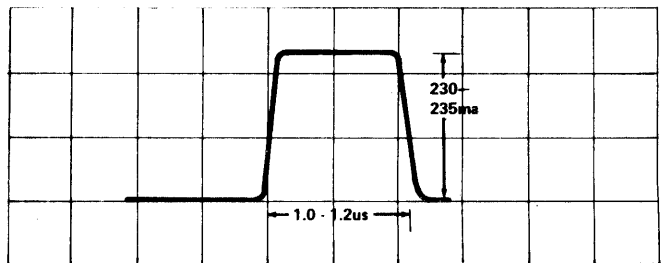


Fig. IV-7

Adjustments

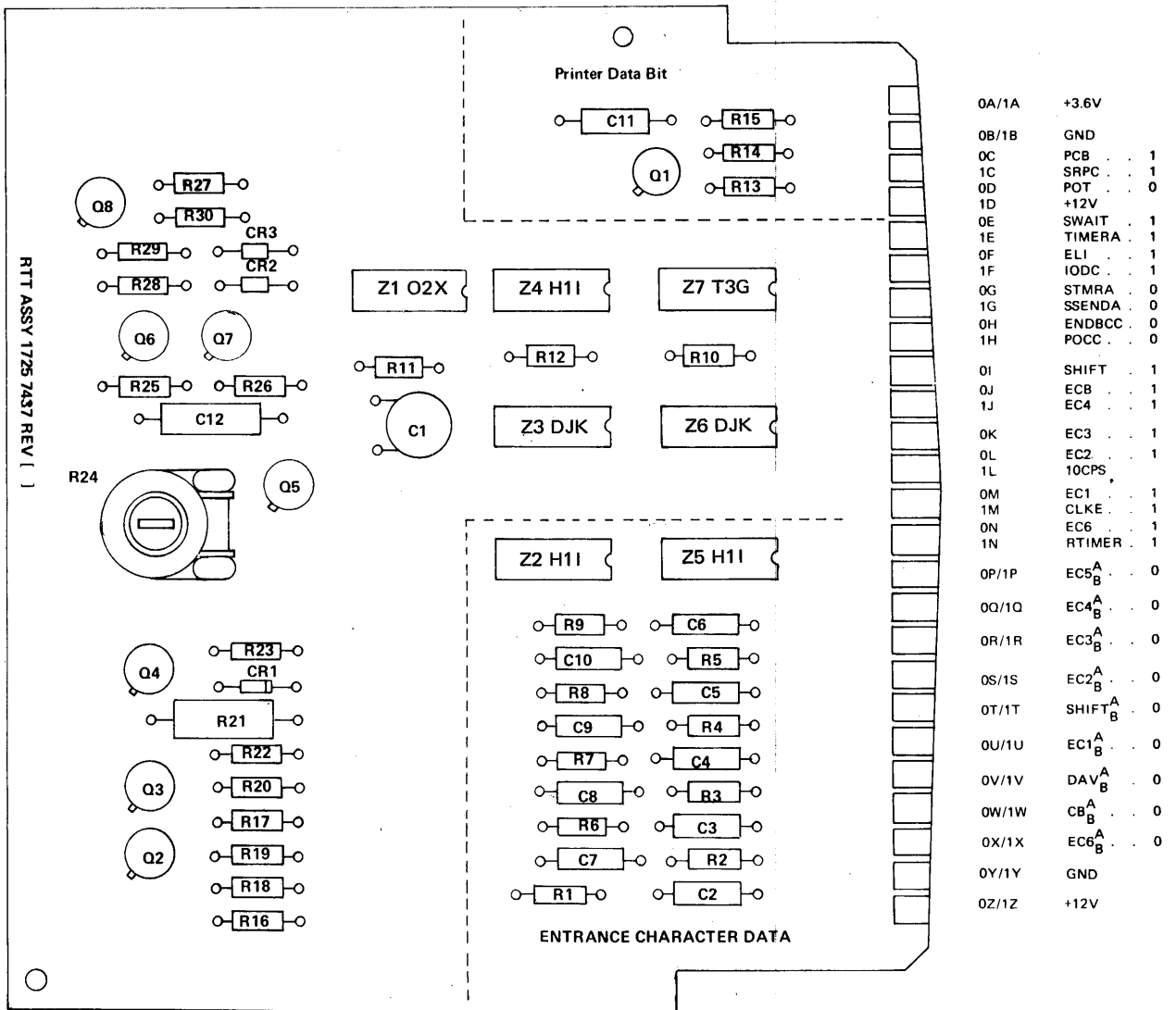


Fig. IV-8

3. The presentation should be similar to the waveshape in Figure IV-7.
4. When an adjustment is indicated, adjust R13 for read current or R6 for write current to at least 230 ma but not more than 235 ma.

RETRANSMIT TIMER

The retransmit timer consists of chip logic and discrete component logic all mounted on the card assembly at location 310. When an ENQ is transmitted to the processor system, the timer is armed. If either a NAK or no response is received when the timer times out, the RETRANSMIT light will be lit to indicate to the operator to retransmit the message.

Adjustment

CAUTION

Remove the Block Check Register card from location 212 before executing the following procedure.

1. The oscilloscope is set up as follows:  
Sync - INT +  
Time base - .5 sec/cm  
Vertical - .1 v/cm
2. Connect the scope probe to FC8E (TIMERA-1) on the backplane.
3. With a short jumper lead, apply +3.6 volts to FC7H (ENDBCC-0) momentarily. (+3.6 may be obtained at FC7A). It may be necessary to repeat this step in order to adjust the scope trigger.
4. The timer should time out in approximately 4 seconds. Adjust R24 (Figure IV-8) if required.

### Adjustments

#### MONITOR ADJUSTMENT

##### BRIGHTNESS

To prevent burning the face of the CRT, the Limit Control, R5, is adjusted to limit the setting of the operator controlled Brightness thumb wheel. The following procedure is to be used to set the limit control:

Adjustment (Refer to Figure IV-9A)

1. Display a line of characters from the Keyboard.
2. Turn the limit control fully clockwise.
3. Rotate the Brightness thumbwheel fully to the right for maximum brightness.
4. Adjust R5 to the maximum desired brightness, while retaining a sharp character presentation. Excess brightness will cause the characters to blur and can cause premature failure of the CRT.

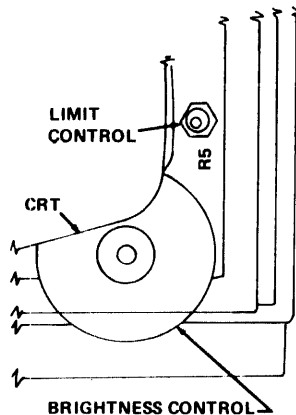


Fig. IV-9A

##### DEFLECTION AMPLIFIER

On each of the two deflection amplifiers are four potentiometers. They function as follows:

1. R-2 Raster Gain - Controls overall width and height.
2. R-7 Centering Control - Centers display on the screen.
3. R-8 Symbol Gain - Controls character size.
4. R-26 Crossover - Controls quiescent operating point of the second differential amplifier.

Adjustments: Horizontal (Refer to Figure IV-9B)

The following adjustments all require a display presentation of a full line of 80 characters:

- Raster gain – Adjust R2 of the XDA board for a line width of approximately 11.5 inches.
- Centering – Adjust R7 of the XDA board to center the displayed line on the screen.
- Symbol gain – Adjust R8 of the XDA board to provide proper character width. Adjacent characters

should not touch but should have sufficient space to be legible.

- Crossover – Adjust R26 of the XDA board so that horizontal strokes are straight (most affect at pos. 32 and 64).

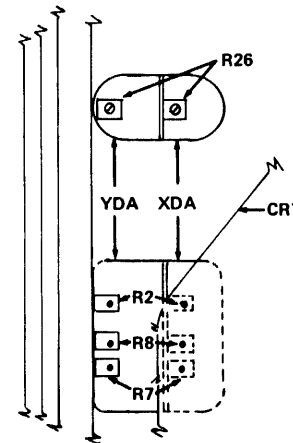


Fig. IV-9B

Adjustments: Vertical (Refer to Figure IV-9B)

The following adjustments all require a display presentation of a full line of 80 characters at the top line and the bottom line.

- Raster gain – Adjust R2 of the YDA board so that the distance from the top of the top line to the bottom of the bottom line is approximately 7.5 inches.
- Centering – Adjust R7 of the YDA board to center the display on the screen.
- Symbol gain – Adjust R8 of the YDA board to provide proper character height. Characters should be higher than they are wide.
- Crossover – Adjust R26 of the YDA board so that vertical strokes are straight (most affect at pos. 32 and 64).

##### LOW VOLTAGE

The 3.6 volt supply in the Monitor is the only voltage that is adjustable. Potentiometer R2 on the PSLD board mounted to the left of the CRT on the chassis should be adjusted to 3.6v at connection E16 of the board. (Refer to Figure IV-9C.)

##### 'Z' AMPLIFIER

The 'Z' axis, or intensity amplifier has two inputs, the Z1 and Z2 lines. The Z1 level alone provides the intensity for single length strokes, while Z1 and Z2 together provide the intensity for double length strokes. The intensity level produced by Z1 is controlled by potentiometer R9 and is



Adjustments

adjusted in the following manner:

1. Display a line of zeros.
2. Adjust R9 so that the intensity of the short diagonal

portions of the characters is the same as for the longer strokes. (Refer to Figure IV-9C.)

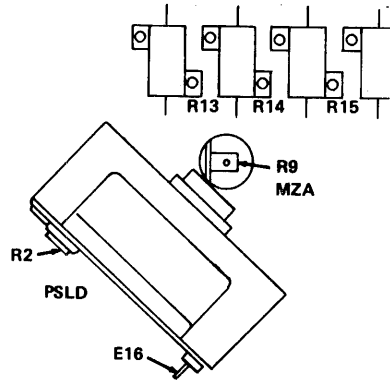


Fig. IV-9C

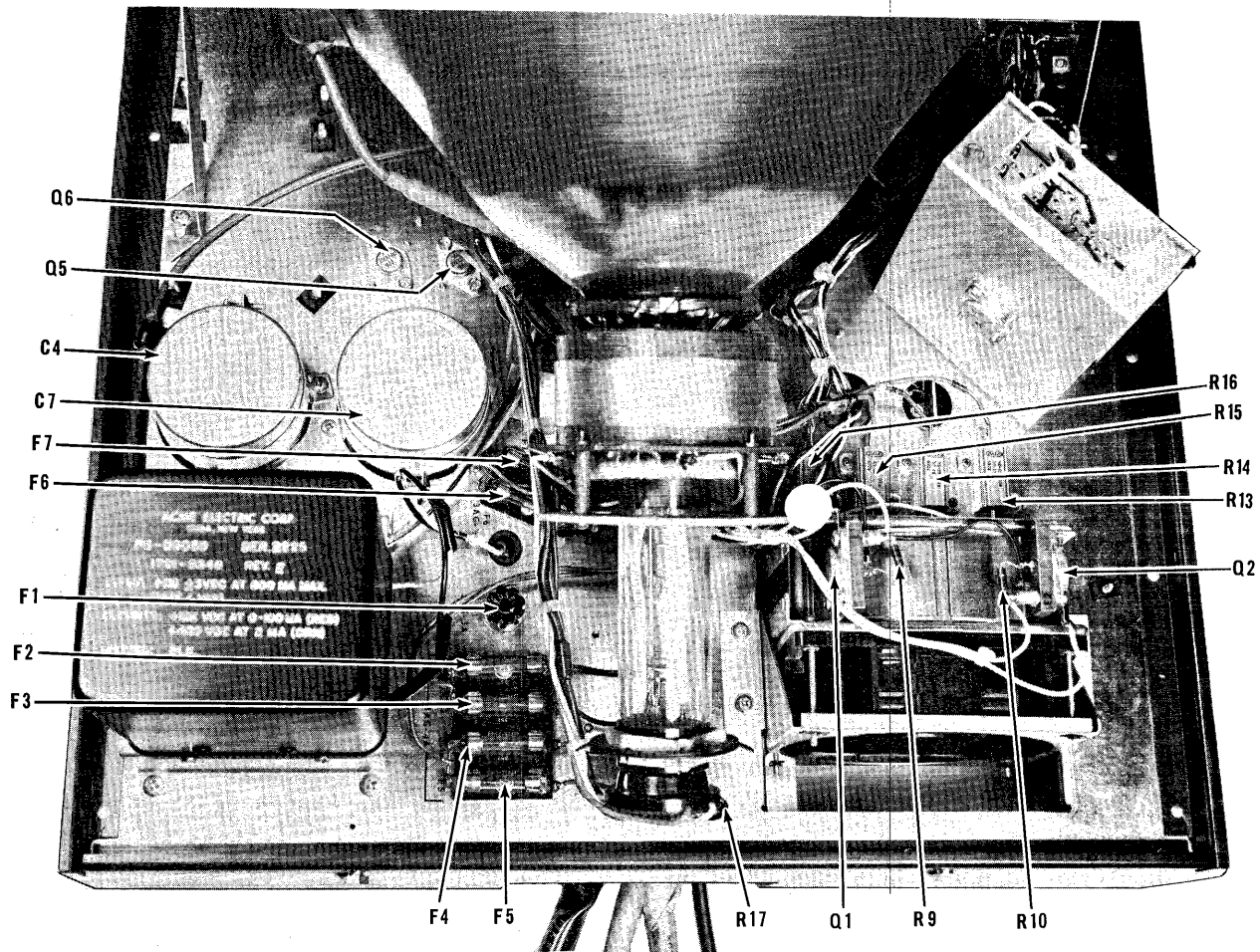


Fig. V-1

ILLUSTRATIONS

| Figure | Description                                    | Location        |
|--------|--|-----------------|
| 1      | Monitor Chassis – Top View . . . . .           | Monitor         |
| 2      | Monitor Chassis – Bottom View . . . . .        | Monitor         |
| 3      | Monitor Deflection Amp. Assy. . . . .          | Monitor         |
| 4      | Monitor Intensity (Z) Amp. Assy. . . . .       | Monitor         |
| 5      | Monitor Power Supply Lamp Driver Assy. . . . . | Monitor         |
| 6      | Drivers and Receivers . . . . .                | 126,127         |
| 7      | Inhibit Drivers . . . . .                      | 128             |
| 8      | Sense Amplifiers . . . . .                     | 130             |
| 9      | Information Drivers. . . . .                   | 131             |
| 10     | X-Y Switch Core Matrix(2X2) . . . . .          | 133             |
| 11     | X,Y Switch Core Matrix(2X4) . . . . .          | 134,135         |
| 12     | Clock Osc. . . . .                             | 221             |
| 13     | Core Translator . . . . .                      | 129/230         |
| 14     | Translator Sense Amplifier. . . . .            | 231             |
| 15     | Switches and Drivers(X,Y) . . . . .            | 233,234         |
| 16     | X Digital/Analog . . . . .                     | 301             |
| 17     | Line Drivers . . . . .                         | 302,303,304,305 |
| 18     | Y Digital/Analog . . . . .                     | 306             |
| 19     | Data Set Interface . . . . .                   | 311             |
| 20     | Two Wire Direct Interface . . . . .            | 311             |

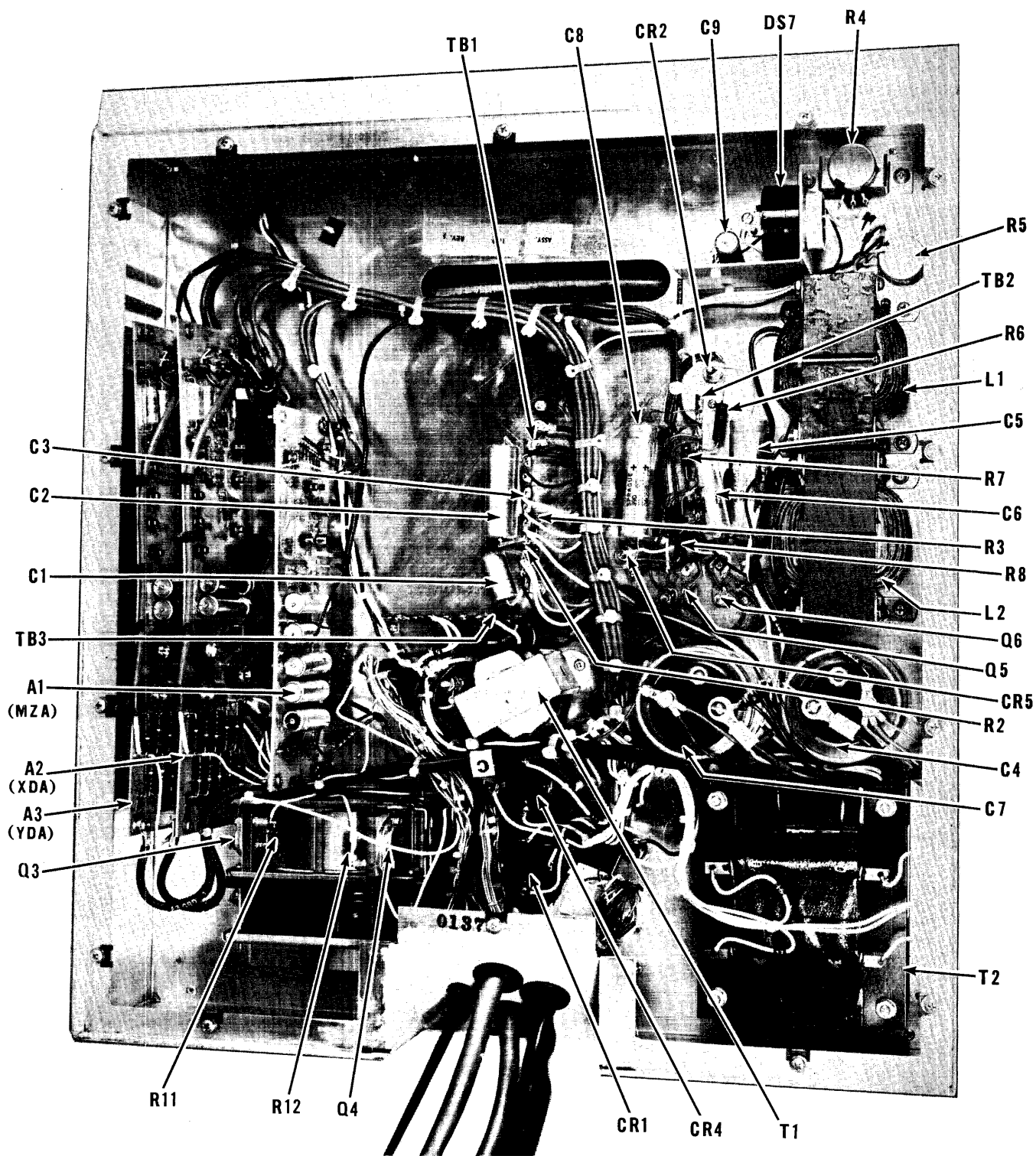


Fig. V-2

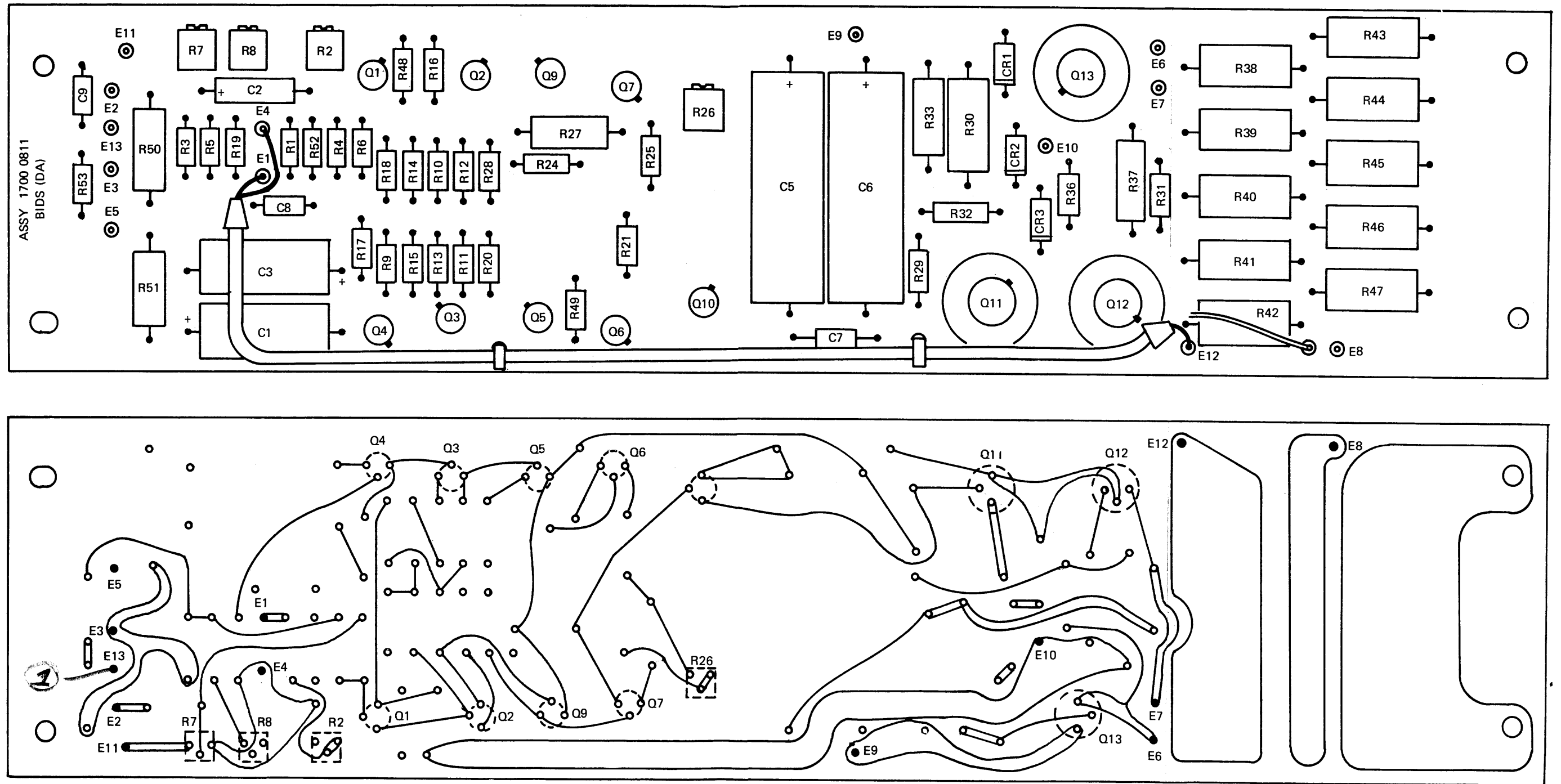


Fig. V-3

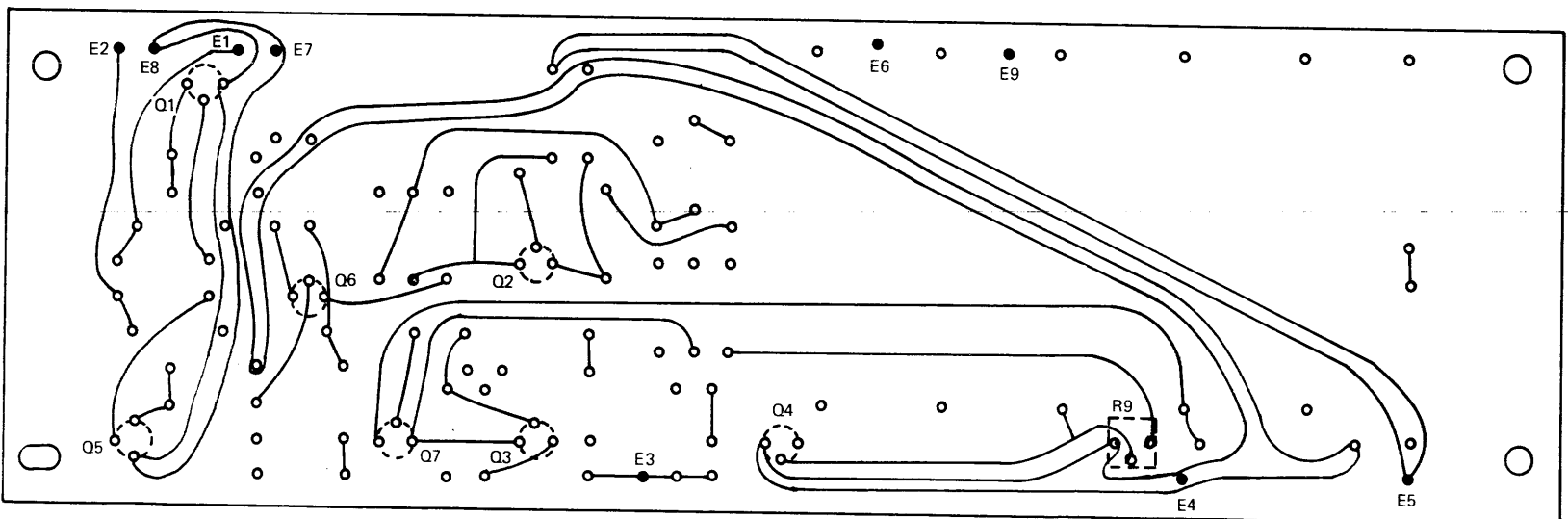
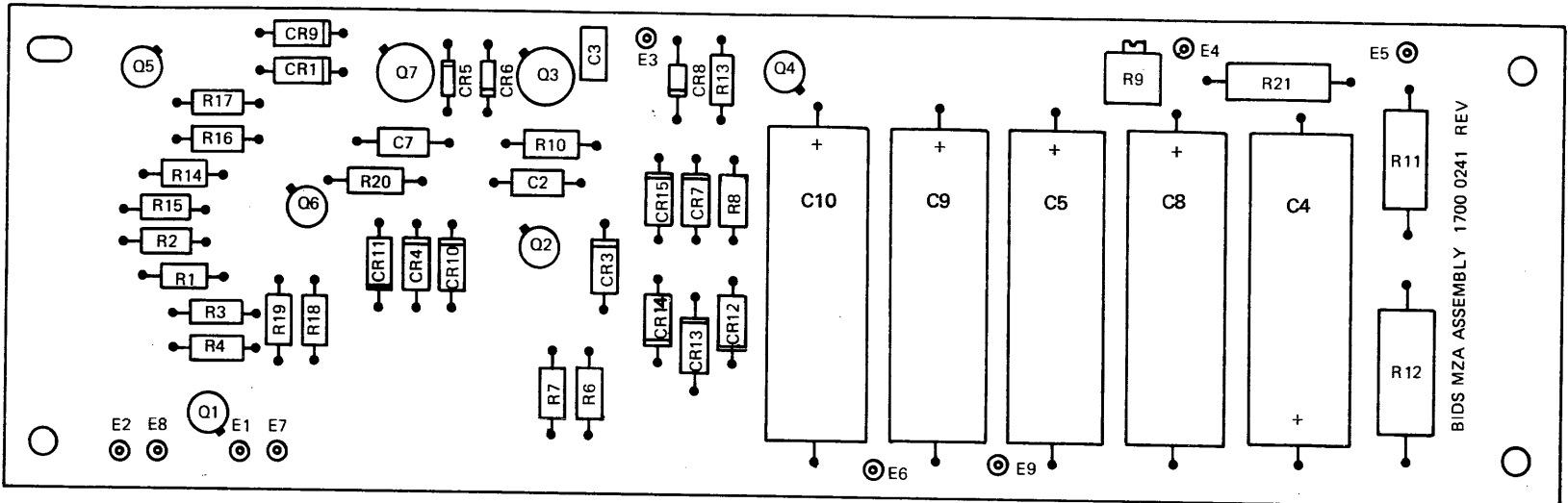


Fig. V4

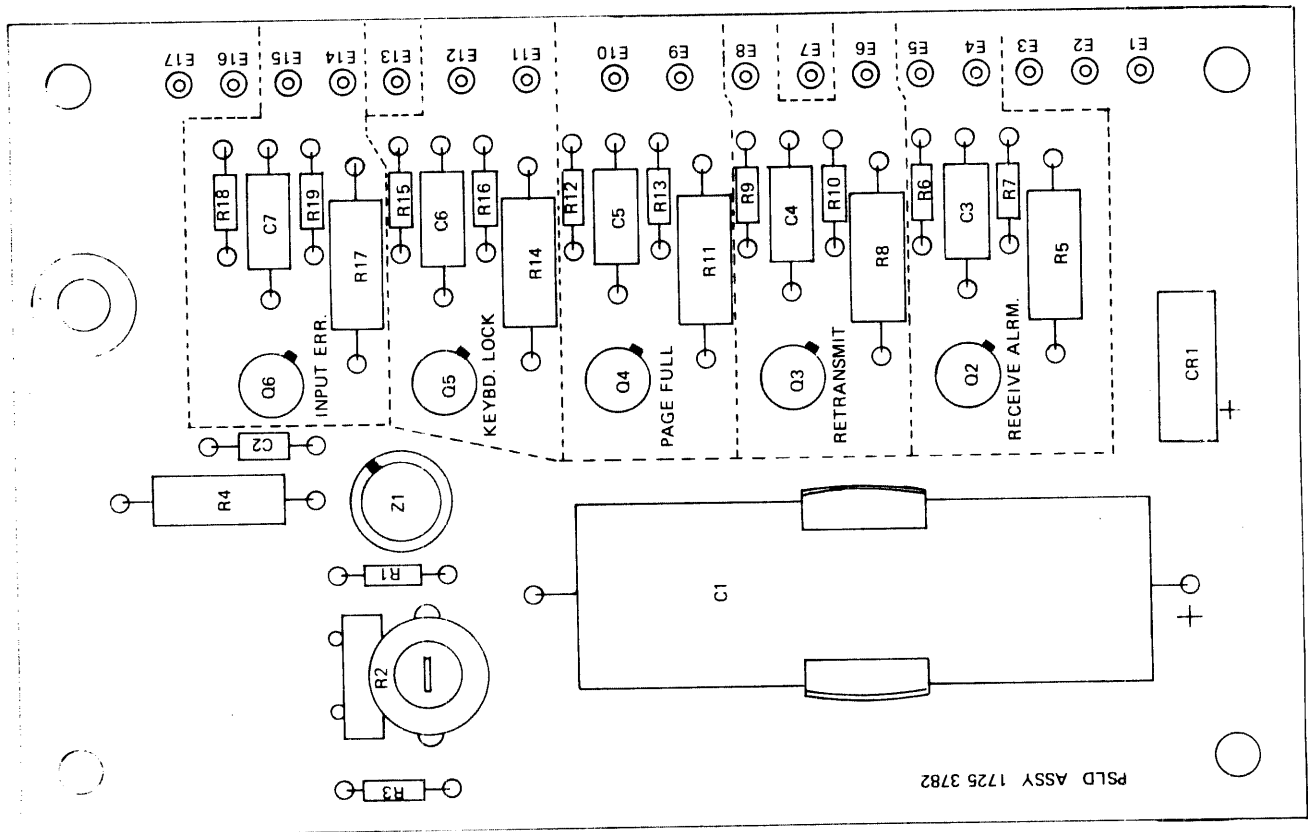
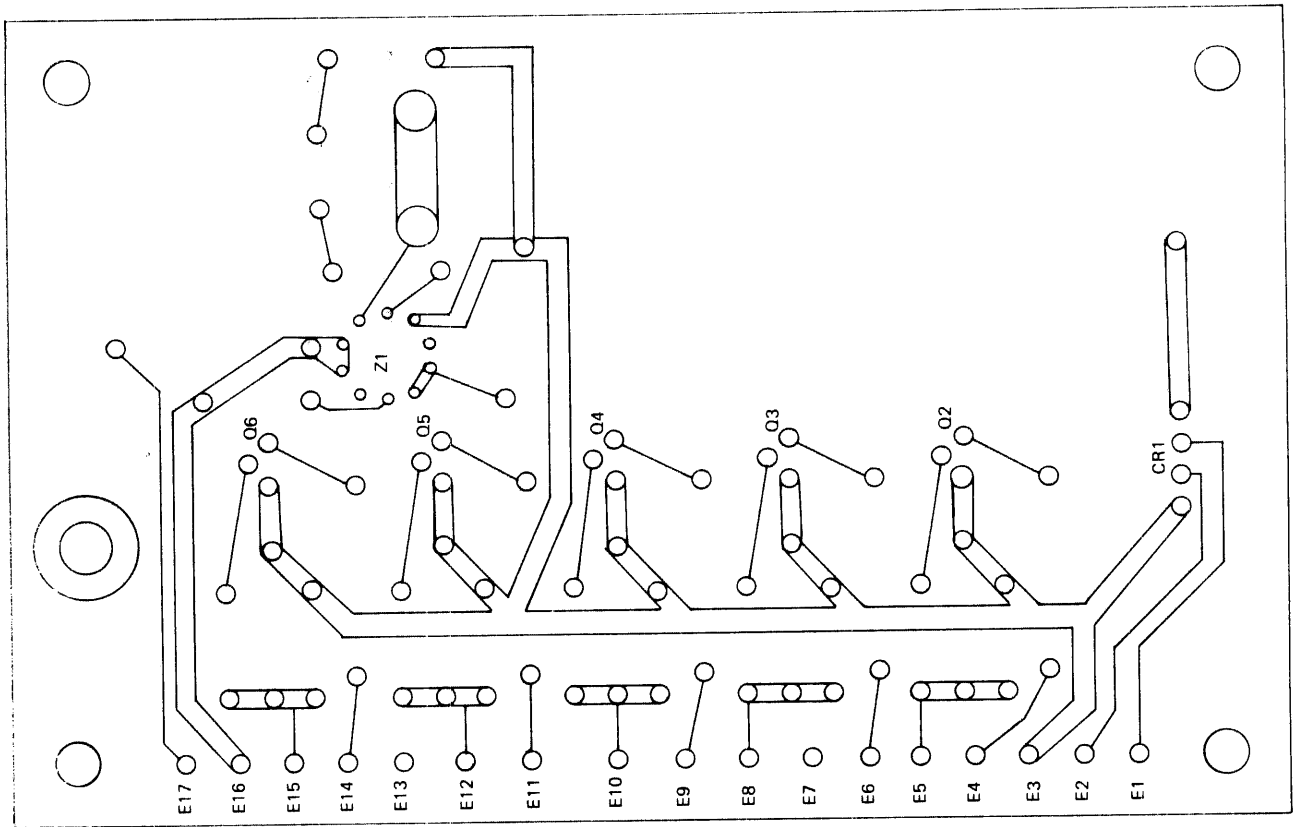


Fig. V-5

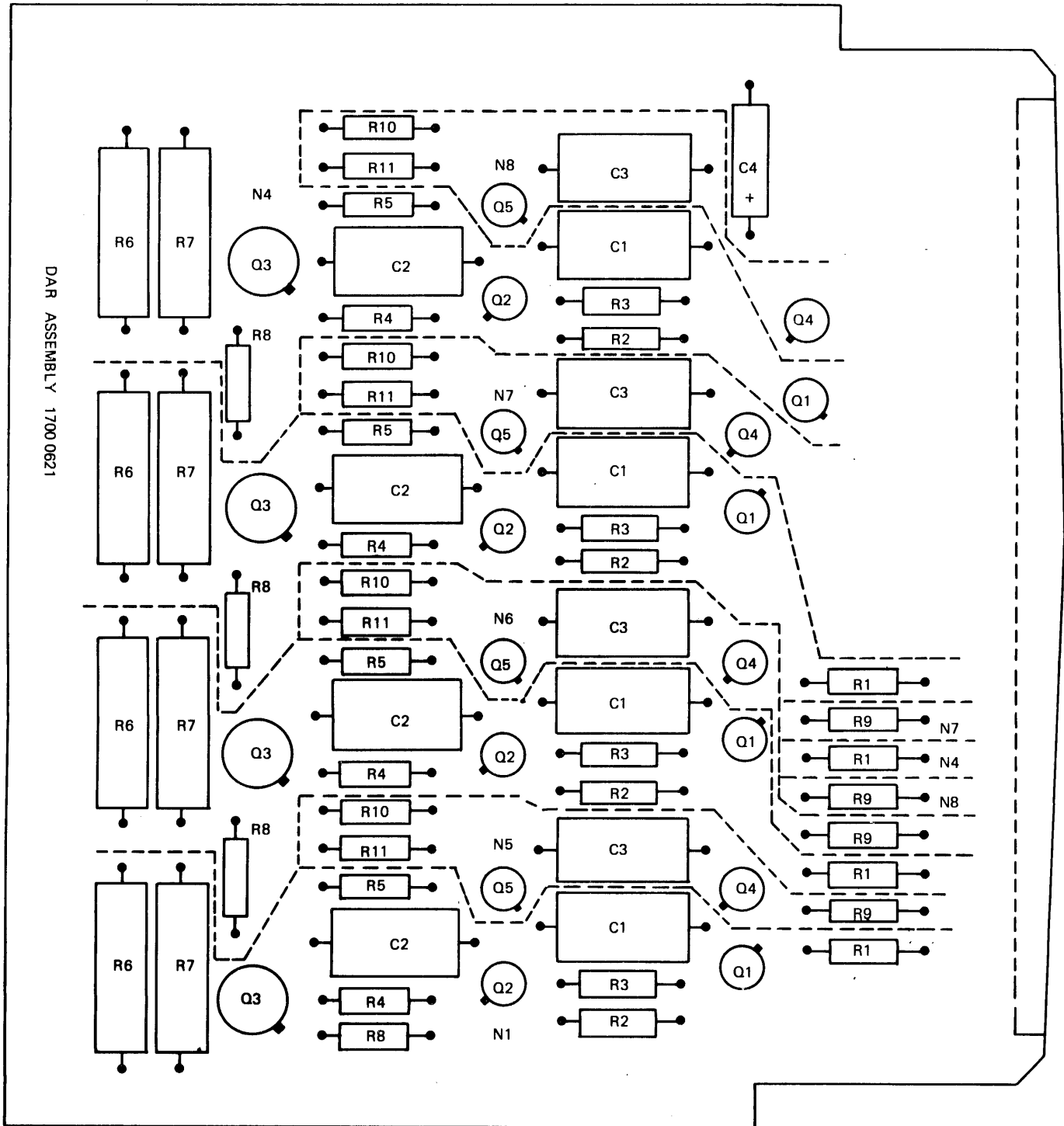


Fig. V-6

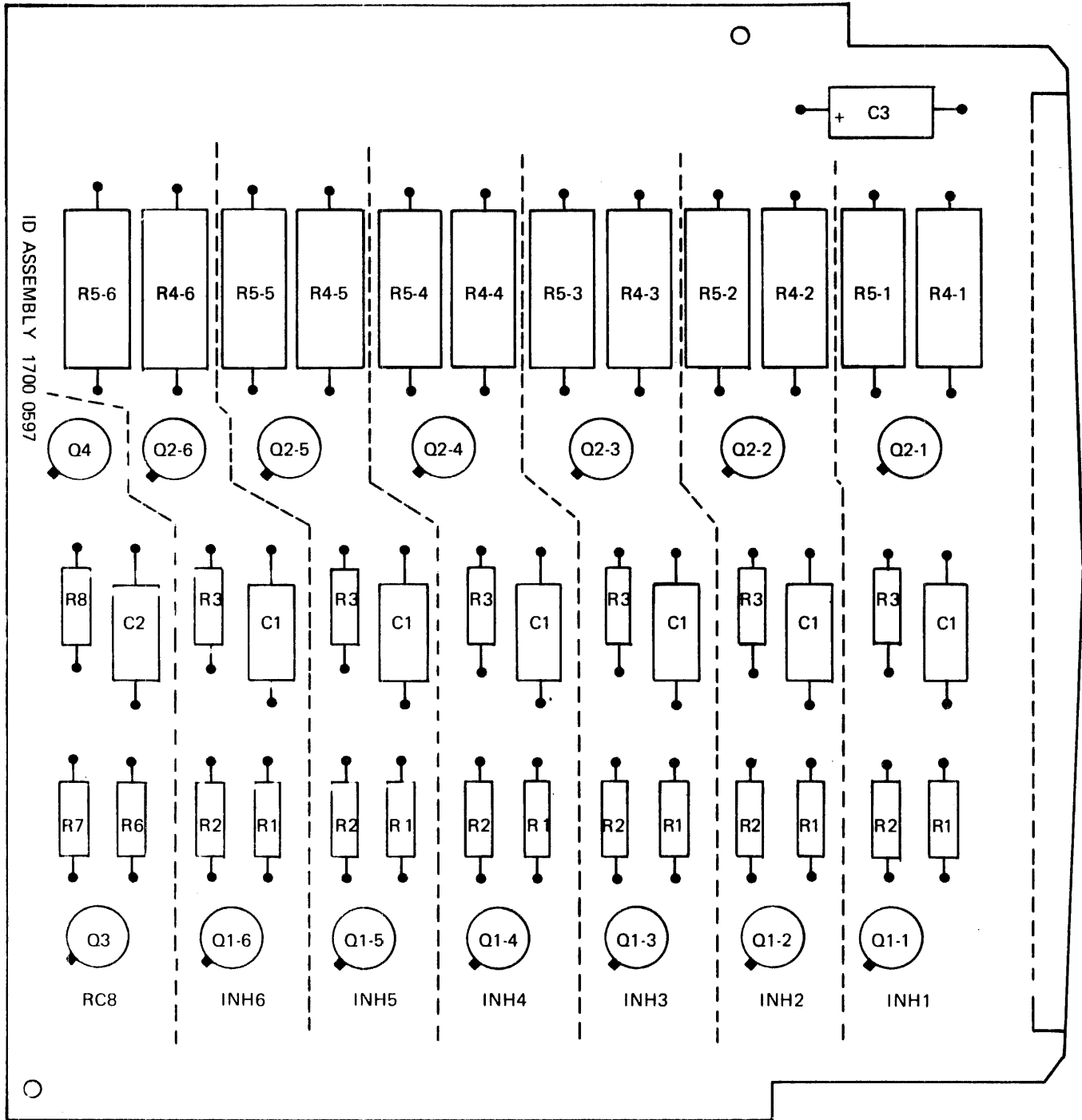


Fig. V-7



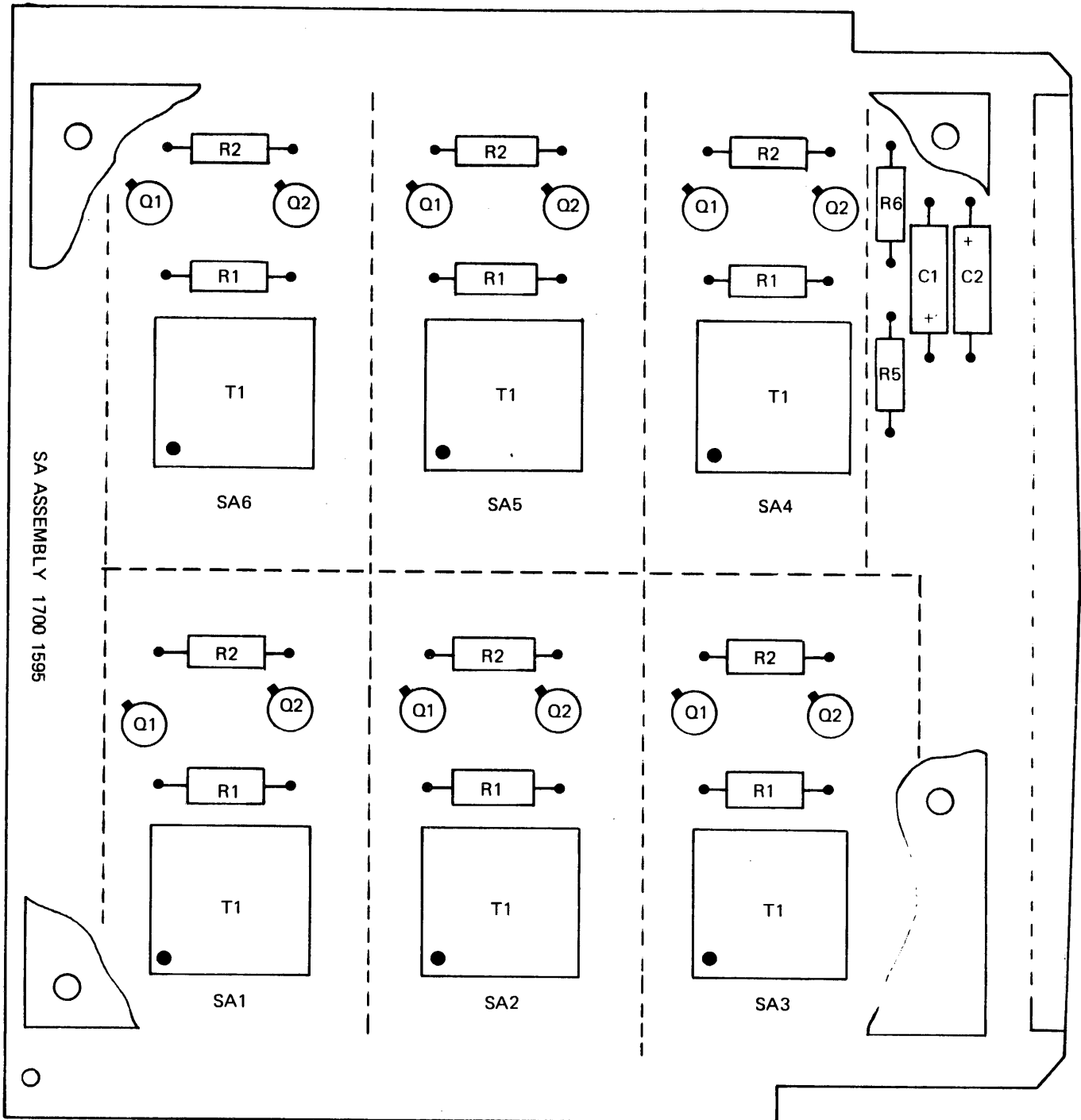


Fig. V-8

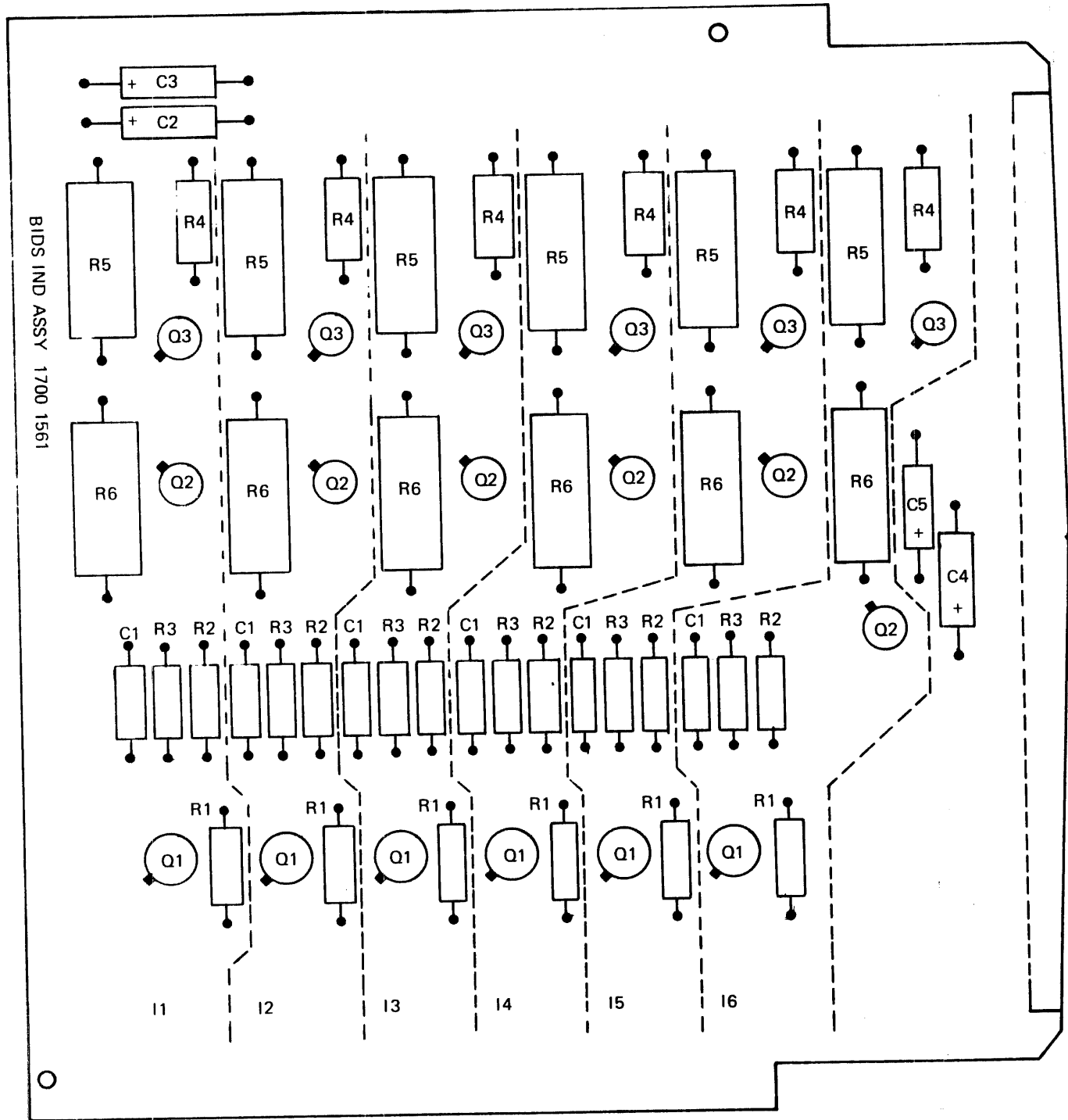


Fig. V-9

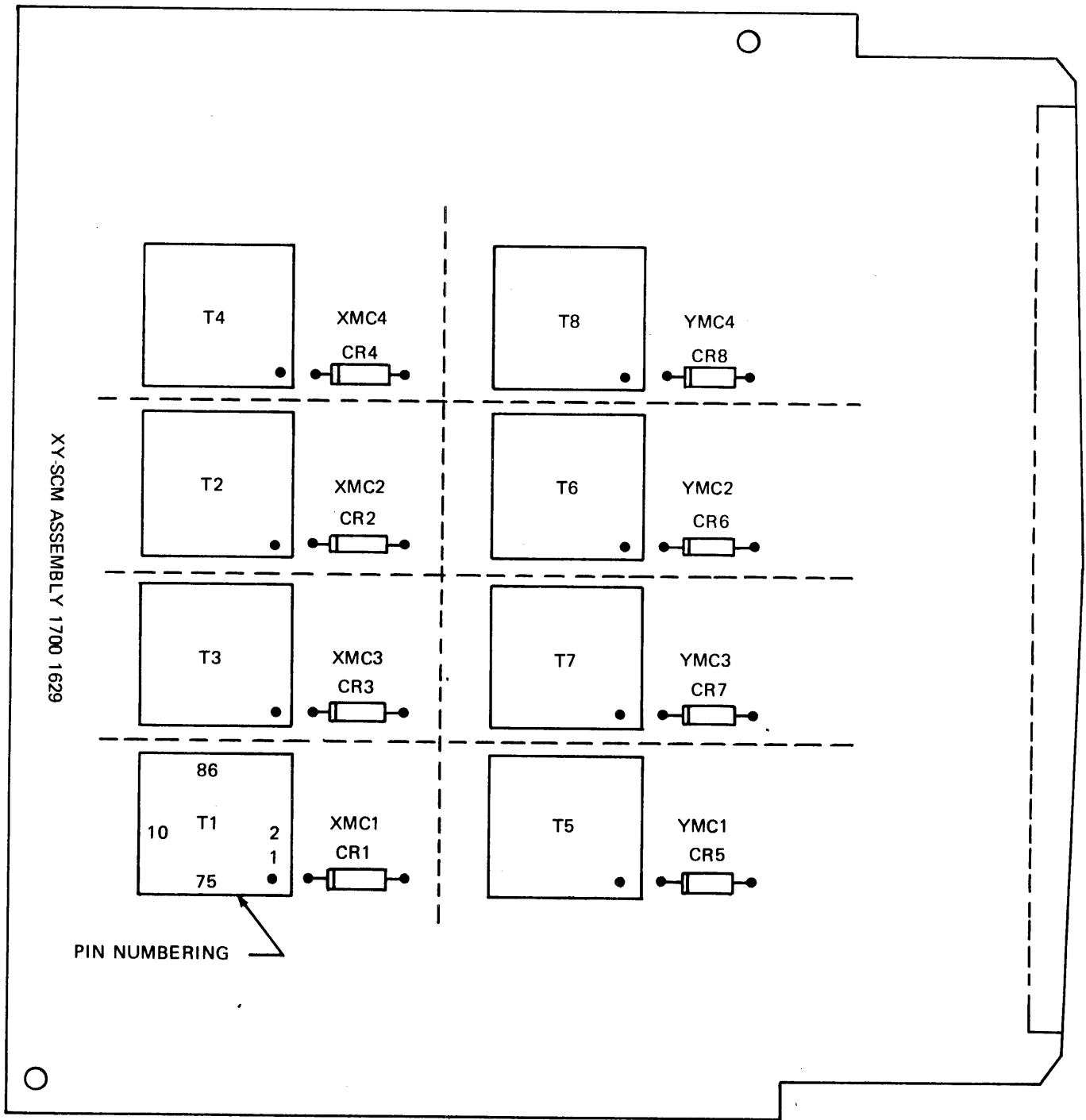


Fig. V-10

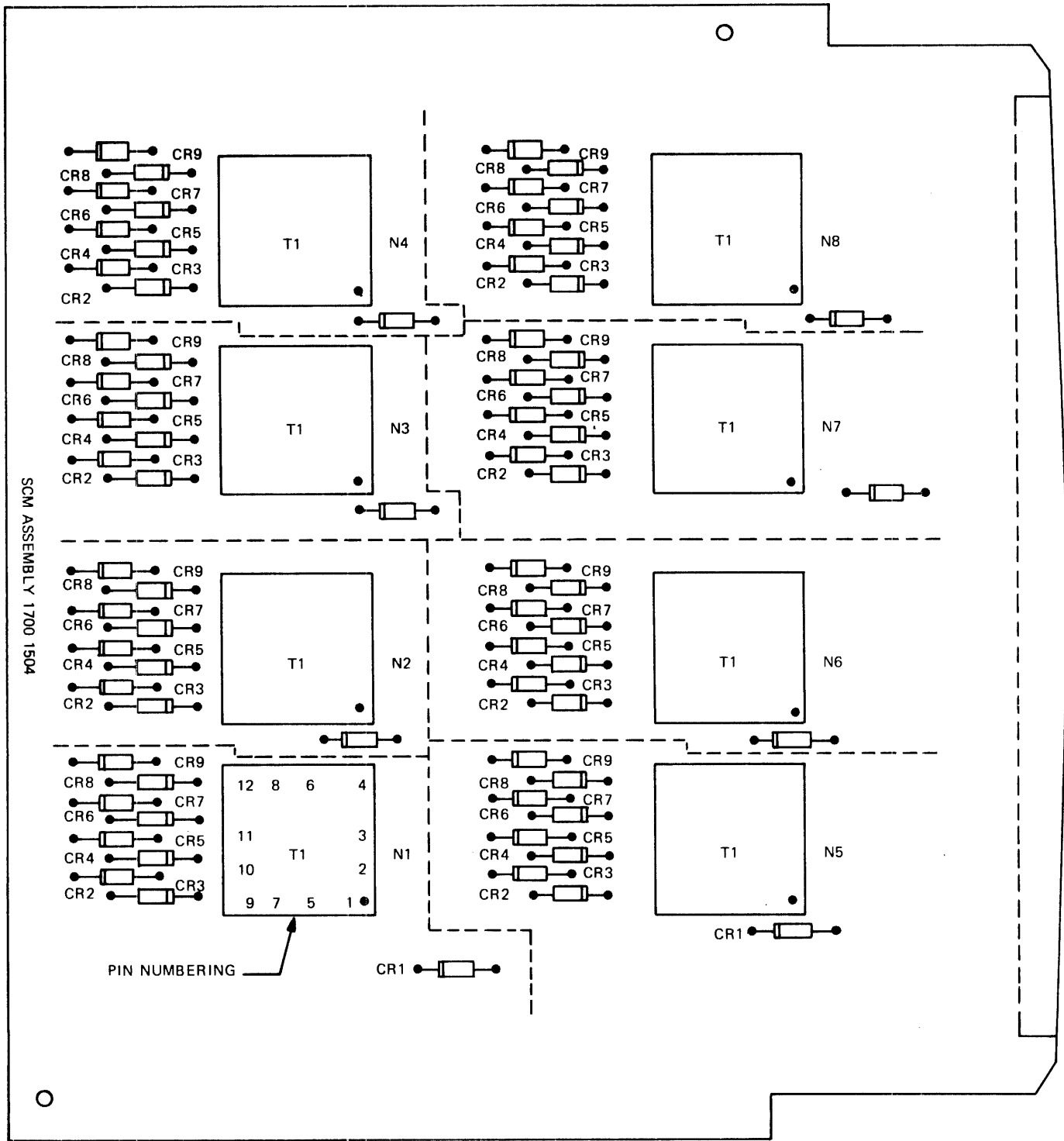


Fig. V-11

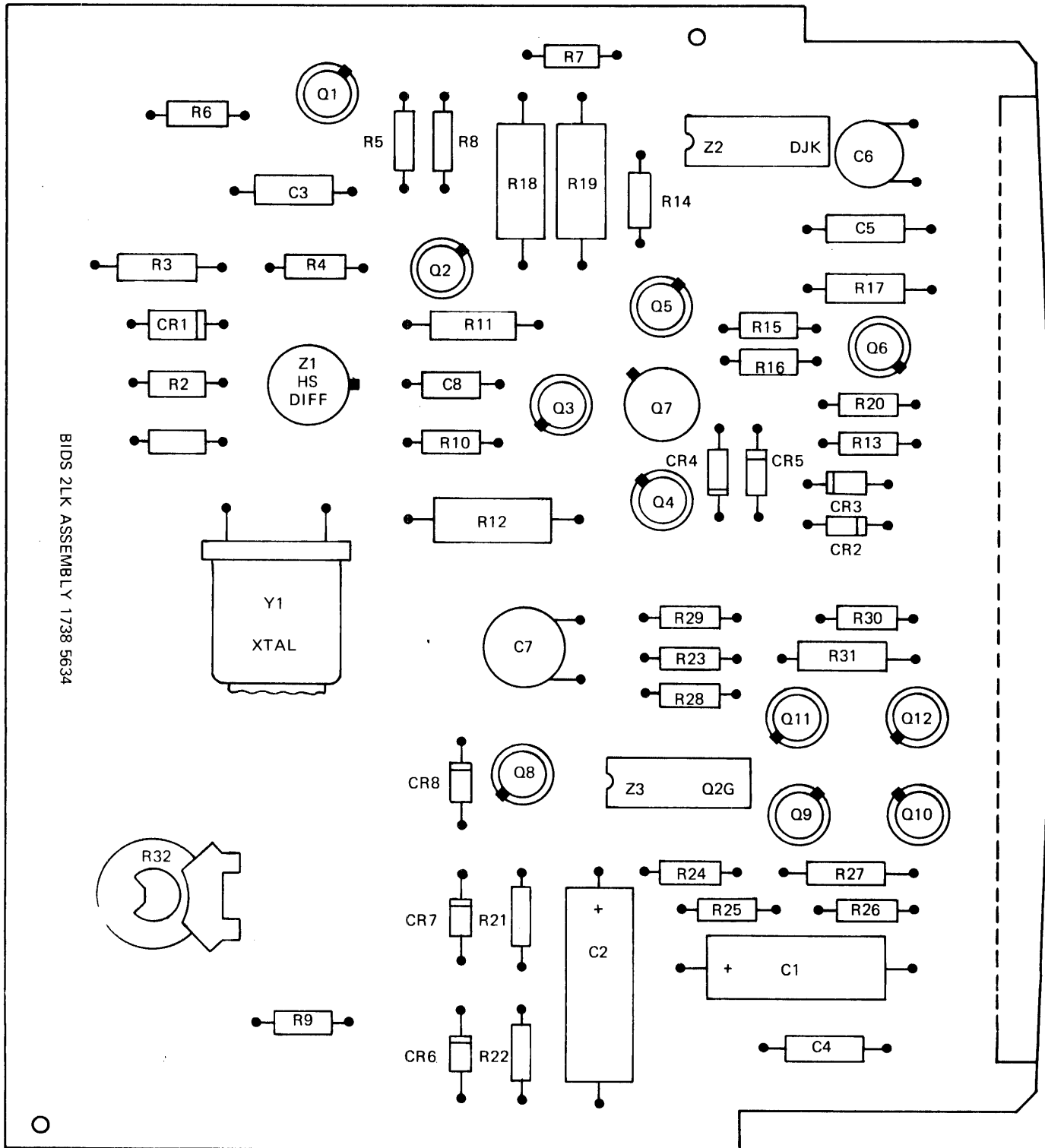


Fig. V-12

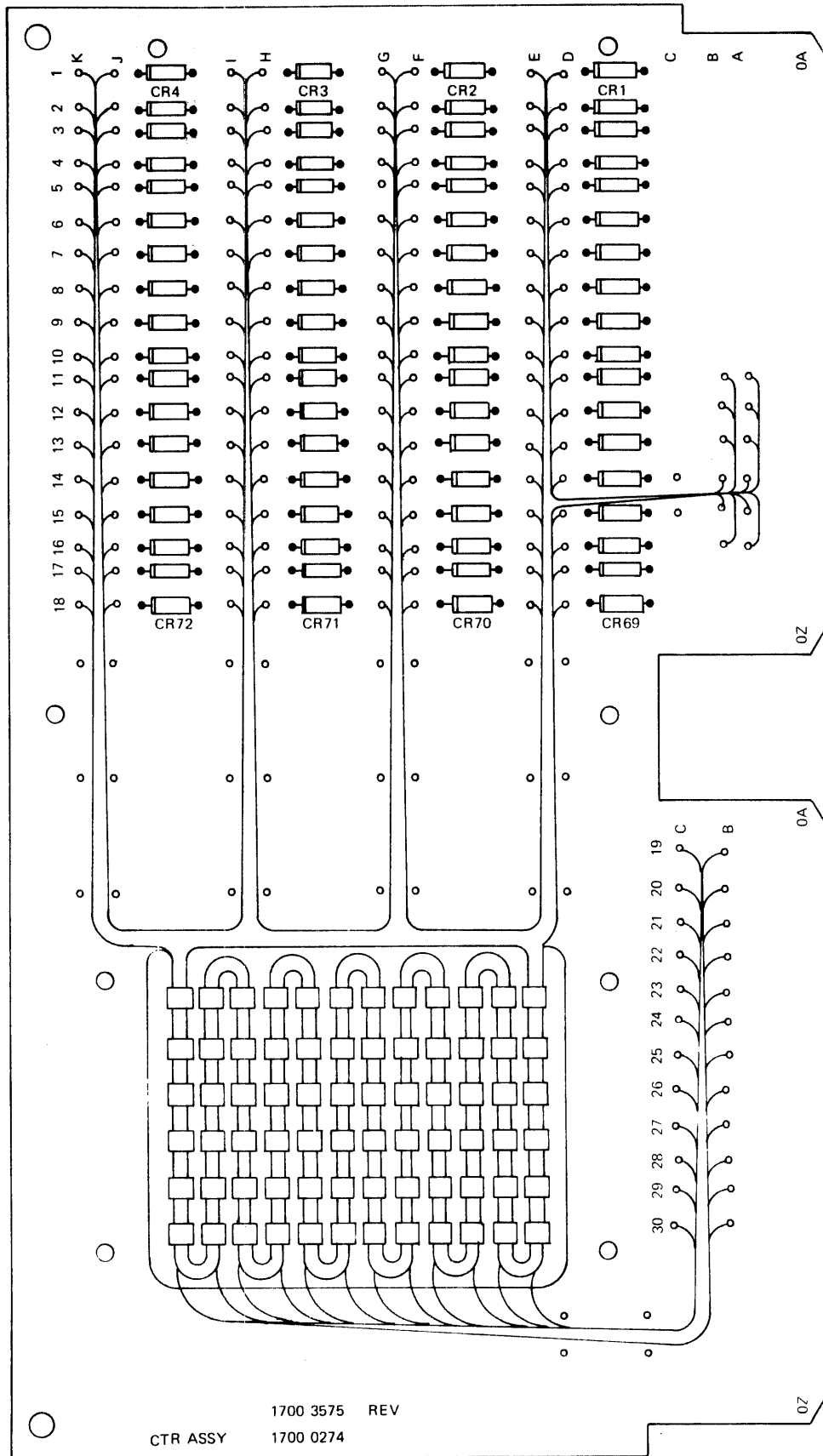


Fig. V-13

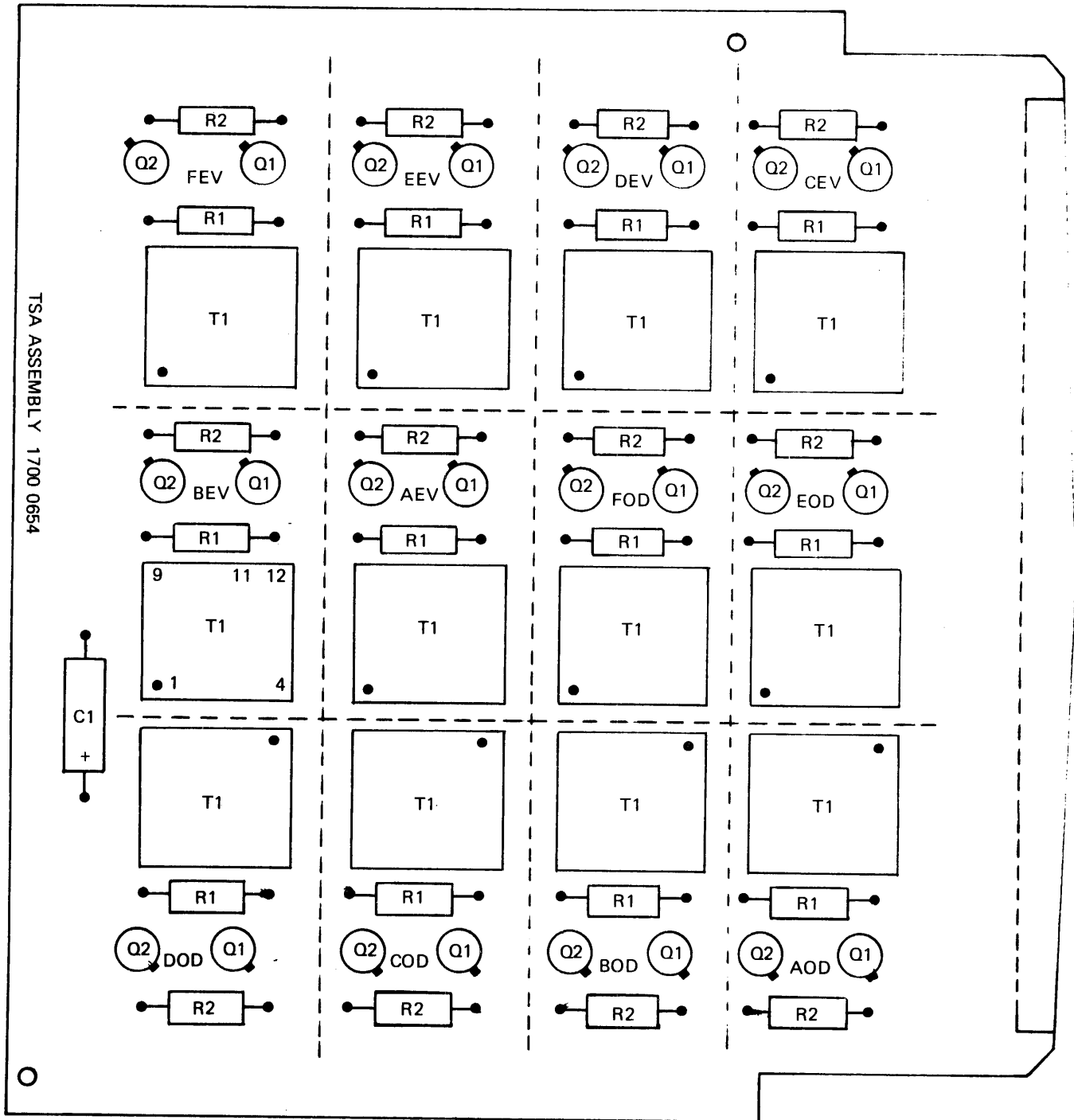


Fig. V-14

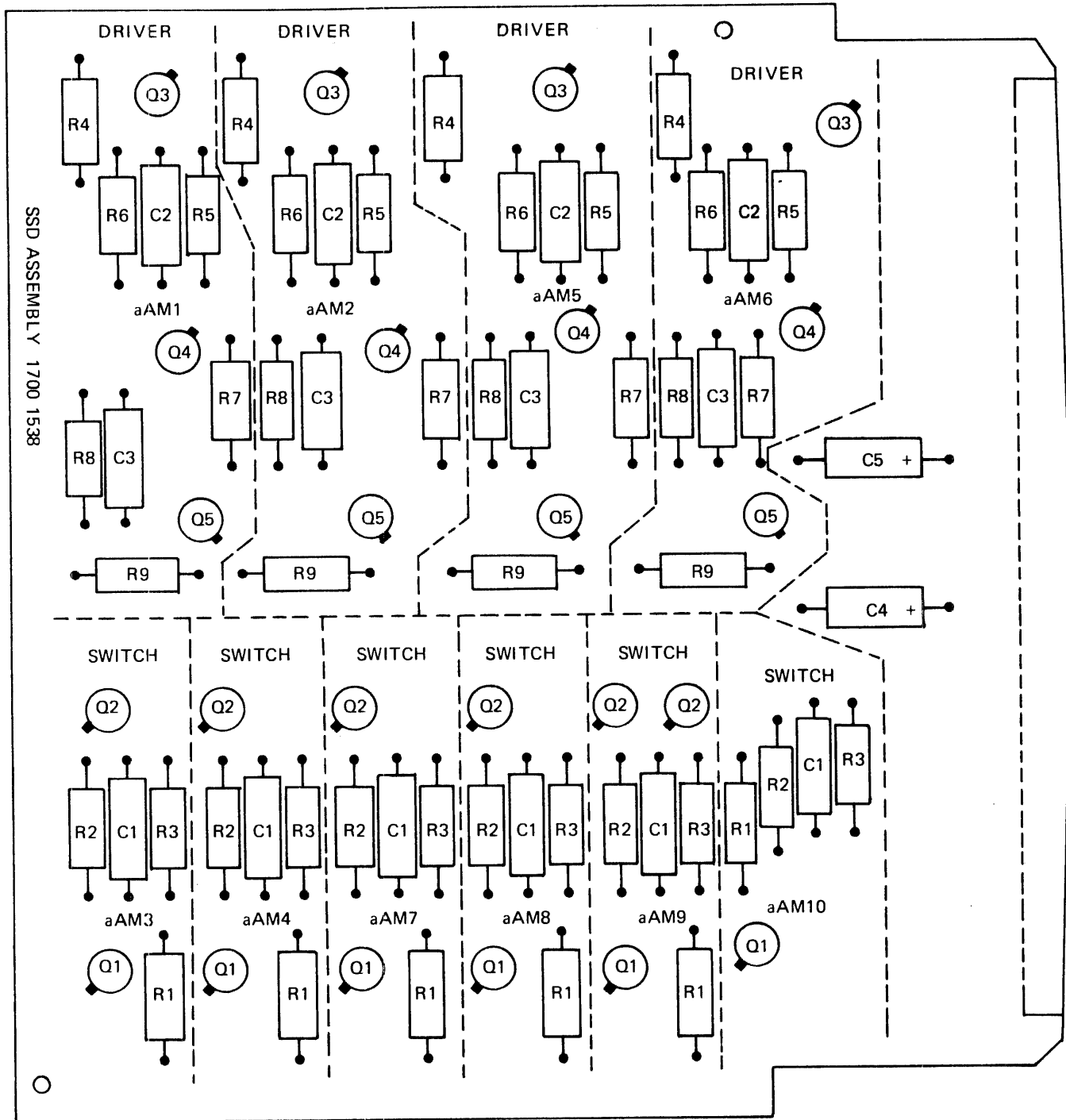


Fig. V-15



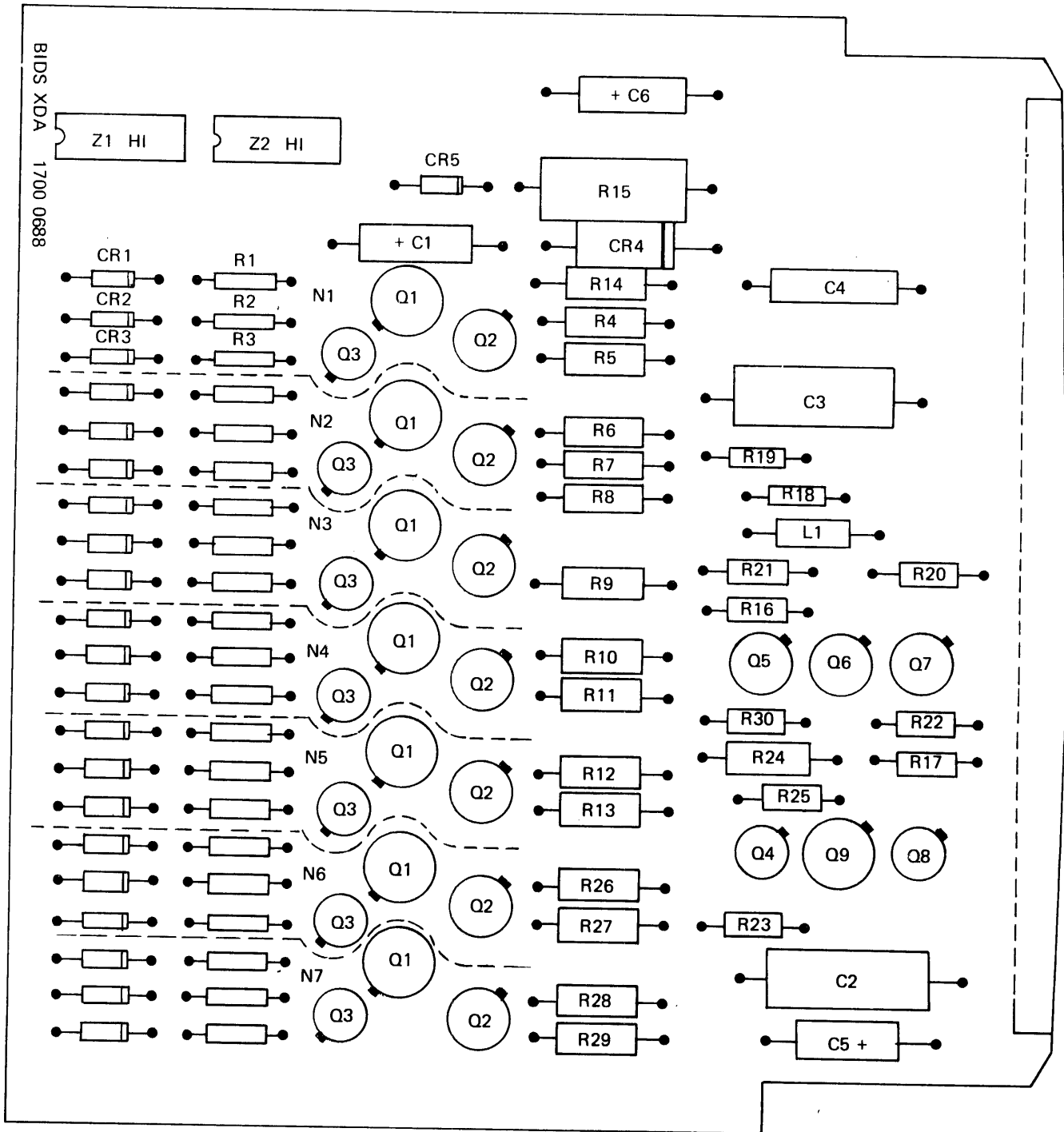


Fig. V-16

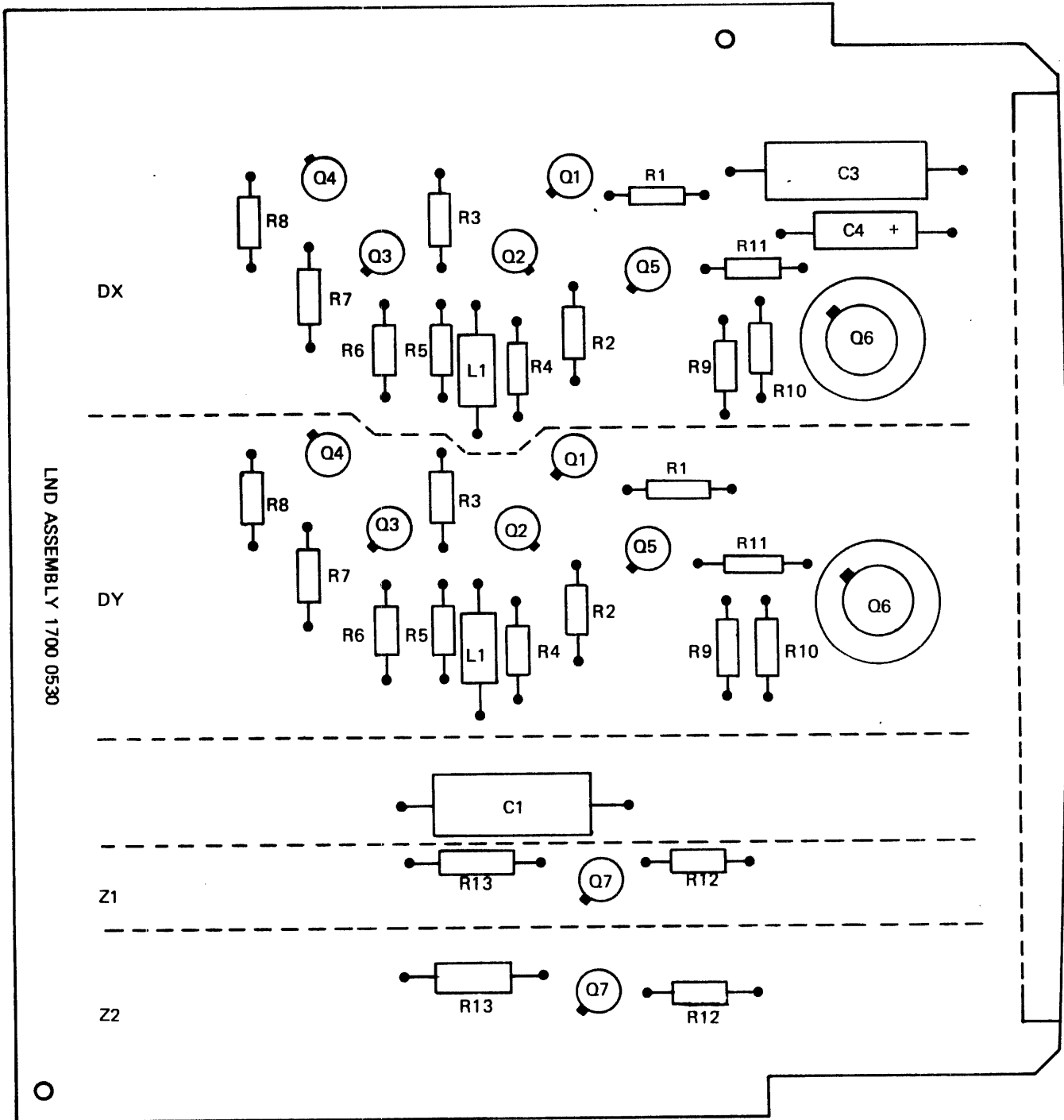


Fig. V-17

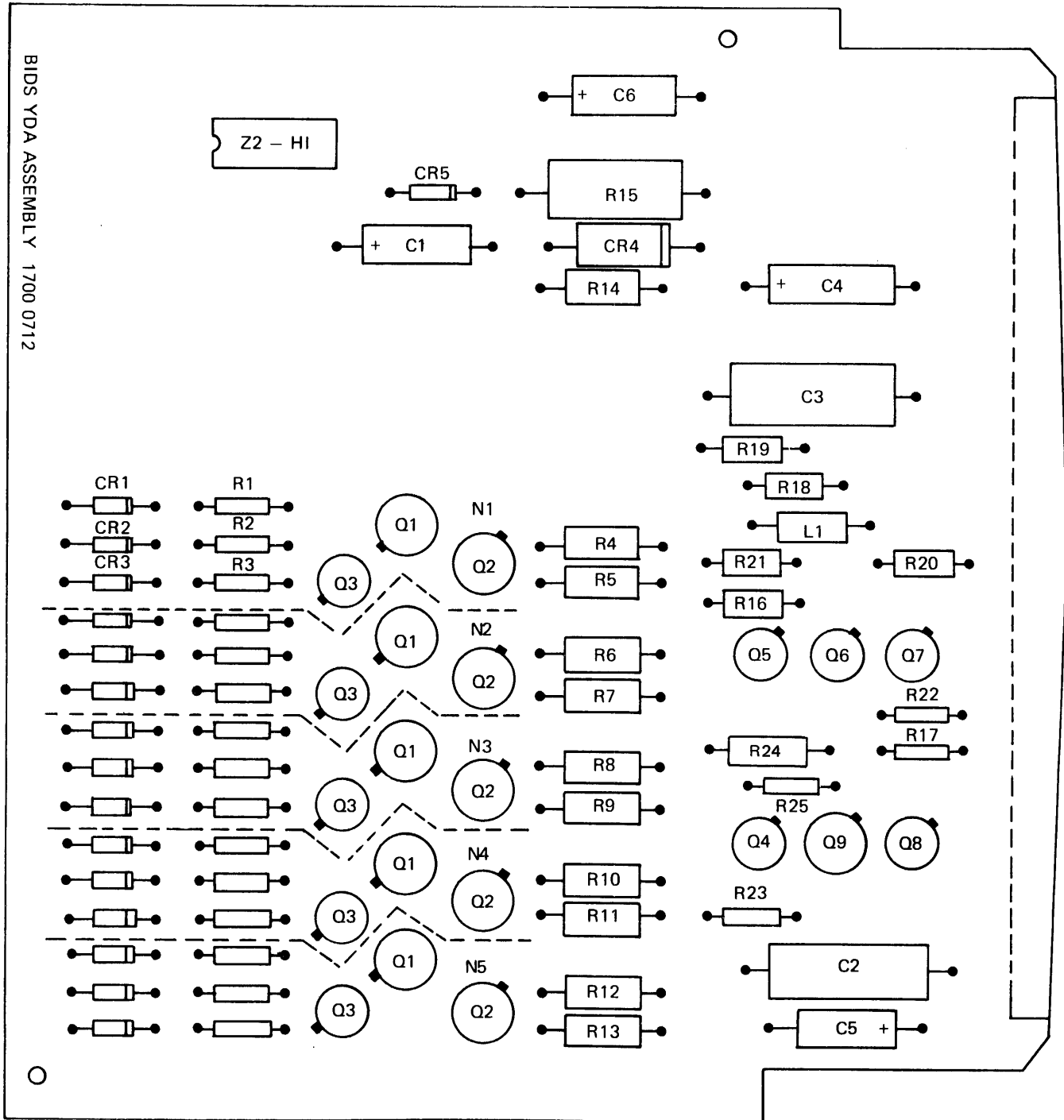


Fig. V-18

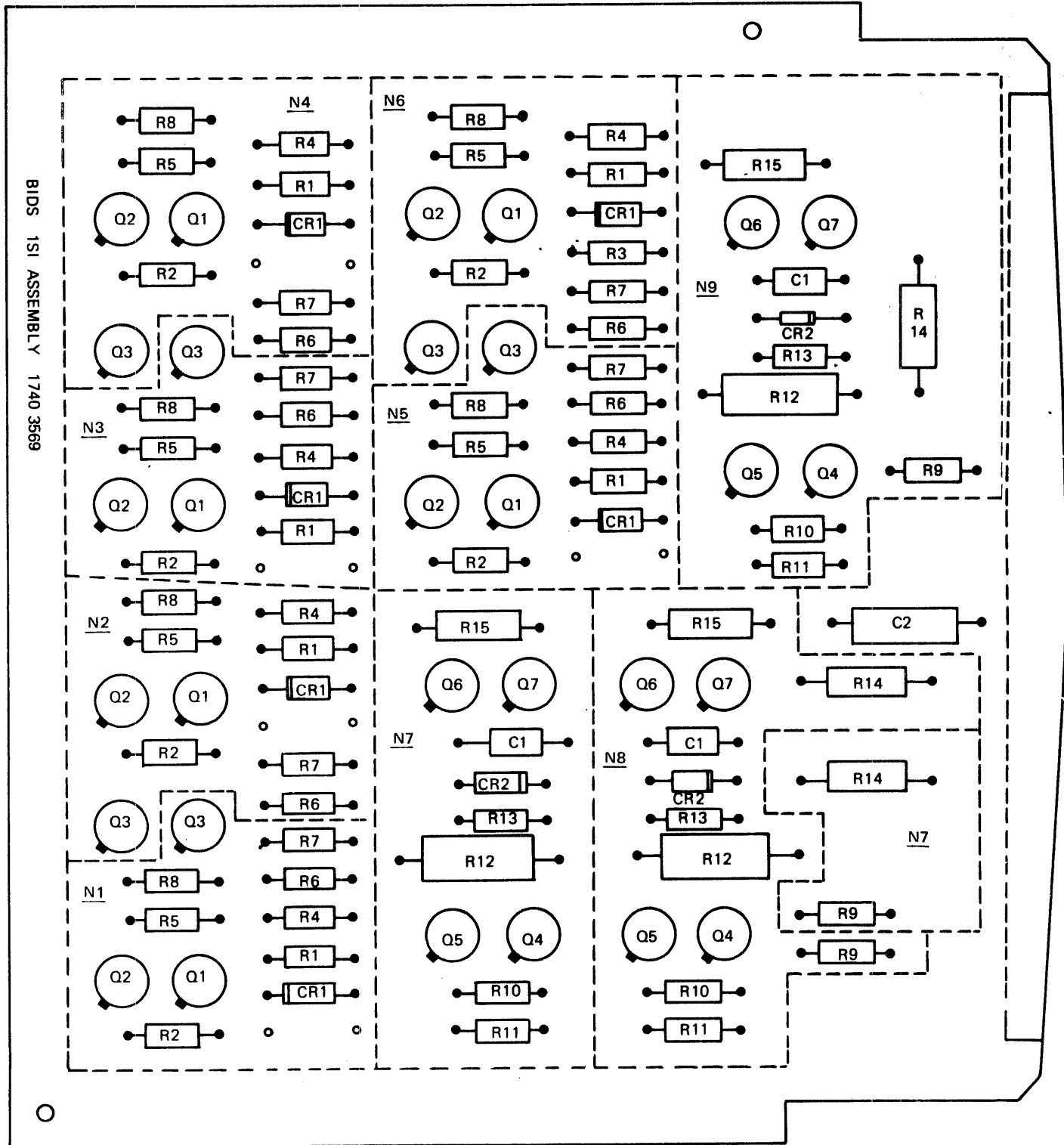


Fig. V-19

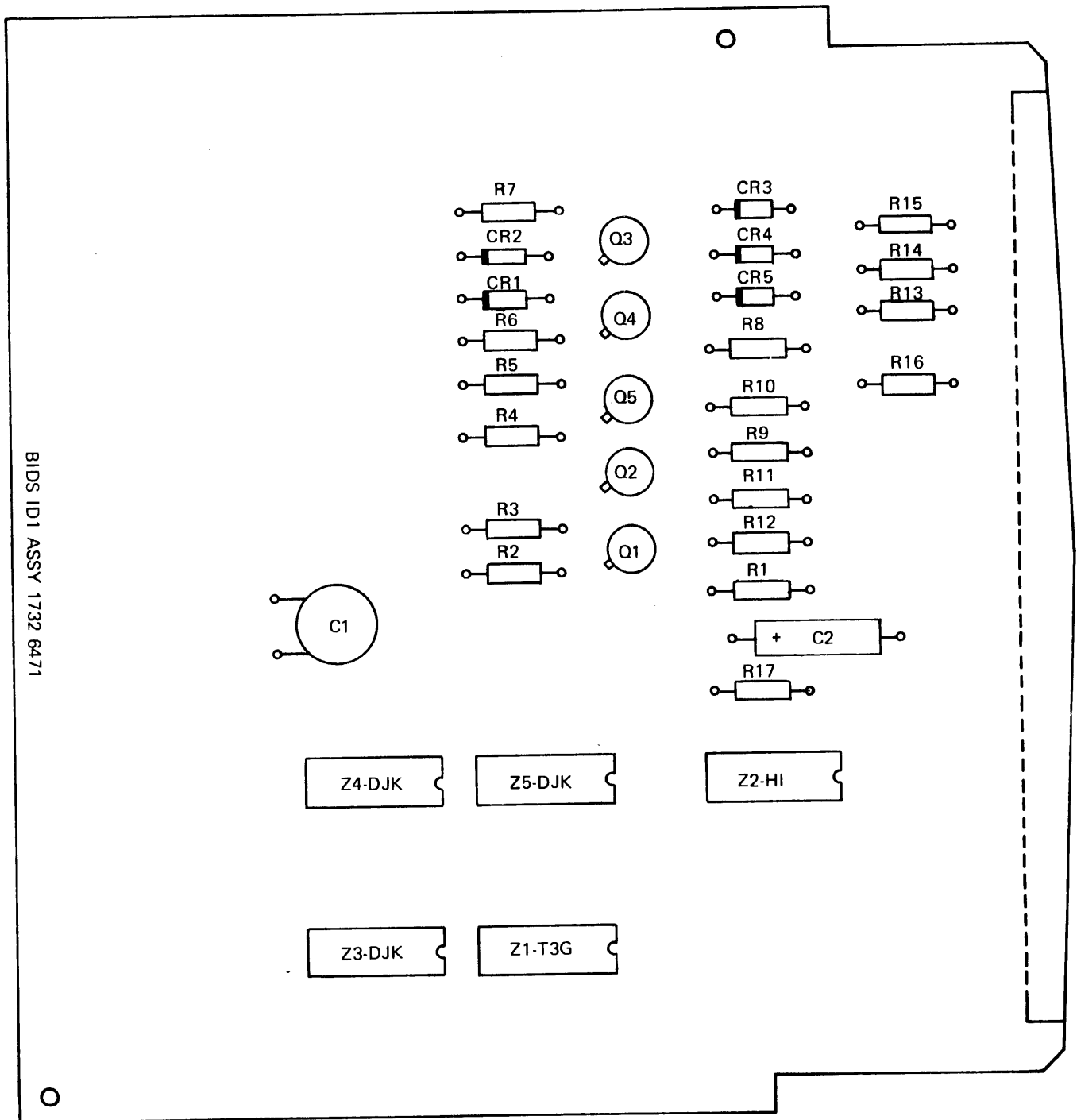


Fig. V-20

## FAULT ISOLATION PROCEDURE FOR B9353 DEFLECTION SYSTEM

### PURPOSE

The electromagnetic deflection system in the B9353 monitor consists of two, separate, high gain operational amplifiers which deliver high currents to the deflection yoke. The system is equipped with protective fusing to prevent secondary component failure in the event of a deflection system malfunction. However, because of the inherent nature of the deflection circuits, a haphazard approach to circuit fault isolation can result in extensive and unnecessary damage to the deflection system. The following procedure presents a systematic and effective method of isolating circuit faults without causing additional circuit damage.

1. Never work on the monitor deflection circuits while the monitor is connected to the Control Unit. This can result in damage to the Control Unit output drivers.
2. Never replace the monitor fuses with fuses having higher than specified ratings. If you started with a

single component failure, you'll guarantee a few additional bad components and you may cause irreparable additional damage.

3. If you are not certain whether the deflection problem is in the monitor or in the Control Unit, turn off the Control Unit and turn up the monitor Brightness Control. (Not too bright or you'll burn the CRT.) If the CRT beam is positioned in the upper right hand corner of the CRT and can be moved both horizontally and vertically by adjusting the deflection amplifier centering potentiometers, the problem is NOT in the deflection amplifier circuit. Check the Control Unit monitor drivers, D/A converters or the interface cables.
4. Never replace a deflection amplifier without first following the full fault isolation procedure. If the fault happens to lie in other than the deflection amplifier assembly, you'll either destroy the new assembly or use many fuses.
5. Check to see if RIN 4187-016 is installed in the monitor. If it's not installed, put it in. It will improve the reliability of the deflection system.

---

### FAULT ISOLATION PROCEDURE

1. Turn off both monitor and Control Unit power and disconnect the monitor interface cables from the Control Unit.
2. Remove fuses F2 through F7 from the monitor.
3. Disconnect DC power input wires from both X and Y deflection amplifiers (poke-home connectors at A2E9, A2E10, A2E11 and A3E9, A3E10 and A3E11). Make certain that the disconnected wires are either covered or safely positioned to prevent a short circuit.
4. Turn on monitor AC power and measure the voltages at the following points to ground:

| <u>Terminal</u> | <u>Voltage</u> | <u>If reading is abnormal</u>                                  |
|-----------------|----------------|--|
| F2-1, F3-1      | +26 to +40 VDC | Check +28V Supply (C4, L1, CR1, T2, F1 and associated wiring). |
| F4-1, F5-1      | -26 to -40 VDC | Check -28V Supply (C7, L2, CR4, T2, F1 and associated wiring). |

5. Turn monitor power off. Check and replace fuses F6 and F7 in their respective clips. Turn power on and measure voltages on the open deflection amplifier input wires removed in Step 3 above.

| <u>Wire Terminal</u> | <u>Color</u> | <u>Voltage</u> | <u>If reading is abnormal</u>                                 |
|----------------------|--------------|----------------|---|
| A2E9, A3E9           | Red          | +23 ±3VDC      | Check +23V regulator (Q5, CR2, C5, R6 and associated wiring). |
| A2E10, A3E10         | Violet       | -23 ±3VDC      | Check -23V regulator (Q6, CR5, C8, R8 and associated wiring). |
| A2E11, A3E11         | Red          | +15 ±3VDC      | Check +15V regulator (CR3, C6, R7 and associated wiring).     |

- 6 Turn off monitor power. Measure continuity between the following points: (Q1, Q2, Q3 and Q4 are mounted on the heatsink)

| <u>Location</u> | <u>Normal</u> | <u>Abnormal</u>  |
|-----------------|---------------|--|
| Q1B to A2E7     |               |  |
| Q2B to A2E6     | Less than     | Open circuit. Check interconnections.                    |
| Q3B to A3E7     | 20 ohms       |  |
| Q4B to A3E6     |               |  |
| Q1E to A2E8     |               |  |
| Q2E to A2E8     | Continuity    | Open Circuit. Check interconnection and yoke continuity. |
| Q3E to A3E8     |               |  |
| Q4E to A3E8     |               |  |

7. Replace fuses F2 and F5 (to the X deflection output drivers) and turn on monitor power. If fuses blow, check and replace associated X deflection power transistors (F2 - Q2, F5 - Q1). Replace fuses.
8. Turn off monitor power. Reconnect X deflection amplifier DC connections (A2E9, A2E10 and A2E11).
9. Turn on monitor power. If fuses blow, (F2, F5, F6 or F7) the X deflection amplifier is faulty. Remove and repair or replace the A2 Assembly.
10. With the monitor power on, turn up the brightness controls, R4 and R5, until a spot is just visible on the CRT. (Don't burn the CRT.) Adjust potentiometer A2R7 (X centering). If the X deflection system is functioning properly, rotation of potentiometer A2R7 will cause the CRT beam to move from the center to the right edge of the CRT. If the spot does not move, either the output transistors (Q1 or Q2) are open or the X deflection amplifier, A2, is bad. Check the output transistors first. If they check 'OK', remove, repair and/or replace the X deflection amplifier assembly.
11. Turn off monitor power. Disconnect the X deflection amplifier DC input wires (A2E9, A2E10 and A2E11). Protect these loose wires against accidental short circuit.
12. Replace the Y deflection output fuses, F3 and F4 and turn on monitor power. If the fuses blow, check and replace the associated Y deflection power transistors (F3 - Q4, F4 - Q3). Replace fuses.
13. Turn off monitor power. Reconnect Y deflection amplifier DC connections A3E9, A3E10 and A3E11.
14. Turn on monitor power. If fuses blow (F3, F4, F6 and/or F7), the Y deflection amplifier (A3) is faulty. Remove, repair and/or replace the A3 assembly.
15. With the monitor power on, turn up the brightness controls, R4 and R5, until a spot is just visible on the CRT. Adjust the Y centering potentiometer, A3R7. If the Y deflection system is functioning properly, rotation of A3R7 will cause the CRT beam to move from the center to the top edge of the screen. If the spot does not move, either an output transistor is open (Q3 or Q4) or the Y deflection amplifier assembly, A3, is bad. Check the output transistors first. If they check 'OK', remove and repair or replace the Y deflection amplifier.
16. Turn off monitor power. Reconnect both X and Y deflection amplifier DC input wires (A2E9, A2E10, A2E11, A3E9, A3E10 and A3E11).
17. Turn on monitor power and adjust potentiometers A2R7 and A3R7 to position the CRT beam to the upper right hand corner of the CRT.
18. With Monitor and Control Unit power off, connect the Monitor cables to the Control Unit. Power up and re-adjust the deflection amplifier gain and centering controls as required. Refer to B9353 Technical Manual, Section IV for complete adjustment procedure.

TROUBLE ANALYSIS

1. For trouble analysis in local mode, remove all option cards. Refer to Card Allocation Chart.
2. Refer to Table 1 for trouble analysis to a card.

NOTE: The probable cause section is by card location. Check the cards at additional locations

if they are of the same type as the ones listed.

3. Refer to Table 2 for trouble analysis of a card to a possible chip.

TABLE 1

Backline (BL)

| <u>Error</u>                    | <u>Probable Cause</u>        |
|---------------------------------|------------------------------|
| Backline goes to EOS:           | 116                          |
| Backline fails:                 | 110, 121, 123, 205, 220, 312 |
| Backline blanks screen:         | 116, 220                     |
| Backline deletes data:          | 114, 116, 220, 232           |
| Backline moves data:            | 121                          |
| Backline deletes line:          | 108                          |
| Backline changes data:          | 110                          |
| Backline gives character 9:     | 105                          |
| Backline fails after VTAB:      | 105                          |
| Backline over data fails:       | 121                          |
| Backline fails after page full: | 121                          |

Backspace (BS)

| <u>Error</u>                   | <u>Probable Cause</u> |
|--------------------------------|-----------------------|
| Backspace goes to position 80: | 116                   |
| Backspace deletes data:        | 114, 108              |
| Backspace fails:               | 223, 301, 312         |
| Backspace changes data:        | 312                   |
| Backspace over tab fails:      | 110                   |
| Backspace fails in forms:      | 224                   |
| Double backspace:              | 220                   |

Character and Line Insert/Delete

| <u>Error</u>                            | <u>Probable Cause</u>             |
|---|-----------------------------------|
| Line delete fails:                      | 108, 109, 110, 115, 116, 121, 218 |
| Character insert fails:                 | 107, 108                          |
| Line delete blanks screen:              | 108, 215                          |
| Line insert fails:                      | 108, 205, 223                     |
| Character delete fails:                 | 205                               |
| Line delete fails on screen 1, 2 and 3: | 218                               |
| Depressing space bar — inserts spaces:  | 113                               |



TABLE 1 (continued)

Cursor

| <u>Error</u>                         | <u>Probable Cause</u>  |
|--------------------------------------|--|
| No cursor movement:                  | 101, 109, 116, 119, 203, 223, 227  |
| Does not update:                     | 109, 117, 124, 203, 212, 214, 315, 319   |
| Does not home:                       | 134, 203, 224, 235, 312  |
| Distorted cursor:                    | 107, 112, 126, 128, 131, 203, 222, 227, 228, 231   |
| Double cursor:                       | 117, 118, 201, 227   |
| No cursor displayed:                 | 112, 116, 117, 118, 122, 126, 134, 135, 203, 205, 216, 222,<br>223, 226, 233, 234, 235, 302, 320, H.V.P.S. |
| Loads data within cursor:            | 119, 120, 130, 205, 217  |
| Backspaces two characters at a time: | 220  |
| Loads cursor randomly:               | 122, 216, 227  |
| Cursor blanks data:                  | 228  |
| Intermittent cursor:                 | 105, 107, 122, 201, 217, 227   |
| Loads data — no cursor displayed:    | 203, 222   |
| Cursor in top right-hand corner:     | 301, 303   |
| Cursor blanks on NL:                 | 122  |
| Cursor always blinks:                | 206  |
| Cursor does not move:                | 116  |

Display

| <u>Error</u>                                | <u>Probable Cause</u>                            |
|---|--|
| Display is intermittent:                    | 112, 124, 131, 134, 201, 203, 215, 222, 302, 306 |
| Gaps in symbols:                            | 129, 307   |
| Display shifted to right:                   | 301, 302   |
| Symbols distorted:                          | 103, 126, 128, 129, 222, 227, 229, 231, 307      |
| No horizontal count past 64:                | 202, 301   |
| No horizontal count:                        | 121, 202, 203, 205, 301, 302                     |
| Characters will not load:                   | 125, 126, 206, 214, 224, 229, 312, 329, H.V.P.S. |
| Characters change:                          | 122, 223, 232, 322                               |
| Loads double character:                     | 109, 125, 223                                    |
| Only loads first line:                      | 122  |
| Display overlaps:                           | 301  |
| Loads two screens:                          | 113, 218, 227                                    |
| Loads wrong data:                           | 114, 119, 121, 130, 213, 231, 310, 312           |
| No horizontal count past 12:                | 217  |
| Will not load full memory:                  | 104, 112, 123, 124, 220                          |
| No vertical count:                          | 202, 302, 306                                    |
| Loads line 2 on line 1:                     | 202  |
| Loads full screen — no ES:                  | 109, 118, 121, 214                               |
| Won't clear memory:                         | 122  |
| Screen 0 and 3 interact:                    | 113  |
| Screen 0 and 2 interact:                    | 113  |
| Display blinks:                             | 206  |
| U.S. will not display:                      | 224  |
| Displays data from wrong section of memory: | 118  |
| HP65 or 75 distorted:                       | 301  |
| Displays same character over entire screen: | 215  |

TABLE 1 (continued)

DISPLAY (continued)

|   |  |
|---|--|
| Won't load data from keyboard:                  | 113, 114, 118, 122, 123, 125, 203, 205, 215, 221, 223, 233, 324, 327 |
| Won't space over data:                          | 223  |
| Pull by one replace fails:                      | 116  |
| Loads only escape character:                    | 228  |
| Screen Ø displays data from screens 1, 2 and 3: | 227  |
| No horizontal count past 3:                     | 118  |
| No display after break key is depressed:        | 112  |

Input Error

| <u>Error</u>                    | <u>Probable Cause</u> |
|---------------------------------|-----------------------|
| Error light stays on:           | 222, 317, 318, 324    |
| Input error light-data correct: | 212                   |
| Input error won't reset:        | 324                   |

Forms

| <u>Error</u>             | <u>Probable Cause</u> |
|--------------------------|-----------------------|
| Forms fails:             | 107                   |
| Alterable forms fails:   | 215, 224              |
| Cursor jumps in forms:   | 224                   |
| Unalterable forms fails: | 107                   |

Indicators

| <u>Error</u>                     | <u>Probable Cause</u> |
|----------------------------------|-----------------------|
| Page full always on:             | 117, 123, 214         |
| Receive alarm and retransmit on: | 318                   |
| Keyboard light on:               | 110, 310, 326, 327    |
| No keyboard lock light:          | 117, 327              |
| Page full on after break key:    | 122                   |

Miscellaneous

| <u>Error</u>           | <u>Probable Cause</u> |
|------------------------|-----------------------|
| No POC:                | 221                   |
| Retransmit always set: | 327                   |
| Shift always on:       | 114, 310, 312         |
| Shift fails:           | 114, 119, 310, 401    |
| Repeat function fails: | 201                   |
| Clear home fails:      | 115, 224              |
| Break function fails:  | 122                   |

TABLE I (continued)

New Line (NL)

| <u>Error</u>                                | <u>Probable Cause</u>   |
|---|---|
| New Line at position 80 bad:                | 214   |
| New Line fails:                             | 110, 111, 119, 122, 124, 202, 203, 205, 215, 216, 217, 219,<br>220, 301, 302, 306, 312, 318 |
| Automatic New Line fails:                   | 121, 203, 301, 306  |
| New Line loads 2 characters:                | 121   |
| New Line goes to VTAB:                      | 216   |
| No New Line after line delete:              | 108   |
| New Line blanks screen:                     | 122   |
| New Line fails after backspace or backline: | 203   |
| New Line with forms fails:                  | 107, 224  |
| New Line repeats:                           | 135, 220  |
| New Line fails after page full:             | 220   |

Receive

| <u>Error</u>                            | <u>Probable Cause</u>  |
|---|--|
| Screen blanks on receive:               | 111  |
| Won't receive:                          | 111, 116, 125, 203, 208, 210, 211, 212, 219, 311, 314, 315,<br>317, 318, 319, 320, 324 |
| Cursor won't update on receive:         | 111, 219, 315  |
| No receive alarm:                       | 210, 327   |
| Input error on receive:                 | 212  |
| Always in receive mode:                 | 315, 317, 321, 327   |
| Receive alarm always on:                | 207, 310, 318, 324   |
| Receive alarm during XMIT:              | 321  |
| Receive alarm - all screens:            | 207  |
| Receive alarm fires at POC:             | 324  |
| Won't receive more than one message:    | 215, 219   |
| Receive alarm screen $\emptyset$ fails: | 321  |
| Won't receive New Line:                 | 219  |

TABLE 1 (continued)

Tab – Fixed Tab (FT), Variable Tab (VTAB)

| <u>Error</u>                                     | <u>Probable Cause</u>                            |
|--|--|
| VTAB causing smearing:                           | 121  |
| VTAB fails:                                      | 103, 104, 105, 107, 108, 110, 115, 121, 201, 202 |
| Fixed tab fails:                                 | 105, 107, 110, 113, 121, 202, 215, 220, 224, 329 |
| All tabs fail:                                   | 215  |
| VTAB sluggish:                                   | 215  |
| No tab or GS:                                    | 312  |
| Insert tab fails:                                | 108, 114, 115, 205, 214, 223                     |
| Insert tab blanks memory:                        | 119  |
| VTAB moves data:                                 | 103, 121   |
| VTAB set fails:                                  | 103, 104, 105, 201, 202, 215                     |
| Fixed tab increments by 5:                       | 103  |
| Fixed tab loses cursor:                          | 202  |
| Fixed tab gives new line:                        | 110  |
| Fixed tab moves data:                            | 110, 224   |
| No fixed tab without data:                       | 214  |
| VTAB off by 2 counts:                            | 105  |
| Fixed tab goes past HP-80:                       | 214, 223   |
| Fixed tab to HP-10 (ZD3):                        | 202  |
| Tab delete fails:                                | 108, 121, 205, 214                               |
| Tab delete — delete data:                        | 218  |
| VTAB in forms fails:                             | 107  |
| Fixed tab at HP50 only:                          | 103  |
| Fixed tab fails beyond last displayed character: | 110, 223   |

Transmit

| <u>Error</u>               | <u>Probable Cause</u>                  |
|----------------------------|--|
| Won't XMIT:                | 208, 209, 211, 212, 310, 314, 318, 319 |
| Retransmit light stays on: | 324, 325, 326                          |

TABLE 2

Backline/Backspace

| <u>Error</u>               | <u>Probable Cause</u> |
|----------------------------|-----------------------|
| Backline moves data:       | 121-ZB0               |
| No backspace:              | 223-ZC1               |
| Backline blanks data:      | 220-ZB2; 232-ZA2      |
| Double backspace:          | 220-ZD0, ZB2, ZC1     |
| No backline or new line:   | 220-ZB2, ZD0          |
| No backline:               | 110-ZD1; 312-ZD2, ZE2 |
| Backspace deletes at HP80: | 108-ZC0, ZD0          |

Cursor

| <u>Error</u>                      | <u>Probable Cause</u>                            |
|-----------------------------------|--|
| Distorted cursor:                 | 222-ZC2; 228-ZC3, ZE0                            |
| Random cursor:                    | 118-ZD0; 216-ZD0; 227-ZA3                        |
| No cursor movement:               | 109-ZE1; 223-ZE2; 227-ZD2                        |
| No cursor displayed:              | 117-ZD1; 118-ZD0; 122-ZE1; 203-ZA1, ZD1; 222-ZC1 |
| No cursor greater than HP1:       | 119-ZE2  |
| No cursor, RS, US or GS:          | 224-ZE0  |
| No home cursor:                   | 203-ZD1  |
| No cursor update after RCV:       | 203-ZE0  |
| Cursor in top right corner:       | 303-QO6  |
| Cursor blanks data:               | 228-ZA0  |
| Double cursor:                    | 118, ZA3, ZC3                                    |
| No cursor count greater than 127: | 217-ZD2, ZE2                                     |

Display

| <u>Error</u>                              | <u>Probable Cause</u>   |
|---|---|
| Loads double characters:                  | 109-ZD3; 223-ZD1  |
| No space or ES:                           | 121-ZA2, ZB0  |
| Missing symbols:                          | 126-QO3; 228-ZA0, ZE0   |
| No display:                               | 109-ZA0; 114-ZB0; 117-ZD1, ZD3, ZE1; 118-ZA0, ZC0, ZD0, ZD2; 121-ZA2; 122-ZA3, ZD0; 125-ZE3; 132-QO1; 203-ZD1; 217-CO2; 218-ZE2; 221-YO1; 222-ZC1, ZE0; 228-ZE0; 229-ZC1; 233-QO2, QO5; 302-QO6 |
| All characters change to space character: | 122-ZE2   |
| Screen 0 and 2 interact:                  | 113-ZD0   |
| No horizontal count:                      | 202-ZB1; 203-ZC1  |
| Lose character strokes:                   | 222-ZC2   |
| Screen 0 and 3 interact:                  | 113-ZC0   |
| Distorted symbols:                        | 227-ZB1, ZC1, ZD2, ZE1, ZB0   |
| No horizontal count past HP12:            | 217-ZB1   |
| No horizontal count past HP64:            | 202-ZA0   |
| Loads two screens:                        | 218-ZA0   |
| Characters change:                        | 322-ZB0   |
| Erratic data:                             | 226-ZE2   |
| No data past HP3:                         | 118-ZD0   |
| Loads data continuously:                  | 215-ZC0, ZA1  |
| No alpha characters:                      | 114-ZE2   |

TABLE 2 (continued)

Input Errors

| <u>Error</u>                | <u>Probable Cause</u> |
|-----------------------------|-----------------------|
| Error light on:             | 222-ZC2               |
| Input error will not reset: | 318-ZC0               |

Indicator

| <u>Error</u>             | <u>Probable Cause</u>      |
|--------------------------|----------------------------|
| Error light on:          | 222-ZC2; 324-ZC2, ZE0, ZE1 |
| Page full always on:     | 214-ZB1                    |
| No keyboard lock light:  | 117-ZD1; 327-ZE1, ZC2      |
| Keyboard light stays on: | 326-ZD0, ZD2               |

Insert/Delete Character and Line

| <u>Error</u>                                  | <u>Probable Cause</u>           |
|---|---------------------------------|
| No line delete:                               | 108-ZE3, ZC3, ZD0; 218-ZE0, ZA0 |
| Character delete -- deletes three characters: | 116-ZB0                         |

Miscellaneous

| <u>Error</u> | <u>Probable Cause</u>                |
|--------------|--------------------------------------|
| No shift:    | 114-ZD3, ZB3, ZC3, ZE3, ZC2; 401-ZG0 |

New Line

| <u>Error</u>                       | <u>Probable Cause</u>  |
|------------------------------------|--|
| New line fails:                    | 113-ZC0; 121-ZD0; 202-ZE2; 203-ZA1, ZD0, ZC0; 215-ZD0;<br>219-ZD2; 220-ZD0, ZD1; 302-QO6 |
| New line and Backline fails:       | 220-ZB2, ZD0   |
| New line repeats:                  | 220-ZB2, ZD0   |
| Automatic new line fails:          | 202-ZE2  |
| New line always equals four:       | 202-ZE1  |
| New line fails after page full:    | 217-ZE2  |
| New line fails in alterable forms: | 224-ZE0, ZE1, ZE2, ZE3   |

Receive

| <u>Error</u>                | <u>Probable Cause</u>     |
|-----------------------------|---------------------------|
| Will not receive:           | 219-ZC0; 317-ZC2; 324-ZC3 |
| No receive alarm:           | 210-ZB0                   |
| Receive alarm fires at POC: | 324-ZE0                   |
| Receive alarm clear fails:  | 324-ZB2, ZC2              |

TABLE 2 (continued)

Tab

| <u>Error</u>                   | <u>Probable Cause</u>                   |
|--------------------------------|---|
| VT set fails:                  | 103-ZE0, ZB0; 201-ZO4, ZO5              |
| No even VT:                    | 201-ZO1, ZO2                            |
| VT fails:                      | 104-ZE2, ZA2; 107-ZE0; 201-ZO4; 202-ZD3 |
| No HP5 and HP6:                | 201-ZO4                                 |
| No HP7:                        | 201-ZO5                                 |
| Tab insert fails:              | 114-ZB2; 214-ZD2; 223-AZ3               |
| FT delete fails:               | 121-ZB0; 108-ZB1                        |
| FT fails:                      | 110-ZA2, ZA3                            |
| Delete tab - deletes data:     | 218-ZD0                                 |
| No VT greater than HP73:       | 104-ZA1                                 |
| No tab greater than HP13:      | 201-ZO1                                 |
| FT at HP10 only:               | 202-ZD3                                 |
| VT fails after displayed data: | 202-ZB0                                 |

XMIT

| <u>Error</u>    | <u>Probable Cause</u>     |
|-----------------|---------------------------|
| Transmit fails: | 208-ZD0; 211-ZA1; 212-ZC1 |

Installation ProceduresINTERUNIT CABLING

Figure VI-1 illustrates the connections for the various system configurations of the Display System. A single Control Unit IV can drive up to four Keyboard/Monitor combinations. The Control Unit I is limited to a single Monitor/Keyboard.

OPTIONS

There are required and optional pluggable additions in the Control Unit. Refer to the card allocation chart located in the Test and Field documents for placement of the various pluggable assemblies. Table VI-1 is a tabulation of all pluggable options with their respective placement in the Control Unit.

A.C. Power

Control Units and Monitors are provided with power supplies to allow operation at either 50 or 60 Hertz, single phase voltages, from 100 to 240 volts.

Control Unit

Two types of Control Units are provided, 50 Hertz and 60 Hertz. The 60 Hertz Control Units are intended for Domestic use and operate on 115 volts only. A receptacle, J3, is provided in these units to allow A.C. connection of the Monitor for the Free Standing configuration. When the 60 Hertz Control Units are used in International installations, an optional external transformer, 1725 8435, is available for voltages from 100-250 volts. Connections for specific voltages are as shown on the transformer nameplate.

The 50 Hertz Control Units are intended for International use only. Voltage taps are internally available at 100, 110, 127, 200, 220, 230 and 240 volts. A single brown jumper to terminal strip TB5 is field adjustable to select any one of these voltages.

Monitor

Two types of Monitors are provided, Domestic and International. The Domestic Monitor operates on 50 or 60 Hertz, 115 volts only. The International Monitor operates on 50 or 60 Hertz and contains voltage taps on an internal transformer for operation at 100, 110, 115, 127, 200, 220, 230 and 240 volts. A single brown wire can be soldered to any of these taps and is field adjustable.

Kits

A keyboard Attachment Kit (1738 5568) is provided with each Keyboard to allow the Keyboard to be attached directly in front of the Monitor. Additionally, a Control Unit Attachment Kit (1738 5592) is available to be used in conjunction with the Keyboard Kit to form a stand-alone configuration. Instructions are included with each kit and detail the assembly and installation.

A connector kit (1726 4508) is available for connection of a Two-wire Direct Connection Interface to the Display System Control Unit. Installation instructions (1726 4516) are included with the kit and detail the assembly and wiring.

Control Unit Connector Panel

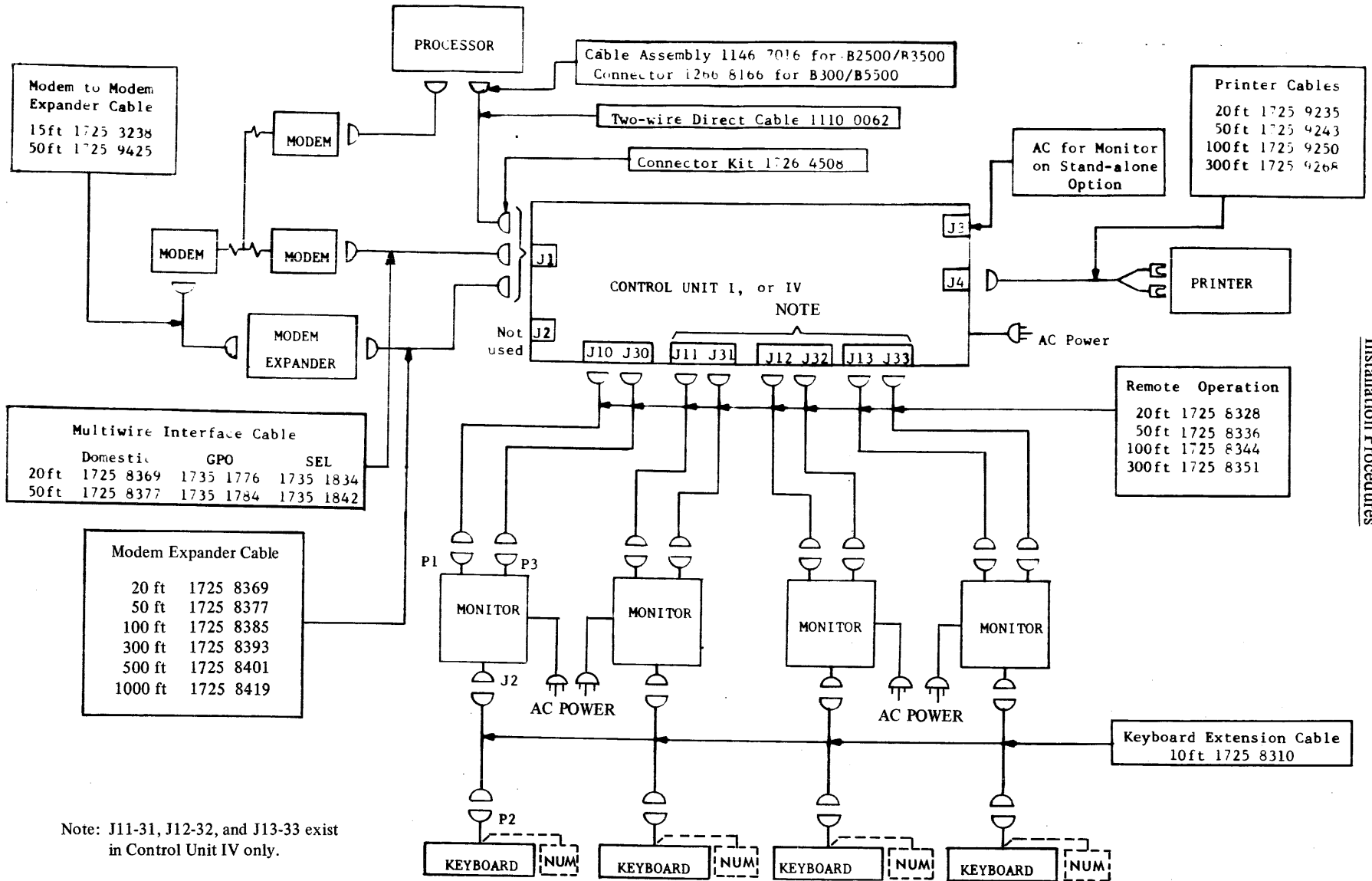
A detailed illustration of the Control Unit connector panel is shown in Figure VI-2.

IDENTIFICATION PLATES

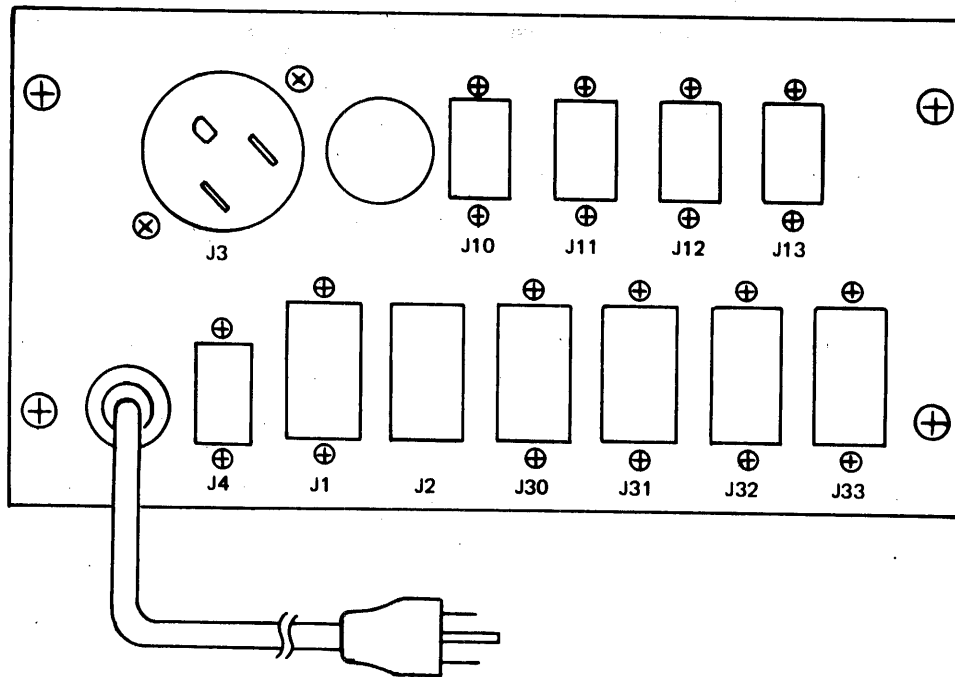
The ID plate for the Control Unit is located approximately as shown in Figure VI-3. The Monitor ID plate is located approximately as shown in Figure VI-4. The Keyboard ID plate on later units is located on the bottom approximately as shown in Figure VI-5A. Earlier units have the ID plate located approximately as shown in Figure VI-5B.



Fig. VI-1 INTER-SYSTEM CONNECTIONS



Note: J11-31, J12-32, and J13-33 exist in Control Unit IV only.

Installation Procedures**Control Unit Connector Panel**

|                |   |
|----------------|---|
| <b>J1</b>      | <b>Data Set Interface cable or Two-wire Direct Interface twisted pair cable</b> |
| <b>J2</b>      | <b>Not used</b>   |
| <b>J3</b>      | <b>AC receptacle – fused at 3A.</b>   |
| <b>J4</b>      | <b>Printer Interface cable connector</b>  |
| <b>J10-J30</b> | <b>Signal cable connectors for Monitor 0.</b>                                   |
| <b>J11-J31</b> | <b>Signal cable connectors for Monitor 1.</b>                                   |
| <b>J12-J32</b> | <b>Signal cable connectors for Monitor 2.</b>                                   |
| <b>J13-J33</b> | <b>Signal cable connectors for Monitor 3.</b>                                   |

**Fig. VI-2 CONTROL UNIT CONNECTOR PANEL**

Installation Procedures

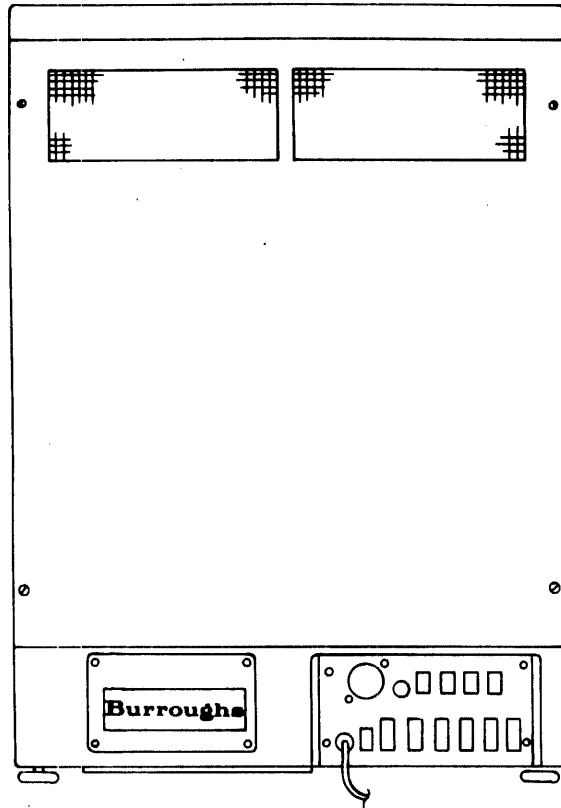


Fig. VI-3 CONTROL UNIT ID PLATE LOCATION

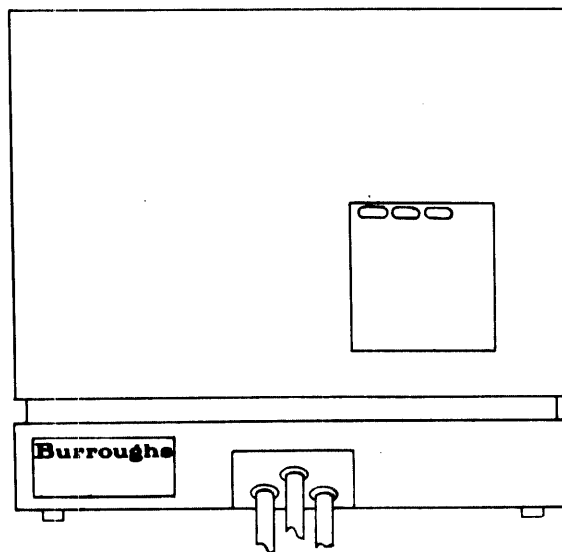


Fig. VI-4 MONITOR ID PLATE LOCATION

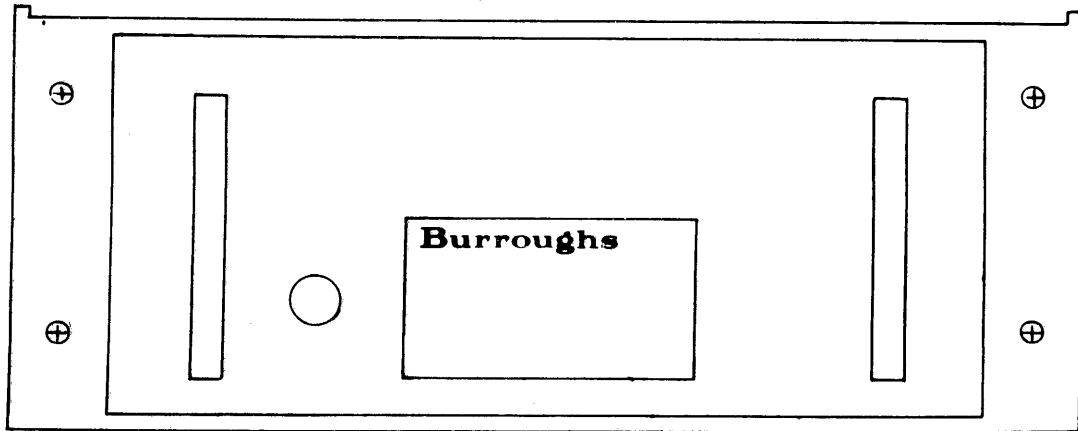
Installation Procedures

Fig. VI-5A KEYBOARD ID PLATE LOCATION (LATE)

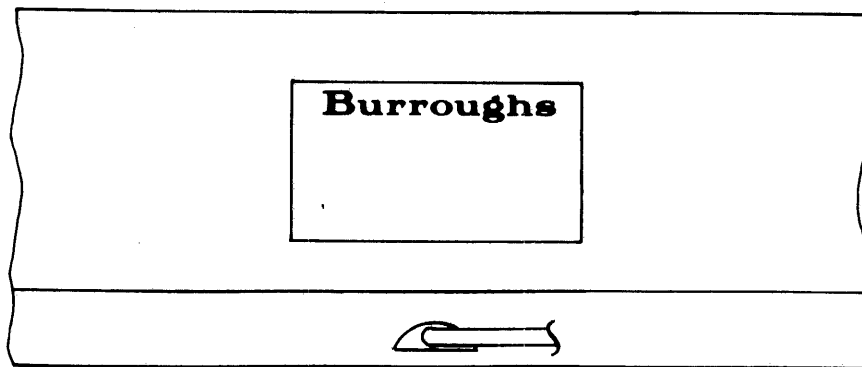


Fig. VI-5B KEYBOARD ID PLATE LOCATION (EARLY)

SPECIAL WIRING AND CABLING

For the Free Standing Configuration, cables are provided for the Keyboard/Monitor/Control Unit connections. All Other Cables Are Optional and should be ordered on Form 2280. Refer to Figure VI-1.

Up to four Keyboard/Monitors can be connected to a Control Unit IV. The connector placement and jumpers required for all memory allocations are shown in Figure VI-6.

Two kinds of addressing are provided as options: Selection Addressing and Identification Addressing. Both of these are implemented by means of jumpers. Selection Addressing jumper connections are made to Card Type n64, located at position 320, and illustrated in Figure VI-7. Factory setting for Selection Addressing will be 0, 0.

Identification Addressing jumper connections are made on Card Type n47, located at position 319, and illustrated in Figure VI-8. Factory setting for this card is also 0, 0. Note that in both cases, bits 1, 2, and 3 for the AD2 character are NOT jumpered. Bit 3 is programmatically used to activate the Print Option, when installed as part of the system. Bits 1 and 2 are handled internally to select the desired screen.

DATA RATE SELECTION

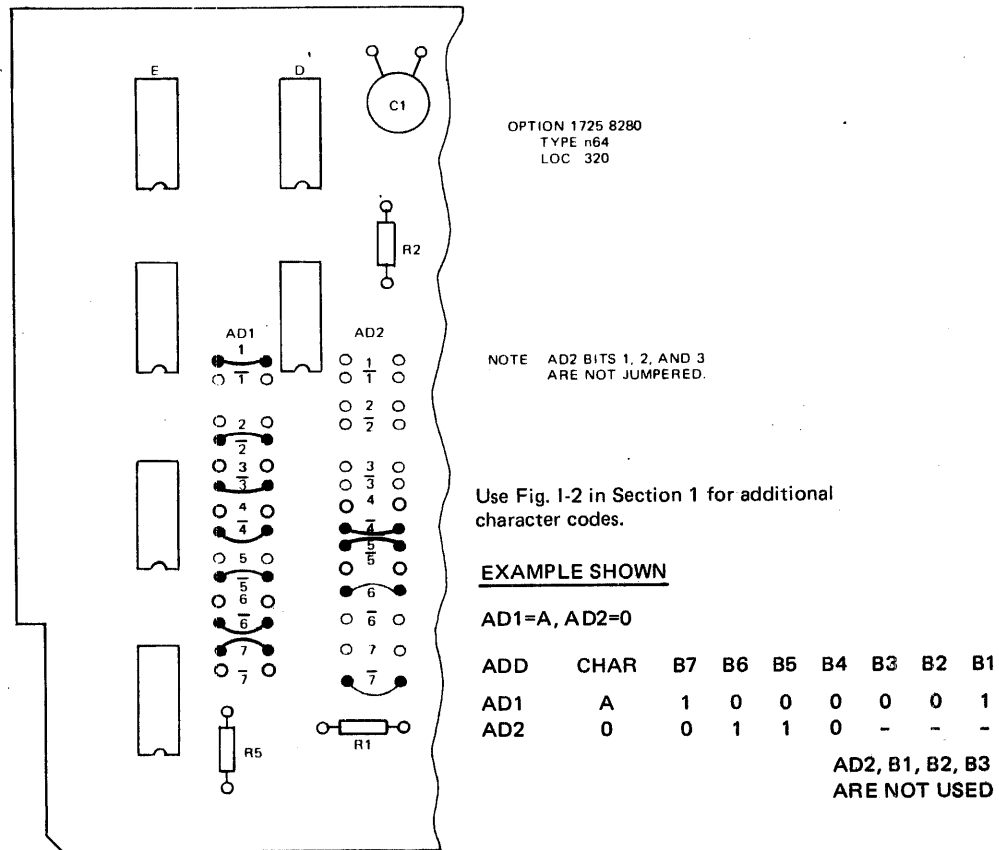
Baud rates of 150, 300, 600, 1200, 1800, 2400, 4800, 9600, 19200 and 38400 are provided for Asynchronous operation by means of jumpers. Factory setting for Asynchronous speed will be 1200 Baud. Jumper connections are made on Card Type n66, located at position 209, and illustrated in Figure VI-9.

Installation Procedures

MEMORY ALLOCATION  
(Number of Characters)

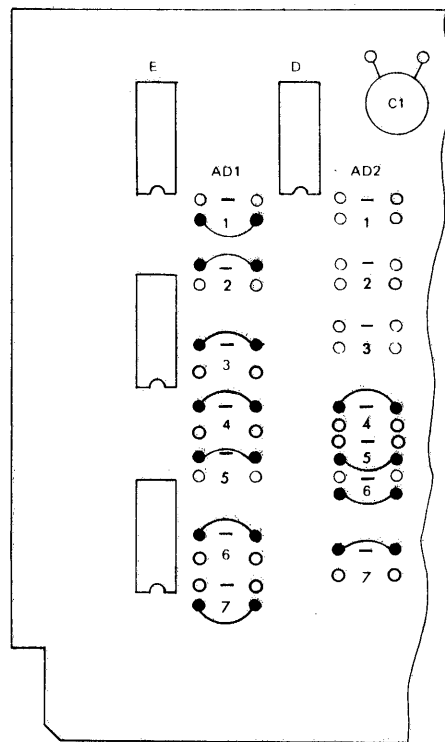
| Jumpers on<br>Card Type n10<br>Location 112 | Keyboard<br>/Monitor | Connector |        |        |        |
|---|----------------------|-----------|--------|--------|--------|
|   |                      | J10/30    | J11/31 | J12/32 | J13/33 |
| -   | (0)                  | 1000      |        |        |        |
| -   | (0)                  | 250       |        |        |        |
| E5 to E6                                    | (1)                  |           | 250    |        |        |
| -   | (0)                  | 500       |        |        |        |
| E3 to E4                                    | (0)                  |           |        | 500    |        |
| -   | (0)                  | 750       |        |        |        |
| E1 to E2                                    | (0)                  |           |        |        | 250    |
| -   | (0)                  | 250       |        |        |        |
| E5 to E6                                    | (1)                  |           | 250    |        |        |
| E3 to E4                                    | (2)                  |           |        | 500    |        |
| -   | (0)                  | 500       |        |        |        |
| E3 to E4                                    | (2)                  |           |        | 250    |        |
| E1 to E2                                    | (3)                  |           |        |        | 250    |
| -   | (0)                  | 250       |        |        |        |
| E5 to E6                                    | (1)                  |           | 250    |        |        |
| E3 to E4                                    | (2)                  |           |        | 250    |        |
| E1 to E2                                    | (3)                  |           |        |        | 250    |

Fig. VI-6 MEMORY ALLOCATION



✓ Fig. VI-7 SELECTION ADDRESS

Installation Procedures



OPTION 1725 8302  
TYPE n47  
LOC 319

NOTE AD2 BITS 1, 2, AND 3  
ARE NOT JUMPED

Use Fig. I-2 in Section 1 for additional  
character codes.

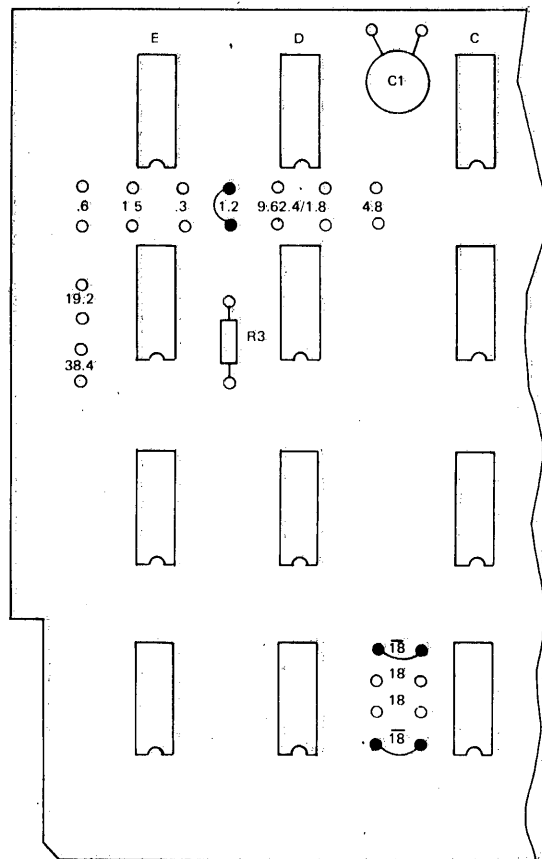
EXAMPLE SHOWN

AD1=A, AD2=0

| ADD | CHAR | B7 | B6 | B5 | B4 | B3 | B2 | B1 |
|-----|------|----|----|----|----|----|----|----|
| AD1 | A    | 1  | 0  | 0  | 0  | 0  | 0  | 1  |
| AD2 | 0    | 0  | 1  | 1  | 0  | -  | -  | -  |

AD2, B1, B2, B3  
ARE NOT USED

✓ Fig. VI-8 IDENTIFICATION ADDRESS



OPTION 1725 8195  
TYPE n66  
LOC 209

EXAMPLE SHOWN  
SPEED = 1200 BPS

NOTE: JUMPERS REQUIRED  
FOR NOT 1800 BPS

Fig. VI-9 DATA RATE SELECTION

Installation Procedures

**AUTOMATIC DISCONNECT**

To allow this option a jumper should be placed between E3 and E4 on Card Type n35, at location 208, Break and Disconnect Control. When the option is not specified, the jumper should be placed between E3 and E5. Refer to Figure VI-10.

**TWO/FOUR WIRE OPERATION**

This operation is referring to the type of telephone connection being installed. Two-wire operation provides for half-duplex communication and four-wire provides for full-duplex communication. Until such time as the memory system is changed to operate in full-duplex mode, the two-wire circuit will be used, therefore jumpers should be placed between E1 and E2 and between E6 and E7 on Card Type n35, location 208. Refer to Figure VI-10.

**SYNCHRONOUS DIRECT CONNECT**

For Direct Synchronous connection to a Processor, the sync. pulses can be eliminated by removing Card Type n36 from location 209.

**PRINTER BUFFER**

When sections of memory are assigned as Printer Buffer areas, the memory must be allocated first as in screen allocation (refer to Figure VI-6). The Printer Buffer address for area 1, 2, and 3 can be assigned by placing jumpers from Q1A to Q1B, Q2A to Q2B or Q3A to Q3B respectively, on Card Type n60 at location 333 (refer to Figure VI-11). To assign the printer buffer to area 0, no jumper is required on this card, however, to assign area 0 with 750 characters of memory, a jumper is required on Card Type n46 at location 327, between E2 and E3. The jumper should be between E1 and E2 for normal operation, (refer to Figure VI-11A).

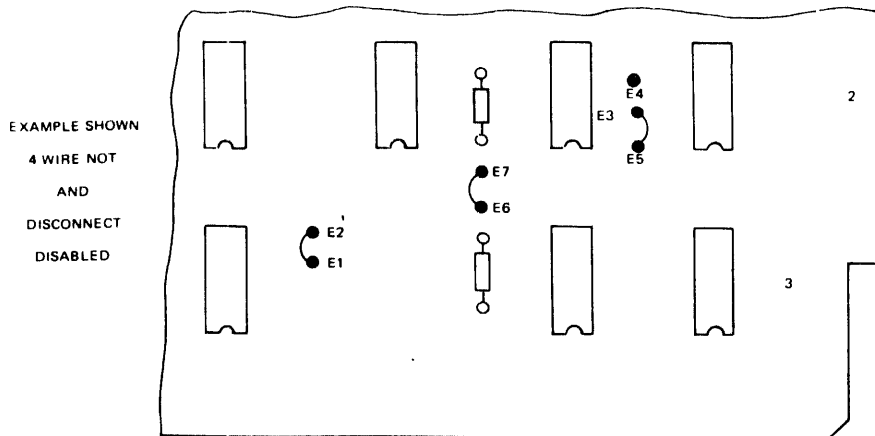


Fig. VI-10 AUTO DISCONNECT – TWO/FOUR WIRE

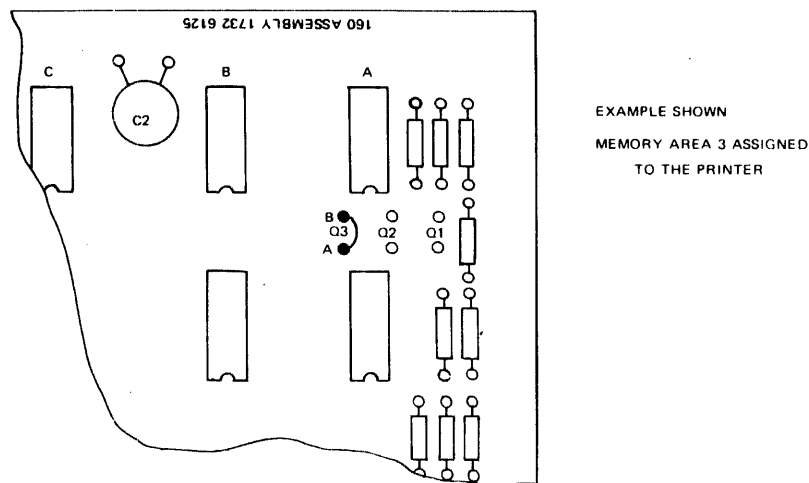


Fig. VI-11 BUFFERED PRINTER

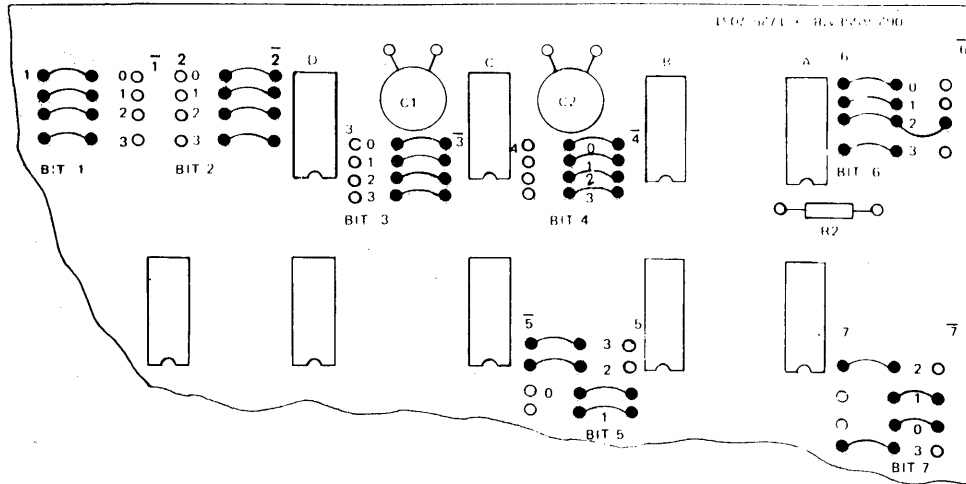
Installation Procedures

GROUP SELECT

Jumper connections are required on Card Type n62, location 323. Any character from the USASCII code chart, columns 2 through 6 can be used for the Group Select character. Figure VI-12A illustrates the jumpering for a typical Card Type 062. Figure VI-12B illustrates the jumpering for a typical Card Type 162.

PRINTER CONNECTION

✓ The modified Model 33 Teletypewriter can be connected to the Control Unit by means of the optional cables shown in Figure VI-1. At the printer end, the connections are made to terminal strip X at the rear of the unit. Pin A (PDB) of the connector is connected to Pin 6 and Pin B (+12) of the connector to Pin 7.



EXAMPLE SHOWN

| SCREEN   | CHARACTER CODE | OCTAL CODE | BINARY CODE |   |   |   |   |   |   |
|----------|----------------|------------|-------------|---|---|---|---|---|---|
|          |                |            | 7           | 6 | 5 | 4 | 3 | 2 | 1 |
| SCREEN 0 | 1              | 061        | 0           | 1 | 1 | 0 | 0 | 0 | 1 |
| SCREEN 1 | 1              | 061        | 0           | 1 | 1 | 0 | 0 | 0 | 1 |
| SCREEN 2 | A              | 101        | 1           | 0 | 0 | 0 | 0 | 0 | 1 |
| SCREEN 3 | a              | 141        | 1           | 1 | 0 | 0 | 0 | 0 | 1 |

OPTION 1725 8252  
TYPE 062 (ABOVE)  
TYPE 162 (BELOW)  
LOC 323

Fig. VI-12A GROUP SELECT

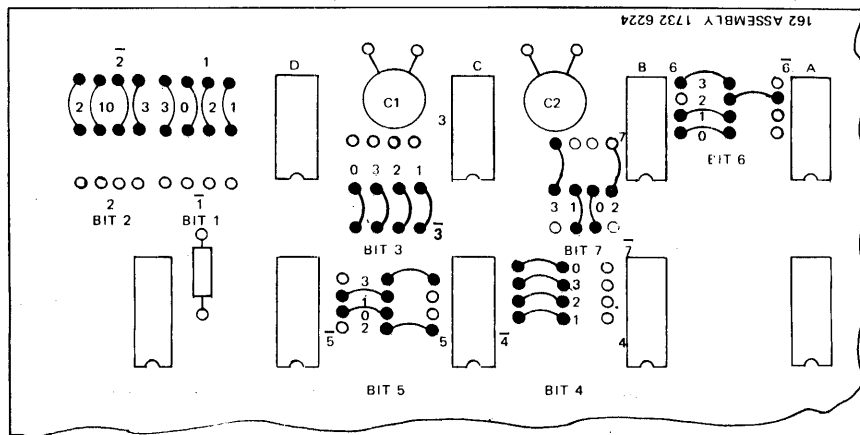


Fig. VI-12B GROUP SELECT



Installation Procedures

TABLE VI-1

| <u>Option</u>                                    | <u>M &amp; E #</u> | <u>Location</u>    |
|--|--------------------|--------------------|
| Interface, Synchronous                           | 1725 8187          | 209, 311           |
| Interface, Asynchronous                          | 1725 8195          | <u>209, 311</u>    |
| Interface, Two-wire Direct                       | 1725 8203          | 209, 311           |
| Interface, High Speed                            | 1725 8427          | 209, 311           |
| Insert/Delete                                    | 1701 4770          | 108                |
| Forms  | 1701 4788          | 107                |
| Variable Tab                                     | 1701 4796          | 103, 104, 105      |
| Paging   | 1725 8237          | 106                |
| Reposition Cursor                                | 1701 4804          | 101, 102           |
| Printer Adapter                                  | 1701 4762          | 329, 331, 333, 335 |
| Block Check                                      | 1725 8211          | 212                |
| ✓ AIC (D/L 7 and above)                          | 1743 4010          | 204, 206           |
| Fast Select Polling and Select – Fast Select (1) | 1725 8278          | 321, 319, 320      |
| Broadcast Select (2)                             | 1725 8260          | 322                |
| Group Select (3)                                 | 1725 8252          | 323                |
| Sequential Select (4)                            | 1725 8294          | 207                |
| Selection Address                                | 1725 8286          | 320                |
| Identification Address                           | 1725 8302          | 319                |
| Keyboard 1                                       | *                  | 326                |
| Keyboard 2                                       | *                  | 325                |
| Keyboard 3                                       | *                  | 324                |
| Monitor 1  | *                  | 303                |
| Monitor 2  | *                  | 304                |
| Monitor 3  | *                  | 305                |

Notes: \* Contained in Control Unit Options

1 – Includes Selection and Identification Addressing Options

2 – Requires Polling and Select – Fast Select Option

3 – Requires Broadcast Select Option

4 – Requires Group Select Option



Reliability Improvement NoticesINTRODUCTION

Although the prime function of Section VII is RIN storage, the following categories have been included to aid the Field Engineer in determining the RIN status for each unit.

1. Unit Travel Log - Used to ascertain the Summary Level (S/L) of the unit.
2. Index - RIN/EI, EWI, ECN - Used for RIN cross reference and information.
3. RIN File - Used to store copies of each RIN.

UNIT TRAVEL LOG

The unit travel log portrays the basic unit manufacturing level plus additional changes incorporated prior to final production test.

Figure VII-1 shows a typical unit travel log. Applicable information is noted in the circled Items 1 thru 6. An explanation of these circled items follows:

- Item 1. The numbers in this column are used as a cross reference between the upper and lower part of the travel log to correlate the changes with their particular top unit.
- Item 2. Identifies the unit name.
- Item 3. Top Unit Number (manufacturing number).
- Item 4. Summary Level Number (S/L) is the manufacturing level to which the unit was built. It is incremented as additional changes are incorporated into manufacturing paperwork.

NOTE

Some of these changes cannot be cross referenced to a RIN because they are for manufacturing purposes only.

- Item 5. The serial number of the unit.
- Item 6. This column shows all Engineering changes, i.e., EWI's (Engineering Wiring Instructions) and ECN's (Engineering Change Notices) made to the equipment that are installed above the indicated Summary Level. Only those EWI's or ECN's which involve field retrofit changes are listed on the EWI/RIN Index.

The Unit Travel Log should become Page 1 of each Site RIN Log.

INDEX - RIN/EI-EWI-ECN

Chart VII-1 is a cross reference for RIN's to EI's, EWI's, and ECN's. Unless the EI's, EWI's, or ECN's are logical changes they will not appear on this chart since they are not applicable to the field.

RIN FILE


A copy of each RIN or RIN cover sheet should be inserted in sequence in this section. Missing RIN's may be requested through Dearborn Distribution. The branch office should have a reference copy of each RIN.

Updates to the INDEX - RIN/EI-EWI-ECN will be issued on a quarterly basis.

Reliability Improvement Notices

1

**Burroughs Corporation**  
DEFENSE, SPACE AND SPECIAL SYSTEMS GROUP  
 PAOLI, PENNSYLVANIA 19301



2

**UNIT TRAVEL LOG**

3

SYSTEM NO. 123

4

PAGE 1 OF 1

5

| NO. | UNIT NAME              | UNIT NUMBER | DASH | SUB | D/L | S/L | JOB NO. | CUM NO. | SERIAL NO. |
|-----|------------------------|-------------|------|-----|-----|-----|---------|---------|------------|
| 1   | Control Unit I         | 1725-8500   |      |     | 6   | 5   |         |         | B01234-042 |
| 2   | Printer Adapter        | 1701-4762   |      |     | 6   | 5   |         |         | B01235-042 |
| 3   | Insert/Delete          | 1701-4770   |      |     | 6   | 5   |         |         | B01236-042 |
| 4   | Forms                  | 1701-4788   |      |     | 6   | 5   |         |         | B01237-042 |
| 5   | Variable Tab           | 1701-4796   |      |     | 6   | 5   |         |         | B01238-042 |
| 6   | Reposition Cursor      | 1701-4804   |      |     | 6   | 5   |         |         | B01239-042 |
| 7   | Two Wire Direct (TDI)  | 1725-8203   |      |     | 6   | 5   |         |         | B01222-042 |
| 8   | Selection Addressing   | 1725-8286   |      |     | 6   | 5   |         |         | B01230-042 |
| 9   | Identification Address | 1725-8302   |      |     | 6   | 5   |         |         | B01231-042 |
| 10  | Block Check            | 1725-8211   |      |     | 6   | 5   |         |         | B01240-042 |

INCORPORATED CHANGES ABOVE SUMMARY PARTS LIST LEVEL

| NO. | EI/ECN/EWI | LOC | INST | INSP | NO. | EI/ECN/EWI | LOC | INST | INSP | NO. | EI/ECN/EWI | LOC | INST | INSP |
|-----|------------|-----|------|------|-----|------------|-----|------|------|-----|------------|-----|------|------|
|     |            |     |      |      |     |            |     |      |      |     |            |     |      |      |
|     |            |     |      |      |     |            |     |      |      |     |            |     |      |      |
|     |            |     |      |      |     |            |     |      |      |     |            |     |      |      |
|     |            |     |      |      |     |            |     |      |      |     |            |     |      |      |
|     |            |     |      |      |     |            |     |      |      |     |            |     |      |      |
|     |            |     |      |      |     |            |     |      |      |     |            |     |      |      |
|     |            |     |      |      |     |            |     |      |      |     |            |     |      |      |

Unit Travel Log

Figure VII-1

Reliability Improvement Notices

CHART VII-1. INDEX RIN/EI-EWI-ECN

| RIN NO.<br>4187- | EI-ECN | RELEASE<br>DATE | INSTAL.<br>TIME IN<br>HOURS | PRE-<br>REQ. | UNIT | UNITS<br>AFFECTED | DESCRIPTION  |
|------------------|--------|-----------------|-----------------------------|--------------|------|-------------------|--|
| 001              | C73268 | 6-9-70          | 0.2                         | None         | Mon. | Below 04924       | Keyboard Cable Clearance                           |
| 002              | C73581 | 6-9-70          | 1.0                         | None         | C.U. | D/L 4 & below     | Cursor Memory Add Count<br>Correction              |
| 003              | C73205 | 6-9-70          | 1.0                         | None         | Mon. | Below 06123       | Increase Stability of Deflect.<br>Amp.             |
| 004              | C73206 | 10-6-70         | 1.0                         | None         | Mon. | All               | Character Non-Linearity<br>Correction              |
| 005              | C73925 | 10-20-70        | 1.0                         | None         | C.U. | Below 09000       | Erroneous Setting of Input<br>Err. Ind.            |
| 006              | C73842 | 11-5-70         | 1.0                         | 002          | C.U. | Below 07300       | Multiple Screen Op<br>Interference                 |
| 007              | C73848 | 11-17-70        | 2.0                         | None         | C.U. | Below 07300       | Cursor Positioning in<br>Forms Mode                |
| 008              | C73888 | 12-8-70         | 2.0                         | None         | C.U. | Below 07300       | Intermittent Send/Receive<br>Logic Correction      |
| 009              | C73719 | 12-8-70         | 1.0                         | None         | C.U. | Below 07300       | Line Deletion Correction                           |
| 010              | C73876 | 12-8-70         | 1.0                         | None         | C.U. | Below 7300        | Escape Character Transmission<br>Error             |
| 011              | C73897 | 12-8-70         | 2.0                         | None         | C.U. | Below 07300       | Circuit Card Compatibility<br>Correction           |
| 012              | C73544 | 3-4-71          | 1.0                         | None         | Mon. | All               | Heat Dissipator Short to<br>Ground on DA           |
| 013              | C73859 | 3-4-71          | 1.0                         | None         | Key. | Below 07267       | Shift Key and Marginal<br>Noise Correction         |
| 014              | C73993 | 3-4-71          | 1.0                         | None         | C.U. | Below 07400       | Improper Cursor Positioning<br>During Variable Tab |
| 015              | C73860 | 3-4-71          | 1.0                         | None         | C.U. | Below 09000       | Marginal Adj. of RTT                               |
| 016              | C73963 | 5-6-71          | 2.0                         | None         | Mon. | Below 6400        | Stabilization of Power<br>Supplies.                |
| 017              | C74033 | 7-16-71         | 1.5                         | None         | C.U. | Below 8100        | Increase Noise Immunity<br>Data Set Interface      |
| 018              | C74263 | 9-14-71         | 1.0                         | None         | C.U. | Below 14000       | Line Splitting Correction.                         |
| 019              | C73768 | 10-21-71        | 1.0                         | None         | C.U. | Below 2000        | +3.6V Power Supply<br>Improvement.                 |
| 020              | C74339 | 11-2-71         | 1.0                         | None         | C.U. | Below 14185       | Grounding of Levels DB<br>and DD                   |
| 021              | C73430 | 3-2-72          | 1.0                         | None         | Key  | Below 6000        | Non-Skid Pads for Keyboard                         |
| 022              | C74441 | 3-2-72          | 1.0                         | 019          | C.U. | Below 2000        | Compatibility of Regulator<br>Assemblies           |

|                                       |  |                            |                      |
|---------------------------------------|--|----------------------------|----------------------|
| <b>Burroughs</b><br>FIELD ENGINEERING | <b>R</b><br><b>I</b><br><b>M</b><br><b>P</b><br><b>R</b><br><b>O</b><br><b>V</b><br><b>E</b><br><b>M</b><br><b>E</b><br><b>N</b><br><b>T</b><br><b>I</b><br><b>N</b><br><b>D</b><br><b>E</b><br><b>X</b> | SYSTEM SERIES<br>B.        | <b>No.</b> R4187-000 |
|                                       |  | STYLE/MODEL<br>B9351/B9353 | PAGE<br>1 OF 1       |
| ORIGINATOR:<br>III-TREDYFFRIN         |  | TOP UNIT NO.<br>All        |                      |
|                                       |  | DATE<br>6-20-75            |                      |

| <u>R.IN NO.</u> | <u>UNITS AFFECTED</u> | <u>TITLE</u>                                      | <u>DATE</u> |
|-----------------|-----------------------|---|-------------|
| * 001           | Below 04924           | Information in Parts Catalog                      | Cancelled   |
| 002             | D/L 4 and below       | Cursor Memory Add Count Correction                | 6-09-70     |
| 003             | Below 06123           | Increase Stability of Deflection Amp.             | 6-09-70     |
| 004             | All                   | Character Non-Linearity Correction                | 10-06-70    |
| 005             | Below 09000           | Erroneous Setting of Input Error Ind.             | 10-20-70    |
| 006R            | Below S/N 6000        | Multiple Screen Operation Interference Correction | 2-16-71     |
| 007             | Below S/N 6000        | Cursor Positioning Problem in Forms Mode          | 11-17-70    |
| 008             | Below S/N 7300        | Intermittent Send/Receive Logic Corrections       | 12-08-70    |
| 009             | Below S/N 7300        | Line Deletion Correction                          | 12-08-70    |
| 010R            | Below S/N 7300        | Escape Character Transmission Error               | 1-14-71     |
| 011             | Below S/N 7300        | Circuit Card Compatibility Correction             | 12-08-70    |
| 012             | All                   | Heat Dissipator Short to Ground on DA             | 3-04-71     |
| 013             | Below 7267            | Shift Key and Marginal Noise Correction           | 3-04-71     |
| 014             | Below 7400            | Improper Cursor Positioning During Variable Tab   | 3-04-71     |
| 015             | Below 9000            | Marginal Adj. of RTT                              | 3-04-71     |
| 016             | Below 6400            | Stabilization of Power Supplies                   | 5-06-71     |
| 017             | Below 8100            | Increase Noise Immunity - Data Set Interface      | 7-16-71     |
| 018             | Below 14000           | Line Splitting Correction                         | 9-14-71     |
| 019             | Below 2000            | +3.6V Power Supply Improvement                    | 10-21-71    |
| 020             | Below 14185           | Grounding of Levels DB and DD                     | 11-02-71    |
| 021             | Below 6000            | Non-Skid Pads for Keyboard                        | 3-02-72     |
| 022             | Below 2000            | Compatibility of Regulator Assemblies             | 3-02-72     |
| 023             | All                   | Contention Mode Correction                        | 11-16-72    |
| 024             | All                   | Printing Multiple Characters                      | 11-16-72    |

|   |                                |                |                           |                                      |
|---|--------------------------------|----------------|---------------------------|--------------------------------------|
| <b>Burroughs</b><br>FIELD ENGINEERING<br>TECHNICAL OPERATIONS |                                | RELIABILITY    | SYSTEM SERIES<br>Series B | <b>No.</b> 4187-001                  |
|   |                                | IMPROVEMENT    | STYLE/MODEL<br>B9351-4    |                                      |
|   |                                | NOTICE         | TOP UNIT NO.<br>1700 1181 | UNIT DESCRIPTION<br>Monitor Assembly |
| STD. INSTALL. TIME<br>Less than $\frac{1}{2}$ hour            | UNITS AFFECTED<br>Below B04924 | DATE<br>6-9-70 |                           |                                      |
| TITLE<br>Keyboard cable clearance                             |                                |                |                           |                                      |

CONDITION: Keyboard cable is pinched by the Monitor when the cable is routed under the Monitor assembly.

CAUSE: Present bumpers (feet) on the Monitor are not of sufficient height to provide clearance for the Keyboard cable.

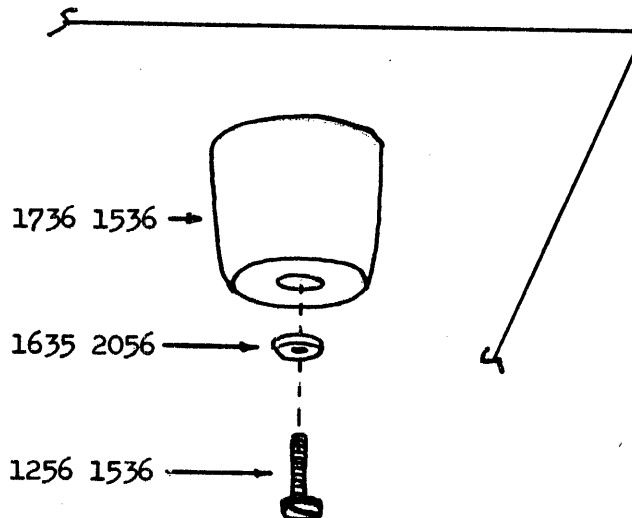
CORRECTION: Replace existing bumpers with new, larger bumpers.

PARTS REQUIRED:

| <u>Number</u> | <u>Description</u> | <u>Qty</u> | <u>Unit List Price</u> |
|---------------|--------------------|------------|------------------------|
| 1256 1536     | Screw, 10-32 1"    | 4          | \$ .06                 |
| 1635 2056     | Washer, flat       | 4          | .06                    |
| 1736 1536     | Bumper             | 4          | .06                    |

INSTRUCTIONS:

1. Remove and discard the four (4) rubber bumpers from the bottom plate of the Monitor.
2. Install the new bumper (1736 1536) using screw (1256 1536) and washer (1635 2056). Refer to Figure 1.



B9351 Technical Manual, Form 1044187, is not available at this time.

|   |                                   |   |                           |                     |
|---|-----------------------------------|---|---------------------------|---------------------|
| <b>Burroughs</b>                                |                                   | <b>R</b> ELIABILITY                       | SYSTEM SERIES<br>Series B | <b>No.</b> 4187-002 |
|   |                                   | <b>I</b> MPROVEMENT                       | STYLE/MODEL<br>B9351-1    |                     |
|   |                                   | FIELD ENGINEERING<br>TECHNICAL OPERATIONS |                           | <b>N</b> OTICE      |
| STD. INSTALL. TIME<br>Up to 1 hour              | UNITS AFFECTED<br>D/L 4 and below | UNIT DESCRIPTION<br>Control Unit          |                           |                     |
| TITLE<br>Cursor Memory Address Count Correction |                                   |   |                           | DATE<br>6-9-70      |

PREREQUISITE: None

CONDITION: Cursor Memory Address does not always count correctly.

CAUSE: Marginal condition exists in flip-flop chips that results in erratic counting of the Cursor Memory Address register.

CORRECTION: Install change described below.

PARTS REQUIRED:

| <u>Number</u> | <u>Description</u>                | <u>Qty</u> | <u>Unit List Price</u> |
|---------------|-----------------------------------|------------|------------------------|
| 1700 8723     | Capacitor, 470pf.<br>± 10%, 100v. | 1          | \$1.35                 |

INSTRUCTIONS: All changes are to be made to the Update Address Control card, Type 034 (P/N 1725 5225) or Type 134 (P/N 1732 4922), at location 203.

1. Cut the etching as shown in Figure 1 on page 2.
2. Add wire from C2-F to C2-J as shown in Figure 1.
3. Add capacitor C5 as shown in Figure 1.
4. Redline logic diagram 1725 5266 or 1732 4930 as shown in Figure 2.

Note: Approximately 6 inches of insulated 26 AWG solid wire ( 1142 8075 or equivalent ) is required.



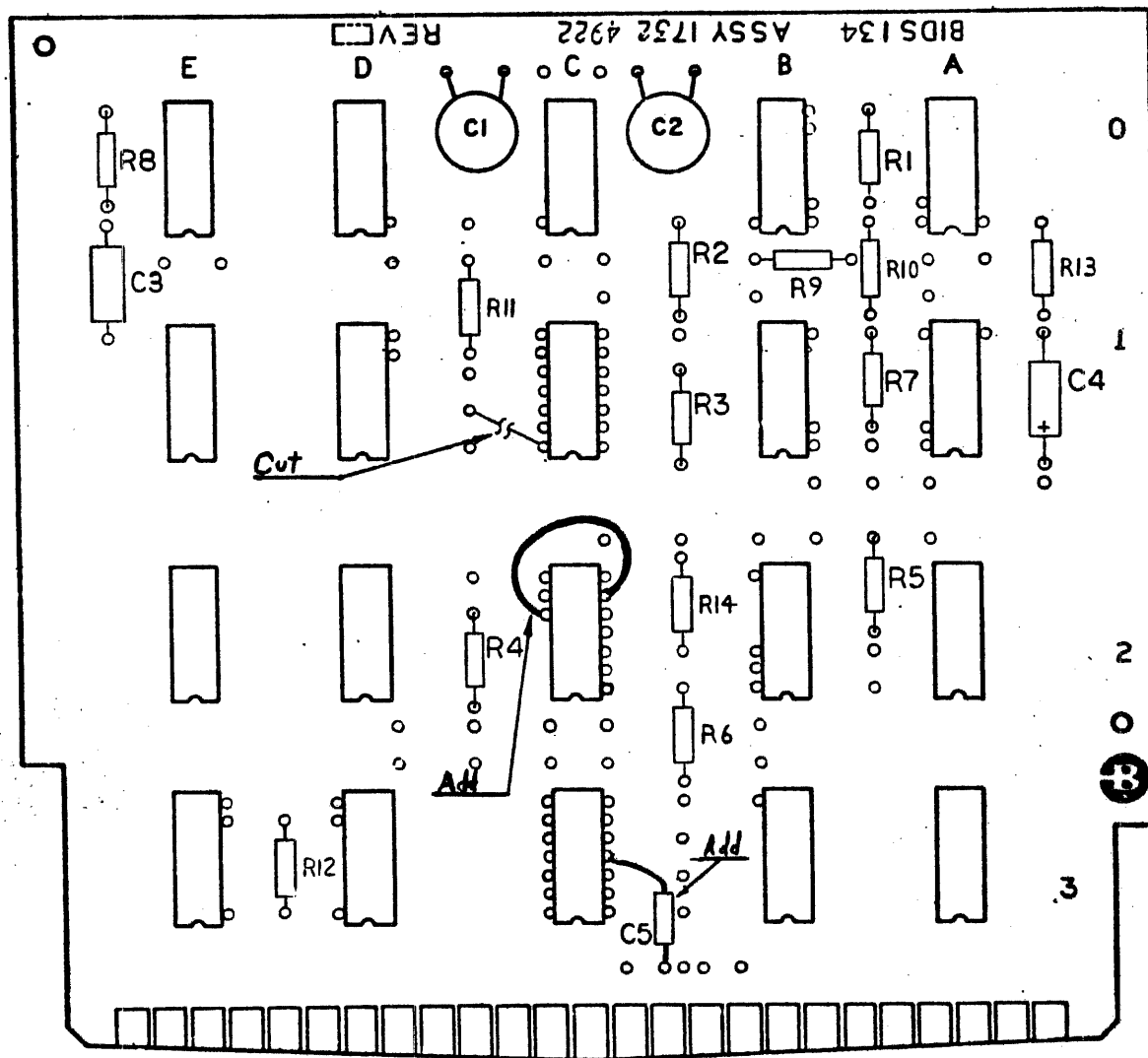
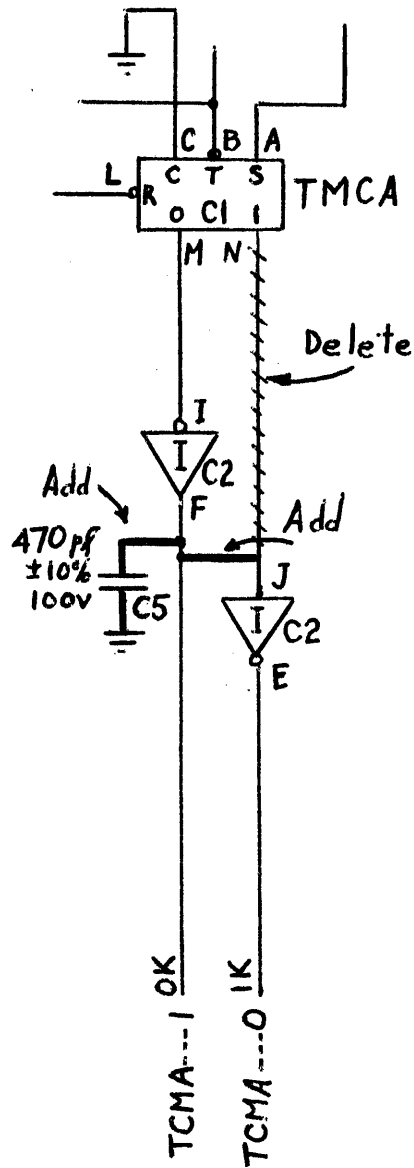


Figure 1



Redline Logic Diagram 1725 5266 or 1732 4930 (Loc 203)

Figure 2

|   |                              |                                      |                           |                     |
|---|------------------------------|--------------------------------------|---------------------------|---------------------|
| <b>Burroughs</b>                                    |                              | <b>R</b> ELIABILITY                  | SYSTEM SERIES<br>Series B | <b>No.</b> 4187-003 |
| FIELD ENGINEERING<br>TECHNICAL OPERATIONS           |                              | <b>I</b> MPROVEMENT                  | STYLE/MODEL<br>B9351-4    | PAGE<br>1 OF 2      |
|   |                              | <b>N</b> OTICE                       | TOP UNIT NO.<br>1700 1181 |                     |
| STD. INSTALL. TIME<br>Up to 1 hour                  | UNITS AFFECTED<br>Below 6123 | UNIT DESCRIPTION<br>Monitor Assembly |                           |                     |
| TITLE<br>Increase stability of Deflection Amplifier |                              |                                      |                           | DATE<br>6-9-70      |

CONDITION: Some deflection amplifiers have a tendency to go into oscillation. This condition can be determined by observing a screen presentation. If the characters appear to oscillate, this change should be installed.

CAUSE: Deflection amplifier unstable.

CORRECTION: Change capacitor C7 from 180pf to 270pf.

PARTS REQUIRED:

| <u>Number</u> | <u>Description</u>           | <u>Qty</u> | <u>Unit List Price</u> |
|---------------|------------------------------|------------|------------------------|
| 1700 8699     | Capacitor, 270pf, +10%, 100v | 2          | \$1.35                 |

Instructions:

1. Remove C7 (180pf) from deflection amplifier assembly (P/N 1700 0811). Location is shown in Figure 1. It is NOT necessary to remove the DA from the Monitor.
2. Install new C7 capacitor (270pf).
3. Repeat steps 1 and 2 for the remaining DA.
4. Redline the DA schematic (1700 0829) as shown in Figure 2.

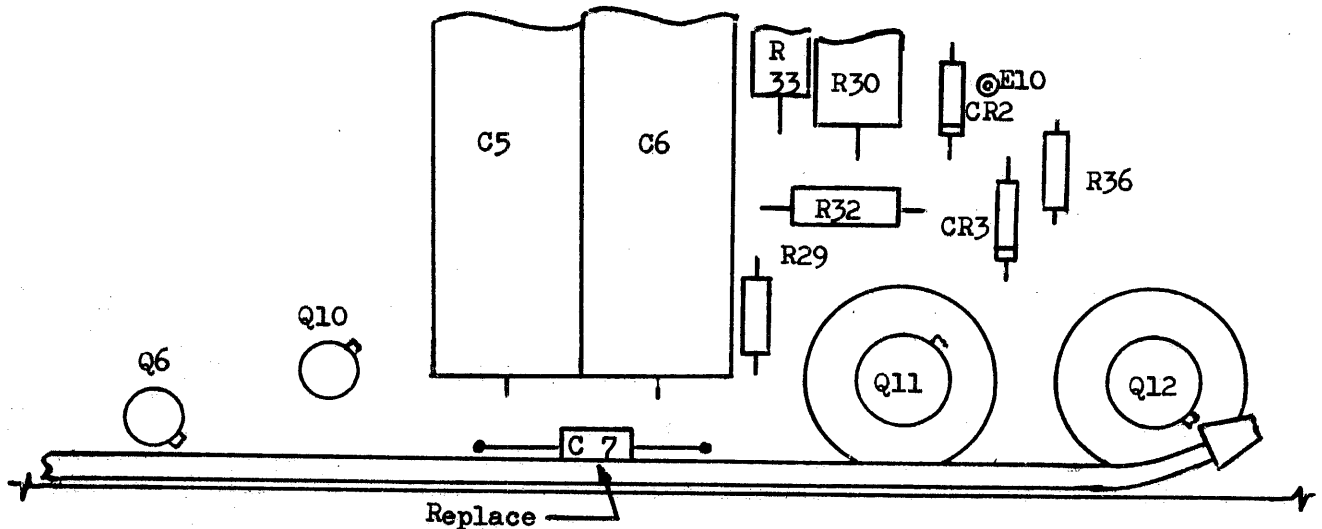
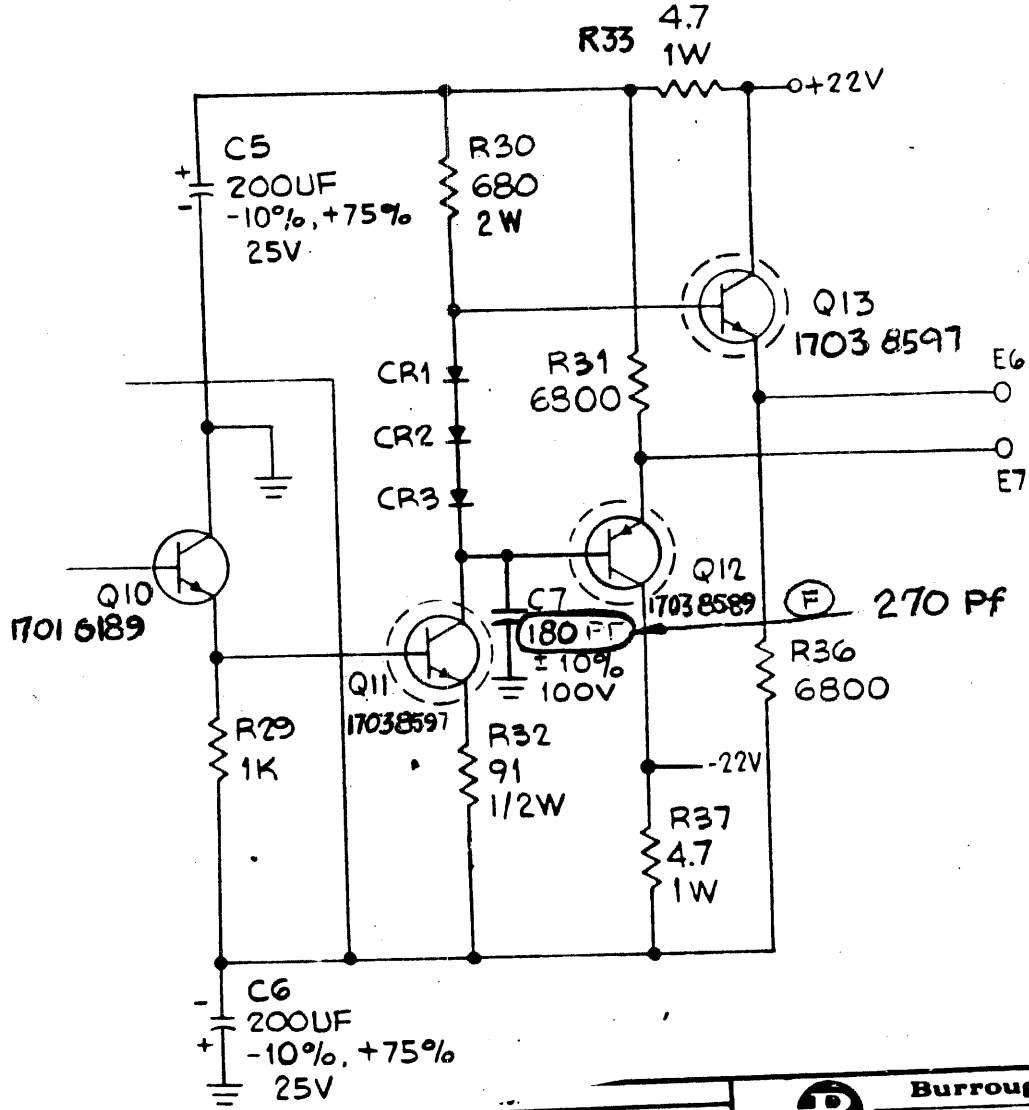


Figure 1

| REVISIONS |      |             |               |
|-----------|------|-------------|---------------|
| ZONE      | LTIC | DESCRIPTION | DATE APPROVED |
|           | B    |             |               |
|           | C    | C-72266     |               |
|           | D    | C-72421     |               |
|           | E    | C72519      |               |
|           | F    | C-73205     |               |

Add →



ANGLES 2.010 ±2°  
 SURFACE FINISH NOT TO EXCEED .001"  
 REMOVE ALL BURRS AND BREAK ALL SHARP EDGES

|              |  |
|--------------|--|
| MATERIAL     |  |
| DATE USED ON |  |
| REVISION     |  |

|                         |                 |
|-------------------------|-----------------|
| PREPARED BY<br>W. YOUNG | DATE<br>5-20-67 |
| CHECKED                 | 5-2-67          |
| DESIGNED BY             | 1-10-67         |
| APPROVED                | 5/1/67          |
|                         | 7/26/67         |

**Burroughs Corporation**  
 DEFENSE AND SPACE GROUP  
 MOULTON HILLS, PENNSYLVANIA

**SCHMATIC DIAGRAM, ELECTRONIC DEFLECTION AMPLIFIER BIDS (DA)**

SIZE: **D 08241**      1700 0829

SCALE: \_\_\_\_\_      D SHEET

DOC NO 1700 0829      REV F

Change

|   |                       |   |                     |
|---|-----------------------|---|---------------------|
| <b>Burroughs</b><br>FIELD ENGINEERING<br>TECHNICAL OPERATIONS | <b>RELIABILITY</b>    | SYSTEM SERIES<br>Series 'B'                 | <b>No.</b> 4187-004 |
|   | <b>IMPROVEMENT</b>    | STYLE/MODEL<br>B9351-4                      | PAGE<br>1 of 2      |
|   | <b>NOTICE</b>         | TOP UNIT NO.<br>1700 1181                   |                     |
| STD. INSTALL. TIME<br>less than 1 hour                        | UNITS AFFECTED<br>ALL | UNIT DESCRIPTION<br><b>Monitor Assembly</b> |                     |
| TITLE<br><b>Character Non-Linearity Correction</b>            |                       |   | DATE<br>10-6-70     |

**CONDITION:** Characters constructed of horizontal strokes, such as 'E' and 'F', are distorted on some units.

**CAUSE:** Variations in the characteristics of the deflection yoke.

**CORRECTION:** Increase the value of resistor R1 across the vertical deflection coils of the yoke assembly.

**PARTS REQUIRED:**

| <u>Number</u> | <u>Description</u>                             | <u>Qty</u> | <u>Unit List Price</u> |
|---------------|--|------------|------------------------|
| 1265 4455     | Resistor, 470 ohm, $\pm 5\%$ , $\frac{1}{4}$ w | 1          | \$ .33                 |

**INSTRUCTIONS:**

1. Remove R1 (390 ohm) from the deflection yoke assembly, terminals 7 and 8. Refer to Figure 1.
2. Install 470 ohm resistor (1265 4455) between terminals 7 and 8.
3. Redline schematic diagram, Monitor chassis (1700 0761 Sht 2), according to Figure 2.

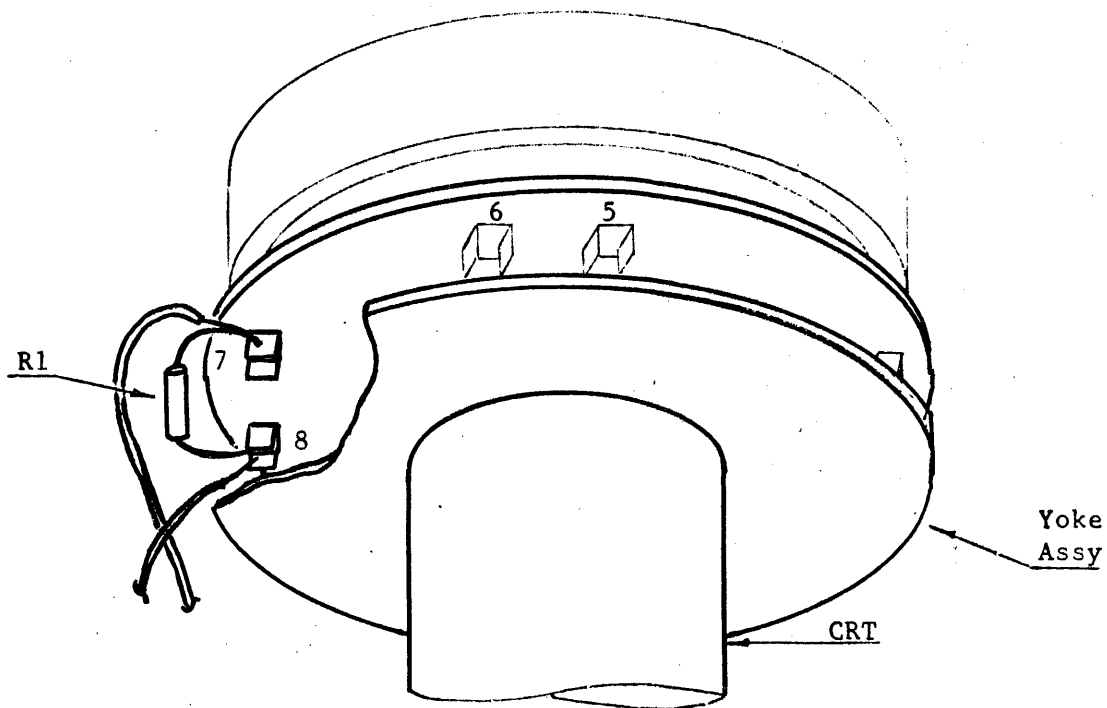


Figure 1

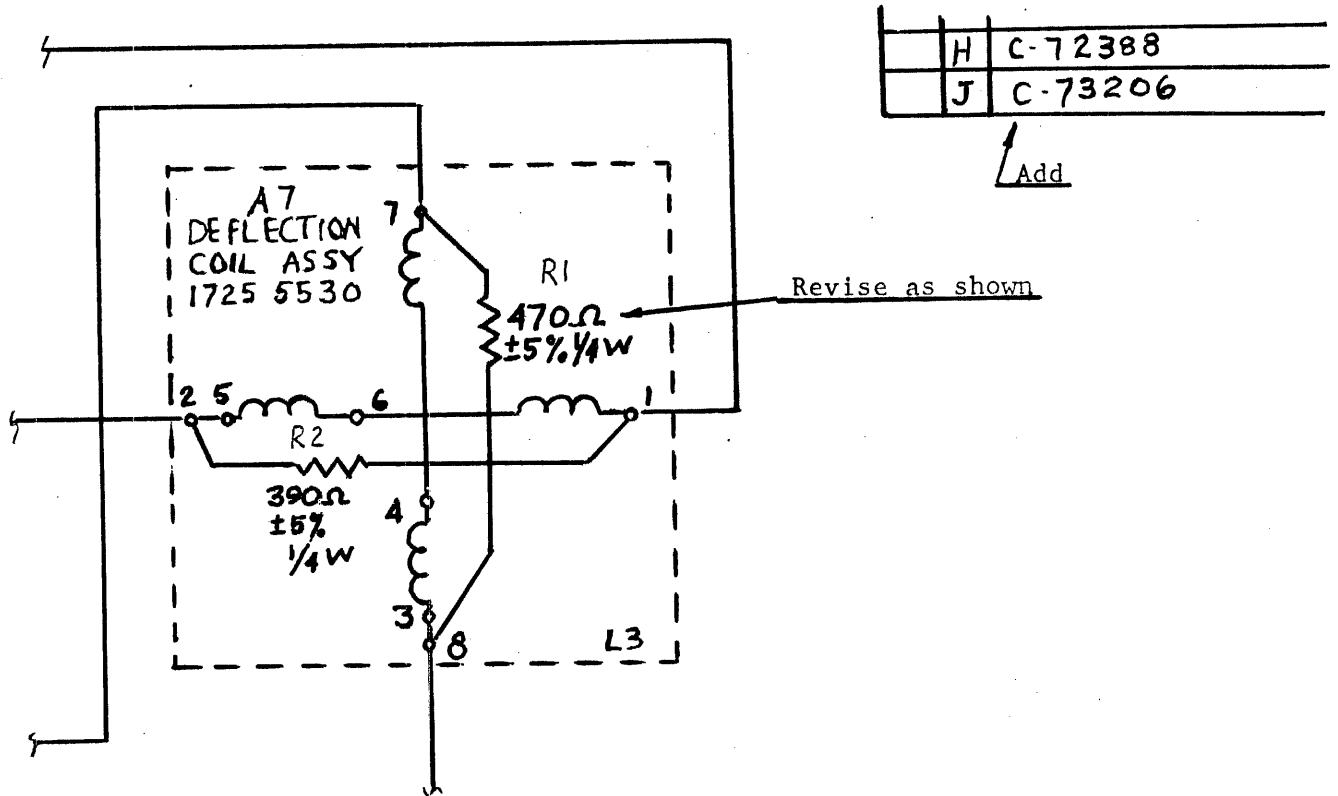


Figure 2

|   |                                     |                                  |                     |
|---|-------------------------------------|----------------------------------|---------------------|
| <b>Burroughs</b><br><br><b>FIELD ENGINEERING<br/>TECHNICAL OPERATIONS</b> | <b>R</b> ELIABILITY                 | SYSTEM SERIES<br>Series 'B'      | <b>No.</b> 4187-005 |
|   | <b>I</b> MPROVEMENT                 | STYLE/MODEL<br>B9351-1/B9353-1   | PAGE<br>1 of 1      |
|   | <b>N</b> OTICE                      | TOP UNIT NO.<br>See below        |                     |
| STD. INSTALL. TIME<br>less than 1 hour                                    | UNITS AFFECTED<br>Below serial 9000 | UNIT DESCRIPTION<br>Control Unit |                     |
| TITLE<br>Erroneous Setting of Input Error Indicator                       |                                     |                                  | DATE<br>10-20-70    |

Top Unit # 1701 4697, 1701 4705, 1701 4713, 1725 8500

CONDITION: Correct data occasionally causes the Input Error indicator to be set.

CAUSE: Logic error.

CORRECTION: Modify backplane as indicated in the instructions below.

PARTS REQUIRED: None

Note: About 2' of wire (1142 8075) required.

**INSTRUCTIONS:**

1. Make the following backplane changes:

| * | Signal   | From   | Z | To     | Z |
|---|----------|--------|---|--------|---|
| D | RCVMSG1  | AADB9X | 2 | AAFD5D | 2 |
| D | RLAA---1 | DB8G   | 2 | FD4P   | 2 |
| A | ENDBLK-1 | DB9X   | 1 | DB8G   | 1 |
| A |          | BD2S   | 2 | DB9X   | 2 |

\* A = Add  
D = Delete

2. Redline Logic Diagram 1737 4513, location 210, as shown in Figure 1
3. Redline circuit list to reflect above backplane change.

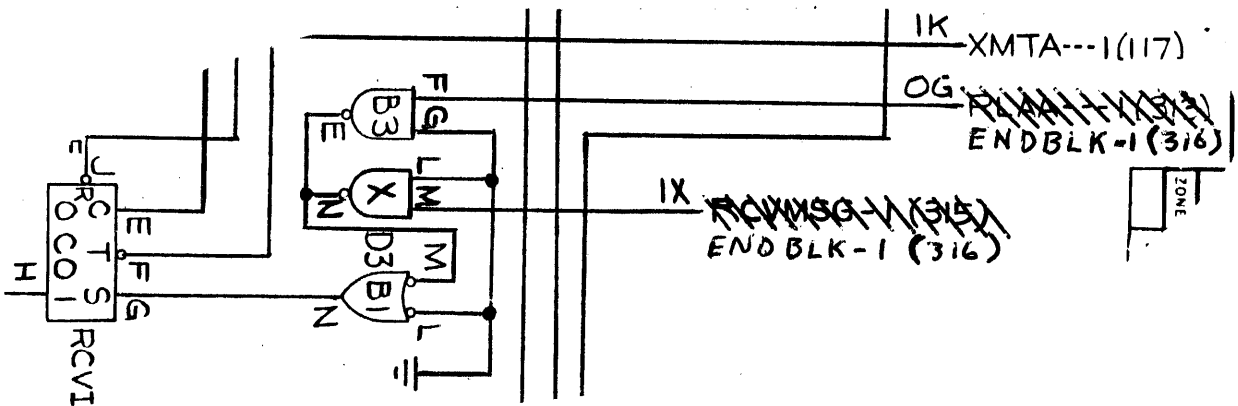


Figure 1

|  |                                     |  |                                |
|--|-------------------------------------|--|--------------------------------|
| <b>Burroughs</b>   | <b>R</b> ELIABILITY                 | SYSTEM SERIES<br>Series 'B'                    | <b>No.</b> 4187-006<br>REVISED |
|  | <b>I</b> MPROVEMENT                 | STYLE/MODEL<br>B9351/B9353 -1                  | PAGE<br>1 of 4                 |
|  | <b>N</b> OTICE                      | TOP UNIT NO.<br>See below                      |                                |
| STD. INSTALL. TIME<br>Up to 1 hour                         | UNITS AFFECTED<br>Below serial 6000 | UNIT DESCRIPTION<br>Control Unit II, III, & IV |                                |
| TITLE<br>Multiple Screen Operation Interference Correction |                                     |  | DATE<br>2-16-71                |

Top Unit No. 1701 4697, 1701 4705, 1701 4713

PREREQUISITE: RIN 4187-002

CONDITIONS: 1. Intermittent clearing of the screen occurs when the display time approaches 4 msec.

2. Two cursors may appear on the screen as the display time approaches 4 msec.

CAUSE: Logic error.

CORRECTION: Modify two (2) cards as indicated in the instructions below.

PARTS REQUIRED:

| <u>Number</u> | <u>Description</u>   | <u>Qty</u> | <u>Unit List Price</u> |
|---------------|----------------------|------------|------------------------|
| 1265 4919     | Resistor, 2K, 5%, ½w | 1          | \$ .33                 |

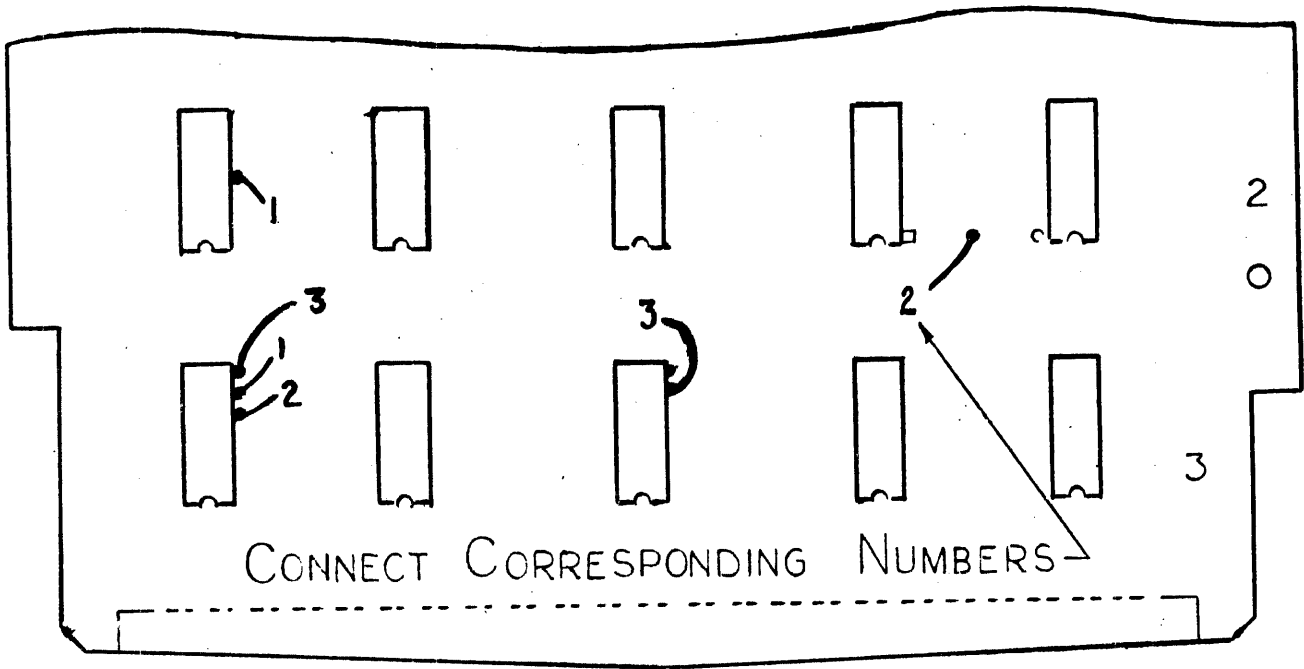
About 1' of wire (1142 8075) required.

INSTRUCTIONS:

1. Modify card type 014 (location 109) as shown in Figure 1. Relabel card as type 114 and P/N 1738 5022.
2. Modify card type n34 (location 203) by replacing resistor R7 with a 2K resistor as shown in Figure 2. Relabel card as type 334 and P/N 1738 5170.
3. Redline Card Allocation Chart 1725 8120 or 1737 8753 to reflect the new part numbers.
4. Replace Logic Diagrams for locations 109 and 203 with pages 3 and 4 of this RIN.

✓ Changes or additions since last issue.





✓ Figure 1

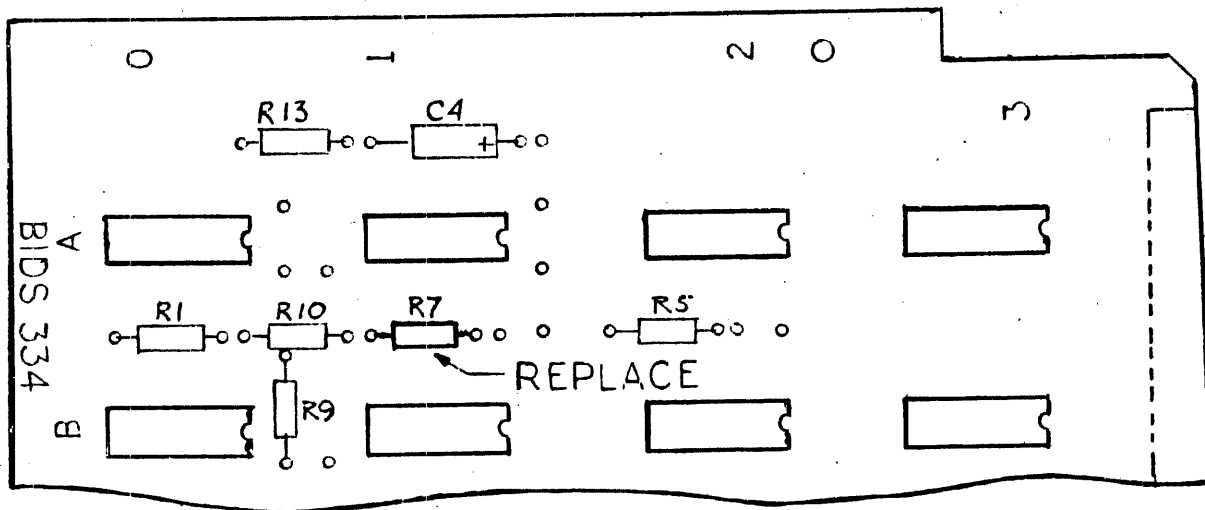
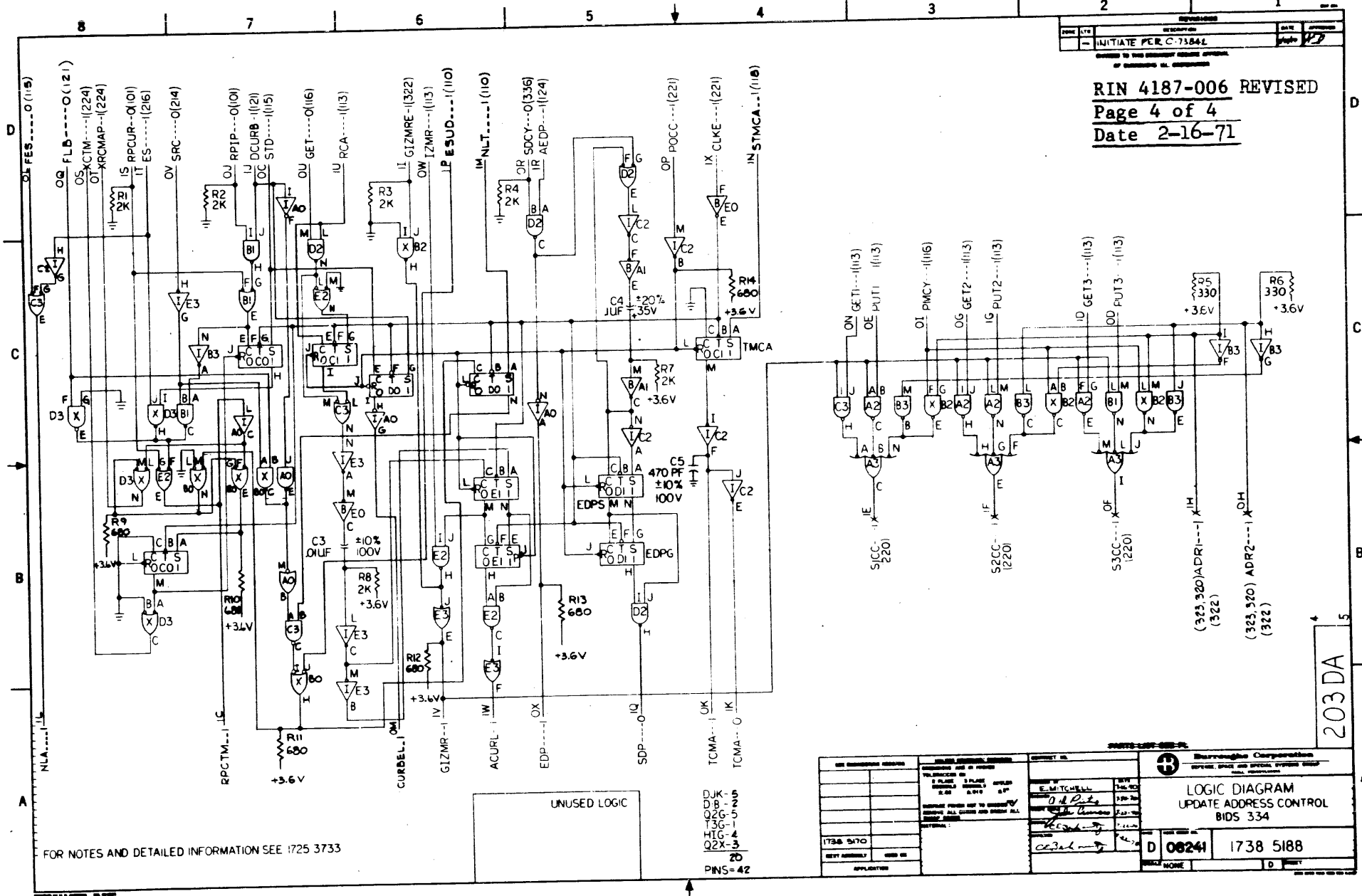


Figure 2



|     |                      |      |
|-----|----------------------|------|
| REV | DATE                 | BY   |
| 1   | INITIATE PER C-73844 | 1/77 |

RIN 4187-006 REVISED  
 Page 4 of 4  
 Date 2-16-71



FOR NOTES AND DETAILED INFORMATION SEE 1725 3733

UNUSED LOGIC

DJK-5  
 DB-25  
 Q2G-5  
 T3G-1  
 HIG-4  
 Q2X-3  
 20  
 PINS=42

203 DA

|           |             |            |                        |
|-----------|-------------|------------|------------------------|
| DATE      | DESIGNED BY | CHECKED BY | THORNTON CORPORATION   |
| 1738 5170 | E. MITCHELL |            | LOGIC DIAGRAM          |
| REV       | DATE        | BY         | UPDATE ADDRESS CONTROL |
| 1         |             |            | BIDS 334               |
| 2         |             |            |                        |
| 3         |             |            |                        |
| 4         |             |            |                        |
| 5         |             |            |                        |
| 6         |             |            |                        |
| 7         |             |            |                        |
| 8         |             |            |                        |
| 9         |             |            |                        |
| 10        |             |            |                        |
| 11        |             |            |                        |
| 12        |             |            |                        |
| 13        |             |            |                        |
| 14        |             |            |                        |
| 15        |             |            |                        |
| 16        |             |            |                        |
| 17        |             |            |                        |
| 18        |             |            |                        |
| 19        |             |            |                        |
| 20        |             |            |                        |
| 21        |             |            |                        |
| 22        |             |            |                        |
| 23        |             |            |                        |
| 24        |             |            |                        |
| 25        |             |            |                        |
| 26        |             |            |                        |
| 27        |             |            |                        |
| 28        |             |            |                        |
| 29        |             |            |                        |
| 30        |             |            |                        |
| 31        |             |            |                        |
| 32        |             |            |                        |
| 33        |             |            |                        |
| 34        |             |            |                        |
| 35        |             |            |                        |
| 36        |             |            |                        |
| 37        |             |            |                        |
| 38        |             |            |                        |
| 39        |             |            |                        |
| 40        |             |            |                        |
| 41        |             |            |                        |
| 42        |             |            |                        |
| 43        |             |            |                        |
| 44        |             |            |                        |
| 45        |             |            |                        |
| 46        |             |            |                        |
| 47        |             |            |                        |
| 48        |             |            |                        |
| 49        |             |            |                        |
| 50        |             |            |                        |
| 51        |             |            |                        |
| 52        |             |            |                        |
| 53        |             |            |                        |
| 54        |             |            |                        |
| 55        |             |            |                        |
| 56        |             |            |                        |
| 57        |             |            |                        |
| 58        |             |            |                        |
| 59        |             |            |                        |
| 60        |             |            |                        |
| 61        |             |            |                        |
| 62        |             |            |                        |
| 63        |             |            |                        |
| 64        |             |            |                        |
| 65        |             |            |                        |
| 66        |             |            |                        |
| 67        |             |            |                        |
| 68        |             |            |                        |
| 69        |             |            |                        |
| 70        |             |            |                        |
| 71        |             |            |                        |
| 72        |             |            |                        |
| 73        |             |            |                        |
| 74        |             |            |                        |
| 75        |             |            |                        |
| 76        |             |            |                        |
| 77        |             |            |                        |
| 78        |             |            |                        |
| 79        |             |            |                        |
| 80        |             |            |                        |
| 81        |             |            |                        |
| 82        |             |            |                        |
| 83        |             |            |                        |
| 84        |             |            |                        |
| 85        |             |            |                        |
| 86        |             |            |                        |
| 87        |             |            |                        |
| 88        |             |            |                        |
| 89        |             |            |                        |
| 90        |             |            |                        |
| 91        |             |            |                        |
| 92        |             |            |                        |
| 93        |             |            |                        |
| 94        |             |            |                        |
| 95        |             |            |                        |
| 96        |             |            |                        |
| 97        |             |            |                        |
| 98        |             |            |                        |
| 99        |             |            |                        |
| 100       |             |            |                        |

|   |  |   |                        |
|---|--|---|------------------------|
| <b>Burroughs</b><br><br><b>FIELD ENGINEERING<br/>TECHNICAL OPERATIONS</b> | <b>RELIABILITY</b>                         | <b>SYSTEM SERIES</b><br>Series 'B'      | <b>No.</b> 4187-008    |
|   | <b>IMPROVEMENT</b>                         | <b>STYLE/MODEL</b><br>B9351/B9353       |                        |
|   | <b>NOTICE</b>                              | <b>TOP UNIT NO.</b><br>See below        |                        |
| <b>STD. INSTALL. TIME</b><br>Up to 2 hours                                | <b>UNITS AFFECTED</b><br>Below serial 7300 | <b>UNIT DESCRIPTION</b><br>Control Unit |                        |
| <b>TITLE</b><br>Intermittent Send/Receive Logic Corrections               |  |   | <b>DATE</b><br>12-8-70 |

Note: Check Unit - RIN may have been previously installed.

Top Unit No. 1701 4697, 1701 4705, 1701 4713, 1725 8500

**CONDITIONS:**

1. Cursor intermittently repositions to End-of-Screen when sending or receiving.
2. Screens intermittently flash when sending or receiving.
3. Data intermittently jumps up one line when typing past position 80.
4. Transmission or reception of Fixed Tab characters cause characters to change to spaces intermittently.

**CAUSE:** Logic error.

**CORRECTION:** Install backplane changes (1 access and 3 adds) and modify four (4) cards - Card Type 004, 110, 123, and 142. Modified cards can be identified as Card Types 104, 210, 423 and 242.

**INSTRUCTION MEDIA PACKAGE NO. 1051158**  
(Includes one set of the following Media:)

Installation Procedure (3 pages)  
Logic Diagrams (4 sheets)

**PARTS REQUIRED:**

| <u>Number</u>                          | <u>Description</u>          | <u>Qty</u> | <u>Unit List Price</u> |
|--|-----------------------------|------------|------------------------|
| 1265 4570                              | Resistor, 680 ohm, 5%, 1/2w | 1          | \$1.13                 |
| About 7' of wire (1142 8075) required. |                             |            |                        |

|   |  |   |                        |
|---|--|---|------------------------|
| <b>Burroughs</b><br><br><b>FIELD ENGINEERING<br/>TECHNICAL OPERATIONS</b> | <b>RELIABILITY</b>                         | <b>SYSTEM SERIES</b><br>Series 'B'      | <b>No.</b> 4187-009    |
|   | <b>IMPROVEMENT</b>                         | <b>STYLE/MODEL</b><br>B9351/B9353 -1    | <b>PAGE</b><br>1 of 2  |
|   | <b>NOTICE</b>                              | <b>TOP UNIT NO.</b><br>See below        |                        |
| <b>STD. INSTALL. TIME</b><br>Up to 1 hour                                 | <b>UNITS AFFECTED</b><br>Below serial 7300 | <b>UNIT DESCRIPTION</b><br>Control Unit |                        |
| <b>TITLE</b><br>Line Deletion Correction                                  |  |   | <b>DATE</b><br>12-8-70 |

Top Unit No. 1701 4697, 1701 4705, 1701 4713, 1725 8500

CONDITION: Deleting a line containing 80 characters may, in some units, delete more than one line.

CAUSE: Logic error.

CORRECTION: Modify card type 007 according to the instructions below.

**PARTS REQUIRED:**

| <u>Number</u>  | <u>Description</u>                     | <u>Qty</u> | <u>Unit List Price</u> |
|--|--|------------|------------------------|
| 1265 4570  | Resistor, 680 ohm, 5%, $\frac{1}{2}$ w | 1          | \$1.13                 |
| Approximately 6 inches of wire (1142 8075) required. |  |            |                        |

**INSTRUCTIONS:**

1. Modify card type 007 (location 226) by adding the two wires and the resistor shown in Figure 1. Relabel the card as type 1Q7 and P/N 1737 8779.
2. Redline the Card Allocation Chart to reflect the new part number.
3. Replace the Logic Diagram for location 226 with page 2 of this RIN.

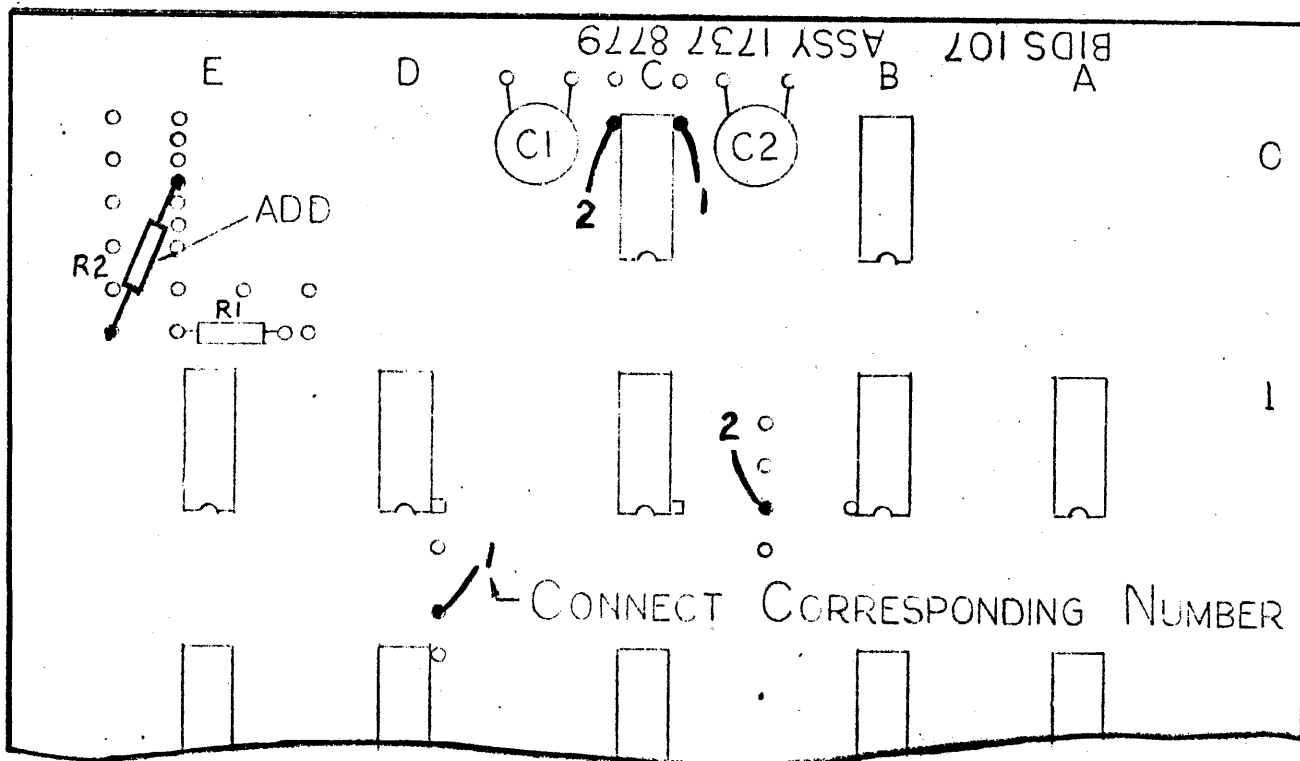
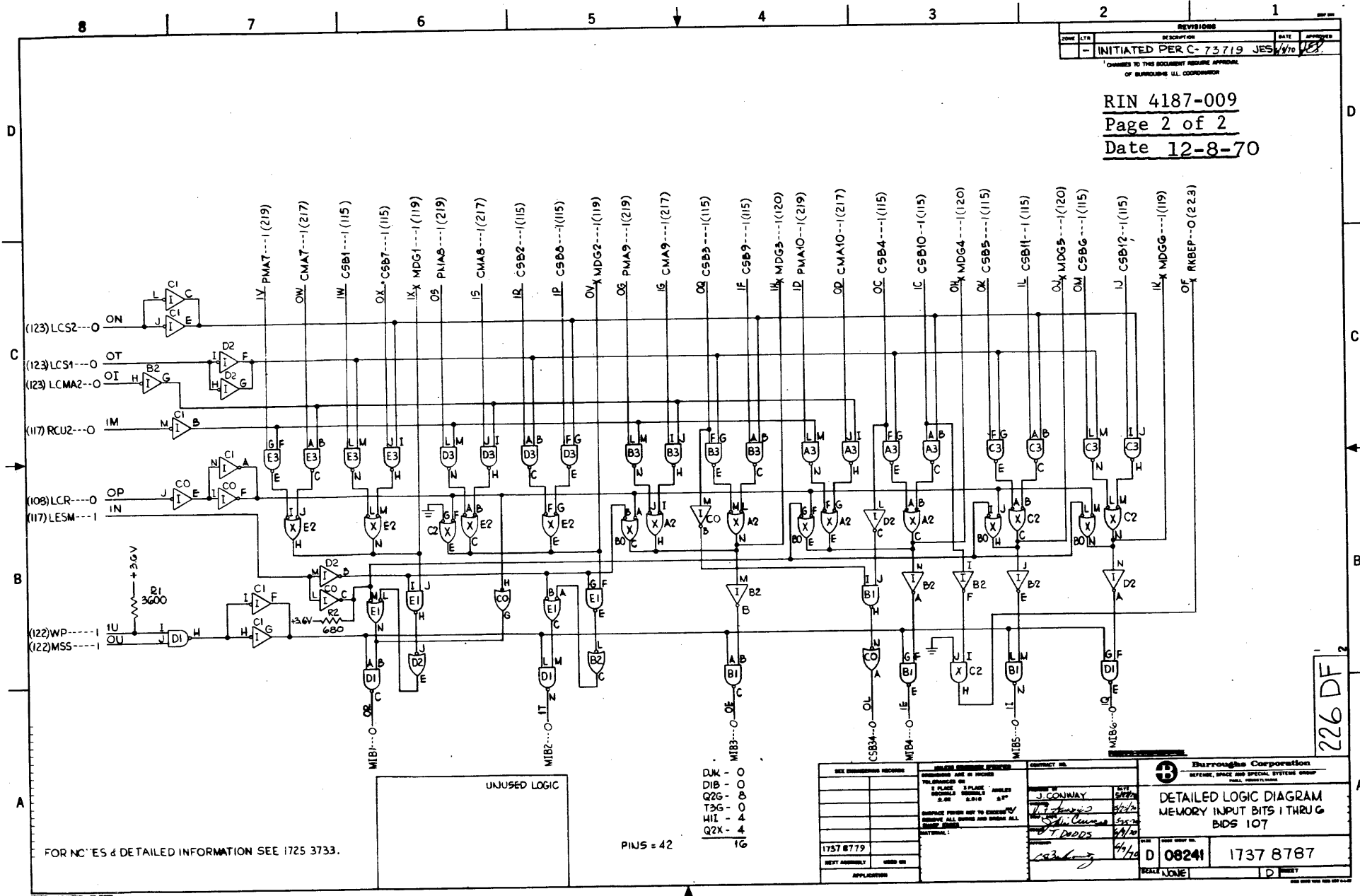


Figure 1

| REVISIONS |     |                       |          |
|-----------|-----|-----------------------|----------|
| ZONE      | LTB | DESCRIPTION           | DATE     |
| -         | -   | INITIATED PER C-73719 | JES 4/70 |

CHANGES TO THIS DOCUMENT REQUIRE APPROVAL OF BARRON'S ILL. COORDINATOR

RIN 4187-009  
 Page 2 of 2  
 Date 12-8-70



FOR NOTES & DETAILED INFORMATION SEE 1725 3733.

PINS = 42

- DUX - 0
- DIB - 0
- Q2G - 0
- T3G - 0
- H11 - 4
- Q2X - 4
- IG - 6

226 DF

|                         |   |   |  |
|-------------------------|---|---|--|
| SEE ENGINEERING RECORDS | DESIGNED AND IN CHARGE  | CONTRACT NO.  | Barron's Corporation<br>DEFENSE, SPACE AND SPECIAL SYSTEMS GROUP<br>FALLS CHURCH, VA |
|                         | TRANSDUCED ON 1 PLACE<br>ORIGINALS MADE 1 PLACE<br>COPIES MADE 1 PLACE<br>5" x 8" | DATE OF J. CONWAY<br>DATE OF J. CONWAY<br>DATE OF J. CONWAY |  |
| 1737 8779               | CHECKED BY<br>DATE  | DATE OF J. CONWAY<br>DATE OF J. CONWAY                      | REACT NONE   |

|  |                                     |                                  |                                |
|--|-------------------------------------|----------------------------------|--------------------------------|
| <b>Burroughs</b>                             | <b>R</b> ELIABILITY                 | SYSTEM SERIES<br>Series 'B'      | <b>No.</b> 4187-010<br>REVISED |
|  | <b>I</b> MPROVEMENT                 | STYLE/MODEL<br>B9351/B9353       | PAGE<br>1 OF 2                 |
|  | <b>N</b> OTICE                      | TOP UNIT NO.<br>See below        |                                |
| STD. INSTALL. TIME<br>Up to 1 hour           | UNITS AFFECTED<br>Below serial 7300 | UNIT DESCRIPTION<br>Control Unit |                                |
| TITLE<br>Escape Character Transmission Error |                                     |                                  | DATE<br>1-14-71                |

Note: Check unit - RIN may have been previously installed.

Top Unit No. 1701 4697, 1701 4705, 1701 4713, 1725 8500

CONDITION: When an escape character (\$, %, &, , x, , ≠, or ) is displayed in position 1 of line 1 and selective send is evoked, the first character transmitted may have a bit 3 error.

CAUSE: Logic error.

CORRECTION: Modify card type 143 according to the instructions below which adds the level RPMA---1 to the gating circuit to set the SCHAR Flip-Flop.

PARTS REQUIRED:

| <u>Number</u>                          | <u>Description</u>           | <u>Qty</u> | <u>Unit List Price</u> |
|--|------------------------------|------------|------------------------|
| 1265 4570                              | Resistor, 680 ohms, 5%, 1/4w | 1          | \$1.13                 |
| About 1' of wire (1142 8075) required. |                              |            |                        |

INSTRUCTIONS:

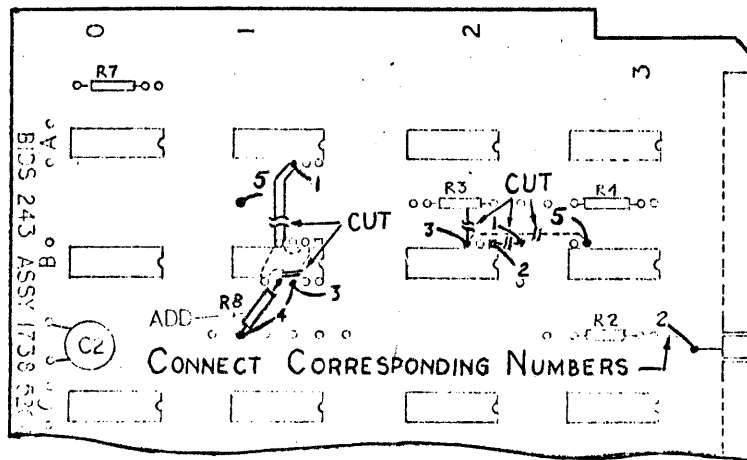
1. Modify card type 143 (location 316) as shown in Figure 1. Note that the IC at location B1 must be removed in order to make the cut at B1-L. Note also that two (2) other cuts are on the back side of the board. There are five (5) cuts in all.
2. Relabel the card as type 243 and part number 1738 5204.
3. Replace the Logic Diagram for location 316 with page 2 of this RIN.
4. Redline the Card Allocation Chart, 1725 8120 or 1737 8753, to reflect the new part number.

Cut the following:

- A1-L to Gnd
- B2-A to B0-J
- B2-C to B3-F
- B1-L to Gnd

Add to following:

- A1-L to B0-J
- B2-A to OK
- B2-C to B1-L
- B1-L to R8
- A1-I to B3-F
- R8 to B1-K (3.6v)



✓ Figure 1

✓ Changes or additions since last issue.





|   |                                     |                                  |                             |                     |
|---|-------------------------------------|----------------------------------|-----------------------------|---------------------|
| <b>Burroughs</b><br><br>FIELD ENGINEERING<br>TECHNICAL OPERATIONS |                                     | <b>R</b> ELIABILITY              | SYSTEM SERIES<br>Series 'B' | <b>No.</b> 4187-011 |
|   |                                     | <b>I</b> MPROVEMENT              | STYLE/MODEL<br>B9351/B9353  | PAGE<br>1 of 4      |
|   |                                     | <b>N</b> OTICE                   | TOP UNIT NO.<br>See below   |                     |
| STD. INSTALL. TIME<br>Up to 2 hours                               | UNITS AFFECTED<br>Below serial 7300 | UNIT DESCRIPTION<br>Control Unit |                             |                     |
| TITLE<br>Circuit Card Compatability Correction                    |                                     |                                  |                             | DATE<br>12-8-70     |

Top Unit No. 1701 4697, 1701 4705, 1701 4713, 1725 8500

CONDITION: Logic card assemblies are not always interchangeable with card assemblies in another Control Unit.

CAUSE: Marginal clock timing.

CORRECTION: Provide an adjustment for the 3CLK to ensure that the positive going edge of the pulse precedes the positive going edge of CLKA by 15 nanoseconds.

**PARTS REQUIRED:**

| <u>Number</u> | <u>Description</u>                     | <u>Qty</u> | <u>Unit List Price</u> |
|---------------|--|------------|------------------------|
| 1265 3978     | Resistor, 100 ohm, 5%, $\frac{1}{2}w$  | 1          | \$ .41                 |
| 1265 4554     | Resistor, 620 ohm, 5%, $\frac{1}{2}w$  | 1          | 1.13                   |
| 1265 4695     | Resistor, 1k, 5%, $\frac{1}{2}w$       | 1          | .69                    |
| 1700 8566     | Capacitor, 56pf, 7.5%, 100v            | 1          | 4.80                   |
| 1721 4115     | Resistor, Var., 1k, 0%, $\frac{1}{2}w$ | 1          | 2.15                   |

About 1' of wire (1142 8075) required.

**TOOLS REQUIRED:**

$\frac{1}{16}$  in. and  $\frac{1}{8}$  in. twist drill bits.  
Hand or  $\frac{1}{2}$  in. Power Drill.

**INSTRUCTIONS:**

1. Modify the Clock card, (location 221), by first drilling the six holes indicated in Figure 1.
2. Remove resistor R9, 1800 ohms, and discard. Cut the etching as shown in Figure 2. Cut the etching at two (2) locations on the opposite side of the board as shown in Figure 3.
3. Replace R10, 510 ohms, with the 100 ohm resistor. When mounting the resistor, leave the lead nearer to Q3 at least  $\frac{1}{2}$ " beyond the pad. Mount the capacitor, 56pf, at the former R9 location. Leave the lead nearer to Z1 long enough to reach the 'Gnd plane' etching. The remaining lead is to be soldered to the lead extending from R10. See Figure 3.
4. Mount the potentiometer, R32, by bending the tabs over on the back side of the board. Mount the new R9 resistor, 620 ohms, as shown in Figure 2, soldering the lead nearer CR6 to the '+12 volt plane'. The opposite end is connected to one terminal of R32.
5. Connect a wire from the remaining terminal of R32 to one of the mounting tabs of the potentiometer. Connect a wire from the same terminal to R10 and then to the base of Q2. See Figures 2 and 3.
6. Replace R15, 1800 ohms, with the 1K resistor.
7. Check the wiring against the new schematic, page 4 of this RIN.

8. Adjustment of R32 is done in the following manner:
  - a. Trace A is connected to CLKA---1 at backplane pin DE1E.
  - b. Trace B is connected to 3CLK---1 at backplane pin DE1J.
  - c. Sync positive on the A trace.
  - d. Adjust R32 so that the positive going edge of 3CLK precedes the positive going edge of CLKA by 15 nanoseconds.
9. The procedure in step 8 above should be added to section IV of the Technical Manual.

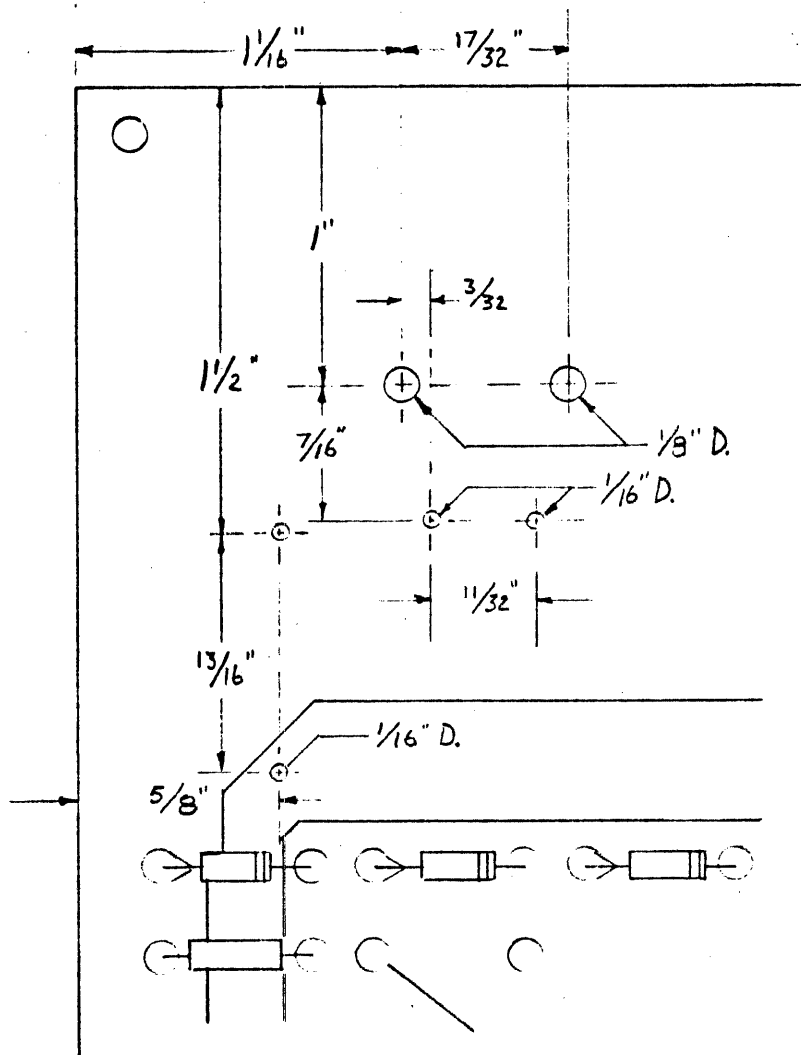


Figure 1

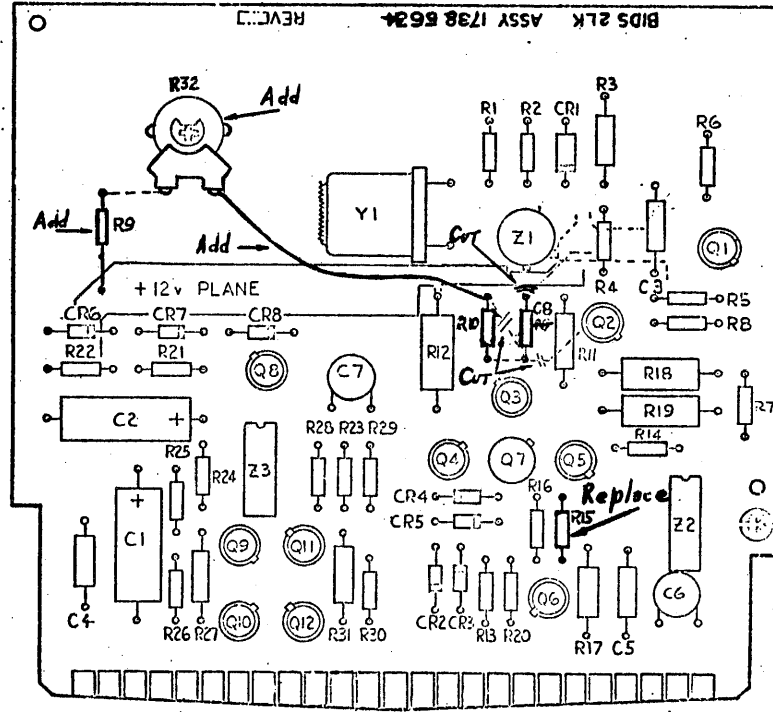


Figure 2

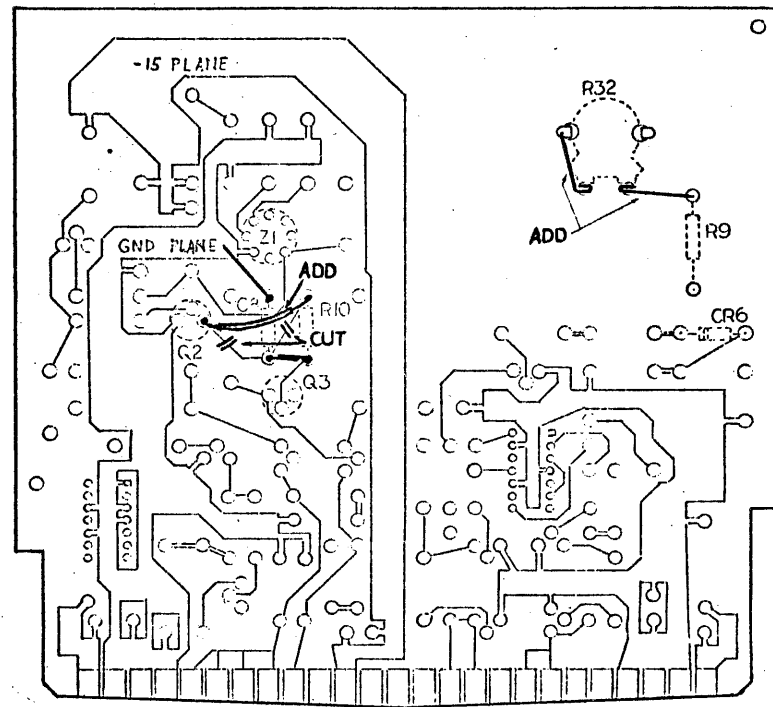
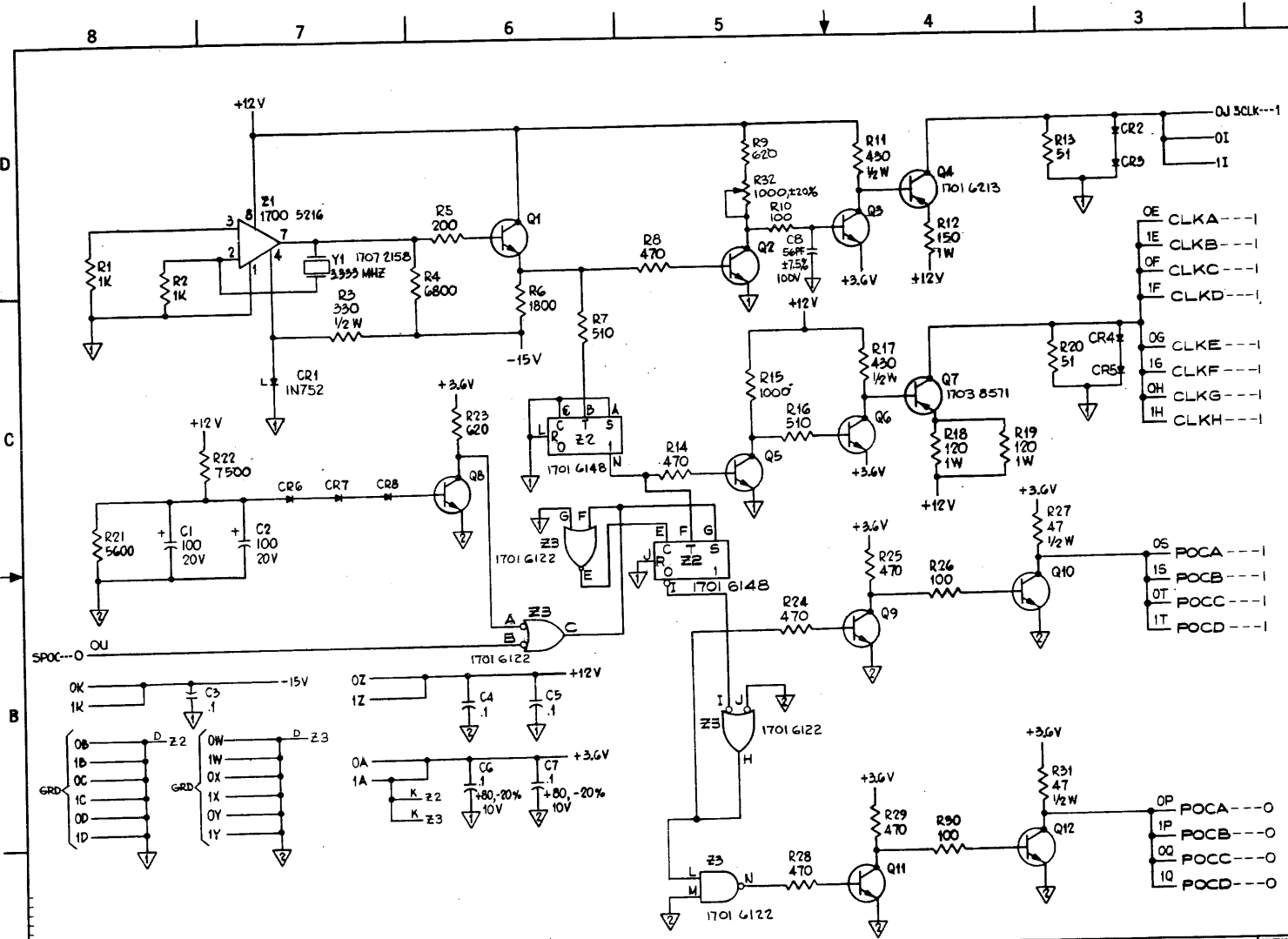


Figure 3

| REVISIONS |    |  |          |          |
|-----------|----|--|----------|----------|
| DATE      | BY | DESCRIPTION  | DATE     | APPROVED |
|           |    | CHANGES TO THIS DOCUMENT REQUIRE APPROVAL OF MANAGERIAL SUPERVISOR |          |          |
|           |    | - INITIATED PER C-73897  | 11/27/70 |          |

RIN 4187-011  
 Page 4 of 4  
 Date 12-8-70



1. UNLESS OTHERWISE SPECIFIED:  
 ALL RESISTOR VALUES ARE IN OHMS,  $\pm 5\%$ ,  $\frac{1}{4}$  W.  
 ALL CAPACITOR VALUES ARE IN MICROFARADS,  $\pm 10\%$ , 50V.  
 ALL DIODES ARE 1703 8662.  
 ALL TRANSISTORS ARE 1701 6205.

NOTES:

|   |  |                                   |   |
|---|--|-----------------------------------|---|
| SEE CONNECTIONS RECORD  | SEE FOR DIMENSIONAL SPECIFICATIONS   | CONTRACT NO.                      | Boeing Corporation<br>DEFENSE, SPACE AND AERIAL SYSTEMS GROUP<br>PHOENIX, ARIZONA |
| DIMENSIONS ARE IN INCHES<br>TOLERANCES ON:<br>2 PLACE DECIMALS .010"<br>3 PLACE DECIMALS .005"<br>SURFACE FINISH SEE TO EXCEED 100<br>REMOVE ALL SPICES AND BREAK ALL<br>SHARP EDGES TO .010" RADIUS<br>MATERIAL: | DESIGNED BY<br>J. SCHMIDT<br>DRAWN BY<br>J. SCHMIDT<br>CHECKED BY<br>T. D. DAVIS<br>DATE<br>11/27/70 | PART NO.<br>08241<br>REV.<br>NONE |   |
| 1738 5642   | PART NO.<br>1738 5642  | SCALE<br>NONE                     | SHEET NO.<br>4  |

221 DE 2

|   |                       |                                      |                     |
|---|-----------------------|--------------------------------------|---------------------|
| <b>Burroughs</b><br><br><b>FIELD ENGINEERING<br/>TECHNICAL OPERATIONS</b> | <b>R</b> ELIABILITY   | SYSTEM SERIES<br>Series 'B'          | <b>No.</b> 4187-012 |
|   | <b>I</b> MPROVEMENT   | STYLE/MODEL<br>B9351-4               | PAGE<br>1 OF 1      |
|   | <b>N</b> OTICE        | TOP UNIT NO.<br>1700 1181            |                     |
| STD. INSTALL. TIME<br>Up to 1 hour  | UNITS AFFECTED<br>ALL | UNIT DESCRIPTION<br>Monitor Assembly |                     |
| TITLE<br>Heat Dissipators Shorting to Ground on Deflection Amp.           |                       |                                      | DATE<br>3-4-71      |

CONDITION: Heat dissipators for the output transistors on the Deflection amplifiers have a tendency to short to the ground plane that surrounds the dissipator.

CAUSE: Insufficient clearance between heat sink and ground plane.

CORRECTION: Install non-conductive washer between heat sink and ground plane of Deflection Amplifier card assembly (1700 0811).

PARTS REQUIRED:

| <u>Number</u> | <u>Description</u>     | <u>Qty</u> | <u>Unit List Price</u> |
|---------------|------------------------|------------|------------------------|
| 1734 4813     | Washer, non-conductive | 6          | \$ .15                 |

INSTRUCTIONS:

1. Examine the position of the heat sinks for transistors Q11 and Q12 (Refer to Figure 1). If a piece of punch card stock, .007", can not be slipped under the heat sink, the transistor should be remounted with the non-conductive washer placed between the heat sink and the board.
2. If there is sufficient clearance, the non-conductive washer can be installed when it is necessary to replace the transistor.
3. Repeat steps 1 and 2 for the remaining Deflection Amplifier.
4. The remaining washers are for Q13 of the DA .

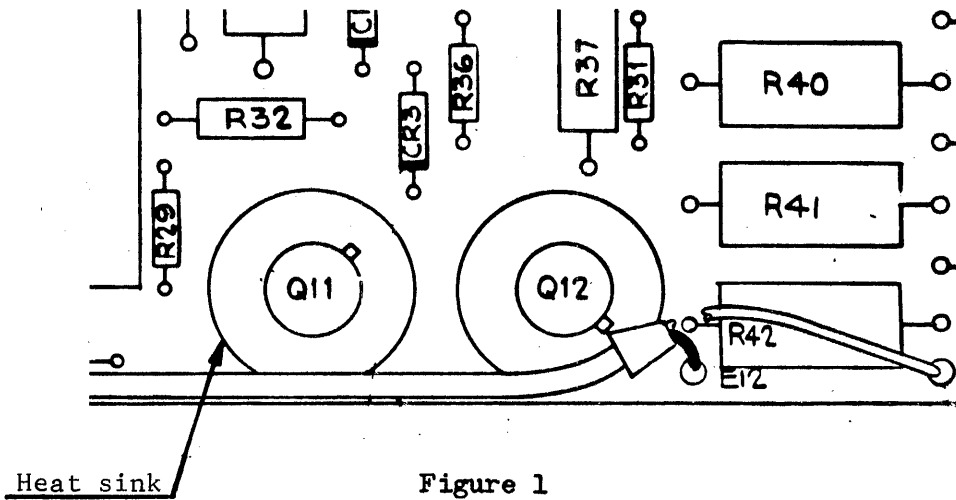


Figure 1

|   |                                      |                              |                     |
|---|--------------------------------------|------------------------------|---------------------|
| <b>Burroughs</b><br>FIELD ENGINEERING<br>TECHNICAL OPERATIONS | <b>R</b> ELIABILITY                  | SYSTEM SERIES<br>Series 'B'  | <b>No.</b> 4187-013 |
|   | <b>I</b> MPROVEMENT                  | STYLE/MODEL<br>B9351-5       | PAGE<br>1 of 2      |
|   | <b>N</b> OTICE                       | TOP UNIT NO.<br>1700 1199    |                     |
| STD. INSTALL. TIME<br>Up to 1 hour                            | UNITS AFFECTED<br>Below serial 07267 | UNIT DESCRIPTION<br>Keyboard |                     |
| TITLE<br>Shift Key and Marginal Noise Correction              |                                      |                              | DATE<br>3-4-71      |

**CONDITIONS:**

1. Shift function intermittently remains on for an additional character time.
2. A marginal noise condition exists on the 'Repeat' function.

**CAUSE:** Shift flip-flop not being turned off properly and capacitors C5, C6, C7 and possibly C18 reversed.

**CORRECTION:** Modify logic for clearing flip-flop SFT, change RC network for the 10CPS line, and remount C5, C7, C8 and C18 if necessary.

**PARTS REQUIRED:**

| Number    | Description                 | Qty | Unit List Price |
|-----------|-----------------------------|-----|-----------------|
| 1265 4455 | Resistor, 470 ohm, 5%, 1/2w | 1   | \$ .41          |
| 1703 7292 | Capacitor, 4.7ufd, 10%, 10v | 1   | 2.25            |

Note: About 3" of wire (1142 8075) required.

**INSTRUCTIONS:**

1. Cut etching on Keyboard logic board assembly between G2-C and G3-K and add a wire from G2-C to G2-I as shown in Figure 1.
2. Remove capacitor C6 and resistor R15 and install new capacitor and new resistor as shown in Figure 1.
3. Reverse the polarity of capacitors C5, C7 and C8 as shown in Figure 1.
4. Verify the polarity of C18 and reverse if necessary.
5. Redline Keyboard logic diagram 1725 4640 (1 of 2) as shown in Figure 2.

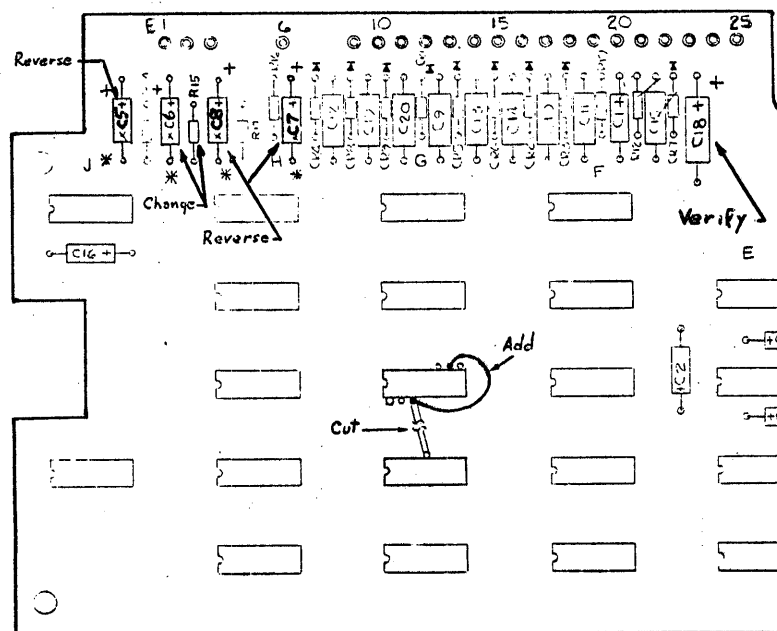
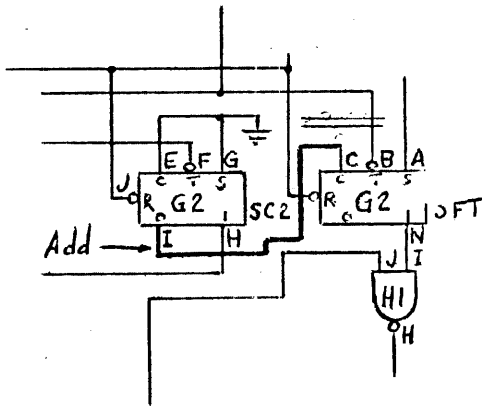
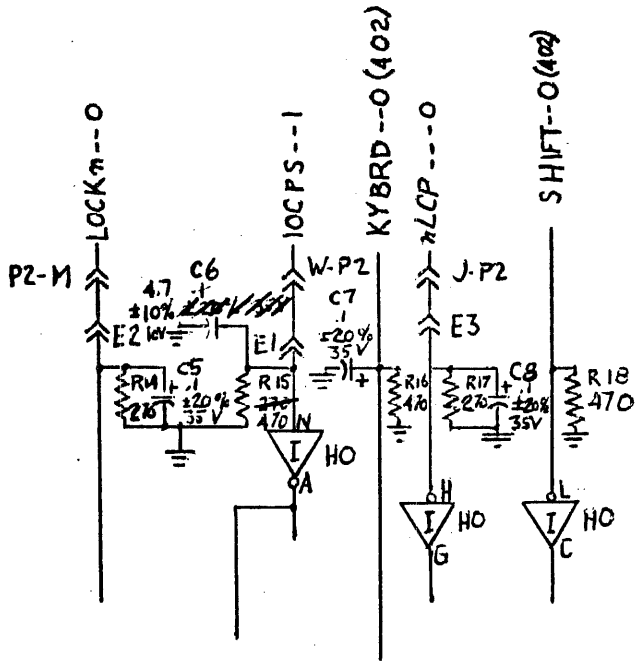


Figure 1

| REVISIONS |     |                        |      |          |
|-----------|-----|------------------------|------|----------|
| ZONE      | LTR | DESCRIPTION            | DATE | APPROVED |
| A         |     | C 72406, SEE SHEET 2   |      |          |
| B         |     | C 72512                |      |          |
| C         |     | SEE SHEET 2 FOR C72794 |      |          |
| D         |     | C-73859                | RR   |          |



401

A

**Burroughs Corporation**  
DEFENSE, SPACE AND SPECIAL SYSTEMS GROUP  
 PAOLI, PENNSYLVANIA

|       |                |              |
|-------|----------------|--------------|
| DATE  | 1-25-69        |              |
| Y     | 2-20-69        |              |
|       |                |              |
|       |                |              |
|       |                |              |
| SIZE  | CODE IDENT NO. |              |
| D     | 08241          | 1725 4640    |
| SCALE | ONE            | SHEET 1 OF 2 |

Figure 2

|   |                              |                                  |                             |                     |
|---|------------------------------|----------------------------------|-----------------------------|---------------------|
| <b>Burroughs</b><br><br>FIELD ENGINEERING<br>TECHNICAL OPERATIONS |                              | RELIABILITY                      | SYSTEM SERIES<br>Series 'B' | <b>No.</b> 4187-014 |
|   |                              | IMPROVEMENT                      | STYLE/MODEL<br>B9351/B9353  |                     |
|   |                              | NOTICE                           | TOP UNIT NO.<br>See below   |                     |
| STD. INSTALL. TIME<br>Up to 1 hour                                | UNITS AFFECTED<br>Below 7400 | UNIT DESCRIPTION<br>Control Unit |                             |                     |
| TITLE<br>Improper Cursor Positioning During Variable Tab          |                              |                                  |                             | DATE<br>3-4-71      |

Top Unit No. 1701 4697, 1701 4705, 1701 4713, 1725 8500

Note: Check Unit - RIN may have been previously installed.

CONDITION: In some units, the cursor will not move to the correct horizontal position when Variable Tab is evoked.

CAUSE: Gating level, TVS, is not of sufficient duration to allow the complete transfer of the Variable Tab setting to the Horizontal Position Counter.

CORRECTION: Add a capacitor to delay the turn-off of TVS.

PARTS REQUIRED:

| <u>Number</u> | <u>Description</u>                   | <u>Qty</u> | <u>Unit List Price</u> |
|---------------|--------------------------------------|------------|------------------------|
| 1700 8723     | Capacitor, 470pfd, $\pm 10\%$ , 100v | 1          | \$1.70                 |

Approx. 6" of wire, P/N 1142 8075, required.

INSTRUCTIONS:

1. Modify card type 050 (location 103) as shown in Figure 1.
2. Redline the logic diagram (1725 6165) as shown on page 2.

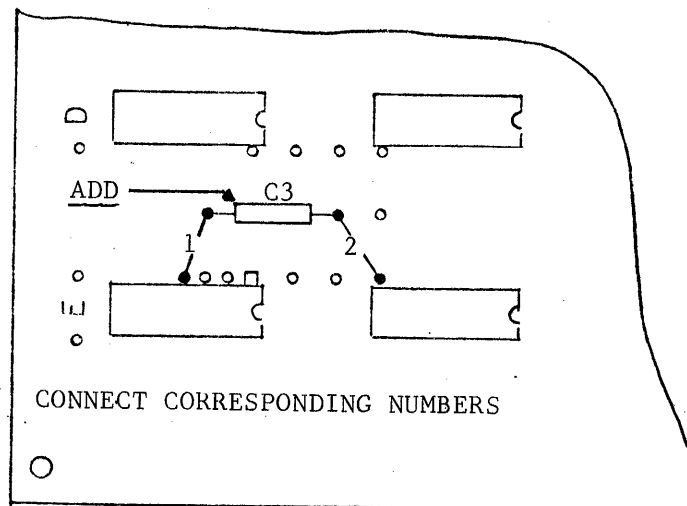
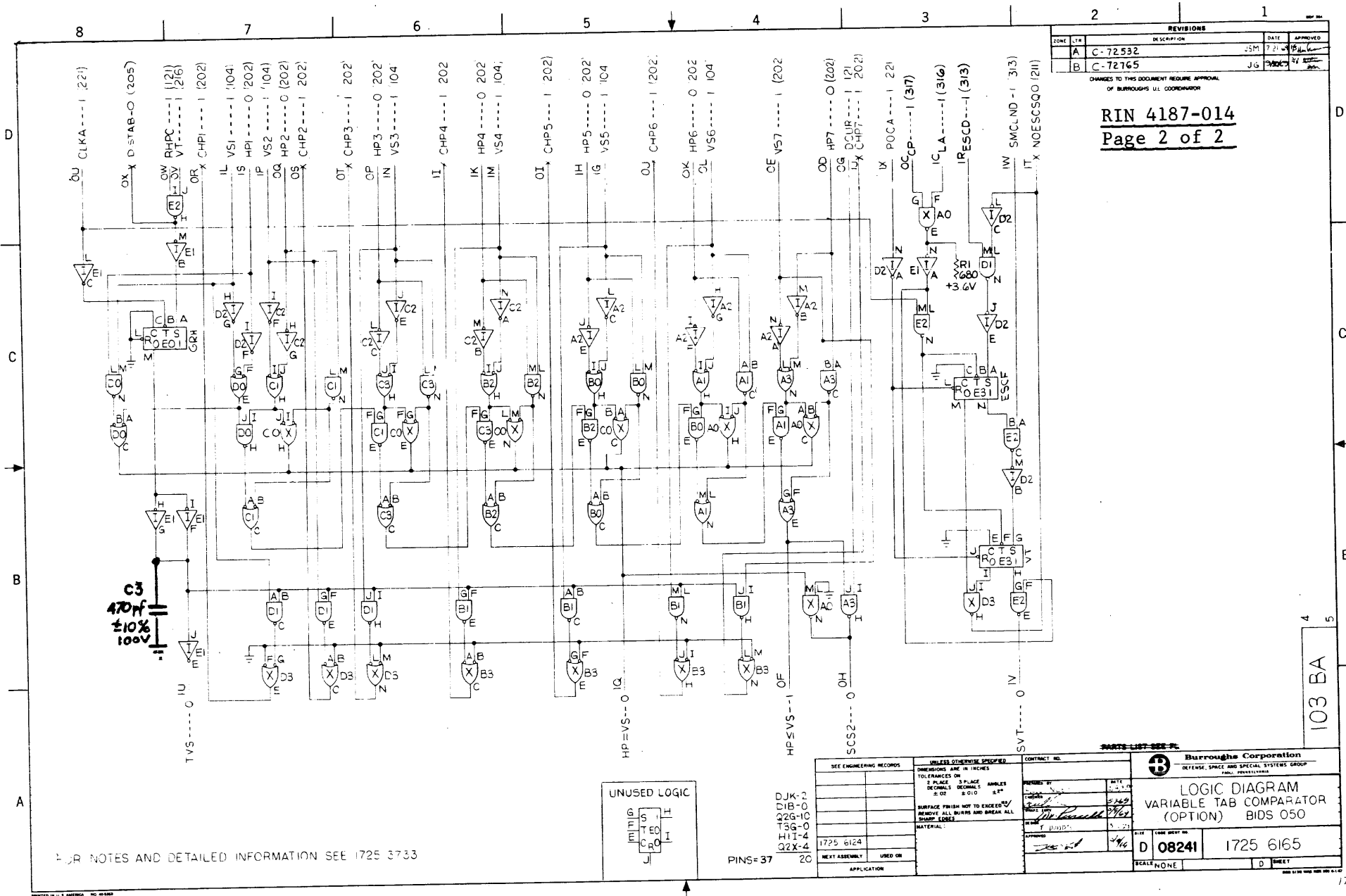


Figure 1



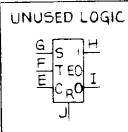


| REVISIONS |         |             |          |
|-----------|---------|-------------|----------|
| ZONE      | DATE    | DESCRIPTION | APPROVED |
| A         | 7-21-64 | C-72532     | JSM      |
| B         |         | C-72165     | JG       |

RIN 4187-014  
Page 2 of 2

103 BA

FOR NOTES AND DETAILED INFORMATION SEE 1725 3733



DJK-2  
DIB-0  
22G-10  
TSG-0  
HIT-4  
22X-4

PINS= 37 20

|                         |  |              |
|-------------------------|--|--------------|
| SEE ENGINEERING RECORDS | UNLESS OTHERWISE SPECIFIED                 | CONTRACT NO. |
|                         | DIMENSIONS ARE IN INCHES                   |              |
|                         | TOLERANCES ON                              |              |
|                         | 2 PLACES 3 PLACES                          |              |
|                         | DECIMALS DECIMALS                          |              |
|                         | ± .02 ± .01                                |              |
|                         | SURFACE FINISH NOT TO EXCEED .001          |              |
|                         | REMOVE ALL BURRS AND BREAK ALL SHARP EDGES |              |
| MATERIAL:               |  |              |
| 1725 6124               |  |              |
| NEXT ASSEMBLY           | USED ON                                    |              |
| APPLICATION             |  |              |

|            |         |
|------------|---------|
| PRINTED BY | DATE    |
| CHAS       | 5-24-64 |
| BY         | 5-22-64 |
| BY         | 5-22-64 |
| BY         | 5-22-64 |

|  |              |
|--|--------------|
| Burrhoughs Corporation                   |              |
| DEFENSE, SPACE AND SPECIAL SYSTEMS GROUP |              |
| PHILADELPHIA, PENNSYLVANIA               |              |
| LOGIC DIAGRAM                            |              |
| VARIABLE TAB COMPARATOR                  |              |
| (OPTION) BIDS 050                        |              |
| FILE                                     | DATE SHIPPED |
| D 08241                                  | 1725 6165    |
| SCALE: NONE                              | SHEET: D     |

|   |                                |                                  |                     |
|---|--------------------------------|----------------------------------|---------------------|
| <b>Burroughs</b><br><br><b>FIELD ENGINEERING<br/>TECHNICAL OPERATIONS</b> | <b>R</b> ELIABILITY            | SYSTEM SERIES<br>Series 'B'      | <b>No.</b> 4187-015 |
|   | <b>I</b> MPROVEMENT            | STYLE/MODEL<br>B9351/B9353       | PAGE<br>1 of 2      |
|   | <b>N</b> OTICE                 | TOP UNIT NO.<br>See below        |                     |
| STD. INSTALL. TIME<br>Up to 1 hour  | UNITS AFFECTED<br>below serial | UNIT DESCRIPTION<br>Control Unit |                     |
| TITLE<br>Marginal Adjustment of RTT                                       |                                |                                  | DATE<br>3-4-71      |

Note: Check unit - change may have been previously installed.

Top Unit No. 1701 4697, 1701 4705, 1701 4713, 1725 8500

CONDITION: In many units, the 3 second timer can not be adjusted properly.

CAUSE: Range of adjustment inadequate.

CORRECTION: Change value of resistor R23 to 180K and variable resistor R24 to 1 Megohm.

**PARTS REQUIRED:**

| <u>Number</u> | <u>Description</u>       | <u>Qty</u> | <u>Unit List Price</u> |
|---------------|--------------------------|------------|------------------------|
| 1265 6310     | Resistor, 180K, 5%, 1/2w | 1          | \$1.13                 |
| 1721 4156     | Resistor, Var., 1 Meg    | 1          | 2.15                   |

**INSTRUCTIONS:**

1. Modify card Type RTT or 1TT (location 310) as shown in Figure 1.
2. Redline Schematic Diagram 1725 7478 (RTT) or 1732 6430 (1TT) as shown in Figure 2.
3. Adjust R24 according to the procedure called out in Section IV, Adjustments, page 5, of the Technical Manual. The time out should be adjusted to 3 seconds, not 4.

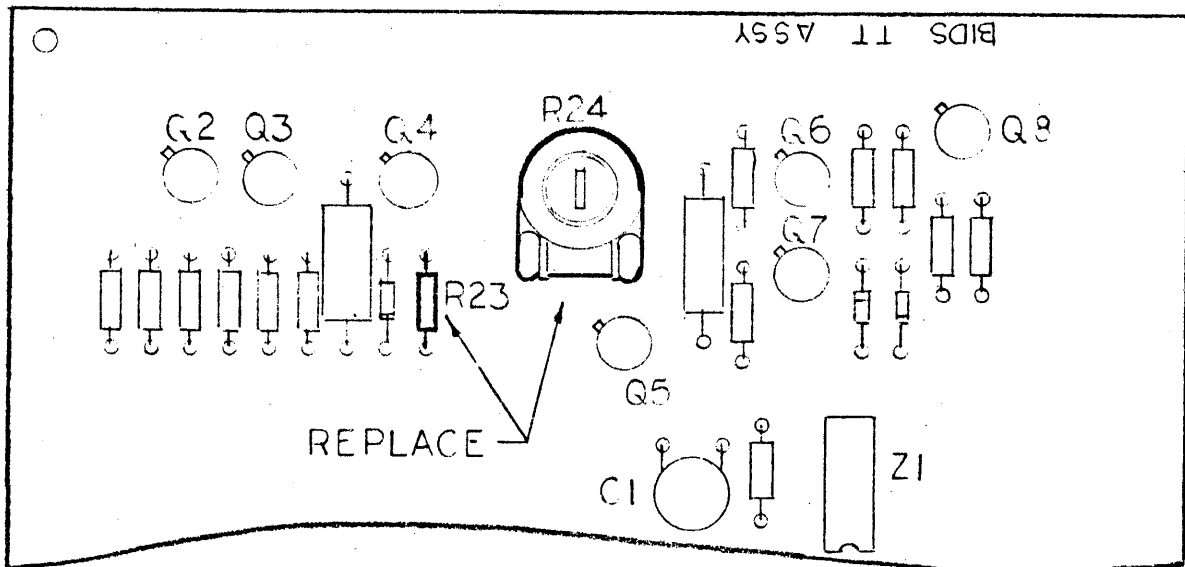


Figure 1

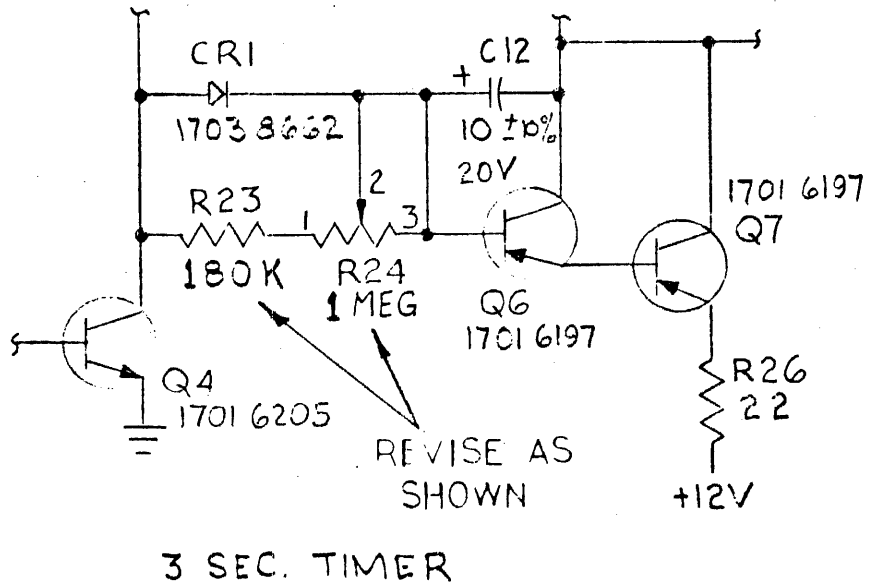


Figure 2

|   |   |                                    |                       |
|---|---|------------------------------------|-----------------------|
| <b>Burroughs</b><br><br><b>FIELD ENGINEERING<br/>TECHNICAL OPERATIONS</b> | <b>RELIABILITY</b>                          | <b>SYSTEM SERIES</b><br>Series 'B' | <b>No.</b> 4187-016   |
|   | <b>IMPROVEMENT</b>                          | <b>STYLE/MODEL</b><br>B9351/B9353  |                       |
|   | <b>NOTICE</b>                               | <b>TOP UNIT NO.</b><br>1700 1181   |                       |
| <b>STD. INSTALL. TIME</b><br>Up to 2 hours                                | <b>UNITS AFFECTED</b><br>Below serial #6400 | <b>UNIT DESCRIPTION</b><br>Monitor |                       |
| <b>TITLE</b><br>Stabilization of Power Supplies (ECN C73963)              |   |                                    | <b>DATE</b><br>5-6-71 |

Note: Check Unit - RIN may have been previously installed.

CONDITION: A marginal fusing condition exists which results in the intermittent failure of fuses F6 and F7.

CORRECTION: Add a resistor between the +28v and -28v power supplies.

**PARTS REQUIRED:**

| <u>Number</u> | <u>Description</u>          | <u>Qty</u> | <u>Unit List Price</u> |
|---------------|-----------------------------|------------|------------------------|
| 1706 8594     | Screw, Self Tap #2-32 x .25 | 2          | \$ .09                 |
| 1736 9901     | Resistor, 120 ohm, 50 w     | 1          | 4.30                   |

About 1 foot of wire (1597 8505) #20 gauge, stranded required.

**INSTRUCTIONS:**

1. Remove the CRT from the monitor chassis. Do not carry the CRT by the neck of the tube.
2. At the rear of the chassis, position the resistor as shown in Figure 1.
3. Using the resistor as a template, mark the chassis and drill two, 5/64" holes.

**CAUTION**

Do Not Drill Into Cables  
Underneath the chassis

4. Mount the resistor using the two (2) self tapping screws.
5. Wire the resistor as shown in Figure 1.
6. Install the CRT on the monitor chassis. Adjust as required.
7. Redline the monitor chassis schematic (1700 0761, sht 1) as shown in Figure 2.

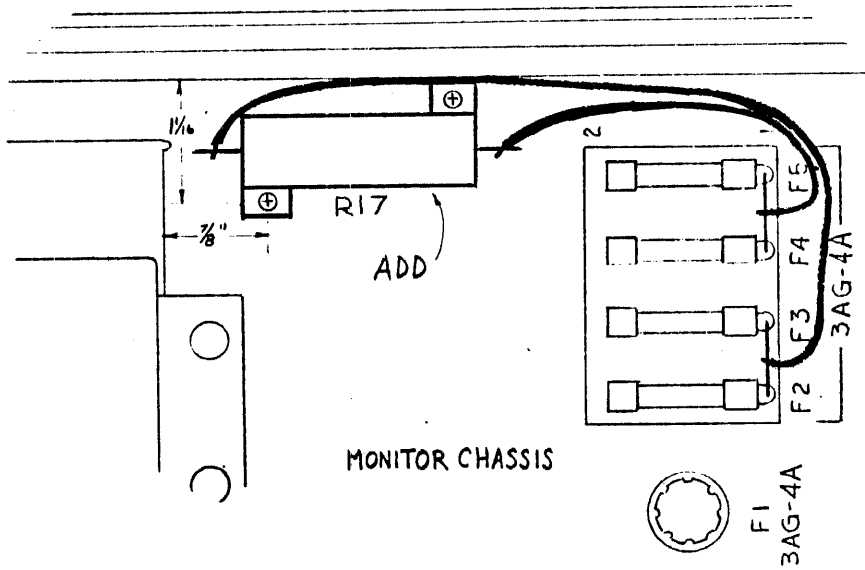
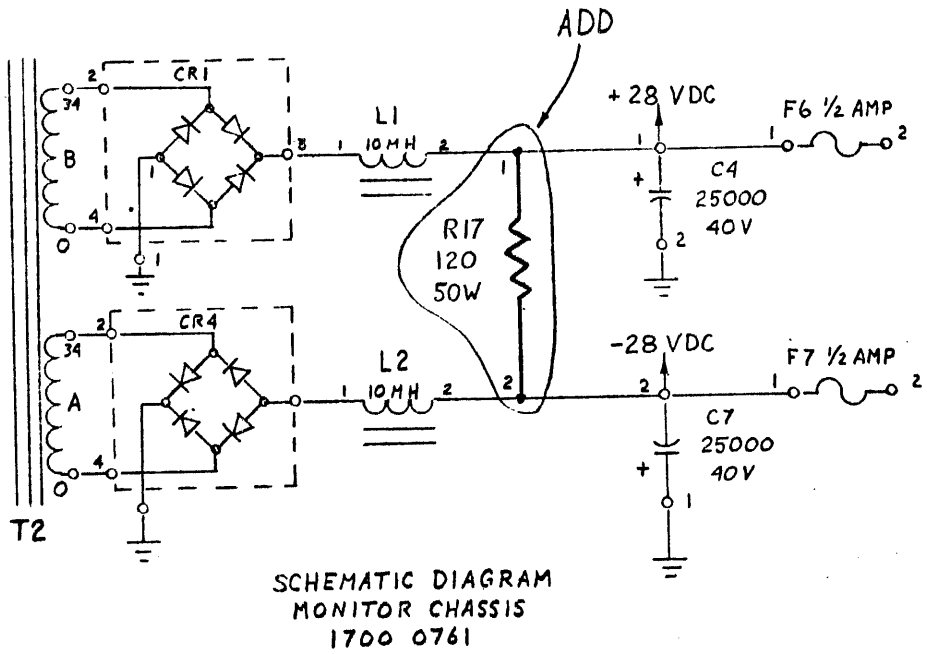


Figure 1



SCHEMATIC DIAGRAM  
 MONITOR CHASSIS  
 1700 0761

Figure 2

|   |                                     |  |                             |                                  |
|---|-------------------------------------|--|-----------------------------|----------------------------------|
| <b>Burroughs</b><br><br><b>FIELD ENGINEERING<br/>TECHNICAL OPERATIONS</b> |                                     | <b>R</b> ELIABILITY  | SYSTEM SERIES<br>Series 'B' | <b>No.</b> 4187-017              |
|   |                                     | <b>I</b> MPROVEMENT  | STYLE/MODEL<br>B9351/B9353  |                                  |
|   |                                     | <b>N</b> OTICE   | TOP UNIT NO.<br>See below   | UNIT DESCRIPTION<br>Control Unit |
| STD. INSTALL. TIME<br>Up to 1.5 hours                                     | UNITS AFFECTED<br>Below serial 8100 | TITLE<br>Increase Noise Immunity - Data Set Interface (ECN C74033) |                             | DATE<br>7-16-71                  |

Top Unit No. 1701 4697, 1701 4705, 1701 4713 and 1725 8500

CONDITION: Intermittent errors.

CAUSE: Line noise being detected in the receiver circuits.

CORRECTION: Modify the DSI receiver circuits to increase noise immunity.

**PARTS REQUIRED:**

| <u>Number</u> | <u>Description</u>                    | <u>Qty</u> | <u>Unit List Price</u> |
|---------------|---------------------------------------|------------|------------------------|
| 1265 4737     | Resistor, 1100 ohm 5% $\frac{1}{2}$ w | 5          | \$ .85                 |
| 1265 5478     | Resistor, 12K 5% $\frac{1}{2}$ w      | 5          | .23                    |

**INSTRUCTIONS:**

1. From the DSI card (location 311) remove resistor R3 (39K) from circuits N1 thru N5 as shown in Figure 1.
2. On the same card, change resistor R2 (36K) to 1100 ohms (P/N 1265 4737) in circuits N1 thru N5. Change resistor R8 (2400) to 12K (P/N 1265 5478) in circuits N1 thru N5.
3. Redline the schematic diagram for location 311 as shown in Figure 2. Note that the circuit for receiver N6 is added because no changes were made to that circuit.
4. Return the card to the unit and test 'On-Line'.

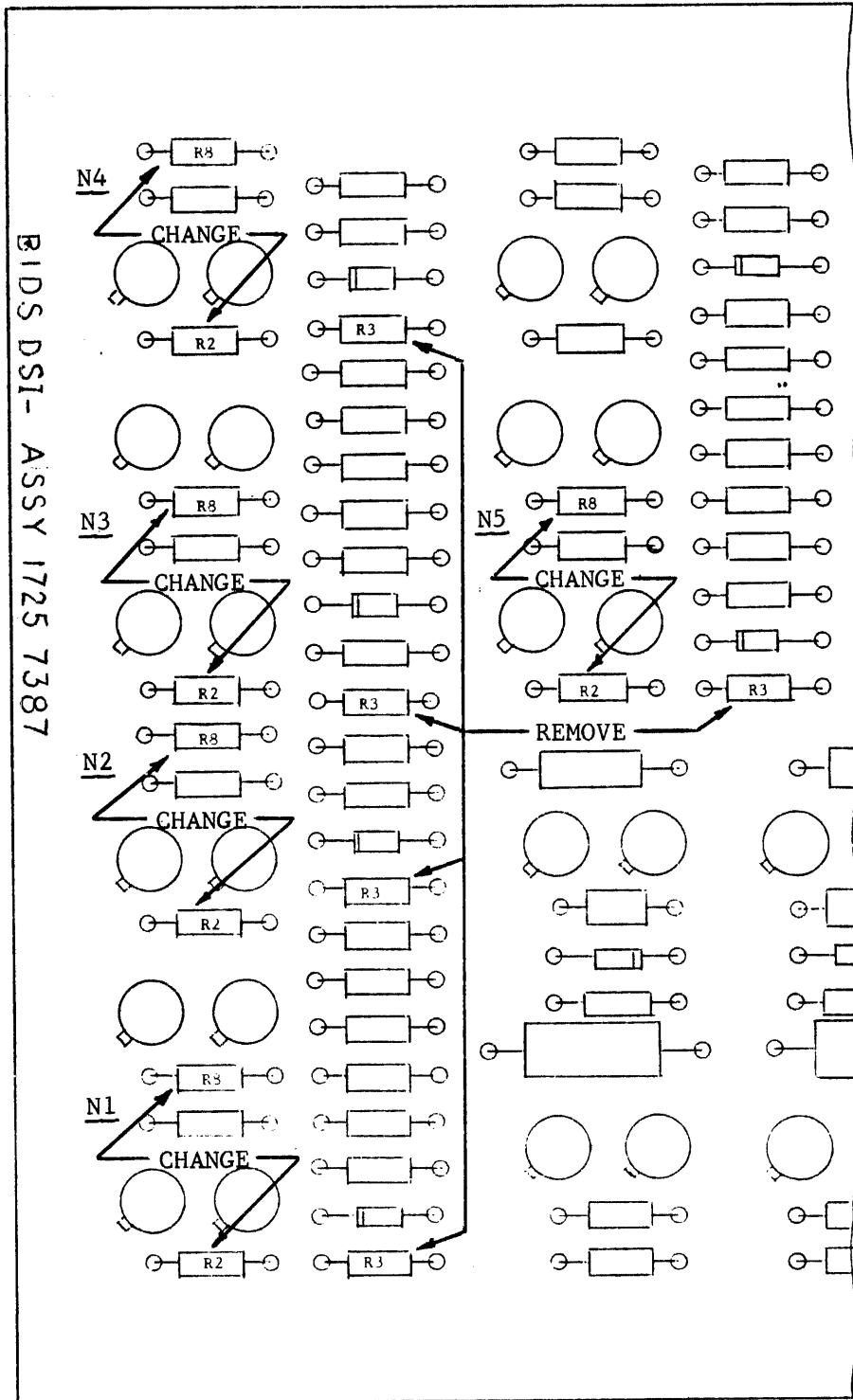


Figure 1

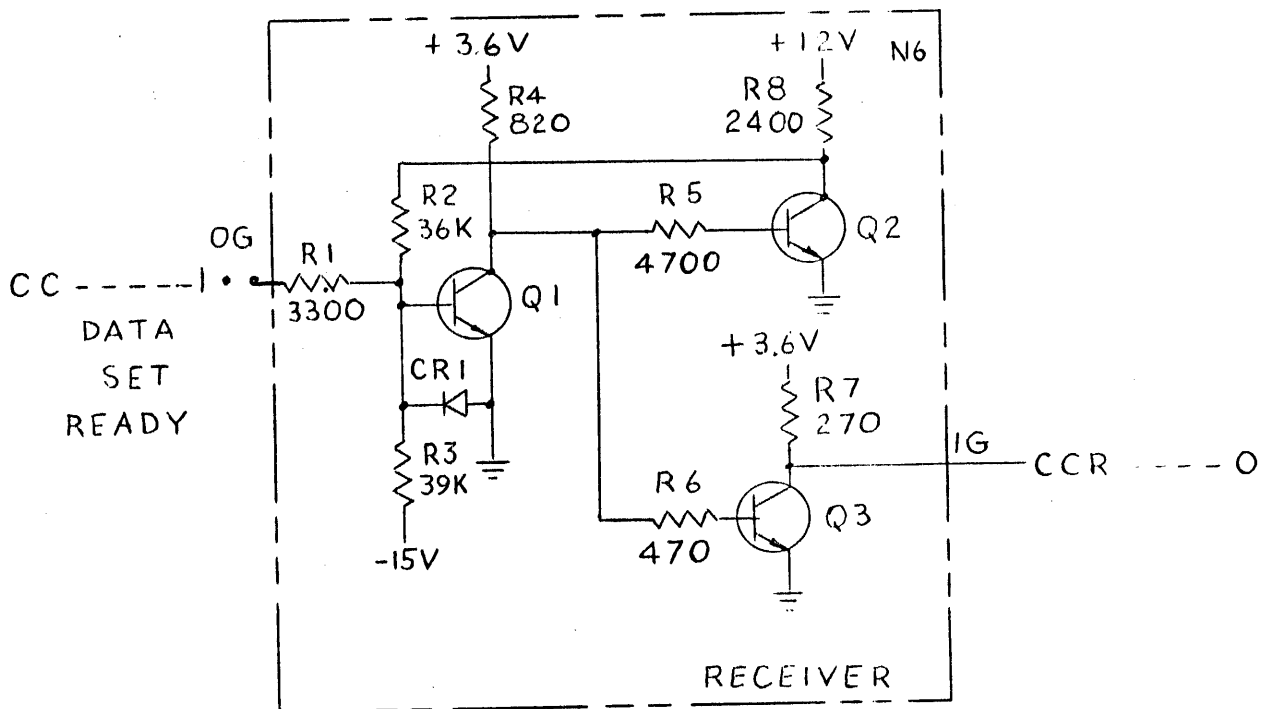
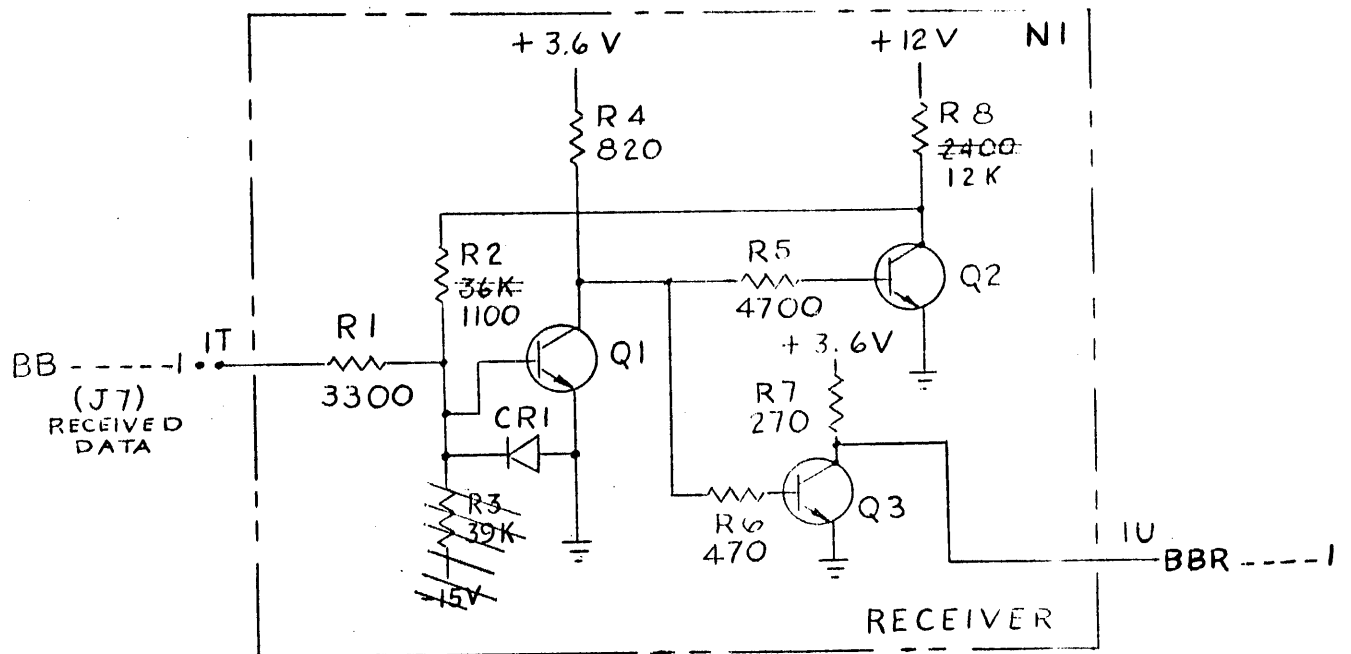


Figure 2



|   |                                   |                                  |                     |
|---|-----------------------------------|----------------------------------|---------------------|
| <b>Burroughs</b><br><br><b>FIELD ENGINEERING<br/>TECHNICAL OPERATIONS</b> | <b>RELIABILITY</b>                | SYSTEM SERIES<br>Series 'B'      | <b>No.</b> 4187-018 |
|   | <b>IMPROVEMENT</b>                | STYLE/MODEL<br>B9351/B9353       | PAGE<br>1 of 2      |
|   | <b>NOTICE</b>                     | TOP UNIT NO.<br>See below        |                     |
| STD. INSTALL. TIME<br>Up to 1 hour  | UNITS AFFECTED<br>Below S/N 14000 | UNIT DESCRIPTION<br>Control Unit |                     |
| TITLE<br>Line Splitting Correction  |                                   | (C74263)                         | DATE<br>9-14-71     |

Top Unit No. 1701 4697, 1701 4705, 1701 4713, 1725 8500

CONDITION: Extra, blank, lines are inserted during Display cycles that occur when loading from the Processor and the Cursor is positioned over NL.

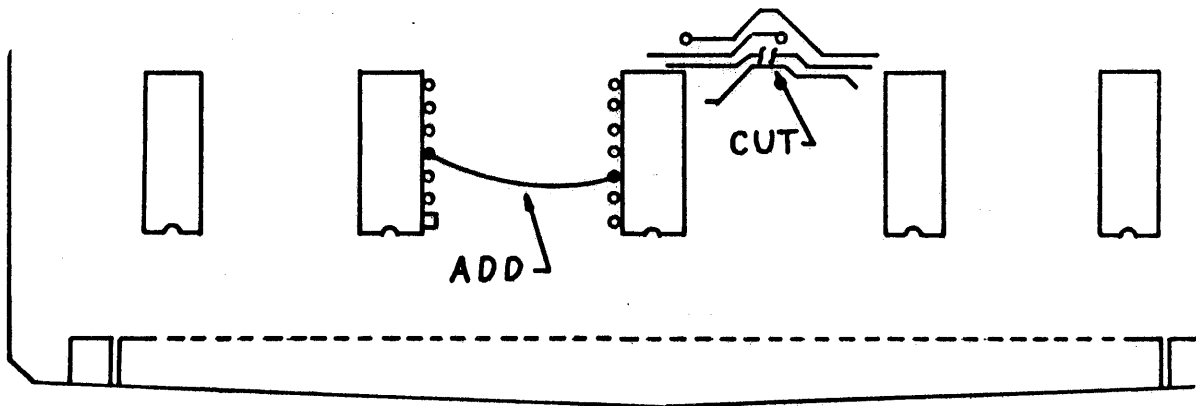
CAUSE: Logic error.

CORRECTION: Remove the level CSB9 from the logic used to generate CTTS.

PARTS REQUIRED: None

INSTRUCTIONS:

1. Modify card type 215, location 214, by making one etching cut and adding one jumper. Refer to Figure 1.
2. Redline the Logic Diagram for location 214 as shown in Figure 2.



Cut - B2-C to C3-L      Jumper - C3-L to D3-D (gnd)

Figure 1

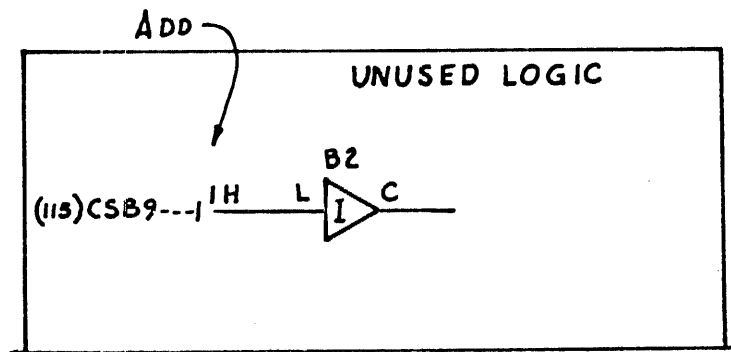
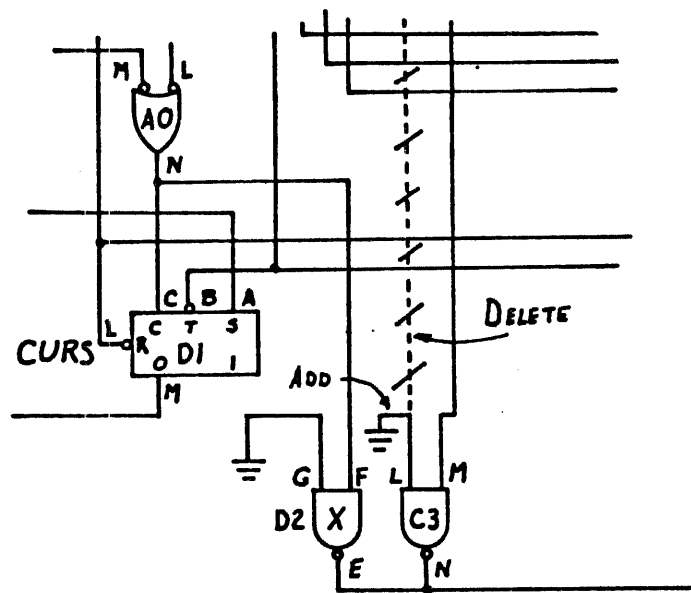


Figure 2

|   |                                     |                                  |                     |
|---|-------------------------------------|----------------------------------|---------------------|
| <b>Burroughs</b><br><br><b>FIELD ENGINEERING<br/>TECHNICAL OPERATIONS</b> | <b>RELIABILITY</b>                  | SYSTEM SERIES<br>Series 'B'      | <b>No.</b> 4187-019 |
|   | <b>IMPROVEMENT</b>                  | STYLE/MODEL<br>B9351/B9353       | PAGE<br>1 of 3      |
|   | <b>NOTICE</b>                       | TOP UNIT NO.<br>See below        |                     |
| STD. INSTALL. TIME<br>Up to 1 hour  | UNITS AFFECTED<br>Below serial 2000 | UNIT DESCRIPTION<br>Control Unit |                     |
| TITLE<br>+3.6V Power Supply Improvement                                   |                                     |                                  | DATE<br>10-21-71    |

Top Unit No. 1701 4697, 1701 4705, 1701 4713, 1725 8500

Note: Check Unit - RIN may have been previously installed.

CONDITION: Circuit Breaker, CB3, trips occasionally when unit is powered on.

CAUSE: Intermittent triggering of the overvoltage sensing circuit.

CORRECTION: Add a capacitor network to the circuit involving CR10 and add a zener diode to establish a better reference voltage for the gate.

**PARTS REQUIRED:**

| <u>Number</u> | <u>Description</u>        | <u>Qty</u> | <u>Unit List Price</u> |
|---------------|---------------------------|------------|------------------------|
| 1265 4380     | Resistor, 360 ohm, 5%, ½w | 1          | \$ .23                 |
| 1265 8860     | Resistor, 120 ohm, 5%, ½w | 1          | .69                    |
| 1736 7772     | Capacitor, .01ufd., 1KV   | 3          | .86                    |
| 1736 8143     | Diode, Zener, 1N750A      | 1          | 1.15                   |

**INSTRUCTIONS:**

1. Modify the component board assembly, as shown in Figure 1, and located on the power supply gate, by replacing R22 with the zener diode CR11 (1736 8143). The cathode end is connected to the gate (G) of CR10. Change R21 from 4.3K to 360 ohms (1265 4380). Add capacitors C9, C10 and C11 as shown. Note that components R20, CR7 and CR8 may or may not be present. If they are, they may be removed or left as is. The circuit is not being used.
2. Change resistor R8, on the voltage adjustment panel, Figure 2, from 300 ohm to 120 ohms (1265 8860) as shown to provide a better adjustment range for the +3.6 volt supply.
3. Apply power to the unit and check the +3.6VDC on the backplane at pin DEOA/1A. Adjust R7 as required.
4. Redline sheet 2 of the schematic diagram, 1700 5505, as shown in Figure 3.

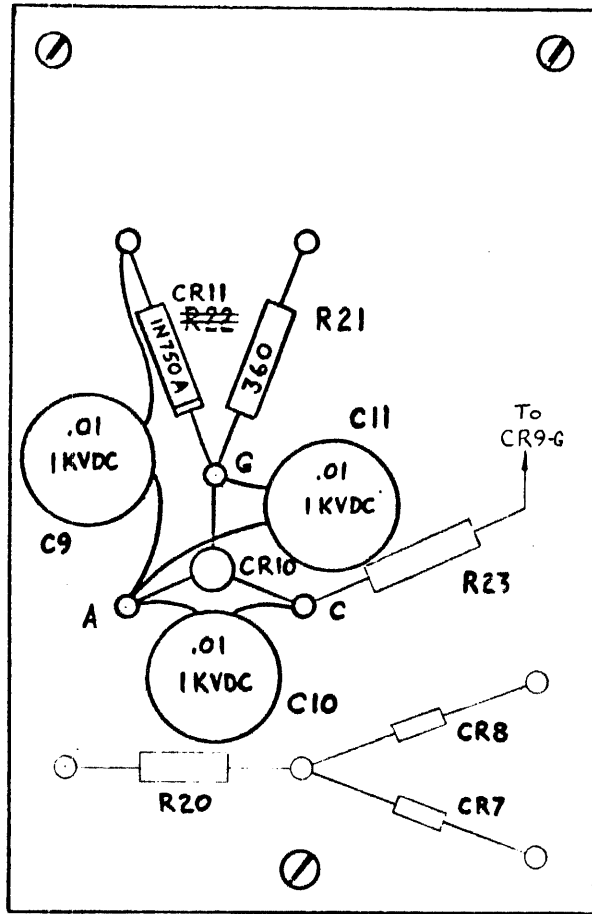


Figure 1

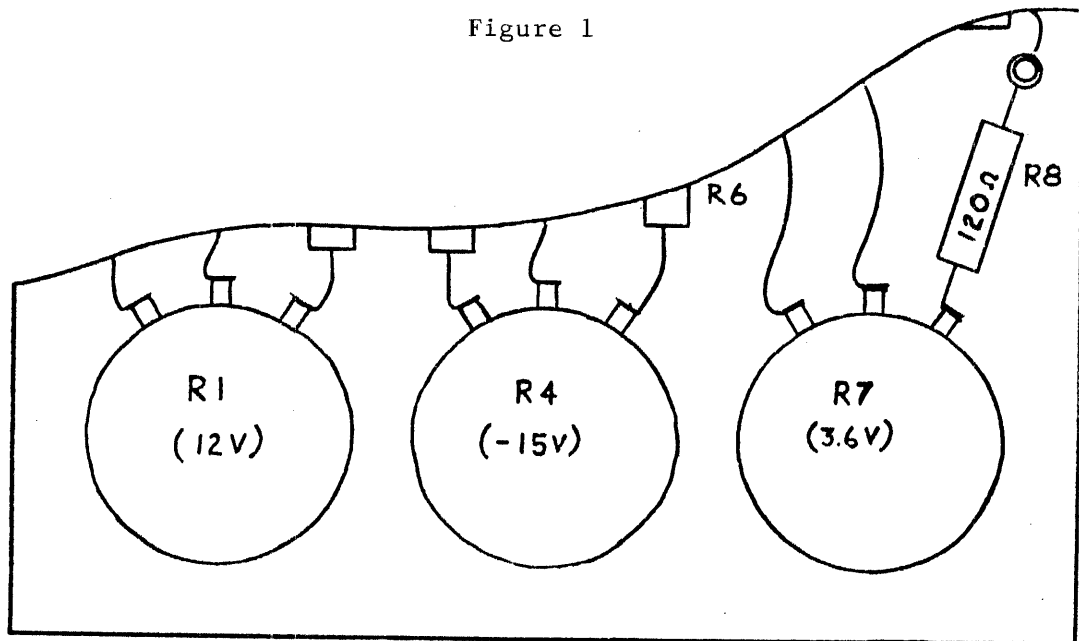


Figure 2

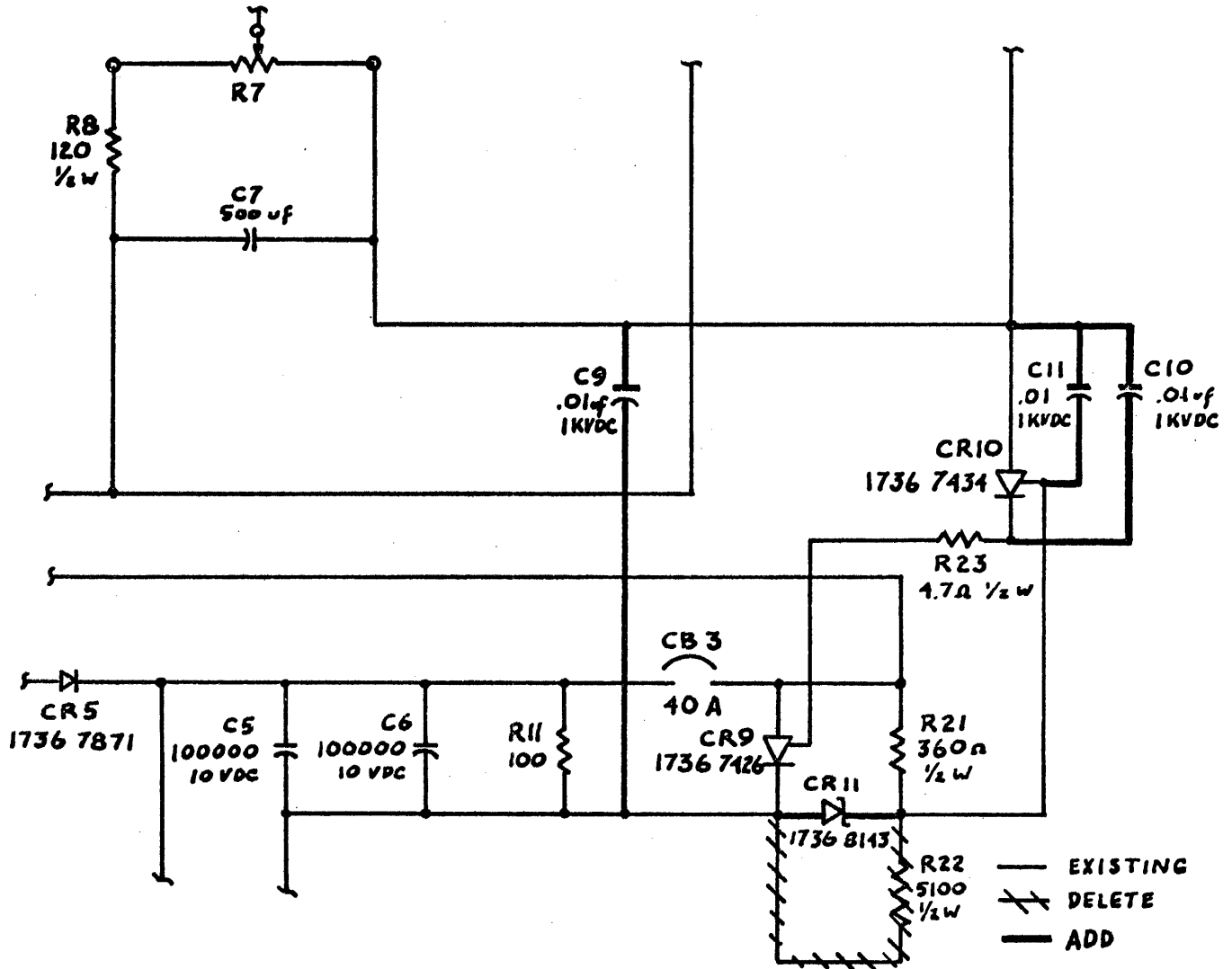


Figure 3

|   |                                      |                                  |                     |
|---|--------------------------------------|----------------------------------|---------------------|
| <b>Burroughs</b><br><br><b>FIELD ENGINEERING<br/>TECHNICAL OPERATIONS</b> | <b>R</b> ELIABILITY                  | SYSTEM SERIES<br>Series 'B'      | <b>No.</b> 4187-020 |
|   | <b>I</b> MPROVEMENT                  | STYLE/MODEL<br>B9351/B9353       | PAGE<br>1 of 1      |
|   | <b>N</b> OTICE                       | TOP UNIT NO.<br>See Below        |                     |
| STD. INSTALL. TIME<br>Up to 1 hour  | UNITS AFFECTED<br>Below Serial 14185 | UNIT DESCRIPTION<br>Control Unit |                     |
| TITLE<br>Grounding of Levels DB and DD (C-74339)                          |                                      |                                  | DATE<br>11-2-71     |

Top Unit No. 1701 4697, 1701 4705, 1701 4713, 1725 8500

CONDITION: Data Set levels DB and DD are not properly disabled when operating in an Asynchronous mode.

CORRECTION: Install the backplane change as indicated in the instructions below.

PARTS REQUIRED: None.

Note: About 2' of wire (1142 8075) required.

**INSTRUCTIONS:**

1. Make the following backplane changes:

| * | Signal   | From   | Z | To     | Z |
|---|----------|--------|---|--------|---|
| R | DDR----0 | AAFD1I | 2 | AADB8X | 2 |
| D | DB-----1 | FDOF   | 1 | DB6W   | 1 |
| D | DD-----1 | FD1J   | 1 | DB6H   | 1 |
| A | DBR----0 | DCOM   | 2 | DB6W   | 2 |
| A | DDR----0 | DB8X   | 1 | DB6H   | 1 |

\* A = Add  
D = Delete  
R = Access

2. Redline circuit list to reflect the above changes.
3. Redline Logic Diagram 1732 6380 (Type 166) or 1740 3627 (Type 266), location 209 as shown in Figure 1.

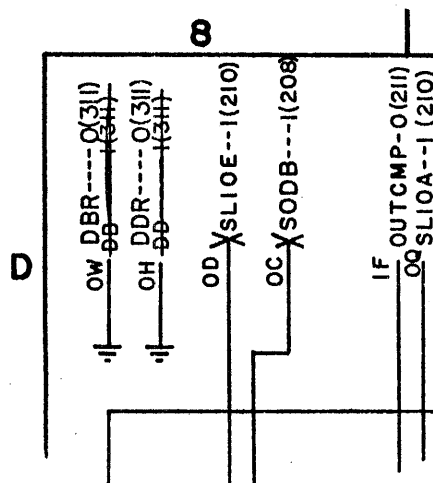


Figure 1

|   |                              |                              |                     |
|---|------------------------------|------------------------------|---------------------|
| <b>Burroughs</b><br><br>FIELD ENGINEERING<br>TECHNICAL OPERATIONS | RELIABILITY                  | SYSTEM SERIES<br>Series 'B'  | <b>No.</b> 4187-021 |
|   | IMPROVEMENT                  | STYLE/MODEL<br>B9353         |                     |
|   | NOTICE                       | TOP UNIT NO.<br>1700 1199    |                     |
| STD. INSTALL. TIME<br>less than 1 hr.                             | UNITS AFFECTED<br>below 6000 | UNIT DESCRIPTION<br>Keyboard |                     |
| TITLE<br>Non-Skid Pads for Keyboard                               |                              | (C-73430)                    | DATE<br>3-2-72      |

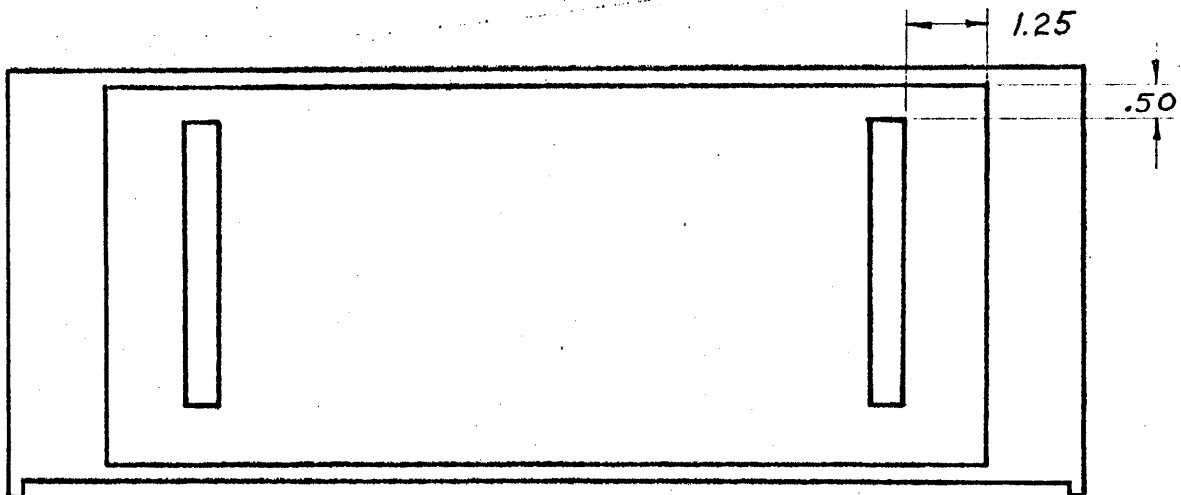
CONDITION: The Keyboard has a tendency to slide on smooth surfaces when not using the keyboard attachment kit.

CORRECTION: Apply 'Non-Skid' pads to the base of the keyboard.

PARTS REQUIRED:

| <u>Number</u> | <u>Description</u> | <u>Qty</u> | <u>Unit List Price</u> |
|---------------|--------------------|------------|------------------------|
| 1737 1402     | Pad, Keyboard      | 2          | \$.25                  |

INSTRUCTIONS: Remove the protective backing material from the pads and apply to the base of the keyboard as shown in Figure 1.



Keyboard Assembly

Figure 1

|   |                                     |                                  |                     |
|---|-------------------------------------|----------------------------------|---------------------|
| <b>Burroughs</b><br><br>FIELD ENGINEERING<br>TECHNICAL OPERATIONS | <b>RELIABILITY</b>                  | SYSTEM SERIES<br>Series 'B'      | <b>No.</b> 4187-022 |
|   | <b>IMPROVEMENT</b>                  | STYLE/MODEL<br>B9351/B9353       | PAGE<br>1 of 1      |
|   | <b>NOTICE</b>                       | TOP UNIT NO.<br>See Below        |                     |
| STD. INSTALL. TIME<br>Less than 1 hour                            | UNITS AFFECTED<br>below Serial 2000 | UNIT DESCRIPTION<br>Control Unit |                     |
| TITLE<br>Compatibility of Regulator Assemblies (C-74441)          |                                     |                                  | DATE<br>3-2-72      |

Top Unit No. 1701 4697, 1701 4705, 1701 4713, 1725 8500

PREREQUISITE: RIN 4187-019

CONDITION: Incompatibility of new Regulator Assemblies with early Power Supplies.

CORRECTION: Replace resistor R16 as outlined below.

PARTS REQUIRED:

| <u>Number</u> | <u>Description</u>  | <u>Qty.</u> | <u>Unit List Price</u> |
|---------------|---------------------|-------------|------------------------|
| 1265 4414     | Res. 390 ohm ½W, 5% | 1           | \$ .23                 |

INSTRUCTIONS:

1. Remove Regulator card from Power Supply gate.
2. Replace R16 with a 390 ohm ½W Resistor, 1265 4414.
3. Redline schematic as shown in figure 1.

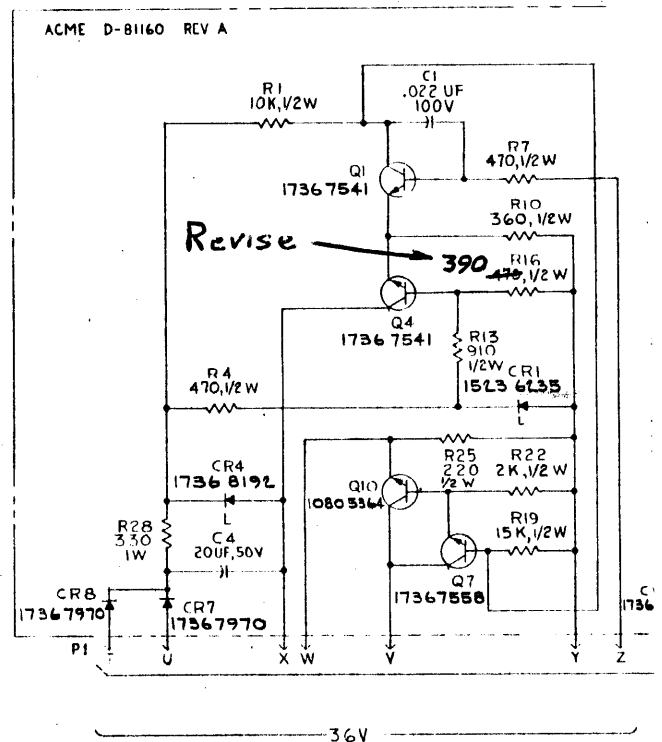
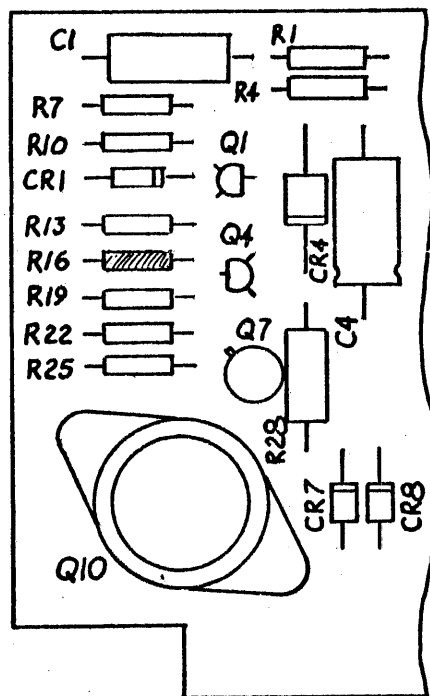


Figure 1



|   |                       |                                  |                     |
|---|-----------------------|----------------------------------|---------------------|
| <b>Burroughs</b><br><br><b>FIELD ENGINEERING<br/>TECHNICAL OPERATIONS</b> | <b>R</b> ELIABILITY   | SYSTEM SERIES<br>B               | <b>No.</b> 4187-023 |
|   | <b>I</b> MPROVEMENT   | STYLE/MODEL<br>B9351/B9353       | PAGE<br>1 OF 3      |
|   | <b>N</b> OTICE        | TOP UNIT NO.<br>See Below        |                     |
| STD. INSTALL. TIME<br>Less than 1 hour                                    | UNITS AFFECTED<br>All | UNIT DESCRIPTION<br>Control Unit |                     |
| TITLE<br>Contention Mode Correction                                       |                       |                                  | DATE                |

TOP UNIT #1701 4697, 1701 4705, 1701 4713, 1725 8500

**CONDITION:** After transmitting a message, the unit remains in the transmit mode.

**CAUSE:** Logic Error

**CORRECTION:** Modify card type 242 according to instructions below.

**PARTS REQUIRED:** None

**NOTE:** About 2' of wire (1142 8075) required.

**INSTRUCTIONS:** 1. Make the following backplane changes:

| * | SIGNAL   | FROM | Z | TO   | Z |
|---|----------|------|---|------|---|
| R | PARERR-0 | DC5D | 2 | FE0D | 2 |
| A |          | FD8F | 1 | DC5D | 1 |

\* A = Add    R = Access (Remove/Replace)

2. Modify card type 242 (Location 315) by making one etching cut and adding three jumpers. Refer to Figure 1.
3. Relabel the card as type 342 and part number 1743 5538.
4. Replace the Logic Diagram for Location 315 with Page 3 of this RIN.
5. Redline the Card Allocation Chart, 1738 6533 or 1743 4069, to reflect the new part number.
6. Redline Circuit List to reflect above backplane change.





|   |                       |                                  |                     |
|---|-----------------------|----------------------------------|---------------------|
| <b>Burroughs</b><br><br>FIELD ENGINEERING<br>TECHNICAL OPERATIONS | <b>R</b> ELIABILITY   | SYSTEM SERIES<br>B               | <b>No.</b> 4187-024 |
|   | <b>I</b> MPROVEMENT   | STYLE/MODEL<br>B9351/B9353       | PAGE<br>1 of 1      |
|   | <b>N</b> OTICE        | TOP UNIT NO.<br>See Below        |                     |
| STD. INSTALL. TIME<br>Less than 1 hour                            | UNITS AFFECTED<br>All | UNIT DESCRIPTION<br>Control Unit |                     |
| TITLE<br>Printing Multiple Characters (ECN 74636)                 |                       |                                  | DATE<br>11-16-72    |

TOP UNIT #1701 4697, 1701 4705, 1701 4713, 1725 8500

**CONDITION:** Printing multiple characters with single depression of keyboard switch.

**CAUSE:** Logic error.

**CORRECTION:** Modify backplane according to instructions below.

**PARTS  
REQUIRED:** None

**NOTE:** About 3' of wire (1142 8075) required.

**INSTRUCTIONS:** 1. Make the following backplane changes:

| * | SIGNAL   | FROM | Z | TO   | Z |
|---|----------|------|---|------|---|
| D | EDP----1 | DC9G | 2 | BC2X | 2 |
| D | EDP----1 | DA4X | 1 | DC9G | 1 |
| A | EDP----1 | DA4X | 2 | BD2N | 2 |
| A | STDP---1 | BC3S | 2 | DC9G | 2 |

\* D = Delete, A = Add

2. Redline card type 216 (location 215) as shown in Figure 1.

3. Redline circuit list to reflect above backplane change.

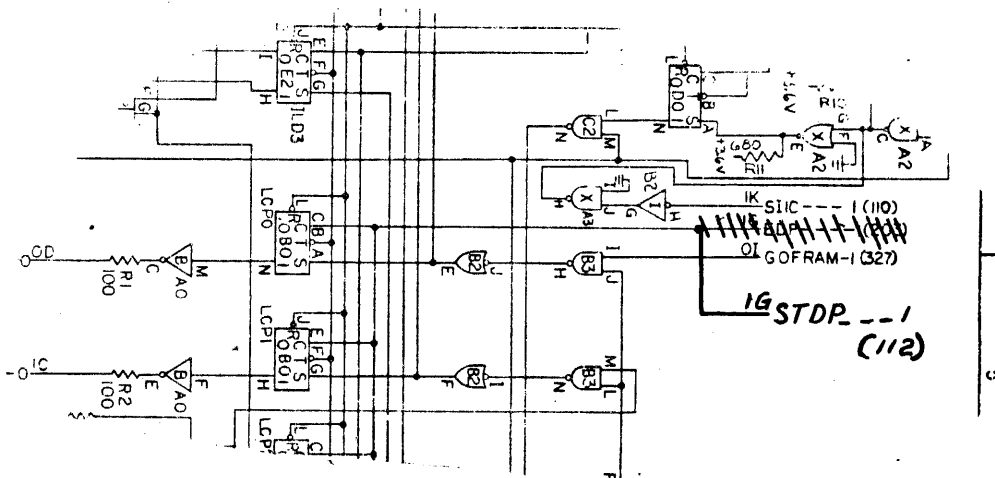


FIGURE 1.

Optional Features**AUTOMATIC ITEM CORRECTION (AIC) OPTION**GENERAL

To provide a convenient, time saving and efficient operator assist function for use in display applications involving record update, editing and item correction, the Automatic Item Correction feature was developed. Although primarily directed at banking Item Processing Systems, the features to be described are not limited to that application.

Included in this feature is a right justification provision to automatically position data to a specified right hand column or margin. This function, when used with the auxiliary numeric only keyboard, enables the operator to input data without regard for the least significant digit position and without the need to insert leading zeros or spaces to determine this position. Field length is completely variable and is under processor (program) control. Definition of the field is that space bounded on the left by a Group Separator ( $\Delta$ ) and on the right by a Record Separator ( $\triangleleft$ ).

Also included in this feature is a blinking cursor to assist the operator in identifying the characters or fields to be corrected. This function provides an effective means of locating the cursor position after each display input.

The numeric only keyboard is an extension of the existing alphanumeric keyboard. It interfaces directly with this keyboard through a 3 foot cable. No connectors are involved. It contains 13 keys; the numerals 0 through 9, a minus sign, a SEND key and a SKIP key. With the exception of the SKIP key, all keys are connected in parallel with the corresponding keys on the alphanumeric keyboard. The SKIP key, when used with the AIC option, will position the cursor to the next display position to be edited. This may be any of the following positions:

1. A position containing a question mark (?)
2. The position following a Group Separator ( $\Delta$ )
3. The position following a Unit Separator ( $\triangleright$ ).

FUNCTIONAL DESCRIPTION**AUTOMATIC ITEM CORRECTION AND RIGHT JUSTIFICATION**

This option consists of two printed circuit logic assemblies which plug directly into the B9353 (D/L7) backplane at locations 204 and 206. (This option CANNOT be installed in D/L6 units or below.) Described below are the functional characteristics of the AIC and Right Justify option.

**Error Indication**

To specify a single character error to be corrected, the processor (program) will send a Vertical Mark (1) character (1111100) to the display terminal for each

character requiring correction. Upon detection of this character, the terminal will convert it to a question mark (?) and will display the question mark at the screen position or positions to be corrected.

**Cursor Positioning**

The cursor will be positioned automatically to the next character to be corrected whenever the cursor is positioned over a question mark (?) and this character is corrected by the operator or whenever the SKIP key is depressed.

Positioning to the first screen position to be corrected, after receipt of a message, must be accomplished manually by the operator unless the Reposition Cursor option is also installed. With this option, the processor may direct the terminal to position the cursor to any illegal horizontal position. The unit will then react by positioning the cursor to the first position to be corrected. The following escape sequence will accomplish the initial Automatic Positioning:

```

E           S
S   :   (h) P
C

```

where (h) is any character from column 7 of the ASCII code chart except Delete (DEL). No time fill is required when the escape sequence is positioned immediately preceding the ETX of the message.

**Cursor Blink**

The cursor will "blink" at a rate of approximately 10 character/second when positioned over a question mark or when it is in a character position following a Group Separator ( $\Delta$ ) or Unit Separator ( $\triangleright$ ).

**Selective Send**

With the AIC option installed, the Selective Send function will operate only when a "Form" is on the screen.

**Compose Forms Switch**

Placing the Compose Forms Switch in the Compose position will disable the automatic functions of the AIC option and will allow operator generation of the display fields.

**Right Justification**

The field length for the right justify function is normally established by the program. For testing purposes, it can also be set up by the operator by using the Compose Forms switch. The length of the field is established by preceding the field with a Group Separator ( $\Delta$ ) and terminating the field with a Record Separator ( $\triangleleft$ ). The length of the field then is the number of characters between the two separators. The characters entered may be any alphanumeric character but may not be format effectors (Tab or NL). When entering a field to be right justified, the

### Optional Features

cursor will position to the character following the Group Separator. As the first character is entered, it will appear preceding the Record Separator. At the same time, the character under the cursor will be deleted, the cursor will not move. As each subsequent character is entered, the existing information in the field is shifted to the left, the new character always being at the right. Should the number of characters entered exceed the size of the field, the most significant characters will be lost. To move the cursor to the next area to be corrected, the SKIP Key is depressed.

The following is an illustration of the sequence of events when the numerals 1 thru 6 are entered in a 5 character field:

```

Δ [ ]- - - - < Start
Δ [ ]- - - 1 < 1st character entered
Δ [ ]- - 1 2 < 2nd character entered
Δ [ ]- 1 2 3 < 3rd character entered
Δ [ ] 1 2 3 4 < 4th character entered
Δ [ 1 ] 2 3 4 5 < 5th character entered
Δ [ 2 ] 3 4 5 6 < 6th character entered

```

In the event the operator makes a mistake and wishes to correct the error, the complete field must be retyped including all preceding zeros.

#### FUNCTIONAL DETAIL

##### **AUTOMATIC ITEM CORRECTION AND RIGHT JUSTIFICATION**

###### **Error Indication**

When the Vertical Mark (I) character is detected in the IOB Decoder, the levels 765---1, 43---1, N21---1 and RCVNMSG1 will all go TRUE. These four levels are ANDed to generate the level ENQME--1 (206). A Question Mark (?) is then stored in memory in the same manner that a character containing a parity error would be stored.

###### **Cursor Positioning**

Two methods are provided for moving the cursor to the FIRST position to be corrected; Programatic and Manual. With the Reposition Cursor option installed, reception of the Reposition Cursor escape sequence with the horizontal position character taken from column 7 of the ASCII code chart (except DEL) will be stored in the normal manner. This character would normally be invalid for the Reposition Cursor option. Refer to the Logic Diagram for location 204. During the next display cycle, the level RPEA---1 will go TRUE when a Question Mark (?), Unit Separator (▷) or Group Separator (Δ) is detected in MIR. RPEA ANDed with the storage flip-flops will result in the level RPCE---0 going FALSE (location 102). This signal then actuates the cursor movement logic in the normal manner.

Should the Reposition Cursor option not be installed, it is necessary that the operator depress the SKIP key on the Numeric Key Pad to move the cursor to the first location to be corrected. Detection of the SKIP decode during the edit cycle will cause Flip-flop MVE (card location 204) to be set. The next display cycle will begin with MVE set and REPQM reset. These two conditions, together with normal display cycle logic, will cause Flip-flop MVS to be set. MVS will remain set until either a Question Mark (?), Unit Separator (▷) or Group Separator (Δ) is detected in MIR. At this time MVS and MVE will both be reset.

Detection of a Question Mark (?) will also cause Flip-flop QM to be set and the level RCMAP to go TRUE. This then initiates the logic to reset the Cursor Address and replace it with the current Memory Address.

Should a Unit Separator (▷) or a Group Separator (Δ) be detected, the Flip-flop USOGS+1 is set to allow the new Cursor Address to be counted up one to position the cursor to the right of the separator. Detection of US or GS will also cause RCMAP to go TRUE as with the Question Mark. With USOGS+1 set, another Flip-flop, ACC, will be set on the next clock pulse. This flip-flop will allow the level CCAU to go TRUE which in turn will generate the Count Cursor Address pulse.

###### **Cursor Blink**

Three Flip-flops, BL1, BL2 and BL3, are used as a counter to control the repetition rate of the blinking cursor. When to allow the cursor to blink is controlled by Flip-flop BLS which is set when a Group Separator (Δ) has set Flip-flop GSS or when Flip-flop QS has been set. QS is used to remember that a Question Mark (?), Group Separator (Δ) or Unit Separator (▷) had been detected during the previous display cycle. This information is stored in bit 6 of the second control cell of the screen presently active. This bit was a spare.

The level OFF---0 will go FALSE when the three BL Flip-flops are all reset, BLS is set and DCUR is TRUE. With OFF FALSE, the Z1 and Z2 levels (location 227) will be inhibited, thereby disabling the display of the cursor for that display cycle.

###### **Right Justification**

This feature makes use of the "Character Delete" logic with two exceptions; the "Pull" cycle is terminated when a Record Separator (<) or End-of-Screen is detected, and the character being entered is stored in the last location of the field. Normally a "Delete" operation would terminate when End-of-Screen was detected and no new character would be stored. As each keyboard entry is made, the character under the cursor is deleted and memory information, up to the RS, is pulled to the left. The character being entered is then stored in the location preceding the RS character.

Optional Features

During the previous display cycle, Flip-flop GSS would have been set when the Group Separator ( $\Delta$ ) was detected. The level SCS6--0 will then go FALSE at MT3 resulting in the setting of Control Storage Bit 6. With CSB6 set and CODA--1 TRUE, the level NSTB occurring when a character is entered from the keyboard will cause Flip-flop JUST to be set. At the same time, the level SSCDL will go TRUE which will initiate the "Character Delete" operation. In this manner the character under the cursor is deleted and characters to the right are shifted left one place. Addressing for this operation makes use of both the Cursor Address Register and the Memory Address Register. The Cursor Address Register is used to access the next character and is therefore one count ahead of the Memory Address Register. The Memory Information Reg. is used to store the information accessed by the Cursor Address. It can be seen then that two memory cycles are required to "Pull" one character to the left. This operation is terminated when the Record Separator ( $\leftarrow$ ) or End-of-Screen is detected.

Detection of RS or ES will cause Flip-flop RSS to be set. With RSS and JUST both set, the levels TIDD--1 and DSRC--1 will go TRUE. TIDD allows the input character to be gated into the Display while DSRC is the clock. The Memory Data Gates are then enabled with the information in DSR and the level LDSR--0 being FALSE (locations 119 and 120). The input character is then written into memory adjacent to the RS and is therefore right justified. As with the delete operation, the original Cursor Address is accessed to keep the cursor at the same location.

In order to move the cursor out of the field, the operator presses the "Skip" key. This action will cause the cursor to be positioned to the next field or the next question mark if the "Form" option is not installed. With the "Forms" option installed cursor movement from the last Right Justified field will depend on the information following that field. The cursor may return to the first field if the form is not composed correctly.

GLOSSARY OF SIGNAL NAMES

| <u>Signal</u> | <u>Page</u> | <u>Definition</u>                               |
|---------------|-------------|---|
| AICI          | 204         | AIC Inhibit — false when in Compose Mode        |
| EL6           | 206         | Encoded Least Significant Bits = 6              |
| ESORR         | 205         | End-of-Screen Or Record Separator (formerly ES) |
| GGS           | 201         | Gated Group Separator                           |
| GS            | 201         | Group Separator                                 |
| IOB3P         | 227         | IOB3 and Parity Error not                       |
| IOB67         | 101         | IOB6 and IOB7                                   |
| OFF           | 206         | Off (Blank Cursor)                              |
| QMD           | 204         | Question Mark Decode                            |
| RPEA          | 204         | Reposition Cursor Enable — AIC                  |
| SCS6          | 206         | Set Control Storage Bit 6                       |
| SJUST         | 204         | Set JUSTify                                     |
| 43            | 313         | IOB4 and IOB3                                   |

AIC FLIP-FLOPS

| <u>Name</u> | <u>Page</u> | <u>Definition</u>                     |
|-------------|-------------|---------------------------------------|
| ACC         | 204         | Allow Cursor Control                  |
| ACTC        | 204         | Allow Cursor To Count                 |
| BLS         | 206         | Blink Storage                         |
| BLn         | 206         | Blink Counter (n = 1, 2, 3)           |
| GSS         | 206         | Group Separator Storage               |
| JUST        | 204         | Justify                               |
| MVE         | 204         | Move                                  |
| MVS         | 204         | Move Storage                          |
| QM          | 204         | Question Mark                         |
| QS          | 206         | QM, GS, or US Detected                |
| REPQM       | 204         | Replace Question Mark                 |
| RSS         | 204         | Record Separator Storage              |
| USOGS+1     | 204         | Unit Separator or Group Separator + 1 |
| USS         | 206         | Unit Separator Storage                |