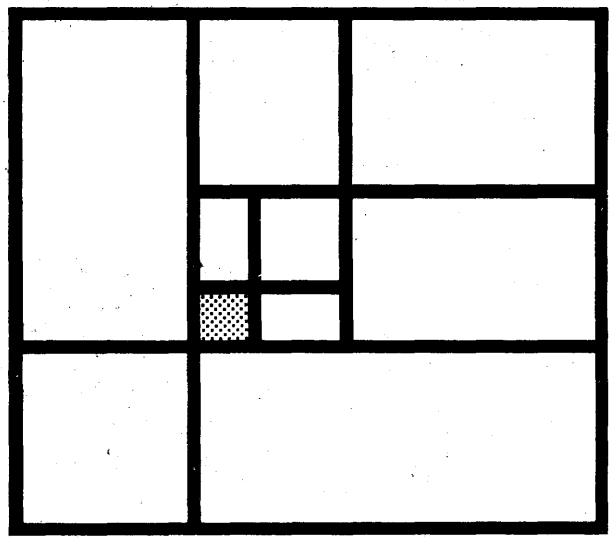
PROCESS COMPUTER

GEPAC® 4010



GENERAL ELECTRIC

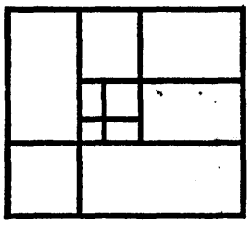
UTILITY AND PROCESS AUTOMATION
PRODUCTS DEPARTMENT
PHOENIX, ARIZONA

GE-NPGGD CAROLINA I (AZNA) REQN: 282-L3285

The GE PAC® 4010 Process Computer System

VOLUME III

I/O BUFFER
PERIPHERALS
BULK MEMORY
CONSOLES



This instruction book is provided as a basic source of technical information on this system. If problems arise in installation, operation, or maintenance that are not covered in these instructions, the matter should be referred to General Electric Company, Process Computer Products Department, Phoenix, Arizona, U.S.A. Attn: Technical Publications.

General Electric reserves the right to make changes in the equipment or software, and its characteristics or functions, at any time without notice.

GENERAL ELECTRIC

4822A CSU BASIC I/O BUFFER

TABLE OF CONTENTS

4822A CSU BASIC I/O BUFFER

The interfaces between the Arithmetic Unit and a GE-PAC* 4010 computer system's peripheral devices and communications channels are provided by the I/O Buffer. The I/O Buffer consists of a basic buffer module, one control (or drive) module for each input channel, and one control module for each output channel.

The basic buffer contains data control logic, test and alarm logic, and a parity check/generate circuit. The control modules contain logic and circuitry necessary to operate the attached peripheral device or to communicate through the communications channels. Each control module incorporates a holding register which stores each data character while input channels await access to the AU, or while awaiting access to output devices or channels.

This publication describes both the 4800AS14 Standard CSU Basic I/O Buffer and the 4822ASID Optional CSU I/O Buffer. One of these two basic buffers will be contained in each system. The 4800AS14 Basic Buffer is prewired to accommodate the most common used complement of peripheral drives. The 4822ASID Basic Buffer is wired unique for each system, permitting a wider range of peripheral drives to be implemented. Logic for both of these basic buffers is contained in 70C180909.

This publication provides generalized information which applies to the complete I/O Buffer, and the theory of operation of the basic I/O Buffer. The theory of operation for the control modules (4820BSXX) is provided in the publications which follow this basic description. Maintenance information for the I/O Buffer is included in the publications covering the peripheral devices and communications channels, in the Computer Maintenance Manual.

INSTRUCTIONS

In normal on-line operation, virtually all data transfers through the I/O Buffer are accomplished by the TIM/TOM feature. Once the program has set up a TIM or TOM table, configured the control words in the appropriate API response addresses, and set a buffer channel busy, data records are exchanged via TIM/TOM, with no further interference with the running program, and using very little AU time.

When the control module determines that the final character in a record has been transferred, or when a TIM table becomes full or a TOM table becomes empty, an end-of-record API is generated for the channel, which informs the program that the channel is ready for the next operation. Refer to the description of the Arithmetic Unit in Volume I of this bookset for a detailed description of TIM/TOM operation.

In addition to the TIM/TOM feature, seven GEN 2 instructions affect the I/O Buffer as described in the following paragraphs. Refer to the AU Theory description in Volume I for a more detailed description of these instructions.

OPR - Operate

OPR sets the addressed channel (Fig. 1) busy (not ready) and may transfer the contents of the 7 least significant bits of the A Register to the addressed control module. If transferred, the A Register data is decoded to determine a required channel configuration or action to be taken. The A Register remains unchanged. If set, the channel error flip-flop is cleared.

JCB - Jump If Channel Busy

If executed while the addressed channel (Fig. 1) is busy, program control "jumps" to the second sequential location. If the addressed channel is not busy, program control is transferred to next sequential location (no jump). A channel is normally busy when a data record transfer is in progress, when an input demand signal from the device or channel is present, and, in some cases, when the companion channel in a dual (input/output) channel is busy (as for an I/O Typer). The K1 bits of the instruction word are decoded in the control module to differentiate between JCB and JDR.

JDR - Jump If Data Ready

If, when JDR is executed, the addressed channel (Fig. 1) is busy and is holding a data character for transfer to the AU (input channels), or is ready to accept a data character from the AU (output channels), program control is transferred to the second sequential location. If the channel is not ready to transfer a data character to, or from, the AU, program control is transferred to the next sequential location. The K1 bits of the instruction word are decoded in the control module to differentiate between JDR and JCB.

IN - Input

IN transfers a data character from the addressed input channel to the A Register in the AU. The data is right justified (12 bits, max.) and unused bits of A are set to zero. IN resets the data ready test line from the addressed channel. The TIM function has the same effect except that the character is transferred to core through the B Register, leaving the A Register undisturbed.

OUT - Output

OUT transfers the contents of the A Register in the AU (12 bits, max.) to the addressed channel. OUT resets the data ready test line from the addressed channel. The TOM function has the same effect except that the character is transferred from core through the B Register, leaving the A Register undisturbed.

* Registered Trademark of General Electric Company

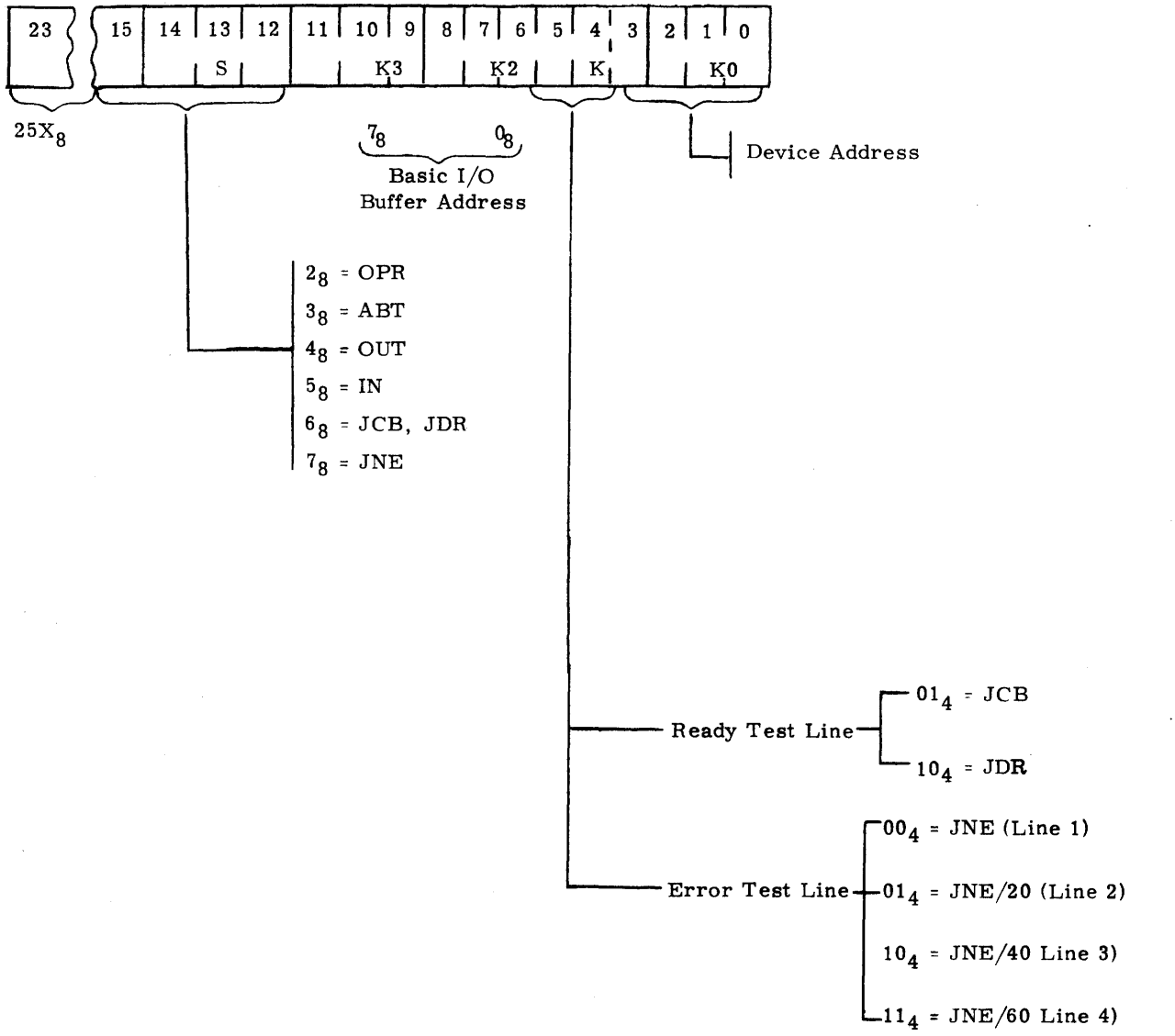


Fig. 1 GEN 2 Instruction Word Format

JNE - Jump If No Error

If, when JNE is executed, the error test line from the addressed channel indicates no error, program control is transferred to the second sequential location. If the test line indicates that there is an error, program control is transferred to the next sequential location. Bits 5 and 4 of the instruction word (K1) are decoded in the control modules to differentiate between up to four types of error conditions (see Fig. 1).

These conditions are defined in the descriptions of the control (drive) modules which follow this publication.

ABT - Abort

ABT initializes the addressed channel, The busy, data ready, and error test lines are all reset. Pushing the ON/INIT pushbutton on the Programming and Maintenance Console while the console is enabled and in MANual mode, (hardware initialize) has the same effect as ABT on all channels.

CHARACTER CODING

The characters exchanged between the Arithmetic Unit and the devices and communications channels are normally coded or decoded by the program to determine their meaning. For example, if the program wants to print the letter "A" on a teletypewriter, a line printer, or on a video display, it may execute OUT while the A Register contains the code 101_8 (on-line programs normally use the TOM function). The device attached to the addressed channel decodes the 101_8 bit pattern to determine that "A" is to be printed or displayed.

In addition to graphic character coding, such as for the letter "A", some channels employ control character coding. Control characters are used to cause predetermined responses in the devices or terminals receiving the character. For example, STX (002_8) indicates the start of text in paper tape records and in some communications channel messages. STX indicates that all of the characters between STX and ETX (003_8) are in the text of the message.

In some channels, the control character codes cause some specific action by the attached device or by the I/O Buffer control module. These codes and the responses to them are defined in the descriptions of the control modules which follow this publication.

Tables 1 and 2 list the character codes in the two principal formats used with GE-PAC equipment. The first table is a partial list of USASCII codes and, in general, is applicable to teletypewriters, line printers, DATANET* Video Display Subsystems, and the communications channels (ACD and SCU). The second table applies to Fixed Carriage Typers. Refer to the control (drive) module descriptions for further definition of the codes and formats used.

*Trademark of Honeywell Information Systems, Inc.

LOGIC DESCRIPTION

The basic CSU I/O Buffer logic appears on GE drawing number 70C180909. Fig. 2 is a block diagram of the basic buffer with references to sheet numbers on the logic drawing. The basic buffer is comprised of four printed wire boards: PTCA1, 2-PBCJ1's, and PJBA1. These boards are located in slots 13 through 16, respectively, in the CSU M panel. The control (drive) module PWB's are located in slots 17 through 32 of the M panel. Only those control module boards required for implemented devices or channels are installed in any CSU.

The interface between the I/O Buffer and the Arithmetic Unit is through the 4078 I/O Expander. This interface is a high speed GEN 2 interface. The K3 bits in all buffer addresses equal 7_8, the basic buffer address is specified by (K2), and the device/channel addresses are specified by the K1 and K0 bits as illustrated on Fig. 1. To trace signals between the basic I/O Buffer and the AU, refer to the I/O Expander logic drawing, 70C180899.

The interface between the I/O Buffer and the attached devices is depicted on sheet four of the basic buffer logic, 70C180909, and with greater detail on an interface summary sheet on each of the logic drawings for the control modules. All of the attached devices are plug-connected to jacks on the C panel as indicated on the table on sheet 4 of the - 909 logic.

The wiring between the basic buffer boards and 16 control module slots requires far more wire-wrap connections than can be accommodated by the back panel pins on the boards containing the basic buffer logic. The PJBA1 board in slot 16 is a jumper board which distributes the signals as shown on the slot 16 fan-out illustration at the right hand edge of sheet 4 of the - 909 logic. Connectors A through K on slot 16 provide adequate wire-wrap pins to accommodate the wiring to the control modules.

Instruction Decoding

The GEN 2 instruction decoding logic appears on sheet 5 of the - 909 logic. The purpose of this logic is to distribute GEN 2 instruction pulses to the control modules when the basic buffer is addressed.

Jumper pins are provided on the two PBCJ1 boards to select the basic buffer addresses. References in this description to logic terms K3 and K2, infer K3, and K2 are "true" when the buffer is addressed. Decoding of the six GEN 2 instructions is as follows:

- OPR - When a GEN 2 instruction with S=2 is decoded in the AU, the EXM0XGS2 signal from the expander goes "true", making N1GOPR on sheet 5 "true". If the basic buffer was addressed, driver D0GOPR is enabled, applying a zero volt OPR pulse to the control modules.

$$D0GOPR = K2 \cdot K3 \cdot OPR$$

Octal Code	Char.	Definition
000	NUL	Null
001	SOH	Start of Header
002	STX	Start of Text
003	ETX	End of Text
004	EOT	End of Transmission
005	ENQ	Enquiry
006	ACK	Acknowledge
007	BEL	Bell
010	BS	Back Space
011	HT	Horizontal Tab.
012	LF	Line Feed
013	VT	Vertical Tab.
014	FF	Form Feed
015	CR	Carriage Return
016	SO	Shift Out
017	SI	Shift In
020	DLE	Data Link Escape
021	DC1	Device Control 1
022	DC2	Device Control 2
023	DC3	Device Control 3
024	DC4	Device Control 4
025	NAK	Negative Acknowledge
026	SYN	Synchronize
027	ETB	End of Trans. Block
031	EM	Entry Marker
032	SUB	Substitute
033	ESC	Escape
036	RS	Record Separation
040	SP	Space (graphic)
041	!	
042	"	
043	#	
044	$	
045	%	
046	&	
047	'	
050	(	
051	)	
052	*	
053	+	
054	,	
055	-	
056	.	
057	/	
060	0	
061	1	
062	2	
063	3	
064	4	
065	5	
066	6	
067	7	
070	8	
071	9	
072	:	
073	;	
074	<	
075	=	
076	>	
077	?	
100	@	
101	A	
102	B	
103	C	
104	D	
105	E	
106	F	
107	G	
110	H	
111	I	
112	J	
113	K	
114	L	
115	M	
116	N	
117	O	
120	P	
121	Q	
122	R	
123	S	
124	T	
125	U	
126	V	
127	W	
130	X	
131	Y	
132	Z	
133	[	
134	\	
135	]	
136	^	
137		
177	DEL	Delete

Table 1 USASCII Codes

Octal Code	Character	Octal Code	Character	Octal Code	Character
000	0	030	H	060	+
001	1	031	I	061	/
002	2	032	&	062	S
003	3	033	.	063	T
004	4	034	]	064	U
005	5	035	(	065	V
006	6	036	<	066	W
007	7	037	\	067	X
010	8	040	↑	070	Y
011	9	041	J	071	Z
012	[	042	K	072	←
013	#	043	L	073	,
014	@	044	M	074	%
015	:	045	N	075	=
016	>	046	O	076	"
017	?	047	P	077	!
020	Space	050	Q	100	CR
021	A	051	R	120	DC4
022	B	052	-	130	Backspace
023	C	053	$	140	HT
024	D	054	*	150	Line Feed
025	E	055	)	160	BLK
026	F	056	;	161	RED
027	G	057	'	171	STX
				177	DEL

CR: Carriage Return
DC4: Terminate output message
HT: Horizontal Tab
BLK: Print black
RED: Print red
STX: Enable input operation
DEL: Delete

Table 2 Fixed Carriage Typer Codes

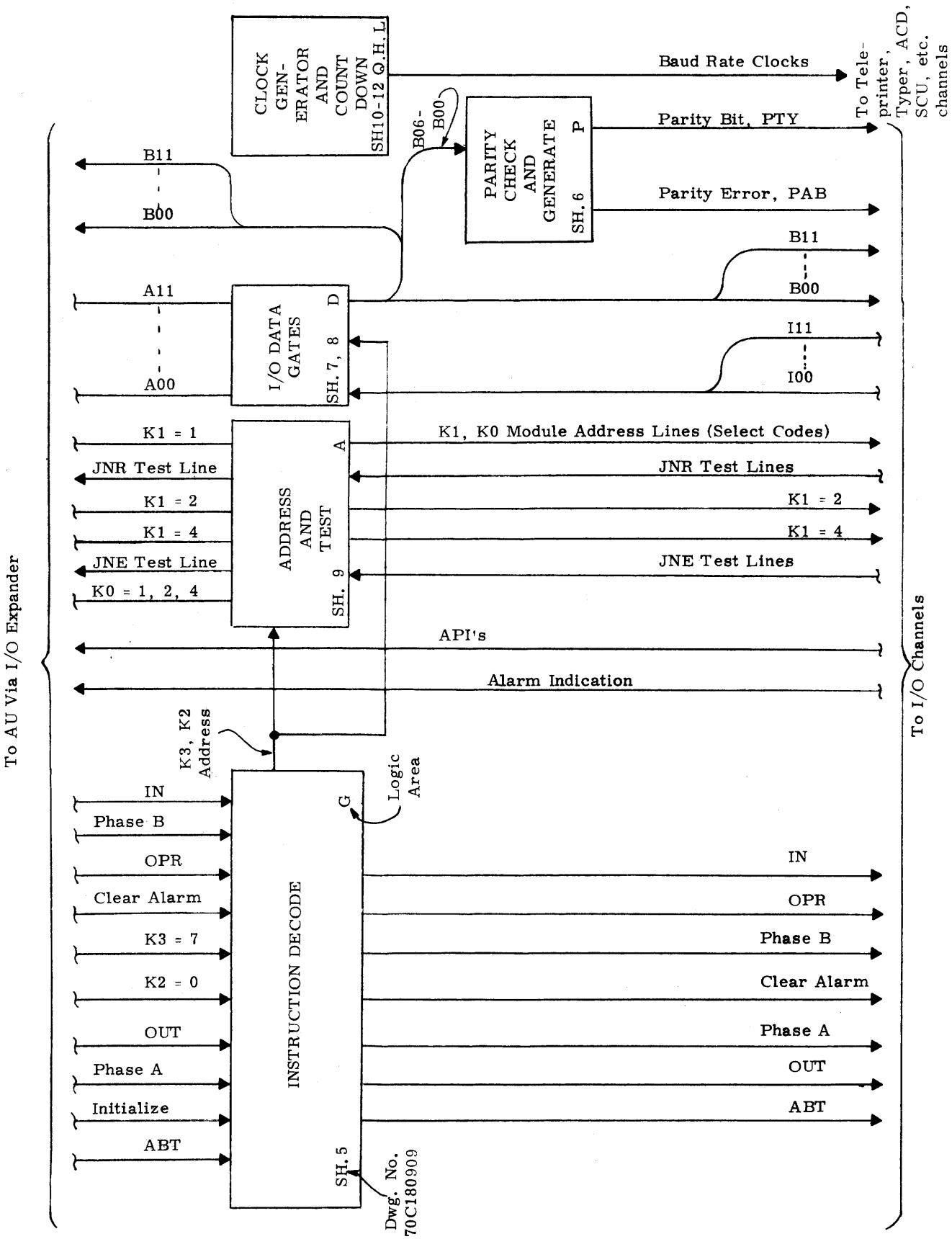


Fig. 2 Block Diagram, Basic I/O Buffer

- JCB, JDR, and JNE - The S bits for these instructions are decoded in the AU and are used in the expander to enable test line gates which transfer the status of the buffer test lines to the AU when the instructions are executed. The K1 and K0 bits are decoded in the basic buffer and the control modules as described under "Tests And Alarms".

- TIM or IN - When a GEN 2 instruction with S = 5 is decoded in the AU, the EXM0XGS5 signal from the expander goes "true", making N1GGIN on sheet 5 "true". If the basic buffer was addressed, driver D0GGIN is enabled, producing a positive going pulse at the output of D1GGIN, which is applied to the control modules.

  D0GGIN = K2 · K3 · GIN

- TOM or OUT - When a GEN 2 instruction with S = 4 is decoded in the AU, the EXM0XS4 signal from the expander goes "true", making N1GOUT on sheet 5 "true". If the basic buffer was addressed, driver D0GOUT is enabled, producing a zero volt pulse, which is applied to the control modules.

  D0GOUT = OUT· K2 · K3

- ABT or Initialize - When a GEN 2 instruction with S = 3 is decoded in the AU, the EXM0XS3 signal from the expander goes "true", making N1GABT on sheet 5 "true". If the basic buffer was addressed, driver D0GABT is enabled, producing a positive pulse at the output of D1GABT, which is applied to the control modules. D1GABT is also made "true" when the ON/INIT pushbutton on the Programming and Maintenance Console is pushed while the console is enabled and in MANual mode, which makes EXM0CINI "true".

  D0GABT = ABT · PHA· K2· K3

  G1GPHA = GEN 2 phase A pulse

I/O Data Gates

The I/O data gates, depicted on sheets 7 and 8 of the -909 logic, control the flow of data between the AU and the I/O Buffer and between the attached devices and the I/O Buffer. Logic is provided to control up to 12 data bits plus a parity bit. Most devices or channels use only 7 data bits plus a parity bit. Only the card reader uses gates I07 through I11, or the full 12 bits.

Data bits are transferred from the AU to the output control modules when either an OPR or OUT instruction is executed (or a TOM transfer is effected). The seventh bit, bit 06, is used in the following example.

The operation is the same for all other bits, 00 through 11:

- If bit 06 of the A Register is set (or the B Register if TOM) while OPR or OUT is executed, EXM0XA06 from the expander will be "true".

  This makes G1DO06 "true".

  G1DO06 = IN + A06

- Input gate G1DI06 will have a "true" output because OPR or OUT has enabled G0GBLB.

  G1DI06 = BLB + Input bit 06

  G0GBLB = OPR + OUT (sheet 5)

- Driver D0DB06 is enabled, applying a negative going bit 06 pulse to all output control modules. The output data character will be gated into the holding register of the addressed control module.

  D0DB06 = K2 · K3· O06·I06

Data bits are transferred from input control modules to the AU when IN is executed (or a TIM transfer is effected). The seventh bit, bit 06, is used in the following example. The operation is the same for all other bits, 00 through 11:

- If bit 06 of the holding register in the addressed control module is set while IN is executed, the bit 06 line from that module to gate G1DI06 (one of 7 input lines) will be "true" (zero volts). Since neither OPR nor OUT are being executed, G1DI06 can be "true" only if one of the input lines is "true".

  G1DI06 = BLB + Input bit 06

  G0GBLB = OPR + OUT

- Output gate G1DO06 will have a "true" output because IN is being executed.

  G1DO06 = IN + A06

- Driver D0DB06 is enabled, applying a negative going bit 06 pulse to EXG1ZD06 in the expander, from which it is transferred to the A (or B) Register in the AU.

  D0DB06 = K2 · K3 · O06·I06

Parity Check And Generate

As has been shown, the D0DBxx drivers are enabled, according to the data, whenever a data character is being transferred to the control modules (output) or to the AU (input). This permits the use of the same logic to generate a parity bit for each output character and to check parity on each input character.

Since no parity bit is read by the card reader, only the seven least significant bits plus the parity bit are used for the parity check. The parity check error signal, $\overline{\text{D0DPAB}}$, is applied to all input control modules, but only those associated with a channel which read a parity bit use the error signal.

The seven least significant output bits are used to generate an even output parity bit for all output characters. Only those control modules associated with output devices or channels which punch and/or check the parity bit, supply the parity bit to the attached device.

An analysis of the parity check and generate logic on sheet 6 of the -909 logic drawing will show that, if an even number of bits are set, of bits D0B00 through D0B06 and the input parity bit D0DPTY, the output driver D0PPAB will be enabled. If an odd number of bits are set, D0PPAB is disabled.

Note that for an output data transfer, D0DPTY (sheet 8) is always enabled because the output signal line to gate G1DOPT is grounded, holding the OPT output perpetually "true". The output of driver D0DPTY then, is representative of parity bits from input channels only. D0DPTY will always be "true" for an output character transfer, and will be "true" on input character transfers only when the input parity bit is "true".

$$\text{D0DPTY} = \text{K2} \cdot \text{K3} \cdot \text{OPT} \cdot \text{PTY}$$

$$\text{G1DPTY} = \text{BLB} + \text{Input parity bit}$$

$$\text{G0GBLB} = \text{OPR} + \text{OUT}$$

As an example of even parity generation, assume that when OUT is executed, the A Register in the AU contains 102_8; an odd number of 1-bits. D0DPAB, then, is enabled (0V) because the D0DBxx bits are odd and D0DPTY is "true", making the total of eight parity circuit inputs, even. The output parity bit driver, D1DPDO, is enabled (0V) in this case, making an even number of 1-bits (0V) on the lines to the output control modules.

As an example of parity checking, assume that when IN is executed, the holding register in the addressed control module contains 102_8, an odd number of 1-bits, but that due to some malfunction, the parity bit input line to G1DPTY is "false". Of the eight parity circuit inputs then, there is an odd number of 1-bits, and D0DPAB is disabled (+3.6V). The $\overline{\text{PAB}}$ signal sets the parity error flip-flop in the addressed control module. The I/O Buffer alarm light on the Programming and Maintenance Console will be illuminated and the error condition may be detected by executing JNE.

Tests And Alarms

As is indicated on Fig. 1, the K1 bits in the GEN 2 instruction words are used to differentiate between JCB and JDR, and both the K1 and K0 bits are used to differentiate between four possible JNE test conditions. The logic within the control modules decodes the K1 and K0 bits, both to determine when a module has been addressed and to differentiate between these test and alarm instructions.

The K1 and K0 bits are routed through logic in the basic I/O Buffer as shown on sheet 9 of the -909 logic drawing. In addition to providing signal drive and distribution, this logic makes all of the K1/K0 outputs "true" when the hardware initialize signal is "true". This is to cause all control modules to appear to be addressed when the initialize signal is present.

The JNE and JNR test line logic is also depicted on sheet 9 of the -909 logic. When the outpus of G1AJNE or G1AJNR are at 0V during the execution of JNE or JNR, respectively, program control in the AU does not jump, but is transferred to the next sequential location.

In the case of a JNE instruction, if no alarm or error condition exists in the addressed control module, one of the 16 input lines to G1ALAL and G1AMAL will go to 0V. G1AJNE will then be enabled (0V) during execution of the instruction, holding the test line in the "no jump" state. If an error or alarm condition does exist, G1AJNE will be disabled (+3.6V), causing the program sequence to "jump".

$$\overline{\text{G1AJNE}}\ (\text{no jump}) = \text{LAL} \cdot \overline{\text{GK2}} + \text{MAL} \cdot \overline{\text{GK2}}$$

$$\overline{\text{N0DGK2}} = \text{K3} \cdot \text{K2}\ (\text{basic buffer addressed})$$

If, in the case of a JCB instruction, the addressed channel is not busy; or in the case of a JNR instruction the addressed channel is not ready for data exchange, one of the 16 input lines to G1ALNR and G1AMNR will go to 0V. G1AJNR will then be enabled (0V) during execution of the instruction, holding the test line in the "no jump" state. If the addressed channel is busy or ready for data exchange, G1AJNR will be disabled (+3.6V), causing the program sequence to jump.

$$\overline{\text{G1AJNR}}\ (\text{no jump}) = \text{LNR} \cdot \overline{\text{GK2}} + \text{MNR} \cdot \overline{\text{GK2}}$$

CLOCK GENERATOR AND COUNTDOWN

A crystal controlled clock generator and frequency countdown logic are provided on the PTCA1 board in slot M13. The outputs of this board are used in timing the operations of the teletypewriter drives, the Asynchronous Communications Drive (ACD), and the Synchronous Communications Unit (SCU).

The ACD or the SCU may be used to operate a DATANET Video Display Subsystem, or if not implemented for that purpose, either or both may interface with a data set (modem) for communication with a distant terminal. One of the clock generator outputs must always be selected to provide the basic timing signal for the ACD. When the SCU is operated through

a data set, the timing pulses are normally supplied by the data set, and the PTCA1 outputs are used only for test purposes. When an SCU is directly connected to a DATANET Display Controller, the 9.6KHz output of the PTCA1 board is normally used for SCU timing at 4800 baud. However, if the Display Controller incorporates a Direct Timing Source, that source is used for the interface timing.

Refer to the ACD and SCU maintenance instructions in the Communication Drive Section of the Computer Maintenance manual for instructions for selecting the timing signals.

Clock Generator

The clock generator circuit, Q1QCLK, is a discrete component circuit consisting of an emitter-coupled crystal oscillator, Q1 and Q2, and two pulse shaper stages Q3 and Q4. The oscillator operates at 633.6 KHz. Pulse shaper stage Q3 is biased near cut-off. The resultant negative going pulses at the collector of Q3 are coupled to the base of Q4. Q4 is normally biased on and is turned off by the pulses. The positive half-microsecond pulses at the collector of Q4 are applied to F1QCLK, which is toggled by each pulse.

Each of the outputs of F1QCLK is a square wave of one half of the crystal oscillator frequency, or 316.8 KHz. The duration of each cycle is 3.157 microseconds, and each half cycle is 1.58 microseconds. The "1" output of F1QCLK is inverted by driver D0QCLK and applied to the high frequency countdown circuits. The "0" output of F1QCLK is inverted by D1QCLK and applied to the low frequency countdown circuit.

Countdown Circuits

Counter stages F1H11A, 11B, 11C, and 11D are interconnected such that G0HC11 is enabled at the trailing edge of every eleventh 316.8 KHz D0QCLK pulse, and disabled at the trailing edge of the next D0QCLK pulse. The C11 pulses are divided by 2 by F1H22A and the output of that stage is applied to the 14.4 KHz output driver, D0HF04, which is enabled by every twenty-second D0QCLK pulse (316.8 KHz/22 = 14.4 KHz).

$$D0HF04 = CLK \cdot H22A \cdot C11$$

Counter stages F1L15A, 15B, 15C, and 15D are interconnected such that G0LC15 is enabled at the trailing edge of every fifteenth 316.8 KHz D1QCLK pulse, and disabled at the trailing edge of the next D1QCLK pulse. The C15 pulses are applied to 21.12 KHz driver, D0LF03 which is enabled by every fifteenth D0QCLK pulse (316.8 KHz/15 = 21.12 KHz).

$$D0LF03 = CLK \cdot C15$$

The remaining countdown outputs are generated in a similar fashion. The rate at which any counter or gate operates can be determined from the logic name in terms of a ratio between the basic clock rate, 316.8KHz, and the count rate. Table 3 provides a tabulation of each countdown output driver and the logic elements which enable them.

Driver D0HF04 may operate at 4.8 KHz or 1.2 KHz, as determined by a jumper between a pair of pin jacks on the PTCA1 circuit board. If the green pin jacks are jumpered, gate G1H264 is disabled, producing a continuous +3.6V at its output. D0HF02 will then operate at a countdown rate of 66:1. If the black pin jacks are jumpered, G1H264 is allowed to operate and D0HF02 operates at a rate of 264:1.

Frequencies	Pulse-to-Pulse Spacing
28.8 KHz	34.7 usec.
21.12 KHz	47.3 usec.
14.4 KHz	69.4 usec.
9.6 KHz	104.1 usec.
4.8 KHz	208.2 usec.
2.4 KHz	416.4 usec.
1.20 KHz	833.3 usec.
1.32 KHz	757 usec.
0.88 KHz	1136 usec.

Output Frequency	Countdown Ratio	Driver Logic Equation
0.88 KHz	360:1	D0LF01 = CLK · F01 · L120
1.20 KHz	264:1	D0HF02 = CLK · H264 · C66 · C11
1.32 KHz	240:1	D0LF02 = CLK · L240 · L120
2.4 KHz	132:1	D0HF01 = CLK · C66 · H132 · C11
4.8 KHz	66:1	D0HF02 = CLK · C66 · C11
9.6 KHz	33:1	D0HF03 = CLK · 33A · C11
14.4 KHz	22:1	D0HF04 = CLK · 22A · C11
21.12 KHz	15:1	D0LF03 = CLK · C15
28.8 KHz	11:1	D0HF05 = CLK · C11

Table 3 Countdown Outputs

4820/4821X01
READER CONTROL DRIVE
TABLE OF CONTENTS

READER CONTROL DRIVE

The Reader Control Drive provides input channel interface logic (70C180344) linking the Basic I/O Buffer with the 4244D Card Reader or the 4212D Paper Tape Reader (one drive per channel).

This publication describes the hardware associated with the following GE-PAC* model numbers:

4DP4820AS01 - Reader Drive for the I/O Buffer in a GE-PAC 4010A System.

4DP4820BS01 - Reader Drive for the Basic Central Systems Unit (CSU) I/O Buffer in a GE-PAC 4010B System.

4DP4821AS01 - Reader Drive for the Auxiliary Systems Unit (ASU) I/O Buffer (4823ASID) in a GE-PAC 4010B System.

4DP4821BS01 - Reader Drive for the Second Central Systems Unit (CSU) I/O Buffer (4824ASID) in a GE-PAC 4010 System.

The logic for the Reader Drive associated with all of these model numbers is provided in GE drawing, 70C180344.

Input operations are performed in a record format. Both reader devices are started (card or tape movement) by program (OPR instruction) and data (characters) are read in as a result of IN instructions or the TIM function. The card reader stops after each card feed cycle. The paper tape reader stops when the DC3 code is read from the tape.

This reader control drive circuitry is contained on the PX1000PBCB1 printed wiring board.

TRANSFER FORMAT

4244D Card Reader

Data from the card reader enters the CPU as illustrated in Fig. 1. Row 9 is transferred to the least significant bit of the A register and row 12 goes to A_{11} position. When the TIM function is used, row 9 goes to either bit position 0 or 12, depending on the character packing mode.

4212D Paper Tape Reader

Data from the paper tape reader enters the CPU as illustrated in Fig. 2. The incoming record will contain the following data in the format and sequence shown in the next column.

Name	Octal Code	Function
NUL		Tape Leader
OPR	021	Device Enable Code
STX	002	Start of Text Code
TEXT ↓ TEXT		
ETX	003	End of Text Code
DC3	023	Device Disable Code
CR	015	Inter-Record Code
LF	012	Inter-Record Code
DC4	024	Inter-Record Code

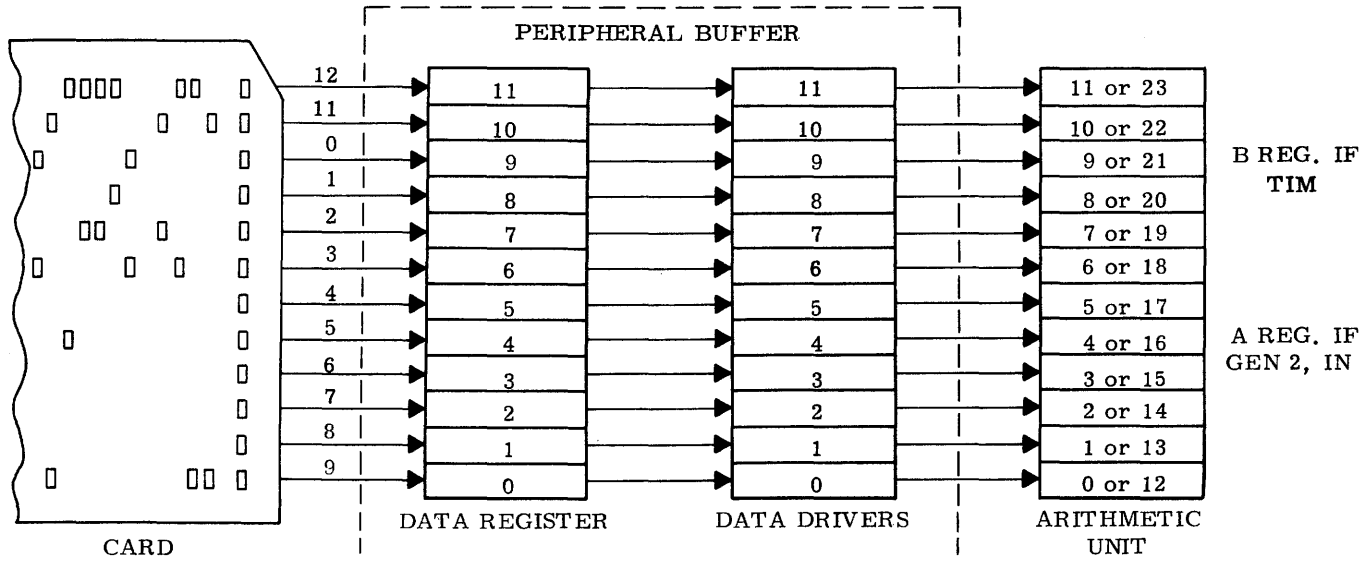


Fig. 1 Card Reader Data Transfer

*Registered Trademark of General Electric Company

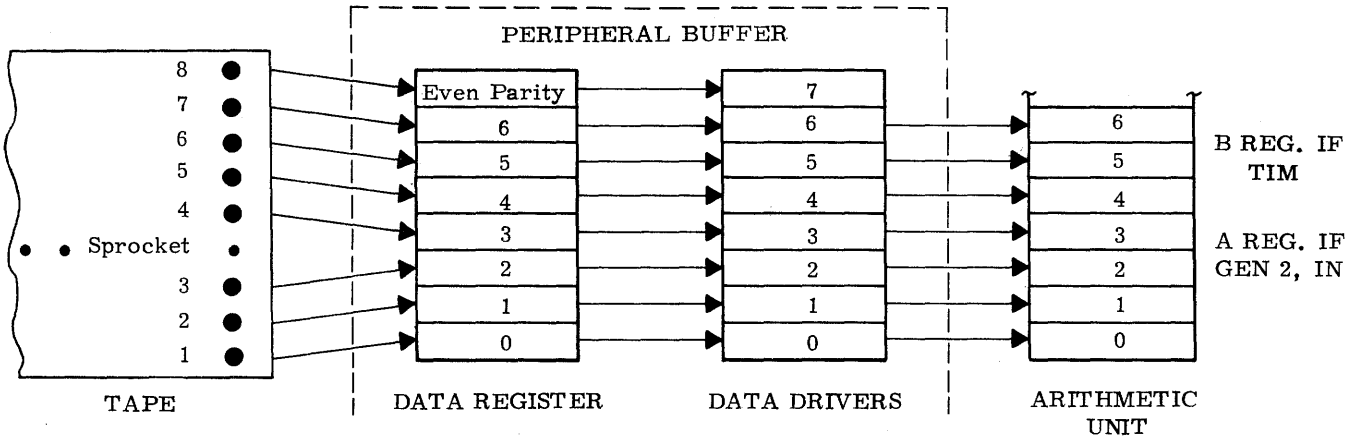


Fig. 2 Tape Reader Data Transfer

SPECIAL CHARACTER DEFINITIONS

4244D Card Reader

The card reader and control drive do not recognize any special character codes. The OPR instruction initiates the operation that causes one card feed cycle. When the OPR instruction is executed, the contents of the A register can be any code.

4212D Paper Tape Reader

As shown in the transfer format, several special characters are required on tape or in the program. These characters are defined below.

- NUL - The NUL code is used as tape leader and does not get read into memory.
- STX - The Start of Text code enables the Data Exchange Ready line which means data is to follow.
- ETX - The End of Text code serves no functional purpose on paper tape input. The ETX code is read into memory.
- DC3 - The Device Control 3 code disables or turns off the reader and generates the End-of-Record signal. The DC3 code must follow the ETX code.

  After the DC3 code is detected, the reader will stop, ready to read the next character which is the CR code. When the next OPR instruction is executed, all characters preceding the next STX are ignored by the I/O Buffer and, therefore, serve only as inter-record codes on paper tape input.
- CR-LF-DC4 - Serve as inter-record codes.

INTERRUPTS

Two interrupts are generated by the reader control drive logic: A data exchange interrupt (DEI) is used to inform the AU that a character is available for input. An end-of-record interrupt (EOR) is used to inform the AU that all data in a record has been read and transferred.

For the card reader, EOR is enabled when the card reader makes a light check at about column 84. For the tape reader, EOR is enabled when the DC3 character is read and decoded. Both devices can simulate the EOR interrupt by depressing then releasing the demand switch.

The DEI is normally non-inhibitable and the EOR is normally an inhibitable interrupt.

ALARMS

Properly encoded JNE instructions, addressed to the desired input device (D) will detect the following alarm conditions:

Alarm Line 1 - Device off line - K1 = $00D_2$

Alarm Line 2 - Device failure - K1 = $01D_2$

Alarm Line 3 - Bin Alert - K1 = $10D_2$ (card reader only)

Alarm Line 4 - Timing error - K1 = $11D_2$

- Device not plugged in
- Parity error (tape reader only)

4244D CONTROL LOGIC DESCRIPTION

The 4244D card readers are capable of reading 80 column, 12 row cards at speeds of 100 or 300 cards per minute, depending on the model option. Characters are read as a result of IN commands; one is required for each of the 80 characters. Each time a character passes over the photocells of the reader device, it is gated from the punched card to the A-register of AU. This action continues until the entire card has passed over the photocells and the end of the card is detected. Continuous card reading is made possible by properly spacing the OPR instructions in the program.

Model 4244D card readers are fully described elsewhere in this manual. The following text is devoted to the logic and timing of the reader input cycle, referenced to the timing diagram of Fig. 3, and to the logic schematics of GE drawing, 70C180344.

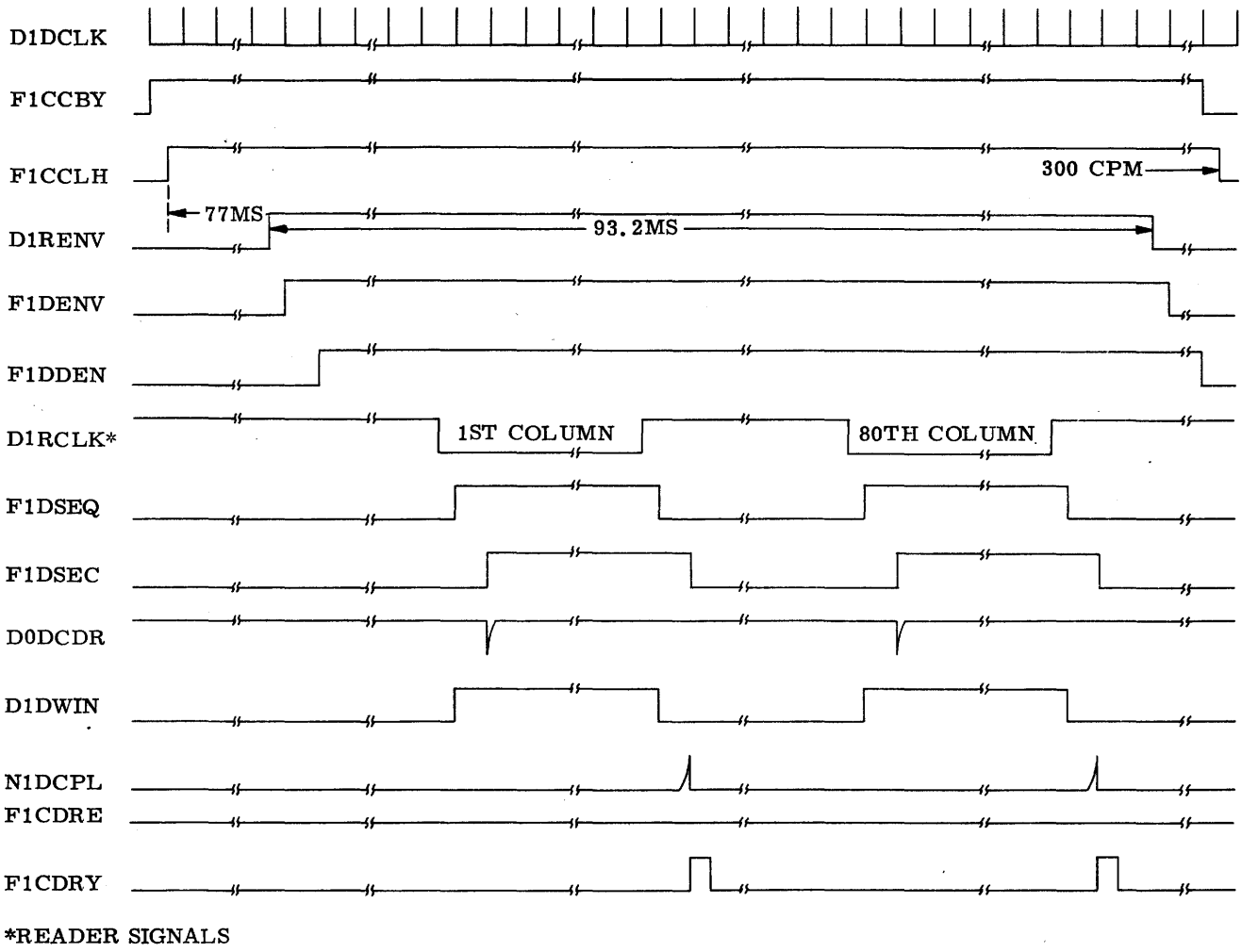


Fig. 3 Card Reader Timing Diagram

Initially, the channel must be ready (not busy). The channel busy flip-flop, F1CCBY, normally had been cleared at the end of the previous record. The ABT instruction will also clear the channel busy condition.

A channel test is made by the program via the JNR instruction. If the channel is not busy, the test line 1 gate, G0ATL1, is enabled and the program will step to the next sequential instruction.

$$G0ATL1 = \overline{CBY} \cdot \overline{DEN} \cdot K12 \cdot ADD$$

$$\overline{F1DDEN} = \text{No manual or prior demand}$$

A demand for an interrupt can be manually initiated by momentarily depressing the DEMAND pushbutton on the card reader device. Flip-flop F1DDEN sets, the channel busy flip-flop resets, and a channel ready interrupt occurs.

$$F1DENV = B1DENV \cdot CLK$$

$$F1DDEN = F1DENV \cdot CLK$$

$$G0DRCB = F1DDEN \cdot \overline{F1DENV} \cdot CLK$$

$$G0AEOR = \overline{F1CCBY} \cdot F1DDEN$$

Card Media Enable

The next sequential instruction is OPR, addressed to the input channel. Any code that may exist in the A-register is disregarded by the reader control logic. OPR sets the input channel busy flip-flop, F1CCBY, which sets clutch flip-flop F1CCLH, if the device is on-line, and clears test line 1. The program is released to await an interrupt.

$$F1CCBY = OPR \cdot PHB \cdot ADD$$

$$F1CCLH = CBY \cdot ONL$$

Card Data Enable

Relay driver D0CCLH activates the reader's card feed mechanism. The timing hole on the reader's clock belt that is associated with each data column generates a signal (DCLK0) that is used as one of the timing signals in the reader control logic.

When the first column passes over the photocells of the reader, its clock signal, at zero volts, sets sequence flip-flop F1DSEC on the next D1DCLK clearing the data register. Window driver D1DWIN is enabled with the setting of F1DSEQ.

$$F1DSEQ = B1DSEQ \cdot F1CCLH \cdot D1DCLK$$

$$F1DSEC = F1DSEQ \cdot D1DCLK$$

$$D1DWIN = F1DSEQ$$

$$D0DCRD = D1DCLK \cdot F1DSEQ \cdot \overline{F1DSEC}$$

The first data character is gated into the data register.

$$DR\ SET = DATA \cdot D1DWIN$$

Data ready enable flip-flop F1CDRE is set continuously, the only condition being that the card reader device is connected to the control module. Data ready flip-flop F1CDRY sets with a control pulse that is generated as a result of the space between clock holes on the reader clock belt being sensed. Data exchange interrupt gate G0ADEI is enabled.

$$F1CDRY = F1CDRE \cdot N1DCPL$$

$$N1DCPL = \overline{F1DSEQ} \cdot F1DSEC \cdot D1DCLK$$

$$G0ADEI = F1CDRY \cdot \overline{F1DSEC}$$

The computer responds to the interrupt with an IN command. The data is transferred from the data register, through the I/O Buffer, to the AU. Flip-flop F1CDRY is cleared to await the next character. All 80 characters are transferred to memory in a like manner, i.e., an IN command is required for each character.

Card Media Disable

At the end of a card (approximately column 84), the card envelope signal B1DENV goes to zero volts. This condition clears the channel busy flip-flop, clears the clutch flip-flop, and generates an end-of-record interrupt. The computer must respond with another OPR before the next card will feed.

4212D CONTROL LOGIC DESCRIPTION

The 4212D paper tape reader is capable of reading 8-channel (maximum) tape at a speed of 100 characters per second. Characters are read as a result of the IN command. One character at a time is gated from the tape to the A register of the AU each time a character passes over the photo-diodes of the reader device. This action continues until the media-disable code (DC3) is detected. The ETX code that precedes the DC3 code is read into memory as data. Tape movement ceases so that the frame following the DC3 code is positioned for reading.

The basic IN command is discussed earlier in this section. The 4212D reader is fully described elsewhere in this manual. The following text is devoted to the logic and timing of the reader input cycle, referenced to the timing diagram of Fig. 4 and to the logic schematics of GE drawing, 70C180344.

Initially, the channel must be ready (not busy). The channel busy flip-flop, F1CCBY, is normally cleared (reset) by the media-disable code (DC3) from the previous input operation. The Abort instruction will also clear the channel busy condition.

A channel test is made by the program via the JNR instruction. If the channel is not busy, the test line 1 gate, G0ATL1, is enabled and the program will step to the next sequential instruction:

$$G0ATL1 = \overline{CBY} \cdot \overline{DEN} \cdot K12 \cdot ADD$$

$$\overline{DEN} = \text{No Manual Demand.}$$

A demand for an interrupt may also be manually initiated by momentarily depressing the READ push-button on the paper tape reader device. Flip-flop F1DDEN sets, the channel busy flip-flop resets, and a channel ready interrupt occurs.

$$F1DENV = B1DENV \cdot CLK$$

$$F1DDEN = F1DENV \cdot CLK$$

$$G0DRCB = F1DDEN \cdot \overline{F1DENV} \cdot CLK$$

$$G0AEOR = \overline{F1CCBY} \cdot F1DDEN$$

Tape Media-Enable

The next sequential instruction is OPERATE, addressed to the input channel. The DC1 code (021_8), if present in the A register, is disregarded by the reader control logic. OPR sets the input channel busy flip-flop, F1CCBY; clears the data ready enable flip-flop, F1CDRE; sets the clutch flip-flop, F1CCLH, if the device is on-line; and clears test line 1. The program is released to await an interrupt.

$$F1CCBY = OPR \cdot PHB \cdot ADD$$

$$F1CCLH = CYB \cdot ONL$$

Tape Data-Enable

Relay driver D0CCLH activates the tape feed mechanism. The sprocket hole associated with each frame of information is used as one of the timing signals. All inter-record codes prior to the STX code are gated to the data register but are discarded because only the STX code can set the data ready enable flip-flop which is necessary to generate a data exchange interrupt.

When the STX code passes over the photo diodes of the reader, its sprocket signal, RT1 at zero volts, sets sequence flip-flop F1DSEQ. Sequence control

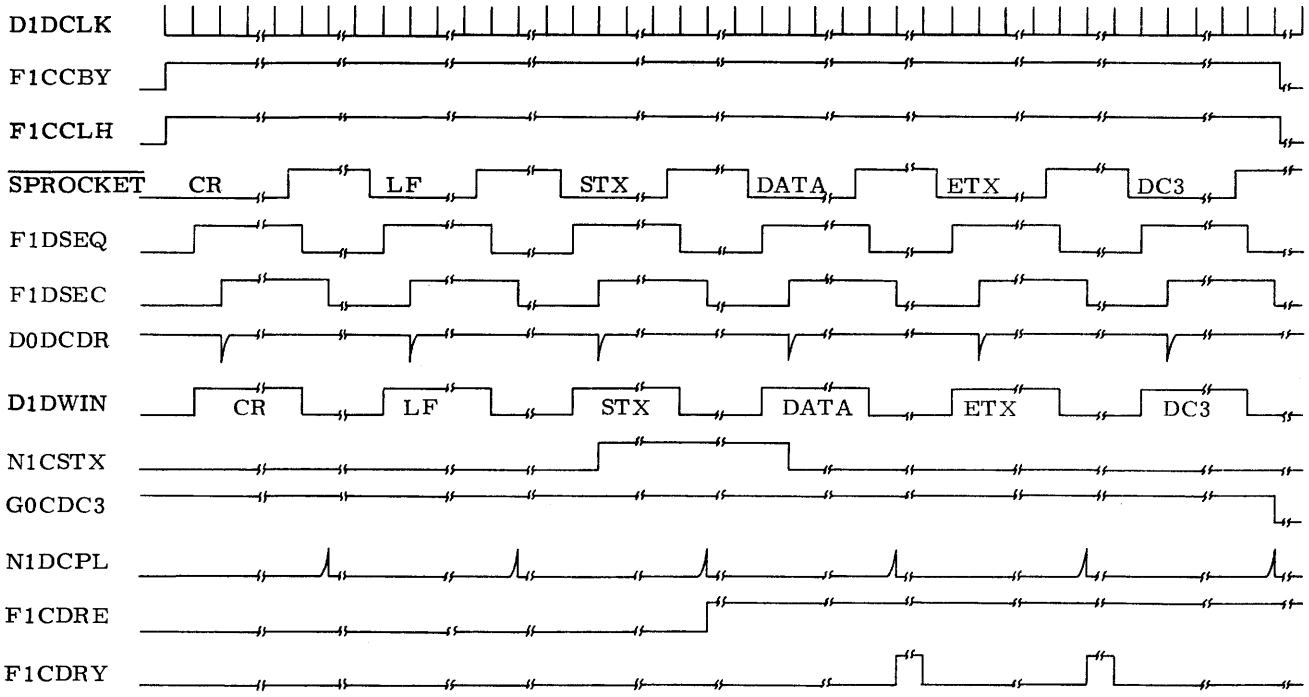


Fig. 4 Tape Reader Timing Diagram

flip-flop F1DSEC sets at the following clock pulse and the data register is cleared. Window driver D1DWIN is enabled with the setting of F1DSEQ.

$$F1DSEQ = B1DSEQ \cdot F1CCLH \cdot D1DCLK$$

$$F1DSEC = F1DSEQ \cdot D1DCLK$$

$$D1DWIN = F1DSEQ$$

$$D0DCDR = D1DCLK \cdot F1DSEQ \cdot \overline{F1DSEC}$$

The STX code (002_8) is gated to the data register.

$$DR\ SET = CODE \cdot D1DWIN.$$

The STX code is decoded by gate G0CSTX and the signal is inverted by N1CSTX. The data ready enable flip-flop F1CDRE sets, indicating that the next data received from the reader will be meaningful.

$$F1CDRE = N1CSTX \cdot F1CCBY \cdot N1DCPL$$

$$N1DCPL = F1DSEC \cdot \overline{F1DSEQ} \cdot D1DCLK$$

Data ready flip-flop F1CDRY sets at the next control pulse, enabling data exchange interrupt gate G0ADEI. By this time the first data character has set the data register.

$$G0ADEI = F1CDRY \cdot \overline{F1DSEC}$$

The program is interrupted and either an IN command is executed or a TIM operation is started. The data is transferred from the data register, through the I/O Buffer, to the AU. Flip-flop F1CDRY is cleared to await the next character.

Each data character, including the ETX code, is transferred to memory in a like manner, i.e., an interrupt and an IN command are required for each character.

Tape Media-Disable

When the DC3 code (023_8) has been detected by the reader and is in the data register, it is decoded by gate G0CDC3. Channel busy flip-flop, F1CCBY, and clutch flip-flop, F1CCLH, are cleared. With F1CCBY cleared, gate G0AEOR is enabled and its output goes from 3.6V to 0V. This signal is recognized by the AU as the End-of-Record interrupt.

PARITY ERROR (4212D)

In addition to the seven data characters on the paper tape, there is an even parity bit. This bit, like data, is gated to the data register and subsequently to the I/O Buffer. If the parity checking circuitry of the I/O Buffer detects a parity error, signal D0PPAB will be 3.6V. Alarm gate G0AALM is enabled and the alarm flip-flop, F1AALM, is set.

$$G0AALM = \overline{D0PPAB} \cdot N1RPTR \cdot N1COIN$$

TIMING ERROR

Alarm gate G0AALM is also enabled if a timing error has occurred.

$$G0AALM = \overline{SEC} \cdot SEQ \cdot DRY$$

If, for any reason, the AU has not serviced the Data Ready interrupt, the alarm condition can indicate that a data character has been lost.

PAPER TAPE/CARD PUNCH DRIVE

TABLE OF CONTENTS

PAPER TAPE/CARD PUNCH DRIVE

INTRODUCTION

The Punch Drive is an optional part of the I/O Buffer that can be used to drive the 4253 Paper Tape Punch or the 4283 Card Punch. Its primary functions are to provide a holding register and program selectable data formating (USASCII or EIA Standard) for output information being sent to the punch. This frees the basic control portion of the I/O Buffer for communication with other optional drive logics. Information held in the holding register is transferred to the respective device through transformer coupled AC drive lines, providing isolation of the device's inherently noisy power supplies from those of the computer.

This publication describes the hardware associated with the following GE-PAC* model numbers:

4DP4820AS02 - Punch Drive for the I/O Buffer in a GE-PAC 4010A System.

4DP4820BS02 - Punch Drive for the Basic Central Systems Unit (CSU) I/O Buffer in a GE-PAC 4010B System.

4DP4821AS02 - Punch Drive for the Auxiliary Systems Unit (ASU) I/O Buffer in a GE-PAC 4010B System.

4DP4821BS02 - Punch Drive for the Second Central Systems Unit (CSU) I/O Buffer in a GE-PAC 4010B System.

The logic for the Punch Drive associated with all of these model numbers is provided in GE drawing, 70C180346 (Transformer Coupled Drive I).

All necessary circuitry for the drive is contained on one standard-size GEAPS** printed wiring board (PX1000PBCF1). When installed as part of the I/O Buffer it is hard-wired to the Basic I/O Buffer and to the output connector jack. Since control and data requirements vary slightly between the two devices, signal connections at the device end differ according to which particular one is connected. Also, significance of the alarms sent back to the drive logic from the device differ according to which device is connected to the drive.

Further device distinction is accomplished by decoding the K1 bits of the OPR command, used to initiate device operation. The status of the K1 bits during execution of an OPR command determines which data format will be used and automatically enables the logic to detect the correct end-of-record code for the corresponding device.

Although the logic (drawing number 70C180346) is virtually the same, regardless of the device connected to it, differences in the format, interrupt usage, alarm significances and actual device operation require an independent description of the logic for each device. For this reason there are two separate descriptions, one for each possible device, following this introduction. If desired, the description not relating to the drive as implemented for your system can be removed.

*Registered Trademark of General Electric Company

**Trademark of General Electric Company

PAPER TAPE PUNCH DRIVE

The 4253 Paper Tape Punch connects to the I/O Buffer through the Punch Drive Logic to provide an output medium for permanently recording programs and data on paper tape. By specifying K_1 = 00X during the OPR command, ensuing data is punched in the USASCII code. EIA standard RS-244 code will be punched following the execution of an OPR with K_1 = 01X. Although the tape punches incrementally, by punching the transmitted code and spacing the tape ready for the next character, output operations to the punch are performed in a record format.

INTERRUPTS

Paper tape interrupts require two program interrupt points. The data-exchange interrupt signals the computer each time data is required by the punch and the end-of-record interrupt informs the computer when the record has been completed. The data-exchange interrupt is normally non-inhibitable and the end-of-record interrupt is usually inhibitable.

RECORD FORMAT

The outgoing data should contain the following data in the format and sequence shown:

Name	Octal Code		Function
OPR K_1=00X	022	*	Media enable-ASC II Code
K_1=01X	033	*	Media enable-EIA Code
OUT-NUL**	000		Tape Leader
STX	002		Data Enable Code (not used with EIA Standard)
TEXT ↓ TEXT			
ETX	003		Software recognition (not used with EIA Standard)
DC3	023		(For paper tape reader)
CR	015		Software Standard
LF	012		Software Standard
OUT-DC4	024		Media disable for ASC II Code
-DEL	177		Media disable for EIA Std. Code

* Contents of A-register during OPR. Not significant for this device; ensures program compatability with TTY punch.

** NUL Code repeated until desired amount of tape leader has been generated.

TRANSFER FORMAT

Data are transferred from the central processor to the paper tape punch as illustrated in Fig. PTP. 1. When the TOM function is used, the least significant bits of the B-register are transferred.

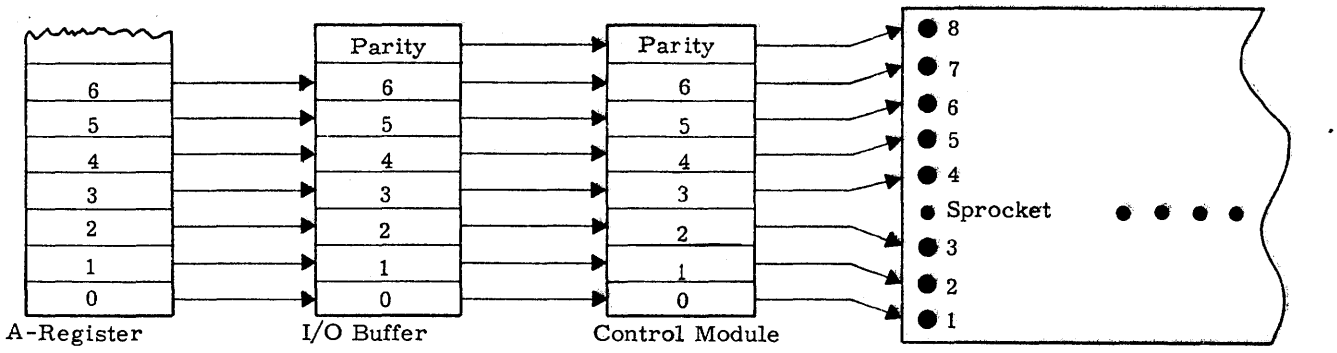
SPECIAL CHARACTER DEFINITIONS

As shown in the record format, several special characters are recorded in the tape record. These special characters are defined below:

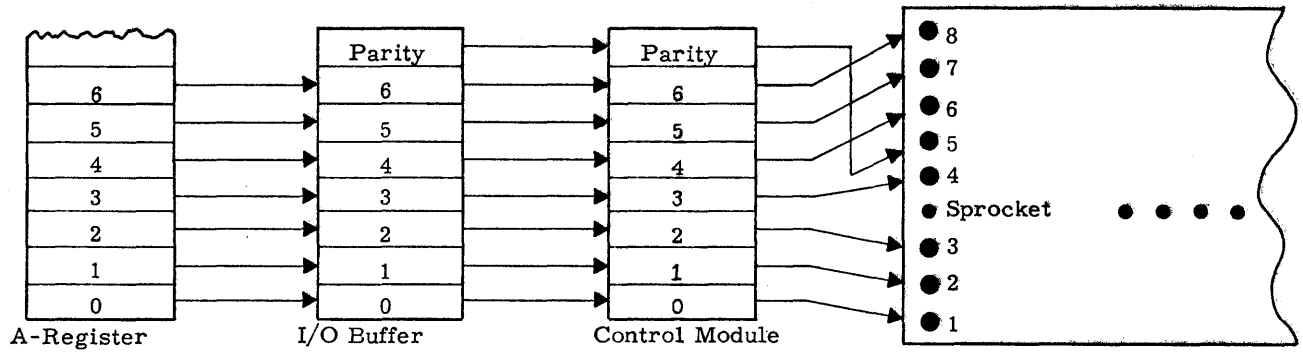
- NUL - The NUL code is used as tape leader.
- DC2 - The Device Control 2 code has no functional purpose on paper tape output but is used to be consistent with TTY punch requirements. The DC2 code is not punched.
- STX - The Start-of-Text code has no functional purpose on paper tape output but is required by tape readers. The STX code is punched.
- ETX - The end-of-Text code has no functional purpose on paper tape output but is required to be consistent with TTY punch software when using ASCII code. The ETX code is punched.
- DC3 - The Device Control 3 code has no functional purpose on paper tape output but is required by tape readers when using ASCII code. The DC3 code is punched.
- CR/LF - The Carriage Return and Line Feed codes have no functional purpose on paper tape output but are used to be consistent with TTY Punch software requirements. The CR and LF codes are punched.
- DC4 - The Device Control 4 generates the End-of-Record signal when outputting ASC II format. The DC4 code is punched.
- ESC - The Escape code signals the control module that the EIA Standard data is to be punched. Tape channels are switched as shown in Fig. PTP. 1. The ESC code is not punched.
- DEL - The Delete code generates the End-of-Record signal when punching EIA Standard data. The DEL is not punched.

CONTROL LOGIC DESCRIPTIONS

Model 4253 Paper Tape Punches are capable of punching 8-channel (maximum) tape at a speed of 120 frames per second. Characters are punched as a result of OUT commands. One character at a time is sent from the central processor to the punch each time the punch signals that it is positioned ready to receive the next character.



(a) ASCII Format



(b) EIA Standard Format

Fig. PTP. 1. Transfer Formats

This action continues until the media-disable code is detected by the control logic. Tape movement then ceases with the frame following the media-disable code positioned for punching.

A description of the OUT command is contained in the Arithmetic Unit portion of Volume I. The Model 4253 Paper Tape Punch is fully described elsewhere in this volume. During the following description, reference should be made to the flow chart of Fig. PTP. 2, the timing diagram of Fig. PTP. 3, and to the logic schematics of GE Drawing 70C180346.

Alarms

The following alarm conditions can be detected by using properly encoded JNE commands addressed to the drive logic:

- Text Line 0 - Test Line 0 is "true" when the punch is in the off-line condition.
- Test Line 1 - Test Line 1 is "true" when the punch is out of tape or the tape has broken.

Both alarm lines are sampled at gate G0ATL2 in the drive logic. The specific line sampled during the JNE command's execution is determined by the status of the JNE's K1 bits:

- K1 = 00X - Text Line 0
- K1 = 01X - Test Line 1

The output of G0ATL2 is routed through the Basic I/O Buffer to the AU where the decoded JNE causes the line to be sampled. Should the test line being sampled be "true", the corresponding AND input is enabled and G0ATL2 yields zero volts. A zero volt output during the JNE's execution causes the program to step to the next sequential location, indicating an error condition. If the test line being sampled is "false", the AND gate is disabled and G0ATL2's output is plus five volts. A plus five volt output indicates that there is no error and the program jumps to the second sequential location.

$$G0ATL2 = CADD\ (\overline{K14}\cdot\overline{K12}\cdot TL0 + \overline{K14}\cdot K12\cdot TL1)$$

Test lines two and three are not connected to the punch.

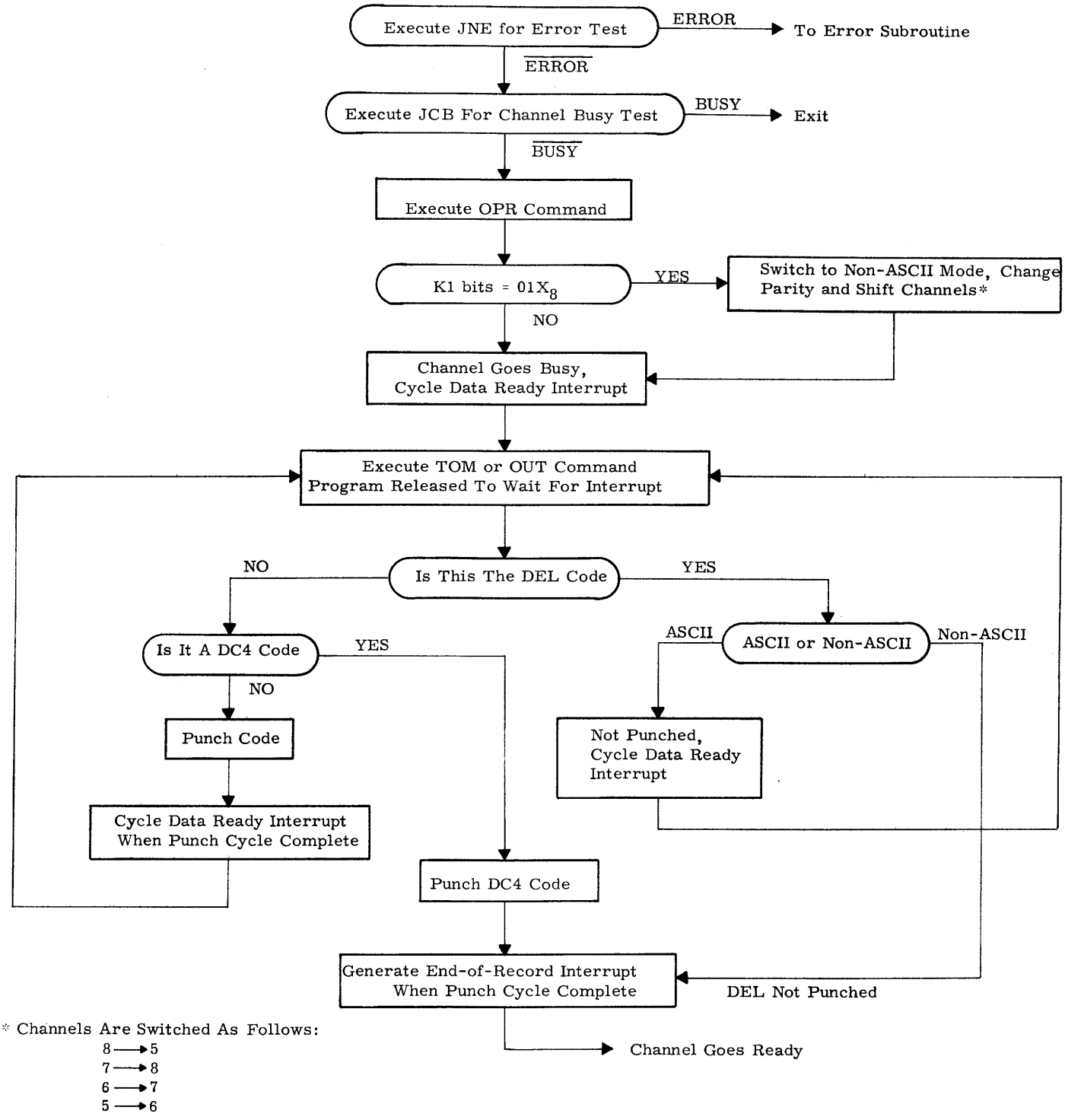


Fig. PTP. 2. Paper Tape Punch Flow Chart

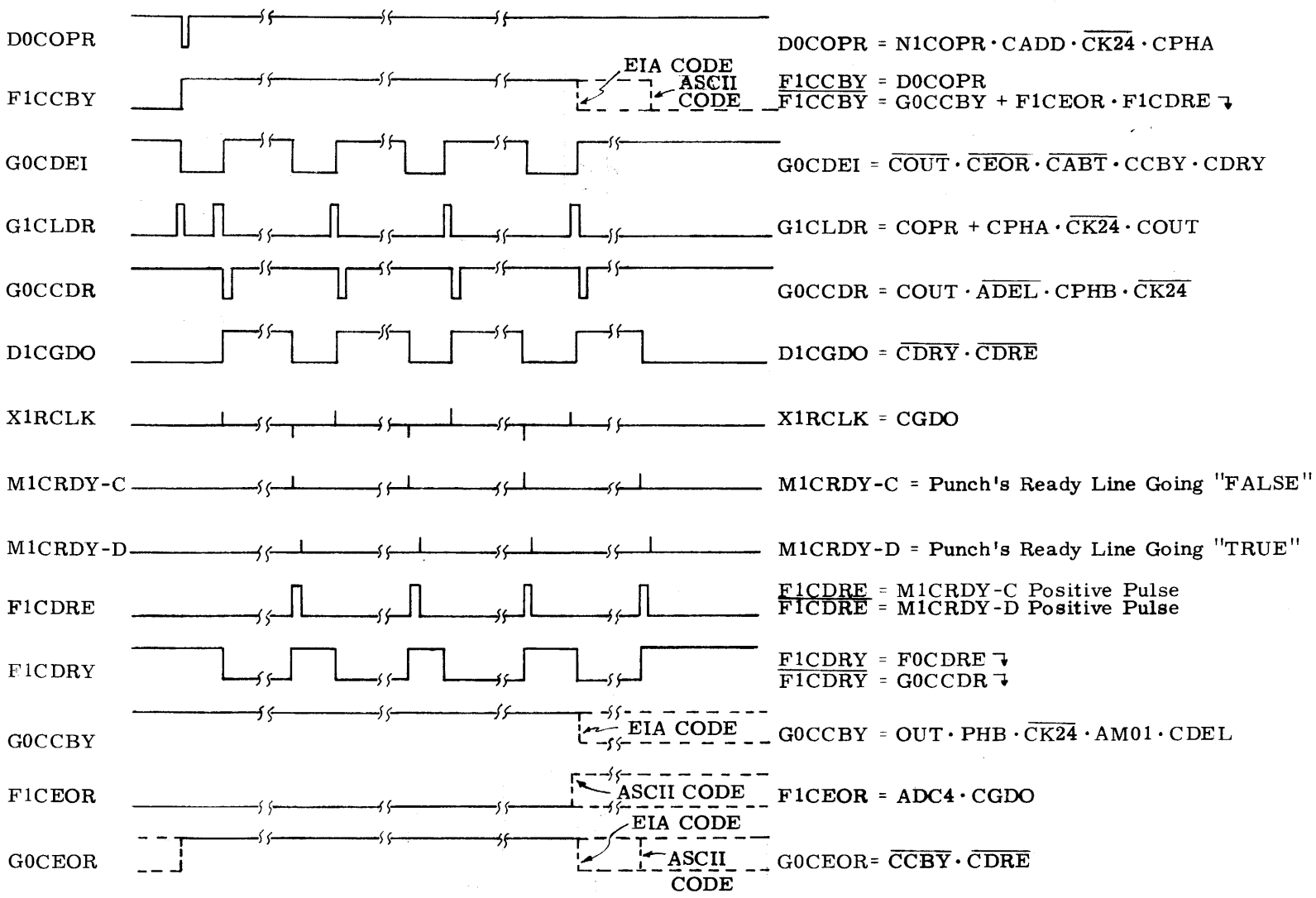


Fig. PTP. 3 Timing Diagram - Paper Tape Punch

Busy Tests

A JDR (Jump if Data Ready) or JCB (Jump if Channel Busy) test can be addressed to the drive logic to determine the respective data ready or channel busy status of the channel. Both commands are variations of the JNR command; the JDR is a JNR (2506_8) with K1=4 and the JCB is a JNR with K1=2.

When the channel is addressed with K1=2, the status of the channel busy flip-flop is sampled by an AND input to G0ATL1 of the drive logic. This line is routed through the Basic I/O Buffer to the AU where the decoded JCB causes the line to be sampled. Should the channel-busy (CCBY) or data-ready-enable (CDRE) flip-flops be set, the AND input is disabled and G0ATL1's output is plus five volts. A plus five volt output indicates that the channel is busy and the program jumps to the second sequential location. If the channel-busy and data-ready-enable flip-flops are both cleared, the AND input to G0ATL1 is enabled and G0ATL1's output is zero volts. The zero volt output during the JCB's execution causes the program to step to the next sequential location.

Addressing the channel with K1=4 causes the data-exchange-interrupt gate (CDEI) to be sampled by the other AND input to G0ATL1. The decoded JDR causes the line to be sampled in the AU. Should the data-exchange-interrupt gate be at zero volts, indicating data is not ready, the AND input to G0ATL1 is disabled and G0ATL1's output is plus five volts. A plus five volt output indicates that the channel is busy and the program jumps to the second sequential location. If the data-exchange-interrupt gate is at plus five volts, indicating a data ready condition, the AND input to G0ATL1 is enabled and G0ATL1's output is zero volts. A zero volt output indicates the channel is ready to accept new data and the program steps to the next sequential location.

Media Enable

Channel action is initiated with the execution of an Operate (OPR) instruction, addressed to the output channel. OPR sets the channel-busy flip-flop (F1CCBY) which cycles the data-exchange interrupt, clears the end-of-record flip-flop (F1CEOR), and gates the eight least significant bits of the A-register to the column counter (via the holding register).

$$D0COPR = OPR \cdot PHA \cdot ADD$$

$$G0COPR = OPR \cdot PHB \cdot ADD$$

$$F1CCBY = G0COPR$$

$$\overline{F1CEOR} = G0COPR$$

$$G0CDEI = \overline{OUT} \cdot \overline{EOR} \cdot \overline{ABT} \cdot CBY \cdot DRY$$

$$D1ROPR = G0COPR$$

Data being gated to the column counter is held temporarily in the holding register. At phase A of the OPR, the column counter is preset to "ones" and the data from the A-register is gated to the holding register.

At phase B of the OPR command, a single-ended transfer of the information in the holding register is made to the clear side of the column counter's flip-flops. This transfer is not significant for the paper tape drive function, but is used for the card punch drive.

Data Enable

As a result of the negative transition of signal G0CDEI, when the channel busy flip-flop (CBY) was set by the OPR command, an API is generated to inform the computer to initiate data transfers to the paper tape punch. These data transfers are accomplished using GEN II OUT commands. During phase A time of the OUT command, the load-data-register gate, D1CLDR, is enabled and the eight bits, including parity, from the basic I/O Buffer are gated to the holding register of the control drive:

$$D1CLDR = G0CLDR$$

$$G0CLDR = PHA \cdot \overline{K24} \cdot OUT$$

The transfer format (USASCII or EIA Standard) for the entire record was established by the status of the K1 bits during the OPR's execution. Mode 01 (K1 = $01X_8$) establishes the EIA format; mode 00 (K1 = $00X_8$) provides USASCII format. The corresponding translation is performed by gates at the output of the holding register.

$$\text{USASCII format} = \overline{G0AM01}$$

$$\text{EIA Std format} = N1AM01$$

$$N1AM01 = G0AM01 = MX1 \cdot \overline{M1X}$$

MX1 = K1 equal to 2 during execution of OPR

The data are transferred to the device through the output drivers. Accompanying the data is a clock signal, X1RCLK, to gate the data into the device's holding register. Both the clock signal's generation and the gating of data from the holding register to the output driver are accomplished by the gate-data-output signal, D1CGDO. D1CGDO is enabled upon receipt of the OUT command and remains "true" until a data received signal (Ready) is received from the device.

$$X1RCLK = D1CGDO$$

$$D1CGDO = G0CGDO = \overline{N0CDRY} \cdot \overline{F0CDRE}$$

$$\overline{N0CDRY} = \overline{F1CDRY}$$

$$\overline{F1CDRY} = G0CCDR = OUT \cdot \overline{DEL} \cdot PHB \cdot \overline{K24}$$

F1CDRY = F0CDRE going negative

(F0CDRE remains "true" during receipt of Ready signal from device)

The first X1RCLK signal activates the punch motor and when the motor is up to speed the first character is punched. Following the punch operation a ready signal (M1CRDY) is sent to the punch drive logic. The output of M1CDRY appears as a positive pulse at line receiver output "C" that corresponds to the leading edge of the ready signal from the device, followed by a positive pulse from "D" that corresponds to the trailing edge of the ready signal. The pulse from point C sets F1CDRE, which again clears when the pulse from point D is received. The output of the data-ready-enable flip-flop, F1CDRE, is used to set the data-ready flip-flop, F1CDRY, which enables the data-exchange-interrupt gate, G0CDEI, causing an API requesting new data.

The punch motor continues to run. With each OUT command that responds to the data exchange interrupt, characters are transferred to the punch in the manner described by the foregoing text until a media-disable code is recognized by the drive logic.

Media Disable

When punching in the USASCII format, the DC4 code (024_8) follows the data and inter-record codes, is decoded in the drive logic, and is recognized as the end-of-record signal. The end-of-record flip-flop sets as soon as the DC4 code is recognized by the decode logic. However, the actual end-of-record interrupt is not generated until the character has been punched on the tape and the punch ready signal returned to the drive logic.

The end-of-record flip-flop, when set, enables the "clear" input to the channel-busy flip-flop, F1CCBY, which actually is triggered to the reset condition by the trailing edge of the ready signal from the device. As F1CCBY clears, it enables G0CEOR to generate the end-of-record interrupt.

$$G0CERO = \overline{F1CCBY} \cdot F0CDRE$$

$\overline{F1CCBY}$ = F1CEOR•F1CDRE going negative

F1CEOR = G0CCER = DC4

F1CDRE goes negative on the trailing edge of the Ready signal from the punch.

When punching in the EIA Standard format, the DEL (no operation) code (177_8) is used at the end of the record in place of the DC4 code. The DEL is decoded at phase A time of the OUT command that delivers it and prevents clearing of the data-ready flip-flop. This prevents the gate-data-output gate (CGDO) from being enabled and the character is not transmitted to the punch. At phase B of the same OUT command, the DEL code is used to clear the channel busy flip-flop, via G0CCBY. The clearing of F1CCBY enables the end-of-record gate, G0CEOR, generating the end-of-record interrupt.

F1CDRY = G0CCDR

$\overline{G0CCDR}$ = G0ADEL

$G0CEOR = \overline{F1CCBY} \cdot \overline{DRE}$

$\overline{F1CCBY}$ = G0CCBY = OUT•$\overline{K24}$•MO1•DEL

MO1 = EIA Standard Code established at OPR execution.

Regardless of which format is used, the device logic turns off the punch drive motor when there have been no X1RCLK (activate) signals from the control module for approximately one second.

CARD PUNCH DRIVE

The 4283 Card Punch connects to the I/O Buffer through the Punch Drive Logic to provide an output medium for permanently recording programs and data on 80-column cards. This is the only one of the two devices described in this publication that requires the use of the K1 = 4 bit when addressing the drive with an OPR command. As a result of the OPR's execution, the contents of the A-register are transferred to a column counter in the drive logic. Ultimately, this count determines the point at which the end-of-record interrupt is generated. This occurs when the column counter, decremented once with each data transfer that follows the OPR command, changes from zero to minus one.

Each card that is to be punched requires the execution of an OPR command to initiate a card punch operation. This is followed by the desired number of data transfers, established by the column count sent with the OPR command and requested by the device at the proper times.

Data are transferred to the punch in eight-bit bytes containing six information bits, one filler bit, and one parity bit. The parity bit is inserted by the basic I/O Buffer before it reaches the drive logic and is sampled when it reaches the device.

The format of the data punched on the card is established by the arrangement of the bits in the corresponding register of the AU at the time of their transfer to the punch. Transfers are binary and, except for the addition of the parity bit, the bits are unchanged. The transfer sequence is such that the first data transfer appears in rows 12, 11, 10, 1, 2 and 3, while the second transfer appears in rows 4 through 9 of column one. Succeeding column transfers are in the same order.

No special characters are utilized for communicating with the card punch. The record format sequence and the logical description of the drive's operation follows.

RECORD FORMAT

The outgoing record contains the following information in the format and sequence shown.

Name	Octal Format	Function
OPR	K1 = 11X	Media Enable

OUT - column 1 (12-3)
- column 1 (4-9)
- column 2 (12-3)
- column 2 (4-9)
- ↓
- column X* (12-3)

OUT - column X (4-9)

* X = last column requested to be punched.

INTERRUPTS

Model 4283 Card Punches utilize two interrupts. A data-exchange-ready interrupt signals the computer each time data is required by the device and an end-of-record interrupt signals the computer after a punch cycle is completed. The data-exchange-ready interrupt is normally non-inhibitable and the end-of-record interrupt is usually inhibitable.

LOGIC DESCRIPTION

This description covers the logic that appears on drawing 70C180346, as it pertains to interfacing the card punch with the Basic I/O Buffer. The description of the Basic I/O Buffer precedes this publication in this section of the manual. The device description is also found in this volume under publication number 4283. Reference should be made to the simplified block diagram, Fig. CP. 1, the flow chart, Fig. CP. 2, and the timing diagram, Fig. CP. 3.

Alarms

The following alarm conditions can be detected by using properly encoded JNE instructions addressed to the drive logic:

- Test Line 0 - Test Line 0 is designated the "Halt" line by the Serial Card Punch Logic. It is "true" whenever any detectable punch problem occurs or the punch is switched to the off-line mode.
- Test Line 1 - Indicates an interlock open, a punch error, registration error, or a jam.
- Test Line 2 - Indicates that either the input hopper is empty or the output stacker is full.
- Test Line 3 - Indicates a data parity error, a card eject late, or a data late condition.

Definitions of the individual alarms can be found in the 4283 Serial Card Punch Theory publication.

All four alarm lines are sampled at gate G0ATL2 in the drive logic. The specific line sampled during the JNE command's execution is determined by the status of the JNE's K1 bits:

- K1 = 00X - Test Line 0
- K1 = 01X - Test Line 1

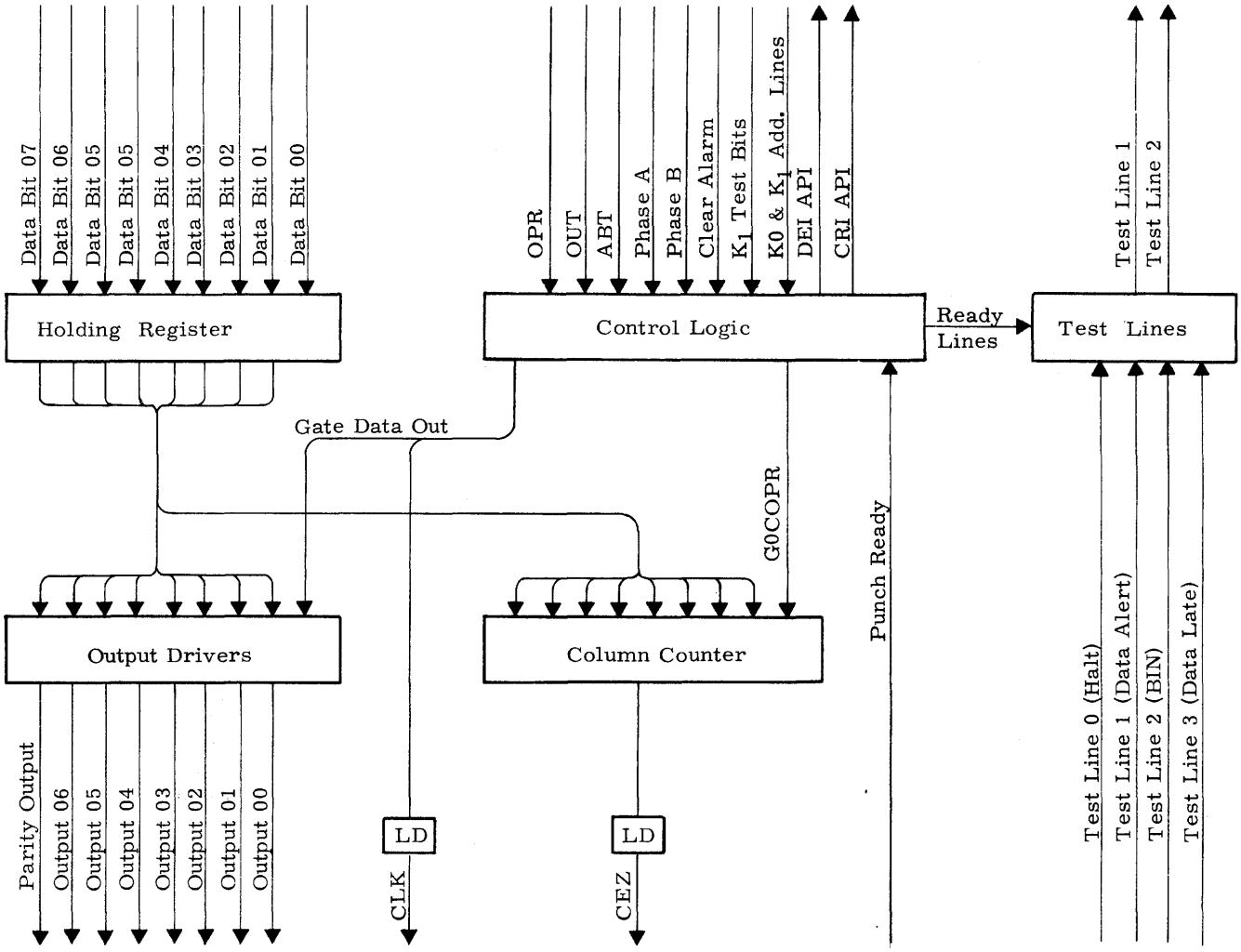


Fig. CP. 1 Simplified Block Diagram - Card Punch Drive

- K1 = 10X - Test Line 2
- K1 = 11X - Test Line 3

The output of G0ATL2 is routed through the Basic I/O Buffer to the AU where the decoded JNE causes the line to be sampled. Should the test line being sampled be "true", the corresponding AND input is enabled and G0ATL2 yields 0V. A 0V output during the JNE's execution causes the program to step to the next sequential location, indicating an error condition. If the test line being sampled is "false", the AND gate is disabled and G0ATL2's output will be +5V. A +5V output indicates that there is no error and the program jumps to the second sequential location.

$$G0ATL2 = CADD\ (\ \overline{K14}\cdot\overline{K12}\cdot TL0 + \overline{K14}\cdot K12\cdot TL1 + K14\cdot\overline{K12}\cdot TL2 + K14\cdot\overline{K12}\cdot ALM)$$

$$F1AALM = CBY\cdot TL3$$

Test line 3 going "true" sets the drive logic's alarm flip-flop. While set, the output drives an alarm display. This flip-flop is cleared, turning off the display, when the clear alarm switch is depressed, or when an Operate or Abort command are sent to the drive logic.

Busy Tests

A JDR (Jump if Data Ready) or JCB (Jump if Channel Busy) test can be addressed to the drive logic to determine the respective data ready or channel busy status of the channel. Both commands are variations of the JNR command; the JDR is a JNR (2506_8) with K1 = 4 and the JCB is a JNR with K1 = 2.

When the channel is addressed with K1 = 2, the status of the channel busy (CCBY) flip-flop is sampled by an AND input to G0ATL1 of the drive logic. This line is routed through the Basic I/O Buffer to the AU where the decoded JCB causes the line to be sampled. Should the channel-busy (CCBY) or data-ready-enable (CDRE) flip-flops be set, the AND input is disabled and G0ATL1's output is +5V. A+5V output indicates that the channel is busy and the program jumps to the second sequential location. If the channel-busy and data-ready-enable flip-flops are both cleared, the AND input to G0ATL1 is enabled and G0ATL1's output is 0V. The 0V output during the JCB's execution causes the program to step to the next sequential location.

Addressing the channel with K1 = 4 causes the data-exchange-interrupt gate (CDEI) to be sampled by the other AND input to G0ATL1. The decoded JDR causes the line to be sampled in the AU. Should the data-exchange-interrupt gate be at 0V, indicating data is not ready, the AND input to G0ATL1 is disabled and G0ATL1's output is +5V. A +5V output indicates that the channel is busy and the program jumps to the second sequential location. If the data-exchange-interrupt gate is at +5V, indicating a data ready condition, the AND input to G0ATL1 is enabled and G0ATL1's output is 0V. A 0V output indicates the channel is ready to accept new data and the program steps to the next sequential location

Media Enable

Channel action is initiated with the execution of an OPERATE (OPR) instruction, addressed to the output channel. OPR sets the channel-busy flip-flop (F1CCBY), which cycles the data-exchange interrupt, clears the end-of-record flip-flop (F1CEOR), and gates the eight least significant bits of the A-register to the column counter (via the drive's holding register).

$$D0COPR = OPR\cdot PHA\cdot ADD$$

$$G0COPR = OPR\cdot PHB\cdot ADD$$

$$F1CCBY = G0COPR$$

$$\overline{F1CEOR} = G0COPR$$

$$G0CDEI = OUT\cdot EOR\cdot ABT\cdot CBY\cdot DRY$$

$$D1ROPR = G0COPR$$

During phase A of the OPR command, the column counter is preset to "ones" and the data from the I/O Buffer is gated into the drive logic's holding register. A direct transfer of the eight bits from the basic I/O Buffer is made to the holding register. Bit seven is gated into the holding register's parity flip-flop while the parity bit from the Buffer is ignored due to the enabling of G0CBIN. G0CBIN is enabled throughout the execution time of the OPR command.

At phase B time, the data is gated from the holding register to the clear sides of the column counter's flip-flops. In each case where there was a "zero" in the corresponding bit of the holding register, the column counter flip-flop is DC cleared. This establishes the column count for the card and will determine the point in the card where punching will terminate.

Data Enable

As a result of the negative transition of signal G0CDEI, when the channel busy flip-flop (CBY) was set by the OPR command, an API is generated to inform the computer to initiate data transfers to the card punch. These data transfers are accomplished using GEN II OUT commands. During phase A time of the OUT command, the load-data-register gate, D1CLDR, is enabled and the eight bits, including parity, from the basic I/O Buffer are gated to the holding register of the control drive:

$$D1CLDR = G0CLDR$$

$$G0CLDR = PHA\cdot K24\cdot OUT$$

Although the transfer of information during the execu-

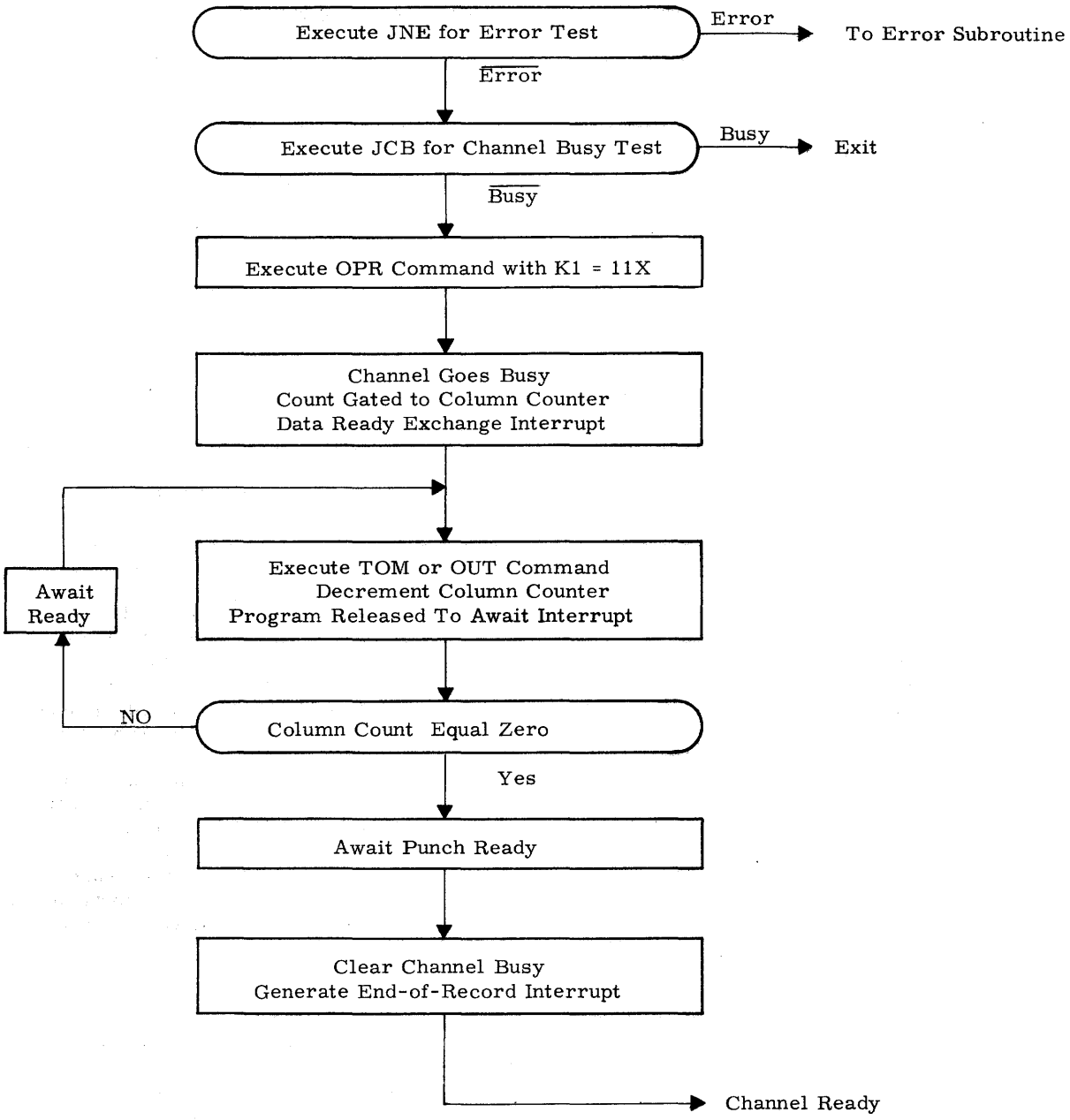


Fig. CP. 2 Card Punch Drive Flow Chart

tion of the OPR was strictly binary, with the parity bit being ignored, data transfers during OUT commands will be binary with the parity bit included. The parity will be checked in the punch upon its receipt of the data from the drive logic.

The data are transferred to the device through the drive logic's output drivers. Accompanying the data is a clock signal, X1RCLK, to inform the punch logic that the data is available. Both the clock signal's generation and the gating of data from the holding register to the output drivers are accomplished by the gate-data-output signal, D1CGDO. D1CGDO is enabled upon receipt of the OUT command and remains "true" until a data-received signal (Ready) is received from the punch.

X1RCLK = D1CGDO

D1CGDO = G0CGDO = $\overline{\text{N0CDRY}}\cdot\overline{\text{F0CDRE}}$

$\overline{\text{N0CDRY}}$ = $\overline{\text{F1CDRY}}$

$\overline{\text{F1CDRY}}$ = G0CCDR = OUT$\cdot\overline{\text{DEL}}\cdot$PHB$\cdot\overline{\text{K24}}$

F1CDRY = F0CDRE going negative

Upon recognition of the data from the drive logic, the punch's ready flip-flop, F1CRDY is "cleared". (The punch's ready flip-flop should not be confused with the drive logic's flip-flop, F1CDRY.) When the punch's ready flip-flop goes "false", it causes a pulse to be transmitted through transformer coupled drive lines to line receiver M1CRDY in the drive logic. The resulting positive pulse from pin "C" of M1CRDY ANDs with G0CHLT to cause the output of G1CSDE to go to zero volts. This sets the data-ready-enable flip-flop, F1CDRE. F0CDRE, going to zero volts, disables the gate-data-out gate (CGDO) and causes the drive logic's ready flip-flop (CDRY) to set. The setting of F1CDRY causes the data-exchange-interrupt line (CDEI) to go to zero volts, initiating an interrupt request for new data.

G0CDEI = $\overline{\text{COUT}}\cdot\overline{\text{CEOR}}\cdot\overline{\text{CABT}}\cdot$CCBY$\cdot$CDRY

CDRY = F0CDRE going negative

$\overline{\text{CGDO}}$ = F0CDRE

CDRE = G1CSDE at zero volts

CDSE at zero volts = CHLT and positive pulse from "C" of M1CRDY

Positive pulse at "C" of M1CRDY = Punch's ready line going "false"

If the data transfer is the first one for a column, the punch's ready line returns "true" approximately 3 microseconds later, the time necessary for the data to be stored in the Punch Register. However, if the data transfer is the second for the column (two data transfers per column), the ready line does not return "true" until the punch action for that column takes place. If it is the final column, the second data ready for that column does not return "true" until 348° of the punch's feed eject cycle.

When the punch's ready line returns "true", another pulse is sent through the transformer coupled drive lines to M1CRDY. This time a positive pulse is generated at pin "D" that causes G1CCDE to go to zero volts. G1CCDE, at zero volts, clears the data-ready-enable flip-flop, F1CDRE.

$\overline{\text{F1CDRE}}$ = G1CCDE at zero volts

G1CCDE at zero volts = Positive pulse from "D" of M1CRDY

Another data transfer is normally sent in response to the data exchange interrupt. It is decoded at phase B time of the OUT command by G0CCDR. When G0CCDR goes to zero volts it clears the drive logic's ready flip-flop. This again enables the gate-data-out gate, CGDO, and the data from the drive logic's holding register is gated out to the punch. This action is repeated until the total number of transfers equals that of the column counter at the beginning of the transfers.

Media Disable

The media disable occurs when the column count reaches zero. Originally containing a count received during the execution of the OPR command, the column counter decremented once for each data transfer that occurred during the punch cycle for that card. Since a K1=4 bit (K1=11X) was present during the execution of the initial OPR command, flip-flop F1AM1X remains set. This allows the count-equals-zero signal, N1RCEZ, to enable D0RCEZ during the last transfer for the card. D0RCEZ enables line driver X1RCEZ to to send a count-equals-zero pulse to the punch to terminate data requests.

Signal G0CRNT is enabled during the period that the column count is equal to zero. G0RCNT enables G1CTRM, which ANDs with D1CGDO to enable G0CCER to set the end-of-record flip-flop, F1CEOR. F1CEOR, set, enables the clear side of the channel-busy flip-flop, F1CCBY. The clearing of the data-ready-enable flip-flop at approximately 348° of the feed/eject cycle clocks the channel-busy flip-flop, F1CCBY, to the cleared condition. This causes the end-of-record interrupt gate, G0CEOR, to go to zero volts, generating an end-of-record interrupt.

G0CEOR = $\overline{\text{CCBY}}\cdot\overline{\text{CDRE}}$

CCBY = CCER

CEOR = RCNT

RCNT = Column count equal to zero

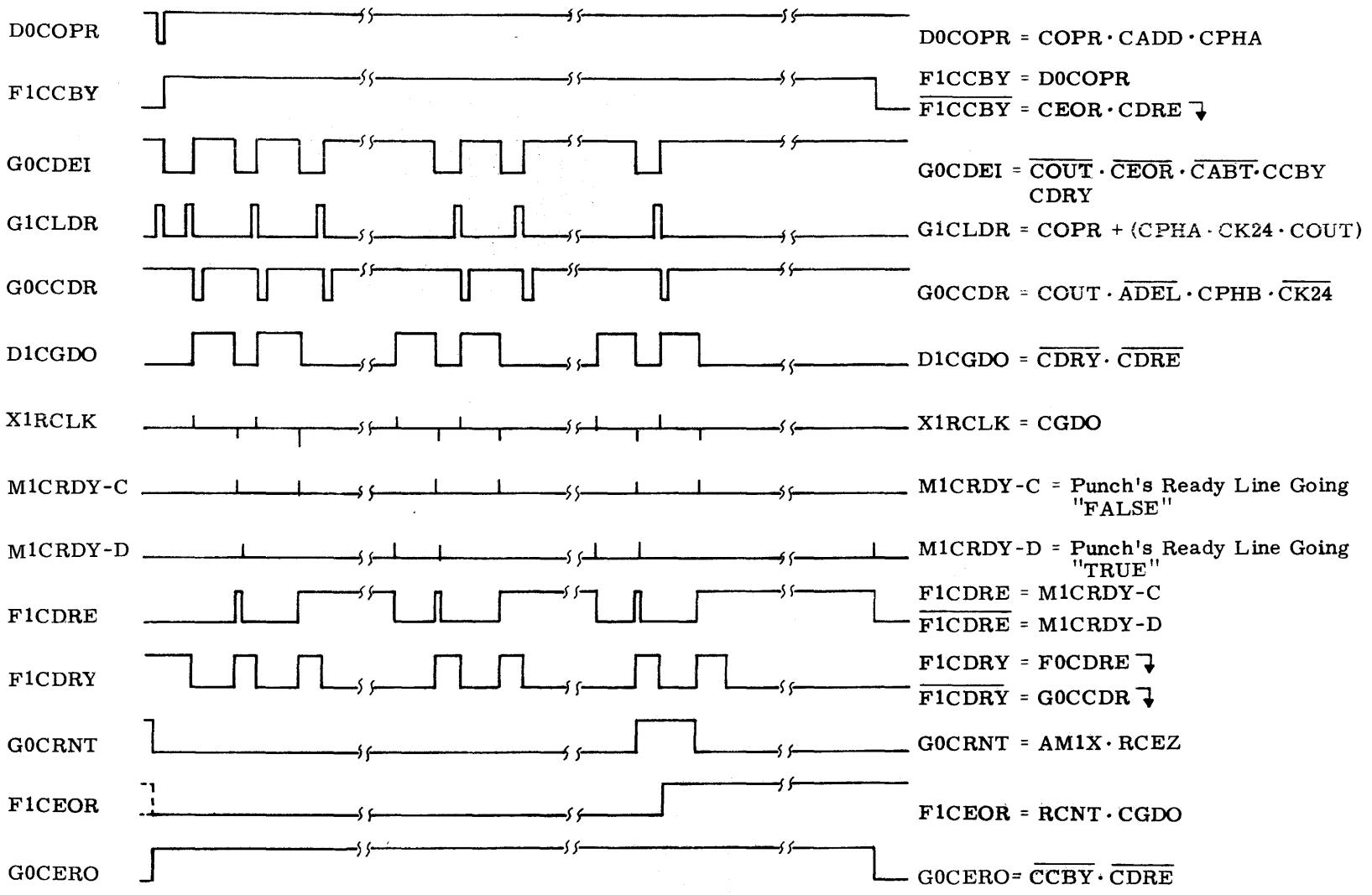


Fig. CP. 3 Card Punch Drive Timing Diagram

4820B/4821AS06
SYNCHRONOUS COMMUNICATIONS UNIT II

TABLE OF CONTENTS

SYNCHRONOUS COMMUNICATIONS UNIT II

The Model 4820B/4821AS06 Synchronous Communications Unit II (SCU II) provides a communications link between the Central Processor and a serial bit, synchronous, bidirectional, data communications device. Such devices use modulator-demodulators (modems) for data transmission to distant points over telephone lines and other communications media. Typical of devices used with the SCU II are American Telephone and Telegraph Co. Data Set Models 201A3 and 201B1. The SCU II may also be connected directly to a Display Controller (760-786).

The SCU II operates with and is functionally a part of I/O Buffer Model 4202. The basic I/O Buffer must include option 102 or 103. These options provide a crystal controlled clock pulse generator circuit board which generates timing pulses necessary for the operation of the SCU II.

The SCU II is installed in the same panel as the basic I/O Buffer. All communications with the Central Processor is through the basic I/O Buffer. Inter-connections between the SCU II and the communications devices are made by a multiconductor cable, a maximum of 50 feet in length.

Communication through the SCU II may be simplex, half-duplex, and full-duplex, as determined by the external communications device and the implemented program.

SCU Model 4820BS06 is the version implemented in the Central System Cabinet of 4010B process computer systems. Model 4821AS06 is implemented in the Auxiliary System Unit of a 4010B system. These versions are functionally identical to Model 4202X466/566. Maintenance information for these versions is provided in Publication No. 4820B/4821A06-M. The logic drawing is 70C180349.

INTERRUPTS

Four interrupts are produced by the SCU II. Each of the two channels produces a data exchange interrupt and an end-of-record interrupt. The interrupts signify the following:

Transmitter Channel:

- Data Exchange - The holding register is ready for the next character.
- End-of-Record - Data set is ready with carrier on, Data set is ready with carrier off, the final character in a message has been transmitted, or this station has been called.

Receiver Channel:

- Data Exchange - The receiver holding register contains a complete character.
- End-of-Record - An "End-of-Message" character has been received, or "line turn-around" is required or is completed.

TRANSFER FORMAT

Data are transferred to, and from, the seven least significant bits of the A register in the case of OUT, ODL and, IN, IDL instructions. When the TIM/TOM functions are used, data are transferred to and from the seven least significant bits of the B register.

Special character recognition is provided which allows the transmission or reception of seven or eight bit binary data in a "Transparent Text" mode. In this mode, all end of message character recognition is disabled, and the program must determine the character with which the message is ended. The SCU II transmitter goes to Transparent Mode when a DLE (020_8) character is detected in the output data stream, followed immediately by an STX (002_8) character. The SCU II receiver goes to Transparent Mode when DLE followed by STX is detected in the input data stream.

Odd parity is normally transmitted by the SCU II, and the reception of odd parity is expected. The transmitted parity bit is generated in the basic I/O Buffer or the Central Processor, as determined by a jumper on the transmitter circuit board, PSTB1. If the jumper is connected between the green pin jacks marked "P" the parity bit is supplied by the Central Processor. If the "S" pin jacks are jumpered, an even parity bit is generated in the basic I/O Buffer and converted to odd in the SCU II. In both channels, the parity bit is always contained in character channel 8, or the last bit in the serial data.

The received data is converted to even parity in the SCU II for checking by the basic I/O Buffer parity check circuit. The received parity bit may be transferred in to A (or B) register bit 07 of the Central Processor or it may be applied to the parity check circuit only. If a jumper is connected between the green pins jacks on the receiver circuit board, PSRB1, labled "P", the parity bit is transferred into the Central Processor. If the "P" pin jacks are not jumpered, zero will be transferred to bit 07 of the Central Processor A (or B) register.

The transmitter channel converts the eight bit parallel character from the Central Processor to an eight bit serial character. Each bit of the serial character occupies one unit of time. The reciprocal of the duration of the bit time is the baud rate of the SCU II and the external communications device. There must be no time interval separating data characters in such synchronous communications systems. After answering a call and turning the transmitter carrier on, the SCU II and the communications device transmit a minimum of four SYN (026_8) characters before the first character in a message.

The receiver channel converts the serial characters from the external communications device to an eight bit parallel character. There must be no time interval separating the data characters. While no data transfer is in progress, the receiver channel is continually expecting to receive SYN characters. After answering a call and detecting the received carrier "on", the SCU II receiver requires at least two SYN characters before the first character in a message.

Normally, any data character, with parity, following two or more SYN characters, is recognized by the receiver as the first character in a message. As an additional safeguard against line noise distorting SYN characters and causing them to look like a data character, a Front End Protection (FEP) feature is available

under program control. The FEP feature causes the receiver to recognize only an SOH, STX, or DLE character as the first character in a message.

ALARMS

Properly encoded JNE instructions will detect the following alarm conditions:

Transmitter Channel

Alarm Line 1 (K1 = $00X_2$) - Not used

Alarm Line 2 (K1 = $01X_2$) - Not used

Alarm Line 3 (K1 = $10X_2$) - DSR from data set does not agree with DTR from SCU II (DSR $\neq$ DTR).

Alarm Line 4 (K1 = $11X_2$) - CTS from data set does not agree with SCU II control register (CTS $\neq$ B04)

Receiver Channel

Alarm Line 1 - Not used

Alarm Line 2 - Received carrier is not on

Alarm Line 3 - Not used

Alarm Line 4 - Received carrier dropped out during message reception, a parity error was detected, or a data character was lost.

FUNCTIONS

Each of the SCU II channels accomplishes specific major functions as described in the following. Detailed theory of operation of the SCU II logic is provided under "Logic Description".

Transmitter Channel

- Converts baud rate timing pulses from the basic I/O Buffer, the communications device, or the receiver channel, to character rate timing pulses.
- Detects incoming calls (ring) and provides automatic answer capability, under program control.
- Provides line "turn-around", under program control, for alternate transmission to, and reception from, half-duplex communications devices.
- Converts output characters from parallel to serial form.
- Converts to Transparent Mode for, binary data transmission, upon detection of DLE followed by STX.
- Inverts the parity bit supplied by the Central Processor or the I/O Buffer, so that odd parity is normally transmitted.
- Recognizes end-of-message characters in the transmitted data, and produces an end-of-record interrupt as the final character in a message is transmitted, except in Transparent Mode.
- Causes calls to be abandoned under program control.
- Provides SCU II and communications device condition and alarming signals to the Central Processor, to permit monitoring of the operation.

Receiver Channel

- Converts baud rate timing pulses from the transmitter channel, or the communications device, to character rate timing pulses.
- Detects a requirement for line "turn-around" for alternate transmission to, and reception from, half-duplex communications devices, by detecting modem "carrier on/off".
- Converts to Transparent Mode, for binary data reception, upon detection of DLE followed by STX.
- Converts input characters from serial to parallel form.
- Provides optional input of parity bit to Central Processor A (or B) register bit 07.
- Recognizes "end-of-message" characters in the input data, and produces an end-of-record interrupt as the final character in a message is received.
- Provides "Front End Protection" (FEP), under program control, to allow only SOH, STX, or DLE character to be recognized as the first character in a received message.
- Provides SCU II and communications device condition and alarming signals to the Central Processor, to permit monitoring of the operation.

LOGIC DESCRIPTION

Operation of the SCU II is under the control of the Central Processor through the OPR, OUT, IN, and ABT instructions. Transfer of communications data may be implemented by means of the TIM/TOM or IDL/ODL functions, which appear at the SCU II as IN and OUT functions.

The principal calling and answering functions of the

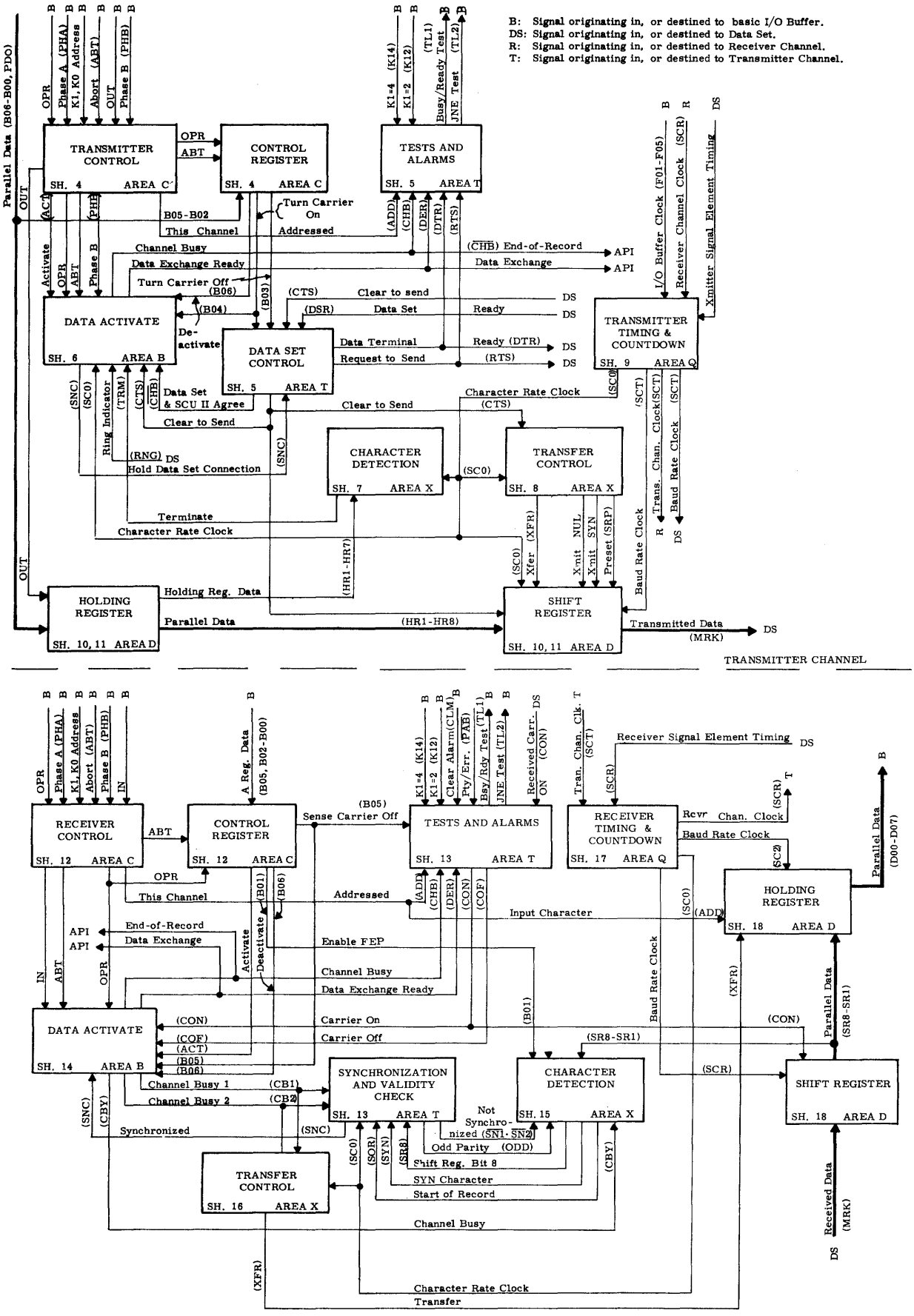


Fig. SCU II . 1. Block Diagram, Synchronous Communications Unit II

SCU II are controlled by the receiver and transmitter control registers. Control register contents are changed by Operate commands (OPR) issued by the Central Processor, in response to SCU II and communications device conditions monitored by the Central Processor, or in response to program requirements.

The control registers and associated logic control the operation of the communications device, activate, and deactivate the data transfer logic. Control register contents and the implementation of the SCU II features are determined by special characters contained in the A register of the Central Processor, at the time when an OPR command, issued to the channel involved, is executed.

When following the theory of operation provided in this Logic Description, frequent reference should be made to Figure SCU II . 1, Block Diagram, and to the SCU II logic schematic, GE drawing no. 70C180349.

In the discussion which follows, the SCU II timing, control registers, automatic answering, and line "turnaround" are covered first. This is followed by a description of the data transfer processes. Next the monitoring and alarming features are covered. Finally, the effect of the Abort (ABT) command is covered.

All of the logic elements in the SCU II are contained on two circuit boards: PSTB1 - transmitter channel, and PSRB1 - receiver channel.

Timing

The SCU II can accomodate communications devices which operate at baud rates up to 20,000 baud (baud = bits per second). Normally, the transmitter and the receiver operate at the same rate, but this is not mandatory. The primary timing signal for either the transmitter or the receiver channels may originate at any of three places: (1) The basic I/O Buffer, (2) the communications device,or (3) the opposite channel. When connected to a data set or to a directly connected Display Controller, the baud rate timing signals are derived from the connected device in nearly all cases.

If the primary timing signal is derived from the basic I/O Buffer, as is the case when testing the SCU II, a jumper pin is connected between one of five pairs of black pin jacks, F1 throught F5, on the transmitter circuit board, PSTB1. The crystal controlled clock generator is contained on a basic I/O Buffer board, PTCA1 or PTCB1, located in slot B32 of the I/O Buffer panel. The crystal controlled clocks are connected from the generator board to the F1 through F5 pin jacks on the SCU II transmitter board. The frequencies applied to each input pin, and the baud rates resulting from the jumpering of each pin pair are as follows:

F1 - 2.4 KHz; 1200 baud.
F2 - 4.8 or 1.2 KHz; 2400 or 600 baud.
F3 - 9.6 KHz; 4800 baud.
F4 - 14.4 KHz; 7200 baud.
F5 - 28.8 KHz; 14,400 baud.

If used, the crystal controlled clock is applied through N1QCLA to F1QSCT. The flip-flop is toggled by each input clock, and the F1QSCT output is at one-half the input frequency and at the baud rate.

If the jumper is connected between the transmitter pin jacks labled "FR" or "FE", one of the two DC clear inputs to F1QSCT is grounded, holding the flip-flop in the clear state, and preventing its operation. With "FR" jumpered, a +3.6V signal is developed by N1QEFR which partially enables an AND gate on G1QSCT which is enabled by the timing pulses from the receiver channel. With "FE" jumpered, a +3.6V signal is developed by N1QEFE which partially enables an AND gate on G1QSCT which is enabled by the timing pulses from the external communications device. G1QSCT, then, produces baud rate clock pulses which originate in one of the three available sources. The SCT signal is used for baud rate timing in the transmitter channel. The SCT signal is also applied to the receiver channel for optional use as a timing source. The SCT clock is available to the communications device through line driver X1QSCT, when the jumper is in any pin jack pair except "FE".

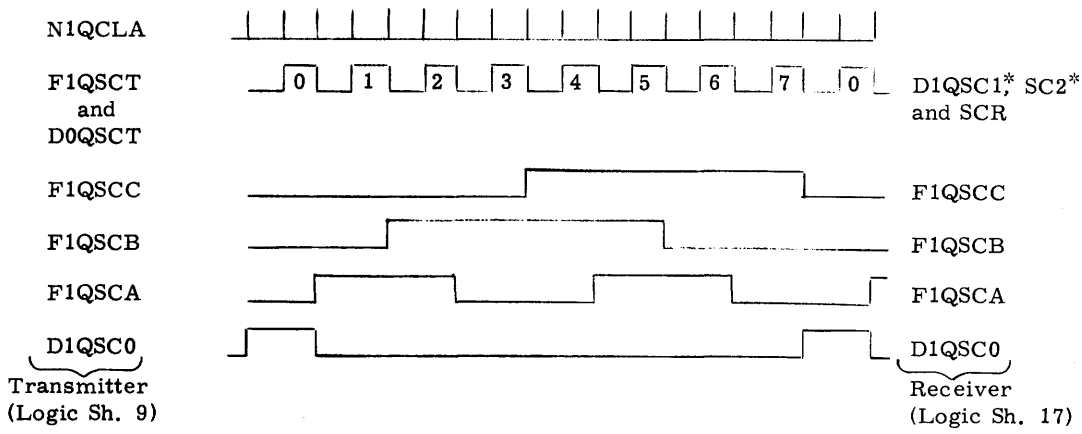
$$X1QSCT = SCT \cdot \overline{FE}$$

Two pairs of black pin jacks are available on the receiver circuit board, PSRB1, for selecting the receiver timing source. If the "FT" pin jacks are jumpered, N1QEFT partially enables an AND gate on G0QSCR, which is enabled by the transmitter baud rate clock from G1QSCT. If the "FE" pin jacks are jumpered, N1QEFE partially enables another AND gate on G0QSCR which is enabled by the timing pulses from the external device. G0QSCR, then, produces baud rate clock pulses which originate in one of two sources. The SCR clock is used for baud rate timing in the receiver channel. The signal is applied to the transmitter channel, through G1QSCR, for optional use as a timing source.

Each channel has an identical character rate clock generator which divides the baud rate clocks from D0QSCT and D1QSC1 by eight to produce the character rate timing pulses at the outputs of the D1QSC0 drivers in each channel. The operation of the countdown circuits is illustrated on Figure SCU II . 2. The receiver countdown circuits operate only if at least one synchronization, SYN, character has been received on the input data line, and they continue to operate only if the receiver channel is activated.

Line Receivers

Four line receivers are included in the transmitter channel to accept signals from the communications device, which indicate the operating configuration. They are: M0TCTS, logic sheet 5; M0DTSR, sh. 5; M0BRNG, sh. 6; and M0QSCT, sh. 9. Three line receivers are used in the receiver channel to accept "carrier on/off", timing, and data signals. They are: M0TCON logic sheet 13; M0QSCR, sh. 17; and M1DMRL, sh. 18. Each of the line receivers operates in conjunction with a gate which inverts the incoming signal, and the inverted signal is connected to a second input on each line receiver. The OR symbol (+) shown



* D1QSC1 and SC2 are held at +3.6V, unless receiver channel is in sync. with data set.

Fig. SCU II . 2 8:1 Counters

on each line receiver indicates that in the case of a momentary drop out of an input signal, the second input will hold the line receiver enabled. However, the OR condition on the line receiver input is temporary, and if the input signal remains off for more than a few microseconds, the line receiver is disabled. This "hysteresis OR" connection eliminates false drop outs of the incoming signals due to noise on the incoming lines.

Control Register Instructions

Table SCU II . 1 indicates the A register data required to make both transmitter and receiver channel control register changes, as an OPR instruction is executed. The control sections of each control register set up the communications device and SCU II configurations necessary to implement the required operations. The data activate sections set the appropriate channels busy, and enable data transfer.

In the transmitter channel, no change in control register bits 04 and 03 is made unless bit 05 is set when the OPR command is executed. Bits 04 and 03 may not both be set when OPR is issued to the transmitter channel with bit 05 set. In both channels, no change in control register bits 01 and 00 is made unless bit 02 is set when the OPR command is executed. The receiver control register does not use bits 04 and 03. When an OPR command is issued to the receiver channel, control register bit 02 may not be set if bit 05 is set. In both channels if bit 06 is set, no other bit may be set. Under these restrictions, the following A register control words are available to set up the SCU II and the communications device (data set):

- Transmitter Channel

  064_8 or 066_8 - Answer call with carrier on.

  050_8 - Answer call with carrier off.

  040_8 - Disconnect call.

  004_8 or 006_8 - Activate data.

  100_8 - Deactivate data.

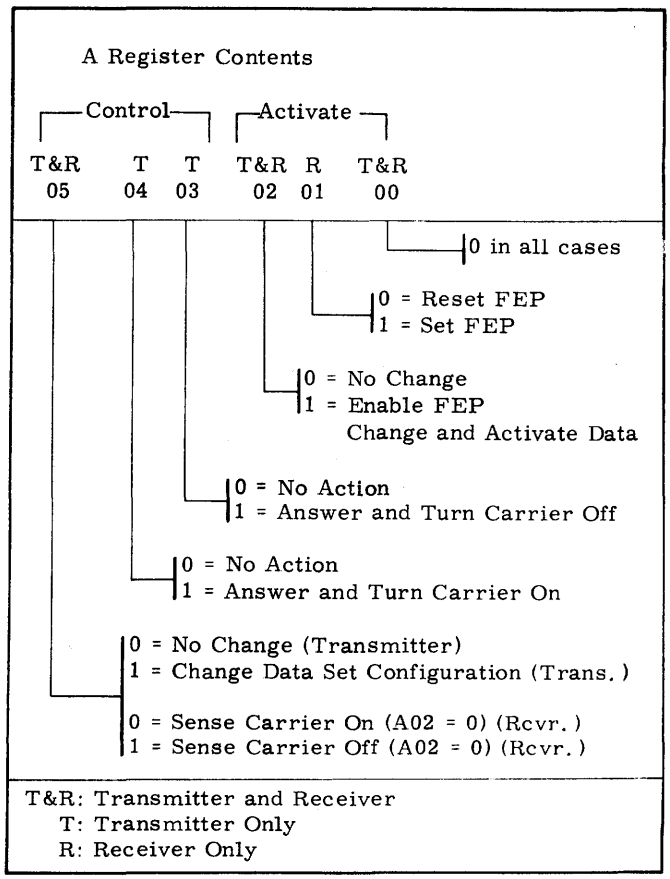


Table SCU II . 1 Control Register Instructions

- Receiver Channel

  004_8 - Activate data without FEP.

  006_8 - Activate data with FEP.

  000_8 - Sense carrier on.

  040_8 - Sense carrier off.

  100_8 - Deactivate data.

  140_8 - Deactivate data and sense carrier off.

Transmitter Control Register

The transmitter control register consists of gates G1BEOR, G1CLNC, and G1CACT, and flip-flops F1CB04, and B03. The effect of the control register instruction on the control register, the SCU II and the data set are as follows. ("X" in the place of a digit indicates any allowable numeral.):

- $06X_8$ - Answer call with carrier on.

  1. The program issues an OPR command with this instruction in response to an end-of-record interrupt caused by an incoming ring signal, or because the test and alarm logic has detected a condition requiring that the modem transmitter carrier be turned on, or because the program requires that the carrier be turned on. In half-duplex operation, this instruction is used to initiate or answer a call in which the SCU II transmits data. In full duplex operation, this instruction is used for transmission and reception.

  2. Gate G0COPR is enabled at phase A of the OPR command. With data bit B05 set, a positive pulse is produced by G1CLNC.

     G0COPR = OPR· PHA.
     G1CLNC = OPR· B05.

  3. Data bit B04 is set and B03 is not set. The LNC pulse sets F1CB04, and clears F1CB03.

  4. "Data terminal ready" is applied to the data set by X1TDTR.

     X1TDTR = G1TDTR = B03 + B04.

  5. In response to DTR, the data set applies "data set ready" to the SCU II, enabling M0TDSR.

  6. "Request to send" is applied to the data set by line driver, X1TRTS.

     X1TRTS = B04· DSR.

  7. In response to RTS, the modem carrier is turned on, and the data set then applies "clear to send" to the SCU II, enabling M0TCTS.

  8. The data set and the SCU II have now acquired the condition requested by the instruction. G0TCHB is therefore disabled, having been enabled from the time the instruction was received. An end-of-record interrupt is developed at the output of G1BCHB, unless the instruction also activated data transfer.

     $\overline{G1BCHC} = \overline{CB1}\cdot \overline{RNG}\cdot \overline{CB2}\cdot \overline{CHB}\cdot \overline{OPR}$.
     $\overline{G0TCHB} = CTS\cdot DSR$.
     G0TCTS = B04· CTS.
     G0TDSR = DTR· DSR.

  9. Flip-flop F1BSN1 is set by the first character rate, SCO, clock after receipt of CTS. F1BSN2 is set at the next SCO clock. +3.6V appears at the output of G1BSNC so long as either of SN1 or SN2 is set, and the SNC signal maintains the DTR and RTS signals, in the event of any control register change, until the data set removes CTS.

     $F1BSN1 = \overline{SN2}\cdot B04\cdot CTS\cdot SC0$.
     $F1BSN2 = SN1\cdot CTS\cdot SC0$.
     $\overline{F1BSN1} = SN2\cdot CTS\cdot SC0$.

- 050_8 - Answer call with carrier off

  1. The program issues an OPR command with this instruction in response to an end-of-record interrupt caused by an incoming ring signal, or because the test and alarm logic has detected a condition requiring data reception, or because the program requires that a call be initiated and data received. This instruction is used only for half-duplex operation.

  2. Gate G0COPR is enabled at phase A of the OPR command. With data bit B05 set, a positive pulse is produced by G1CLNC.

  3. Data bit B04 is clear and B03 is set. The LNC pulse clears F1CB04 and sets F1CB03.

  4. "Data terminal ready" is applied to the data set by X1TDTR.

  5. In response to DTR, the data set applies "data set ready" to the SCU II, enabling M0TDSR.

  6. B04 is not set, and "request to send" is not applied to the data set. The modem carrier is not turned on, and "clear to send" is not applied to the SCU II.

  7. The data set and the SCU II have now acquired the condition requested by the instruction. G0TCHB is therefore disabled, having been enabled from the time

the instruction was received. An end-of-record interrupt is developed at the output of G1BCHB.

$\overline{G1BCHB} = \overline{CB1} \cdot \overline{RNG} \cdot \overline{CB2} \cdot \overline{CHB} \cdot \overline{OPR}$.
$\overline{G0TCHB} = \overline{CTS} \cdot \overline{DSR}$.
$G0TCTS = \overline{B04} \cdot \overline{CTS}$.
$G0TDSR = DTR \cdot DSR$.

- 040_8 - Disconnect call

1. The program issues an OPR command with this instruction in response to an end-of-record interrupt caused by an end-of-message character in the transmitter or receiver data stream, or because the test and alarm logic has detected a condition requiring that a call be abandoned, or because the program requires that the call be abandoned.

2. Gate G0COPR is enabled at phase A of the OPR command. With data bit B05 set, a positive pulse is produced by G1CLNC.

3. Data bits B04 and B03 are clear. The LNC pulse clears F1CB04 and F1CB03.

4. The "data terminal ready" signal is turned off.
   $\overline{XITDTR} = \overline{GITDTR} = \overline{B04} \cdot \overline{B03}$.
5. The data set turns off "data set ready" and "clear to send".

6. The data set and SCU II have now acquired the condition requested by the instruction. G0TCHB is therefore disabled, having been enabled from the time the instruction was received. An end-of-record interrupt is developed at the output of G1BCHB.

$\overline{G1BCHB} = \overline{CB1} \cdot \overline{RNG} \cdot \overline{CB2} \cdot \overline{CHB} \cdot \overline{OPR}$.
$\overline{G0TCHB} = \overline{CTS} \cdot \overline{DSR}$.
$G0TCTS = \overline{B04} \cdot \overline{CTS}$.
$G0TDSR = \overline{DTR} \cdot \overline{DSR}$.

- $0X4_8$ or $0X6_8$ - Activate data

1. This instruction must be issued only with or after an "Answer Call" instruction. Gate G0COPR is enabled at phase A of the OPR command. Data bit B02 is set and a positive pulse produced by G1CACT.

   $G1CACT = OPR \cdot B02$.

2. The ACT pulse, inverted by G0BACT, sets F1BCB1. With CB1 set, gate, G1BCHB, is disabled, holding the channel busy line "true".

   $G0BACT = ACT \cdot B04$.

3. F1BSN2 should be set, "clear to send" applied by the data set, and F1BSN1 clear. (See "Answer call with carrier on").

   $F1BSN1 = SC0 \cdot SN2 \cdot CTS$.

4. When the conditions of 4, above, have been met, F1BCB2 is set, at least one character times later than CB1.

   $F1BCB2 = \overline{SN1} \cdot CB1 \cdot SN2 \cdot SC0$.

5. The setting of CB2 sets F1BDER, through its DC set terminal, via a differentiating network. G0BDER is enabled, producing the first data exchange interrupt.

- 100_8 - Deactivate data

1. This instruction is used to deactivate the transmitter channel, when the channel is in Transparent Mode, and no end of message character detection is possible. Data bit B06 is set and the output of G1BEOR goes "true" during the OPR pulse.

   $G1BEOR = OPR \cdot B06$.

2. The EOR pulse enables G0XETX, which sets F1XETX. This makes G1XTRM "true", F1BCB1 clears at the next SC0 pulse, and F1BCB2 clears at the following SC0 pulse. The transmitter channel is thereby made "not busy" after the "deactivate data" instruction is issued.

3. Since the program may not issue an OUT instruction or a TOM may not be allowed after the "deactivate data" instruction, provision is made to clear F1BDER, should it have been left set.

   $\overline{F1BDER} = \overline{CB1} \cdot SC0$

Receiver Control Register

The receiver control register of gates G0BEOR, G1COPR, and G1CACT, and flip-flops F1CB05, and F1CB01. The effect of the control register instructions on the control register and the SCU II are as follows:

- 004_8 - Activate data without FSC and FEP

1. The program issues an OPR with this instruction after $06X_8$ or 050_8 - "answer call" - has been issued to the transmitter channel, and after the transmitter channel has developed an end-of-record interrupt, indicating that the SCU II and data set have acquired the requested condition.

2. Gate G0COPR is enabled at phase A of the OPR command The OPR pulse is inverted by G1COPR. With data bit B05 clear,

the positive OPR pulse clears F1CB05.

3. With data bit B02 set, the zero volt OPR pulse is inverted by G1CACT.

   G1CACT = OPR· B02.

4. Data bits B01 and B00 are clear; F1CB01 and F1CB00 are cleared by the ACT pulse.

   G0BACT = G1BACT· $\overline{B05}$.

5. The positive ACT pulse is inverted by G0BACT, and the negative going ACT pulse sets F1BCB1 through the DC set terminal. The negative going waveform from the "zero" side of F1BCB1 is delayed 50 nanoseconds through delay line, DL1. After the delay, G1BCHB is disabled, holding the channel busy line in the "true" state.

- 006_8 - Activate data with FEP

1. The same action as steps 1, 2, 3, and 5, under "004_8", above, takes place, setting the channel busy.

2. Data bit B01 is set and F1CB01 is set by the ACT pulse.

3. With F1CB01 set, G1XSOR is partially enabled, such that only SOH, STX, and DLE characters are recognized as the first character in a message. For further analysis, see "Front End Protection".

- 000_8 - Sense carrier on

1. The program would normally use this instruction in half-duplex operation, in which the received "carrier on" signal is used to indicate if line turn-around is completed. The instruction could be issued after 050_8 was issued to the transmitter channel to turn off the local transmitter carrier. When the distant transmitter carrier is received, a receiver end-of-record interrupt is produced as follows.

2. The positive OPR phase A pulse from G1COPR clears F1CB05, as data bit B05 is clear. Data bit B02 is also clear and no ACT pulse is produced.

3. For the duration of any OPR phase A pulse in the receiver channel (approx. 2 or 8 usec.), G1BCHB is disabled, holding the channel busy line "true". Each OPR pulse also attempts setting F1BCON.

4. If the received carrier is on, or when it comes on, line receiver M0TCON is enabled. G1TCON then enables G0TCON, clearing F1BCON, if there was sufficient time between OPR and CON for the flip-flop to set. The end-of-record interrupt is then produced.
   $\overline{G1BCHB}$ = $\overline{CB2}$· $\overline{CB1}$· $\overline{OPR}$· $\overline{F1BCON}$.
   G0TCON = $\overline{B05}$· CON.

- $X40_8$ - Sense carrier off

1. The program would normally use this instruction in half-duplex operation, when an end-of-record interrupt is required to signal the program that the distant station has turned its carrier off and is now ready to receive a transmission. If this instruction is issued to a busy receiver channel, the channel will be deactivated and set not busy when the carrier is turned off.

2. The positive OPR phase A pulse from G1COPR sets F1CB05, as data bit B05 is set. Bit B02 is clear and no ACT pulse is produced.

3. For the duration of any OPR phase A pulse in the receiver channel, G1BCHB is disabled, holding the channel busy line "true". Each OPR pulse also attempts setting F1BCON.

4. If the received carrier is off. or when it goes off, line receiver M0TCON is disabled. $\overline{M0TCON}$ then enables G0TCON, clearing F1BCON, if there was sufficient time between OPR and $\overline{CON}$ for the flip-flop to set. The end-of-record interrupt is then produced.

   G0TCON = B05· $\overline{CON}$

5. If the instruction is issued to a busy receiver, F1BCB1 is immediately cleared, and F1BCB2 is cleared when the carrier goes off.

   $\overline{F1BCB1}$ = G0BCB1 = B05.
   F1BCB2 = G0BCBR.
   G0BCBR = $\overline{CB1}$· G1BCBR.
   G1BCBR = RBC = B05· $\overline{CON}$.

- $1X0_8$ - Deactivate data

1. This instruction is used to deactivate the receiver channel, when the channel is in Transparent Mode, and no end of message character detection is possible. Data bit B06 is set, and G0BEOR is enabled by the OPR pulse.
   G0BEOR = OPR· B06.

2. The EOR pulse clears both channel busy flip-flops which inhibits any further setting of F1BDER. When G1BCHB goes "not true", if DER was left set, it is cleared by $\overline{CHB}$.

Automatic Answer

When the data set has been called from a distant point, a momentary ring signal, of approximately 600 milliseconds duration, is applied to the ring indicator line, which is connected to line receiver M0BRNG, in the SCU II transmitter channel. The RNG signal disables and then enables G1BCHB, if the transmitter channel is not busy and if no OPR pulse is present. When CHB is enabled, an end-of-record interrupt is produced, informing the Central Processor that an incoming call is present, and an answer is required.

In response to the interrupt, the program issues an OPR command to the transmitter channel with an "answer call" instruction.

The program may determine if transmission or reception is appropriate by monitoring the test and alarm lines through the JNE logic.

Line Turn-Around

"Line turn-around" is an automatic change from reception to transmission or from transmission to reception. When the SCU II is used with half-duplex data sets, the requirement for a change in the direction of data flow is detected in the receiver channel and is implemented by the program and the transmitter channel. When full duplex communications devices are in use, no such feature is required.

The following conditions indicate that "turn-around" is required or is completed:

1. The received carrier is turned off while the receiver channel is not busy.

2. The received carrier is on while the receiver channel is busy, and subsequently the receiver channel goes not busy, as the last character in a message is received.

3. The received carrier is turned on while the receiver channel is not busy.

Conditions 1 and 2 indicate that a change from reception to transmission is required. Condition 3 indicates that the distant station has completed a turn-around. Each of these conditions causes a receiver channel end-of-record interrupt. The program may determine which condition caused the interrupt by monitoring the test and alarm lines through the JNE logic.

Conditions 1 and 3 are detected by the receiver channel as described under the Receiver Control Register heading, sub-headings, "000_8 - Sense carrier on", and "$X40_8$ - Sense carrier off". Condition 2 produces an end-of- record interrupt as described under "Receiver End of Message Detection".

Data Transmission

The transmitted data line from the SCU II to the data set is always held in the marking (true or one) state if the "clear to send", CTS, signal from the data set is not on. In the absence of the CTS signal, the transmitted data line driver is disabled, holding the transmitted data line at a negative voltage level. A schematic of the line driver, X0DMRK, appears on sheet 20 of the logic drawing, 68C992325.

The data set turns on the CTS signal in response to an "answer call with carrier on" instruction from the Central Processor, and after the data set transmitting carrier has been turned on. Between the time when CTS comes on and the transmission of the first character in a message, the SCU II provides for the transmission of at least four SYN, 026_8, characters. SYN characters are transmitted from the time when CTS comes on, until the first message character is transferred from the transmitter holding register to the shift register. As will be shown, this takes at least four character periods.

Figure SCU II . 3, SYN, 026_8, Character Transmission illustrates the transmission of the first SYN character after CTS is turned on. Line driver X0DMRK is enabled and produces a positive-space output, only, when CTS is present and shift register stage F1DSR1 is clear. The character to be transmitted is gated into the shift register by arming each shift register stage to clear with the positive character-rate clock from D1QSC0. Those shift register stages, which correspond to a binary zero in the character, are cleared at the trailing edge of the baud rate clock from D0QSCT. The remaining shift register stages remain set. A set or clear signal is provided to F1DSR1 as that stage is the only shift register stage not assured of being set by the end of a character transmission period, and must be set or cleared as required by the next character.

Gates G0XSYN and G0XSRP are enabled from the time CTS comes on, until both channel busy flip-flops have been set.

$$\overline{G0XSYN},\ \overline{G0XSRP} = \overline{CBY} \cdot CTS$$
$$\overline{G0BCBY} = \overline{CB1} + \overline{CB2}$$

Note that the SYN character has binary bits 01, 02, and 04 set, and that these correspond to character channels 2, 3, 5, respectively. The SYN and SRP signals disable G1DSR8, SR7, SR6, SR4, and SR1, such that the corresponding shift register stages are armed to clear and are cleared by the SC0 and SCT pulses. Shift register stages F1DSR5, SR3, and SR2 remain set. When the positive D1QSC0 pulse has passed, the D0QSC0 output becomes positive and the register is armed to shift at the trailing edge of each baud rate clock from D0QSCT. The 026_8 character is thereby shifted through the register, producing the serialized output from X0DMRK. The next D1QSC0 clock, and each thereafter, so long as G0XSYN and G0XSRP remain enabled, gates a new SYN character into the shift register, and no interval occurs between the serial characters.

The timing of a typical Non-Transparent Mode message transmission is shown on Figure SCU II . 4. In the

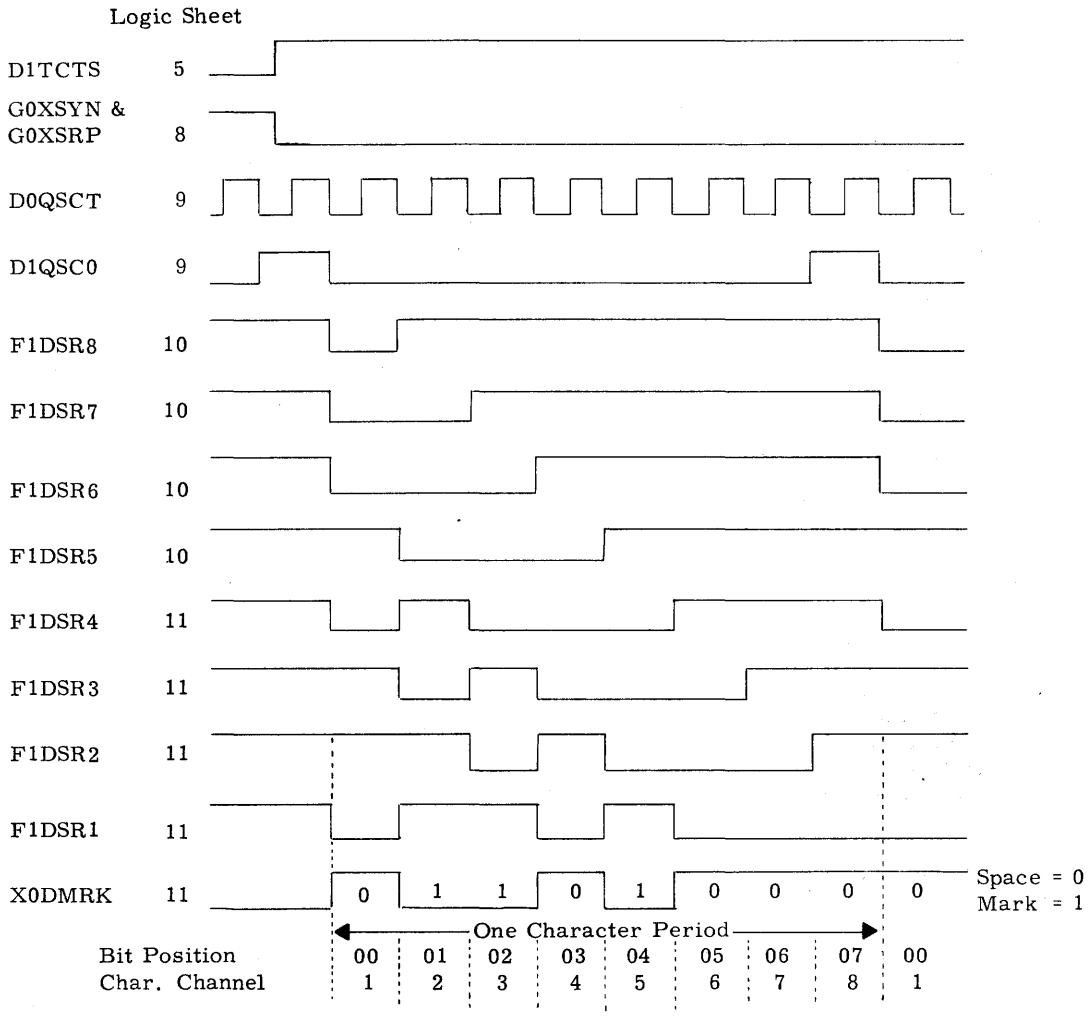


Fig. SCU II . 3 SYN, 026_8, Character Transmission

illustration, the transmitter data activate instruction occurs some time after CTS is received from the data set. The activate instruction could have been issued by the program at any time, from coincident with the "answer call" instruction, to anytime later. As described above, and as shown on Fig. SCU II . 4, SYN characters are transmitted during the character periods between SC0 pulses, until the first character in a message has been gated into the shift register. The first character in a message is normally STX, 002_8, or SOH, 001_8.

The transfer of the first character in a message from the basic I/O Buffer to the transmitter channel shift register is accomplished as follows:

1. The "activate data" instruction causes the channel to go busy and the first data exchange interrupt to be generated, in a minimum of 3 character periods, as shown on Fig. SCU II . 4, and as described under "Transmitter Control Register".

2. The program issues an OUT instruction with the first character. Normally, the OUT arrives in the I/O Buffer in less than one character period after the interrupt. The positive pulse from G1COUT is inverted twice, to produce a positive pulse out of D1COUT.

   G1COUT = OUT· ADD.
   ADD = Transmitter channel addressed.

3. The D1COUT pulse partially enables holding register gates G1DHR8 through G1DHR1. Each holding register flip-flop corresponding to an I/O Buffer data line containing a logical one is set. All other holding register stages are cleared.

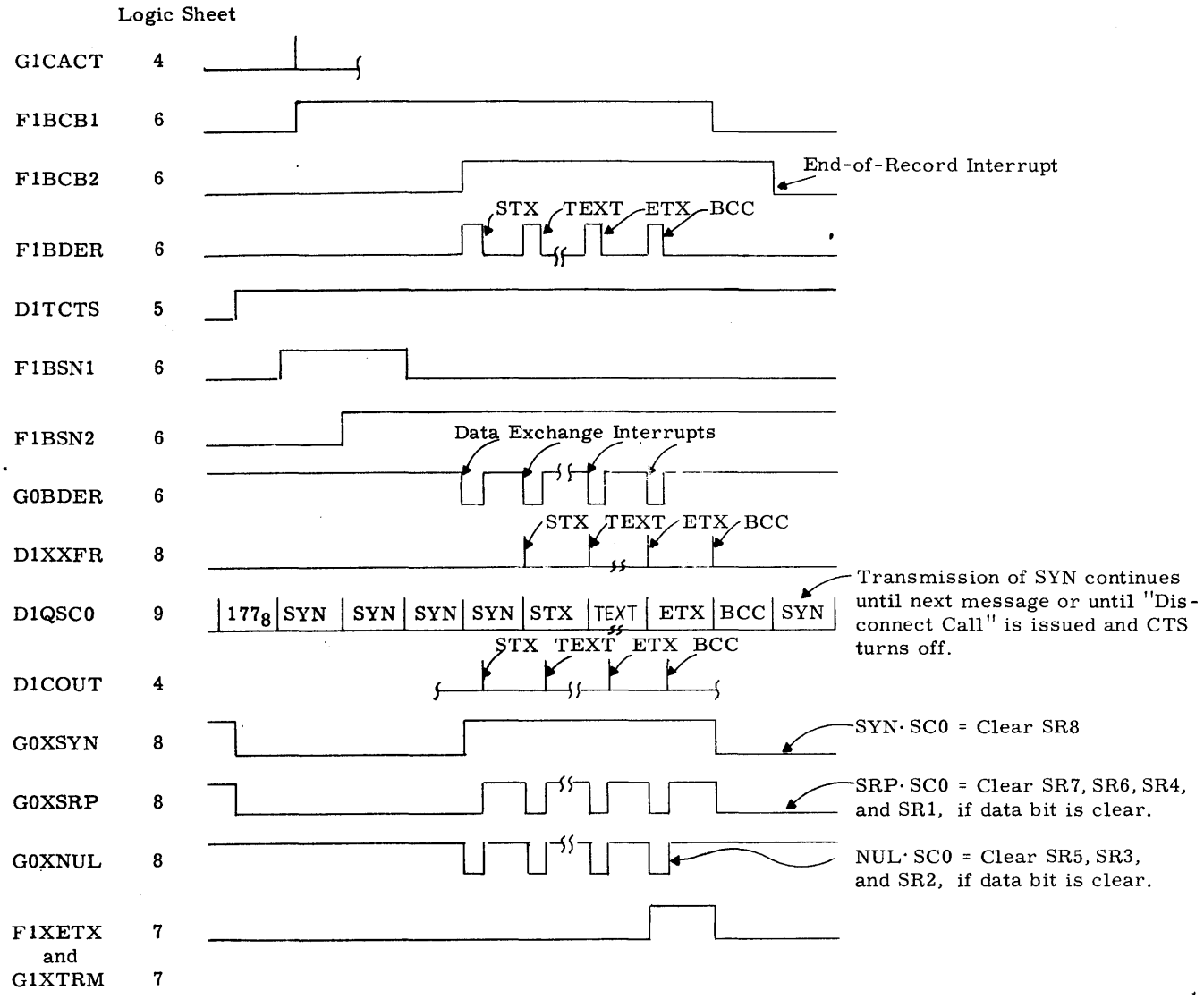


Fig. SCU II. 4. Message Transmission

4. F1BDER is cleared at phase B of the OUT instruction.

   $\overline{\text{F1BDER}}$ = PHB.
   G0CPHB = OUT· PHB.

5. G0XXFR is enabled by the next D1QSC0 pulse. The same SC0 pulse again sets F1CDER. The XFR pulse is inverted and partially enables register transfer gates G0DSR8 through G0DSR1. Those transfer gates connected to a holding register stage containing a binary zero are enabled and the character is transferred to the shift register. G0XNUL, SRP, and SYN are all disabled at this time. The enabled transfer gates arm the corresponding shift register stages to clear at the trailing edge of the SC0 pulse. The remaining shift register stages remain set.

   $\overline{\text{G0XSYN}}$ = CBY.
   $\overline{\text{G0XXFR}}$ = $\overline{\text{DER}}$· CTS· SC0· $\overline{\text{SYN}}$.
   $\overline{\text{G0XSRP}}$ = CBY· $\overline{\text{DER}}$.
   $\overline{\text{G0XNUL}}$ = $\overline{\text{DER}}$.

As was described in the case of the SYN characters, the first character in the message is shifted through the shift register, and through X0DMRK. The eight bit shift is accomplished by the eight baud rate clock pulses, SCT, which occur between each pair of character-rate, SC0, clock pulses. The serialized character is thereby placed on the transmitted data line to the data set.

So long as no terminate signal is present, and while CB2 is set, each SC0 pulse sets F1BDER, producing another data exchange interrupt. The remaining characters in the message are transferred to the holding register by OUT instructions, in response to the interrupts. At the

next SC0 pulse, the holding register contents are transferred to the shift register by the XFR pulse. Note that while a character is being serialized by the shift register, the holding register acquires and stores the next character to be serialized.

$$F1BDER = CB2 \cdot CB1 \cdot \overline{TRM} \cdot SC0 \cdot CB1.$$

Each character in the holding register is accompanied by an even parity bit in F1DHR8. The parity bit is supplied to HR8, either by the parity generator in the basic I/O Buffer, or by the Central Processor, through the basic I/O Buffer data gates. The source of the parity bit is selected by a jumper connected between one of the two pairs of green pin jacks on the PSTB1 circuit board. If the pin jacks labled "P" are jumpered the parity bit is supplied by the Central Processor. If the jacks labled "S" are jumpered, the basic I/O Buffer supplies the parity bit is inverted, and an odd parity bit, therefore, always accompanies the transmitted data.

Should the program not be able to respond to a data exchange interrupt before the trailing edge of an SC0 pulse, which marks the start of the next character period, the entire shift register is cleared and a NUL, 000_8, character is transmitted. If, after an SC0 pulse sets F1BDER, producing a data exchange interrupt, no OUT instruction is issued, DER remains set at the next SC0 pulse. In such a case, G0XSRP and G0XNUL would remain enabled at the trailing edges of the SC0 pulse and the coincident SCT pulse. Each of the shift register stages would be armed to clear by the resultant positive voltage at the outputs of G1DSR7 through G1DSR1, and each stage would be cleared at the trailing edge of the SCT pulse. All zeros would then be transferred through X0DMRK, during the following character period, replacing the missed data character with NUL. G0XSYN is not enabled at this time, and G1DSR8 would remain set, providing the odd parity bit in the NUL character.

Transmitter End of Message Detection

When the final character in a message has been transmitted, the transmitter channel goes not busy and an end-of-record interrupt is produced. The characters shown on Table SCU II . 2 are either the final character in a non-transparent message, or indicate that the final character is about to be transferred. Since the binary data in transparent text can be in any form, the end of message detection circuits, shown on sheet 7 of the logic drawing, 68C982325, are disabled in Transparent Mode. The program determines when the final character is transferred and issues a "deactivate data" instruction to the transmitter channel, which has the same effect as the detection of an end of message character, when transparent text is transmitted.

Fig. SCU II . 4 shows the timing for a typical complete non-transparent message transmission, where the message starts with an STX character, several text characters follow, then an ETX character is transferred followed by a block check character (BCC).

The analysis of the end of message detection logic is simplified by considering that detection of any of the end of message characters, or receipt of a "deactivate data" instruction, leads to developing a "true" positive output from G1XTRM during the character period just preceeding the transmission of the final character in a message. The TRM signal arms channel busy flip-flop F1BCB1 to clear, and CB1 is cleared by the SC0 pulse which marks the beginning of the character period during which the final character is transmitted. At the end of that character period, F1BCB2 is cleared, producing the end-of-record interrupt. F1BDER is cleared by the same SCO pulse.

$$\overline{F1BCB1} = CB2 \cdot TRM \cdot SC0$$
$$\overline{F1BCB2} = \overline{CB1} \cdot SC0$$
$$\overline{G1BCHB} = \overline{CB1} \cdot \overline{RNG} \cdot \overline{CB2} \cdot \overline{CHB} \cdot \overline{OPR}.$$

Table SCU II . 2 lists each of the possible end of message characters with the "true" logic terms, when the character is in the holding register and when the transparent mode flip-flop, F1XTSP, is clear. Note that EOT, ENQ, ACK, and NAK make G1XTRM true when they are in the holding register, and the channel goes not busy immediately after these characters are transmitted.

Characters ENQ, ACK, and NAK are not detected if they are preceeded by an SOH or STX character, which sets F1XSOH.

The ETX and ETB characters cause F1XETX to be set by the SC0 pulse which marks the beginning of the character period during which ETX or ETB is transmitted. These characters are always followed by a BCC character, and the channel remains activated for one additional character period, to allow transmission of the BCC.

$$\overline{F1XETX} = BCC \cdot ETX \cdot XFR \cdot SC0.$$
$$\overline{F1XETX} = XFR \cdot F1XETX$$
$$G1XTRM = ETX.$$

The DLE character causes F1XDLE to be set by the SC0 pulse which marks the beginning of the character period during which DLE is transmitted. This character is always followed by a sequence of characters which ends with any octal numeral from 060 to 077. G0XNUM is enabled by the final character in the sequence, producing the TRM signal.

$$F1XDLE = \overline{ETX} \cdot DLE \cdot XFR \cdot SC0.$$
$$G0XNUM = DLE \cdot HR6 \cdot HR5 \cdot \overline{HR7} = DLE \cdot (060_8 \text{ through } 077_8)$$
$$G1XTRM = NUM$$

The Transparent Mode is set up if a DLE character in the holding register is immediately followed by an STX (002_8) character, as follows:

1. DLE (020_8) enables G0XDLE and G0XCCC.

   $$G0XDLE = HR5 \cdot \overline{HR3} \cdot \overline{HR2} \cdot \overline{HR1}.$$
   $$G0XCCC = \overline{HR7} \cdot \overline{HR6} \cdot \overline{HR4} \cdot \overline{TSP}.$$

2. G1XDLE becomes "true" and F1XDLE is set by the XFR pulse which transfers the DLE

character to the shift register.

F1XDLE = $\overline{ETX}$· DLE· XFR· SC0.

3. The next character in the holding register is STX, which enables G0XSTX, CCC, and makes G1XSTX "true".

   G0XSTX = $\overline{HR2}$· D00.
   G1XSTX = G0XSTX· CCC.

4. The XFR pulse which transfers STX to the shift register, clears F1XDLE and sets F1XTSP.

   F1XTSP = SEQ· STX· XFR· SC0.
   $\overline{G1XSEQ}$ = DLE· $\overline{SEQ}$.
   $\overline{F1XDLE}$ = XFR· $\overline{HR7}$· $\overline{DLE}$· SC0.

5. While TSP is set, G0XCCC is disabled. As indicated in Table SCU II . 2, CCC must be "true" to enable any end of message character detection. TSP has therefore, disabled all end of message character recognition, and the program must deactivate the transmitter when appropriate by issuing "deactivate data", which sets F1XETX via G0XETX. If set, each of F1XETX, F1XDLE, F1XSEQ, F1XTSP, and F1XSOH is cleared when the transmitter channel goes "not busy".

   $\overline{F1XETX}$, $\overline{DLE}$, $\overline{SEQ}$, $\overline{TSP}$, $\overline{SOH}$=$\overline{CBY}$=$\overline{CB1}$· $\overline{CB2}$

Character	Octal Code	"True Logic Terms"
ETX	003	D03, D00, G1XETX, CCC F1XETX, TRM
EOT	004	EOT, CCC, TRM
ENQ	005	EOT($\overline{SOH}$), CCC, TRM
ACK	006	EOT($\overline{SOH}$), CCC, TRM
DLE	020	DLE, CCC, G1XDLE, F1XDLE
	060 } 077 }	F1XDLE, NUM, TRM
NAK	025	EOT($\overline{SOH}$), CCC, TRM
ETB	027	D03, D24, G1XETX, CCC, F1XETX, TRM

Table SCU II . 2. Transmitter End of Message Characters

Data Reception

The local data set receiver is enabled after one of the two "answer call" instructions is issued to the SCU II transmitter channel. In full-duplex systems, "$06X_8$ - answer call with carrier on" is normally used. In half-duplex systems, "050_8 - answer call with carrier off" is used. After one of these instructions has been issued, the data set makes the connection to the communications line and the received carrier detection circuit is enabled.

If the "sense carrier on" instruction has been issued to the receiver channel, detection of the received carrier by the data set causes the SCU II to generate a receiver channel interrupt which indicates to the Central Processor that data is about to be received and that the receiver channel should be activated.

An "activate" instruction may be issued to the receiver at any time. If no message has been received, data exchange interrupts will not occur until the first message character is received. Fig. SCU II . 5, Data Reception Timing illustrates a reception sequence in which "sense carrier on" has been issued to the receiver in response to the interrupt. Fig. SCU II . 5 is applicable to any of the two available "activate" instructions. See "Receiver Control Register" for a description of the effect of the "sense carrier on" and "activate" instructions.

When the distant data set has turned on its carrier and begins transmitting, the SCU II requires at least two SYN characters on the received data line before the first character in a message. There need be no SYN characters between successive messages when the received carrier remains on, but any character periods between messages should contain SYN characters. Prior to the reception of the first of two SYN characters after the received carrier is detected, two of the receiver baud rate clock drivers and the clock countdown circuit are disabled, as G0TSNC is enabled because the synchronization flip-flops F1TSN1 and F1TSN2 have not yet set.

D1QSC1, SC2 = G0TSNC
G0TSNC = $\overline{SN1}$· $\overline{SN2}$

With the received carrier on, G1DMRK and the most significant receiver shift register stage are partially enabled. Marks on the received data line arm F1DSR8 to set and spaces arm SR8 to clear. All of the shift register stages are triggered by the receiver baud rate shift clock from D1QSCR and the received characters are shifted through the register. If the data set is not the timing signal source for the SCU II, the SCR pulses may not be in exact synchronism with the incoming data bit periods, but they will be close enough to assure that each bit is shifted into its respective position in the shift register by every eighth SCR clock pulse.

With an SYN character, character channels 2, 3, and 5 are set (026_8). When a complete SYN character is held by the shift register, F1DSR2, SR3, and SR5 are set, and all other shift register stages are clear. This condition holds for one bit period, as the next SCR pulse shifts the first bit on the next character into the register. As shown on Fig. SCU II . 5, receipt of SYN causes the output of G1XSYN to become "true", as G0XCCP and G0XSYN are enabled when the shift register holds the character.

G1XSYN = CCP· SYN = 026_8
G0XCCP = $\overline{SR7}$· $\overline{SR6}$· $\overline{SR4}$· $\overline{SR8}$ = $00X_8$ or $02X_8$
G0XSYN = SR5· SR3· SR2· $\overline{SR1}$ = 026_8 or 036_8

The SCR clock at the trailing edge of the SYN pulse sets F1TSN1.

F1TSN1 = SYN· CON· SC0· SCR.

With SN1 set, G0TSNC is disabled. As the G0TSNC signal swings positive, the outputs of D1QSC1 and SC2 are driven in a negative direction, releasing the count-down flip-flops and incrementing the count from zero to one. As is shown on Fig. SCU II . 5, positive D1QSC0 character-rate pulses are developed during the first bit period of each incoming character. The SC0 pulse is present during the periods when the shift register holds a complete character.

The synchronization sequence is shown in greater detail on Figure SCU II . 6. The counter operation and generation of SC0 pulses is illustrated on Fig. SCU II . 2.

F1TSN2 is set upon receipt of a second SYN character.

F1TSN2 = SYN· SN1· SC0· SCR.

The output of G1TSNC becomes "true" when both SN1 and SN2 are set. The first channel busy flip-flop, F1BCB1, was set by the ACT pulse when the "activate" instruction was issued to the receiver. F1BCB2 has been awaiting receiver synchronization and receipt of the first character in a message with valid parity sense, before it sets and allows data transfer. The SNC signal indicates synchronization. In the case shown in Fig. SCU II . 5, the output of start of record gate G1XSOR becomes "true" when the shift register holds the STX character.

If the Front End Protection (FEP) feature is enabled, the first character in a message must be SOH, STX, or

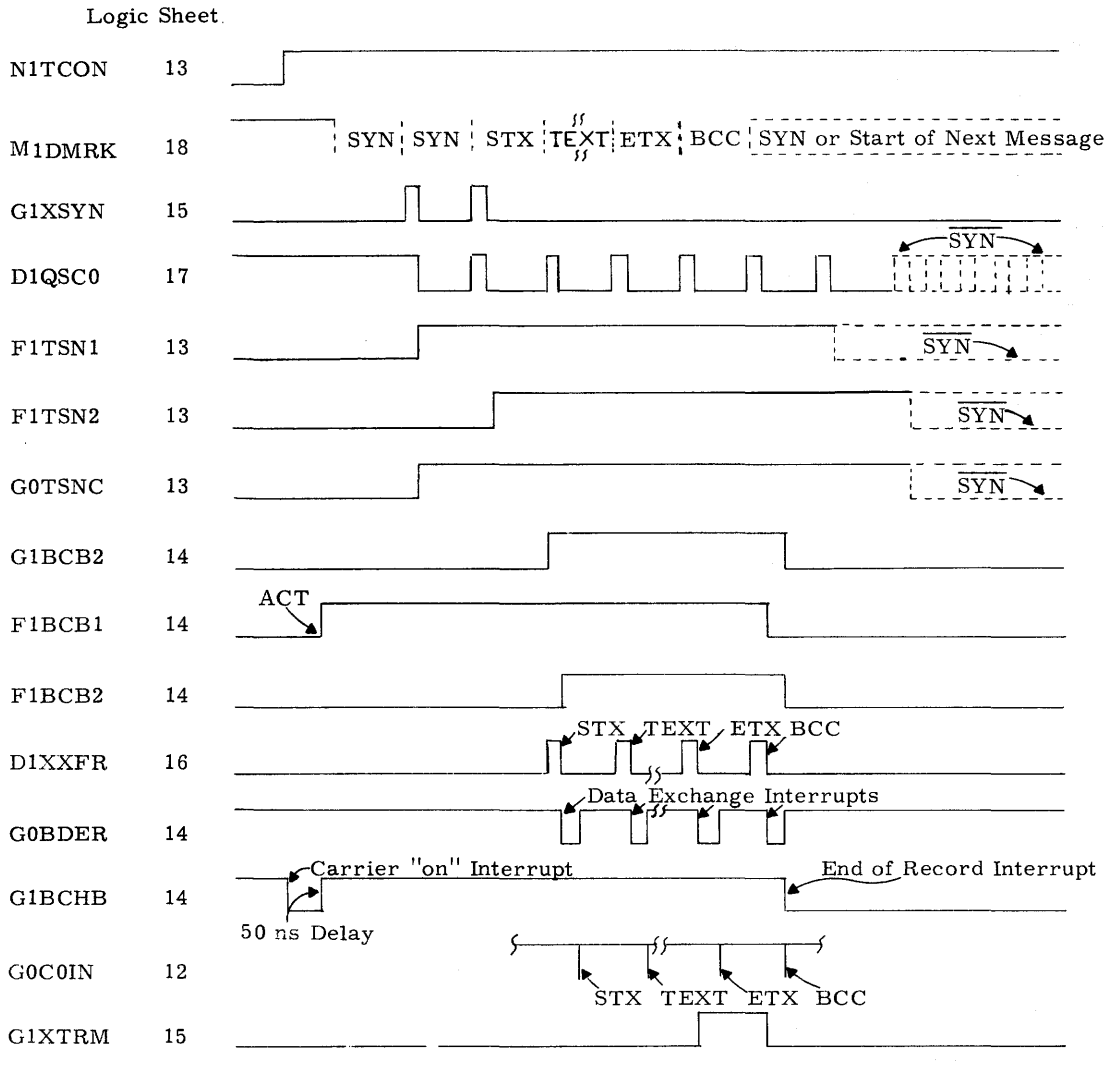


Fig. SCU II . 5. Data Reception

DLE, in which case, the sequence described under "Front End Protection" leads to the SOR signal. Otherwise the first non-SYN character is detected as described in the following.

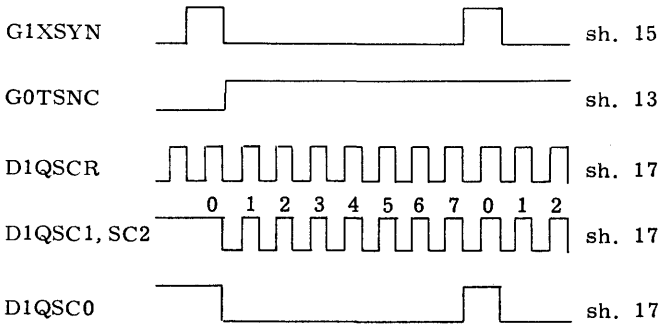


Fig. SCU II . 6. Receiver Synchronization

All characters shifted through the shift register are checked for odd parity sense by F1TODD. The operation of this circuit is illustrated on Fig. SCU II . 7. As shown in the illustration, receipt of a character with an even number of bits set causes ODD to equal SR8 during the SC0 pulse following the character reception. If the parity sense is correct, ODD does not equal SR8 during SC0. During SC0, SR8 contains the character parity bit.

With FEP not enabled, two of the four AND gates on G1XSOR are disabled. The remaining two AND gates on SOR are both disabled, driving the output to +3. 6V, when SR8 is not equal to ODD. If incorrect parity is transmitted by the distant data set, or if noise or some fault scrambles the data in a character such that a bit is lost and an even number of bits set, one of the two AND gates is enabled, driving the SOR output to zero volts. Under these conditions, then, any character (including SYN) with odd parity causes the output of G1XSOR to be "true" during SC0.

$$G1XSOR = \overline{B01} \cdot (\overline{SR8} \cdot ODD + SR8 \cdot \overline{ODD})$$

F1BCB2 sets when the receiver is synchronized and a valid non-SYN character at the start of a message is received.

$$F1BCB2 = CB2 \cdot SC0 \cdot SC1$$
$$N1BCB2 = G0BCB2 = SOR \cdot \overline{SYN} \cdot SNC \cdot CB1.$$

The G0BCB2 signal which helped set F1BCB2, is inverted by G1BCB2. When F1BCB2 sets, the G1BCB2 signal is continued. This signal is "true" from the time of the SC0 pulse representing the first message character, until CB2 clears. Each SC0 pulse then enables G0XXFR. The XFR pulses gate the characters from the shift register to the holding register.

$$G1BCB2 = G0BCB2 + F1BCB2$$
$$D1XXFR = G0XXFR = SC0 \cdot CB1 \cdot G1BCB2$$

The same SC0 pulse which set F1BCB2, sets F1BDER for the first time. G0BDER is then enabled, producing the first data exchange interrupt.

$$F1BDER = G1BCB2 \cdot CB1 \cdot SC0 \cdot SC1$$
$$G0BDER = \overline{ACT} \cdot DER.$$

In response to the interrupt, the program issues an IN instruction to the receiver. The IN instruction enables G0COIN, and the zero volt OIN pulse clears F1BDER.

$$G0COIN = PHB \cdot GIN$$
$$G1CPHB = ADD \cdot PHB$$

The ADD pulse, produced on receipt of the IN instruction, partially enables each of G0DD07 through G0DD00. Each of these gates which is connected to a set holding register stage is enabled, transferring the character on parallel data lines to the basic I/O Buffer. The parity bit contained in F1DHR8 is transferred through G0DD07 to the parity check circuit in the basic I/O Buffer. The shift register always receives odd parity, and the polarity of the parity bit inverted as it is transferred from SR8 to HR8.

If required, the parity bit from G0DD07 is fed through the basic I/O Buffer to the Central Processor with even parity sense, as the green pin jacks labled "P" are jumpered. If these pin jacks are not jumpered, only the seven data bits from G0DD06 through G0DD00 are transferred to the Central Processor.

The receipt of message characters, transfer from the shift register to the holding register, generation of data exchange interrupts, and transfer of parallel data from the holding register to the Central Processor through IN instruction continues until an end of message character is detected in the input data, or until a "sense carrier off" or "deactivate" instruction is issued to the receiver.

Should the Central Processor not respond to a data exchange interrupt, F1BDER remains set into the next SC0 pulse time. This condition sets the receiver channel alarm flip-flop, turns on the I/O BUF alarm light on the operator's console, and enables a test line when a JNE command with K1 = $11X_2$ is issued to the receiver channel. This "loss of data" alarm condition is described in more detail under "Tests and Alarms - Receiver".

If the received carrier drops out while the receiver channel is activated, carrier off flip-flop, F1TCOF, is set, creating the same alarm condition as for "loss of data". F1BCB1 is cleared under this condition, inhibiting shift to holding register transfers and data exchange interrupts. Marks (binary ones) are continuously shifted through the shift register with no received carrier. If the received carrier comes back on, the alarm condition may be cleared by a new "activate data" instruction. Since F1TSN1 and SN2 are cleared where there is no carrier, at least two SYN characters

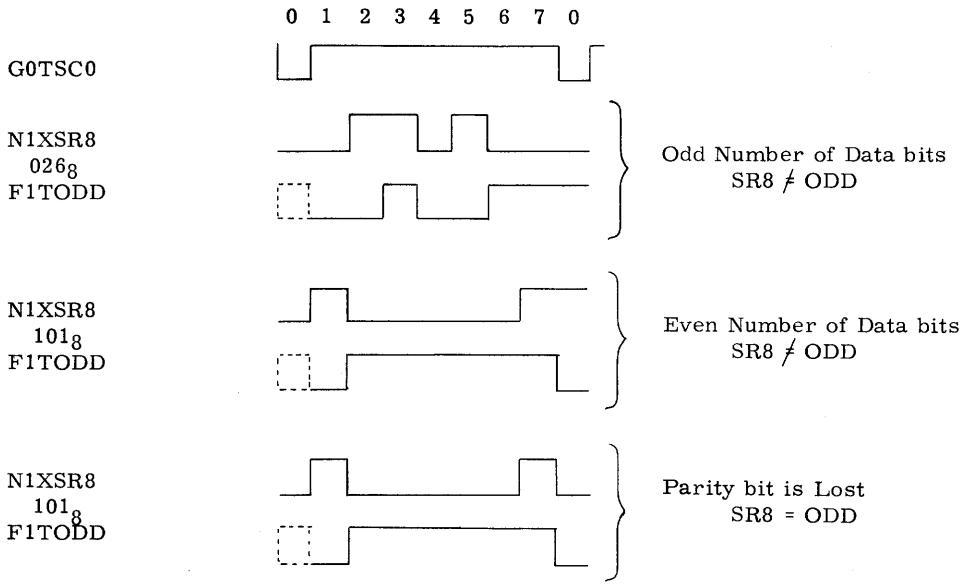


Fig. SCU II . 7. Parity Check Timing

must be received before new data can be received.

$\overline{F1BCB1}$ = G0BCB1 = CB2· $\overline{CON}$
F1DSR8 = G1DMRK· SCR
G1DMRK = $\overline{CON}$

If, for some reason, the receiver channel goes "not busy" ($\overline{F1BCB1}$ followed by $\overline{F1BCB2}$ at the next SCO pulse) while F1BDER is set, and before DER can be cleared by the next IN instruction or TIM operation, provision is made to clear F1BDER with the same SCO pulse which clears F1BCB1. For example, after "sense carrier off" has been issued to the receiver, and the received carrier goes off, the channel may go "not busy" while DER is set and awaiting another IN.

$\overline{F1BDER}$ = SC1 SCO $\overline{CB1}$

Front End Protection

Front End Protection (FEP) is enabled if, when the "activate data" instruction is issued to the receiver channel, Central Processor A register bit B01 is set. As described under "Receiver Control Register", an OPR command with such an instruction, sets control register flip-flop, F1CB01.

As is described under data reception, every character received should have odd parity sense, in which case F1DODD will not equal F1XSR8 during the SC0 pulse which follows the reception of each character. With FEP not enabled, this logic alone disables each of the four AND gates on G1XSOR, driving its output "true". When FEP is enabled, F1CB01 partially enables two additional AND gates on G1XSOR, making it necessary that, in addition to having odd parity sense, the first message character must be SOH (001_8), STX (002_8) or DLE (020_8). When any of these characters is held in the shift register, during SC0, both G0XCCC and G0XSOH or both G0XCCC and G0XDLE are enabled, making SOR "true", if parity is correct, and the receiver is not in Transparent Mode.

G1XSOR = B01· CCC· SOH· (SR8≠ODD)+
B01· CCC· DLE· (SR8≠ODD)

With FEP enabled then, gate G0BCB2, (which enables F1BCB1 to set and begin data transfer) is not enabled on any non-SYN character, but only when the character is SOH, STX or DLE.

G0BCB2 = SOR· $\overline{SYN}$· SNC· CB1
F1BCB2 = CB2· SC0· SC1

Receiver End of Message Detection

When the final character in a message has been received in the Central Processor, the receiver channel goes not busy and an end-of-record interrupt is produced. The characters shown on Table SCU II . 3 are either the final character in a non-transparent message, or indicate that the final character is about to arrive. Since the binary data in transparent text can be in any form, the end of message detection circuits, shown on sheet 15 of the logic drawing 68C982325, are disabled in Transparent Mode. The program determines when the final character has been received and issues a "deactivate data" instruction to the receiver channel, which has the same effect as the detection of an end of message character, when transparent text is received.

The analysis of the end of message detection logic is simplified by considering that detection of any of the end of message characters, or receipt of a "deactivate data" instruction, leads to a "true" output from G1XTRM during the XFR pulse which transfers the final message character from the shift register to the holding register.

Fig. SCU II . 5, shows the timing for the reception of a typical complete non-transparent message, where the message starts with STX, several text characters follow, then an ETX character is received, followed by BCC.

The TRM signal deactivates the receiver channel in the following manner:

1. The SC0 pulse and TRM arm F1BCB1 to clear.

   $F1BCB1 = EOR \cdot TRM \cdot SC0 \cdot SC1$
   $G1BEOR = CB2.$

2. The IN pulse which transfers the final character to the Central Processor clears F1BCB2 and generates an end-of-record interrupt.

   $\overline{F1BCB2} = CBR = \overline{CB1} \cdot G1BCBR.$
   $G1BCBR = RBI = \overline{COF} \cdot \overline{OIN}.$
   $G1BCHB = \overline{CB2} \cdot \overline{CB1} \cdot \overline{OPR} \cdot \overline{CON}.$

If neither SYN nor a valid SOR character is received following an end of message character, the following takes place:

3. If the channel is not reactivated and a new SOR character received before the next SC0 pulse, F1TSN1 clears.

   $\overline{F1TSN1} = SC0 \cdot \overline{SYN} \cdot \overline{SNR} \cdot \overline{SCR}.$
   $\overline{G1TSNR} = \overline{G0TSNR} = \overline{CB2} \cdot \overline{SOR}.$
   $\overline{G0TSOR} = \overline{SOR} + \overline{CB1}.$

4. If SN1 clears, F1TSN2 clears at the next SC0.

   $\overline{F1TSN2} = SC0 \cdot \overline{SN1} \cdot \overline{SYN} \cdot \overline{SCR}.$

5. When SN2 clears, G0TSNC is disabled and the clock counter is disabled.

Each of the possible end of message characters is listed in Table SCU II . 3, with the logic terms which are "true" when the character is held in the shift register and when F1XTSP is clear. Note that EOT, ENQ, ACK, and NAK make G1XTRM true during the SC0 pulse when they are held in the shift register, and the channel goes not busy immediately after the character is transferred to the Central Processor.

Characters ACK and NAK are not detected if they are preceeded by an SOH or STX character which sets F1XSOH.

The ETX and ETB characters cause F1XETX to be set by the XFR pulse which transfers these characters to the holding register. These characters are always followed by a BCC character, and the channel remains activated for one additional character period, to allow reception of the BCC.

$F1XETX = ETX \cdot XFR \cdot SC1$
$\overline{F1XETX} = XFR \cdot F1XETX$
$G1XTRM = ETX$

The DLE character causes F1XDLE to be set by the XFR pulse which transfers DLE to the holding register. This character is always followed by a sequence of characters which ends with any octal numeral from 060 to 077. G0XNUM is enabled by the final character in the sequence, producing the TRM signal.

$F1XDLE = \overline{ETX} \cdot DLE \cdot XFR \cdot SC1$
$G0XNUM = DLE \cdot SR7 \cdot SR6 \cdot SR5 =$
$DLE \cdot (060_8$ through $077_8)$

The transparent mode is set up if a DLE character in the shift register is immediately followed by an STX character, as follows:

1. DLE enables G0XDLE and G0XCCC.

   $G0XDLE = SR6 \cdot \overline{SR3} \cdot \overline{SR2} \cdot \overline{SR1}.$
   $G0XCCC = \overline{SR7} \cdot \overline{SR6} \cdot \overline{SR4} \cdot \overline{TSP}.$

2. G1XDLE becomes "true" and F1XDLE is set by the XFR pulse which transfers the DLE character to the holding register.

   $F1XDLE = \overline{ETX} \cdot XFR \cdot SC1$

3. The next character in the shift register is STX, which enables G0XSTX, CCC, and makes G1XSTX "true".

   $G0XSTX = \overline{SR5} \cdot \overline{SR3} \cdot SR2 \cdot \overline{SR1}.$
   $G1XSTX = G0XSTX \cdot CCC.$

4. The XFR pulse which transfers STX to the shift register, clears F1XDLE and sets F1XTSP.

   $F1XTSP = SEQ \cdot STX \cdot XFR \cdot SC1.$
   $G1XSEQ = DLE \cdot \overline{SEQ}.$
   $F1XDLE = XFR \cdot \overline{SR7} \cdot \overline{DLE} \cdot SC1.$

5. While TSP is set, C0XCCC is disabled. As indicated in Table SCU II . 3, CCC must be "true" to enable any end of message character detection. TSP has therefore, disabled all end of message character recognition, and the program must deactivate the receiver, when appropriate by issuing "deactivate data".

Character	Octal Code	"True Logic Terms"
ETX	003	G0XETX, CCC, G1XETX, F1XETX, TRM
EOT	004	EOT, CCC, TRM
ENQ	005	EOT, CCC, TRM
ACK	006	EOT($\overline{SOH}$), CCC, TRM
DLE	020	DLE, CCC, G1XDLE, F1XDLE
	060 } 077 }	F1XDLE, NUM, TRM
NAK	025	EOT($\overline{SOH}$), CCC, TRM
ETB	027	G0XETX, CCC, G1XETX, F1XETX, TRM

Table SCU II . 3. Receiver End of Message Characters

If set, each of F1XETX, F1XDLE, F1XSEQ, F1XTSP, and F1XSOH is cleared when the receiver channel goes "not busy".

$\overline{F1XETX}$, $\overline{DLE}$, $\overline{SEQ}$, $\overline{TSP} = \overline{CBY} = \overline{CB1} \cdot \overline{CB2}$

Tests and Alarms - Transmitter

The following test and alarm conditions may be detected in the Central Processor when properly encoded JDR, JCB, JNR, and JNE instructions are addressed to the SCU II transmitter channel. In each case, the test or alarm condition which causes the test line gate to be enabled is given. An enabled test line gate causes the alarm or ready lines from the basic I/O Buffer to the Central Processor to go to the "true" or "no jump" condition.

- The transmitter channel is not busy. JCB enables G0TTL1 (K1 = $01X_2$).

  $G0TTL1 = \overline{CHB} \cdot \overline{K14} \cdot ADD$

- The holding register is not ready to accept a new character. JDR enables G0TTL1, (K1 = $10X_2$).

  $\underline{G0TTL1} = \underline{ADD} \cdot \overline{DER} \cdot \overline{K12}$
  $\overline{G0BDER} = \overline{DER} + ACT$

- The transmitter is not in Transparent Mode. JNR enables G0TTL1 (K1 = $11X_2$).

  $G0TTL1 = \overline{TSP} \cdot K14 \cdot K12 \cdot ADD$

- "Clear to send" from data set does not agree with control register. JNE enables G0TTL2 (K1 = $10X_2$).

  $\underline{G0TTL2} = \overline{CTS} \cdot \underline{K14} \cdot \underline{K12} \cdot ADD$
  $\overline{G0TCTS} = CTS \cdot \overline{B04} + \overline{CTS} \cdot B04$

- DSR from the data set does not agree with DTR from the SCU II. JNE enables G0TTL2 (K1 = $11X_2$).

  $\underline{G0TTL2} = \underline{ADD} \cdot \overline{DSR} \cdot K14 \cdot \underline{\overline{K12}}$
  $\overline{G0TDSR} = \overline{DSR} \cdot DTR + DSR \cdot \overline{DTR}$

Tests and Alarms - Receiver

The following test and alarm conditions may be detected in the Central Processor when properly encoded JDR, JCB, JNR, and JNE instructions are addressed to the SCU II receiver channel. In each case, the test or alarm condition which causes the test line gate to be enabled is given.

An enabled test line gate causes the alarm or ready lines from the basic I/O Buffer to the Central Processor to go to the "true" or "no jump" condition.

- The receiver channel is not busy. JCB enables G0TTL1 (K1 = $01X_2$).

  $G0TTL1 = \overline{CHB} \cdot \overline{K14} \cdot ADD$

- No received character is held by the holding register. JDR enables G0TTL1 (K1 = $10X_2$).

  $\underline{G0TTL1} = ADD \cdot \overline{DER} \cdot \overline{K12}$
  $\overline{G0BDER} = ACT + \overline{DER}$

- The receiver is not in Transparent Mode. JNR enables G0TTL1 (K1 = $11X_2$).

  $G0TTL1 = \overline{TSP} \cdot K12 \cdot K14 \cdot ADD$

- The received carrier is off. JNE enables G0TTL2 (K1 = $01X_2$).

  $G0TTL2 = \overline{K14} \cdot K12 \cdot \overline{CON} \cdot ADD.$

- The carrier off flip-flop, F1TCOF, or the parity flip-flop, F1TPTY, is set. In addition to enabling the alarm line, each of these flip-flops, when set, causes the I/O BUF light on the operator's console to be lit. JNE enables G0TTL2 (K1 = $11X_2$).

  $G0TTL2 = ADD \cdot K14 \cdot K12 \cdot ALM.$
  $G1TALM = COF + PTY$

The two alarm flip-flops, COF, and PTY, are cleared each time an OPR command with a new control register instruction is issued to the receiver channel. They may also be cleared by depressing the ALARM CLEAR pushbutton on the operator's console or with an ABT command. Each of these flip-flops is normally clear during data transfer.

If, while the receiver channel is activated, the received carrier drops out, F1TCOF sets.

$F1TCOF = G0TCOF = CB2 \cdot \overline{CON}$

If, during an IN instruction, the parity check circuit in the basic I/O Buffer detects an odd number of bits set, F1TPTY is set.

$F1TPTY = G0TPTY = PAB \cdot PHA \cdot GIN \cdot DER.$

If F1BDER remains set until the trailing edge of the next SC0 pulse, a new character has replaced the contents of the holding register and the previously held character has been lost. This condition sets F1TPTY.

$G0TPTY = DER \cdot CB1 \cdot SC0 \cdot SC1$

Abort

An abort command, ABT (S = 3_8), addressed to either SCU II channel causes any operation in progress to be terminated. Depressing the ON pushbutton on the operator's console with the console enabled and the computer on manual operation, has the same effect as the ABT command.

An abort command issued to the transmitter channel enables G0CABT (Logic sheet 4). If set, the G0CABT pulse clears the following flip-flops: F1BCB1, F1BCB2, F1BDER, F1CB04, and F1CB03. With CB1, CB2, and DER clear, data transmission is terminated. With B04 and B03 clear, the "data terminal ready" signal is removed from the data set, and the data set goes to the disconnected state.

G0CABT = ADD· ABT

An abort command issued to the receiver channel enables G0CABT (Logic sheet 12). If set, the G0CABT pulse clears the following flip-flops: F1BCB1, F1BCB2, F1BDER, F1TPTY, F1TCON, F1CB05.

When issued to either channel, while the channel is busy an end-of-record interrupt is produced when the ABT command has set the channel not busy. The transmitter channel interrupt may be held up until the data set disconnects.

4820/4821X09
INPUT/OUTPUT TYPER DRIVE
TABLE OF CONTENTS

INPUT/OUTPUT TYPER DRIVE

The Input/Output Typer Drive controls, monitors and operates the Model 4270D Input/Output Typewriter, and synchronizes the transfer of data between the typer and the Arithmetic Unit. The typer prints or transmits characters on an incremental basis. However, the data is transferred between the typer and the Arithmetic Unit in a record format. That is, a complete block of data, consisting of several characters or words is transferred before the control and typer are released and made ready for the next record.

The Model 4270D typer is capable of printing data transferred from the Central Processor, or transmitting data from the keyboard to the Arithmetic Unit. When printing data from the Arithmetic Unit, the typer is operated through the output channel of the control module, in the same manner as the Model 4221D (output only) Typer is operated, via its output drive. When transmitting data from the keyboard to the Central Processor, data is transferred through the input channel of the control module.

This publication describes the theory of operation for the hardware associated with the following model numbers:

- 4DP4820AS09 - Input/Output Typer Drive for the I/O Buffer in a GE-PAC* 4010A System.
- 4DP4820BS09 - Input/Output Typer Drive for the Central Systems Unit (CSU) I/O Buffer in a GE-PAC 4010B System.
- 4DP4821AS09 - Input/Output Typer Drive for the Auxiliary Systems Unit (ASU) I/O Buffer (4823ASID) in a GE-PAC 4010B System.
- 4DP4821BS09 - Input/Output Typer Drive for the Central Systems Unit (CSU) Second I/O Buffer (4824ASID) in a GE-PAC 4010B System.

Detailed theory of operation for the output channel of the I/O Typer Drive is provided in publication 4820/4821X10-T. Logic for the I/O Typer Drive is provided in GE drawing, 70C180348.

INTERRUPTS

Four interrupts are produced by the control module. A data exchange interrupt is developed to signal the computer when the typer and the output channel of the control module are ready to accept an increment of data. An end-of-record interrupt is developed to signal the computer that all of the data in an output record has been transferred and typed. A data exchange interrupt is developed to inform the computer that the input channel buffer contains a data increment ready for transfer to the Central Processor. An end-of-record interrupt is developed to inform the computer that the last character in a record has been transmitted from the keyboard.

*Registered Trademark of General Electric Company

TRANSFER FORMAT

The data transferred through the output channel are the seven least significant digits of the A register when an OUT command is used, or from the B register when the TOM function is used. No parity bit is transferred through the output channel.

Data is transferred from the input channel to the seven least significant digits of the A register when an IN command is used, or to the seven least significant digits of the B register when a TIM function is used. A parity bit is transferred through the input channel to the parity checking circuit in the I/O Buffer.

RECORD FORMAT

The output format will contain the following data in the sequence shown:

Name	Octal Code	Function
OPR-DEL	177	
TEXT ↓ TEXT	TEXT ↓ TEXT	
CR	100	Increment Vertical Format
	120	End-of-Record

The input format will contain the following data in the sequence shown:

Name	Octal Code	Function
OPR-STX	171	Unlock keyboard, light Input lamp
TEXT ↓ TEXT	TEXT ↓ TEXT	
CR	100	End-of-Record

ALARMS

Properly encoded JNE instructions will detect the following alarm conditions:

Output Channel

Alarm Line 1 - Typer Off-Line

Alarm Line 2 - Not Used

Alarm Line 3 - Not Used

Alarm Line 4 - Not Used

Input Channel

Alarm Line 1 - Typer Off-Line

Alarm Line 2 - Not Used

Alarm Line 3 - Not Used

Alarm Line 4 - Parity or Timing Error

LOGIC DESCRIPTION

Both the input and output typer channels operate at a maximum rate of 15.5 characters per second. For an output operation, data are transferred through the drive and characters printed as a result of an OUT command, or a simulated OUT command due to a TOM function. For an input operation, data are transferred into the Arithmetic Unit by an IN command, or a simulated IN command due to a TIM function. The IN instructions are generated in response to interrupts generated, or tests of the data ready line (JDR), when the input channel data register contains a character, transmitted from the keyboard.

When a complete record has been transferred through the output channel, an octal 120 code is transferred by an OUT instruction. The 120_8 code is not printed, but rather causes the output channel to go "not busy", and be released, ready to accept the next operation.

When the RETURN key is depressed during an input operation, returning the carriage to the left hand margin, a 100_8 code is transferred through the input channel. This code causes the control module to generate an end-of-record interrupt, to go "not busy", and be released, ready for the next operation.

The OPR and OUT instructions are described earlier in this section. The 4270D I/O Typer is fully described elsewhere in this manual. The operation of the output channel is identical to that of the output drive for the Model 4221 Fixed Carriage Typer. For a discussion of the output channel refer to publication no. 4820/4821X10-T, which is included in this section. The discussion which follows pertains to the operation of the input channel. Reference is made to the block diagram Fig. 1, the flow chart Fig. 2, the timing diagram Fig. 4, and to the logic schematic, General Electric drawing 70C180348.

When following the Logic Description of the output channel in publication no. 4820/4821X10-T, reference should be made to sheets 15 through 19 of drawing 70C180348. These sheets depict the logic for the output channel, which is contained on a single PBCD1 printed wire board. This board is installed in the B panel of the computer cabinet.

The logic elements for the input channel are contained on a single PBCB1 printed wire board, which is installed adjacent to the PBCD1 board, in the B panel. The discussion which follows pertains to the logic for the input channel.

The operation of the input channel will be covered in the order in which a typical input operation would occur. This sequence is shown on the flow chart Fig. 2. The data ready, parity, and timing error tests are covered at the point in the input operation at which, if used, they would occur. The off-line and channel busy tests are covered near the end of the discussion, and finally the effect of the ABT command is discussed.

20 KHz Clock

A clock generator consisting of a free running bistable multivibrator (Q2 and Q3) and an inverter (Q4) is contained within circuit 2, Q1DCLK, as shown on the logic drawing. This non-adjustable circuit is designed to operate at approximately 20 KHz. The output of Q1DCLK is inverted by gate G0DCLK, and again by clock driver D1DCLK. The output of the driver is a positive pulse; one microsecond in duration. The clock pulses are approximately 50 microseconds apart. These pulses are used for timing within the input channel of the control module.

Input Demand

With the I/O Typewriter connected to the computer, and with its primary power connected and turned on, but prior to any enabling function, the keyboard will be locked. Output operations (printing) may take place if implemented by the program, but transmission to the Arithmetic Unit will not be possible.

The operator depresses the INPUT key on the keyboard to initiate an input operation. The simplified schematic of the input demand circuit, Fig. 3, illustrates the effect of depressing the key. Normally, the ENV line is returned to ground through a resistor network, while the DMD line is returned to a positive DC voltage through the normally closed contacts on the INPUT key. Depressing the key removes the positive voltage from the DMD line and applies a similar positive voltage to the ENV line. This arms flip-flop F1DENV to set at the next 20 KHz clock pulse. F1DENV will remain set until the first clock pulse after the key is released.

F1DENV, when set, arms F1DDEN to set, and when clear arms DEN to clear. F1DDEN therefore, sets one clock pulse after ENV sets and clears one clock pulse after ENV clears, as shown on the timing diagram, Fig. 4.

The input channel busy flip-flop F1CCBY, would normally be clear at this time, but under some circumstances, it may be set. If so, it is necessary to clear it. This is accomplished by a zero volt pulse from reset channel busy gate, G0DRCB, during interval when ENV is clear, while DEN is set. (DEN clears on the trailing edge of the clock pulse.)

$$G0DRCB = DEN \cdot \overline{ENV} \cdot CLK$$

When the input and output channels are not busy, channel ready interrupt gate, G0ACRI, is enabled.

$$G0ACRI = \overline{CBY}\ \text{(input ch.)} \cdot \overline{G0ADMD}$$

$$G0ADMD = DEN \cdot \overline{CB1}\ \text{(output ch.)}$$

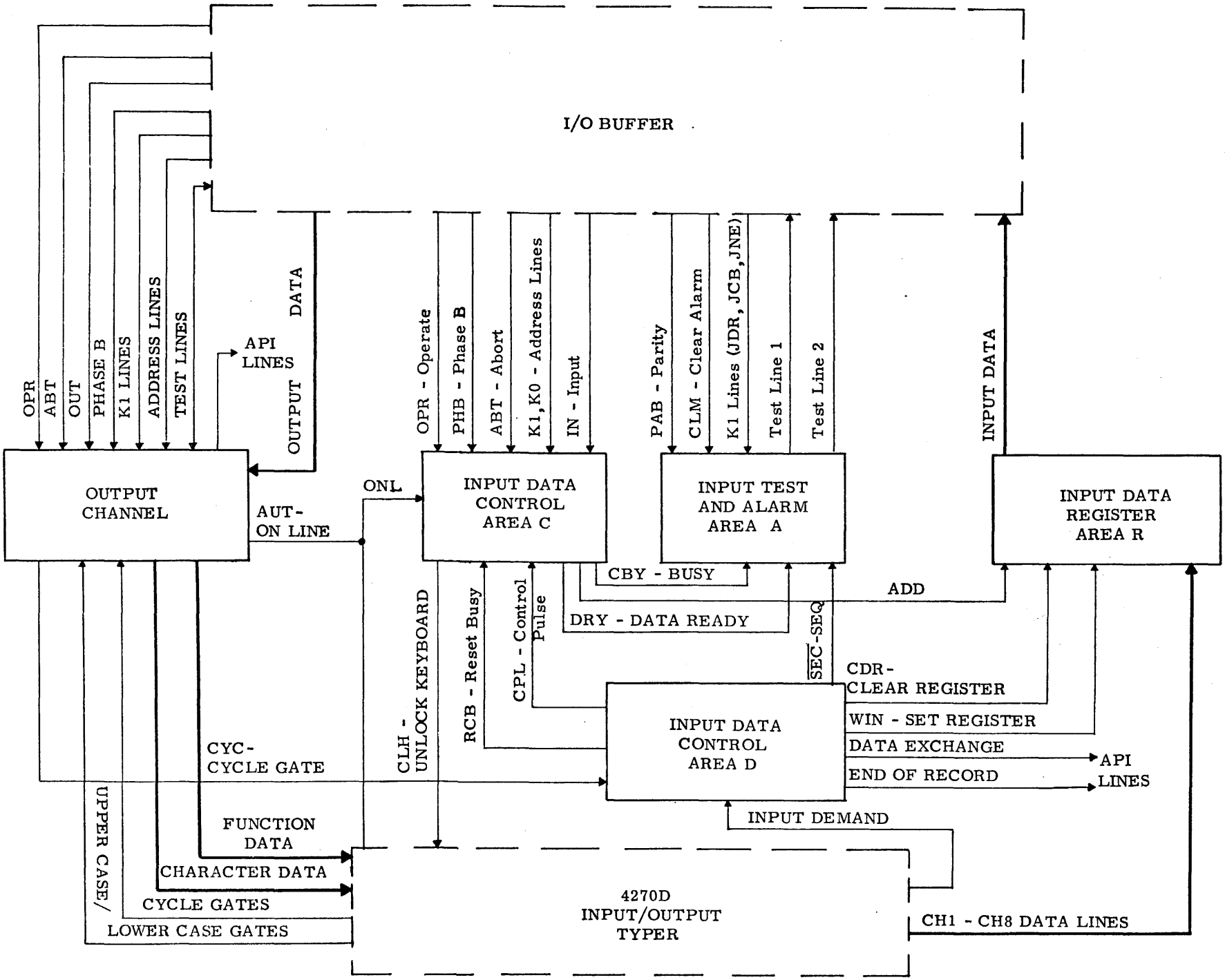


Fig. 1 Block Diagram, I/O Typer Control

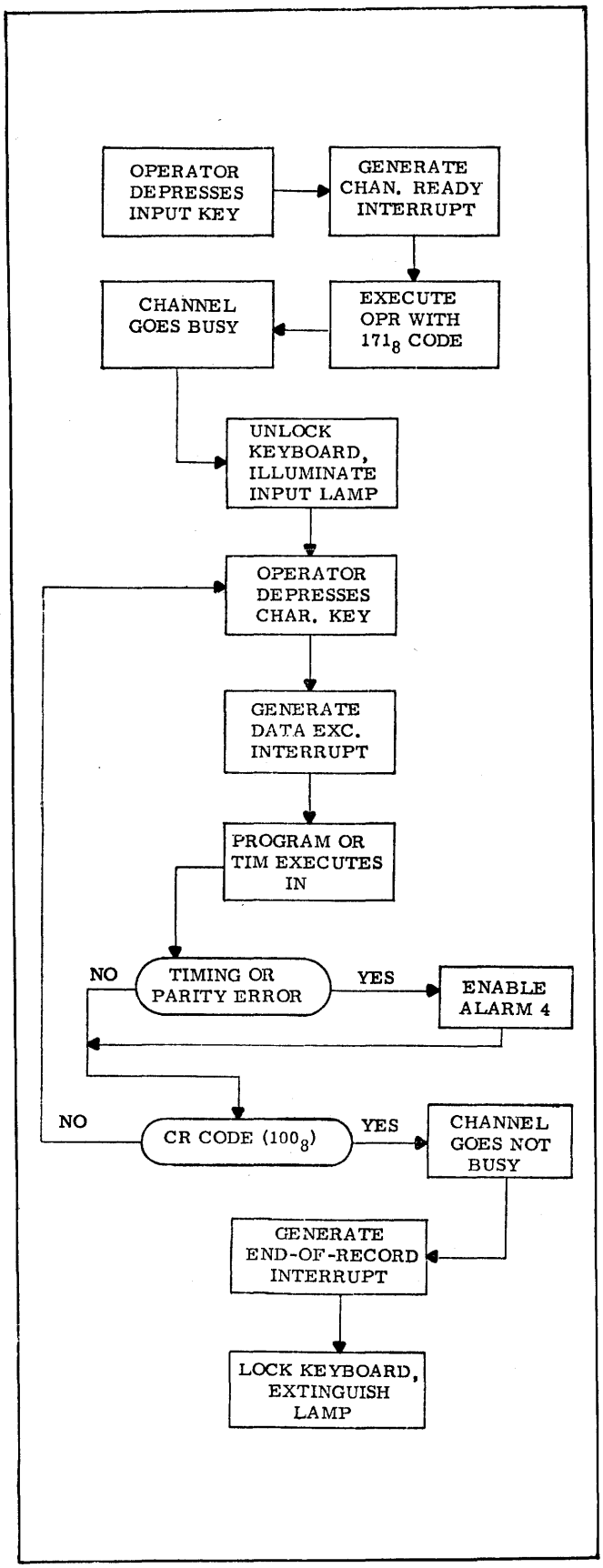


Fig. 2 Typer Input Flow Chart

To produce a negative going interrupt signal, CRI must first be disabled, and then again enabled. This occurs when the DEN gate is produced after the typer INPUT key is depressed. With the output channel not busy and the DEN gate present, G0ADMD is enabled, inverting the gate. With the input channel not busy, G0ACRI will again invert the gate. The negative going trailing edge of the positive gate at the output of CRI, is the channel ready interrupt signal.

Keyboard Enable

In response to the interrupt, the program executes an OPR command, normally with 171_8 in the A register. The A register contents however, are not used by typer nor the control.

A zero volt pulse from gate G0COPR sets channel busy flip-flop F1CCBY, at phase B of the OPR command.

$$G0COPR = OPR \cdot PHB \cdot ADD$$

ADD = This channel addressed.

Flip-flop F1CEOR, which was set through its DC set terminal when the channel was not busy, is cleared through its DC clear terminals by the zero volt pulse from G0COPR. Clutch flip-flop F1CCLH, is set when the channel goes busy. With EOR clear, and CLH set, relay driver P0CCLH is enabled.

$$P0CCLH = F1CCLH \cdot \overline{EOR}$$

The enabled relay driver provides a ground return for a relay in the typer which unlocks the keyboard. The driver also provides a ground return which lights a lamp within the INPUT key. The illuminated lamp informs the operator that the computer is ready to accept keyboard data.

Character Transmission

Synchronization of the control module with the typer operation is accomplished by means of cycling gates generated in the typer's cycling circuit, and connected to cycling flip-flops F1CCNO, and F1CCYC, in the output channel. A typer cycle synchronization gate from the zero side of F1CCYC is applied to sequence gate G0DSEQ. This gate is enabled during most of the typer's 64.5 millisecond cycle period, but when valid data is present on the eight data lines from the typer, G0DSEQ is disabled for several milliseconds.

The typer data lines, CH1 through CH8, may not pick up valid data immediately as the operator depresses a key. Appropriate contacts in the typer's input translator will be properly positioned however, providing valid data, by the time the cycle gate disables G0DSEQ. The positive gate from SEQ is inverted twice and arms F1DSEQ to set, and it sets at the next 20 KHz clock pulse.

Each stage of the data register is preset to zero, by a zero volt pulse from driver D0DCDR, after SEQ sets, and before F1DSEC sets.

$$D0DCDR = CLK \cdot SEQ \cdot \overline{SEC}$$

The gate from the zero side of SEQ is inverted by D1DWIN, and the resultant positive gate partially enables data input gates, G0RS00 through G0RSP7. Those input gates connected to data lines containing a logical one (positive voltage), will be enabled, causing the corresponding data register flip-flops to set. (In this application of the PBCB1 circuit board, data register stages F1B08 through B11 are not used.)

When SEQ sets, F1DSEC is armed to set, and it sets at the next clock pulse. When the typer cycle gate terminates, F1DSEQ is armed to clear, and it clears at the next clock pulse. With SEQ clear, F1DSEC is armed to clear, and it clears at the next clock pulse. The gates produced by SEQ and SEC are initiated by the typer cycle, are in synchronism with the 20 KHz clock, and SEC sets and clears one clock pulse period later than SEQ.

A control pulse used to trigger the data ready and end-of-record flip-flops is produced by gate G0DCPL and inverted by N1DCPL, during the interval when SEQ is clear and SEC is set.

$$G0DCPL = CLK \cdot SEC \cdot \overline{SEQ}$$

With EOR clear, the CPL pulse sets data ready flip-flop F1CDRY, through gate G0CDRE. The DRY gate is coupled to data exchange interrupt gate G0ADEI, through N1CDRY. With SEC clear, and DRY set, G0ADEI is enabled, producing the data exchange interrupt.

Data Ready Test

If the Automatic Program Interrupt system is not used, the Arithmetic Unit may test the data ready status by means of test line one, which is routed to the AU via the JNR circuit in the basic I/O Buffer. A JDR command addressed to the input channel will enable test line one gate, G0ATL1, if G0ADEI is not enabled, indicating that data is not ready. If data is ready, TL1 will be disabled.

$$G0ATL1 = ADD \cdot K14 \cdot \overline{DEI}$$

$$\overline{G0ATL1} = ADD + K14 + DEI$$

Character Input

In response to the data exchange interrupt, the program executes an IN command, or a simulated IN command is produced by the TIM function. Gate G0COIN is enabled at phase B of an IN instruction, addressed to the input channel.

$$G0COIN = ADD \cdot PHB \cdot IN$$

The zero volt pulse from OIN, clears DRY, which in turn, disables DEI, to await the next interrupt.

The address pulse, produced by G0CADD, and inverted by driver D1CADD, enables those data output gates which are connected to register flip-flops containing a one bit. Data is thereby transferred to the Arithmetic Unit, via the basic I/O Buffer.

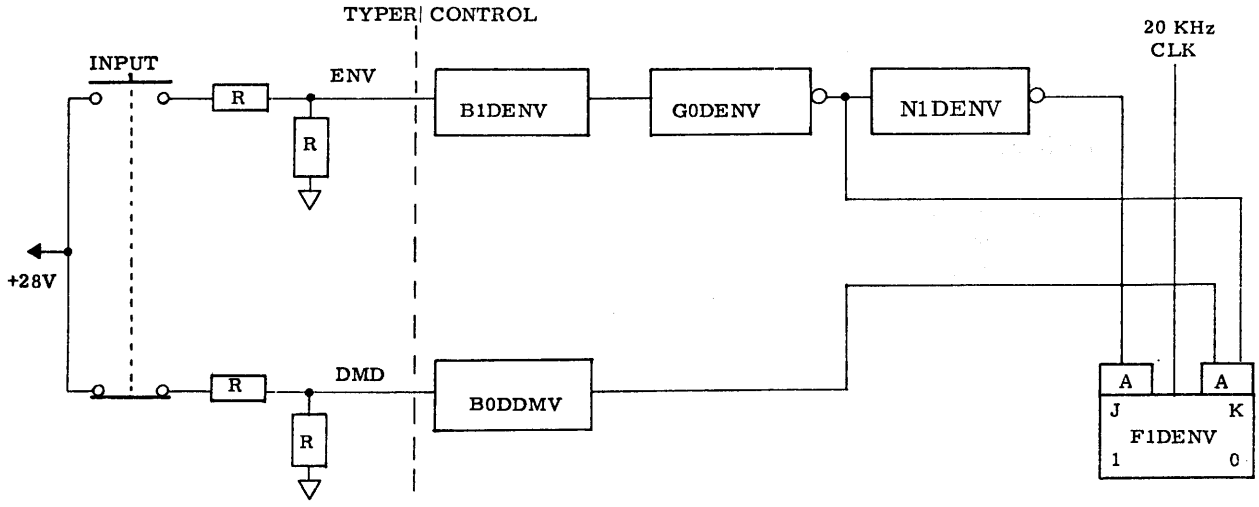


Fig. 3 Simplified Schematic, Input Demand Circuit

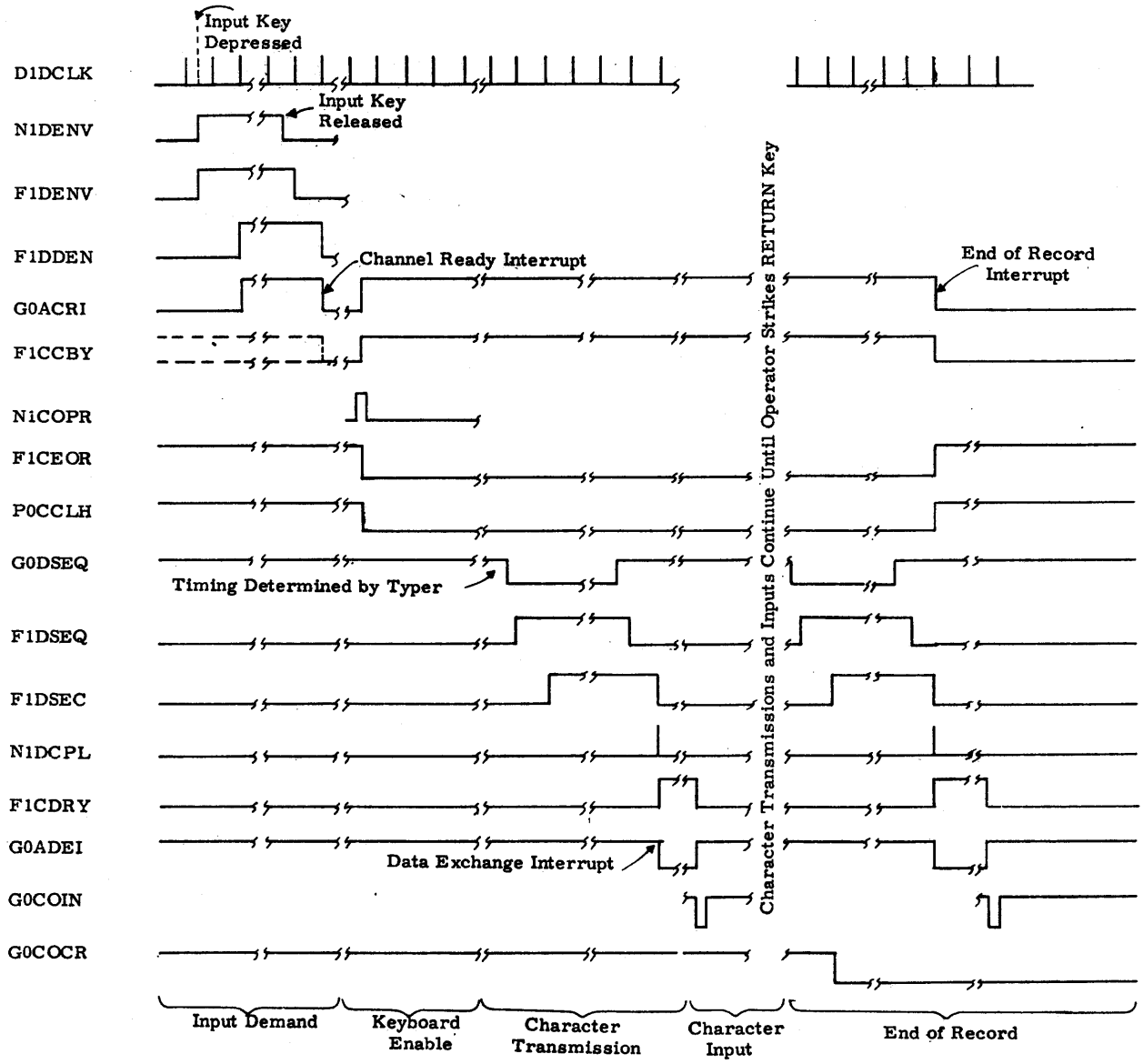


Fig. 4 Timing Diagram, Input Channel

Timing, Parity Error Tests

If with a character in the data register, an IN instruction is not executed before a new character is fed into the data register, DRY will remain set when the next typer cycle gate sets SEQ. This condition will enable gate, G0AALM.

$$G0AALM = \overline{SEC} \cdot SEQ \cdot DRY$$

When the typer is modified to operate with GE-PAC computers, circuitry is added to the typer's input translator to insure that with valid data, an even number of data lines will be true (positive voltage).

If a parity error occurs (odd number of bits set), the parity check circuit in the basic I/O Buffer will enable G0AALM, when the IN instruction is executed.

$$G0AALM = D0PPAB \cdot N1COIN$$

If G0AALM is enabled, flip-flop F1AALM is set, and will remain set until it clears at the beginning of another record by an OPR command, or until it is cleared by an ABT command, or by a clear alarm signal.

With F1AALM set, a JNE command encoded to test alarm line four, will enable test line two gate, G0ATL2.

$$G0ATL2 = K14 \cdot K12 \cdot ALM \cdot ADD$$

End-of-Record

Character transmission and inputs will continue until the operator depresses the RETURN key. The typer carriage is then returned to the left hand margin, and 100_8 is placed on the data lines. When this code is placed in the data register, gate G0COCR is enabled. The data exchange interrupt is developed and the IN instruction executed as before, placing the CR (100_8) character in the AU. In addition, the sequence of events which make the channel go not busy, and generate the end-of-record interrupt, begins.

The CR code is placed in the data register after the first clock pulse, during the SEQ gate. G0COCR is then enabled, and its output is inverted by N1COCR. The OCR signal and CBY arm the end-of-record flip-flop F1CEOR to set, and it is set by the CPL pulse. When EOR sets, the CLH relay driver is disabled, the keyboard is locked and the INPUT lamp goes out.

The OCR signal and the CPL pulse enable G0CEOR, and the zero volt pulse from that gate clears F1CCBY. When CBY clears, G0ACRI is enabled, producing the end-of-record interrupt.

Off Line Test

A +28V DC signal is connected from the typer, through the output channel data cable, to a test circuit in the output channel, and to termination circuit B1CONL, in the input channel. If the typer is on line, with its primary AC power turned on, a positive DC voltage from ONL will enable the CLH flip-flop to operate as described above, and will enable gate G1AHLT.

Should the typer go off-line, HLT will be disabled, producing +3.6V at its output. With HLT disabled, a JNE command addressed to the input channel, and encoded to test alarm line one, will enable test line two gate, G0ATL2.

$$G0ATL2 = HLT \cdot K12 \cdot K14 \cdot ADD$$

Channel Busy Test

Gate G0ACRI is enabled when there is an input demand (DEN) or when the input or output channels are busy. CRI is inverted by busy or demand gate, G1ABOD. A JCB command, addressed to the input channel, will enable test line one gate, G0ATL1, if there is an input demand, or if the input channel is busy and the output channel is not busy.

$$G0ATL1 = BOD \cdot \overline{CBY}\ (\text{output ch.}) \cdot K12 \cdot ADD$$

Abort

An ABT command, addressed to the input channel enables gate, G0CABT. If an input demand is occurring at that time, the demand flip-flops ENV and DEN are cleared by the zero volt ABT pulse, terminating the demand.

If the channel is busy, the ABT pulse clears F1CCBY, terminating the input operation.

If the alarm flip-flop, F1AALM, is set, the ABT pulse clears it.

4820/4821X10
FIXED CARRIAGE TYPER OUTPUT DRIVE

TABLE OF CONTENTS

FIXED CARRIAGE TYPER OUTPUT DRIVE

The Output Typer Drive decodes and transfers data from the I/O Buffer to a Model 4221D or 4270D Fixed Carriage Typer, and synchronizes the transfer of data from the Arithmetic Unit through the I/O Buffer and to the typer, with the operation of the typer. Although the fixed carriage typer prints characters on a incremental basis, the data is processed into the typer in a record format. That is, a complete message or block of data is typed before the output drive and typer are released and made ready for the next record.

This publication describes the theory of operation for the hardware associated with the following model numbers:

- 4DP4820AS10 - Output Typer Drive for the I/O Buffer in a GE-PAC* 4010A System.
- 4DP4820BS10 - Output Typer Drive for the Central Systems Unit (CSU) I/O Buffer in a GE-PAC 4010B System.
- 4DP4821AS10 - Output Typer Drive for the Auxiliary Systems Unit (ASU) I/O Buffer (4823ASID) in a GE-PAC 4010B System.
- 4DP4821BS10 - Output Typer Drive for the Second Central System Unit (CSU) I/O Buffer (4824ASID) in a GE-PAC 4010B System.
- 4DP4820AS09 - Output Channel** of this I/O Typer Drive in a GE-PAC 4010A System.
- 4DP4820BS09 - Output channel** of this I/O Typer Drive for the CSU I/O Buffer in a GE-PAC 4010B System.
- 4DP4821AS09 - Output channel** of this I/O Typer Drive for the ASU I/O Buffer (4823ASID) in a GE-PAC 4010B System.
- 4DP4821BS09 - Output channel** of this I/O Typer Drive for the CSU Second I/O Buffer (4824ASID) in a GE-PAC 4010B System.

Logic for the Output Typer Drive is provided in GE drawing, 70C180345. Logic for the output channel of the I/O Typer Drive is provided on sheets 12-19 of 70C180348.

*Registered Trademark of General Electric Company

**The input channel for the I/O Typer Drive is described in the 4820/4821X09-T publication.

INTERRUPTS

Two interrupts are produced by the output drive. A data exchange interrupt is developed to signal the computer when the typer and drive are ready to accept the next increment of data. An end-of-record interrupt is developed to inform the computer that all of the data in a record has been transferred and typed. The data exchange interrupt is normally noninhibitable and the end-of-record interrupt is normally inhibitable.

TRANSFER FORMAT

The data transferred from the processor to the output drive are the seven least significant digits of the A register in the case where an OUT command is used. When the TOM function is used, the seven least significant digits of the B register are transferred. No parity bit is transferred.

RECORD FORMAT

The output format will contain the following data in the sequence shown.

Name	Octal Code	Function
OPR-DEL	177	
TEXT ↓ TEXT	TEXT ↓ TEXT	
CR	100	Increment Vertical Format
	120	End of Record

ALARM

A properly encoded JNE instruction will detect off-line condition. Alarms 2, 3, and 4 are not used. Alarm 1 - Device off line, will be detected if the typer is turned off or is disconnected from the driver.

LOGIC DESCRIPTION

The typer operates at a maximum rate of 15.5 characters per second. The maximum operating rate using API interrupts is 15 characters per second. Data are transferred through the drive, and characters printed by the typer as a result of the OUT command, or a simulated OUT command due to the TOM function. After the drive and typer are enabled by an OPR command, each OUT command causes the transfer of a single character.

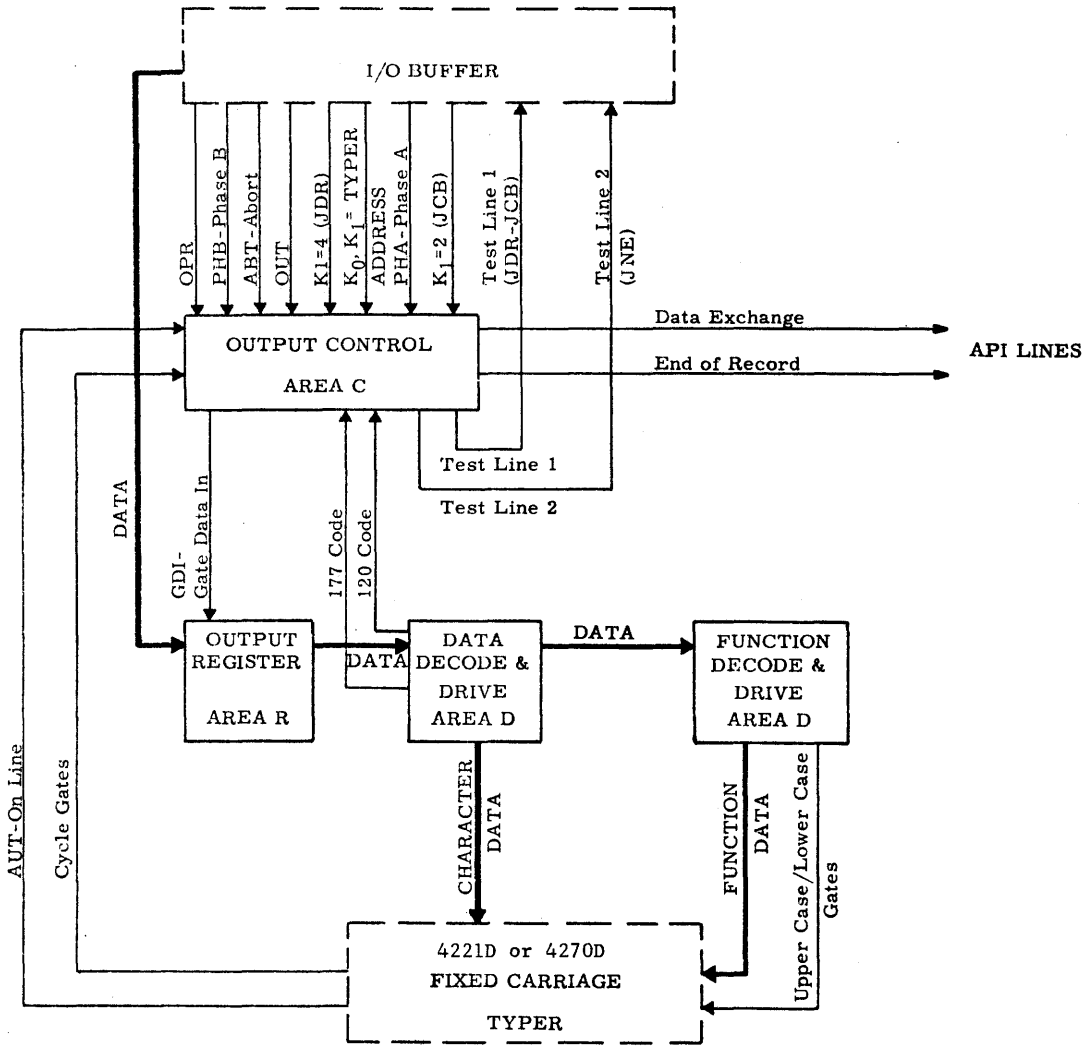


Fig. 1 Block Diagram, Typer Output Drive

When a complete record has been transferred, an octal 120 code is transferred by an OUT command. The 120_8 code is not printed, but rather causes the drive and typer to go "not busy", and be released, ready to accept the next record.

The OUT and OPR commands are described earlier in this section. The typewriters are fully described elsewhere in this manual. The following text is devoted to the logic and timing of the typer output drive. Reference is made to the block diagram of Fig. 1, the flow chart Fig. 2, the timing diagram Fig. 3, and the logic schematic, General Electric Drawing 70C180345 or sheets 12-19 of 70C180348.

The typical sequence of events in the operation of the typer output is shown in the flow chart Fig. FC TYP. 2. The following paragraphs discuss the logic in the order of events shown on the flow chart. (The JDR instruction is discussed under "Channel Busy Test" and the discussion of the ABT instruction follows "Media Disable").

Off-Line Test

The off-line test is made by means of a JNE command. The output of G0CTL2, test line two, is monitored in the AU, via the I/O Buffer. A disconnected or off-line typer disables gate N0CAUT. With N0CAUT disabled, and a properly encoded JNE command present, G0CTL2 is enabled:

$$G0CTL2 = ADB \cdot \overline{K14} \cdot \overline{K12} \cdot \overline{AUT}$$

ADB = this channel addressed

Channel Busy Test

When a properly encoded JCB command is present, the output of G0CTL1, test line one, is monitored in the AU, via the I/O Buffer. Test line one is also monitored when a JDR command is executed. In the case of the JCB command, the condition of G0CTL1 is determined by the busy flip-flop F1CCBY:

$$G0CTL1 = ADA \cdot \overline{CBY} \cdot K12$$

ADA = this channel addressed

The JDR command is used to test the readiness of the channel to accept a new increment of data when API interrupts are not used. In this case the condition of the data exchange interrupt gate, G0CDEI, determines the condition of test line one:

$$G0CTL1 = \overline{DEI} \cdot K14 \cdot ADA$$

Media-Enable

Execution of the OPR instruction causes the transfer of octal code 177 to the I/O Buffer, however, this data is not gated into the typer drive, and no character is printed.

The channel busy flip-flop, F1CCBY, is set a phase B of the OPR command. At the trailing edge of Phase B, a data exchange interrupt is generated by gate G0CDEI:

$$F1CCBY = G0COPR$$

$$G0COPR = OPR \cdot PHB \cdot ADA$$

$$G0CDEI = CBY \cdot \overline{OUT} \cdot \overline{OPR} \cdot \overline{F1CDEI}$$

Data-Enable

The OUT instruction gates data into the output register and sets in motion the chain of events which cause the typer to print a character or execute a function called for by the data. At the completion of the typer operation, another data exchange interrupt is generated.

Seven data bits from the I/O Buffer are gated into the output register when the data input gate, N0CGDI, is enabled. This gate is enabled and its output inverted at phase A of the OUT command. The presence of the resulting positive pulse at the input gates of the output register allows the data flip-flops, F1RD00 through F1RD06, to be set, or cleared, according to the data.

$$N0CGDI = OUT \cdot ADB \cdot PHA$$

The data exchange interrupt flip-flop, F1CDEI, and the enable data out flip-flop, F1CEDO are set at phase B of the OUT command, provided the output register does not contain 177_8, nor 120_8. These flip-flops are set by gate G1CEDI.

$$G1CEDI = OUT \cdot \overline{177} \cdot \overline{DC4}$$

Appropriate data and function relay drivers are enabled by the data in the output register and the enable data out gate, D0CEDO. The relay drivers enable solenoids in the typer which cause the typer to execute the functions or to type the characters.

$$D0CEDO = AUT \cdot EDO$$

The function solenoid drivers (red, black, tab, and carriage return) are enabled when called for by the data and when +3.6V is present at the output of G1DEFS (data bit 6 is set for all function codes):

$$G1DEFS = G0DB06 = D06 \cdot EDO$$

The data solenoid drivers, which cause characters to be printed, are enabled when called for by the data and when +3.6V is present at the output of G1DEDS:

$$G1DEDS = G0DEDS = EDO \cdot \overline{CCH} \cdot \overline{020} \cdot \overline{D06}$$

The space solenoid driver, P0DSSP, is enabled when data octal code 020 is detected by G0D020, and when +3.6V is present at the output of G1DEDO.

$$N1DSSP = G0DSSP = EDO \cdot SSP$$

$$G1DSSP = G0D020$$

Function solenoid drivers, P0DSLC (lower case), and P0DSUC (upper case), are enabled as required by the data, and according to which solenoid was last energized. Upper case gate, G1DUPC, produces a +3.6V output when the data corresponds to a special character (such as @, ?, %, &), which requires that the typer go to the upper case mode. If the typer was

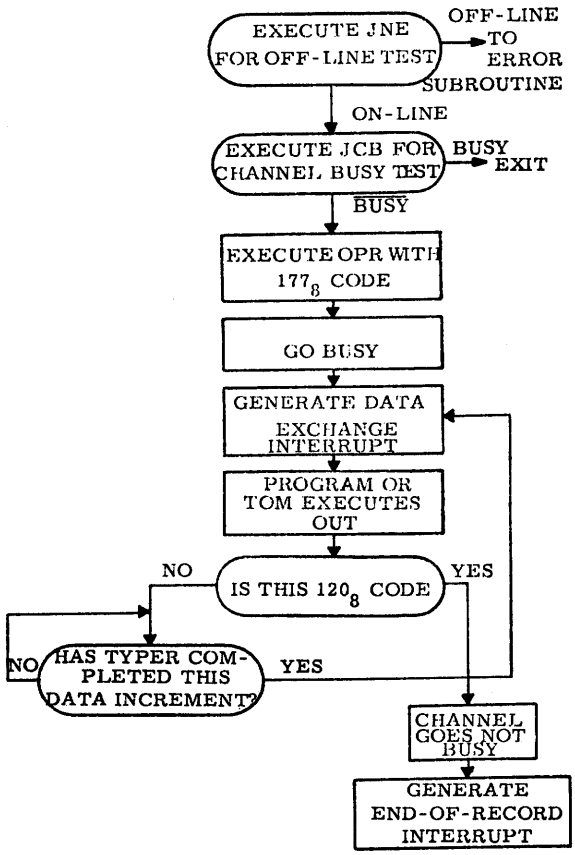


Fig. 2 Typer Output Drive Flow Chart

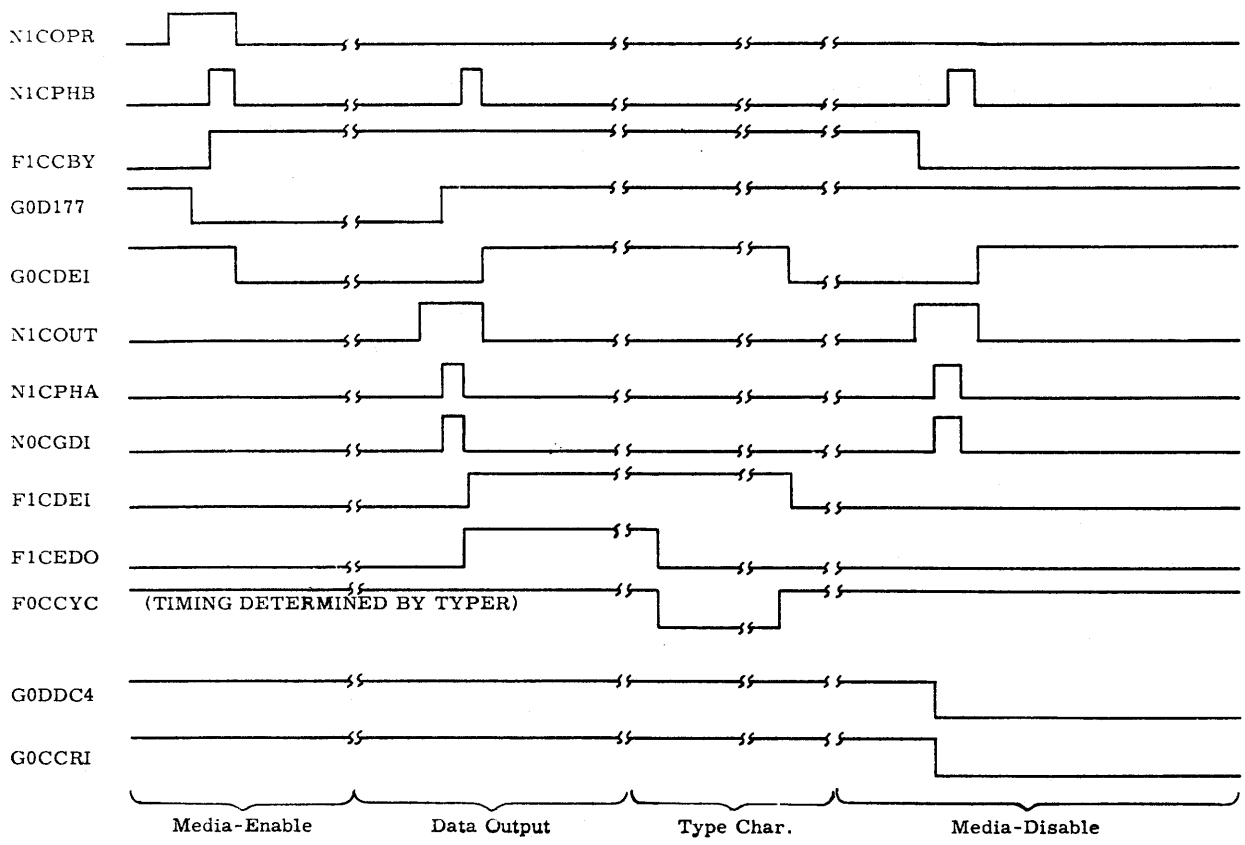


Fig. 3 Timing Diagram

already in the upper case mode, the case change gate, G0DCCH, will not be enabled. G0DCCH will be enabled if the typer is in a mode other than the one called for by G1DUPC. The case change gate is combined with the data output enabling signals and routed to the enable case solenoid gate, N1DECS. The output of N1DECS is applied to both case solenoid drivers. With the typer in one case, and +3.6V present at the output of N1DECS, the solenoid driver of the opposite case will be energized, and the typer case mode changed. Gates M1DUPC and M1DLOC convey the status of the typer case mode to the case mode solenoid drivers.

Synchronization of the data transfer from the computer to the operating cycle of the typer is accomplished by means of the cycle flip-flops, F1CCNO and F1CCYC. Flip-flop F1CCNO is clear throughout most of the typer cycle period, due to a positive voltage from the typer's normally closed contacts, applied through B1CCNC. The outputs from F1CCNO hold flip-flop F1CCYC in the clear state also. When the typer has reacted to a data increment, a positive voltage from its normally open contacts will be applied through B1CCNO, and both flip-flops will be momentarily set. The resulting zero volt signal at the output of F0CCYC clears both the data output flip-flop, F1CEDO, and the data exchange interrupt flip-flop, F1CDEI. When F1CDEI goes clear, a data exchange interrupt is produced at the output of gate G0CDEI.

Should F1CEDO not be cleared by the typer cycle within 200 milliseconds of the time when it was set by the OUT command, it will be cleared by G0CRDO. Gate G0CRDO will be enabled approximately 200 milliseconds after the flip-flop sets, by the single-shot circuit Y1CEDO. This prohibits the transfer of data to the typer if it has not completed the cycle within that time period.

Media-Disable

When an OUT instruction causes octal code 120 to be gated into the output register, gate G0DDC4, is enabled. The zero volt output of this gate is applied to G1CEDI, where it prohibits the setting of the data output flip-flop F1CEDO, and the data exchange interrupt flip-flop, F1CDEI. The output of gate, G1DDC4, is applied to the channel busy flip-flop F1CCBY, causing it to clear. When the busy flip-flop clears, gate G0CCRI is enabled, producing an end-of-record interrupt at its output.

Abort

The execution of an ABT command enables gate G0CABT. The output of this gate clears the channel busy flip-flop, F1CCBY, and the data exchange interrupt, F1CDEI.

READER COMMENTS

The General Electric Company solicits your comments on publications covering Process Computer equipment. Please explain any "no" responses in the COMMENTS section. Your comments and suggestions become the property of the General Electric Company.

- Name of Manual: _____
- What is your computer application: _____

  _____
- How is this publication used:

  Familiarization ☐ Reference ☐

  Training ☐ Maintenance ☐

  Other (Explain) _____

	YES	NO
• Does this publication meet your requirements	☐	☐
• Is the material:		
1) Presented in clear text	☐	☐
2) Conveniently organized	☐	☐
3) Adequately detailed	☐	☐
4) Adequately illustrated	☐	☐
5) Presented at appropriate technical level	☐	☐

- Please provide specific text references (page number, line, etc.) with your comments.

NAME _____ DATE _____

TITLE _____

COMPANY NAME _____

AND ADDRESS _____

COMMENTS:

No postage necessary if mailed in the U. S. A.

Staple

Communications concerning Technical Publications should be directed to:

Manager, Technical Publications
Utility and Process Automation Systems Operation
2255 West Desert Cove Road
Phoenix, Arizona 85029

Fold Fold

FIRST CLASS
Permit No. 4091
Phoenix, Arizona

BUSINESS REPLY MAIL
NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY...

GENERAL ELECTRIC COMPANY
UTILITY and PROCESS AUTOMATION
SYSTEMS OPERATION
2255 West Desert Cove Road
Phoenix, Arizona 85029

Attention: Technical Publications

Fold Fold

Cut Along Line

Additional Comments:

PAPER TAPE READER 4212D/4213D

TABLE OF CONTENTS

INTRODUCTION

The information in this section applies to the GE-PAC* Model 4DP4212D/4213D Paper Tape Reader. The unit consists of a Digitronics Model 2500 Perforated Tape Reader Chassis mounted in a metal desk-top cabinet. The cabinet contains a cooling fan (in addition to the Digitronics fan) and a PDTA board. The optional tape handler, including a supply reel and take-up spooler, is mounted directly above the reader assembly as shown in Fig. INT. 1.

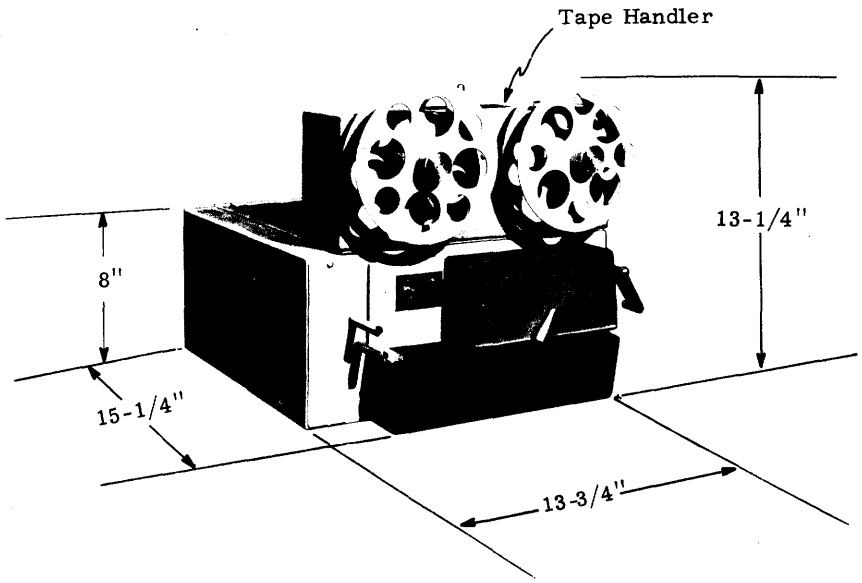
GENERAL

The Paper Tape Reader is a Process Computer System on-line input device which photo-electrically converts data punched into tape into electrical signals representing the punched data in a form which is acceptable by the I/O Buffer. The following is a summary of specifications:

• Input Speed	100 CPS(2), 200 CPS (3)
• Input Media	Paper, Aluminum, or plastic tape
• Tape dimentions	1 inch/. 005 inch
• Frames/inch	10
• Channels/frame	7 Data, 1 Parity
• Read Mode	Character Serial
• Interrupts	Data Ready & End of Record
• Checking	Parity/Timing/Off-line
• Reel Capacity	300 feet
• Rewind Speed	100 IPS
• Environment	Class A - General Industrial
• Power Requirement	115V AC ±10%, 50 or 60 Hz

REFERENCE DRAWINGS

70C179987	Reader Logic
70D195093	4212D Reader Assembly
70D195094	4213D Reader Assembly
70C180018	Tape Handler Assembly



Weight is 43 lbs.

Fig. INT. 1 Reader Dimensions

*Registered Trademark of General Electric Company

OPERATION

LOADING TAPE

Fig. OP. 1 shows the correct tape routing for a Paper Tape Reader connected to the usual overhead tape handler. The tape must be positioned with the sprocket hole in the fourth position away from the front panel of the reader. If this positioning reverses the indicated exit path from the Feed reel (i. e. , counter-clockwise instead of clockwise), the tape is spooled backwards and must be rewound before proceeding.

Rotate the Load switch fully clockwise to release the brake and turn the tape spooler OFF. Pull enough tape off the Feed reel to reach entirely across the reader and up to the Take-up reel (about 2 feet); replace the cover on the Feed spool to prevent further tape uncoiling during the load process. Beginning at the right-hand side of the reader, push the tape horizontally under the spring guide, over the tape sensing arm, between the pressure plates of the brake, under the Load switch and into the gap between the pinch roller and the capstan. Loop the tape over the left-hand guide bars as shown by Fig. OP. 1, place the end of the tape in the slot of the Take-up reel and manually wrap the excess tape on the reel.

With the tape spooler still OFF, manually position the first character to be read over the photo-diode lenses. Close the Load switch, pushing or pulling on the handle to align the guide arm with the outside edge of the tape. Turn the tape spooler and reader power switches ON. The Paper Tape Reader is now ready for operation.

REMOVING TAPE

To remove tape from the reader simply turn the spooler OFF, turn the Load switch clockwise, and pull the tape horizontally from the reader. It helps to leave the covers on the spooler reels to prevent the tape from uncoiling. The Load switch may be left in LOAD or RUN, as desired.

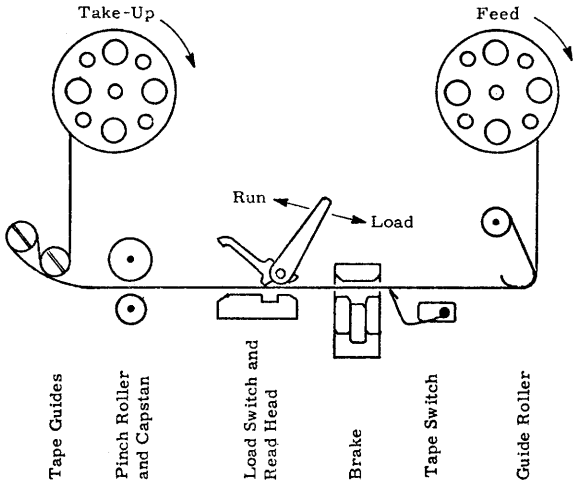


Fig. OP. 1. Tape Routing

TAPE SPLICING

Due to the photo-electric nature of the Paper Tape Reader, a broken paper tape may be spliced simply by wrapping the break with transparent tape (e. g. , Scotch brand "Magic Mending Tape" or equivalent); this method does not apply to oiled paper tape, since transparent tape will not adhere to the slick surface. If the light source is weak or improperly adjusted, it may be necessary to punch out the sprocket and data holes under the splice with a sharp, pointed instrument. Any gaps caused by missing fragments of tape may be eliminated by "painting" the area with dark ink (not pencil); this procedure also works very well on areas of tape which are thin enough to be translucent.

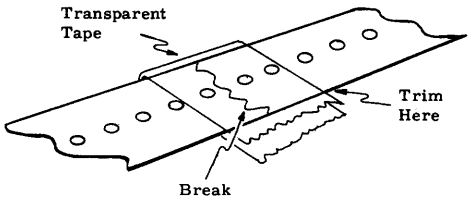


Fig. OP. 2. Splicing Paper Tape Breaks

PARITY PROBLEMS

The computer program normally monitors input parity on reader data and disables its "read" routine if an error is detected. When a "parity halt" occurs, the error character should be one position to the left of the photo-diodes; remove the tape and check for the following:

- Tape guide improperly adjusted.
- Dust accumulations on the photo-diode lenses.
- Torn tape.
- Translucent spots (thin tape, pin-holes, or oil spots on tape).
- Imperfect punch (chad still in hole).
- Incorrect punch (even parity).
- Poor registration.

Translucent spots are most often a problem on light-colored paper tape; only dark-colored paper tape is recommended for use in the Paper Tape Reader. For one cure for any existing translucent areas, refer to "Tape Splicing" above.

If none of these checks point out the problem area, record the binary configuration of the error character in your log and continue with the program. Records such as this, gathered over a period of time, can be invaluable in the eventual maintenance effort which corrects the problem.

THEORY OF OPERATION

As modified for use in General Electric Process Computer Systems, the Paper Tape Reader will operate either as a free-running device or in a single-character stepping mode. In the latter case, an automatic stop is initiated each time a sproket hole is sensed and the STEP (run) signal is no longer present. Free-running operation is achieved by continuously applying the STEP signal. The maximum reading rate is determined by the slew speed of the tape which, in turn, is fixed by the capstan speed reduction assembly used by this particular unit.

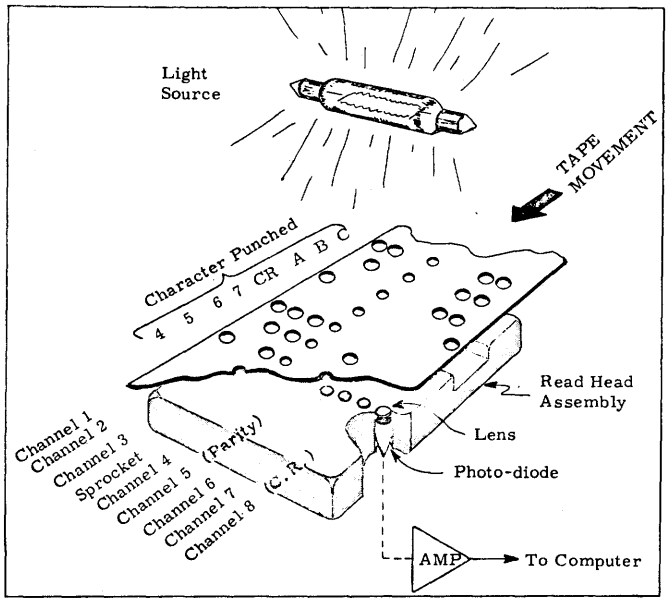
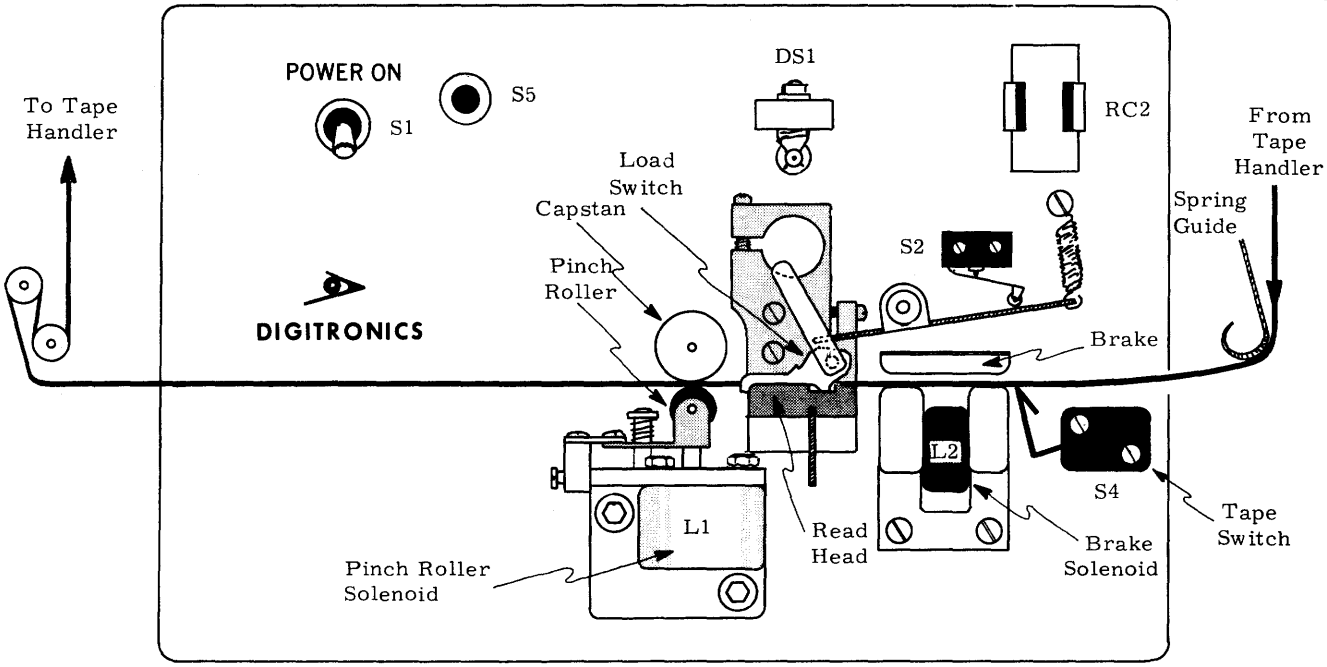


Fig. THEORY. 1. Simplified View of Read Station

As shown by Fig. THEORY. 1, the read function is implemented by passing the punched paper tape between a light source and a row of photo-diodes so that light falls on the photo-diodes only if a hole is present. The resultant signal from the photo-diode is amplified and sent to the controlling device as one of eight data signals which make up a particular character. The sprocket hole is sensed in an identical manner, but is used only for timing and control. The photo-diodes are permanently mounted in the read-head assembly, which also includes a small focusing lens for each diode.

Fig. THEORY. 2, shows a view of the front panel component arrangement. Tape movement is implemented by a pinch roller which presses the tape against a constantly-rotating capstan. Tape movement is stopped by a brake mechanism which clamps the tape between two pressure plates. The pinch roller and brake are controlled by the opposite outputs of a flip-flop in the reader electronics, thus assuring that the two solenoids cannot be energized at the same time.

In addition to the power On-Off switch, two control elements are mounted on the front panel of the reader. The first of these, a sensitive snap-action switch designated Tape Switch (S4), is fitted with a small wire sensing arm to detect the presence of tape in the reader. This switch disables the pinch roller and energizes the brake when the reader is out of tape. The second control



NOTE: Pinch Roller shown energized and Brake de-energized.

Fig. THEORY. 2. Front Panel Arrangement

element, designated Load Switch, is made up of a rotary tape guide assembly that is mechanically coupled to a two-gang, roller-plunger actuated, snap-action switch (S2A and S2B). The switch handle can be adjusted in and out so that the tape guide can accommodate 5, 6, 7, or 8-channel paper tape. The switch can also be rotated to the Load (manual) and Run (automatic) positions. When the reader is on and the Load Switch is in the Load Position (rotated clockwise), both the brake and pinch roller solenoids are disabled, allowing insertion or manual movement of the paper tape. Normal operation is implemented by placing the Load Switch in its Run position with the reader on.

A fourth switch, located on the front panel, is a momentary pushbutton type designated READ (S5). It is actuated by the operator to demand a channel ready interrupt so that the data on the paper tape can be read into the computer as soon as the reader has been loaded. The switch is rendered inoperative if the reader power is off or if the reader is out of tape.

TAPE CONTROL

On/Off Sequencing

In Model 4212D/4213D Readers, the brake solenoid is not energized when reader power is off or when the tape switch (S4) is open and the Load Switch is in run. Inadvertent tape movement is prevented when reader power is initially turned on by virtue of the Run/Brake flip-flop clearing, thus energizing the Brake, before the slower acting relay, K1, fully energizes and signals the computer that reader power is no longer off.

S2B informs the computer that the reader is On-Line when the Load Switch is in Run position and that the reader is Off-Line when the Load Switch is in the Load position.

No-Tape Halt

When the input tape supply is exhausted, the reader locks itself in a "not-ready" state awaiting manual intervention. The Tape Switch (S4), located just ahead of the Brake assembly, is helt closed by the presence of tape in the reader. When this switch opens, K1A000 is de-energized, informing the computer of the "not-ready" condition and disabling the Read and Load switches.

Loading Tape

When the Load Switch is rotated fully clockwise, the Brake and Pinch Roller are both released to allow insertion or removal of tape. S2A opens the -15S line, thus removing power from the Brake and Pinch Roller solenoids and de-enertizing K1A000. Since the -15S line is already open, the function of K1A000 is limited to informing the I/O Buffer of the "not-ready" condition (See Fig. THEORY.7). When the Load Switch is rotated counter-clockwise to the "Run" position control of the Brake and Pinch Roller is returned to the Run/Brake flip-flop

Normal Operation

Under normal operating conditions, the reader must move one character past the read head for every computer "read" command issued. The read cycle begins and ends with a character positioned directly over the photo-diodes; the character read is the one initially positioned over the photo-diodes.

Referring to Fig. THEORY.6 and THEORY.7 at the end of this section, the Run/Brake flip-flop is initially in the reset state, holding the Brake energized and the Pinch Roller de-energized. The execution of a computer IN command, in conjunction with the proper device address, samples the character currently in the read station and simultaneously issues the STEP signal.

STEP is routed through the PDTA1 board to the Set input of the Run/Brake flip-flop (8). The Brake is released and, shortly thereafter, the Pinch Roller presses the tape against the rotating capstan (10). The tape is accelerated to slew speed and held there until the next sprocket hole is sensed. At this time, the negative-going PSP signal (5) is coupled through C2 to clear the Run/Brake flip-flop, (9). The Pinch Roller is then released and the Brake energized, stopping the tape with the next character centered over the photo-diodes.

Thus the reader steps through the tape one character at a time, under control of the IN command. If "continuous" reading is desired, the STEP signal must be received within about 0.5 msecs. after sensing the sprocket hole. PSP would be true (-6V) during sprocket hole time while the tape is being slewed. The signal, applied to C2, produces only a negative spike at the Clear input of the Run/Brake flip-flop. Due to the presence of the STEP signal at the Set side, the flip-flop is prevented from changing state. The Clear side goes to zero volts only during the time of the negative spike, which is of insufficient duration to energize the brake solenoid.

CIRCUIT OPERATION

Three printed circuit boards are located inside the reader assembly. A standard General Electric circuit board (PDTA1) implements the interface between GE/PAC and Paper Tape Reader circuitry. Two Paper Tape Reader boards (LAC and MPC) implement the basic read and tape control functions. Refer to Fig. THEORY.7 throughout the following discussion.

PDTA1 Board

This board, shown by Fig. THEORY.3, contains the following circuits:

- Sprocket Amplifier (Q1-2): This two-stage amplifier converts the PSP (sprocket) signal from the LAC board to a voltage system compatible with the Peripheral Buffer input circuitry. A "1" (hole) input, nominally -7V (5), produces a 0V output (6). A "0" input, nominally 0V, produces a +28V output.

- Data Amplifiers (Q3): Eight identical single-stage smplifiers convert the Paper Tape Reader data signals to a voltage system compatible with the Peripheral Buffer input circuitry. A "1" (hole) input from the LAC data amplifiers, nominally -7V ⑦ , produces a +28V output. A "0" input (0V) produces a 0V output.

- -15V Sensing Circuit (Q4-5): This circuit provides +28V at 150 ma. (max.) to an external relay whenever the MPC -15V supply is more negative than -10V. This function is required during a reader power on sequence and as a power failure protection device.

- STEP Amplifier (Q6): This circuit converts the Peripheral Buffer STEP signal to a level acceptable to the MPC Run/Brake flip-flop. When STEP is false (+28V), the output is nominally +7V. When STEP is true (0V), the output is nominally -8V. The voltage divider in the Q6 base circuit, in conjunction with R8 and CR13, completely disables the STEP output if the +28V supply falls below +22V or, during a power on sequence, until this supply reaches +22V.

- Miscellaneous Components: CR10 and CR11 are used to clamp the Brake Solenoid to -15V and 0V, respectively. C2 serves as a transient filter for the +28V bus.

The PDTA1 board is located behind a removable cover at the left-hand side of the reader assembly. For further details, refer to the circuit drawing, IC3600PDTA1.

LAC Board

The LAC (Logic Amplifier Card) circuit board is located at the right-hand side of the reader assembly. This board contains the amplifiers which monitor the sprocket and data channels of the paper tape. Refer

To Fig. THEORY.4 for the LAC schematic and to Figs. THEORY.6 and THEORY.7 for interconnection and waveform information.

The Sprocket amplifier is made up of three circuits: a biased amplifier (T1-2), a Schmitt Trigger (T3-4), and an output inverter (T5). The input to T1 is taken directly from the Sprocket photo-diode. At the beginning of a read cycle, when the sprocket hole is positioned over the photo-diode, light breaks down the diodes back-resistance, coupling +4V to the base of T1; R23 determines the amount of bias across the photo-diode and thus its conduction point. The output of the photo-diode is shaped and inverted by T1-2, applying a negative signal to the Schmitt Trigger circuit ③ .

In the quiescent state, T3 is off and T4 is on, holding the output (pin S) at about -6V through voltage dividers R10 and R11. When the sprocket hole is sensed, the negative output of the sprocket amplifier begins to turn T3 on. The resultant positive swing at the collector of T3 is coupled through C3 to the base of T4; conduction in T4 decreases, decreasing the current in the common emitter-resistance R11 and thus raising the emitter potential of both T3 and T4. T3 is biased further on, and the increased positive swing at its collector is coupled through C3 to turn T4 further off; thus the circuit "locks-in" on the sprocket pulse, producing a fast rise-time output pulse ④ . This pulse is inverted by T5, producing the PSP output signal (pin V) ⑤ .

Eight identical circuits shape and amplify the channel 1-8 data signals for transmission to the PDTA1 board. Each amplifier is composed of an emitter follower stage (T1)and two series inverters (T2-3). Inputs are accepted directly from the photo-diodes ② ; a 200K trimpot (R23) varies the reverse bias on each photo-diode and thus its conduction point. The output of each stage, when a hole occurs in the associated tape channel, is a negative pulse ⑦ .

On Fig. THEORY.4, notice the diode D5 in the base circuit of the second stage (T2) of each data amplifier. This diode, in conjunction with the PSPG circuitry (T6), makes up a standard Paper Tape Reader hardware option known as "gated" operation; the philosophy is simply that the sproket hole, which lies exactly in the center of each line of data holes, marks the best time to sample data. To implement this option, the "G1" output of the Sprocket amplifier (pin S) is jumpered externally to the PSPG input (pin T). PSPG then holds the anode of D5 at +4V except when the sprocket hole is present, preventing the conduction of T2 and thus disabling the amplifier.

When the sprocket is sensed, PSPG swings negative to about -7V, reverse-biasing D5 and allowing signals to pass through the amplifier.

This option is not used on the GE/PAC version of the Paper Tape Reader or, rather, is implemented in an entirely different manner. The PSPG input (pin T) is not connected; thus PSPG remains at a constant negative potential and the D5 diodes are ineffective. The Sprocket signal, however, controls the Ready logic in the Peripheral Buffer so that the reader is only considered "ready" if the sprocket hole is over its photo-diode. Since the IN command is only effective if the addressed device is ready, the effect is the same as "gated" operation.

MPC Board

The MPC board, shown by Fig. THEORY.5, is located next to the LAC board at the right-hand side of the reader assembly. It contains the following circuit elements:

- Run/Brake Flip-Flop (T1-2): This standard diode-coupled flip-flop accepts set inputs through D1-2 and clear inputs through D4-5. One output of the flip-flop (T2 collector) controls the Brake relay driver (T3-5). The other output of the flip-flop (T1 collector) controls the Pinch Roller relay driver (T4-6). Refer to the "Tape Control" discussion earlier in this section for a description of the various inputs.

- Brake Relay Driver (T3-5): When the Run/Brake flip-flop is cleared, this circuit closes the positive (0V) connection to the Brake solenoid. A 2N1545 is used in the output stage to handle the 1.5 amp surge current which occurs as the solenoid switches.

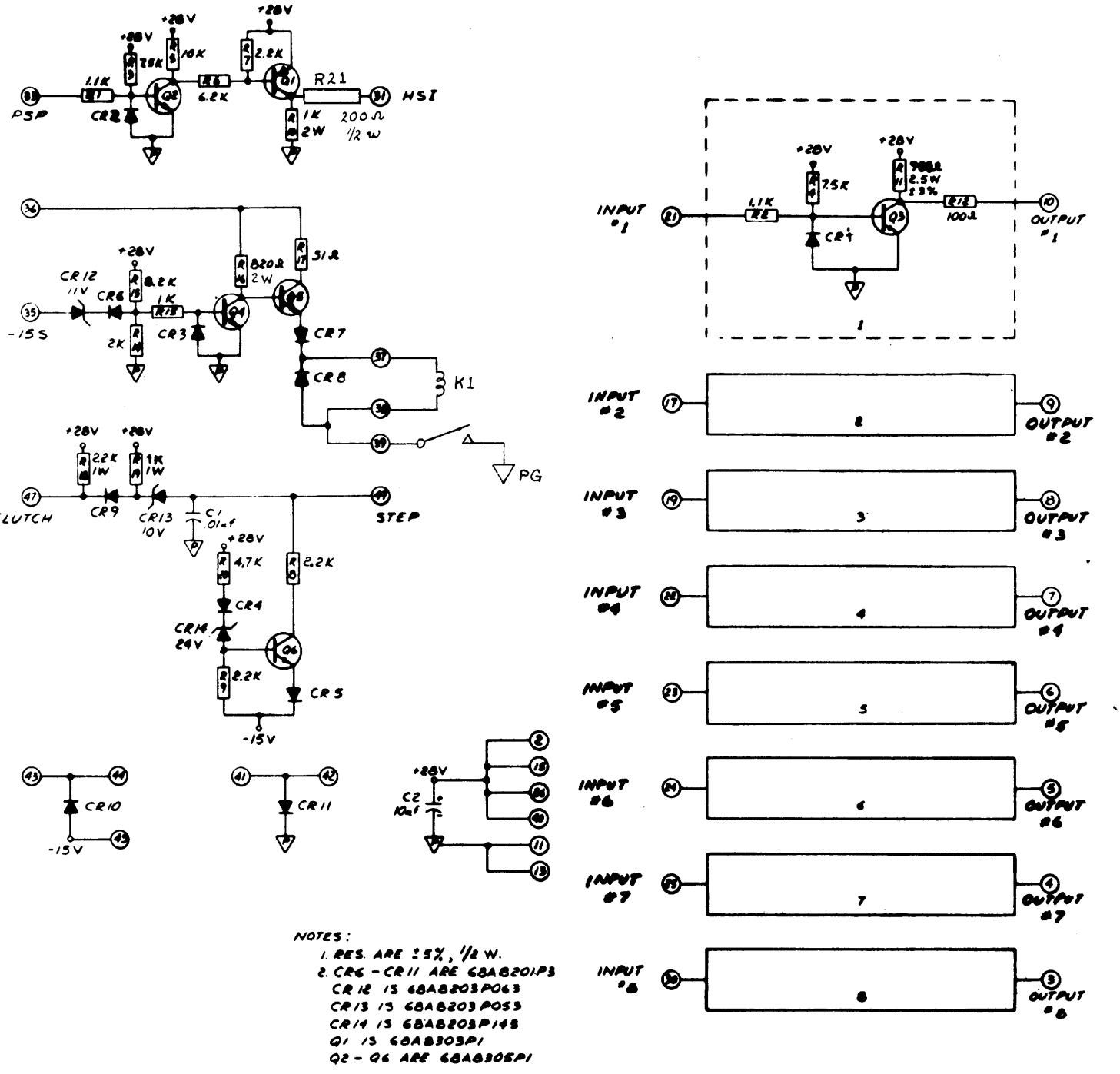
- Pinch Roller Relay Driver (T4-6): When the Run/Brake flip-flop is set, this circuit closes the positive (0V) connection to the Pinch Roller solenoid. A 2N1545 is used in the output stage to handle the 1.5 amp surge current which occurs as the solenoid switches.

- Full Wave Rectifier (CR3): This circuit is part of the ±15V power supply. See Fig. THEORY.7 for the complete schematic.

- Filter Capacitors (C4-7): These capacitors aid in removing transients caused by switching the Brake and Pinch Roller solenoids.

MAINTENANCE

See Section 12 of the Computer Maintenance Manual.



NOTES:
1. RES. ARE ±5%, 1/2 W.
2. CR6 - CR11 ARE 68A8201P3
CR12 IS 68A8203P063
CR13 IS 68A8203P053
CR14 IS 68A8203P143
Q1 IS 68A8303P1
Q2 - Q6 ARE 68A8305P1

Fig. THEORY.3. PDTA Schematic

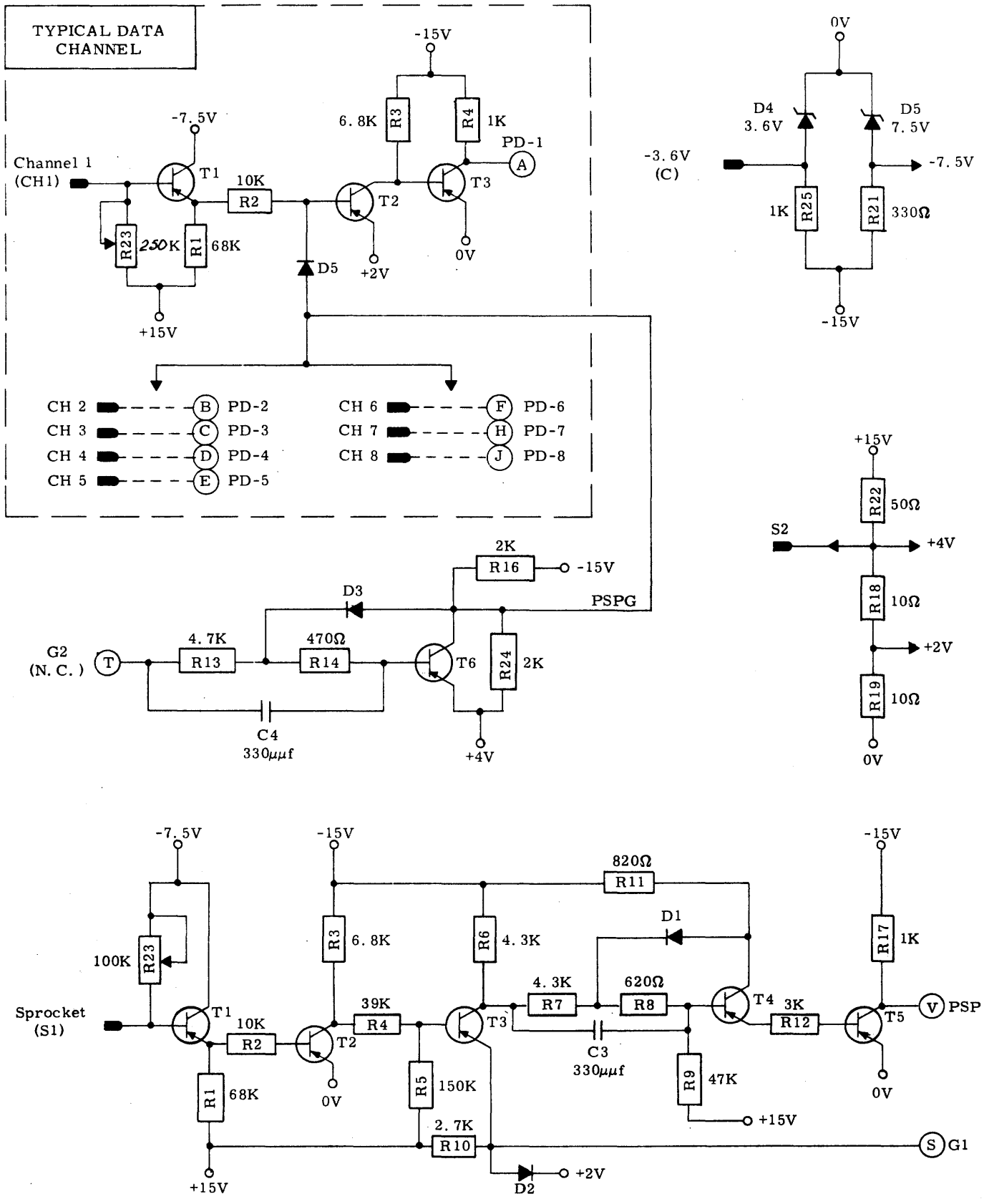


Fig. THEORY.4. LAC Schematic

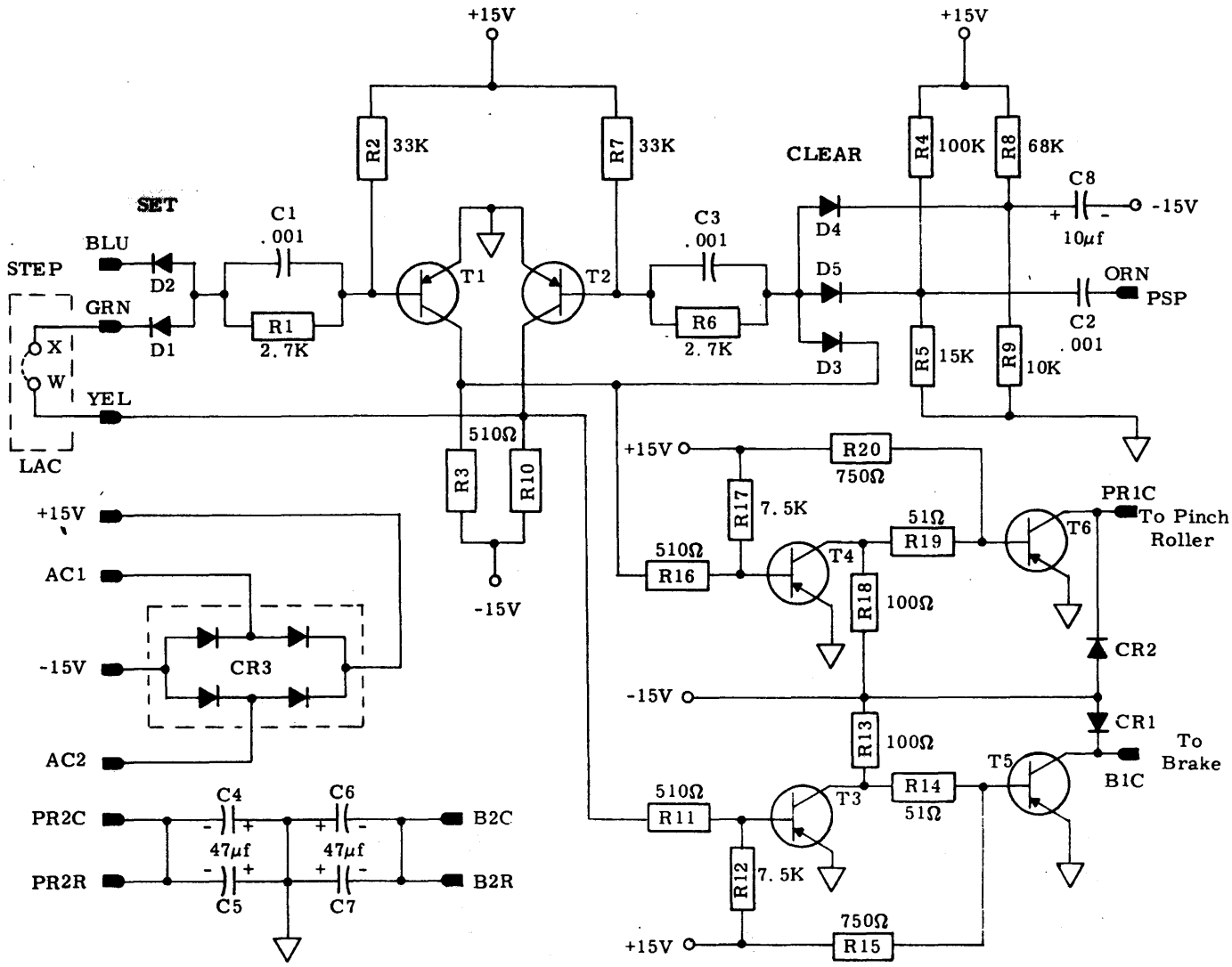
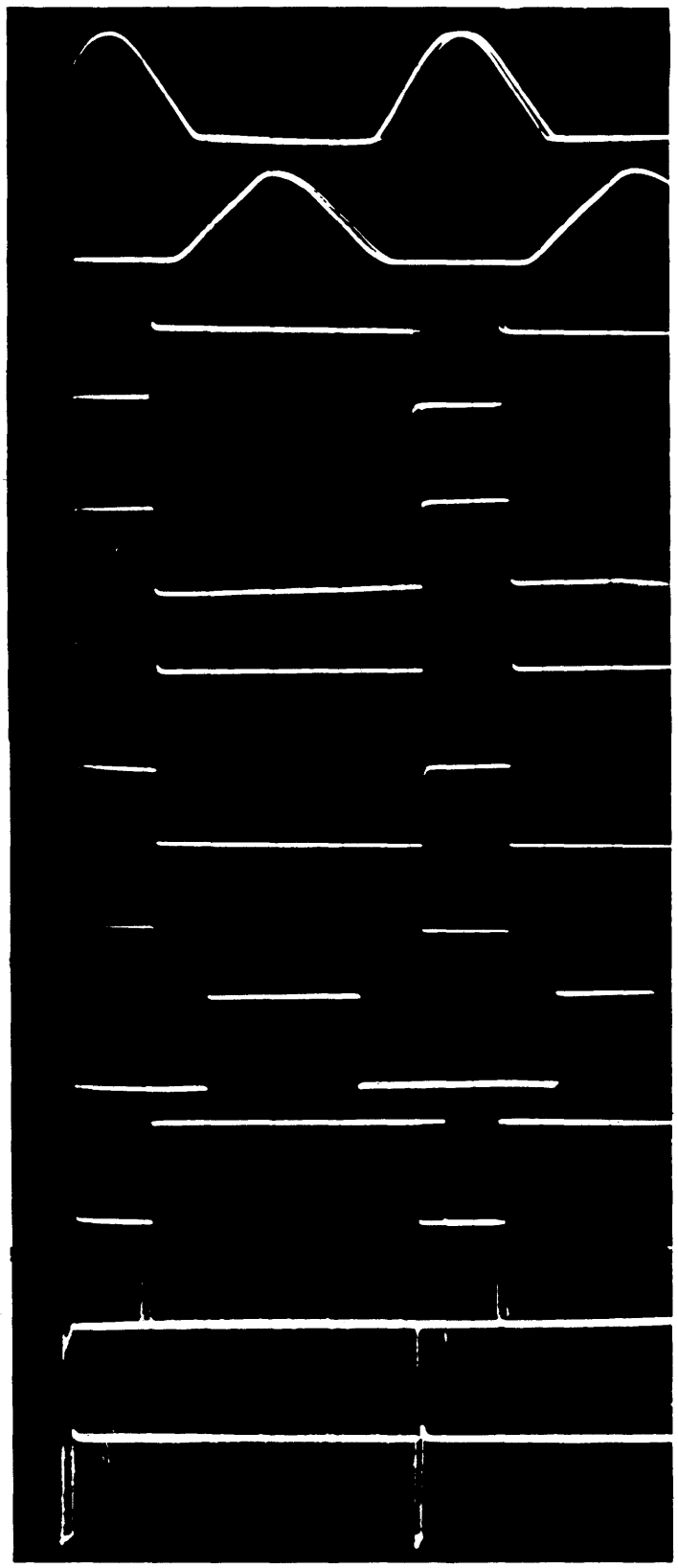


Fig. THEORY.5. MPC Schematic



1. Sprocket Input (LAC-S1)
2. Channel 1 Input (LAC-CH1)
3. Sprocket Amp Output (LAC-R3/R4)
4. Schmitt Trigger Output (LAC-R10-02)
5. PSP (MPC-ORN)
6. Y1ARTI (PDTA-31)
7. Channel 1 (PDT4-21)
8. Y0ARST (LAC-BLU)
9. PSP (MPC-C2/R4)
10. Pinch Roller

NOTE: These waveforms taken with "continuous" reading of 200 ops reader, syncing on STEP (8).

Fig. THEORY. 6 Waveforms

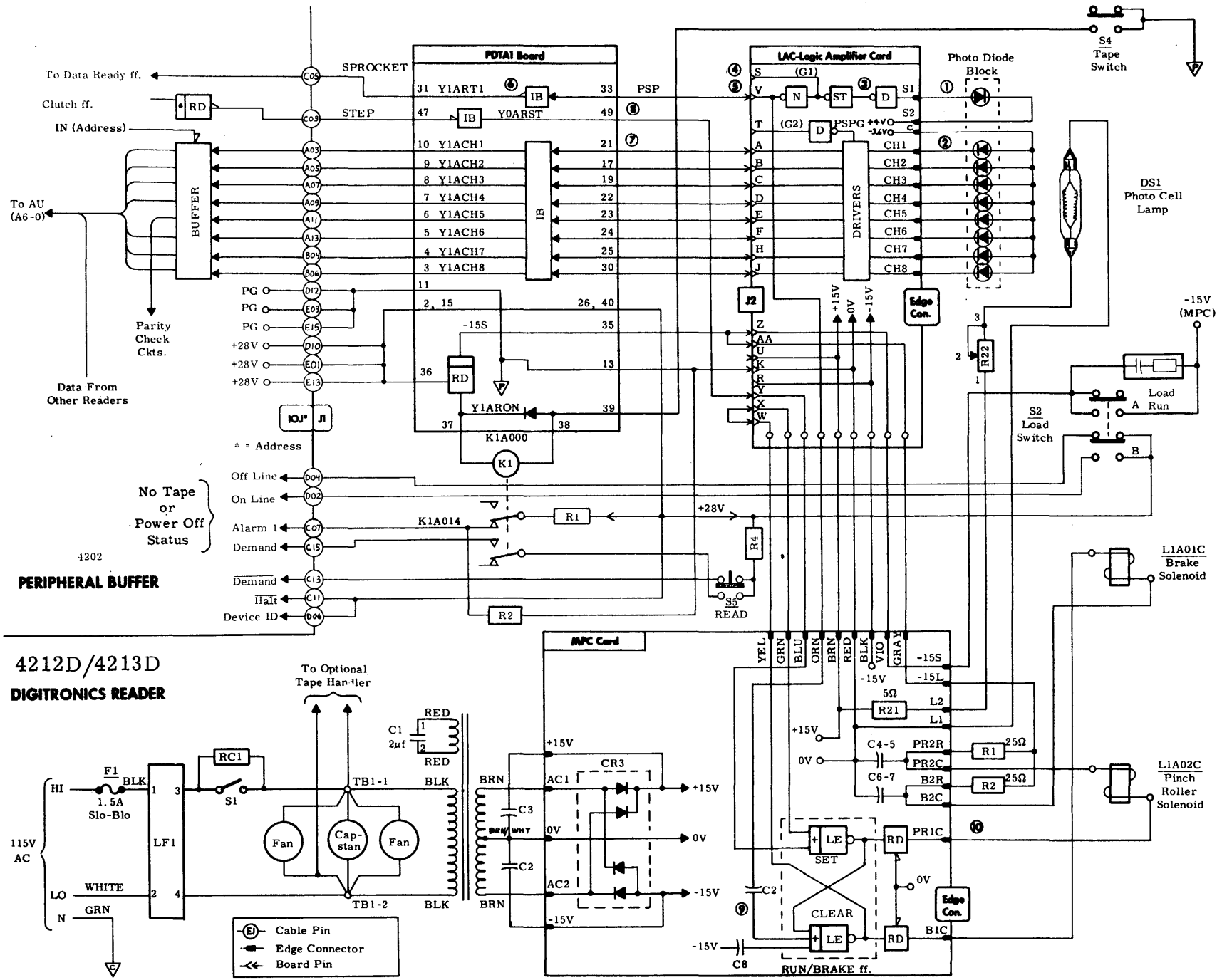


Fig. THEORY.7. Overall Reader Schematic-4212 C Model

READER COMMENTS

The General Electric Company solicits your comments on publications covering Process Computer equipment. Please explain any "no" responses in the COMMENTS section. Your comments and suggestions become the property of the General Electric Company.

- Name of Manual: _____
- What is your computer application: _____

  _____
- How is this publication used:

  Familiarization ☐ Reference ☐

  Training ☐ Maintenance ☐

  Other (Explain) _____

	YES	NO
• Does this publication meet your requirements	☐	☐
• Is the material:		
1) Presented in clear text	☐	☐
2) Conveniently organized	☐	☐
3) Adequately detailed	☐	☐
4) Adequately illustrated	☐	☐
5) Presented at appropriate technical level	☐	☐

- Please provide specific text references (page number, line, etc.) with your comments.

NAME _____ DATE _____

TITLE _____

COMPANY NAME _____

AND ADDRESS _____

COMMENTS:

No postage necessary if mailed in the U. S. A.

Staple

Communications concerning Technical Publications should be directed to:

Manager, Technical Publications
Utility and Process Automation Systems Operation
2255 West Desert Cove Road
Phoenix, Arizona 85029

Fold Fold

FIRST CLASS
Permit No. 4091
Phoenix, Arizona

BUSINESS REPLY MAIL
NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY...

GENERAL ELECTRIC COMPANY
UTILITY and PROCESS AUTOMATION
SYSTEMS OPERATION
2255 West Desert Cove Road
Phoenix, Arizona 85029

Attention: Technical Publications

Cut Along Line

Fold Fold

Additional Comments:

FIXED CARRIAGE TYPERS 4221D/4270D

TABLE OF CONTENTS

INTRODUCTION

Two Fixed Carriage Typer models are available for use in GE-PAC* 4010 computer systems. Model 4221D is the output only version which, when on-line, types data transferred from the Arithmetic Unit, through the 4820A I/O Buffer. Model 4270D is the input/output version which, when on-line, types data transferred through the I/O Buffer, and generates parallel binary-coded characters as the operator strikes keys, for transfer through the I/O Buffer to the Arithmetic Unit. Both models may be operated off-line, as normal electric typewriters.

Both models incorporate a special BCD version of an IBM Series 72 typer mechanism. The basic machine is an IBM Model 735, which uses a tilting/rotating type ball on a carrier mechanism as the printing device. The typer is modified by the Process Computer Department for compatability with the GE-PAC computer. The modification includes the exchange of the computer interface connector at the rear of the typer, the addition of a PTBA1 printed wire board in the base of the typer, the addition of AUTO alternate action pushbutton, and in Model 4270D, the addition of the INPUT alternate action pushbutton. The modification is described on GE drawing no. 68D974220.

The two IBM publications which follow this introduction describe the theory of operation of the Series 72 mechanism and the Series 72 as adapted for use as an input/output typer, which becomes Series 73. The GE logic drawings for the two typer models, listed under references are virtually self explanatory. Reference to the typer logic, the I/O Buffer logic, and the appropriate I/O Buffer theory publications, should lead to an understanding of the electrical operation of these typers.

REFERENCES

Theory of Operation (all of the following which are applicable to a system are in Volume III of this book set):

Basic I/O Buffer - pub. no. 4820A

F.C. Typer Output Drive - pub. no. 4820A-10

Input/Output Typer Control - pub. no. 4820A-09

Maintenance Instructions: Section 15 of the blue Computer Maintenance Manual.

Logic Drawings:

Basic I/O Buffer - 70C180136

4820AS10 Output Drive - 70C180354

4820A09 I/O Typer Control - 70C180348

4221D Typer (as modified) - 68C972518

4270D Typer (as modified) - 68C972521

SPECIFICATIONS

- Input Speed: 15.5 chars./sec. maximum.
- Output Speed: 15.5 chars./sec. maximum.
- Keyboard: Special BCD, see Fig. INT. 1.
- Two-magnet latching ribbon shift.
- Solenoid operated keyboard lock.
- 24V magnets and solenoids used throughout.
- Carriage: 15.5 inches with pin-feed platten.
- Carrier Return and Tab: 17" per second.
- Start-up Delay: 2 seconds (maximum) after AC power is applied (no program delay required).
- Input Power: 115 VAC ± 10%, 60 Hz ± 3%, single phase.
- Current: 1 ampere. (nominal)
- Case and Ribbon Shift Magnets: 24V DC @ 370 ma.
- All Other Magnets: 24V DC @ 185 ma.
- Solenoids: 24V DC @ 240 ma.
- Environment: Class C per drawing no. 68974933.

*Registered Trademark of General Electric Company

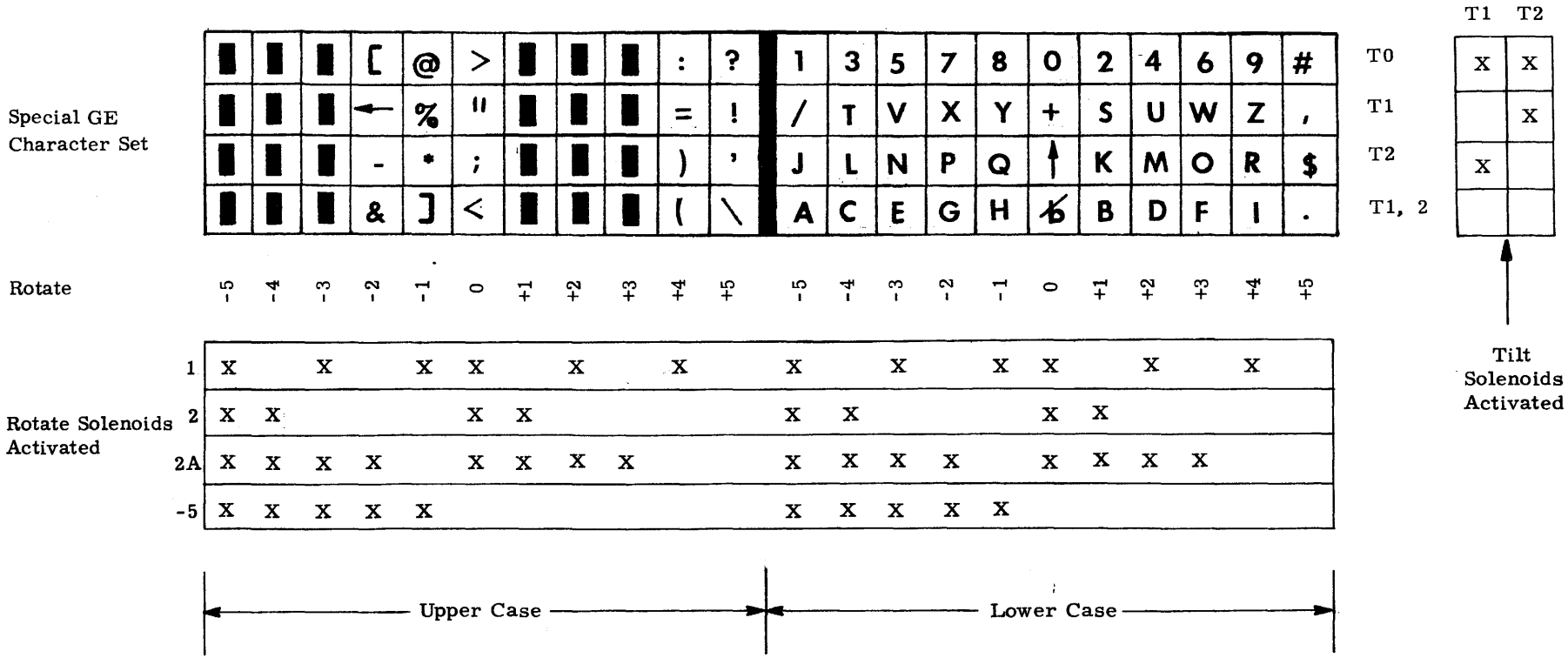
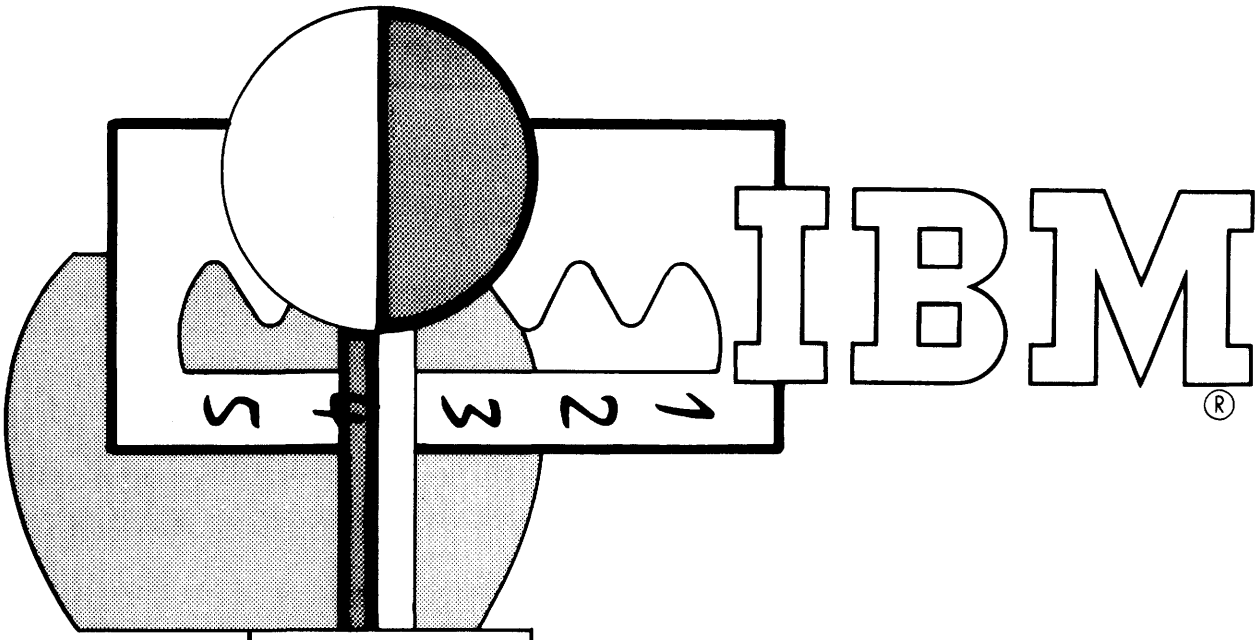


Fig. INT. 1 Type-Ball Characters



SELECTRIC INSTRUCTION MANUAL

FORM / PART NO. 241-5032-1

APRIL 1964

reprinted courtesy of IBM

TABLE OF CONTENTS

Figures 59, 60, 61, 62, 63, and 64 have been deleted.

MOTOR AND DRIVE

Motor and Electrical

The motor used in the Series 72 Selectric Typewriter is a three-inch shaded pole, induction type motor that requires 115 volts, 60 cycles A.C. (Fig. 1). It is rated at 1/40 h.p. The motor is mounted at the left rear corner of the machine with the pulley toward the right. It is attached to an adjustable bracket at each end by ring shaped spring retainers that encircle the rubber motor mounts.

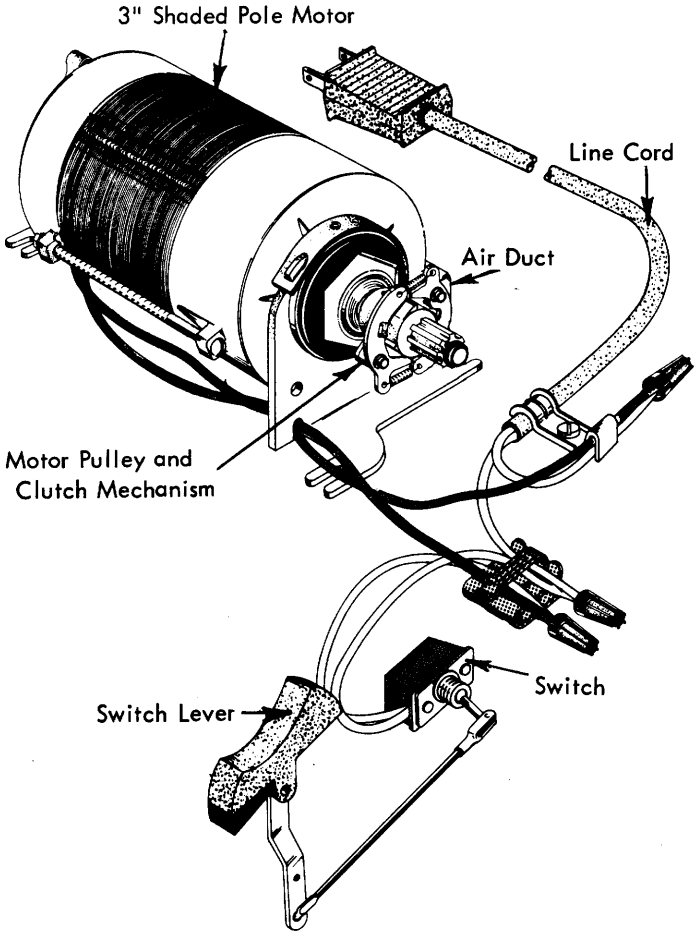


FIGURE 1. Shaded Pole Motor

The motor may be removed easily after snapping the retainers off the brackets. With the belt removed the right end of the motor may be moved to the rear, and the motor may be removed toward the right. With this bracket design, the force of the motor in operation is against the bracket rather than against the retaining clips.

The motor bearings are babbit inserts pressed solidly into the motor end bells. Each bearing is completely surrounded by a saturated oil wick. A small square wick fits into a notch in the top of each bearing to supply oil directly to the rotor shaft.

It should seldom be necessary to disassemble the motor; however, care should be taken in replacing either end bell that is not inverted when installed. The oil hole in the left end bell and the notch in the bearing would then be on the bottom.

The starting torque for the motor is provided by the shaded pole principle. No capacitor is required as in motors used with previous models of IBM Electric Typewriters.

The motor has an internal circuit breaker to prevent damage to the field coil in the event the switch is left ON with the machine stalled. The circuit will open only if the motor is allowed to remain stalled for a period of time; therefore there is no danger of an open circuit during normal operation. The motor will stall only in rare cases where a maladjustment or parts breakage causes the machine to lock. After the motor has regained normal temperature, the circuit breaker will again close the circuit. The circuit breaker will continue to open and close as long as the motor is stalled and the switch is left ON.

Because of its design, the shaded pole motor tends to run at a higher temperature than other IBM typewriter motors. Care should be taken in handling the motor to prevent being burned. In order to prevent over-heating, a cooling system is incorporated in the design of the motor. A fan attached to the right end of the rotor pulls air through the left end bell and across the field coils. The air is then discharged through a duct at the right rear corner of the motor. The air escapes through a grill in the rear of the machine cover. The hot air being discharged pulls cool air with it from within the machine. The air that ultimately emerges from the machine is cooled to near room temperature so that no objectionable heat results.

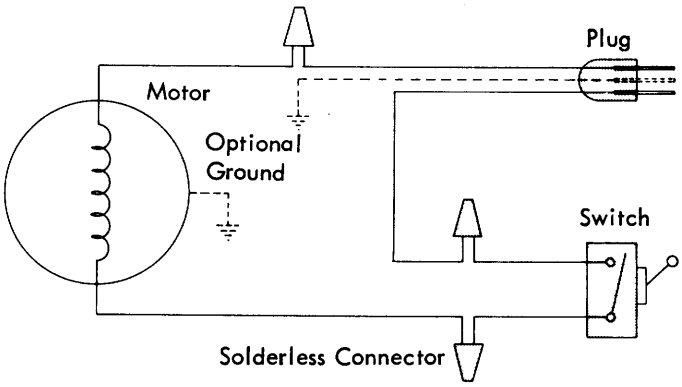


FIGURE 2. Shaded Pole Motor Diagram

In two-wire, ungrounded systems the motor is insulated from the powerframe by its rubber motor mounts. In order to convert to a three-wire grounded system, the line cord must be replaced with a three-wire cord and the ground lead must be attached to the powerframe at the cord clip screw. To complete the grounding a metal clip must be inserted in the motor mount so as to contact the hexagon end of the motor and the metal ring surrounding the rubber mount (Fig. 2).

In the early production Series 72 Selectric Typewriters, a capacitor-start induction type motor is used (Fig. 3). A three mfd. capacitor, in the starting winding circuit, provides a starting torque for the motor and controls the direction of rotation. The capacitor also remains in the circuit while the motor is running. The capacitor is mounted in a vertical position by a bracket at the right rear corner of the machine.

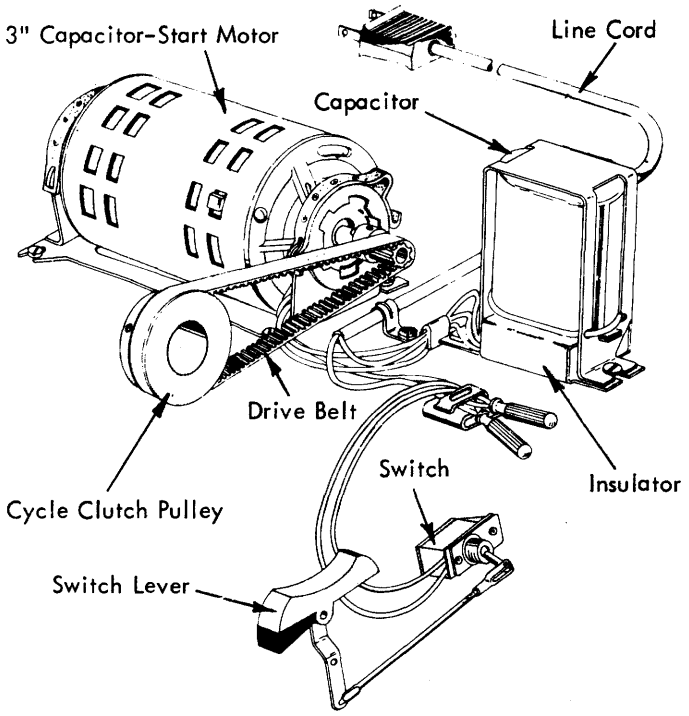


FIGURE 3. Capacitor-Start Motor

In ungrounded systems, the capacitor is isolated from the powerframe by a fiber insulator and nylon mounting screws. The motor is insulated from the powerframe by its rubber ring mounting. To convert an ungrounded system to a grounded one, three steps must be taken: the two-wire line cord must be replaced with a three-wire cord and the ground lead must be attached to the powerframe at the cord clip, a short jumper wire must be connected from the motor to the powerframe, and the capacitor must be grounded to the powerframe by replacing the nylon mounting screws with metal ones (Fig. 4).

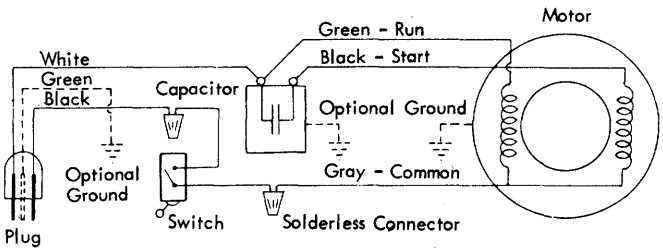


FIGURE 4. Capacitor-Start Motor Diagram

The switch and switch lever are mounted on the right side of the keyboard. The switch lever operates the electrical switch by means of a short link extending to the rear. It is operated by pressing down on the rear of the lever to turn the machine ON and the front to turn the machine OFF. The switch lever is labeled ON and OFF. When the switch lever is in the ON position, a contrasting color at the front of the switch lever shows just above the case. This calls attention to the fact that the machine is ON to minimize chances of the machine being left running when not in use.

In addition to operating the typewriter switch, the switch lever also controls the keyboard lockout mechanism. This mechanism is discussed in the keyboard section.

Drive

An eight-toothed motor pulley provides positive drive for the operation of the machine. A positive-drive belt transfers the rotation of the motor pulley to the cycle clutch pulley with a speed reduction of 3-5/8 to 1.

The shaded pole motor has slightly less starting torque than the capacitor-start motor. To insure that the motor will be allowed to start under a heavily loaded condition, a centrifugal clutch has been incorporated in the motor pulled design. The motor is allowed to approach normal operating speed, then the clutch engages to drive the machine. The momentum developed by the rotor causes the machine to start even though several mechanisms may have been tripped with the machine ON while disconnected from the electrical outlet.

The motor pulley operates freely on the end of the rotor shaft and is held in place by a grip ring retainer. Three ratchet teeth extend radially from the left end of the pulley (Fig. 5). A clutch plate hub assembly is set-screwed to the rotor shaft just to the left of the motor pulley. Pivoted on the plate are two clutch pawls (Fig. 5). When the motor is OFF, the pawls are spring loaded against stop lugs on the clutch plate. When the motor is turned ON, the clutch plate turns with the rotor. Centrifugal force causes the clutch pawls to pivot on the studs of the clutch plate so that the tip of one of the pawls engages a tooth of the motor pulley (Fig. 5). The pulley is then caused to rotate and drive the machine by means of the cycle clutch pulley.

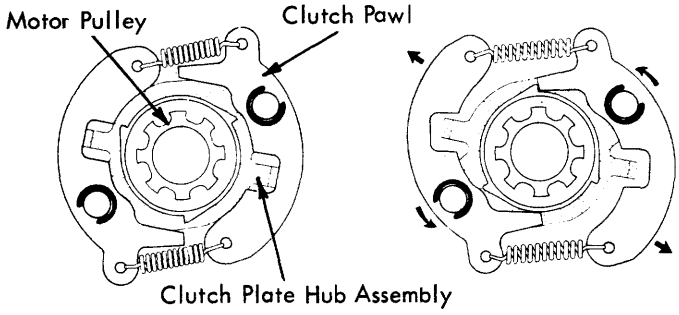


FIGURE 5. Motor Pulley Clutch

The cycle clutch pulley is mounted to a hub in the center of the powerframe. The hub is supported by a porous bronze bearing and is in continuous rotation with the pulley whenever the motor is running.

On either side of the cycle clutch pulley hub is a shaft extending into and supported by the hub (Fig. 6). The shaft to the left of the hub is called the cycle shaft. The cycle shaft is driven by means of a spring clutch. The clutch is allowed to engage whenever a letter keylever is depressed. The cycle shaft powers the positioning of the type head to the desired character. Its rotation is restricted to 180° each time a character prints. After 180° rotation the spring clutch is disengaged allowing the shaft to remain stationary. The cycle clutch is discussed fully in a later section.

Through a series of idler gears at the left, two other shafts are driven by the cycle shaft each time it operates (Fig. 6). They are the filter shaft and the print shaft. The filter shaft operates the character selection mechanisms, the print escapement, the shift interlock, and a spacebar lockout device. The print shaft operates the print mechanism, type aligning mechanism, and ribbon feed and lift mechanisms.

The shaft to the right of the cycle clutch pulley hub is the operational cam shaft (Fig. 6). All powered functional operations are driven by its rotation. The functions involved are spacebar, backspace, carrier return, indexing, and shift. The shaft also controls the speed of the carrier during a tab operation. Each of the functions is discussed in detail in its own section.

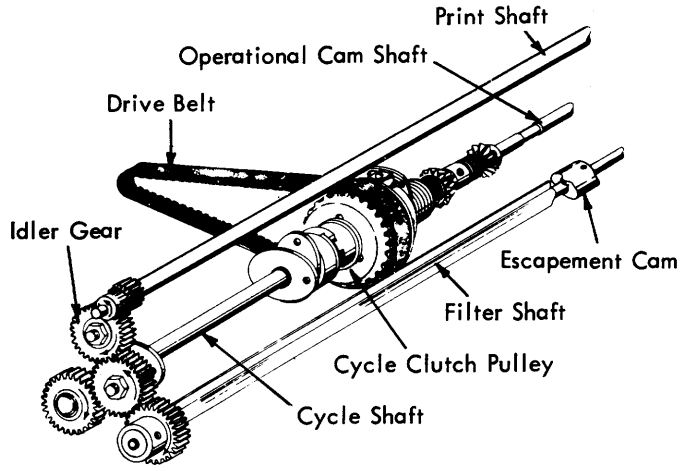


FIGURE 6. Drive Mechanism

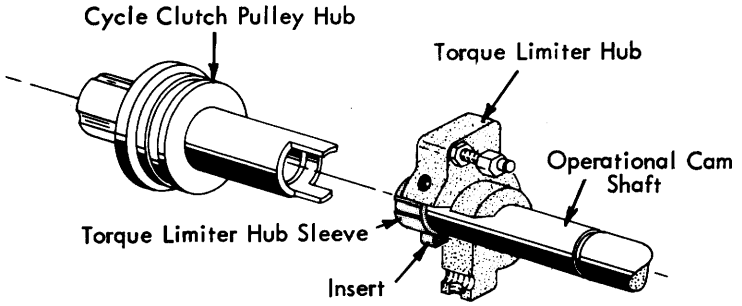


FIGURE 7. Operational Cam Shaft Drive Connection

The operational cam shaft is driven by the cycle clutch pulley hub and is in continuous rotation whenever the motor is running. The right end of the shaft operates in a self-aligning porous bronze bearing. The left end extends into the cycle clutch pulley hub where it is supported by a vinyl sleeve (Fig. 7). The sleeve provides a snug fit for the shaft in the hub to prevent any noise due to vibration. The driving connection between the cycle clutch pulley hub and the operational shaft is made by two extensions of the hub that fit into cut-outs in the left side of the torque limiter hub. The torque limiter hub is held in position at the extreme left end of the shaft by two set screws. Two nylon inserts fit into the cut-outs of the torque limiter hub (around the extensions of the cycle clutch pulley hub). The inserts provide a noiseless driving connection between the two hubs.

Just to the right of the torque limiter hub are three spring clutches and two small pinion gears. These components are part of the carrier return and tab mechanisms and are discussed in their particular sections.

TYPE HEAD

Carrier Assembly

The printing element is a ball shaped type head containing eighty-eight characters. The type head is supported in front of the paper by a framework called the carrier. The carrier is the box-shaped casting that moves laterally just in front of the platen (Fig. 8). Its purpose is to transport the type head and other related mechanisms along the writing line. Almost the entire print mechanism is contained within the carrier assembly. In addition, the carrier also supports the ribbon, ribbon feed and lift mechanisms, and a bracket that controls the left and right margins on the paper.

At the front of the carrier, a sleeve fits into two bronze bearings in the carrier (Fig. 8). The sleeve, called the print sleeve, must rotate within the carrier. It also slides left to right on the print shaft to provide the front support for the carrier. An oil soaked felt ring surrounds the print shaft and is enclosed in a retaining cup at each side of the carrier. As the carrier moves, the felt ring, called the print shaft wiper, spreads a light film of oil on the shaft to lubricate the sliding of the print sleeve. Oil from the print shaft wipers is also absorbed by the bronze bearings in the carrier casting to lubricate the rotation of the print sleeve.

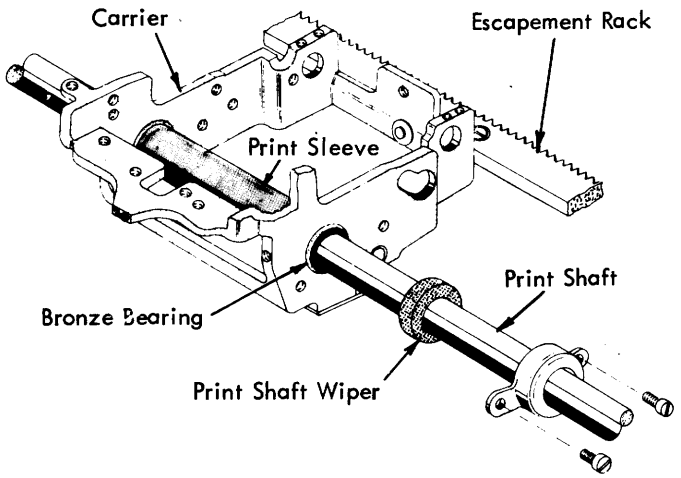


FIGURE 8. Front Carrier Support

Old Style Carrier Support

The front edge of the escapement rack serves as a rail on which the rear of the carrier rides (Fig. 9). The escapement rack is located on the powerframe just to the rear of the carrier. A square block attached to the rear of the carrier slides along the escapement rack. A similar block beneath the escapement rack prevents upward movement of the carrier. The mounting stud for the upper block has an eccentric shoulder to provide an adjustment for the up and down play at the rear of the carrier.

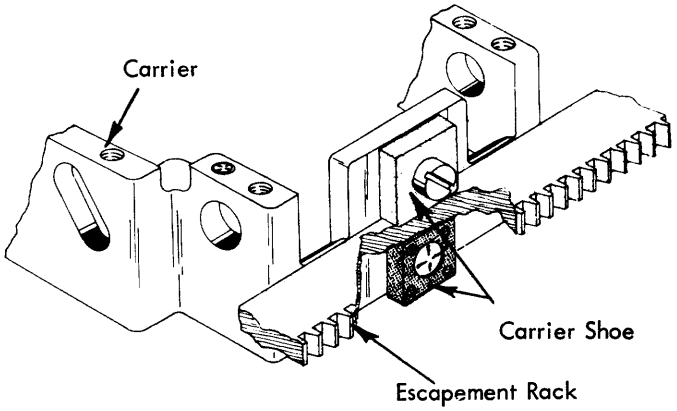


FIGURE 9. Rear Carrier Support (old style)

New Style Carrier Support

The new style carrier support (Fig. 10) differs from the old style in that the upper shoe fits loosely on its eccentric mounting stud and is spring loaded against the top surface of the escapement rack. This spring load, provided by a leaf spring, removes the play between the lower shoe and the bottom of the escapement rack thereby eliminating any vertical play at the rear of the carrier during a print operation. The bottom shoe is a nylatron block tenoned to a plate that is fastened to the rear of the carrier by the same stud that mounts the upper shoe. A stud, riveted to the plate, anchors the left end of the leaf spring. The right end of the spring presses against the underside of the escapement bracket (Fig. 10).

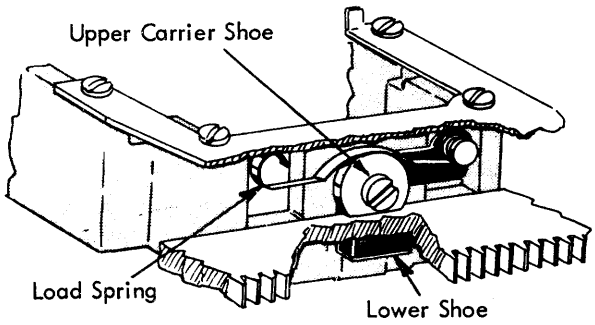


FIGURE 10. Rear Carrier Support (new style)

Rocker

The rocker is a pivoting platform located within the rear portion of the carrier (Fig. 11). Its purpose is to carry the type head to and from the platen for the print operation. Components involved in the type head positioning and aligning operations are also contained within the rocker. The rocker pivots on the rocker shaft at the rear of the carrier. Two bronze bushings, pressed into the rocker, pivot on the shaft and act as the bearing surface for the rocker. A C-clip on the right side of the rocker shaft prevents side play in the rocker. A steel thrust washer at the left of the rocker acts as a lateral bearing surface for the rocker.

Attached solidly to the top of the rocker platform is the yoke (Fig. 11). The yoke has two arms that extend up to provide a pivot mount for the tilt ring. Mounted at the top of the tilt ring is the upper ball socket to which the type head is attached. As the rocket pivots up in front, the yoke moves the tilt ring and the type head toward the platen.

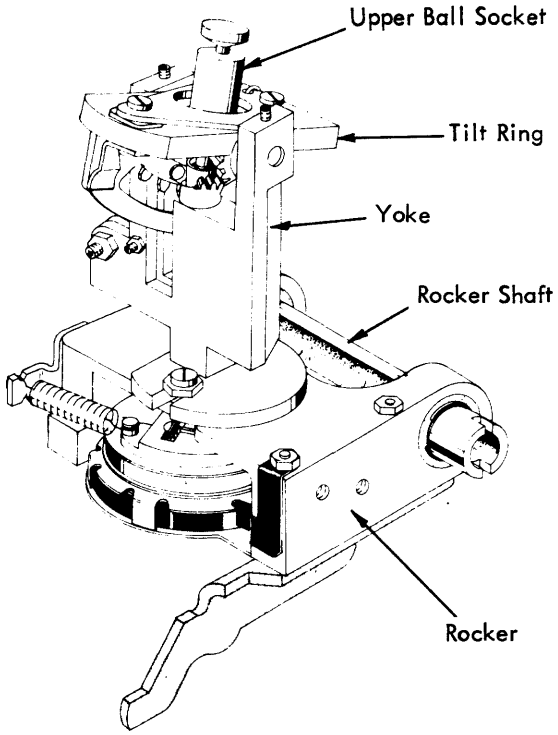


FIGURE 11. Rocker Assembly

Type Head

Before printing can occur, the desired character must be in position to strike the paper. The surface of the type head contains four bands of raised characters with twenty-two in each band (Figs. 12 and 13). Each band has eleven lower case characters in the hemisphere facing toward the platen and eleven upper case characters in the hemisphere facing away from the platen.

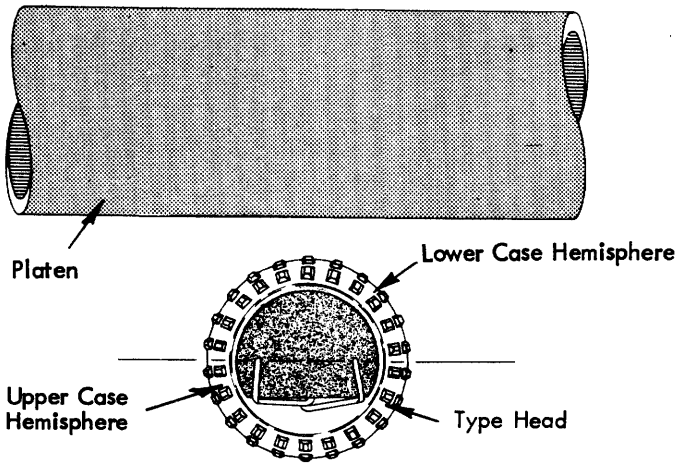


FIGURE 12. Type Head and Platen - Top View

At rest, the position of the type head is such that the middle character of the upper band is in position to strike the platen. This is the letter "z". If any character other than the "z" is desired, the head must be tilted up and/or rotated in either direction until the desired character is in the printing position. Any

lower case character may be reached by rotating the type head up to five positions in either direction and tilting the head as much as three bands from the rest position. The type head always rotates back to the "z" or rest position after a character has been typed. If an upper case character is desired, the head must be rotated counterclockwise an additional 180° so that the upper case hemisphere of the type head is toward the platen. The single-unit type head has the advantage over conventional typebars of being relatively inexpensive and easily detached from the print mechanism. This enables the operator to change quickly and easily from one type style to another merely by replacing the type head.

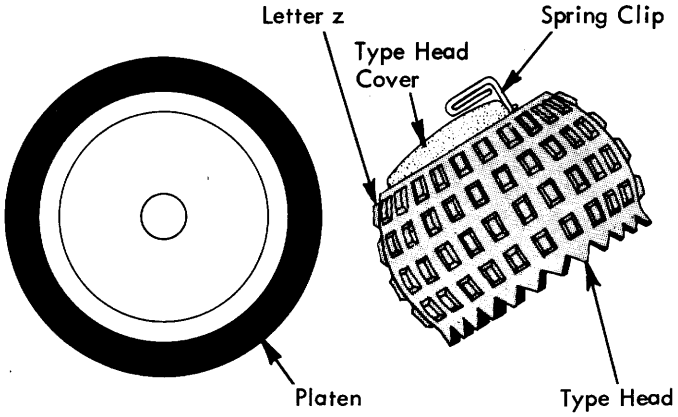


FIGURE 13. Type Head and Platen - Side View

The type head fits around a post called the upper ball socket at the top of the rocker assembly and is held in place by a spring clip. The spring clip is located on top of the type head and fits into a groove in the top of the post (Fig. 14). A convex disc, covers the spring clip except for two ears (Figs. 12 and 13). The ears of the spring clip are used in removing and installing the head. By pressing the two ears together the spring clip is disengaged from the groove in the mounting post allowing the head to be slipped up and off.

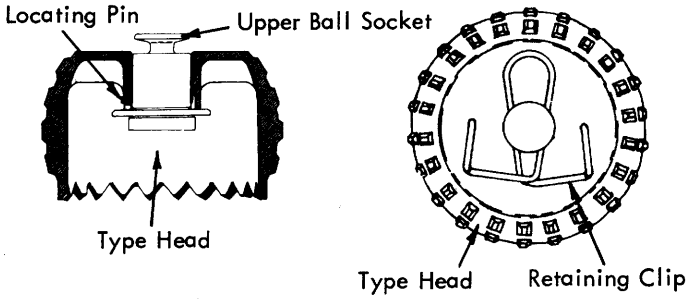


FIGURE 14. Type Head Mounting

The type head may be installed by pressing the ears of the spring clip together and slipping the head into place. The type head is keyed to the upper ball socket by a pin so that it can be installed in one position only (Fig. 14). The head must be rotated as it is installed until it drops into position on the pin. The spring clip is then released to lock the head in place. When the mechanism is at rest, the type head is always in the "z" position. When the shift mechanism is in the lower case position, the ears of the head clip are toward the front of the machine.

TILT MECHANISM

Tilt Operation

The purpose of the tilt mechanism is to raise the rear of the type head to the desired character band so that a character in that band may be brought to the printing point.

The upper ball socket is attached to a platform-like part called the tilt ring. The tilt ring pivots on two pins between yoke arms that fit up inside the hollow of the type head. The yoke assembly is fastened to the rocker to complete the type head mounting (Fig. 11).

The tilt ring is located at about the center of the type head. As the tilt ring pivots on its pins, it causes the type head to tilt. Because the type head rests with the upper band in the print position, all tilt operations are upward from the rest position.

Old Style

Movement of the tilt ring is accomplished by the operation of two beveled gear sectors located directly beneath the tilt ring (Fig. 15). One sector forms part of the tilt ring, while the other is a part of the tilt sector tube. Rotation of the tilt sector tube causes the tilt ring and type head to be tilted.

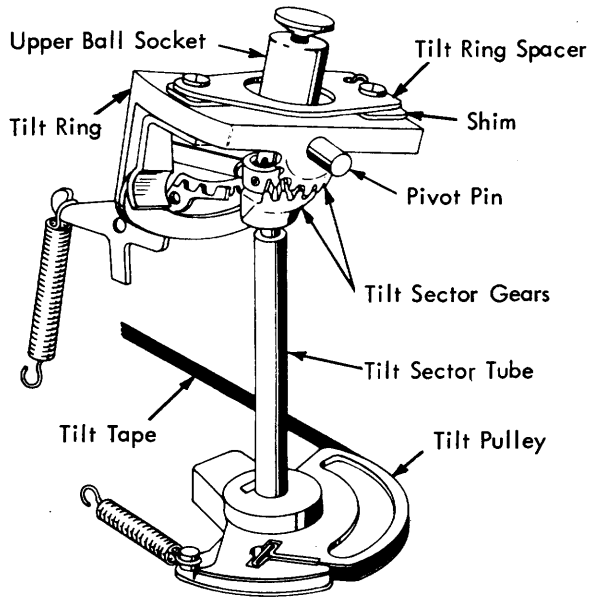


FIGURE 15. Tilt Mechanism - Rocker Portion

The tilt sector tube operates through the yoke and extends below the rocker. A pulley called the tilt sector pulley is attached to the bottom of the tube. The pulley is held to the tube by a set screw that presses a small T-shaped block against a flat surface on the tube. The pulley is held in a clockwise direction (looking from the top) by an extension spring. A small steel tape encircles the pulley and has one end attached to it. When the tape is pulled, it causes counterclockwise rotation of the tilt tube and an accompanying tilt of the type head. When the pull on the tape is relaxed, the tilt pulley is restored by its spring causing the type head to return to the rest position.

From the pulley, the tilt tape is guided through the hollow left end of the rocker shaft by a rounded tape guide block attached to the bottom of the rocker. The tilt tape extends to the left around a small pulley, back to the right around a similar pulley, and is attached to the right side of the carrier. This arrangement allows left to right movement of the carrier without disturbing the position of the type head (Fig. 16).

When a tilt operation is required, the distance between the two side pulleys is increased by moving one of the pulleys away from the other. This causes a pull on the tilt tape. The pulley on the right is solidly mounted and is moved for adjustment only. The left pulley is attached to a pivoting arm called the tilt arm. Movement of the arm to the left exerts a pull on the tilt tape to cause a rotation of the tilt tube and a tilt to the type head.

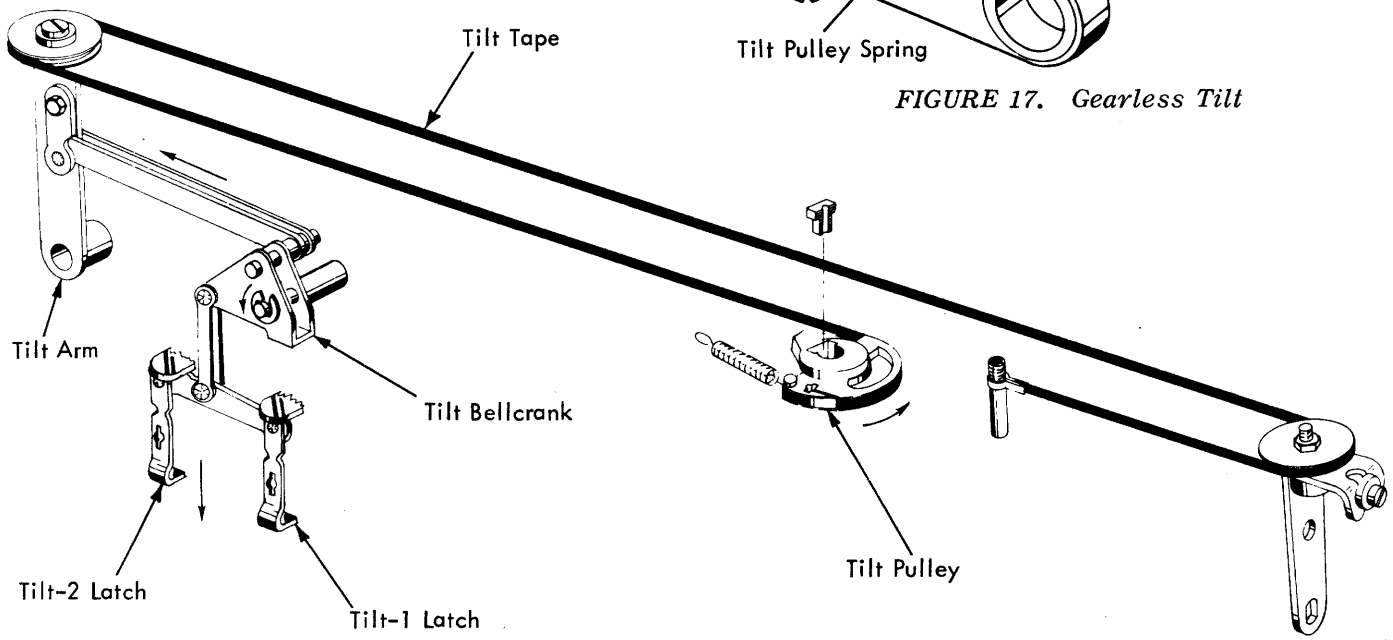


FIGURE 16. Tilt Tape System

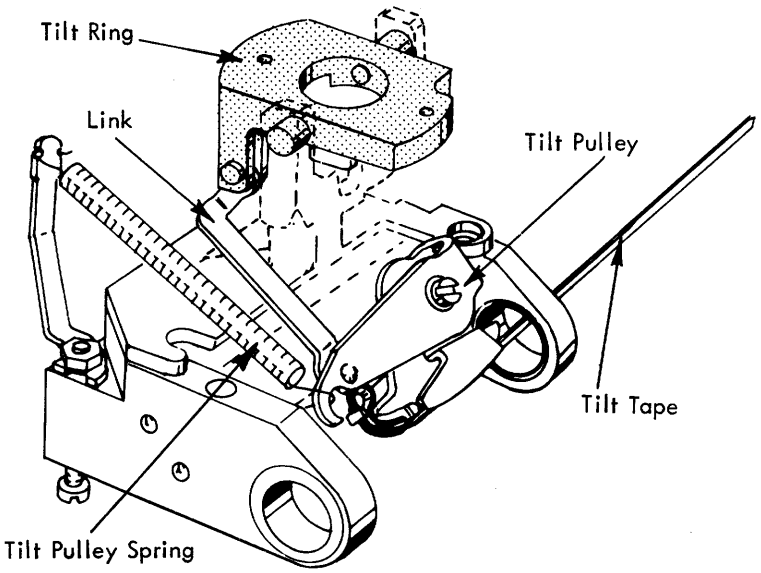


FIGURE 17. Gearless Tilt

New Style (Gearless Tilt)

The "gearless tilt mechanism" operates basically the same as the early style mechanism. Movement of the tilt ring is accomplished by the operation of the tilt pulley (Fig. 17). A pull on the tape causes the tilt pulley to rotate (about its mounting stud) transferring motion to the tilt ring by means of the tilt pulley link. When the pull on the tape is relaxed the tilt pulley is restored to rest by its extension spring.

The tilt pulley link is fastened to the tilt pulley by a ball shouldered rivet to allow the link to pivot in all directions. The other end of the link is fastened to the tilt ring by a pin and C-clip.

Latch Bail

The cycle shaft has three double lobed cams on it that power the positioning of the type head (Fig. 18). The three cams are separated on the cycle shaft, one on the left and two on the right. The left cam and the middle cam are paired to operate the latch bail located just beneath the cycle shaft. The right cam can be disregarded for the moment.

The cycle shaft powers the positioning of the type head by operating the selector latch bail. The latch bail is a box-shaped frame located just beneath the cycle shaft (Fig. 18). Two short arms of the frame extend forward where they pivot on a shaft mounted to the powerframe. Each side of the latch bail contains a roller that is constantly in contact with its respective cycle shaft cam. An extension spring at the rear of the latch bail applies a constant upward pressure to hold the rollers against the cams. Each time the cycle shaft operates 180°, the bail is forced down at the rear pivoting about the bail shaft.

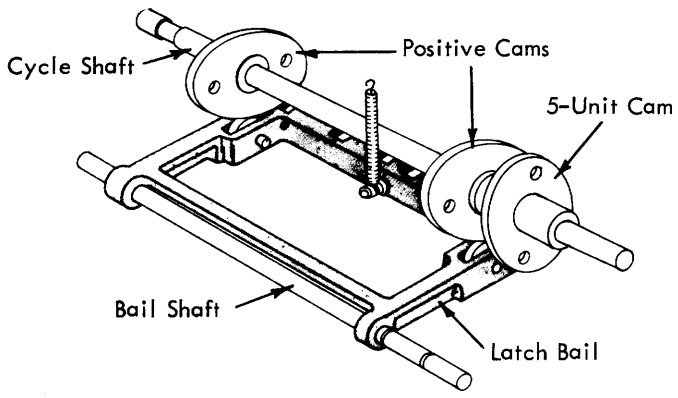


FIGURE 18. Cycle Shaft and Latch Bail

The rear of the latch bail is recessed at six points. Five of the recesses contain selector latches. The sixth recess if for special applications of the machine. Across the rear of the latch bail is a plate attached by four screws. Each selector latch has a lip formed to the rear just under the bail plate (Fig. 19). An extension spring holds each latch to the rear. The five selector latches are components of the differential lever assembly that determines how much tilt and rotate the head will receive. The two latches to the left are concerned with tilting the head, while the three on the right deal with rotating the head.

If the latches remain to the rear under the plate, they will be pulled down when the bail is operated. If any latch is held forward, it is not hooked under the bail plate and will not be pulled down during an operation of the latch bail. The method of pulling the various latches forward is discussed in a later section.

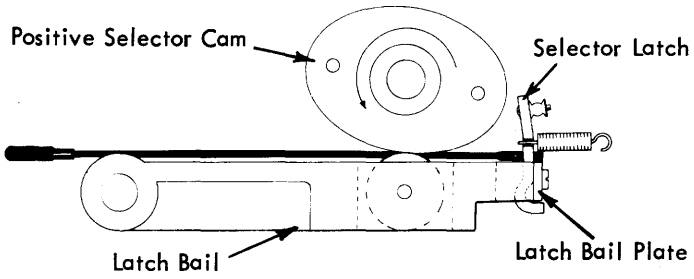


FIGURE 19. Latch Bail - Side View

Tilt Differential

The two tilt latches are attached at each end of a short lever by ball shouldered rivets (Fig. 20). The ball shape of the shoulders allows the latches to pivot in all directions. The lever is attached by a double vertical link to the tilt bellcrank. The bellcrank pivots on a stud at the top of the differential bracket. The connection of the double link is not in the middle of the lever; therefore the leverage developed by one tilt latch is greater than that of the other.

A horizontal link connects the top of the tilt bellcrank to the tilt arm. Operation of the bellcrank forces the tilt arm to the left to exert a pull on the tilt tape. The tilt arm is sometimes referred to as the tilt multiplying arm, because the movement of the horizontal link is increased at the pulley due to the leverage developed.

The left hand tilt pulley is mounted to the tilt arm on a ball shouldered pivot screw. This allows the pulley to remain horizontal regardless of the position of the tilt arm. It must remain horizontal to prevent the tilt tape from coming off the pulley.

The tilt bellcrank is rotated by a pull on the tilt latches. When the left latch is held to the front while the right one remains to the rear, only the right latch is forced down by the latch bail (Fig. 21). As the latch pulls down on its attached lever, the left end of the lever pivots against a stop lug formed out from the differential bracket. The vertical link from the lever is then pulled to operate the tilt bellcrank. The same action occurs if the left latch is pulled down by the latch bail while the right latch is held forward. The distance the vertical link is pulled is not the same for both latches, because the link is not connected to the middle of the lever.

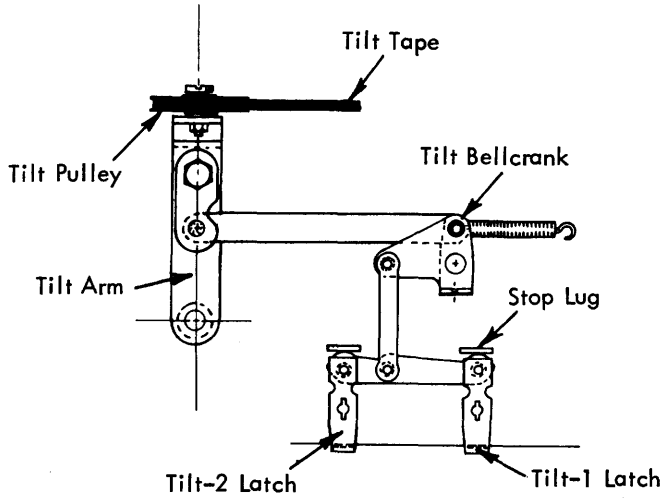


FIGURE 20. Tilt Differential At Rest

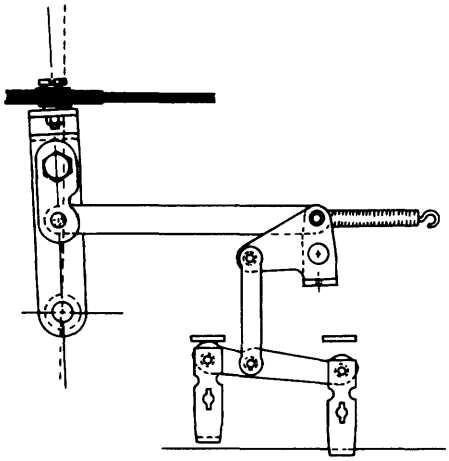


FIGURE 21. Tilt - 1 Operation

When only the right latch is being operated by the bail, the left end of the lever is not pulled down and acts as a pivot point. The vertical link is attached to the lever one-third of the distance from the pivot point to the right latch. This causes the link to be pulled down only one-third as far as the latch is pulled by the bail. The movement of the link is sufficient to cause the type head to tilt a distance of one band of characters. This places the second band from the top in the printing position.

When only the left latch is operated, the right end of the lever acts as a pivot point (Fig. 22). The vertical link is then two-thirds of the distance from the pivot point to the operating latch and is moved two-thirds as much as the latch. This movement is sufficient to cause the type head to tilt a distance of two bands of characters. The third band is then in the printing position.

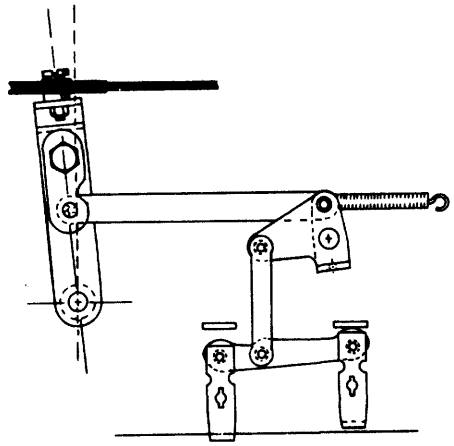


FIGURE 22. Tilt - 2 Operation

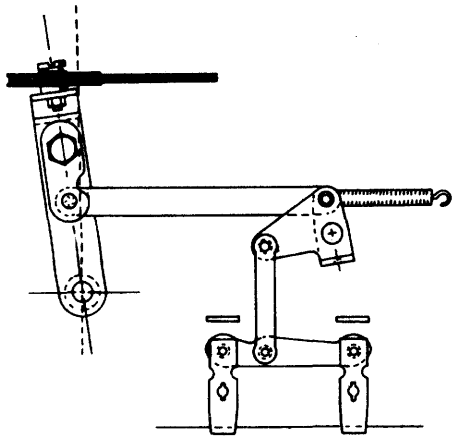


FIGURE 23. Tilt - 3 Operation

When both latches remain to the rear under the latch bail, both are operated. The lever moves straight down and neither end acts as a pivot point (Fig. 23). This causes the double vertical link to receive the same motion as the latches, resulting in three character bands of tilt. The fourth band is then in the printing position.

When operated by itself, the right tilt latch causes a tilt of one character band; therefore it is referred to as the tilt-one latch. Because the left latch causes a tilt of two character bands, it is called the tilt-two latch. Both latches operated together cause a tilt-three action.

ROTATE MECHANISM

Rotate Operation

The rotate mechanism positions the rear of the type head left or right to the desired character within a tilt band.

The upper ball socket to which the type head is attached has a shoulder at the bottom fitted into a hole in the tilt ring. The fit is very close yet permits free rotary motion of the upper ball socket. The upper ball socket is held in place by the tilt ring spacer. The spacer attached directly to the tilt ring and fits around a flange of the upper ball socket. The tilt ring spacer is shimmed to allow rotation of the upper ball socket yet restrict up and down play.

The under side of the upper ball socket is hollow and forms the socket for a ball joint connection (Fig. 24). A dog-bone shaped ball joint fits into the socket over a pin that extends through the socket. The ball joint is identical at both ends. The lower end fits over a pin in the lower ball socket. These two ball and socket connections act as universal joints to permit the type head to be rotated and tilted at the same time.

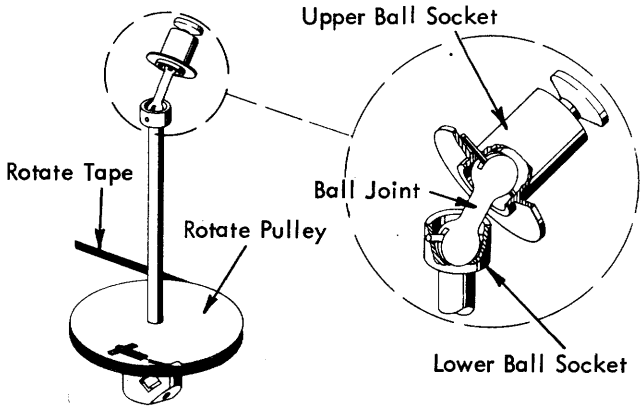


FIGURE 24. Rotate Mechanism - Rocker Portion

The lower ball socket is part of a shaft called the rotate shaft. The shaft operates inside the hollow of the tilt sector tube and extends well below the tilt sector pulley. (On the gearless tilt, the tilt sector tube and the tilt pulley have been eliminated. The rotate shaft operates directly inside the hollow of the yoke). On both styles the bottom of the shaft pivots in a hole in a plate attached to the bottom of the rocker. This plate acts as a bearing for the bottom of the rotate shaft. Attached near the bottom of the shaft is the rotate pulley similar in operation to the tilt sector pulley (Fig. 24). By means of the upper and lower ball sockets and the ball joint, the rotate pulley operates to rotate the type head in either direction.

As on the tilt mechanism, the rotate pulley is operated by a steel tape that passes around two side pulleys and attaches to the carrier. When either of the side pulleys moves away from the other, the type head is rotated counterclockwise by the pull of the tape. When either side pulley moves toward the other, the type head is rotated clockwise by the tension of the rotate pulley spring (Fig. 25).

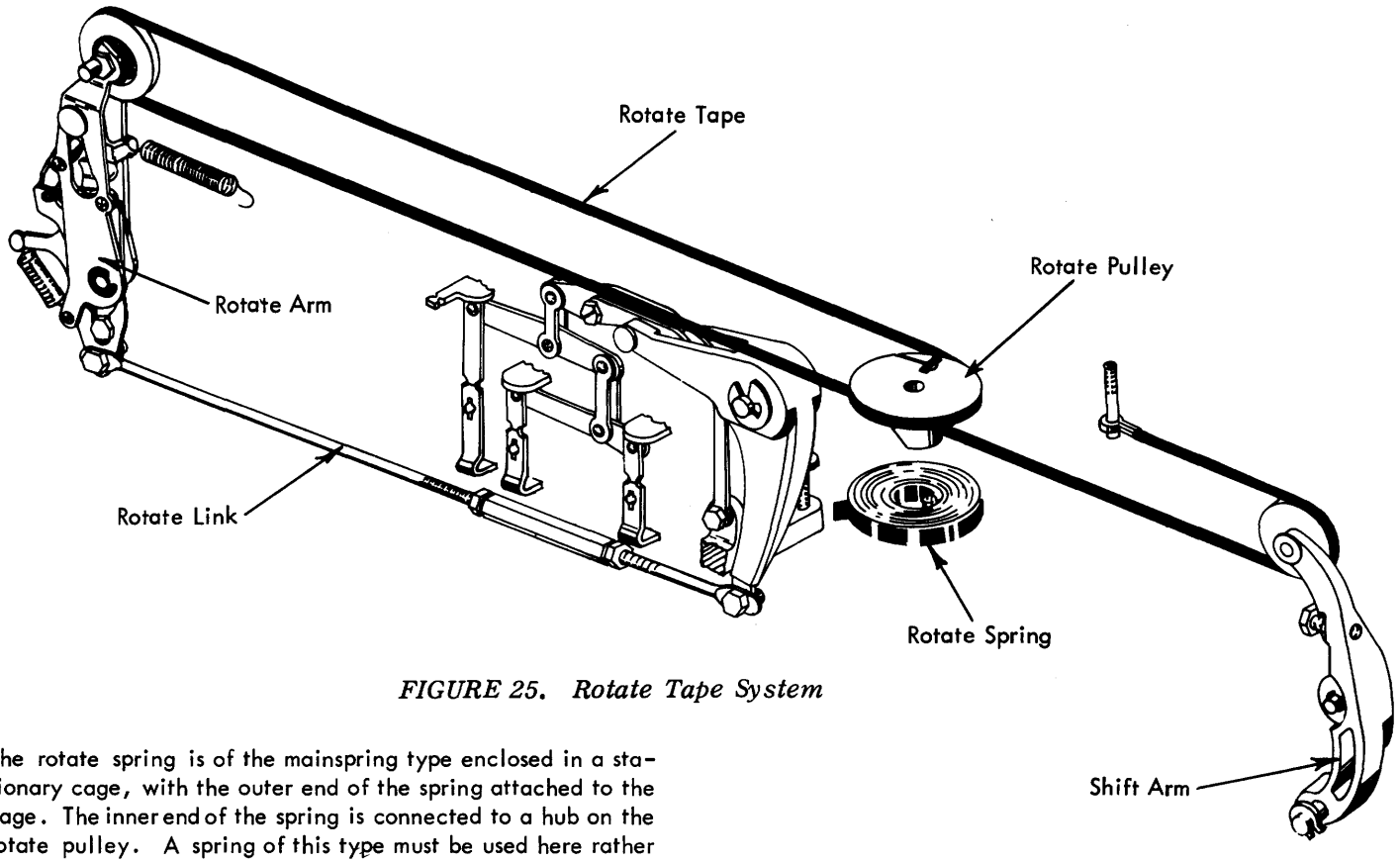


FIGURE 25. Rotate Tape System

The rotate spring is of the mainspring type enclosed in a stationary cage, with the outer end of the spring attached to the cage. The inner end of the spring is connected to a hub on the rotate pulley. A spring of this type must be used here rather than an extension spring as is used on the tilt sector pulley, because the type head is required to rotate almost a full revolution.

The right hand rotate pulley is attached to the shift arm and moves only during the shift operation; therefore consider it to be stationary for the present. The left hand rotate pulley is attached to the rotate arm. When the arm moves away from the side frame, it exerts a pull on the tape to rotate the typehead counterclockwise. This direction is known as the positive direction of rotation.

When the left hand rotate pulley moves towards the sideframe, the rotate spring turns the rotate pulley rotating the typehead in a clockwise direction. This direction is known as the negative direction of rotation.

Positive Rotate Differential

The rotate differential is much the same as the tilt differential. The latches are operated by the latch bail if they are allowed to remain to the rear under the bail. Each operates with a different leverage for different amounts of rotation. Rotation of up to five characters is sometimes required on either side of the rest position. This requires more latches and levers than for a tilt operation.

Consider the positive rotation of the type head first. Three latches and a series of three levers are involved in positive rotation (Fig. 26). The three latches are those farthest to the right in the latch bail. All the latches are spring loaded to the rear under the latch bail and are operated by the bail unless they are pulled forward.

The two latches on the extreme right are mounted by ball shouldered rivets to a short lever similar to the tilt latch mounting. A flat double link extends vertically from the lever to a second lever above it. The connection is at the right end of the second lever and one-third of the distance from the right end of the first lever.

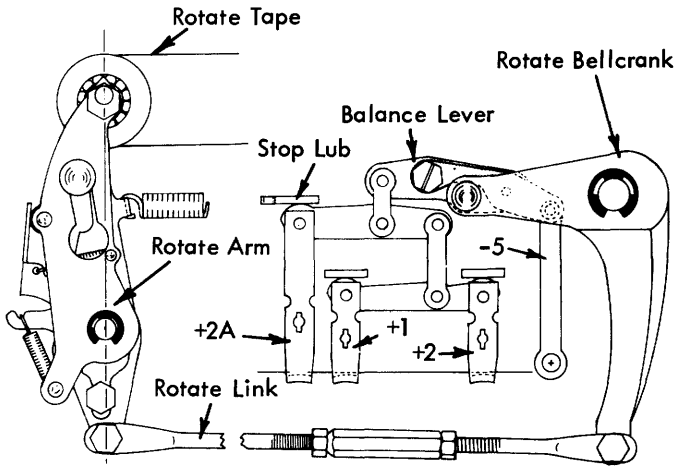


FIGURE 26. Rotate Differential At Rest

The third latch is connected by a ball shouldered rivet to the left end of the second lever. Because its mounting point is higher than the other latches, the third latch is much longer than the others to permit latching under the bail at the bottom.

A second vertical link connects the second lever to the left end of the third lever in the series. The link is attached to the second lever two-fifths of the distance from the right end.

The third lever is an adjustable lever connected at the middle to the horizontal arm of the rotate bellcrank. It is referred to as the balance lever, because its adjustment balances the amount of motion between positive and negative rotation. The right end of the balance lever is held stationary during positive rotation. A downward pull at the left end causes the rotate bellcrank to operate. A heavy link connects the bottom of the rotate bellcrank to the bottom of the rotate arm. The rotate arm is sometimes referred to as the rotate multiplying arm because the movement of the rotate link is increased at the pulley as a result of the leverage. Operation of the bellcrank counterclockwise causes the rotate arm to pivot about its fulcrum point and exert a pull on the rotate tape.

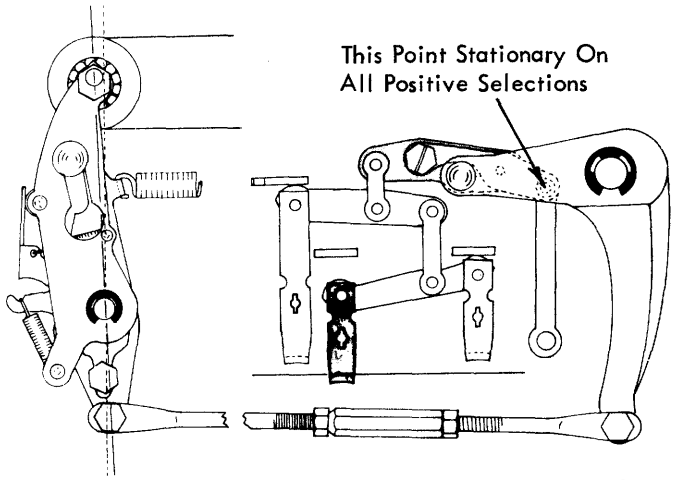


FIGURE 27. Positive - One Rotate Operation

In order to operate the balance lever, one or more of the latches must be pulled down by the latch bail. Consider the latches one at a time starting with the middle of the three rotate latches. When only the middle latch is allowed to remain under the latch bail, it is the only one pulled down when the bail operates (Fig. 27). As the middle latch is pulled down, its attached lever moves down at the left and pivots at the right on a stop lug formed out from the differential bracket. The vertical link is attached to the lever one-third of the distance from the pivot point to the latch. This causes the link to be moved down one-third as much as the latch.

The link exerts a pull on the right end of the second lever causing it to pivot on its stop lug at the left end. The second vertical link is attached to the lever three-fifths of the distance from the pivot point to the operating end of the lever. The second link moves three-fifths as much as the right end of the lever. The right end of the second lever moves one-third as much as the latch. Multiplying the two together, the second vertical link moves three-fifteenths or one-fifth as much as the latch.

The second vertical link operates the left end of the balance lever which, in turn, actuates the rotate bellcrank. This movement is sufficient to cause a positive rotation of one character on the type head. The movement is caused by operation of the middle rotate latch, so it is referred to as the rotate-one latch.

When only the right hand latch is operated, the movement of the rotate bellcrank is doubled, and a positive rotation of two characters is obtained on the type head. The increased movement is obtained at the first lever. When the right hand latch moves down, the lever pivots on a stop lug at the left. The vertical link is attached two-thirds of the distance from the pivot point to the latch; therefore the link moves down two-thirds as much as the latch. This is twice as much as for the rotate-one latch. The right hand latch is referred to as the rotate-two latch.

When both the rotate-one and rotate-two latches are operating together, the first lever moves straight down without pivoting at either end (Fig. 28). This gives the same amount of motion to the first vertical link as is given to the two latches. Movement of the link is three times as much as when operated by the rotate-one latch alone. The additional movement is transferred to the rotate bellcrank to rotate the type head three characters in the positive direction.

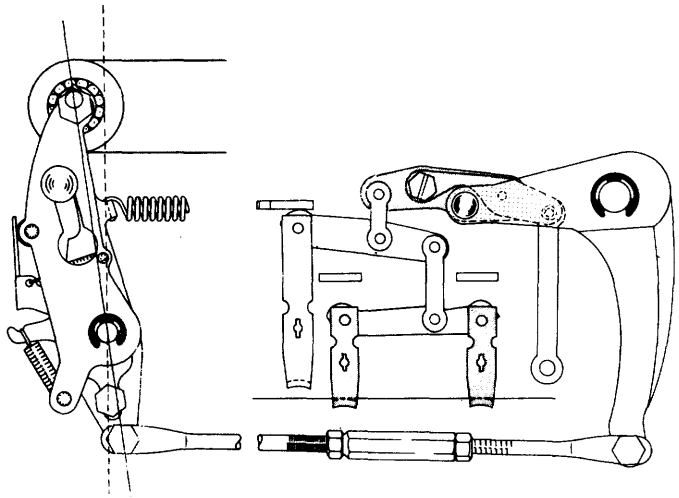


FIGURE 28. Positive - Three Rotate Operation

When additional rotation is required, the third rotate latch must be used. The left hand latch is never used alone; but to clarify its leverage in relation to the other latches, consider it to be the only one in operation.

The left hand latch is attached directly to the end of the second lever. When operated, the latch causes the left end of the second lever to move down. The right end of the lever cannot rise so it acts as a pivot point. The second vertical link is attached to the lever two-fifths of the distance from the pivot point to the latch; therefore the link moves down two-fifths as much as the latch.

Movement of the link is twice as much as when the rotate-one latch is operated alone. Therefore the movement obtained from the left hand latch is sufficient to rotate the type head two characters in the positive direction. Since the right hand latch is called the rotate-two latch, the left hand latch is referred to as the rotate-2A latch.

The 2A latch is never used by itself. When its motion is added to that of one or both the other latches, rotation of four or five characters can be obtained. The rotate-one latch is used for one-character rotation and the rotate-two latch for two-

character rotation. Both are operated together for three-character rotation. The rotate-two and rotate-2A latches are operated for a four-character rotation. The rotate-one, rotate-two, and rotate-2A are operated for a five-character rotation (Fig. 29).

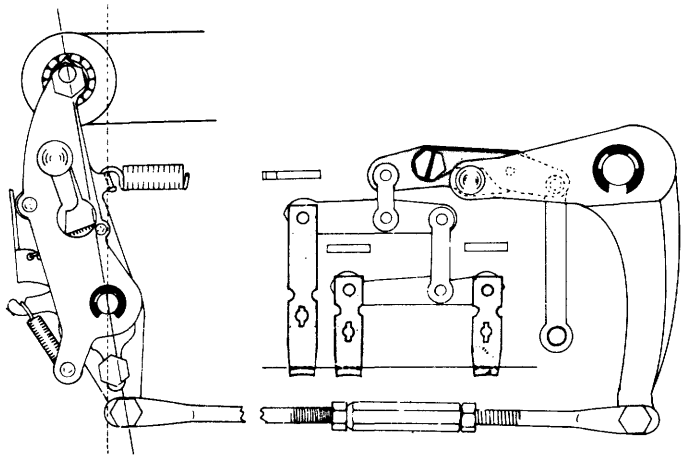


FIGURE 29. Positive - Five Rotate Operation

Negative Rotation

Positive rotation of the type head is achieved, by operating the rotate bellcrank counterclockwise so as to create a pull on the rotate tape. It follows then that operating the rotate bellcrank clockwise will allow the rotate pulley spring to rotate in a negative direction.

The rotate bellcrank is controlled by the balance lever in the differential series. In order for the bellcrank to operate clockwise, the balance lever must be raised. The left end of the lever cannot rise, because the stop lugs on the bracket prevent any upward movement in the lever series. Therefore if the bellcrank is to operate clockwise, the right end of the lever must be raised.

The right end of the balance lever has a flat link connection to the five-unit bail assembly (Fig. 30). The bail is a single arm located under the cycle shaft and pivoted in front on the bail shaft. When the five-unit bail is allowed to rise, the right end of the balance lever rises to allow clockwise operation of the rotate bellcrank.

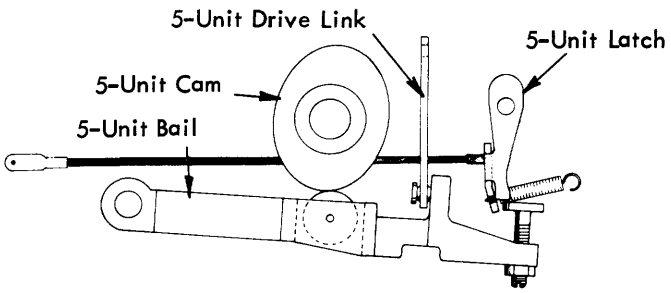


FIGURE 30. Five Unit Bail At Rest

The five-unit bail is prevented from rising by the five-unit latch at the rear (Fig. 31). The latch is mounted to the differential bracket and pivots front to rear. In the rest position the latch is positioned above the head of an adjusting screw at the rear of the five-unit bail. When the latch is pulled forward the bail is allowed to rise (Fig. 32). The force which raises the bail comes from the rotate pulley spring and the spring attached to the rotate arm. These springs are applying a constant force on the rotate bellcrank in the clockwise direction.

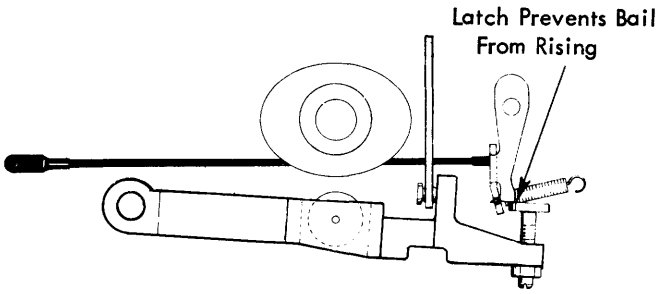


FIGURE 31. Five-Unit Bail During Positive Rotate Cycle

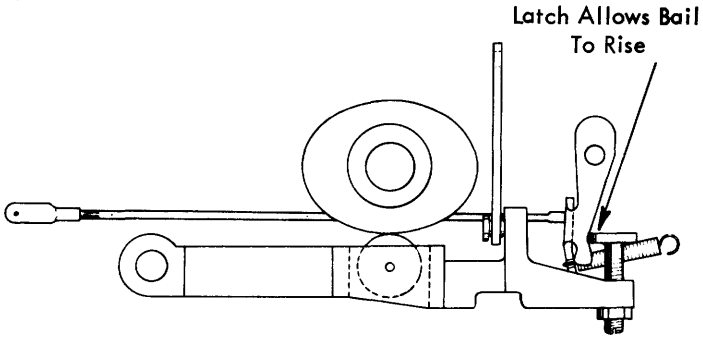


FIGURE 32. Five Unit Bail During Negative Rotate Cycle

The five-unit bail is restored down by the extreme right hand cam on the cycle shaft. The high point of the cam is 90° from the high point of the other two cams. This insures that when the latch bail is driven DOWN in the active position, the five-unit bail can be UP in the active position. Conversely, when the latch bail is UP in the rest position, the five-unit bail will be DOWN in the rest position.

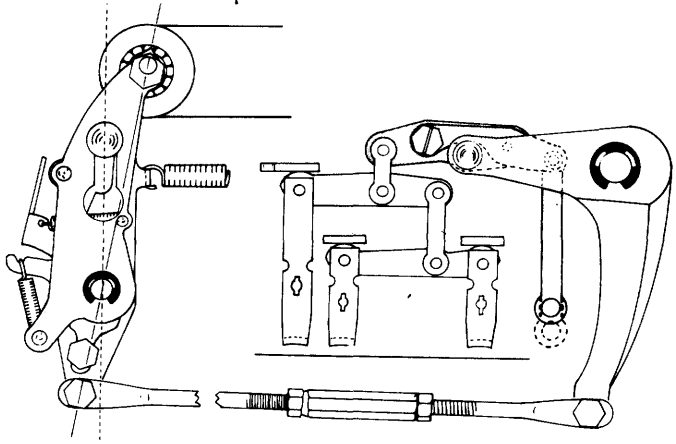


FIGURE 33. Negative-Five Rotate Operation

It should be noted that the five-unit bail is held down in the rest position by the cam rather than by the five-unit latch. In the rest position there is a clearance between the latch and the adjusting screw in the five-unit bail. The clearance must be present to.insure resetting of the latch at the completion of a cycle. During a positive rotate or non-rotate cycle, the five-unit bail rises slightly before being restricted by the five-unit latch (Fig. 31). This upward movement of the five-unit bail allows a slight clockwise, or negative rotation of the type head. The type head rotates from the rest position to a position known as the "latched-home" position. Consider all positive and negative rotations to occur from the latched-home position.

Movement of the five-unit bail from the latched-home point to the low point of the cam allows sufficient clockwise movement of the rotate bellcrank to permit a five character negative rotation of the typehead (Fig. 33). If less than five units of negative rotation is desired, it is necessary to pull down on the left end of the balance lever as the right end goes up. This reduces the amount of clockwise movement of the rotate bellcrank. Operating one or more positive rotate latches down in conjunction with allowing the five-unit bail to rise allows different amounts of negative rotation. The positive rotate-one and negative-five combine to allow a negative-four rotation. A positive-two and negative-tive combination gives a negative-three rotation (Fig. 34). A positive-three and a negative-five operation permits a negative-two rotation. A positive-four plus a negative-five combination gives a negative-one rotation.

KEYBOARD SECTION AND CHARACTER SELECTION

The keyboard section is a compact unit that is detachable as a unit from the rest of the machine. Contained in the keyboard section are all keylevers and allied parts, and a selection mechanism for the differential latches. Depression of a letter keylever prepares the selection mechanism for operation and trips the cycle clutch latch to allow a cycle operation.

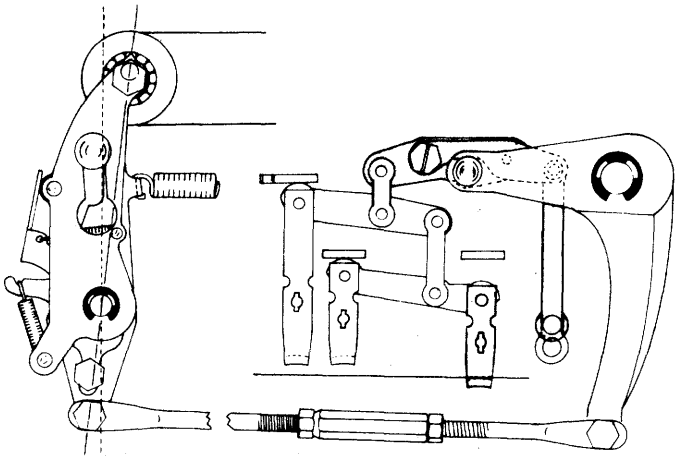


FIGURE 34. Negative-Three Rotate Operation

Keylevers

The keylevers pivot on a fulcrum rod at the rear (Fig. 35). A nylon rod at the top and the bottom of the guide comb slots limit the travel of the keylevers in the front guide comb. A lower extension on each keylever operates in the keylever bearing support to stabilize the keylever.

The keybuttons are designed in the shape of a pyramid to make the use of a keyplate unnecessary. The tops of the keybuttons form a concave slope to the keyboard for ease of operation. Keylever tension is supplied by a set of flat spring fingers under the front of the keylevers. The forward end of each spring finger is cupped so that the spring will maintain its position under the keylever. Different spring tension is supplied to the four rows of letter keylevers by auxiliary leaf springs under the keylever springs. The auxiliary spring fingers vary in length to offset the leverage difference among the four rows of keylevers. This variation in spring tension results in a uniform operating force requirement for all keylevers.

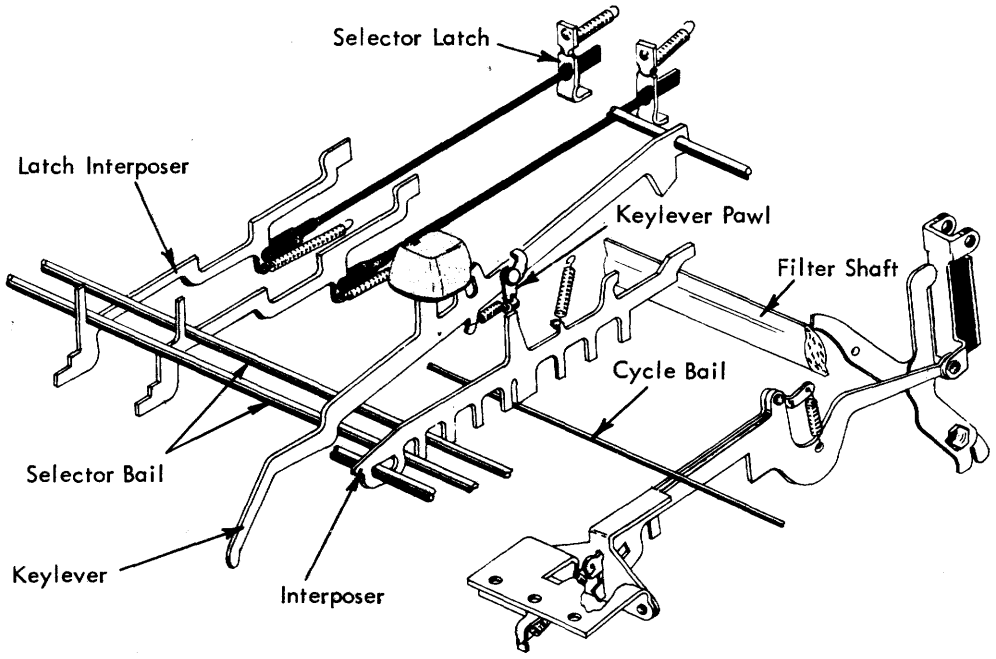


FIGURE 35. Keyboard Section and Character Selection

A shoulder rivet attaches a keylever pawl to each keylever at the rear. A small spring attached from the pawl to the keylever holds the pawl in the rest position. The pawl extends below and is formed under the keylever in position to strike the top of an interposer (Fig. 35).

Adjusting lugs at the rear of the keylevers make it possible to adjust the relative height of each individual keylever and keylever pawl.

Interposers

Each keylever has a character interposer located just below it (Fig. 35). The purpose of the interposers is to select the amount of tilt and rotate needed to bring the desired character to the printing point. A large fulcrum rod passes through an elongated hole in the front of each interposer and provides a support on which the interposers can pivot and slide. A guide comb at the front and rear separates the interposers. The interposers are allowed to move up and down in the rear guide comb as well as front to rear. An extension spring from each interposer to the top of the rear guide comb loads the interposer to the rear and up into the rest position.

The interposers have several lugs extending from them, each with its own operation to perform (Fig. 36). Each interposer has a lug on top in position to be struck by the keylever pawl. On the bottom of the interposers are positions for eight lugs. Seven of the lugs are selective lugs. The absence or presence of these lugs in different position combinations makes the interposers different. There are no two alike. The rearmost selective lug is for special applications of the machine.

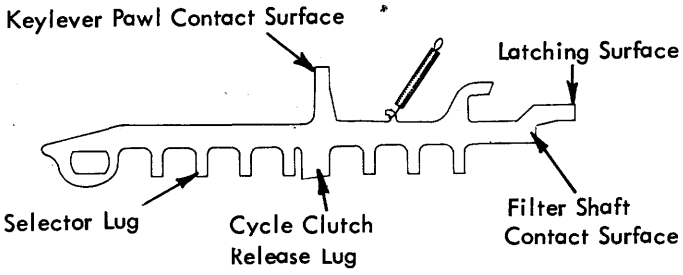


FIGURE 36. Selector Interposer

One lug on the bottom is common to all interposers. It is a wide lug located at the middle of the interposers. Its purpose is to release the cycle clutch for a cycle operation whenever a keylever is depressed. Directly below the lug is the cycle bail that pivots up and down (Fig. 35). Downward movement of any character interposer forces down on the cycle bail to release the cycle clutch latch and allow a cycle operation. The cycle clutch release lug is cut at an angle on the bottom to prevent interference between the lug and the cycle bail as the parts are restoring to the rest position.

Filter Shaft

When the interposer is depressed it pushes the cycle bail down to unlatch the cycle clutch and allow a 180° rotation of the cycle shaft (Fig. 35). Rotation of the cycle shaft 180° also rotates the filter shaft 180°. The filter shaft is a two-bladed shaft located at the rear of the keyboard section and just below the ends of the interposers. It pivots in a bronze bearing at each end and is connected by a gear train on the left side of the cycle shaft.

When an interposer is depressed, the rear of the interposer is moved down in front of one blade of the filter shaft. As the filter shaft turns, the blade drives the interposer forward to operate the character selection mechanism.

Interposer Latch

Resting against the rear of each interposer is a spring finger called the interposer latch spring (Fig. 37). The spring fingers are slightly deflected to the rear when the interposers are at rest. When any interposer moves down at the rear, the spring snaps forward over the interposer and holds it down. The interposer remains down until it is pushed forward enough to clear the spring finger. At that time it is raised and restored by its extension spring.

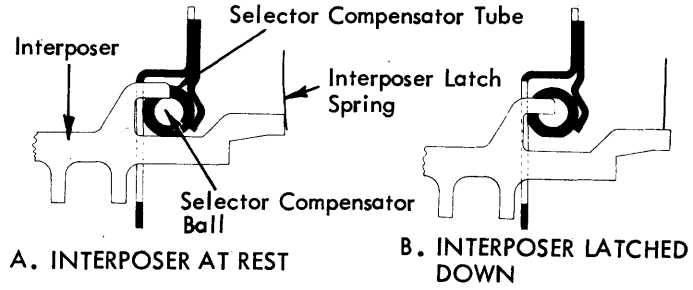


FIGURE 37. Interposer Latch and Selector Compensator

An interposer must be latched down to insure that it will remain in the path of the filter shaft blade until the cycle operation occurs. Unless latched down, the interposer could restore upward without being driven forward. This would result in an erroneous character because of no character selection.

With this latching device, one interposer can be latched down just as the previously depressed interposer is being operated forward. The second interposer will then be operated forward as soon as the cycle operation for the first interposer is completed. This is known as character storage and tends to even erratic typing rhythm.

Selector Compensator

The selector compensator prevents simultaneous depression of two keylevers insuring that only one interposer at a time can be operated down and then forward. If more than one interposer at a time were depressed, they would be operated forward together and a selection error would result causing the wrong character to print. A hook shaped lug at the rear of each interposer operates in a device called the selector compensator attached to the rear interposer guide comb (Fig. 37). The compensator contains closely spaced steel balls that prevent downward movement of two or more interposers simultaneously (Fig. 38a). When an interposer is down, the steel balls shift in the tube of the compensator to block the downward movement of any other interposer (Fig. 38b).

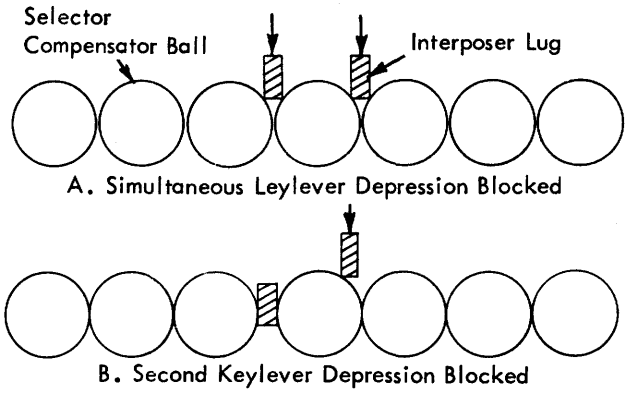


FIGURE 38. Selector Compensator Action

The hook portion of the interposer lugs extends to the rear just to the center line of the steel balls (Fig. 37). When an interposer is driven forward after being depressed at the rear, the lug will not be between the steel balls and another interposer can be depressed. With this arrangement, the operator does not have to wait until an interposer is restored upward before actuating another.

An adjustable stop at each end of the compensator tube keeps the balls somewhat centered between the interposers. The balls are thus prevented from shifting too far left or right. If the balls were allowed to move too far under the interposer lugs, they would partially block the depression of an interposer and the keyboard touch would suffer. The stops are removable so that the steel balls may be removed and cleaned if necessary.

Selector Bails and Links

Six selector bails are mounted between the side frames so they can operate forward and back (Fig. 35). Each bail is in front of a selector lug position on the interposers. When an interposer containing all its lugs is driven forward, all the selector bails are moved forward. If any lug is absent from an interposer, the bail for that position will not be operated forward by that interposer.

Six sliding interposers called the latch interposers are located under the left end of the selector bails (Fig. 35). These interposers operate front to rear in the bottom of the interposer guide comb.

Each latch interposer has a lug that extends up directly in front of a selector bail. As the selector bail is moved forward, the latch interposer is moved forward by the bail. The bails and latch interposers are paired so that each bail operates only one interposer. An extension spring at the bottom of each latch interposer loads the interposer and its selector bail to the rear. The latch interposers rest against adjustable lugs at the rear to prevent the bails from being forced against the selector interposer lugs. This prevents the selector interposers from binding against the bails as the interposers are depressed. The stop lugs also help prevent the selector bails from bouncing as they are restored to the rear.

An adjustable link connects each latch interposer to one of the selector latches of the differential mechanism (Fig. 35). When a latch interposer is moved forward, the selector latch connected to that interposer is pulled forward to prevent its being operated downward by the latch bail.

Cycle Clutch Latch

Although not actually a part of the keyboard section, the cycle clutch latch is directly related to the keyboard mechanisms. It is through depression of a keylever that the cycle clutch is allowed to operate.

The cycle clutch latch pivots on a bracket at the front of the power frame. It pivots from the top and rests in a vertical position just in front of the cycle clutch sleeve. A thin metal plate, mounted in rubber on the rear of the latch prevents rotation of the sleeve thereby preventing the cycle clutch spring from tightening on the cycle clutch pulley hub (Fig. 39). The latch is held in this position by the cycle clutch latch pawl and link assembly that extends forward from the latch. The cycle clutch pawl pivots on the link. An extension spring between the two parts rotates the front of the pawl up into a latched position behind the cycle clutch keeper (Fig. 39). The cycle clutch keeper is an adjustable plate mounted, by means of the cycle clutch keeper bracket, to the guide comb support under the front of the keyboard section. An extension spring exerts a pull toward the front on the cycle clutch latch link. The pawl, being attached to the link, prevents the link from being pulled forward.

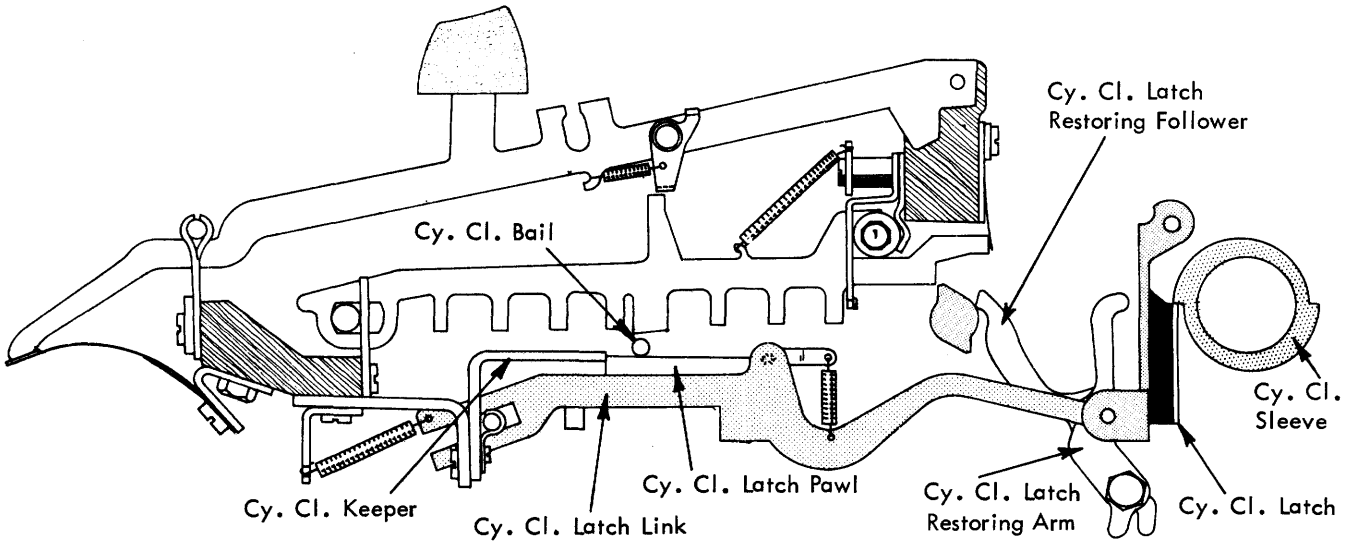


FIGURE 39. Cycle Clutch Latch Mechanism

When the keylever is depressed, the interposer beneath the keylever forces the cycle bail to pivot downward. The cycle bail moves the cycle clutch latch pawl down disengaging it from the keeper. The extension spring at the front of the link is then allowed to snap the link and cycle clutch latch forward disengaging the latch from the clutch sleeve. This allows the clutch spring to tighten and begin a cycle operation.

A small lever, called the cycle bail damper, pivots at each side of the keyboard just above the cycle bail (Fig. 40). An extension of each lever rests against the front of the bail. An extension spring connected between the cycle bail and each damper has the dual purpose of restoring the cycle bail upward and holding the damper against the bail. The purpose of the dampers is to lightly retard the upward movement of the cycle bail so as to prevent the bail from bouncing as it reaches its upward limit. Without the dampers the bail has a tendency to bounce down and retrip the cycle clutch creating an additional cycle.

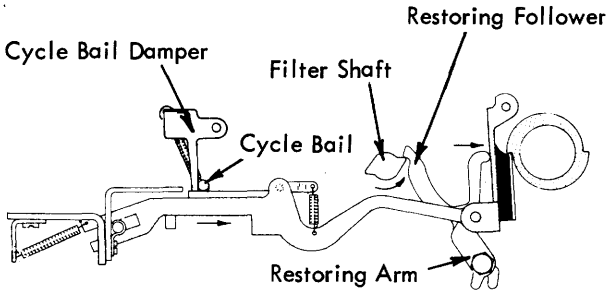


FIGURE 40. Cycle Clutch Latch Restoring Operation (old style)

Old Style Restoring Mechanism

For a cycle operation, the cycle shaft must be limited to 180° rotation. This means that the cycle clutch latch must be restored and held to the rear into the path of the second step on the cycle clutch sleeve. A two-piece adjustable restoring device pivots on the cycle clutch latch bracket. As the filter shaft rotates during a cycle operation, a blade of the filter shaft forces the restoring follower to the rear (Fig. 40). The follower and restoring arm acts as one piece; thus the restoring arm moves to the rear and pushes the cycle clutch latch back into position to stop the cycle clutch sleeve. The latch is restored far enough to the rear to permit the cycle clutch latch pawl to reset on the cycle clutch keeper and maintain the parts in the rest position.

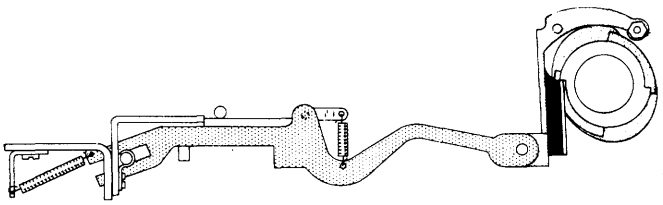


FIGURE 41. Cycle Clutch Latch Restoring Operation (new style)

New Style Restoring Mechanism

The new style cycle clutch latch restoring mechanism consists of a cam and a cam follower. The cam, called the cycle clutch latch restoring cam, is a double lobed nylon cam mounted on the cycle shaft (Fig. 41). The cam follower is an extension of the cycle clutch latch which protrudes to the rear of the machine just above the restoring cam. The extension has a small adjustable steel roller mounted on it which rides on the cam during a restoring operation.

When the machine is at rest the low point of the restoring cam is directly below (but not in contact with) the steel roller (Fig. 41). When a keylever is depressed the cycle clutch latch pawl is pushed off its keeper allowing the cycle clutch latch to swing forward releasing the cycle clutch sleeve. When the cycle clutch latch swings forward the steel roller on the extension drops down onto the restoring cam. As the cam rotates towards its high point the steel roller is forced up swinging the cycle clutch latch to the rear into the path of the cycle clutch sleeve. The latch is restored far enough to the rear to permit the cycle clutch latch pawl to reset on its keeper.

Operation

Depression of a letter keybutton causes the front of the keylever to move down as the rear end pivots about the fulcrum rod. The keylever pawl at the rear of the keylever contacts the top lug of an interposer. Further movement of the keylever causes the rear of the interposer to move down as the front pivots about the interposer fulcrum rod. As the interposer moves down, a lug on the bottom of the interposer forces the cycle bail and the cycle clutch latch pawl down. Further movement of the interposer allows the interposer latch spring to snap forward over the top of the interposer to maintain its downward position. At about the same time that the interposer latches down, the cycle bail trips the cycle clutch latch pawl off its keeper to allow the cycle clutch to begin an operation.

As the cycle shaft turns, the cams of the cycle shaft force the latch bail down. However, the contour of the cams is designed so that cycle shaft rotation does not cause immediate downward movement of the latch bail. During the "dwell" on the cycle shaft cams, the filter shaft is operated to drive the depressed interposer forward.

As the interposer is driven forward, the selector lugs that are present on the interposer push their respective selector bails forward. The bails cause the latch interposers to pull the selector latches forward.

As the selector mechanism is being operated forward, the cycle shaft cams start to force the latch bail down. All latches that remain to the rear are operated down by the latch bail. This does not apply to the negative-5 latch which must be pulled forward in order to allow a negative rotation.

The latches are pulled forward only for an instant. As soon as the interposer has been pushed forward far enough to clear its interposer latch spring, it becomes free to restore vertically. The fact that the interposer cannot move up instantly out of the path of the filter shaft allows it to be driven farther forward. The interposer spring then raises the interposer and restores it to the rear. This allows the latches to restore to the rear. By this

time the latch bail will have been forced down by the cycle shaft cams far enough to prevent resetting of the latches under the bail plate. The latches merely rest against the bail plate until the cycle operation has been completed and the latch bail has restored. The same action applies to the negative-5 latch except that its bail is allowed to rise in the operated position. The negative-5 latch resets above its bail when the bail has been driven down to its rest position.

If the keylever has been held down throughout the operation, the rear edge of the interposer lug strikes the keylever pawl as the interposer restores to the rear. The keylever pawl is then deflected to the rear and remains in this position until the keylever is released (Fig. 42). At that time it snaps forward above the interposer lug ready for the next operation. This arrangement insures a single operation regardless of how long the keylever is held depressed by the operator.

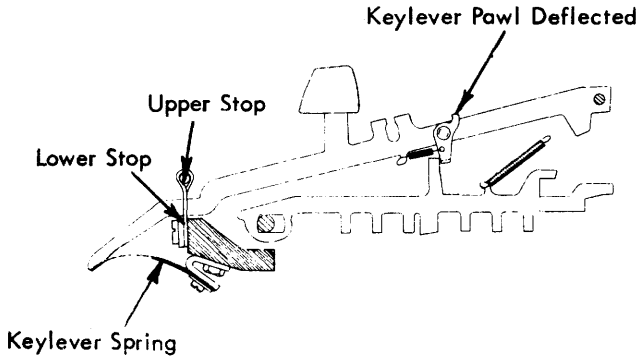


FIGURE 42. Keylever Held Depressed

Repeat/Non-Repeat Keylevers

A repeat/non-repeat operation is provided as a standard feature in the hyphen/underscore position. The operation requires a special, two-piece keylever (Fig. 43). One part of the keylever pivots about the keylever fulcrum rod at the rear and extends forward through the guide comb. It contains no keybutton but has the keylever pawl attached similar to the normal keylever. The second part of the keylever is a short lever containing the keybutton. It pivots on a shouldered rivet at the rear of the long keylever. The front of the short lever is restricted by a shouldered rivet through an elongated hole. A compression spring between the two pieces acts to hold the short lever up in its elongated hole.

When the keybutton is depressed, the two pieces act as one and a single operation results. If the keylever is held down, the keylever pawl is deflected to the rear as the interposer restores (Fig. 43). Additional pressure applied to the keybutton overcomes the compression spring and causes the short lever to pivot downward. A special lug on the bottom of the short lever forces the interposer down for a second operation (Fig. 44). The special lug is wide enough so that the interposer is not allowed to restore upward as long as the keylever is held in the repeat position. The interposer merely travels front to rear. As it does so, it prevents the cycle clutch latch pawl from resetting, and results in a continuous operation.

The repeat/non-repeat feature may be removed from the machine by replacing the two-piece keylever with a conventional keylever.

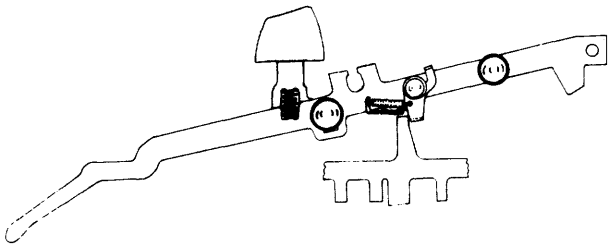


FIGURE 43. Repeat Keylever After Single Operation

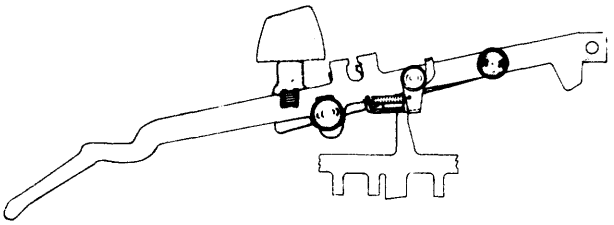


FIGURE 44. Repear Keylever Held Depressed

Keyboard Lock

When the switch is turned OFF, the keyboard must be locked to prevent mechanisms from being tripped while the motor is not running. This is to prevent the motor from having to start under a load and to prevent an unwanted operation the next time the switch is turned ON.

Operation of the switch lever controls the keyboard lock mechanism by rotating the lockout bail that extends across the bottom of the keyboard section (Fig. 47). When the switch lever is in the OFF position, the lockout bail is moved forward into position below an extension of the cycle clutch latch pawl, and prevents the cycle clutch from being released.

As additional insurance against an interposer being latched down, a special bellcrank at the left side of the keyboard is rotated into the selector compensator by a link from the lockout bail. This forces the steel balls to shift in the tube and block the downward movement of all interposers. When the switch is ON, the keyboard lock bellcrank is spring-loaded out of the selector compensator.

A link from the right side of the lockout bail rotates a D-shaped shaft beneath the operational mechanism. The shaft locks the backspace, spacebar, carrier return and indexing keylevers when the switch is OFF. The tab and shift keylevers are not locked.

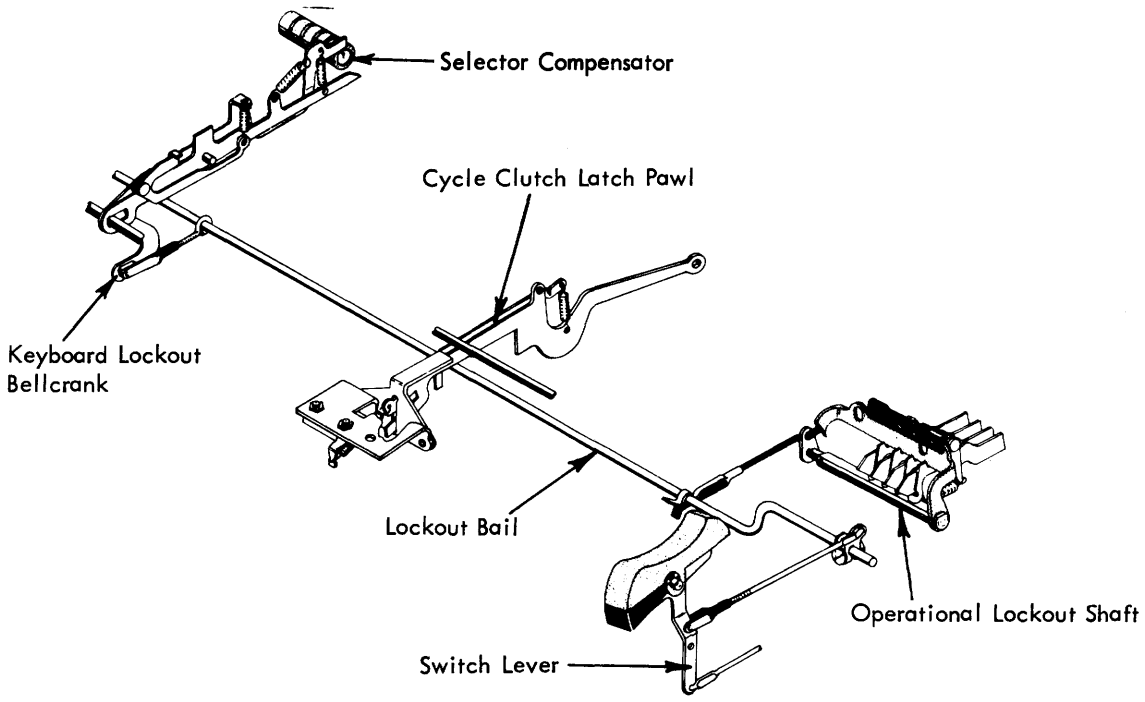


FIGURE 45. Keyboard Lock Mechanism - ON Position

SHIFT

The purpose of the shift mechanism is to rotate the type head 180° in the counterclockwise direction. This action places the upper case hemisphere of the type head near the platen for typing capital letters. Each upper case character is in the same tilt band as its lower case counterpart but 180° from it. Thus depression of a keylever with the shift in operation causes an upper case character to print.

The shift mechanism consists of a shift arm, shift cam, spring clutch, clutch control mechanism and interlocks. The shift mechanism takes its power from the right end of the operational shaft. All the components are concentrated in that area.

Shift Operation

The shift operates by moving the right hand rotate pulley toward the right (Fig. 46). The movement of the pulley creates sufficient pull on the rotate tape to cause 180° type head rotation. The pulley remains in this position as long as the shift keylever is held depressed.

Two keybuttons, one at each front corner of the keyboard, can be used to actuate the shift mechanism (Fig. 47). A bail between the two keylevers causes both of them to move together regardless of which one is depressed by the operator. If the operator desires to keep the mechanism in the upper case position, a shift lock is provided for this purpose. The shift lock is attached to the left keylever and may be locked by depressing the shift lock keybutton. The shift lock may be released by depressing and releasing either shift keybutton.

The right hand rotate pulley is fastened to the top of the shift arm. The arm pivots left to right on a pin at the bottom. A strap from the shift arm to the pivot pin acts to stabilize the shift arm to minimize front to rear movement of the pulley (Fig. 48). In the lower case position, an adjusting screw near the top of the arm rests against the head of a mounting screw on the side of the powerframe.

The shift arm must be forced to pivot outward for a shift operation. A disc-shaped shift cam fits around an extension of the operational shaft bearing outside the sideframe. The cam operates between two rollers located at the rear of the cam (Fig. 48). The roller to the left of the cam is in a fixed position on the powerframe and serves as a back-up roller for the cam. The roller on the right is attached to the shift arm and rides the camming surface of the cam. The camming surface is on the right side of the cam rather than on the perimeter. When the low point of the cam is between the two rollers, the shift arm is in lower case (Fig. 48a). The cam is operated 180° to the high point in order to force the arm into upper case (Fig. 48b).

The shift cam rotates only during a shift operation and receives its motion from the operational shaft. Since the operational shaft turns continuously when the motor is running, a clutch mechanism is required to engage and disengage the shift cam from the operational shaft when a shift operation is desired. A spring clutch "makes" and "breaks" the driving connection between the operational shaft and the shift cam.

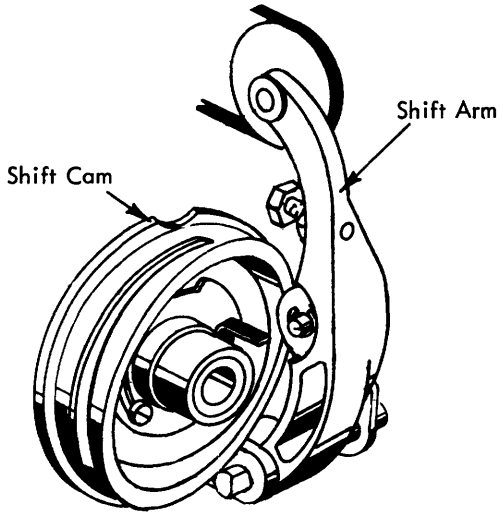


FIGURE 46. Shift Cam and Shift Arm

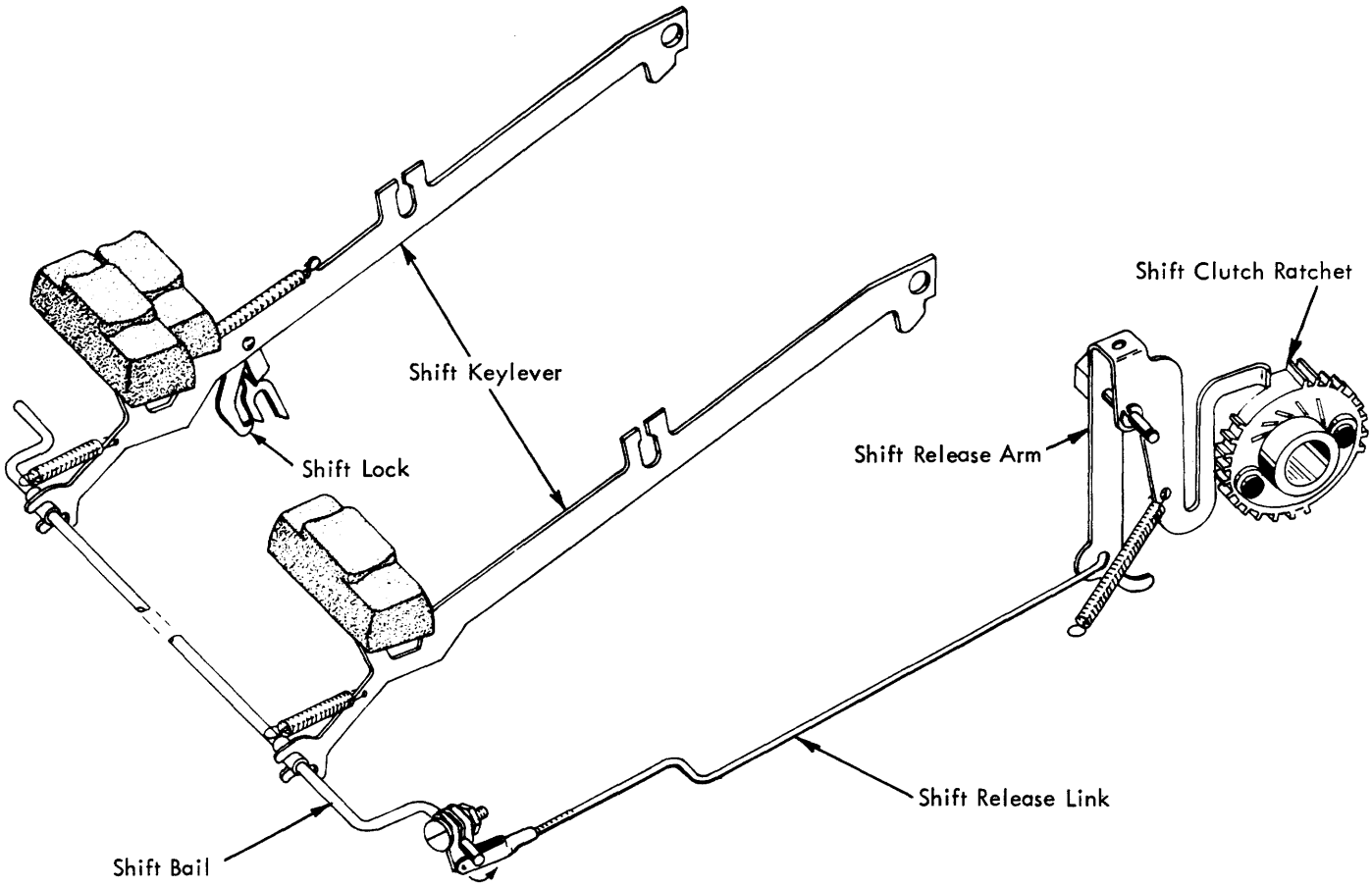
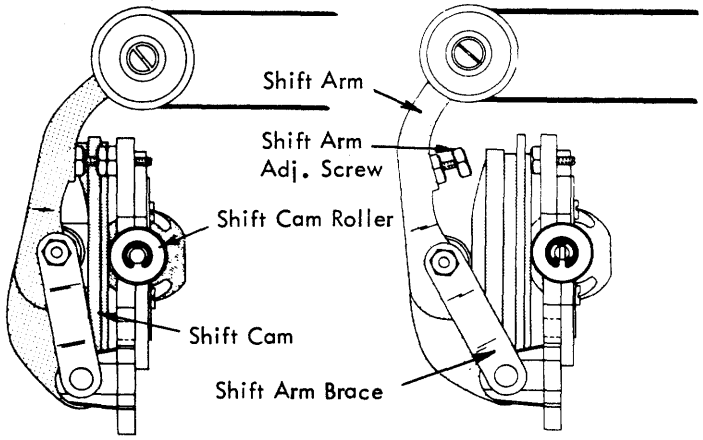


FIGURE 47. Shift Release Mechanism

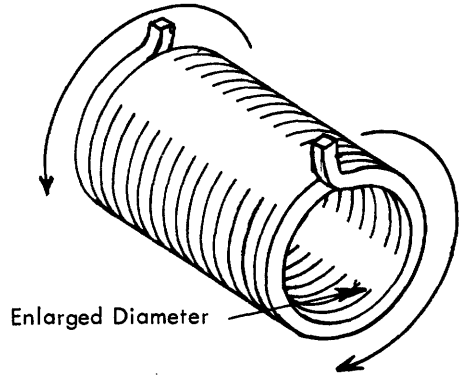
The shift cam mounts on a shoulder on the right hand side of the operational shaft bearing and is free to rotate about this shoulder (Fig. 49). The operational shaft extends through the center of this bearing beyond the right side of the shift cam. An arbor (shift clutch arbor) is set screwed to the shaft just to the right of the shift cam and turns with the shaft continuously (Fig. 49). The driving force to the shift cam comes from this arbor by means of a spring clutch that is fastened to the cam and encircles the arbor. The spring clutch is undersized and wound in the same direction that the arbor is turning. Therefore, anytime that the spring clutch is allowed to collapse about the arbor it will turn with the arbor driving the cam. The cam is driven 180° each time a shift operation occurs.



A. LOWER CASE POSITION B. UPPER CASE POSITION

FIGURE 48. Shift Operation - Rear View

Usually a spring clutch is considered to be one that tightens when its driving hub turns in the direction that the spring clutch is wound and slips if the hub stops or rotates back in the opposite direction. This is not true of all spring clutches. In the shift mechanism the driving hub (the shift clutch arbor) is rotating continuously in the direction that the spring clutch is wound. In order to stop the shift cam, the spring clutch must be allowed to slip while the arbor continues to rotate. The only way to do this is to enlarge or expand the inside diameter of the spring clutch so that the driving action will stop.



Spring Clutch - Unwinding Direction

In the shift mechanism, the shift spring clutch must be held in the unwound position so that it will be enlarged enough to allow the arbor to slip freely inside it. To enlarge the spring clutch, one end of the spring must be held while the other is rotated in the unwinding direction of the spring. The left

end of the spring is formed out and fits into an adjustable plate attached to the cam (Fig. 49). The right end of the spring is formed to the right and fits into a hole in the shift clutch ratchet (Fig. 49). The shift clutch ratchet is a gear-like part mounted on the end of the shift clutch arbor. A large C-clip holds it in place. The arbor turns freely inside the ratchet when the shift is not being operated. Rotation of the ratchet counterclockwise causes the clutch spring to decrease in size. Conversely, if the ratchet is held stationary while the cam is rotated counterclockwise, the spring diameter is increased.

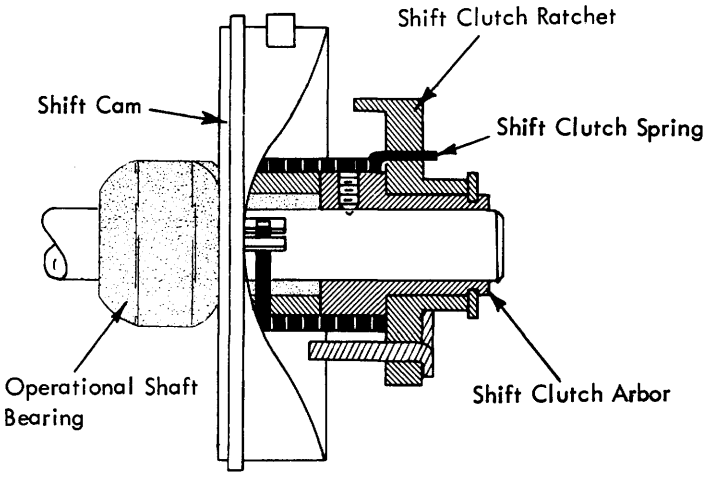


FIGURE 49. Shift Clutch

The shift clutch ratchet has two lugs 180° apart on its left surface. One lug is nearer the center than the other and is referred to as the inner lug. The other lug is called the outer lug. The shift release arm pivoted just in front of the cam blocks the movement of the ratchet lugs to stop the rotation of the ratchet (Fig. 50).

The position of the shift release arm determines the lug of the ratchet that will be stopped. The position of the shift-release arm is controlled by a link from the arm to a lever attached to the end of the shift bail (Fig. 47). When a shift keylever is depressed, the bail rotates to force the link to the rear and operate the shift release arm. When the keylevers are at rest, the release arm is in position to stop the inner lug of the shift clutch ratchet (Fig. 48a). Depression of a keylever causes the release arm to rise out of the path of the inner lug into the path of the outer lug (Fig. 48b).

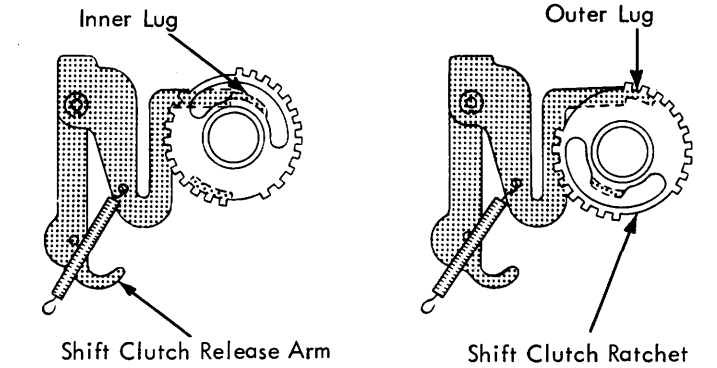


FIGURE 50. Shift Clutch Release Arm

When the inner lug is released, the clutch spring is allowed to decrease in diameter by its own spring tension. It then tightens around the rotating shift clutch arbor and the hub of the cam to drive the cam. The shift clutch ratchet rotates with the cam and clutch spring 180°. The outer lug of the ratchet then contacts the shift release arm, that has been raised into its path, and its rotation is stopped. The momentum of the shift cam causes it to rotate slightly farther. This further rotation causes the clutch spring to unwind and increase in diameter to allow the shift clutch arbor to slip freely inside the spring.

Once the shift cams momentum has carried the cam far enough to properly disengage the spring clutch the cam must be restricted from overthrowing its rest position. Overthrow is controlled by the shift cam stop which is attached to the cam and operates against the inner lug of the shift clutch ratchet (Fig. 51).

As long as the shift keylever is held depressed, the high point of the shift cam remains to the rear holding the shift arm to the right. When the keylever is released, the shift release arm moves down out of the path of the outer lug and back into the path of the inner lug. This allows the clutch spring to again tighten and drive the shift cam and ratchet 180° back to the lower case position. The inner lug of the ratchet then contacts the release arm to disengage the spring clutch as before. The shift cam stop again controls overthrow of the cam.

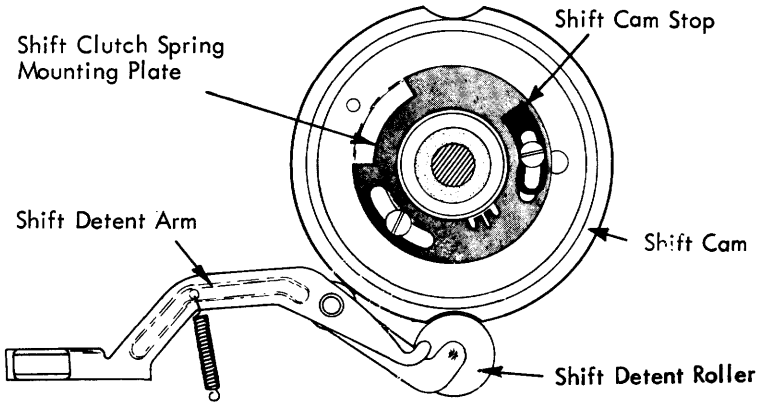


FIGURE 51. Shift Arm and Detent

Note that the shift cam stop always follows behind the inner lug of the ratchet when the ratchet and cam are operating. Whenever the ratchet is stopped by the release arm (on either lug) the momentum of the cam carries the stop further until the stop strikes the rear side of the inner lug. The horse shoe shaped release arm absorbs the shock of stopping the overthrow of the shift cam through the shift clutch ratchet.

Mounted on a pivot stud below and to the front of the shift cam is a small arm called the shift detent arm (Fig. 51). Although the detent arms primary function is to operate the character interrupter mechanism, it is also used as a detent for the shift cam. As cam approaches either the upper or lower case position a nylon roller mounted on the shift detent arm is spring loaded into corresponding detent notches located on the outside surface of the cam. This detenting action helps to place the cam in its proper rest position for both upper and lower case.

Cam overthrow is a greater problem in returning to the lower case than in shifting to upper case. As the cam returns to the lower case position, the roller of the shift arm is rolling from the high point to the low point of the cam. The pressure of the roller against the receding surface tends to accelerate the movement of the cam. Excessive speed of the cam could cause a noisy operation and parts breakage. To prevent cam acceleration, a braking action is necessary the same as an automobile must be braked in descending a hill. A heavy spring steel shift arm brake operates just in front of the shift cam. A nylon brake shoe, attached to the end of the brake, rides a raised surface on the circumference of the cam (Fig. 52). The friction applied by the brake prevents acceleration of the cam and causes the shift action to be the same for both upper and lower case.

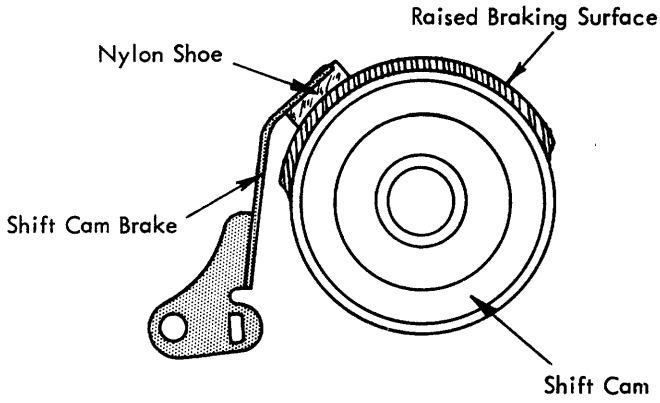


FIGURE 52. Shift Cam Brake

Character Interrupter

If the shift is already in process when a letter key is depressed, the character must be delayed until the shift is completed. Otherwise the shift would be interrupted in mid-operation and an erroneous character would result. The character is delayed by blocking the cycle clutch release during a shift operation. When the shift cam starts an operation, the detent must move out of the recess in the cam. A forward extension of the detent lever operates a bail under the right side of the keyboard (Fig. 53). A small pawl on the left end of the bail is rotated up into the path of the cycle clutch latch link. The pawl blocks the forward movement of the link and prevents the release of the cycle clutch. This interlocking action does not prevent the depression of the keylever nor the interposer. The interposer is merely latched down into storage. When the shift action is completed, the detent enters the recess in the cam. The bail is rotated back to the rest position and the pawl moves out of the path of the cycle clutch latch link. The cycle clutch latch link is then pulled forward by its extension spring and the stored character is printed.

If the shift keylever and a character keylever are depressed simultaneously, both will operate for an instant until the filter shaft has a chance to actuate the shift interlock. During that time a collision will occur between the character interrupter pawl and the bottom of the cycle clutch latch link. The character interrupter mechanism must yield in order to prevent parts damage. A torsion spring around the character interrupter bail loads the interrupter pawl against an adjustable stop on the bail. If a collision occurs, the interrupter pawl can yield by overcoming the torsion spring as the interrupter bail rotates.

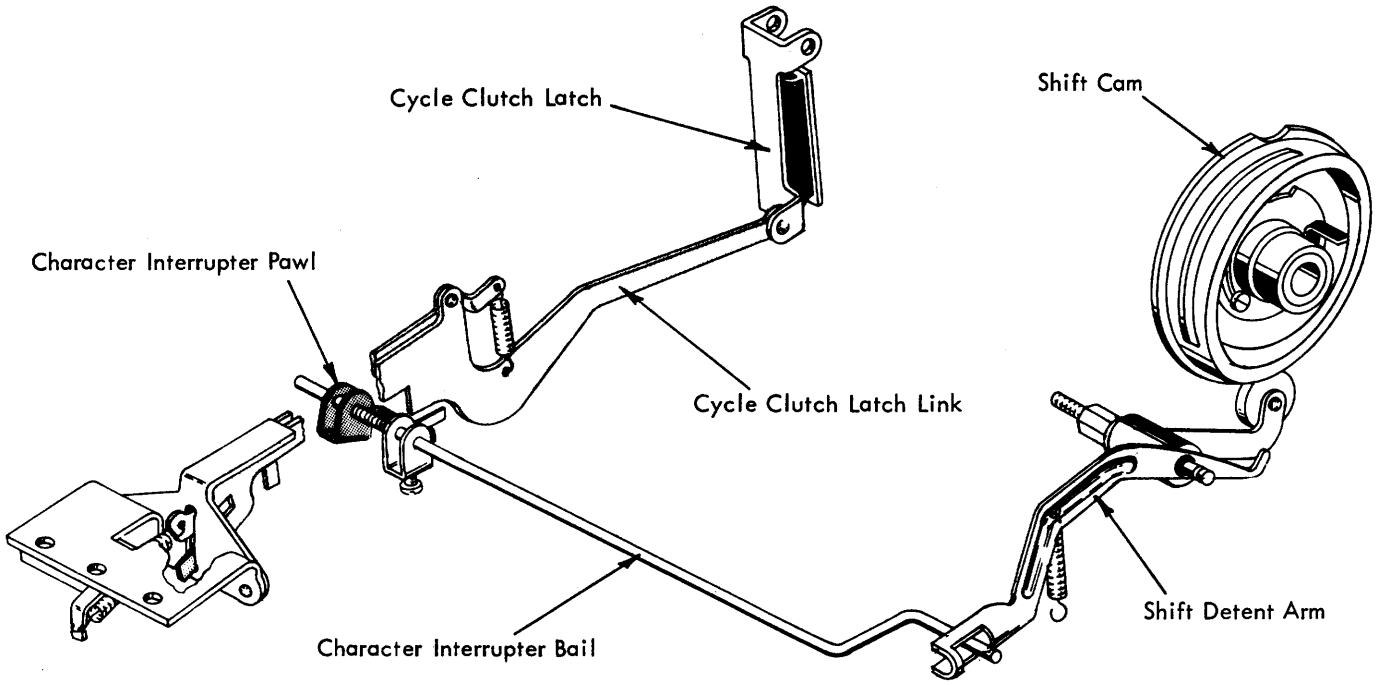


FIGURE 53. Character Interrupter

Shift Interlock

Operating the shift mechanism when the type head is in the process of printing would result in parts damage. The rotate detent would be engaged in a notch of the type head and the type head could be against the platen. At this time no rotation of the type head can be allowed; therefore the shift must be prevented from operating once the type head has started toward the platen.

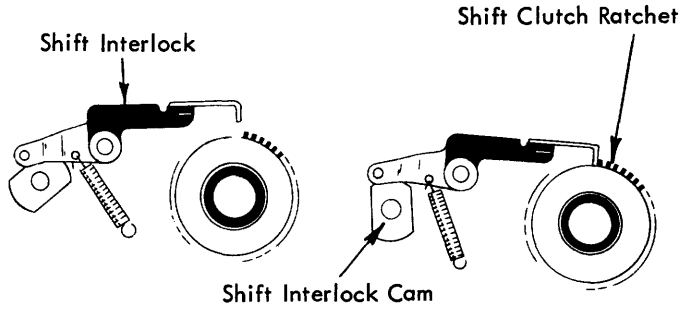


FIGURE 54. Shift Interlock

We have seen that the shift spring clutch remains disengaged as long as the shift clutch ratchet is prevented from rotating. An interlock arm is operated by a cam on the right end of the filter shaft (Fig. 54). The interlock engages the teeth of the shift clutch ratchet and prevents rotation of the ratchet. When the cycle mechanism is at rest, a roller on the interlock rests near the low point of the interlock cam allowing free operation of the shift mechanism (Fig. 54a). As soon as a cycle operation begins, the filter shaft rotates causing the interlock cam to actuate the interlock into the teeth of the shift clutch ratchet (Fig. 54b). This interrupts the shift operation until the cycle operation is completed.

If an operator should operate the shift immediately after striking a character, the shift cam could begin to rotate before the filter shaft had sufficient time to actuate the shift interlock. This could cause an erroneous character to print because the shift arm had already begun to move. This is known as "beating the shift". This condition occurs mostly in shifting from upper to lower case. Shifting from lower to upper case is no problem because the shift arm does not rest against the cam in lower case. The cam must rotate somewhat before it begins to move the shift arm, thereby allowing the filter shaft sufficient time to actuate the shift interlock.

To overcome the problem of "beating the shift" coming out of upper case a redesigned shift cam has been incorporated in the Selectric. This redesigned cam has a longer upper case dwell than the former style cam. The longer dwell allows the filter shaft sufficient time to operate the shift interlock before the shift arm begins to move, thus overcoming the problem.

The new style shift cam requires a different shift clutch ratchet because the cam now rotates 220° going from upper to lower case and 140° going from lower to upper case. With this longer cam rotation when shifting from upper to lower case, the detent notch in the cam (that operates the character interrupter) has been elongated by 40°. This elongation of the notch allows a character to come out of storage a short period of time before the shift operation has completed. This can be done without any ill effects because under dynamic conditions there is a time delay between cycle clutch release and cycle shaft operation. Without early storage release (going to lower case) the operator's typing rhythm could be affected.

CYCLE CLUTCH OPERATION

The cycle operation occurs each time a character prints. Everything concerned with printing a character on the paper is powered by the cycle shaft either directly or indirectly.

The cycle shaft extends from the center of the machine out through the left side frame. The left end of the shaft is supported by a self-aligning porous bronze bearing. The right end fits into the cycle clutch pulley hub in the center of the machine. A bronze sleeve inside the hub acts as a bearing for the cycle shaft.

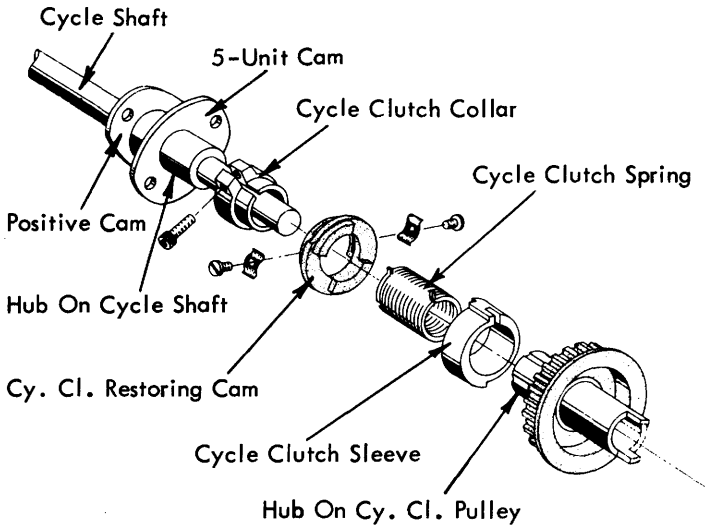


FIGURE 55. Cycle Clutch - Exploded View

The cycle clutch pulley is in continuous rotation whenever the motor is running, but the cycle shaft operates only during a print operation. A spring clutch called the cycle clutch spring (Fig. 55) is the driving connection between the hub on the cycle clutch pulley and a hub on the cycle shaft. The cycle clutch spring provides a means of engaging and disengaging the cycle shaft from the cycle clutch pulley. The cycle clutch spring operates exactly the same as the spring clutch in the shift mechanism.

The left end of the cycle clutch spring fits around the hub on the cycle shaft and is clamped to this hub by the cycle clutch collar. The tip of the spring is turned up so as to fit into a slot in the collar. This arrangement prevents any slippage at the left end of the spring clutch and makes it possible to adjust the position of the spring in relation to the shaft. (The collar does exactly the same job as the spring clutch retaining plate on the shift cam.)

The right end of the cycle clutch spring encircles a hub on the cycle clutch pulley (Fig. 55). (The hub on the cycle clutch pulley functions the same as the shift clutch arbor on the shift mechanism.) The inside diameter of the cycle clutch spring is slightly less than the diameter of the hub on the cycle clutch pulley so that the spring clutch will tighten when the hub ro-

tates. The right hand tip of the spring clutch projects into a notch in the cycle clutch sleeve. The sleeve fits loosely around, enclosing the spring clutch and acts as a control for the right end of the cycle clutch spring. (The sleeve performs the same function as the shift clutch ratchet.)

The cycle clutch sleeve has two steps on its exterior surface 180° apart. As the cycle clutch is driving the cycle shaft, one of the steps on the clutch sleeve contacts a vertical latch that is placed in the path of the sleeve (Fig. 56). This latch, called the cycle clutch latch, stops the rotation of the clutch sleeve thereby stopping the right end of the cycle clutch spring. (The sleeve and latch operate the same as the shift clutch ratchet and shift release arm.) The left end of the cycle clutch spring rotates farther after the right end is stopped because of the momentum built up in the cycle shaft, filter shaft, and print shaft. This additional rotation given to the left end of the cycle clutch spring is in the unwinding direction and causes the spring to expand about the hub on the cycle clutch pulley. This unwinding breaks the driving connection between the hub and the spring.

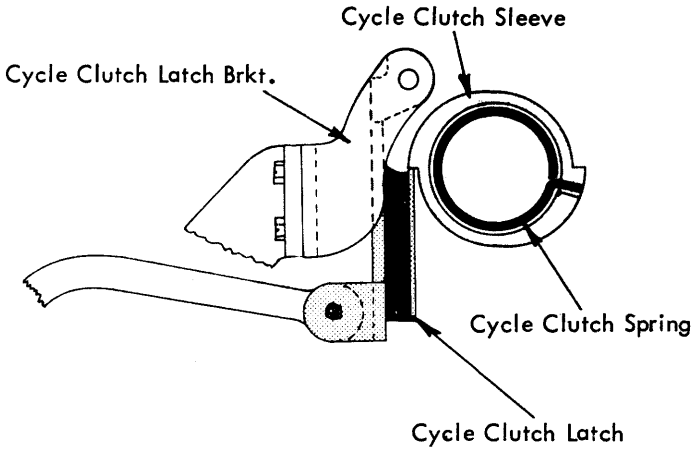


FIGURE 56. Cycle Clutch Latch - Side View

Since the shaft tends to travel beyond its rest position due to momentum, an overthrow stop is required. Two lugs on the side of the nylon cycle clutch restoring cam project into notches in the left side of the cycle clutch sleeve and operate as an overthrow stop (Fig. 57). When the sleeve is stopped by its latch, the cycle shaft continues to rotate under momentum (expanding the cycle clutch spring) until the nylon stop contacts the lugs on the sleeve stopping the overthrow of the shaft.

The shock of stopping the overthrow of the cycle shaft tends to bounce the cycle shaft backwards. To prevent this from occurring, a check pawl drops into a notch in a check ratchet located on the left end of the cycle shaft (Fig. 58). The pawl drops in when the clutch is disengaged. (It performs the same function as the shift detent roller.)

Both the overthrow stop and the check pawl assure that the cycle shaft will return exactly to its rest position at the completion of each cycle operation. (The cycle shaft is in its rest position when the positive selector cams are on their low dwells and the working face of a notch on the check ratchet is against the working face of the check pawl.)

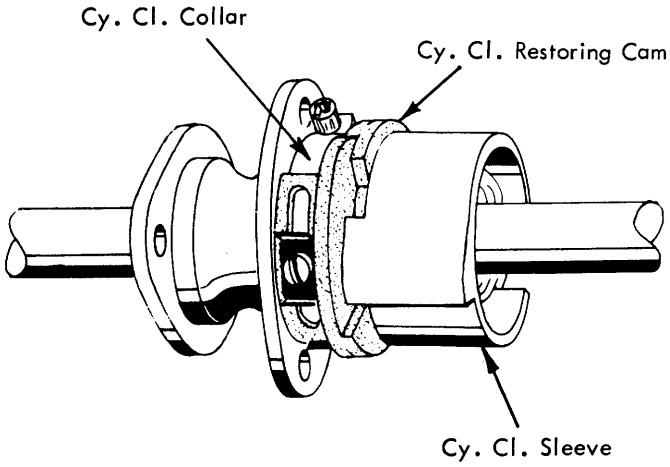


FIGURE 57. Cycle Clutch Stop

The shock of stopping the overthrow of the cycle shaft, filter shaft, and print shaft, is transmitted from the cycle shaft through the collar, the overthrow stop, the sleeve, and to the cycle clutch latch. This is why the cycle clutch latch is designed to absorb shock (vulcanized rubber mounting.)

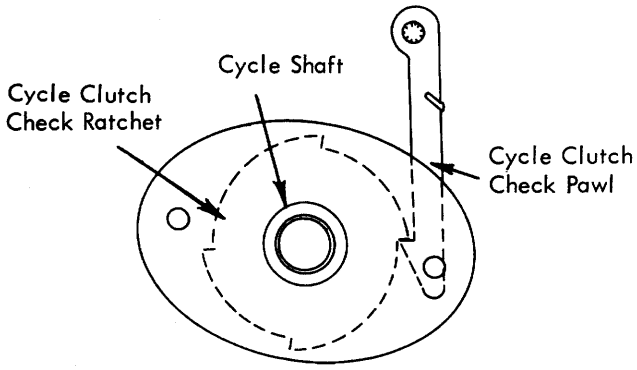


FIGURE 58. Cycle Clutch Check Pawl

The same effect is true in the shift mechanism. The shock of stopping the momentum of the shift cam is transmitted from the shift cam through the overthrow stop, the lug on the ratchet, and to the shift release arm. The release arm is shaped like a horseshoe so that it can absorb this shock.

The cycle clutch is allowed to engage by pivoting the cycle clutch latch forward out of the path of the step on the clutch sleeve. The clutch spring then quickly decreases in diameter because of its own spring tension. The rotating clutch pulley hub tightens the spring and drives the cycle shaft. The entire clutch assembly rotates through 180°. The second step of the clutch sleeve then contacts the cycle clutch latch which has been moved back to the rear into its path. This causes the cycle clutch to be disengaged again as previously described.

Movement of the detent actuating lever is controlled by the detent cam through the detent cam follower (Fig. 65). The cam follower is pivoted on a bracket below and to the rear of the print sleeve and extends up alongside the detent cam in position to operate the detent actuating lever. The camming surface of the detent cam is on the left side so that operation of the cam follower is toward the left against the detent actuating lever.

The rotate detent does not pivot into engagement as the tilt detent does. It contains an elongated hole at the front which allows both the front and the rear of the detent to move up and down (Fig. 66). A small flat link at the top stabilizes the detent. If the rotate detent were pivoted at the front, the amount of travel and timing of the detent would vary with the tilt selection. With the sliding arrangement, the rotate detent action is approximately the same for all tilt selections.

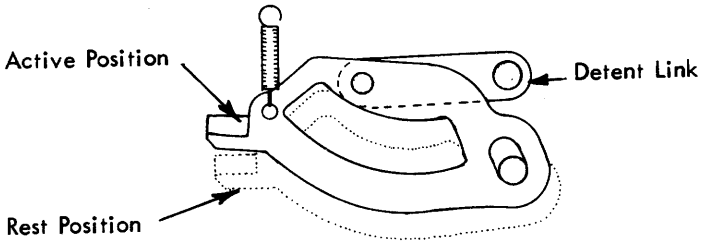


FIGURE 66. Rotate Detent

While the type head is being positioned, the cam follower is against the high part of the detent cam. This causes the detent actuating lever to be held to the left to prevent the detents from engaging their notches. During the positioning of the type head, the detent cam is rotating, but it maintains the same high point for the cam follower. As soon as the rotate and tilt operations have been completed, the cam follower is allowed to move to the right into a recess in the detent cam. This allows the detent actuating lever to move to the right to permit operation of the tilt and rotate detents. As the type head is being moved toward the platen the detent cam continues to turn, but it maintains the same low point for the follower.

As soon as the character has printed, locking the type head in position is no longer necessary; therefore the detents can be disengaged to allow the type head to restore to the rest position. As the rocker is restoring to the rest position, the detent cam moves the cam follower back to the left onto the high part of the cam. This action causes the detent actuating lever to disengage the detents from their notches. Timing of the detenting is such that both detents begin to engage their notches just as the type head completes its positioning and are disengaged just as the type head starts to rotate and/or tilt back to the rest position.

Detenting

The type head cannot be tilted nor rotated with the detents engaged. They must be held out of engagement until the type head has been completely positioned. They must also be removed from engagement before the type head can be restored to rest position. The rotate detent operates against the tilt detent. As long as the tilt detent is not allowed to rise, the rotate detent cannot enter the notches of the type head. The tilt detent is controlled by a small lever called the detent actuating lever located under the left side of the yoke (Fig. 65). The lever pivots at the rear and extends forward along the right side of a lower extension of the tilt detent. When the detent actuating lever is operated to the left, the tilt detent moves down pushing the rotate detent down with it. When the actuating lever moves to the right, both detents are allowed to rise.

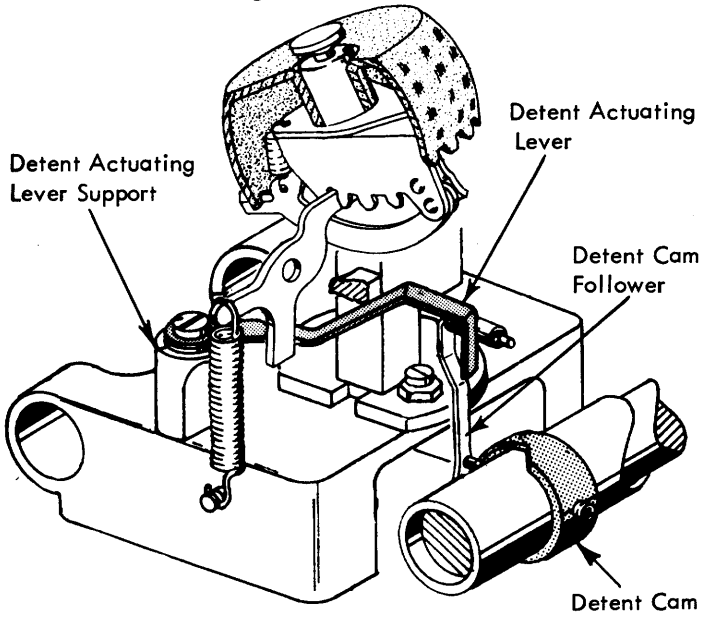


FIGURE 65. Detent Mechanism

Wear Compensator

A wear compensator is a device that senses a position and makes a correction whenever it detects any change from this position. The rotate arm assembly is constructed so as to compensate for wear in the differential system and related components. Because of this aspect of its operation, the rotate arm assembly is referred to as the wear compensator.

Before going into the details of the operation of the wear compensator, it is necessary to understand the alignment problem created by wear in the rotate system.

The rotate pulley spring beneath the type head and the compensating arm spring apply a constant pressure on the rotate system in the negative direction. Whenever wear occurs at any of the pivot points, bearings, stop pads, or linkage connections that oppose the combined tension of the rotate pulley spring and the compensating arm spring, the play (caused by wear) will be removed by these springs in a negative direction.

Removing the play (which is due to normal wear) allows the rotate pulley spring to turn the type head (Fig. 67) slightly in the negative direction. This is commonly referred to as head "drift".

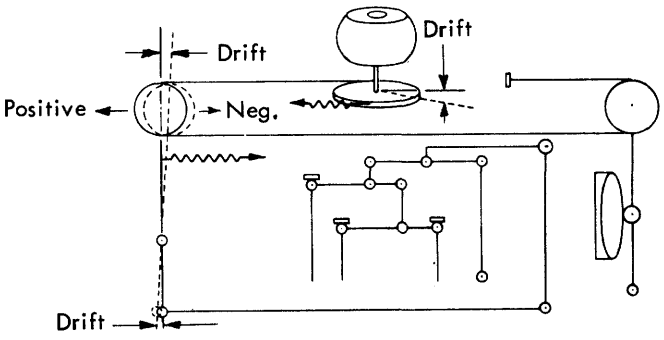


FIGURE 67. Rotate System Drift

As wear increases, head drift becomes excessive and failure of course or fine detenting occurs. Detenting failure necessitates a readjustment (re-homing) of the type head.

The wear compensator cannot prevent wear but it does prevent head drift which is due to wear in the system. The wear compensator is composed of three basic parts: the rotate arm, the compensating arm, and a nylon roller between them (Fig. 68). Other components are present that are indirectly involved in the wear compensator action. At this time, consider only the three parts mentioned.

The rotate arm (Fig. 68) pivots on a large pivot pin in a bracket mounted to the left side of the power frame. The left-hand rotate pulley operates at the top of the rotate arm. The compensating arm pivots at the same point as the rotate arm and extends in two directions from the pivot. The rotate link is fastened to the lower extension of the compensating arm. The upper extension of the compensating arm fits between the side-frames of the rotate arm. The nylon roller operates in a long vertical slot in the rotate arm. This slot and the upper extension of the compensating arm are at a slight angle to each other, thus forming a V-shaped wedging slot. The roller is retained in position in the wedging slot by the rotate tape tension.

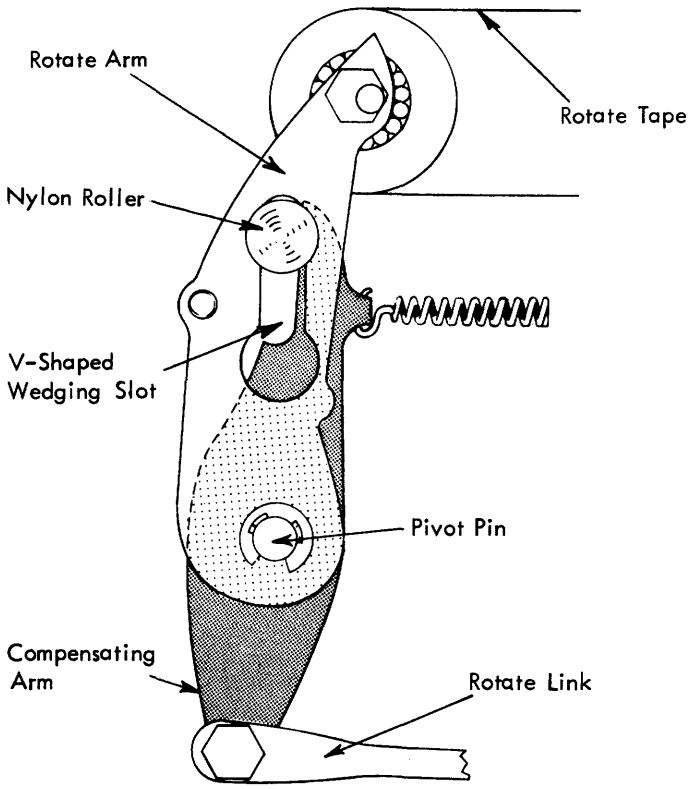


FIGURE 68. Basic Components Of The Wear Compensator

"The Basic Compensator Action of the Wear Compensator"

Figure 69-A shows the wear compensator in a zero rotate position. Note the following in Figure 69-A:

a. The compensating arm spring is applying tension on the upper extension of the compensating arm in a negative direction.

b. The rotate tape is applying tension (by means of the rotate pulley spring) to the top of the rotate arm in a negative direction also.

c. The rotate link attached to the lower extension of the compensating arm is opposing the compensating arm spring tension. The rotate link is also opposing the rotate tape tension through the compensating arm and the nylon roller.

d. The nylon roller is positioned near the top of the V-shaped wedging slot.

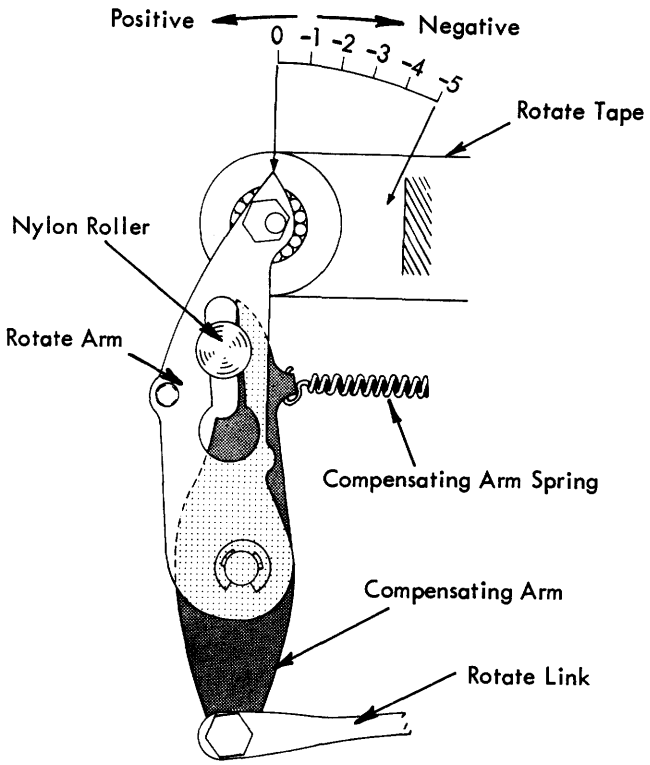


FIGURE 69A. Zero Rotate Position

Figure 69-B shows the wear compensator in the negative 5 position. Note that the eccentric stud mounted on the top of the rotate arm is just barely touching the machine sideframe when the wear compensator is in the negative 5 position.

Figure 69-C shows the wear compensator in the zero rotate position with wear introduced into the rotate differential system.

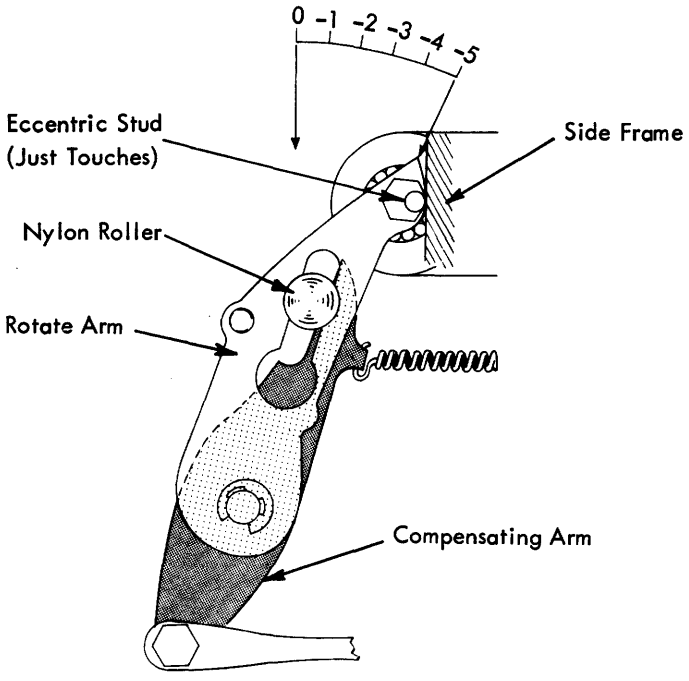


FIGURE 69B. Negative 5 Position

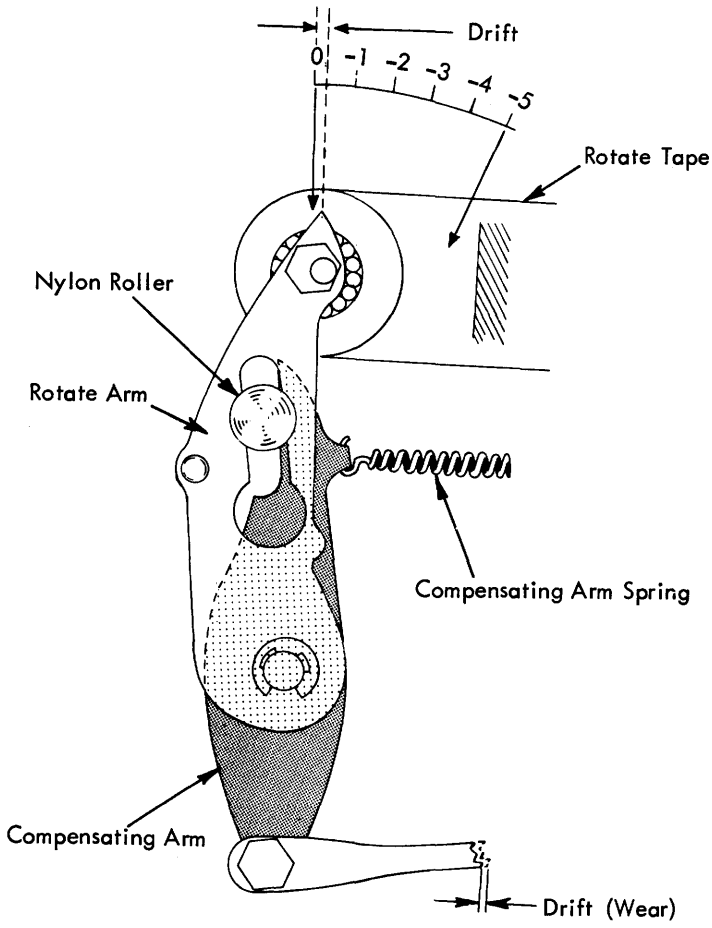


FIGURE 69C. Zero Totate Position With Drift

Note that the wear compensator has drifted away from the zero rotate position in a negative direction. The wear that was introduced into the rotate system was felt as play in the system. The rotate tape tension and the compensating arm spring removed the play in a negative direction which allowed the wear compensator to "drift" in a negative direction.

This places the rotate arm less than five units away from the sideframe when the rotate differential system is in the zero position. The rotate differential system is not aware of this and wants to supply five units of motion to the compensating arm when a negative 5 character is selected. The rotate arm cannot travel a full five units because this would carry it beyond the sideframe. The sideframe limits the motion of the rotate arm at exactly the rotate negative five position. Since there is nothing to stop the motion of the compensating arm, it continues to travel the full five units of motion supplied to it by the rotate differential system. The V-shaped wedging slot opens up and the nylon roller drops lower in the slot (Fig. 69-D). When the wear compensator returns to the zero rotate position, the drift has been eliminated and wear has been compensated for. Basically, this is how the wear compensator works. In order for the wear compensator to operate efficiently, we must overcome the effects of a changing spring load on the rotate differential system as the eccentric stud on the rotate arm contacts the sideframe.

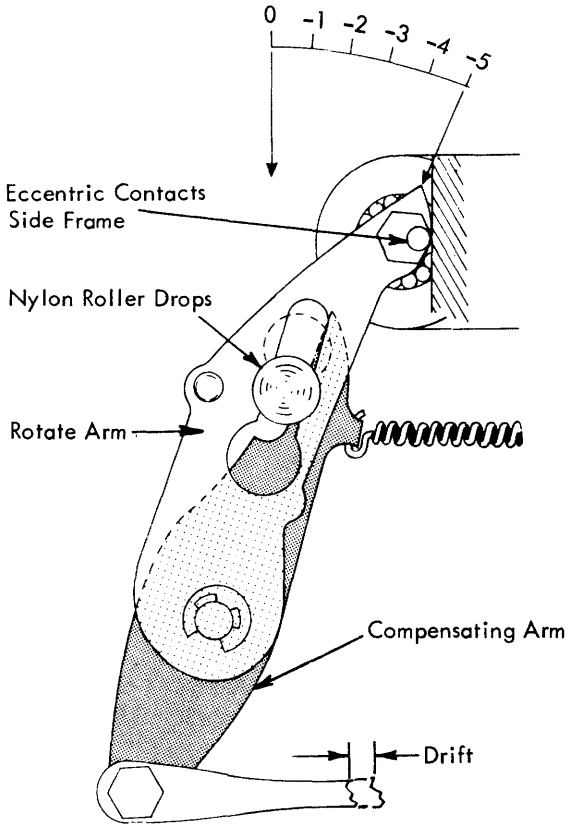


FIGURE 69D. Negative 5 Position With Drift (Nylon Roller Drops)

In the zero rotate position, there are many parts in the rotate differential system that are resisting the combined spring tension of the rotate pulley spring and the compensating arm spring. All the parts that are opposing this spring tension are being stressed (flexed) slightly. Although steel parts appear to be extremely strong and rigid, they do have a measurable amount of elasticity when placed under a stress. Should the stress (spring tension) be reduced, the steel parts will tend to return to their original shape and position.

During a negative 5 selection, the eccentric stud on the rotate arm contacts the machine sideframe. The sideframe now starts to oppose the rotate pulley spring tension; and the spring load on the rotate differential starts to reduce to that of the compensating arm spring. When the spring load on the rotate differential starts to reduce (due to the rotate arm contacting the sideframe), the flexing of the rotate differential begins to reduce also. Even though a portion of the rotate pulley spring tension is being opposed by the sideframe, the rest of the rotate pulley spring tension is still applying tension to the rotate differential through the nylon roller. This keeps the system in a flexed condition maintaining pressure against the nylon roller. Under these conditions, a small amount of wear in the rotate differential would not allow the roller to drop. The wear would only be absorbed in reducing the flexing of the rotate differential at the negative 5 rotate position. The wear compensator would not compensate for wear and "drift" would be apparent in all rotate positions except negative 5 (sideframe would prevent drift at the negative 5 rotate position).

To summarize the above, let's simplify the situation by making an assumption. Suppose that all the flexing in the rotate differential could be placed at one point. Let's assume that the nylon roller is made out of soft rubber that can be compressed easily. With the wear compensator arm assembly in the zero rotate position, the rotate pulley spring tension is squeezing the rubber roller between the rotate arm and the compensating arm. Pretend that the amount the rubber roller is squeezed is the flexing of the differential system. Now, half cycle the machine to a negative 5 position. The eccentric stud just touches the sideframe and the rubber roller remains compressed. At this time, let's assume that a large amount of wear came into the rotate differential system. (Wear shows up as play.) The compensating arm spring would pull the top of the compensating arm in a negative direction, removing the play from the differential system. The rotate arm, restricted by the sideframe, could not follow the compensating arm and the pressure on the rubber roller would be relieved allowing it to expand. The play (caused by wear) in the system was just enough to allow the rubber roller to expand without dropping. Now, if the rotate link starts to pull the arm assembly back to a zero rotate position, some of the rotate link motion is going to be used to compress the rubber roller before the rotate arm leaves the sideframe. This means that the rotate arm will not receive sufficient motion to return to the zero rotate position. Wear has caused the rotate arm to drift in a negative direction.

Conclusion:

a. Flexing in the system is inherent.

b. Any lost motion from the rotate differential to the rotate arm will cause "drift".

c. The pressure on the roller must be relaxed before compensation takes place.

"Wear Compensator Ratio Change"

In the wear compensator, there is a constant leverage ratio between the amount of motion supplied to the bottom of the compensating arm (by the rotate link) and that amount of motion produced at the top of the rotate arm. (The movement of the rotate arm directly depends upon the movement of the upper extension of the compensating arm through the nylon roller.)

If the pivot point of the compensating arm is changed to a lower position (closer to the rotate link) the leverage ratio of the compensating arm will increase. This will cause a greater amount of motion to be produced to the top of the compensating arm for a given amount of motion supplied to the bottom of the arm. When this occurs, it is correct to say that the compensating arm has undergone a "ratio change". The ratio change is required in order to relax the pressure on the nylon roller while maintaining the correct output to the rotate arm at a negative 5 position.

The ratio change begins approximately at the negative 4 position and occurs as the arm sweeps through to a negative 5 position. It provides the upper extension of the compensating arm with sufficient motion to relax the pressure on the nylon roller (without allowing it to drop so that it is ready to drop as soon as the slightest amount of wear is felt in the differential system. With the pressure relaxed on the roller, any wear coming into the differential system allows the compensating arm spring to pull the top of the compensating arm further in a negative direction opening up the V-shaped wedging slot. This causes the roller to move down in the slot compensating for the wear.

In order to relax all of the pressure on the nylon roller when the rotate arm contacts the machine sideframe, the sideframe must oppose all of the rotate spring tension (plus the compensator assist spring which will be covered later). This causes the spring load on the differential system to reduce to just that of the compensating arm spring thereby causing the system, from the nylon roller on down, to partially relax. This partial relaxing of the differential system creates a slight amount of motion to the rotate link in the opposite direction that the link is moving, thus resulting in a reduction of output of the rotate link to the bottom of the compensating arm between a negative 4 and negative 5 rotate position. The ratio change of the compensating arm overcomes this effect by producing more motion to the top of the arm for a reduced amount of motion from the rotate link, thus the rotate arm reaches the negative 5 position and the pressure on the nylon roller is relaxed.

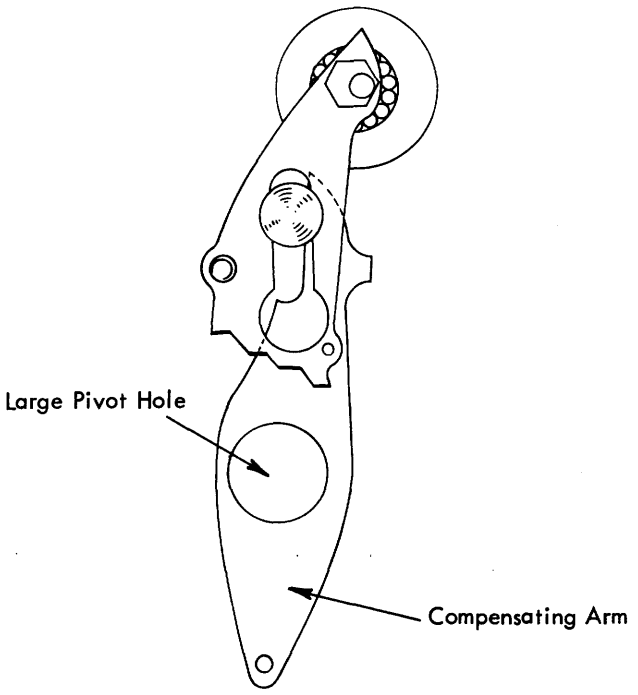


FIGURE 70. Compensating Arm Pivot Hole

The compensating arm contains a large pivot hole (Fig. 70). Fitted inside the large pivot hole is an eccentric shoulder which is part of the rotate eccentric arm (Fig. 71). The compensating arm and the rotate eccentric arm act as one solid piece as they rotate about the pivot pin for all positive and negative rotate positions up to approximately the negative 4 position. If a negative five character is selected, the rotate eccentric arm will be restricted from rotating (about the pivot pin) beyond the negative 4 position and the compensating arm will then begin to rotate about the eccentric shoulder on the rotate eccentric arm.

Since the center of the eccentric shoulder is lower than the pivot pin (closer to the rotate link) and the compensating arm pivot point is shifted (between negative 4 and negative 5) from the pivot pin to the eccentric shoulder, it is correct to say that the compensating arm has undergone a ratio change.

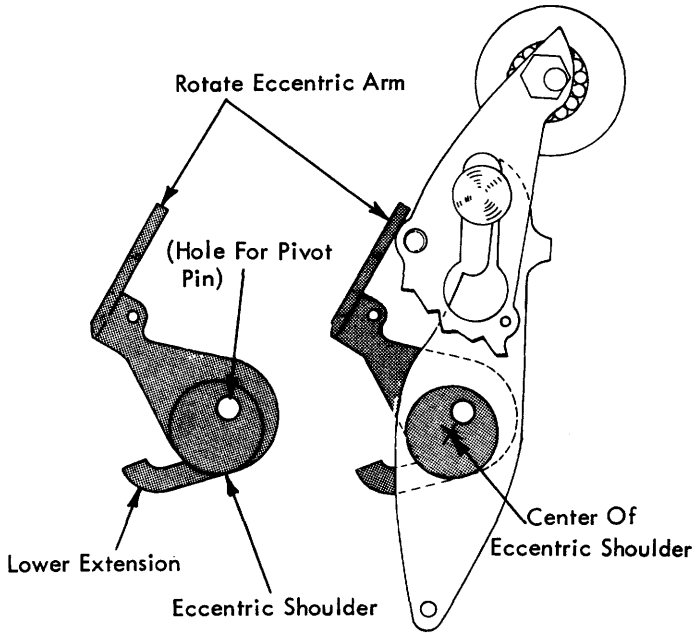


FIGURE 71. Mounting Of Rotate Eccentric Arm

The rotate eccentric arm is spring loaded against the rotate arm by means of the rotate eccentric arm spring (Fig. 72). The rotate arm is spring loaded against the compensating arm by means of the rotate pulley spring. Therefore, the rotate eccentric arm, rotate arm, and compensating arm act as one solid arm rotating about the pivot pin from a positive 5 to the negative 4 rotate position. During a negative 5 selection, the lower extension of the rotate eccentric arm (Fig. 73) contacts the stop lug on the wear compensator bracket (at the negative 4 position). The rotate eccentric arm is stopped and the compensating arm ratio change begins. The rotate arm continues to follow the compensating arm because the rotate pulley spring is stronger than the rotate eccentric arm spring. The rotate arm is stopped by the machine sideframe when it reaches the negative 5 position. The additional motion given to the compensating arm (due to the ratio change) is used up in partially relaxing the rotate differential system. Thus, wear can be compensated for.

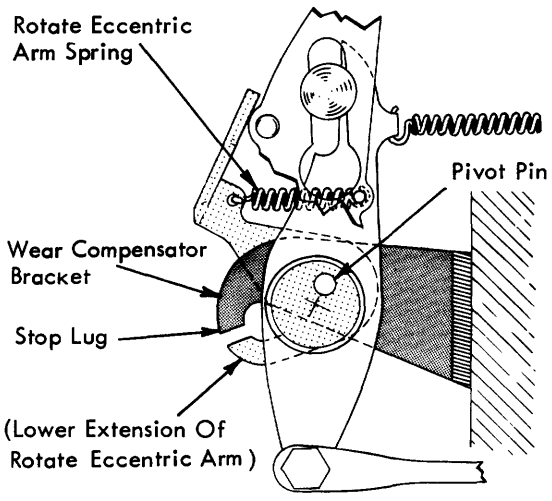


FIGURE 72. Zero Rotate Position

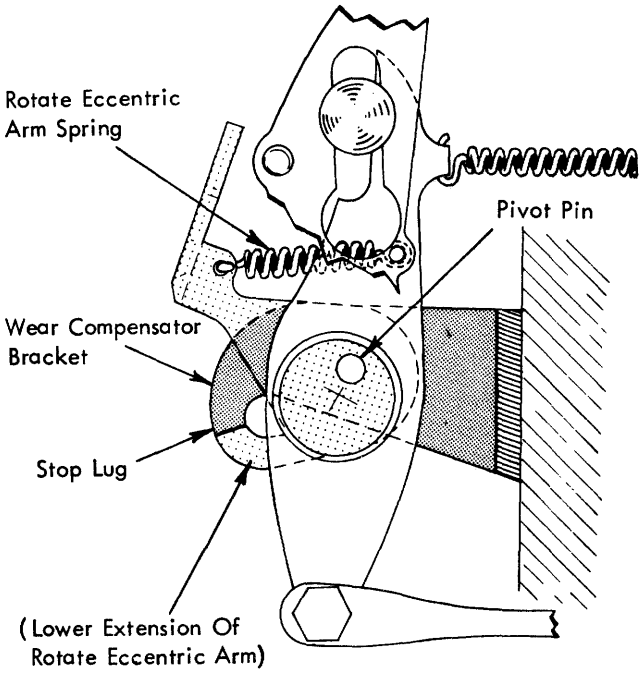


FIGURE 73. Negative 5 Position

The wear compensator compensates for drift due to the wear in the system from the rotate arm down through the linkage to the negative 5 cam and the cycle shaft bearing. Although wear occurs throughout this whole area, the major portion of wear comes from the differential system and related linkages. Compensating for wear in these areas is the major requirement of the wear compensator. Wear in the tape system constitutes only a small portion of the total wear in the system.

Because there is some drift due to the wear in the tape system, the type head is initially set ("homed") to favor the positive direction, relative to the rotate detent. As wear comes into the tape system, the type head will drift slightly in the negative direction. The type head notch will then tend to become centered relative to the rotate detent.

After the initial break-in period, the rate of wear levels off in both the differential system and the tape system. Although wear still occurs, it progresses at a very slow rate.

The wear compensator cannot compensate for wear in the positive latches, the latch bail, and the positive cams because these components are not in use during a compensating operation (negative 5). Wear at these points constitute a portion of the band width and must be considered when achieving an allowable band width.

Figure 74 points out two other springs which have a direct bearing on the wear compensator operation.

An extension spring called the compensator assist spring is connected between the rotate arm and the carriage side frame. The purpose of this spring is to keep the rotate arm in contact with, and applying pressure against, the nylon roller throughout a positive operation. Without the assist spring, the rotate arm tends to overthrow beyond the positive position selected because of a momentum built up in the rotate arm, tape system, rotate

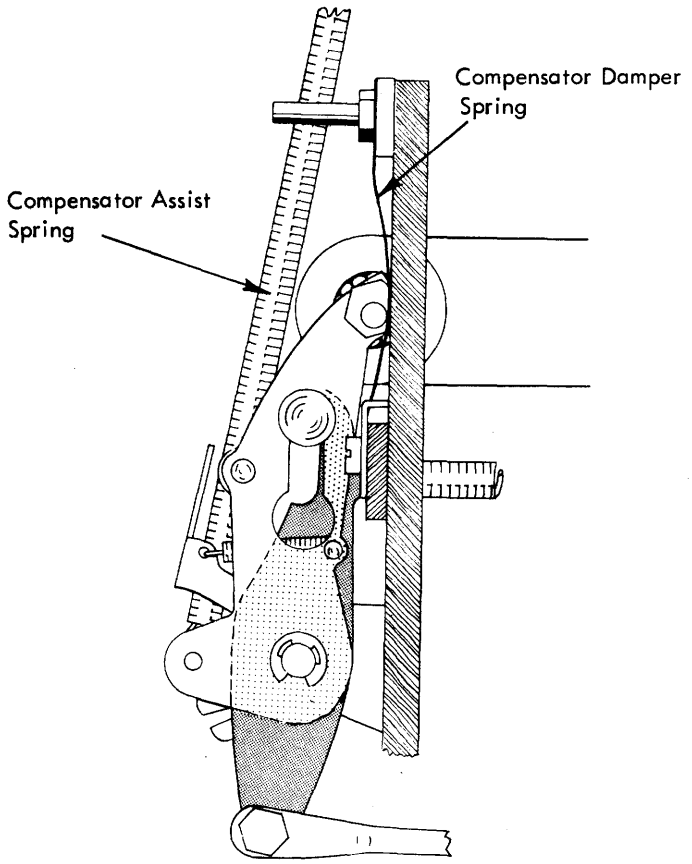


FIGURE 74. Wear Compensator

pulley, type head, etcetera. This overthrow allows the V-shaped wedging slot to open up resulting in unwanted roller droppage.

The compensator damper spring (Figure 74) is a leaf spring designed to absorb the shock of stopping the rotate arm at the negative 5 position. This prevents unnecessary stress on the components and eliminates vibrations in the tape system that would otherwise result. It also prevents the rotate arm from rebounding off the side frame. Rebounding would cause the V-shaped wedging slot to open, resulting in slight roller droppage.

Detent Timing

The timing of the print shaft with respect to the cycle shaft controls the timing of the detents. Detent timing must be set accurately in order to prevent damage from occuring in the tape system or the typehead. If the rotate detent is allowed to engage the typehead too soon, it may enter the wrong detent notch. If this occurred during a negative rotation of the typehead, the rotate tape would tend to slacken and jump off its pulleys. This could lead to tape breakage or roller droppage in the wear compensator. If the detent is allowed to engage the typehead too soon during a positive rotation, tape or typehead breakage may result due to the continued pull on the tape. Also, retarding the detent timing may lead to breakage or roller droppage because the detent would remain in the typehead notch when the head is trying to restore. Detent timing can

also affect print alignment. It is possible for the detents to begin to withdraw before the typehead prints. For this reason the detent timing should be adjusted as late as possible without restricting the typehead from restoring.

Several factors affect detent timing. Some of the most important are: the adjustments of the rotate and tilt systems, typehead homing, detent actuating lever and cam adjustments, and timing of the print shaft relative to the cycle shaft. If any of these are changed, the machine MUST ALWAYS be cycled by hand to check the detenting action before it is operated under power.

It should be noted that the detents are spring loaded into engagement, but they are driven out of engagement by the cam. This is to prevent parts breakage should the type head not be properly positioned when the detents try to engage.

Wear Potential

Wear potential in the tilt and rotate mechanism is defined as: the ability of the tilt and rotate mechanisms to properly coarse align the typehead after a measurable amount of uncompensated wear is felt in either mechanism. Although wear potential is designed into both mechanisms it will only be discussed in the rotate mechanism.

A portion of the typehead play provides the rotate system with a substantial amount of wear potential. In order to explain how this is accomplished, it is necessary to understand the relationship between typehead play, typehead homing and band width.

Figure 75 illustrates the typehead play by showing a single detent notch (of the typehead) in the two positions as allowed by the built in play.

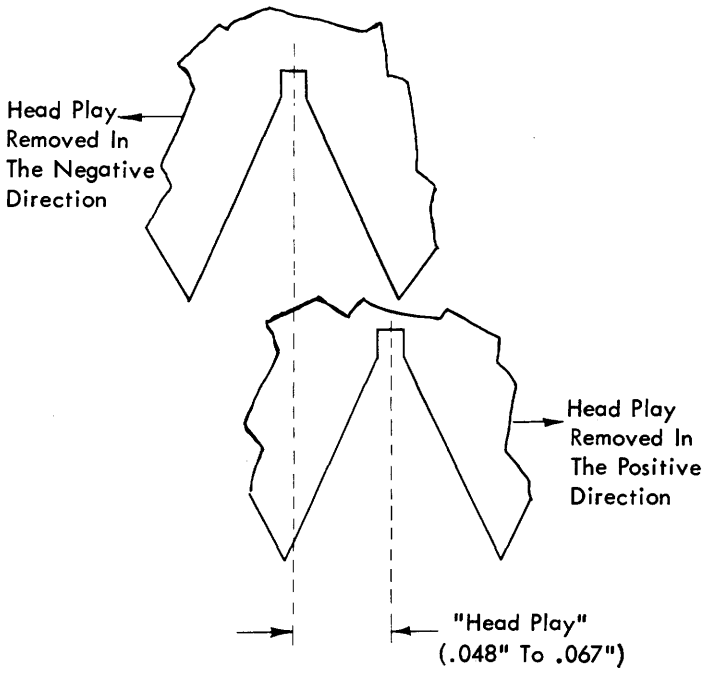


FIGURE 75. Typehead Play

The typehead play, which is approximately .060" measured at the typehead skirt, can be separated into three sections or segments each having their own function. The first segment of head play is spent for detent timing purposes by the typehead homing adjustment (Fig. 76).

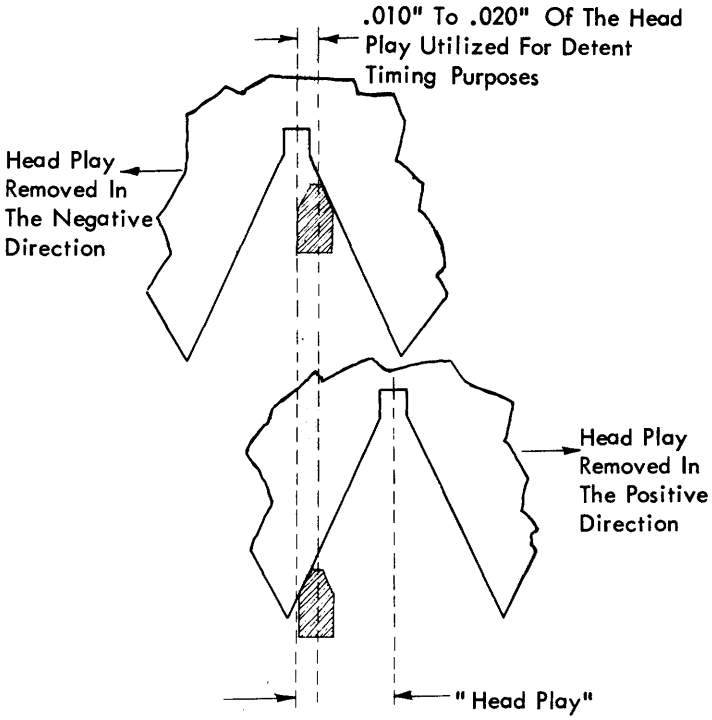


FIGURE 76. Typehead Homing

The typing element is "homed" to the rotate selection that will coarse align the most positive (with respect to the rotate detent) when the head play is removed in the negative direction. It is "homed" so that this rotate selection will detent .010" to .020" on the negative side of the detent notch when the head play is removed in the negative direction (Fig. 76).

The purpose of this adjustment is to slightly retard the restoring of the typehead whenever it restores in the positive direction so that the rotate detent can begin to withdraw before the typehead starts to restore. If the withdrawal of the detent did not lead the restoring of the typehead, the detent would restrict the head from restoring causing breakage in the system. This homing adjustment uses up approximately 1/4 of the typehead play.

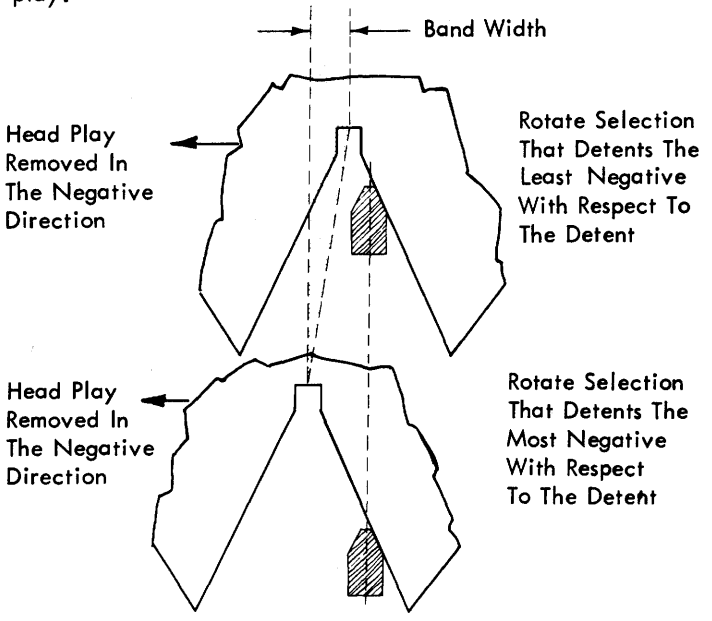


FIGURE 77. Band Width

The second segment of head play is spent on "band width". With head play removed in the negative direction, Fig. 77 illustrates band width by showing the amount of variation (in coarse alignment) between the rotate selection that detents the least negative and the rotate selection that detents the most negative with respect to the rotate detent.

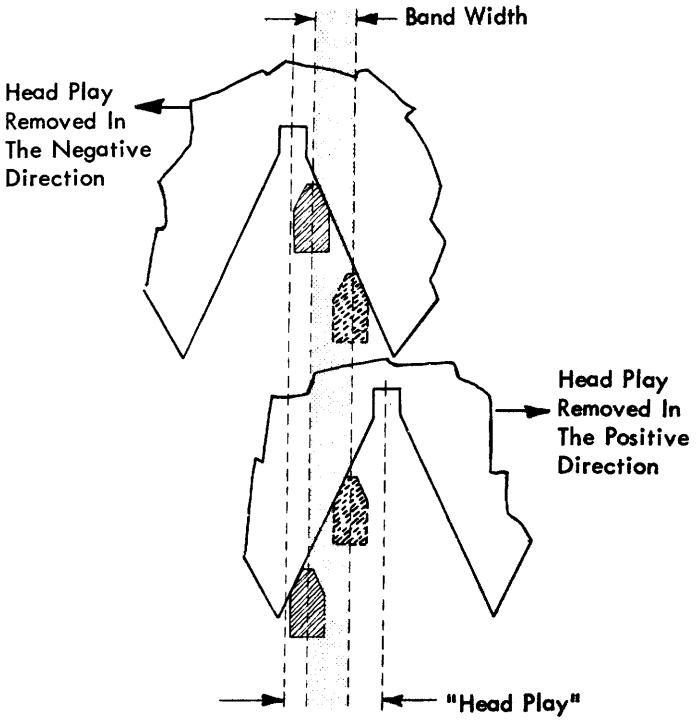


FIGURE 78. Band Width

Fig. 78 illustrates band width in relation to head play. The allowable band width of the rotate system may utilize as much as 1/2 of the typehead play. Note that almost 3/4 of the head play is spent for typehead homing and band width (Fig. 76 and 78). The remaining head play that is left is used for wear potential (Fig. 79).

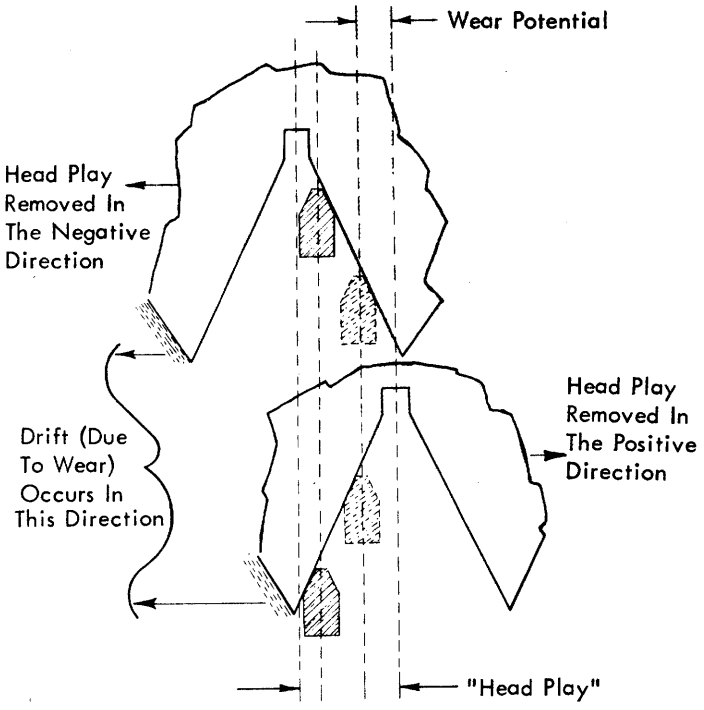


FIGURE 79. Wear Potential

When wear occurs in the system the typehead drifts in the negative direction with respect to the detent. This also causes the head play and band width to drift in the negative direction with respect to the detent. As long as this drift does not exceed the wear potential portion of the head play the detent will continue to fine align the typehead. Once it has exceeded the wear potential, the rotate selection that coarse aligns the most negative with respect to the detent will fail to seat in the detent notch. The detent will then fail to seat causing that character to print out of alignment.

PRINT MECHANISM - OLD STYLE

The purpose of the print mechanism is to actuate the type head against the platen and restore it to the rest position. The print mechanism consists of the print shaft, print sleeve, print cam and follower, velocity control plate, and rocker assembly. The anvil and striker operate to limit the amount of character embossing. A copy control mechanism allows the operator to select the front to rear position of the platen to correspond to the thickness of the typing material.

Print Shaft and Print Sleeve

The print shaft extends between the side frames just to the rear of the keylever fulcrum rod. It is supported at each end by a self-aligning porous bronze bearing. A small gear outside the left side frame connects the shaft, through idler gears, to the cycle shaft gear. The ratio between the two gears is 2:1. This means that each time the cycle shaft completes a cycle operation (180° rotation), the print shaft is rotated 360° (one complete revolution).

The print sleeve of the carrier assembly rides on the print shaft. A keyway throughout the length of the print shaft provides a rotary connection between the sleeve and the shaft yet permits lateral movement of the carrier. A key fits through a hole in the print sleeve and into the keyway of the print shaft (Fig. 80). Whenever the print shaft is rotated, the print sleeve is rotated by the key connection.

The print sleeve contains four cams (Fig. 80). The two middle cams are the ribbon feed cam and the detent cam for the type head alignment. Both of these cams are keyed to the print sleeve by the same key that extends into the print shaft. Each cam is secured to the sleeve by set screws.

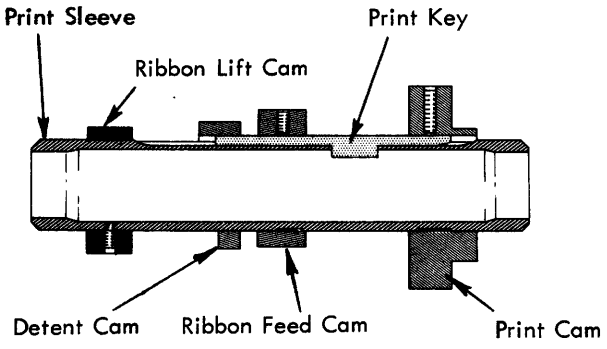


FIGURE 80. Print Sleeve And Cams

The cam at the left is the ribbon lift cam. It is set-screwed to the print sleeve causing the cam to rotate with the sleeve. The set screw mounting also prevents the print sleeve from shifting toward the left.

The cam at the right is a double cam called the print cam. Its function is to power the type head toward the platen and restore it to rest. Two camming surfaces are necessary for this operation. The smaller surface to the right is called the print cam and moves the type head to the platen. The larger cam surface called the restoring cam restores the type head and prevents it from rebounding. The print cam is also keyed to the print sleeve to provide rotation with the sleeve. The print cam is held in position by two set screws to prevent the print sleeve from moving to the right and to provide a solid driving connection between the sleeve and the cam.

Print Cam and Follower

The rocker assembly is powered by the print cam on the print sleeve. A yoke-shaped part called the print cam follower pivots on a stud inside the right side of the carrier (Fig. 81). The two arms of the follower are operated by the print cam. The rear of the print cam follower contains a forked slot. The velocity control plate which is fastened to the right side of the rocker has a stud that projects into and operates in this forked slot (Fig. 82). When the print cam rotates, it forces the front of the follower down causing the rear of follower (slot) to move up creating a rising action to the front of the rocker by means of the stud on the velocity control plate. This rising action powers the typing element to the platen resulting in a print operation.

When the machine is at rest the high point of the print cam is facing toward the front of the machine. This makes the print operation occur late in the cycle which allows sufficient time for the selection and differential mechanisms to position the typing element before the print operation begins.

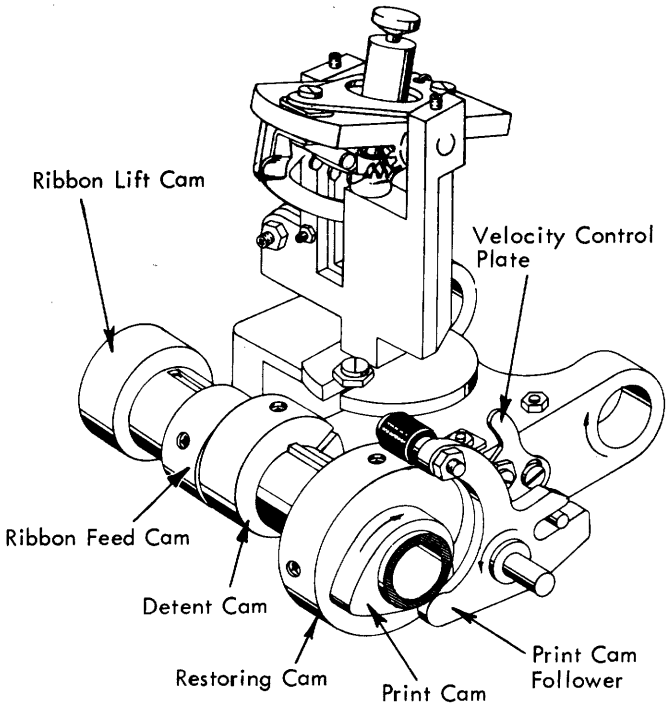


FIGURE 81. Print Mechanism

The larger camming surface on the left of the print cam is called the restoring cam (Fig. 82). Its purpose is to power the rocker back to the rest position after the type head prints thereby preventing any rebounding from occurring. This is accomplished by the restoring roller on the upper arm of the cam follower.

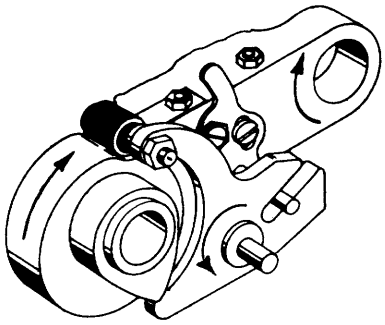


FIGURE 82. Print Cam Action

The camming surface of the print cam is designed so that the type head is powered to within a few thousands of an inch of the platen. To prevent any choking action, the momentum of the rocker carries the type head the remaining distance. The contour of the rebound cam is such that it allows "free flight" of the rocker and type head, when the type head is near the platen. Because the type head is powered nearly all the way to the platen, all the characters are forced to emboss the paper slightly, even those with a large surface area.

Anvil and Striker

Because the type head is powered nearly all the way to the platen, it is necessary to restrict the amount of free travel of the type head so that the amount of embossing for all characters will be the same. A heavy arm called the anvil striker attached to the bottom of the rocker acts as a stop for the rocker (Fig. 83). Just under the front of the carrier and extending

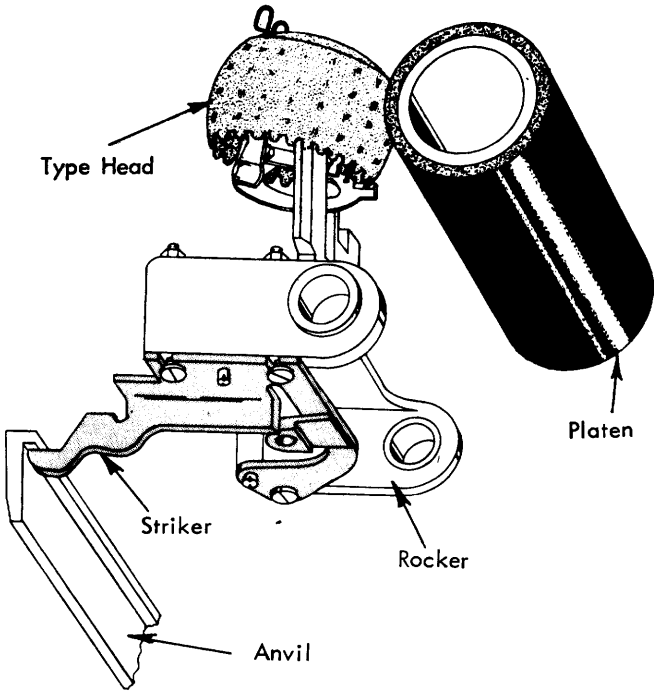


FIGURE 83. Anvil And Striker

between the side frames is the anvil. It is an angle-shaped bar with the lip extending to the rear. As the type head embosses the paper to the correct depth, the striker hits the bottom of the anvil and prevents further travel of the rocker and type head. This means that the impression for each character will be consistent with little variation between characters.

Whenever the carrier prints in the middle of the writing line on long carriage machines, downward flexing of the print shaft occurs resulting in a loss of impression. Carrier buffers (Fig. 84) located on each side (at the front) of the carrier casting limit this flexing action by operating against the top surface of the anvil.

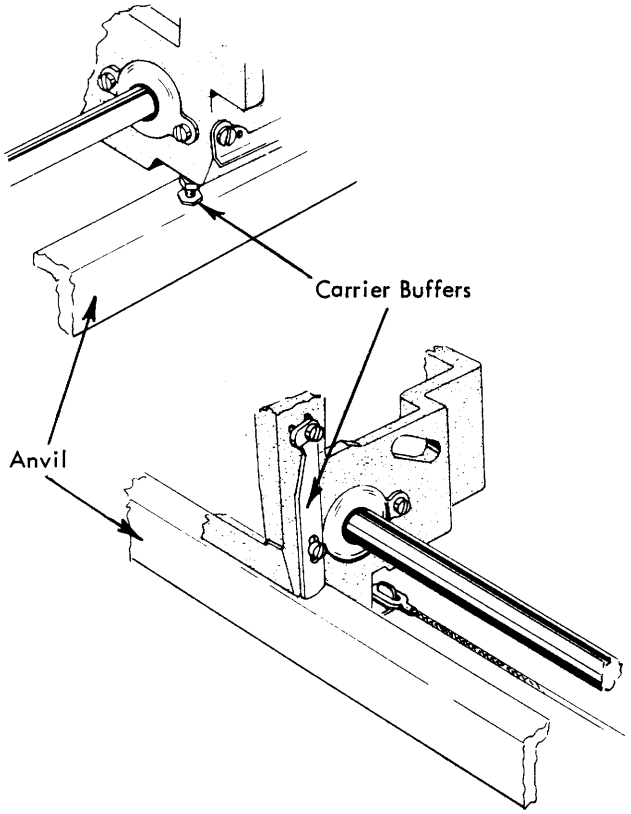


FIGURE 84. Carrier Buffers (long carriage)

Platen

The purpose of the platen is to feed the paper vertically and to provide a solid backing for the paper during a typing operation. The quality of type impression obtained is determined to a large extent by the condition of the platen. Platen rubber may be adversely affected by numerous factors such as light, heat, chemicals, etc. An old or worn platen may be considerably harder than a new platen and may also vary slightly in diameter. This machine is equipped with a platen with a hardness density comparable to the number 2 platen used with the standard IBM Electric Typewriter.

The platen is held in position on the machine by a latch pivoted at the front on each carriage plate (Fig. 85). The platen may be removed by pressing the rear of the latches down and lifting the platen out. It may be installed by snapping it into position without depressing the latches. The camming action of the latches causes them to remove all vertical as well as horizontal play from the platen.

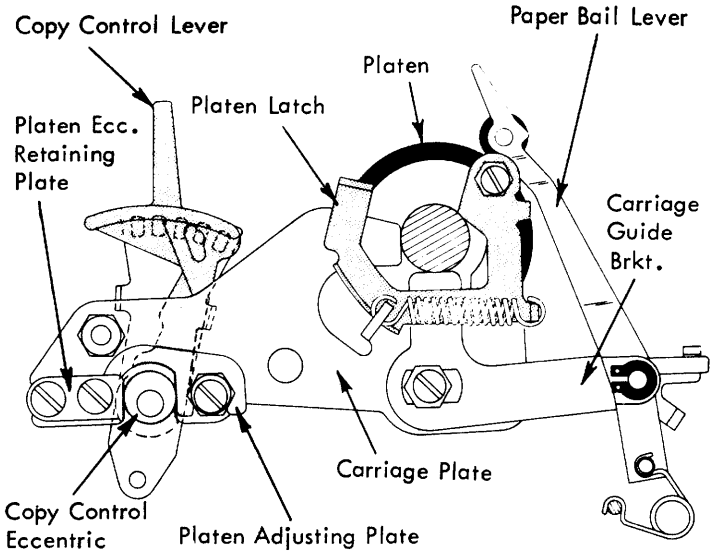


FIGURE 85. Copy Control Mechanism

Copy Control

The purpose of the copy control mechanism is to position the platen forward or back for different thicknesses of typing material. Positioning the platen maintains the correct relationship between the anvil and the point of impact of the type head with the paper.

The copy control is operated by means of the copy control lever located at the left end of the carriage (Fig. 85). The lever is attached to the copy control shaft that extends out through the sides of the powerframe. An eccentric collar at each end of the shaft operates between adjusting parts attached to the carriage ends. When the lever is moved to the rear, the shaft rotates causing the eccentric collars to contact the platen eccentric retaining plates and force the carriage ends to the rear. The platen and entire paper feed mechanism move with the carriage. When the copy control lever is pulled forward, the eccentric collars contact the platen adjusting plates and force the carriage forward into the normal position.

The copy control lever can be set in five different positions. A spring detent attached to the powerframe acts against a knob on the copy control lever to hold it in place.

PRINT MECHANISM - NEW STYLE

The new style print mechanism contains an operator impression control lever which permits the operator to regulate the impression for any application she may type. The operator may change the over-all impression of the typehead by merely positioning the impression control lever to one of five settings.

In addition, the new style print mechanism is equipped with an automatic velocity selection mechanism. The purpose of this mechanism is to provide a lighter impression for the periods, commas, colon, semi-colon, quotation mark, apostrophe, hyphen, and underscore; regardless of where the impression control lever may be set.

Before going into any detail on the impression control mechanism it is necessary to first understand how the automatic velocity selection mechanism operates since both mechanisms are directly related.

Automatic Velocity Selection - ("Dual cam")

The printing operation of all the upper and lower case characters in positions 32, 36, 38, 39, and 42 produce a lighter impression on the typed copy than all the other positions. The reason for this is to improve the general appearance of the typed copy by producing a more uniform impression between all characters, large or small. This is accomplished by using a print cam that has two different camming surfaces (Fig. 86).

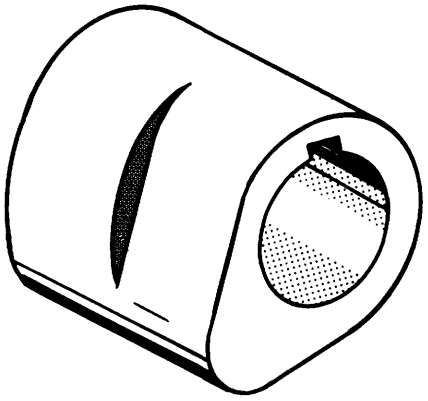


FIGURE 86. Print Cam (dual velocity)

The low and high points of both camming surfaces are identical. The only difference is in the contour (profile) between their low and high points. The contour of one camming surface provides the type head with a lower impact velocity than the other. Thus, a lighter impression is produced when this camming surface is used. The difference in type head velocities produced by the two camming surfaces remains proportional regardless of the impression control lever setting.

A selection mechanism positions the print cam follower roller under the proper camming surface of the dual velocity print cam whenever a character is selected at the keyboard. The print cam follower assembly (Fig. 87) is mounted on a pivot pin located in the right side of the carrier below and to the rear of the print cam (just as it is in the old style print mechanism). The print cam follower roller mounts on a pin on the print cam follower and is free to slide left or right on this pin.

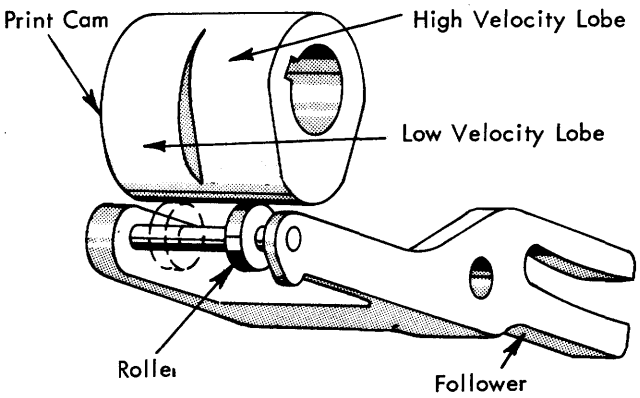


FIGURE 87. Print Cam Follower Assembly

The camming surface or lobe on the print cam that produces the greatest impact velocity is called the high velocity lobe. This is the right hand lobe on the print cam. The left hand lobe (producing less impact velocity) is called the low velocity lobe (Fig. 87).

The roller is positioned (left or right) under either camming surface of the print cam by the print cam follower roller yoke which straddles the roller (Fig. 88). The yoke is mounted on a pin that protrudes from the tab cord anchor bracket assembly. The yoke is also free to slide left or right on its mounting pin.

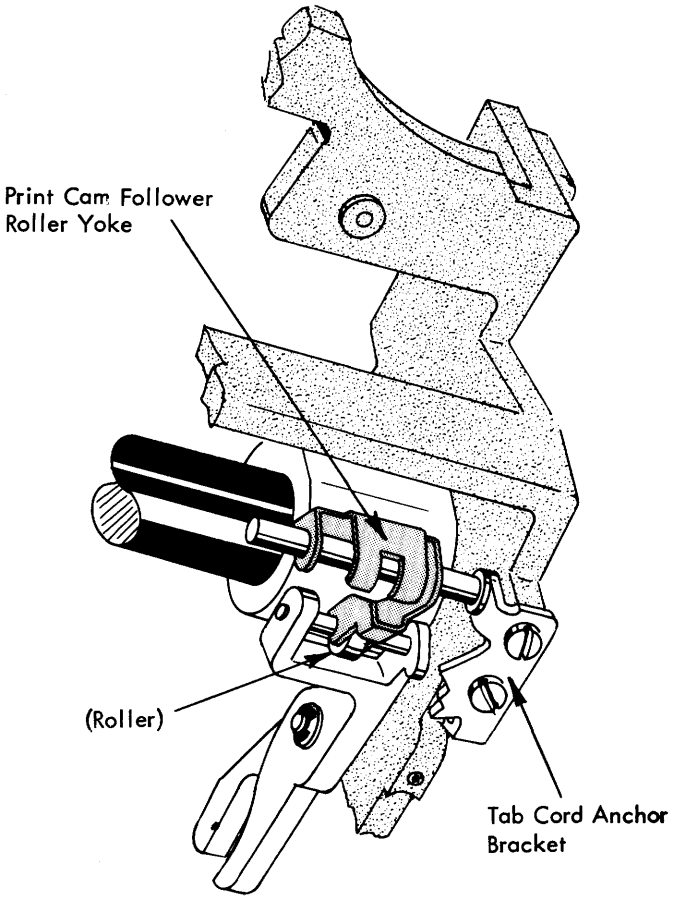


FIGURE 88. Print Cam Follower Roller Yoke

A lever, which controls the lateral position of the yoke and roller, mounts on the tab cord anchor bracket by a shouldered rivet. This lever is called the yoke actuating lever (Fig. 89).

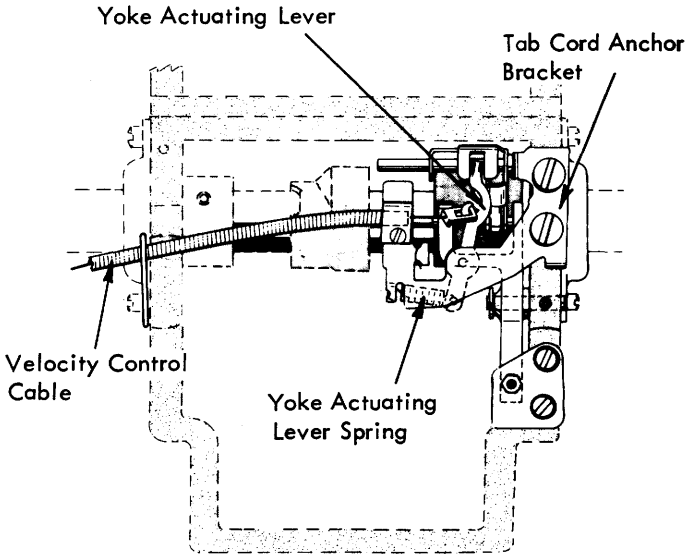


FIGURE 89. Yoke Actuating Lever And Spring

The yoke actuating lever is spring loaded at the rear in a clockwise direction (observed from the bottom of the machine) by the yoke actuating lever spring. This spring tension positions the roller directly beneath the high velocity lobe of the print cam as shown in Fig. 89.

A sheathed cable called the velocity control cable fastens to hooked portion of the yoke actuating lever (Fig. 89). Whenever a pull is produced on the velocity control cable, the yoke actuating lever will rotate counterclockwise, (stretching its spring) shifting the print cam follower roller from the high velocity lobe to the low velocity lobe of the print cam as shown in Fig. 90.

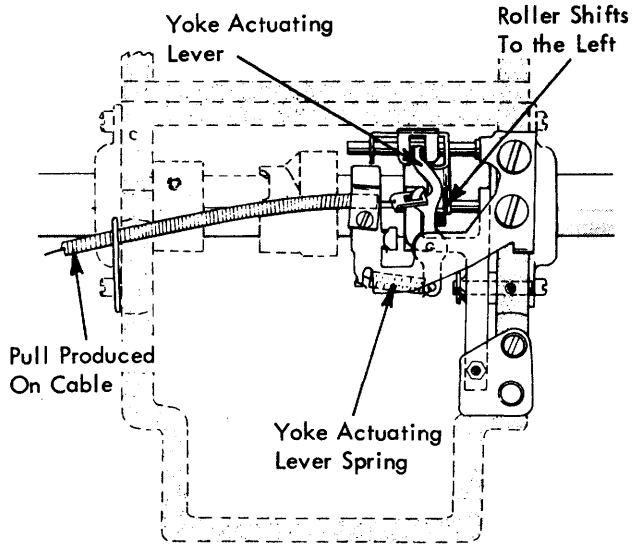


FIGURE 90. Low Velocity Operation

When the pull on the velocity control cable is relaxed, the yoke actuating lever spring shifts the roller back to the right, positioning it under the high velocity lobe of the print cam (Fig. 89).

Whenever a low velocity character is selected, (at the keyboard) a pull on the velocity control cable is produced to shift the roller to the low velocity lobe on the print cam. This pull on the cable is initiated at the keyboard. When any one of the selector interposers that are in positions 32, 36, 38, 39, or 42 are powered forward by the filter shaft, a knob on the forward end of that interposer contacts an upright lug on the low velocity vane causing it to rotate forward (Fig. 91).

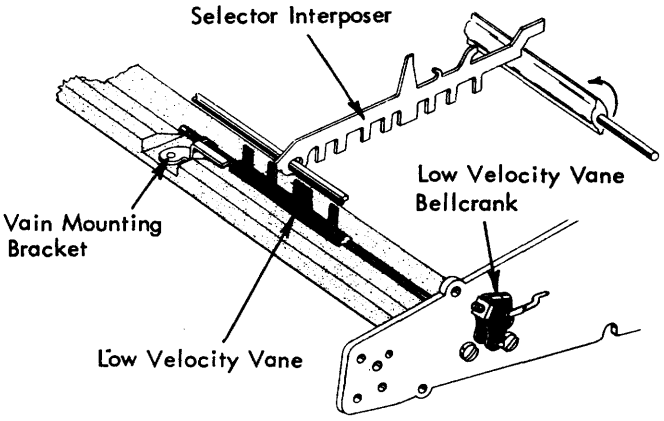


FIGURE 91. Low Velocity Selection

The left end of the low velocity vane pivots in a mounting bracket fastened to the top surface of the front keylever guide comb support. The right end of the vane pivots in a hole in the right hand keyboard sideframe. (The vane extends only halfway across the keyboard.) The lateral position of the vane is controlled by a C-clip located on each side of the vane mounting bracket.

The low velocity vane bellcrank is fastened to the right end of the vane outside the keyboard sideframe (Fig. 91). Rotation of the vane and bellcrank, during a low velocity operation, creates a pull on the link causing the low velocity latch to rotate counterclockwise about its mounting stud. The rotation of the latch swings it out of the operating path of an adjustable stop on the low velocity cam follower. The cam follower and the adjustable stop operates as one piece pivoting about a stud on the keyboard sideframe (Fig. 92). A heavy spring hooked to the adjustable stop and anchored to the latch mounting stud, spring loads the follower in a clockwise direction.

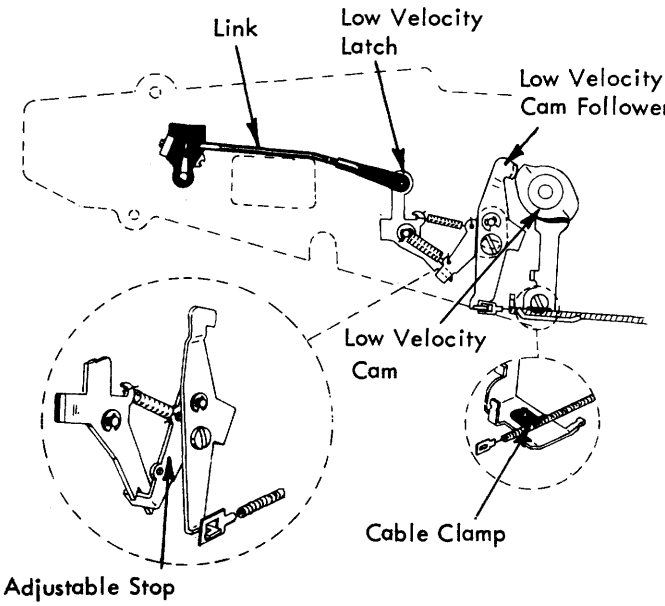


FIGURE 92. Machine At Rest

The velocity control cable is hooked to the lower extension of the follower. The upper extension of the follower is spring loaded against the low velocity cam which is a double-lobed cam set screwed to the right end of the filter shaft directly to the left of the shift interlock cam (Fig. 92). The radial position of the cam is set so that the cam follower will be on the high part of one of the cam lobes when the filter shaft is at rest. Whenever the follower is allowed to follow the contour of the cam towards its low point, a pull will be produced on the velocity control cable by the heavy spring load on the follower.

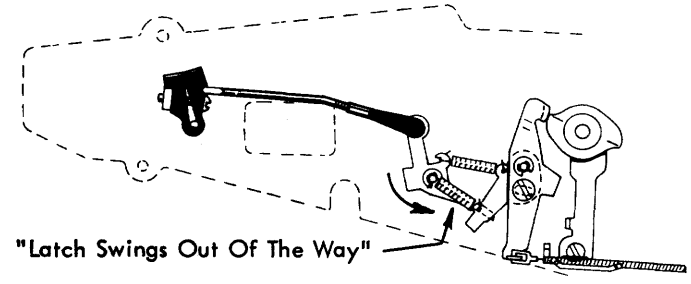


FIGURE 93. Beginning Of A Low Velocity Operation

The cam follower will begin to follow the contour of the cam towards its low point whenever the low velocity latch is rotated out of the path of the adjustable stop as shown in Fig. 93. This occurs each time a low impression character is selected at the keyboard.

Figure 94 shows the low velocity cam follower riding down toward the low dwell of the low velocity cam, thereby causing a pull to be produced on the velocity control cable. Note that the low velocity latch is attempting to restore back to its rest position but cannot fully restore until the cam follower is powered back to its rest position. The latch restoring spring between the latch and the adjustable stop provides the restoring action not only for the latch but for the low velocity vane also.

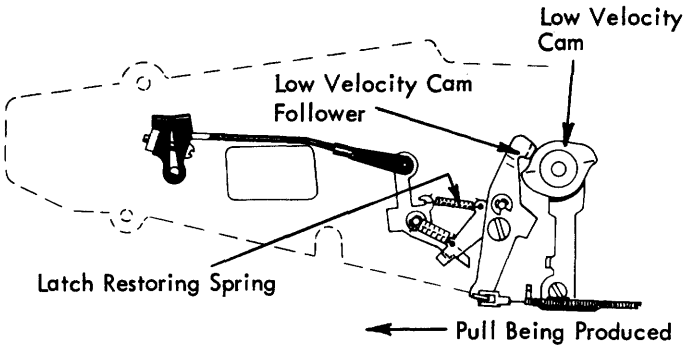


FIGURE 94. Low Velocity Operation

As the low velocity print operation is completing, the follower restores back to the high part of one of the cam lobes on the low velocity cam. This relaxes the pull on the velocity control cable so that the yoke actuating lever spring (Fig. 89) will shift the print cam follower roller back to the right under the high velocity lobe of the print cam.

If a high velocity character is selected at the keyboard, the low velocity latch will remain at rest in the operating path of the stop on the cam follower. The cam follower is restricted from following the contour of the cam towards its low point therefore, no pull is felt on the velocity control cable. The print cam follower roller remains to the right under the high velocity lobe of the print cam and a high velocity print operation results.

To prevent the print cam from interfering with the print cam follower roller as it shifts from one cam lobe to the other during a velocity selection operation, the print cam follower and roller is held disengaged from the print cam until the roller has shifted. This is accomplished by an adjustable stop screw located directly under the rear portion of the print cam follower (Fig. 95). The stop screw, which is threaded through a triangular shaped screws, disengages the print cam follower roller from the print cam as the cam approaches its rest position. The shifting of the roller occurs at the beginning and at the end of a low velocity print cycle which is just when the print cam is leaving or approaching its rest position.

The velocity control cable consist of a thin strand of wire (with eyelets at each end) running through the center of a flexible plastic-coated sheath. The wire slides freely within the sheath transmitting motion form the low velocity cam follower to the yoke actuating lever on the carrier. Both ends of the cable sheath are clamped rigidly to their respective mechanisms by cable clamps (Fig. 90 and 92). Clamping the ends of the sheath

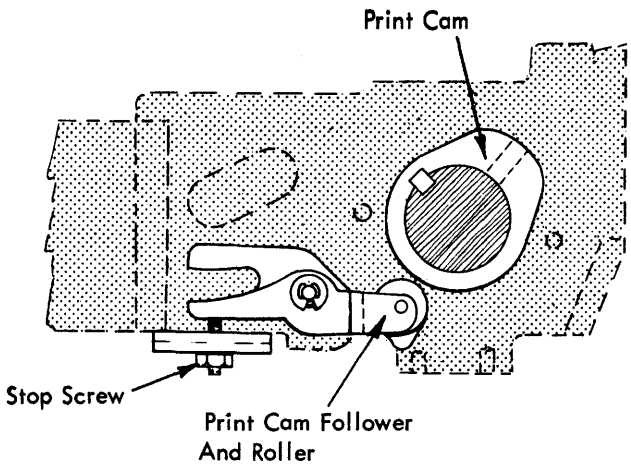


FIGURE 95. Print Cam Follower Stop Screw

allows the velocity control cable to operate efficiently as a motion transmitting device.

The velocity control cable passes through a wire guide located on the left side of the carrier as shown in Fig. 96. The guide retains the cable against the underside of the carrier so that the cable will not hand down and rub on the dust shields (or catch in the mechanism while the dust shields are removed.

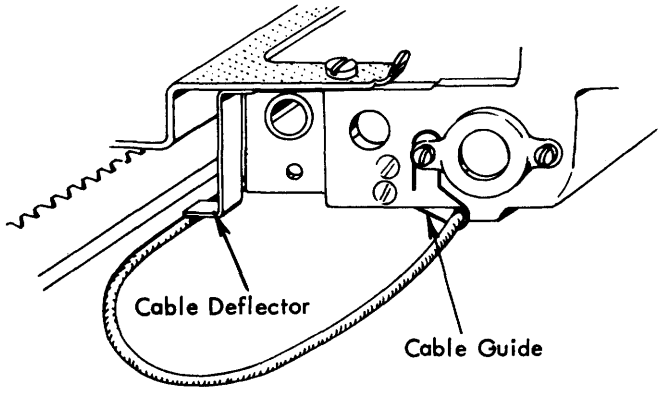


FIGURE 96. Cable Guide And Cable Deflector

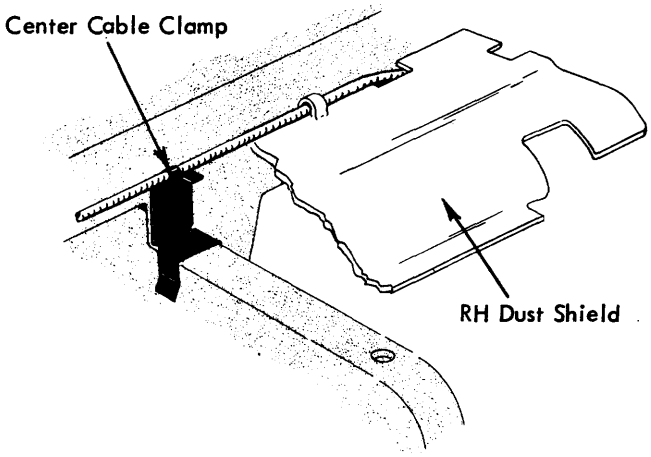


FIGURE 97. Center Cable Clamp

A cable deflector attached to the escapement bracket (Fig. 96) prevents the velocity control cable from getting behind the carrier.

The cable is also clamped along the rear edge of the dust shields by a center cable clamp on the powerframe and a guide lug on the right hand dust shield (Fig. 97). The center cable clamp maintains the cable in its correct lateral position so that the carrier is free to travel the entire length of the carriage without being restricted by the cable.

Impression Control Mechanism - ("Stick Shift")

The impression made by the typing element is determined by the velocity of the typing element upon impact with the paper, the impression control lever is none other than a type head velocity control. By increasing or decreasing the velocity of the typing element with the impression control lever, the impression for all characters can be changed equally regardless of the automatic velocity selection mechanism.

The impression control lever may be positioned by the operator to one of five different impression settings. The lever is held in its selected position by detent notches cut into a detent plate as shown in Fig. 98. When changing the lever to a new setting it must be pushed to the right (disengaging it from the detent notches) before moving it forward or back. Do not ratchet the impression control lever across the teeth on the detent plate as this could cause the plate to become loose.

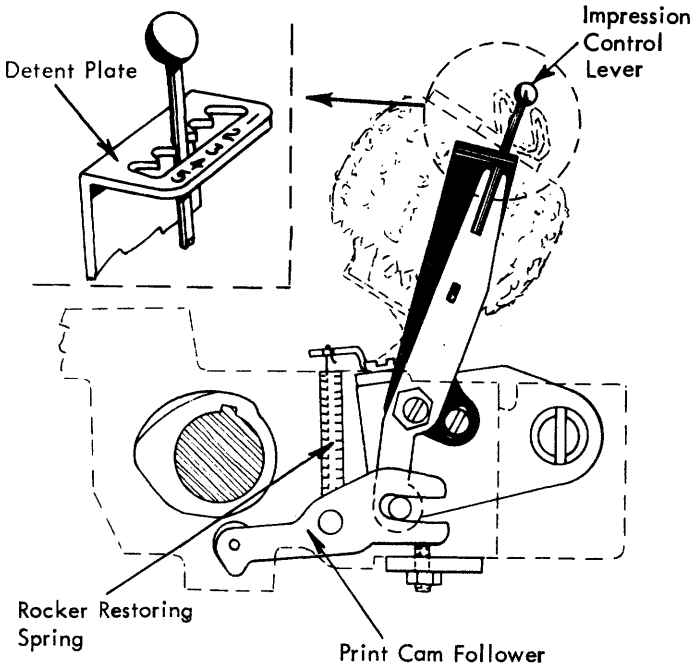


FIGURE 98. Impression Control Mechanism

Changing the position of the impression control lever causes the pin on the lower portion of the lever to move forward or back to a new position in the forked slot of the print cam follower (Fig. 98). Just as in the early style print mechanism, the front to rear position of the pin determines the amount of powered travel that the typehead receives from the print cam follower. This plus the amount of free flight that is in the system determines the velocity of the typehead upon impact with the paper

When the impression control lever is pulled forward to a new position, the pin on the lower extension moves toward the rear in the forked slot of the follower thereby increasing the amount of powered travel that the typing element will receive. The forked slot in the follower is designed so that most of the change in powered travel will be felt as a change in the rest position of the type head and not as a change in the amount of free flight (determined by the limit of powered travel). Throughout the entire range of the impression control lever the free flight of the type head should change slightly (approximately .015). This is necessary in order to maintain the correct timing relation between type head detenting and type head printing as the velocity of the type head is increased or decreased by the impression control lever. In other words, by increasing or decreasing the amount of free flight to compensate for a change in type head velocity, the print time of the type head remains constant in the machine cycle regardless of the impact velocity of the typing element.

Note in Figure 98 that the anvil striker, the restoring cam, and the restoring roller has been eliminated. The rocker is now spring loaded back to rest by a large extension spring (rocker restoring spring). The dual velocity print cam eliminates the need for the anvil and striker. The anvil is now used only as a carrier support for long carriage machines (Fig. 99). The carrier buffers have been eliminated and the bottom of the ribbon feed bracket serves as a buffer.

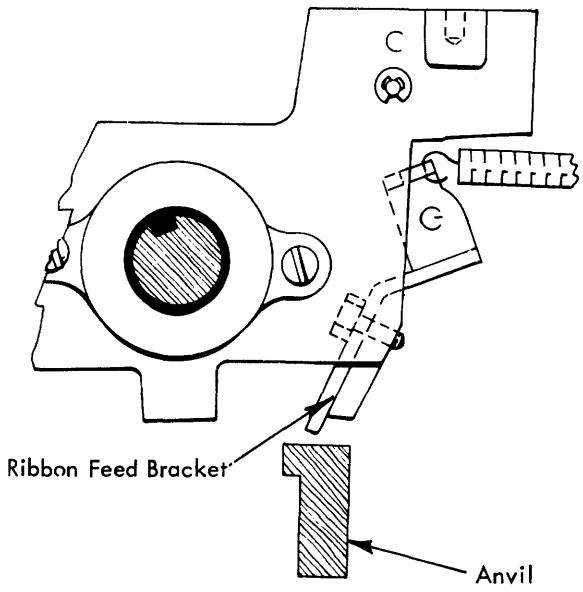


FIGURE 99. Carrier Support (723-725)

PRINT ESCAPEMENT

The escapement mechanism controls the movement of the carrier along the writing line. The print escapement mechanism consists of the escapement bracket assembly, escapement rack, escapement torque bar, escapement trigger lever assembly, and the escapement cam and follower (Fig. 100).

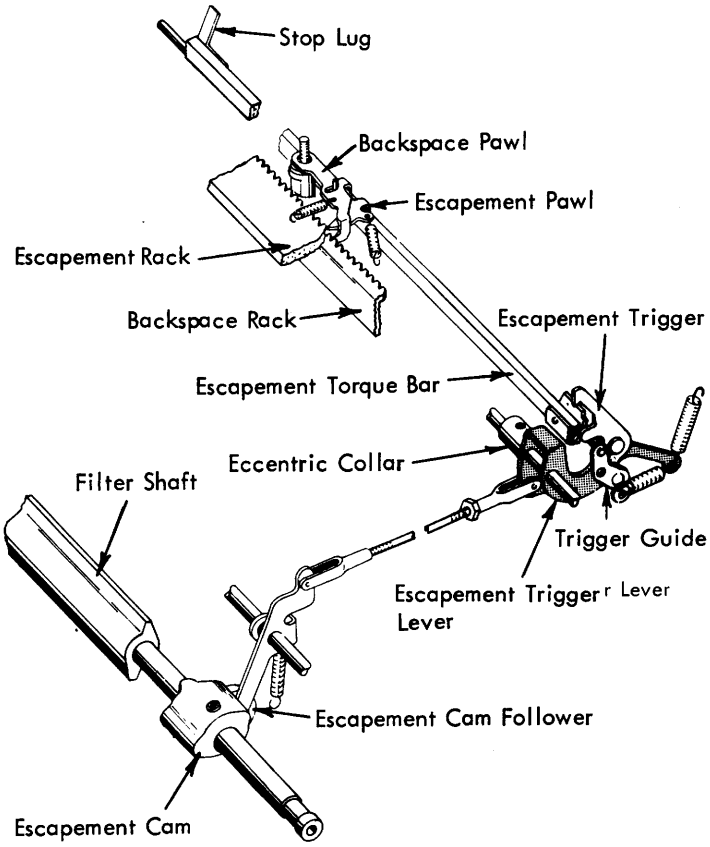


FIGURE 100. Print Escapement Mechanism

Escapement Bracket Assembly

The escapement bracket assembly is a plate attached to the rear of the carrier so that it moves with the carrier. The escapement rack is mounted solidly to the power frame just behind the carrier and beneath the escapement bracket (Fig. 100). A stud at the left rear corner of the escapement bracket mounts the escapement pawl to the bottom of the bracket so that it pivots front to rear. A small spring from the pawl to the bracket exerts a force to the right and forward on the escapement pawl.

Other components are present on the escapement bracket that are only indirectly related to the escapement mechanism. Pivoting on the same stud with the escapement pawl are the backspace pawl, the tab lever, and the tab lever trigger. The tab lever latch mounts on the right side of the escapement bracket.

The escapement pawl contains an elongated hole at its mounting point that allows .022" lateral movement in the pawl. While the pawl is engaged in the rack, the force of the mainspring holds the carrier to the right so that the pivot stud rests against the end of the elongated hole (Fig. 101a).

An escapement operation is obtained by forcing the escapement pawl to the rear out of engagement with the rack teeth (Fig. 101b). Because it is relatively light in weight, the pawl is snapped to the right by the pawl spring as soon as the pawl clears the rack tooth. The escapement pawl is allowed to move back to the front into engagement with the next tooth (Fig. 101c). The carrier then moves to the right until it comes to rest against the escapement pawl (Fig. 101d).

The amount the carrier is allowed to move is determined by the distance from one escapement rack tooth to the next. This is known as the pitch of the machine and is expressed in terms of teeth or spaces per inch. Two pitches are available on the Series 72. They are ten and twelve pitch. The type style to be used is determined largely by the pitch of the machine, because the larger type styles require more space for each character. The pitch and type style together are determined by the operator's preference and by the application for which the machine is to be used.

Having the escapement rack stationary and the escapement pawl moveable is directly opposite to the arrangement on conventional machines. The carrier and escapement pawl must be moved to the left for a backspace operation. Because the backspace pawl is mounted to the escapement bracket, movement of the backspace pawl to the left forces the carrier and escapement pawl to the left.

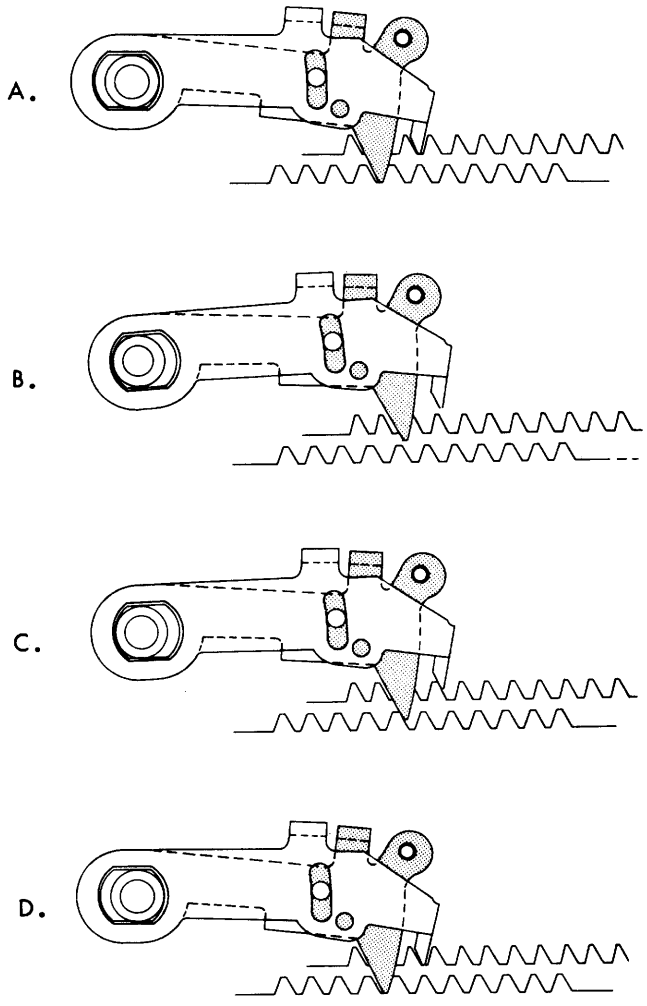


FIGURE 101. Escapement Pawl Operation

The backspace pawl is just above the escapement pawl, but its tooth extends below the escapement pawl. It is held in mesh with the backspace rack by a small extension spring similar to that of escapement pawl. The backspace rack is mounted to the rear of the power frame by shouldered screws through elongated holes in the rack. This mounting arrangement allows lateral movement of the rack. Movement of the rack toward the left forces the backspace pawl to the left to cause a backspace operation.

The backspace operation is pointed out here because of its close association with the escapement mechanism. It is discussed more fully in the backspace section. The backspace pawl is in mesh with its rack when in the rest position. This means that both the backspace and escapement pawls must be removed from their racks in order for the carrier to move to the right.

Figure 101a illustrates a slight clearance between the working surfaces of the backspace pawl and a tooth on the backspace rack when the escapement pawl is holding the carrier. This clearance is necessary for proper operation of the backspace mechanism. The clearance insures that the backspace pawl will properly re-enter the backspace rack at the completion of each backspace operation during a repeat operation (this is when the escapement pawl is holding the carrier and the backspace rack is restoring back to its rest position).

The backspace pawl requires .022" of elongated motion in its mounting hole so that it will operate in unison with the escapement pawl during an escapement operation. Without the elongated motion, the backspace pawl could restore back into the same rack tooth before the carrier began to move during an escapement operation. This could cause partial or half spacing as the two pawls would alternate holding the carrier during an escapement operation.

The backspace and escapement pawl are pinned together so that they will always move together laterally but front to rear independently. The reason for this is covered in the backspace mechanism.

Torque Bar

The torque bar is a flat bar that pivots between the sides of the powerframe just to the rear of the backspace and escapement racks (Fig. 100). Its purpose is to trip the backspace and escapement pawls out of their respective racks. The left end pivots in a hole in the powerframe casting. A small C-clip holds the right end in a large mounting plate on the powerframe.

The rest position of the torque bar on early level machines is determined by a lug on the left end of the torque bar that contacts the tab rack. On late level machines, the rest position is controlled by an adjustable stop which mounts on the right hand tab rack mounting plate and contacts a lug on the right hand end of the torque bar.

The pivot point of the torque bar is near the bottom of the bar. When the torque bar pivots, the top of the bar moves to the rear. The escapement pawl and backspace pawl each have a lug that extends down just behind the torque bar. As the top of the torque bar pivots to the rear, it forces the lugs of the pawls to the rear causing the tips of the pawls to be tripped out of their racks.

Rotation of the escapement torque bar is instantaneous and just sufficient to trip the pawls out of their racks. The torque bar is immediately rotated back to the rest position by an extension spring located at the right end. This allows the pawls to re-enter their racks to limit the carrier movement to one space.

Because of the force required to trip the pawls out of their racks, the torque bar tends to bow toward the front instead of pushing the pawls to the rear. The tendency increases as the carrier moves toward the middle of the torque bar. Bowing of the torque bar could result in a failure of the escapement trip to occur. To overcome this, the pawl pivot stud has a large head that extends down in front of the torque bar to stop any bowing toward the front (Fig. 100). The torque bar actually pries against the pawl stud to trip the pawls from the rack. The head of the pawl stud is flat sided so that the escapement trip in the center of the torque bar can be adjusted to be the same as at the ends. (On late level machines the flat sided head has been replaced with a stud that has an eccentric head.)

On long carriage machines additional support is given to the escapement torque bar to prevent it from bowing to the rear (Fig. 102). A back stop mounted to a stud in the machine powerframe provides the necessary backing.

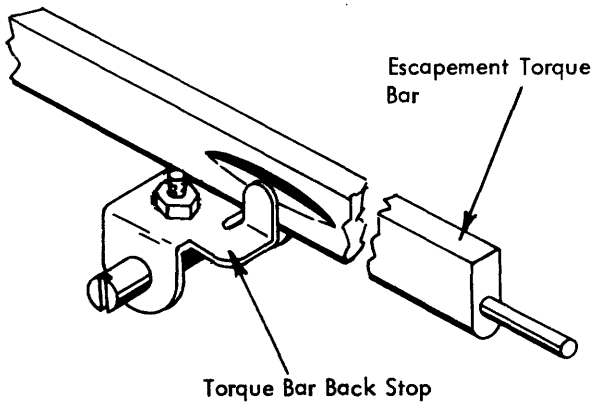
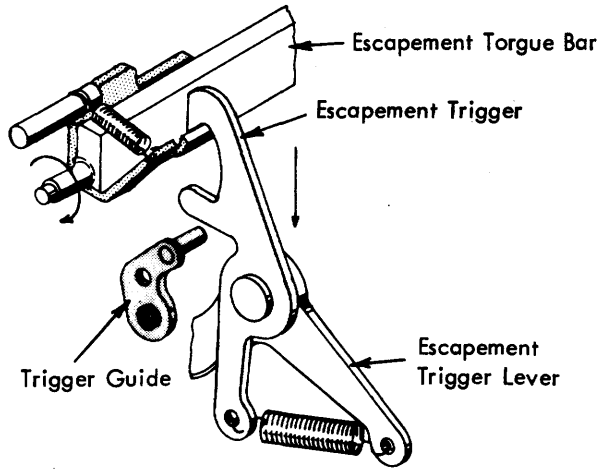


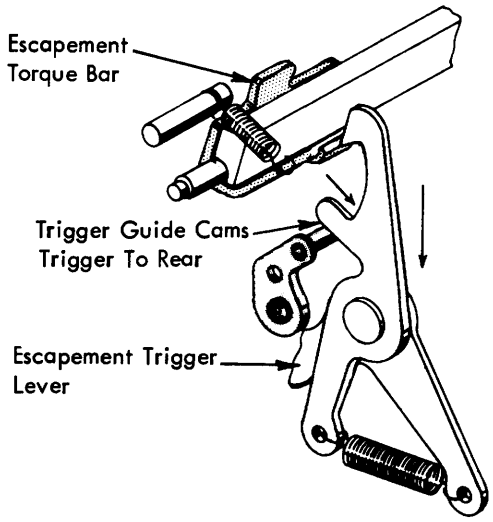
FIGURE 102. Torque Bar Backspace

Escapement Trigger

The escapement trigger operates to rotate the torque bar in order to obtain an escapement operation. The right end of the torque bar contains a lug that extends to the rear. The escapement trigger hooks over the lug and pulls downward to cause the torque bar to rotate (Fig. 103). The trigger pivots on the escapement trigger lever and is held forward over the torque bar lug by an extension spring between the bottom of the trigger and a rear extension of the trigger lever. The trigger lever pivots on a shaft on the operational latch bracket mounted to the rear of the powerframe just below the right end of the torque bar. Downward movement of the trigger lever carries the trigger down to rotate the torque bar. The trigger lever is restored and held in the rest position by an extension spring from the rear of the lever up to a rear extension of the operational latch bracket.



A. REST POSITION

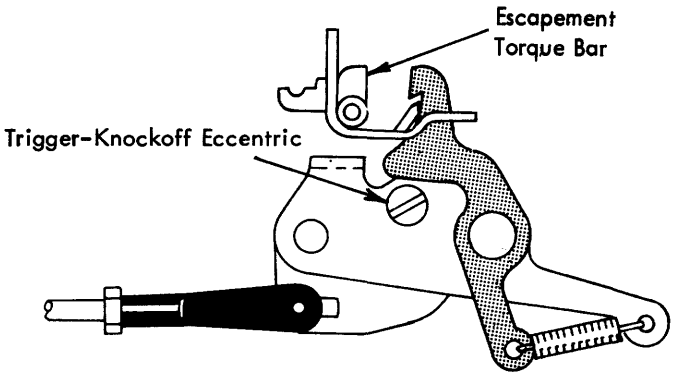


B. ACTIVE POSITION

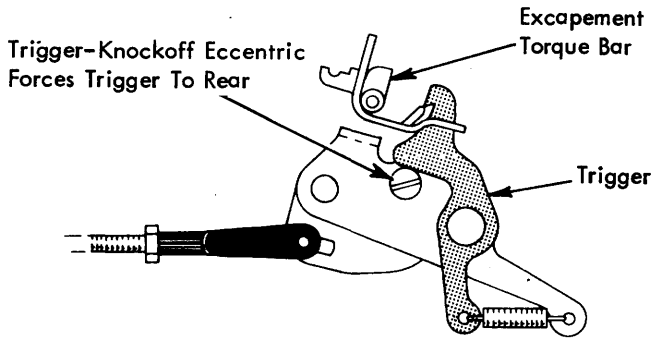
FIGURE 103. Escapement Trigger Operation (old style)

The trigger lever is cam operated; therefore, it can only restore as fast as the cam can rotate from the high point to the low point. The torque bar must be allowed to restore more quickly in order to re-enter the escapement pawl into the correct rack tooth. Delaying the torque bar restoration can result in escapement skipping, especially on 12 pitch machines. To insure that the torque bar can restore quickly enough, the trigger is disengaged from the torque bar lug just after the escapement trip occurs. The operational latch bracket is formed to the rear at the right side. Attached to the inside of this rearward portion is a small plate called the trigger guide (Fig. 103a). The trigger guide has a stud extending to the left just in front of the escapement trigger. A beveled portion of the trigger contacts this stud as the trigger moves down causing the trigger to be cammed to the rear off of the lug on the torque bar (Fig. 103b). The torque bar can then restore without waiting for the restoration of the trigger and trigger lever.

The new style escapement trigger operation is basically the same as the old style except that the trigger knockoff action comes from a screw with an eccentric head that mounts on and moves with the escapement trigger lever (Fig. 104).



A. REST POSITION



B. ACTIVE POSITION

FIGURE 104. Escapement Trigger Operation (new style)

Escapement Cam

Because an escapement operation is necessary each time a character prints, the power to trip the escapement is taken from a portion of the cycle mechanism. A small double lobed cam, called the escapement cam, is attached to the right end of the filter shaft just inside the powerframe (Fig. 100). Each time a cycle operation occurs, the filter shaft operates the cam 180°.

The escapement cam follower pivots on a long pin located in a bracket just to the rear of the filter shaft. A roller at the bottom of the follower is operated to the rear by a lobe of the cam each time a cycle operation occurs. This causes the link at the top of the cam follower to pull forward on the bottom of the escapement trigger lever. The link is connected to the trigger lever below its pivot point; therefore a forward pull causes the trigger to move down at the rear and operate the torque bar.

Because the carrier is relatively light and moves quickly, the escapement trip must not occur before the character prints. If it did, the carrier might move before the character printed or might be moving when the character is printing. This would result in an uneven left margin or poor horizontal alignment and possible smearing of the character depending upon the timing of the trip. To eliminate this possibility, the escapement cam is timed so that the escapement trip occurs just after the type head leaves the platen to restore to rest.

On the 721 only, a small eccentric collar on the pivot pin prevents the pin from bowing forward due to the pull of the escapement trip link. The collar braces the pivot pin by resting against the bracket in which the pivot pin is mounted. Without the collar, some of the trip link motion would be lost in the flexing of the pivot pin. Escapement failure would result, if not enough trip link travel remained to remove the pawls from the rack.

MAINSPRING

Any time the escapement and backspace pawls are removed from their racks, the carrier is pulled toward the right. The mainspring supplies the tension for all movement of the carrier toward the right. It is located at the right rear corner of the machine. Notched lugs of the mainspring cage fit into slots in the backplate (Fig. 105). The cage is turned counterclockwise (facing the rear) so that it locks into position. Tension of the mainspring is changed by turning the entire mainspring cage to a new location. When working with the mainspring, extreme care should be taken to keep it under control.

The center of the backplate contains a ball bearing assembly that supports the rear end of the escapement shaft. The shaft extends to the rear into the mainspring where a hub is setscrewed to the shaft. The inner end of the mainspring is rolled so that it fits into a groove in the hub and supplies rotary force to the hub and escapement shaft.

The escapement shaft extends forward through another ball bearing assembly in the powerframe and has a drum attached at the forward end. This drum is called the cord drum gear and is spirally grooved to accept the escapement/tab cord. The grooves prevent the cord from piling up on the drum insuring uniform tension and minimum wear.

The escapement/tab cord is a small, round, nylon-covered linen cord. The drum end of the cord is knotted and fits into a slot at the rear of the drum. The cord makes several turns around the drum and rides up over a guide roller before passing through the right side of the machine (Fig. 105). Just outside the machine the cord passes around a pulley and extends back to the left where it is attached to a hook on the carrier assembly. As the mainspring turns the escapement shaft, the drum winds up the cord to move the carrier to the right.

More than one operation is performed by the escapement shaft. The gear teeth at the front of the cord drum are involved in both the carrier return and tabulator operations. Between the powerframe and the backplate is another cord drum with a cord attached to it similar to the escapement/tab cord (Fig. 105). This cord is attached to the left side of the carrier and exerts a pull to the left during carrier return. These operations are fully covered in their particular sections. However, the carrier return cord drum is significant in the escapement mechanism, because it must pay out cord in order for the carrier to move to the right. Likewise, the escapement/tab cord must be paid out from its drum in order for the carrier to return to the left.

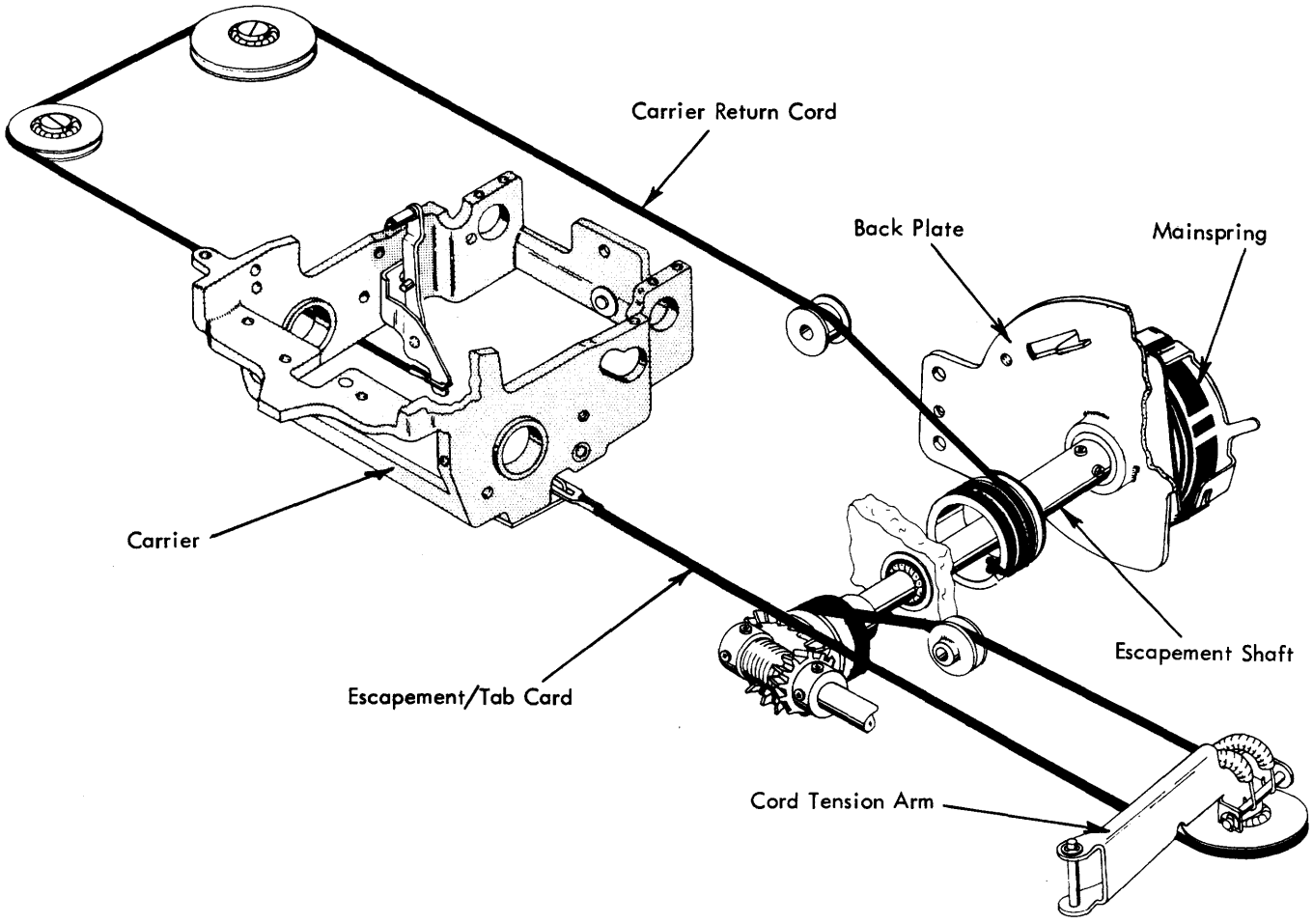


FIGURE 105. Mainspring And Cord System

The constant jerk and pull to which the cords are subjected may tend to stretch them slightly. This could cause the cords to become slack creating erratic movement of the carrier. A method has been provided whereby the slack is automatically removed from both cords. Outside the powerframe, the pulley that guides the escapement cord is mounted to the cord tension arm pivoted at the front (Fig. 105). A pair of spiral springs arched between the arm and a pin on the powerframe apply a constant pressure toward the right. The pressure is sufficient to keep the slack out of the escapement cord. This, in turn, rotates the escapement shaft enough to keep the carrier return cord tight.

OPERATIONAL CAMS AND CONTROL MECHANISM

All powered service operations are driven by the operational cam shaft located on the right side midway back in the machine. The powered operations are the spacebar, backspace, carrier return, indexing, and shift. The tabulator is manually operated and the only non-powered service operation.

The spacebar, backspace, carrier return, and indexing mechanisms are operated by two cam assemblies mounted on the operational cam shaft. The shift is driven by a spring clutch at the right end of the shaft outside the powerframe.

Operational Cams

The two operational cams are located on the right side of the operational shaft just inside the powerframe (Fig. 106). The left hand cam is a double lobed cam that requires only 180° rotation to complete one operation. Its purpose is to power the spacebar and backspace mechanisms. The right hand cam is a single lobed cam requiring 360° rotation to complete one operation. It powers the engaging of the carrier return mechanism and operates the indexing mechanism.

Each operational cam must power two functions. The mechanism to be operated is determined by a selection system released by depressing the desired keylever. The selection operation is discussed later in this section.

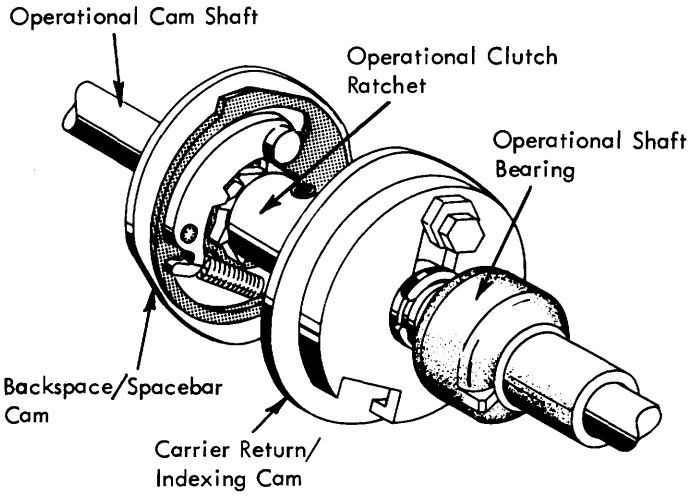


FIGURE 106. Operational Cams

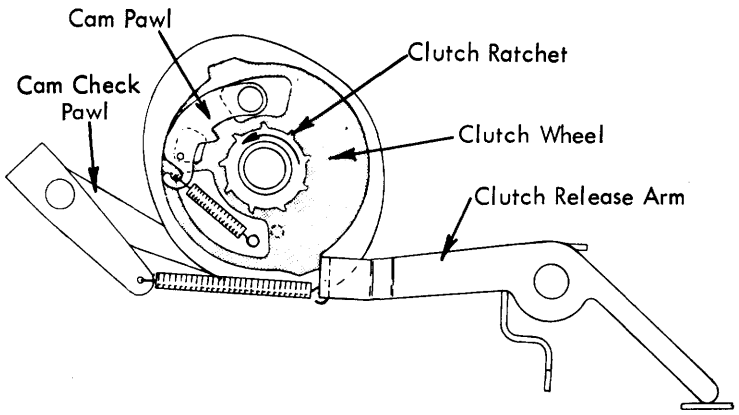


FIGURE 107. Operational Cams At Rest

Both the single and the double lobed cams have the same rise from the low point to the high point. The double lobed cam completes an operation sooner than the single lobed cam, because it requires only 180° rotation compared to 360° for the single lobed cam. The faster double lobed cam is used in the spacebar mechanism because the spacebar action must be as fast as the print action in order to maintain typing rhythm. The backspace operation must operate quickly in order to provide rapid positioning of the carrier; therefore it also employs the double lobed cam. Both the carrier return and indexing keylevers have a repeat/non-repeat feature. When operated in the repeat position, they cause rapid indexing of the platen. Operating too rapidly could create inaccurate indexing due to platen overthrow and failure of the index pawl to restore quickly enough; therefore the slower single lobed cam is used to operate the carrier return and indexing mechanisms.

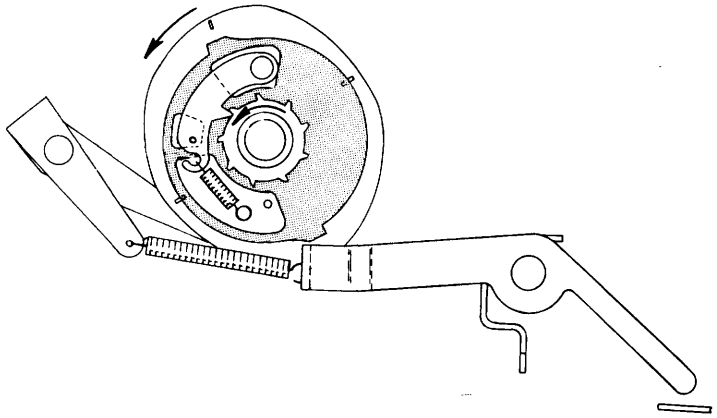


FIGURE 108. Operational Cam Active

A ratchet, called the operational clutch ratchet, is set-screwed to the operational cam shaft (Fig. 106). The cam assemblies are C-clipped into position on shoulders of the ratchet, one on each end. The operational cam shaft and clutch ratchet are allowed to turn while the cam assemblies remain stationary. Steel sleeves fitted into the cam bodies act as bearings for the cams.

A cam rotates only when a particular operation is desired. In order for the cam to rotate it must be "locked" to the operational clutch ratchet. Each cam has a pawl mounted to it in such a manner that the pawl can be pivoted into engagement with the rotating operational clutch ratchet (Fig. 107). Whenever the pawl is permitted to engage in the clutch ratchet, the cam will rotate with the ratchet (Fig. 108).

The operational cam pawl is prevented from engaging the clutch ratchet by a disc called the clutch wheel (Fig. 107). The clutch wheel is attached to and forms a part of the cam assembly. The disc is mounted to the cam assembly so that the cam pawl operates through an oversized hole in the disc. The clutch wheel is free to rotate on the cam assembly, but its movement is restricted by contact with the cam pawl. A pin at the tip of the pawl fits into another hole in the clutch wheel that has one side beveled (Fig. 108). Movement of the cam while the clutch wheel is held stationary causes the pin on the pawl to slide up the beveled side of the hole and disengage the pawl from the ratchet (Fig. 109).

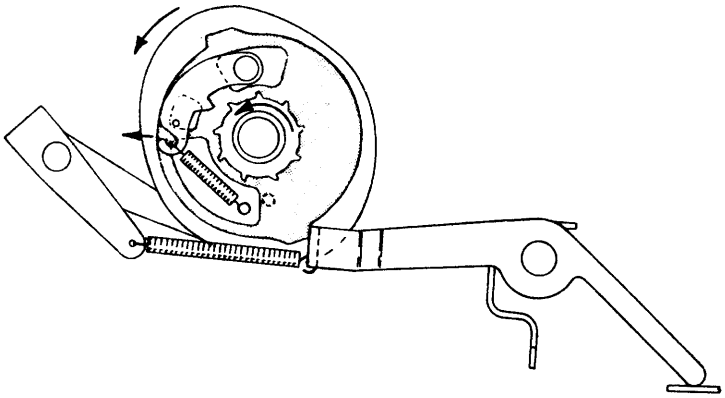


FIGURE 109. Clutch Pawl Being Disengaged

Movement of the clutch wheel is controlled by the clutch release arm pivoted on a shaft below and to the rear of the cam assembly (Fig. 107). The clutch wheel has a tooth that contacts the end of the clutch release arm as the cam assembly rotates. The clutch release arm stops the movement of the wheel. Further rotation of the cam causes the pawl to be disengaged from the ratchet as described. When the clutch release arm is moved down, the clutch wheel is released (Fig. 108). The cam pawl spring is then allowed to pull the pawl into engagement with a tooth of the clutch ratchet by merely rotating the clutch wheel out of the way. As soon as the clutch release arm is allowed to restore, it is pulled by its spring back up into position to contact a tooth of the clutch wheel. There are only two such teeth for the double lobed cam and only one for the single lobed cam. The cam assembly is driven 180° for the double lobed cam and 360° for the single lobed cam before the clutch release arm contacts and stops a tooth of the clutch wheel. Rotation of the double lobed cam through 180° and the single lobed cam through 360° allows the high point of the cam to operate the particular function involved.

The cam pawl spring tries to pull the pawl into engagement with the clutch ratchet. Unless the cam is held in position after the pawl is disengaged from the ratchet, the spring will cause the cam to creep backward allowing the pawl to partially engage the ratchet and create a loud buzzing sound. It cannot fully engage the ratchet because it would be disengaged immediately as previously described.

To prevent backward creep of the operational cams and the resulting noise, a cam check pawl is provided for each cam. The check pawl engages a notch in the cam assembly as soon as it has rotated far enough for the cam pawl to be disengaged from

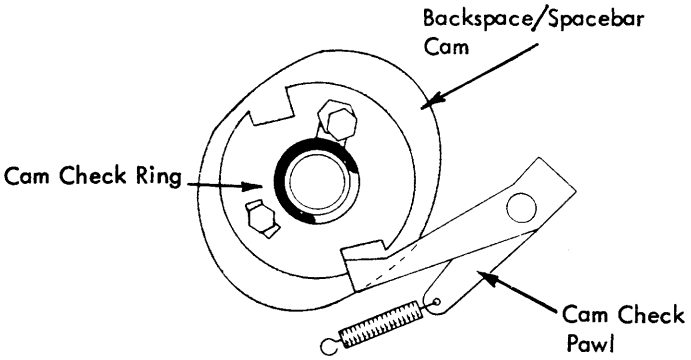


FIGURE 110. Operational Cam Check Pawl

the ratchet (Fig. 110). The check pawls extend to the rear from a pivot pin located just in front of the cams. An extension spring from each check pawl to the clutch release arm maintains the two parts in the rest position. The notch engaged by the cam check pawl is in a disc-shaped part called the cam check ring. The ring is attached to the cam by two screws. An eccentric collar on one of the screws facilitates the adjustment of the cam check ring.

Operational Control Mechanism

The operational control mechanism is a compact unit contained in a bracket located under the operational cams. The purpose of the mechanism is to select the function to be operated, control the movement of the cam, and transfer the cam motion to the selected operation. To fulfill these three purposes, each cam requires four basic parts in its control mechanism. They are: an interposer to select the operation and to help with the cam control, a clutch release arm to control the cam, a restoring device for the interposer, and a cam follower to transfer the cam motion to the operations. The principle of operation for both cam control mechanisms is the same; however, the parts design differs slightly.

1. Interposers

Each mechanism operated by the cams requires an interposer to select the mechanism to be operated and to cause the cam to be engaged. These four interposers operate front to rear through slots in the operational control bracket and are latched forward on an adjustable guide attached to the front of the bracket (Figs. 111 and 112). A position for a fifth interposer is present in the operational control bracket. The fifth interposer is used in special applications of the machine and is not normally included in the mechanism. The interposers are each spring loaded to the rear by an extension spring between the interposer and the rear of the operational control bracket. Attached to the front of each interposer is a small pivoting latch. The latch is spring loaded upward against the interposer so that the top portion forms a hook for the interposer. The latch hooks under the keylever pawl guide bracket to hold the interposer forward. When the interposer is pushed down, the latch clears the bracket allowing the interposer to be pulled to the rear to perform its function. The latch pivots on the interposer so that the entire interposer will not have to move down to allow the latch to relatch on the guide bracket as the interposer restores to the

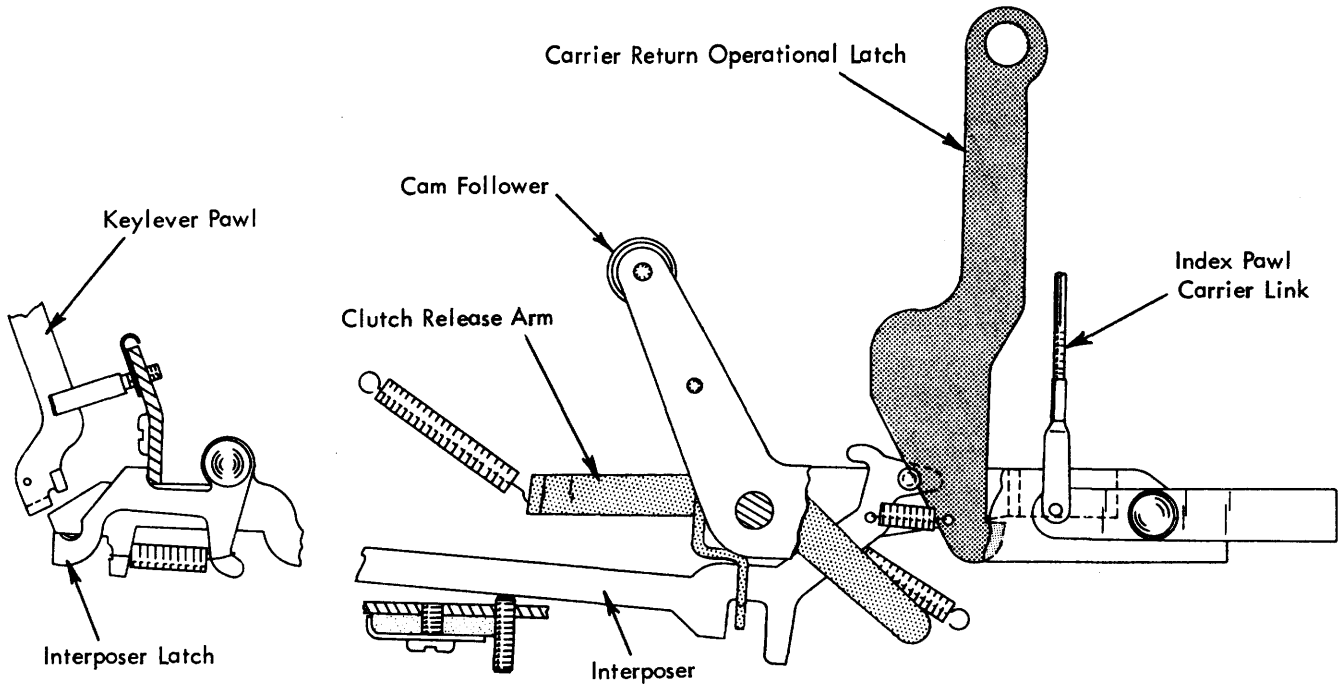


FIGURE 111. Carrier Return/Indexing Operational Control Mechanism

front. This insures positive relatching as the latch can very quickly snap back up into the latched position after it has been cammed down by the guide bracket.

The carrier return, indexing, backspace, and spacebar keylevers each have a pawl attached at the rear that extends down through a slotted guide stud to a position just above its own particular interposer (Figs. 111 and 112). Depression of one of the keylevers causes a lug at the bottom of the keylever pawl to depress the interposer and release it to the rear.

2. Clutch Release Arm

Two clutch release arms are present, one for each cam. The arms are identical in operation but slightly different in design to conform to their position in the machine. The release arms pivot on a shaft at the rear of the cams (Fig. 107).

Each clutch release arm has three extensions from the pivot. The forward extension blocks the clutch wheel to disengage the cam pawl from the ratchet. The lower extension is contacted

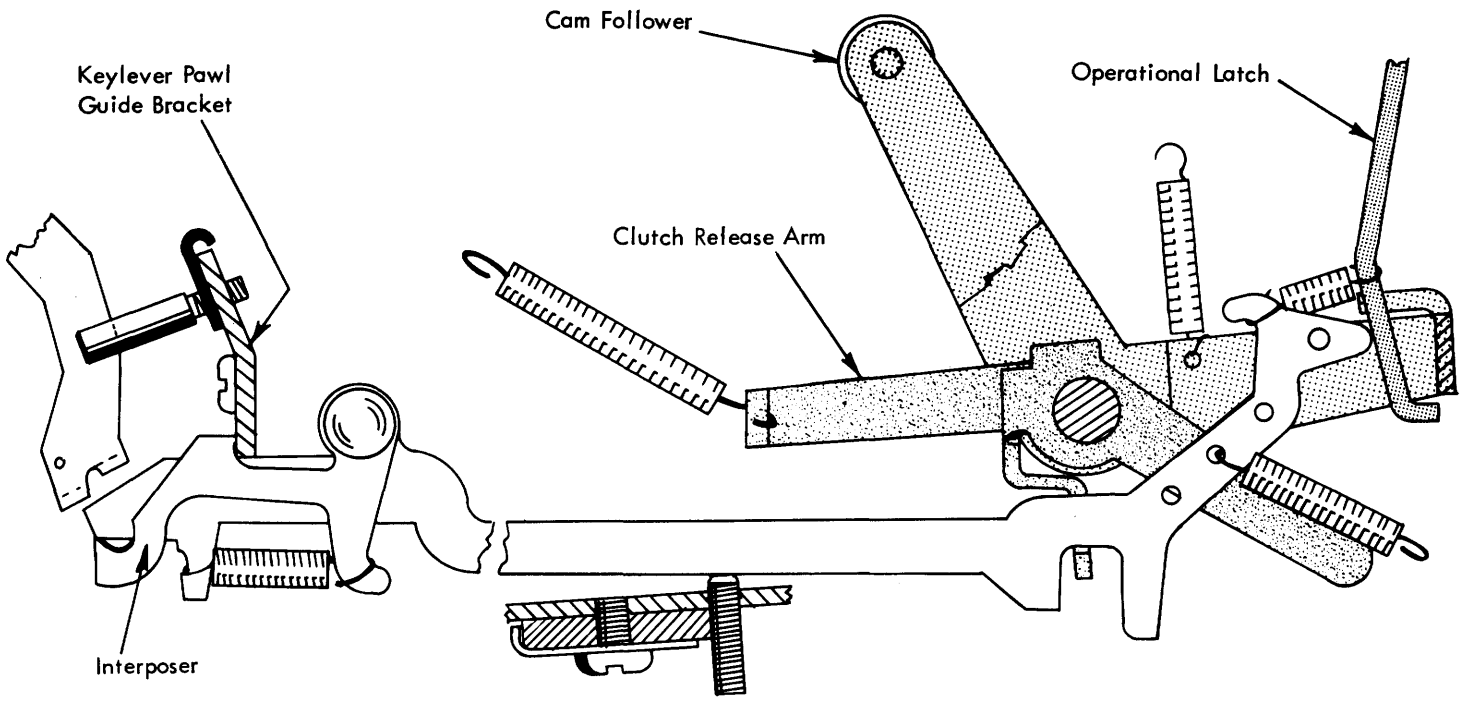


FIGURE 112. Back / Spacebar Operational Control Mechanism

by an interposer. The interposer pivots the forward extension down out of the path of the clutch wheel to allow the cam to be driven. In the rest position, the rear extension of the clutch release arm contacts the bottom of the operational control bracket (Fig. 107). This controls the "bite" between the release arm and the clutch wheel to insure positive release action.

The clutch release arm is restored and held at rest by an extension spring between the forward extension and the cam check pawl (Fig. 107).

3. Cam Follower

Each cam has a cam follower designed to convert the rotary motion of the cam into vertical linear motion at the rear. The cam followers are bellcrank devices mounted just to the rear of the cams (Figs. 111 and 112). The cam followers pivot about the same pivot shaft that supports the clutch release arms. A roller at the top of each cam follower is in continuous contact with its particular cam.

The carrier return/indexing cam follower extends to the rear where it is designed in the form of a bail (Fig. 111). When the cam operates, the rear of the follower is moved down as the roller is forced to the rear by the cam. A link attached near the rear of the cam follower operates the indexing mechanism. Each time the cam operates, an indexing action occurs. A return spring in the indexing mechanism restores the cam follower and holds the roller against the cam.

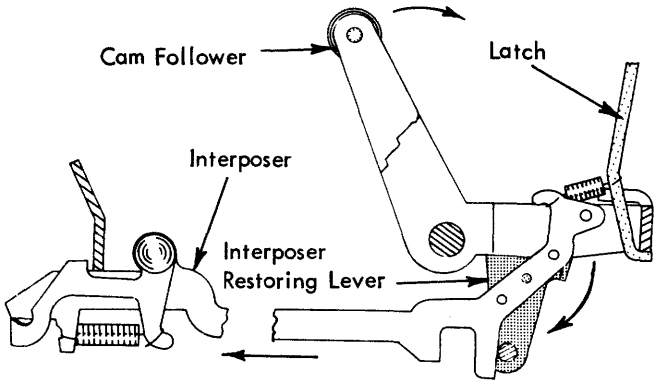


FIGURE 113. Interposer Restoring Operation

The backspace/spacebar cam follower is designed to operate three mechanisms. In normal applications of the machine, only the backspace and spacebar are present. In input/output typewriters the tabulator is also powered by the cam. Because it has three positions to operate, the cam follower must be wider than that of the carrier return. An extension spring from the cam follower to the operational latch bracket acts to restore the cam follower and maintain the position of the roller against the cam (Fig. 112).

Pressure of the cam follower roller against the cam is essential for proper operation of each cam. The spring tension forces the cam follower from the high point to the low point of the cam. This accelerates the cam enough to give the overthrow required to disengage the cam pawl from the ratchet and to allow the check pawl to engage the check ring.

4. Interposer Restoring Lever

After an interposer has been released to the rear to begin an operation, it must be restored forward so that the clutch release arm may pivot back to its rest position to stop the action of the cam.

The interposers are restored by a bail-shaped part that pivots between the sides of the operational control bracket and is located just to the rear of the interposers (Fig. 113). A lug at each side of the interposer restoring lever is in contact with the cam followers. Operation of either cam follower pivots the bottom of the restoring lever forward (Fig. 113). The restoring lever, in turn, forces the interposer forward where it can latch in the rest position.

5. Operational Selection

In addition to causing cam release, the interposer must set up the mechanism to be operated. This is true of all except the indexing interposer. Because the indexing mechanism is directly connected to the cam follower (Fig. 111), merely releasing the cam results in an indexing operation.

The backspace, spacebar, and carrier return interposers each have a small, hook-like latch resting against the rear of the interposer (Figs. 111 and 112). A small extension spring maintains the latch against the interposer. Each latch is suspended from its own particular mechanism located above the rear of the interposer. Pulling any one of the latches down causes it to operate the mechanism involved. The hook portion of the operational latches rests just in front of the cam follower. When one of the interposers is snapped to the rear, the latch for that interposer is pushed to the rear where it hooks under the cam follower. The cam follower is then operated to pull down on the latch and actuate the selected mechanism (Fig. 113).

As the cam follower is operated, the interposer is restored to the front by the restoring lever. This causes the spring between the interposer and the operational latch to extend (Fig. 113). Extending the spring tends to pull the latch out from under the cam follower before the operation is completed. To insure a complete operation, the latch is locked to the rear as soon as

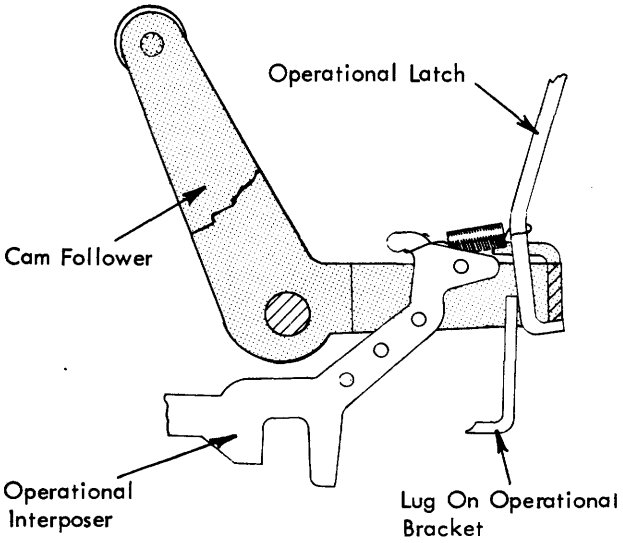


FIGURE 114. Operational Latch Locked To The Rear

the operation begins. As the latch is pulled down by the cam follower, it moves down behind a lug of the operational control bracket (Fig. 114). The lug prevents any forward movement of the latch until the operation is completed and the cam follower restores. The latch is then pulled forward into its rest position against the interposer. The carrier return latch is not locked to the rear, because its beveled forward edge permits it to remain in contact with the interposer throughout the operation (Fig. 111).

6. Operating Sequence

As the desired keylever is depressed, a lug of the keylever pawl contacts an interposer forcing it down to release it from the guide bracket. The interposer is pulled to the rear by its spring. A lug on the interposer contacts the clutch release arm rotating it down at the front to allow the cam clutch to be engaged. At the same time, the interposer forces its selective latch to the rear pushing it under the cam follower. The cam is driven by the clutch ratchet causing the cam follower to move from the low point to the high point of the cam. Movement of the cam follower pulls down on the operational latch to power the mechanism and, at the same time, actuates the interposer restoring lever to restore the interposer forward. The clutch release arm restores into the path of the clutch wheel ready to disengage the cam clutch. The cam follower passes the high point of the cam and restores to the rest position as it reaches the low point. The operational latch is snapped forward against its interposer into the rest position and the operation is completed.

Repeat/Non Repeat

All four mechanisms operated by the operational cams are equipped with a repeat/non repeat feature. When the keylever is depressed to its first limit, only a single operation occurs. Further depression of the keylever causes the first limit to yield and allow a repeating action. Three different methods are used to determine the first limit for the keylevers. Each of the methods is discussed in the mechanism to which it applies.

In order to obtain either a single or a repeating operation, two lugs are needed on each keylever pawl. The front lug is just above the tip of the interposer when both are at rest (Fig. 115a). When the keylever is depressed, this lug causes the interposer to be released. As the interposer snaps to the rear, it moves out from under the lug of the keylever pawl. If the keylever is held depressed to its first limit, the end of the interposer will contact the lug of the keylever pawl as it restores to the front. The interposer will force the pawl forward and relatch on the keylever pawl guide bracket (Fig. 115b). When the keylever is released, the keylever pawl will then reset to the rear above the interposer.

When the keylever is not released, the rear lug on the keylever pawl is now in position to release the interposer (Fig. 115b). Depression of the keylever past its first limit causes the rear lug to trip the interposer from the latch bracket (Fig. 115c). The interposer moves to the rear as before, but it cannot move far enough to get out from under the rear lug of the keylever pawl. Each time the interposer is restored to the front it is prevented from latching because the lug of the keylever pawl continues to hold it down. Because the interposer cannot latch, it is snapped back to the rear by its spring to operate the clutch release arm and to push the operational latch under

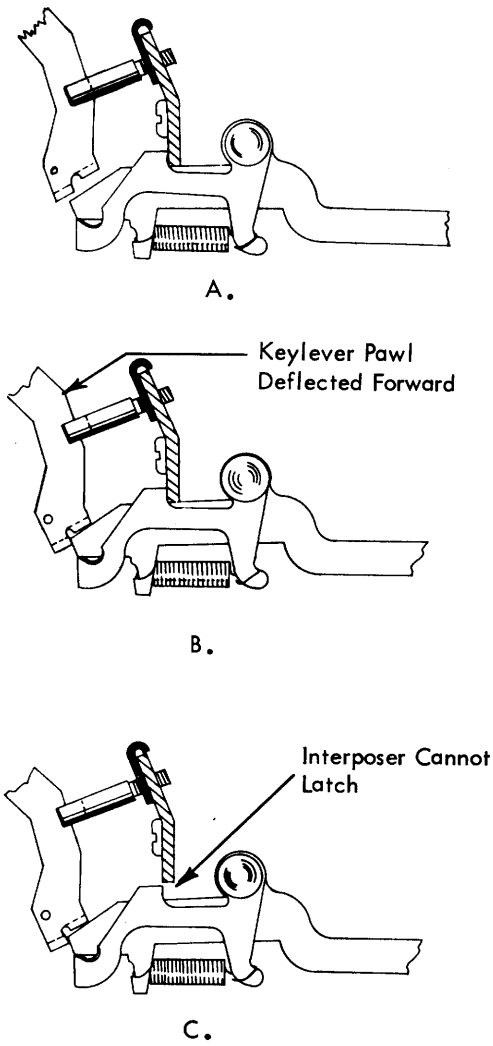


FIGURE 115. Keylever Pawl Operation

the bail of the cam follower. Each time the cam operates, this action of the interposer is repeated causing a continuous operation. There is no operational latch for the indexing mechanism; therefore the index interposer only operates the clutch release arm to cause a continuous cam operation.

When the carrier return keylever is operated in the repeat position, it is always for the purpose of creating a repeat indexing operation with the carrier at the left margin. This is more convenient for the operator than using the index keylever which was designed primarily for indexing with the carrier away from the left margin.

One operation of the carrier return interposer causes an index operation plus a carrier return operation. Only an indexing operation is desired thereafter. Each time the carrier return interposer moves to the rear, the operational latch is moved under the cam follower to cause a carrier return operation. It is undesirable and unnecessary to have a repeating carrier return action at the left margin, because of the shock of the carrier repeatedly striking the left margin. Therefore, the carrier return interposer is not used for a repeating operation.

When the carrier return keylever is depressed and held down for a single operation, the interposer operates to the rear and is restored as on the other mechanisms. The interposer forces

the keylever pawl forward slightly in order to latch. The rear of the keylever pawl, instead of being above the carrier return interposer, is formed to the right above the index interposer (Fig. 116). Depression of the carrier return keylever past its first limit causes the keylever pawl to release the index interposer and hold it down. The index continues to repeat as long as the keylever is held in the repeat position.

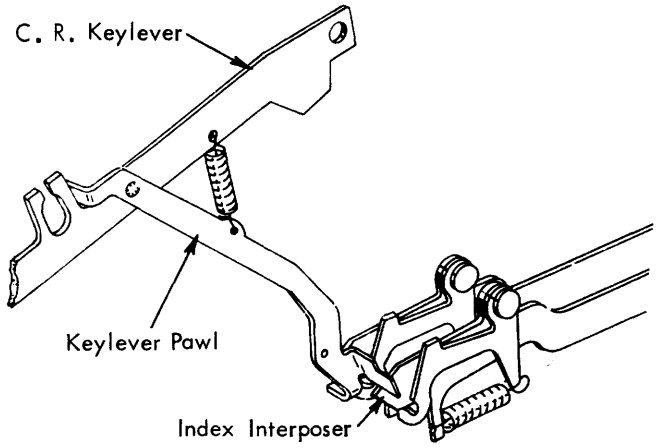


FIGURE 116. Repeat Carrier Return Operation

SPACEBAR

The spacebar mechanism provides the operator with a means of moving the carrier to the right one space at a time without typing a character on the paper. It is used mainly for spacing between words, but it may also be used to space the carrier quickly to any point toward the right on the writing line.

The spacebar operates by tripping the escapement pawl and backspace pawl out of their racks as on a print escapement operation. The two escapement operations are identical except for the method of actuating the escapement trigger lever.

Spacebar Lever Mechanism - Early Level

The spacebar mounts on the keyboard just in front of the keybuttons. It is suspended on two arms extending forward from the spacebar shaft (Fig. 117). The spacebar shaft is shouldered at the left end and pivots in a hole in the left keyboard sideframe. The right side pivots on a pivot screw in the right keyboard sideframe. Vertical motion of the spacebar causes the spacebar shaft to rotate.

The spacebar contains a metal stem pressed into the body of the spacebar (at the center). The stem extends down into a slot in the spacebar return spring. The spacebar return spring is a flat, leaf-type spring that is mounted to the front keylever guide comb support by two screws and performs the function of restoring the spacebar to its rest position. This spring also guides the bottom of the spacebar stem to prevent tipping of the spacebar forward or back. Additional support is provided by a lever that extends forward from the spacebar shaft. A guide stud on the lever fits through an elongated hole in the spacebar stem.

Attached to the right end of the spacebar shaft is the spacebar operating arm which extends to the rear through a slot in the front keylever guide comb (Fig. 117). When the spacebar is depressed the operating arm is raised. The arm contains a slot in the end that fits into a fork in the forward end of an intermediate lever called the spacebar lever. When the spacebar is depressed, the rear of the spacebar lever and the attached spacebar lever pawl are lowered. The spacebar lever pawl then pushes down on the spacebar interposer causing the interposer to unlatch from the keylever pawl guide bracket. The interposer snaps to the rear, releases the spacebar/backspace cam, and pushes the spacebar operational latch under the cam follower.

The spacebar latch is mounted to the spacebar latch lever by a ball shouldered rivet to permit free movement of the latch (Fig. 118). The spacebar latch lever is mounted on a pivot pin on the operational latch bracket assembly which is attached to the rear of the powerframe just in front of the mainspring.

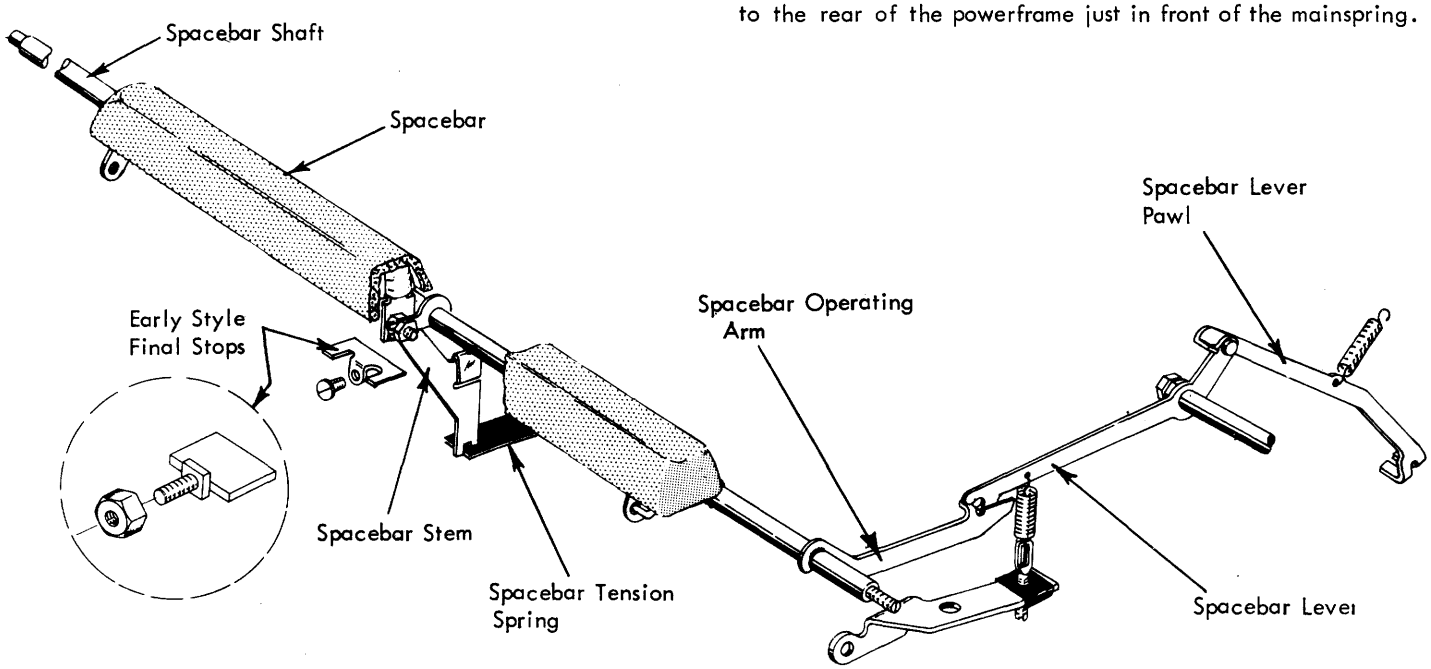


FIGURE 117. Spacebar Lever Mechanism

When the spacebar latch is operated down by the cam follower, the spacebar latch lever pivots about its pivot pin. An adjusting screw on the latch lever contacts the trigger lever causing the trigger lever to rotate about its pivot pin (Fig. 118). This causes the trigger on the trigger lever to rotate the escapement torque in the same manner as it does during a print escapement operation. Thus the escapement and backspace pawls are removed from their racks and an escapement operation takes place.

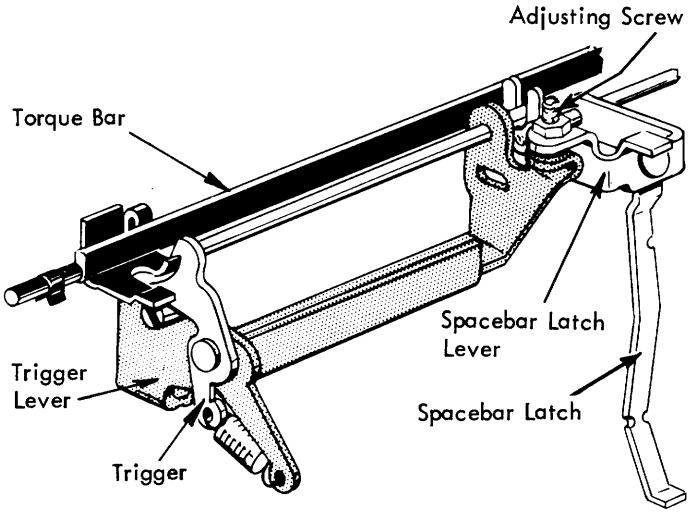


FIGURE 118. Spacebar Latch Operation

Spacebar Stops

Two styles of final stops have been produced and used on the early style spacebar mechanism. Both styles attach to the middle of the keylever guard at the front of the machine and function by blocking the downward travel of the spacebar stem (Fig. 117).

Upward travel of the spacebar is limited by the spacebar stem. A step in the stem contacts the bottom of the spacebar shaft

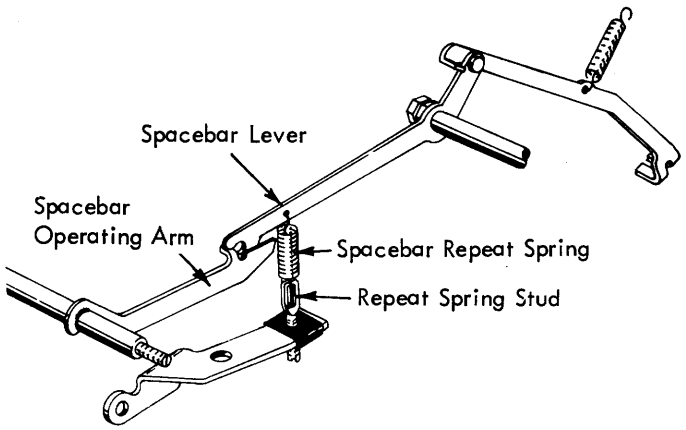


FIGURE 119. Spacebar Repeat Stop - New Style

(Fig. 117). A vinyl sleeve on the stem cushions the spacebar in the rest position to prevent noise.

The downward travel of the spacebar is limited at two positions. The first limit allows a single operation only. Further depression of the spacebar causes the first limit to yield to permit a repeat operation. A final limit prevents further depression.

An extension spring at the front of the spacebar lever acts as a first limit for the mechanism (Fig. 119). The lower end of the spring is formed into a long hook. The hook is connected into an elongated slot in the repeat spring stud. As the spacebar is depressed, the front of the spacebar lever and the repeat spring are raised. The spacebar reaches the first limit when the repeat spring contacts the top of the elongated slot in the repeat spring stud. Further depression of the spacebar causes the repeat spring to yield and allow the spacebar lever to move into the repeat position.

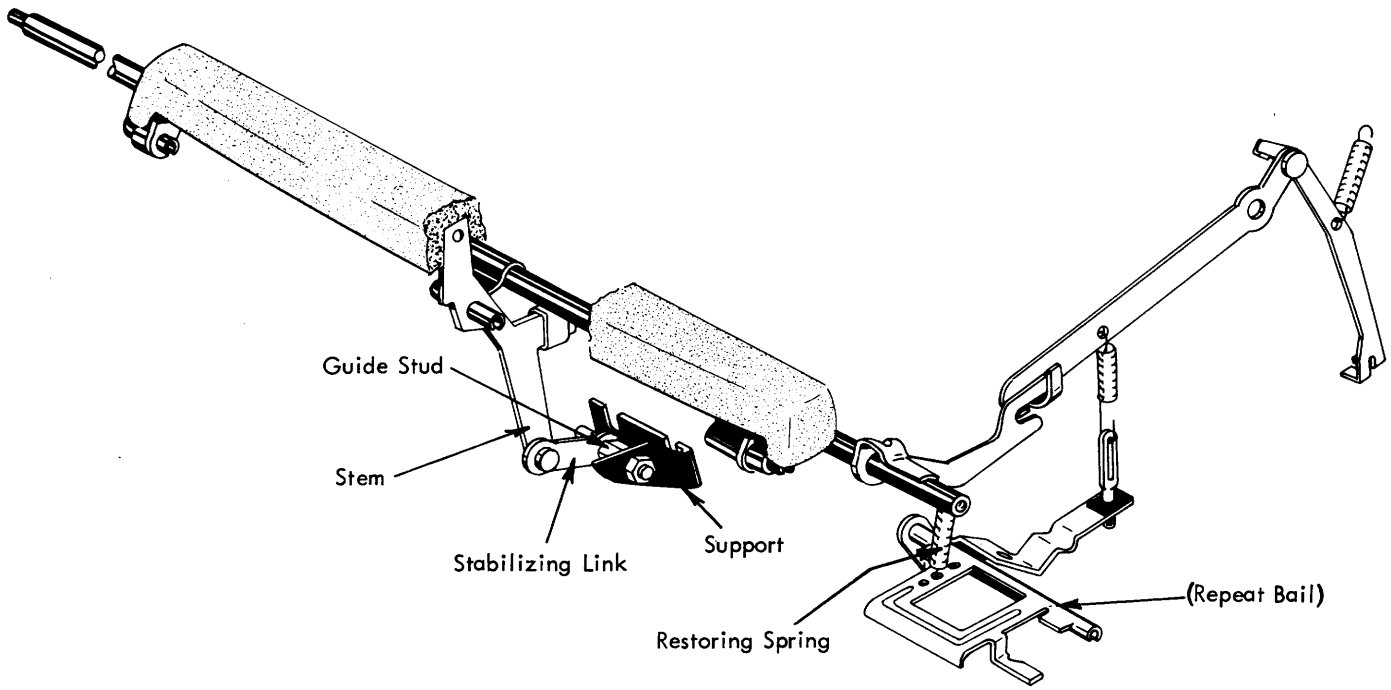


FIGURE 120. Spacebar Lever Mechanism - Late Level

Spacebar Lever Mechanism - Late Level

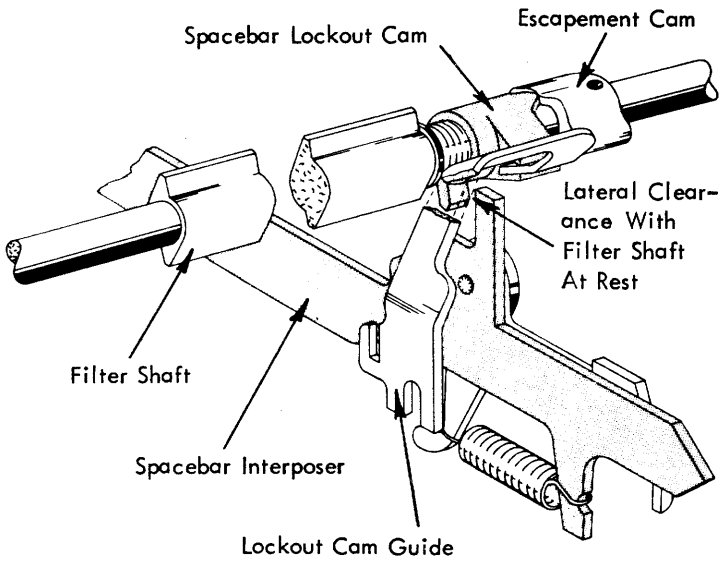
The late level spacebar mechanism is basically the same as the early level. The main difference is in the spacebar stem. A stabilizing link has been attached to the bottom of the stem by a shouldered rivet (Fig. 120). The rear portion of the link mounts on and pivots about an adjustable guide stud. This stud is fastened to a support on the front keylever bearing support by two screws. The stabilizing link controls the bottom of the spacebar stem to prevent the spacebar from tipping forward or back.

The spacebar restoring action is supplied by an extension spring that is attached to the spacebar operating arm and anchored to one of three holes in the carrier return/backspace repeat bail. In addition, the spacebar guide stud and the final stop as found on the early level mechanism has been eliminated from the late level spacebar mechanism.

Spacebar Lockout - Old Style

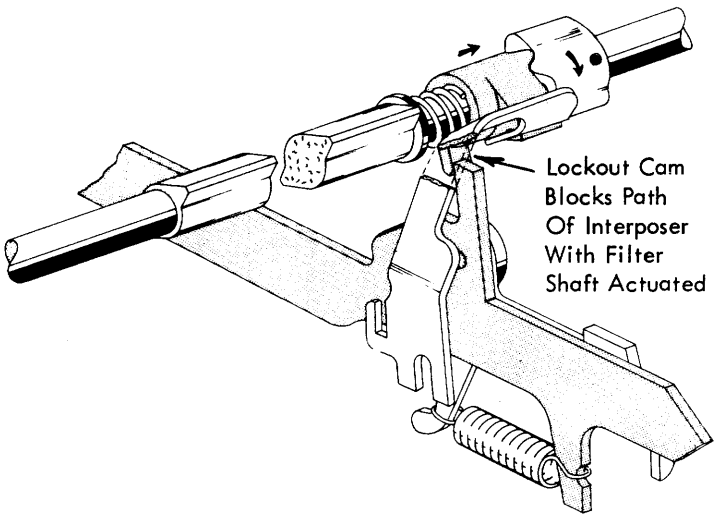
As previously mentioned, both the spacebar escapement and the print escapement operate by actuating the escapement trigger lever to cause an escapement operation. Because of the inter-relationship of these two mechanisms, operating both of them together causes only one space of escapement. Likewise if they are alternately operated too rapidly in succession, only one space of escapement will result. This happens occasionally when the operator hits the spacebar too soon after striking a letter keylever. As a result no space appears between the words. The spacebar fails to operate the escapement trip because the escapement trigger does not have sufficient time to reset above the lug on the torque bar before being actuated down again. Operator timing seldom causes the print escapement to fail after a spacebar operation.

In order to insure that the spacebar will always actuate the escapement following a print operation, it is necessary to place the spacebar mechanism into storage until the print escapement is completed. The spacebar can then operate to space the carrier. The spacebar storage feature greatly reduces the possibility of spacebar failure during a letter-space-letter operation. Failure results only if the spacebar operation is not stored. The spacebar will fail to store only if the spacebar and letter keylever are depressed simultaneously.



A. FILTER SHAFT AT REST

FIGURE 121a. Spacebar Lockout Mechanism



B. FILTER SHAFT ACTIVE

FIGURE 121b. Spacebar Lockout Mechanism

Spacebar storage is accomplished by blocking the rearward movement of the spacebar interposer. The interposer travel is blocked by the spacebar lockout cam on the filter shaft (Fig. 121). The lockout cam is spring-loaded toward the right against the escapement cam. In the rest position, the lockout cam is held to the left by the lateral camming surfaces on the two cams. In this position, the spacebar interposer is free to operate without interference (Fig. 121a).

During a character operation, rotation of the filter shaft allows the lockout cam to move toward the right because the high points of the lateral camming surfaces are no longer in contact (Fig. 121b). The lockout cam is prevented from rotating with the filter shaft by a guide bracket that fits in a slot in the front of the cam. As the lockout cam slides to the right an extension at the bottom of the cam moves into the path of a lug on the spacebar interposer (Fig. 121b). If the spacebar interposer is tripped, it will move to the rear slightly until it contacts the lockout cam extension. The interposer will remain tripped against the lockout cam until the print cycle has been completed. As the filter shaft approaches the rest position, the lockout cam is forced back to the left by the escapement cam. The spacebar interposer is then released to the rear for a spacebar operation.

Spacebar Lockout - New Style

The spacebar interlock mechanism is mounted within the operational control bracket just below the filter shaft. It consists of a bracket, an interlock interposer (which acts as a cam follower) and an interlock cam mounted on the filter shaft (Fig. 122).

When the filter shaft is in its rest position the upper extension of the interlock interposer is resting on the high point of one of the lobes on the interlock cam. This allows the horizontal

lug on the lower extension of the interlock interposer to just clear the adjustable stop on the spacebar interposer as it operates to the rear during a spacebar operation (Fig. 122).

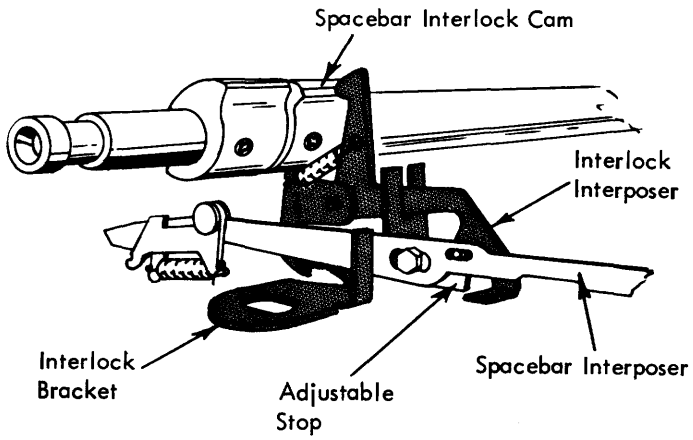


FIGURE 122. Spacebar Lockout - New Style

Whenever a print operation occurs, the filter shaft and interlock cam will begin to turn causing the rear portion of the interlock interposer to rise into the operating path of the spacebar interposer. If the spacebar interposer were then released the interlock interposer would interrupt and store the rearward travel of the spacebar interposer until the filter shaft completes its cycle. At this time the spacebar interposer would be released to finish its rearward travel and the spacebar operation would complete. Note that the interlock interposer is spring loaded into its active position and powered to its rest position. This is to prevent breakage should both the spacebar and print operation be operated simultaneously.

BACKSPACE

The backspace mechanism provides the operator with a method of positioning the carrier to the left one space at a time. It is used primarily in error correction and in centering headings, but it may also be used to position the carrier quickly to any point toward the left on the writing line.

The backspace operates by forcing the carrier to the left until the escapement pawl clicks from one tooth of the escapement rack to the next. The backspace pawl is mounted on the escapement bracket with the escapement pawl so that movement of the backspace pawl to the left also moves the carrier and escapement pawl to the left (Fig. 123). The tip of the backspace pawl engages the teeth of a sliding backspace rack located on the back of the power frame just below the escapement rack. Movement of the backspace rack to the left forces the backspace pawl to the left to cause the backspace operation.

The backspace rack is spring loaded toward the right by an extension spring between the rack and the operational latch bracket. Upon completion of the operation, the backspace rack restores to the right causing the backspace pawl to click from one tooth to the next in preparation for the next operation.

The escapement pawl has a small stud on its upper surface that fits into a slot in the backspace pawl (Fig. 124). The slot is elongated front to rear to allow the escapement pawl to click from one escapement rack tooth to the next as the backspace rack forces the carrier to the left (Fig. 124a).

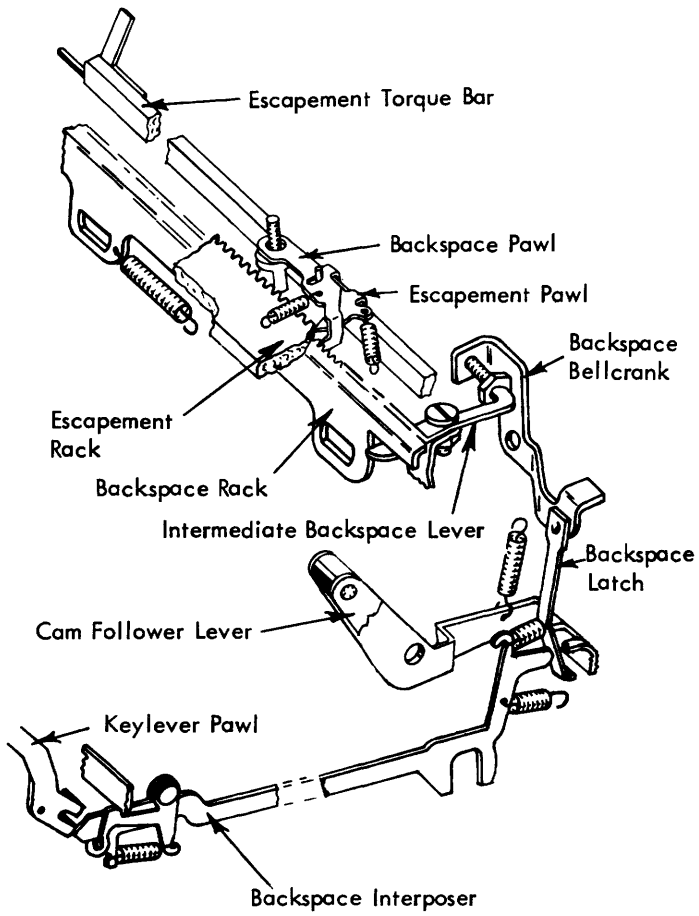


FIGURE 123. Backspace Mechanism

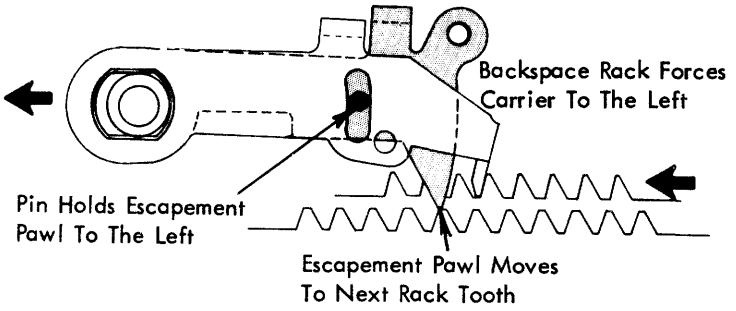
The slot also permits the backspace pawl to click into the next tooth as the backspace rack restores (Fig. 124b). Because of the elongated pivot holes in the two pawls, the stud connection is necessary to insure that the pawls move together during a backspace operation and to prevent the backspace pawl from moving to the right as the backspace rack restores. Without the stud connection, the backspace rack would require .044" additional travel to compensate for the .022" elongation in each pawl.

The backspace rack is actuated by a bellcrank pivoted on the front of the back plate (Fig. 123). The bellcrank operates the rack through an adjustable intermediate lever pivoted at the top of the operational latch bracket. The backspace operational latch is mounted to the horizontal arm of the backspace bellcrank by a ball shouldered rivet to permit free rotation of the latch. The latch extends down into position to hook under the bail of the operational cam follower. When the backspace interposer is released, the latch is pushed to the rear under the cam follower. Operation of the cam causes the cam follower to pull the latch down rotating the backspace bellcrank. A large headed adjusting screw on the vertical arm of the bellcrank operates the intermediate backspace lever to force the backspace rack to the left.

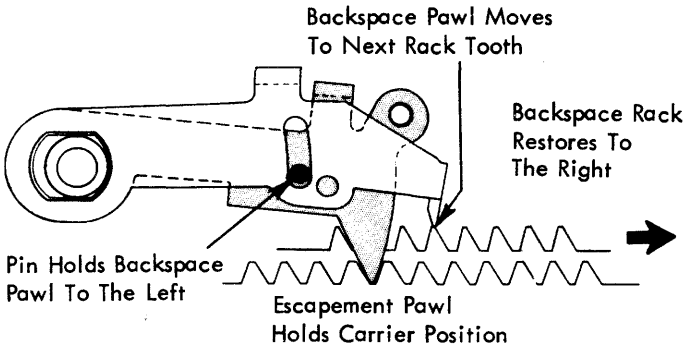
The intermediate backspace lever is adjustable forward or back to obtain the proper throw of the backspace rack. The intermediate lever is adjusted farther forward on the 12 pitch machines than on the 10 pitch. The 12 pitch intermediate lever is shorter than the 10 pitch to prevent interference with the power frame. It is too short to be used on 10 pitch machines. The 12 pitch intermediate lever is identified by a notch in the right side. The 10 pitch lever is unmarked.

The backspace interposer is released by depression of the backspace keybutton located at the upper right hand corner of the keyboard (Fig. 125). The backspace keylever is mounted in the keyboard assembly in the same manner as the letter keylevers. An extension spring between the keylever and the keylever guard restores the keylever to the rest position. A keylever pawl attached to the rear of the keylever extends down through a slotted guide stud in position just above the backspace interposer. Depression of the keylever causes the interposer to be released to the rear to begin the operation.

The upward travel of the keylever is limited by a fulcrum rod located at the top of the guide comb. A bail, located under the right front corner of the keyboard, determines the first limit for the keylever depression. The left end of the bail pivots in a small bracket attached to the bottom of the guide comb support. The right end pivots in a hole in the keyboard sideframe. A lug on the right side of the repeat bail limits against the keyboard sideframe to determine the rest position of the repeat bail. An extension spring inside the sideframe holds the repeat bail in the rest position and offers a resistance to further depression of the keylever.



A. BACKSPACE ACTUATING STROKE



B. BACKSPACE RESTORING STROKE

FIGURE 124. Backspace Operation

When the keylever is depressed, the front of the keylever is forced down against the repeat bail. This stops the travel of the keylever and allows a single operation of the mechanism. Additional pressure on the keylever causes the repeat bail to yield allowing the keylever to move into the repeat position. Holding the keylever in the fully depressed position causes the keylever pawl to hold the interposer down allowing a continuous operation of the mechanism. The final travel of the keylever is reached when the keylever bottoms in the guide comb.

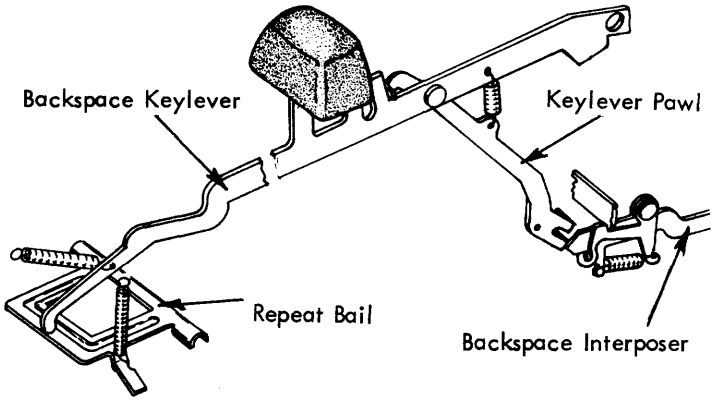


FIGURE 125. Backspace Keylever Mechanism

CARRIER RETURN (721)

The carrier return mechanism provides the operator with a powered return of the carrier to the left margin and an automatic line spacing of the paper. Depressing the carrier return keylever into the repeat position causes additional line space operations, if so desired. This can be done while the carrier is being returned making it unnecessary for the operator to wait until the carrier reaches the left margin.

The carrier return operates by winding the carrier return cord onto a drum at the rear of the machine (Fig. 126). The carrier return cord is hooked to the bottom of the carrier, passes around two pulleys at the left, extends back to the right over a guide roller, and attaches to the carrier return cord drum. The drum has spiral grooves for winding up the cord as on the escapement cord drum. The carrier return cord drum is attached by set screws to the escapement shaft just in front of the mainspring. Rotation of the escapement shaft causes the drum to wind up the cord and move the carrier to the left.

Movement to the left opposes the mainspring tension causing the mainspring to tighten. The power to rotate the escapement shaft in opposition to the manispring is taken directly from the operational cam shaft. The escapement cord drum (at the front of the escapement shaft) has a beveled gear molded on its front (Fig. 126). This beveled gear meshes with a small pinion gear on the operational shaft. The pinion gear pivots freely on the shaft between two C-clips. By means of a spring clutch the pinion is made to rotate with the operational shaft. The pinion gear drives the escapement cord drum in a clockwise direction causing the carrier return cord to be wound onto its drum. The carrier return pinion has a hub that forms a part of the spring clutch. A second hub just to the left of the pinion is in continuous rotation with the operational shaft. A clutch spring

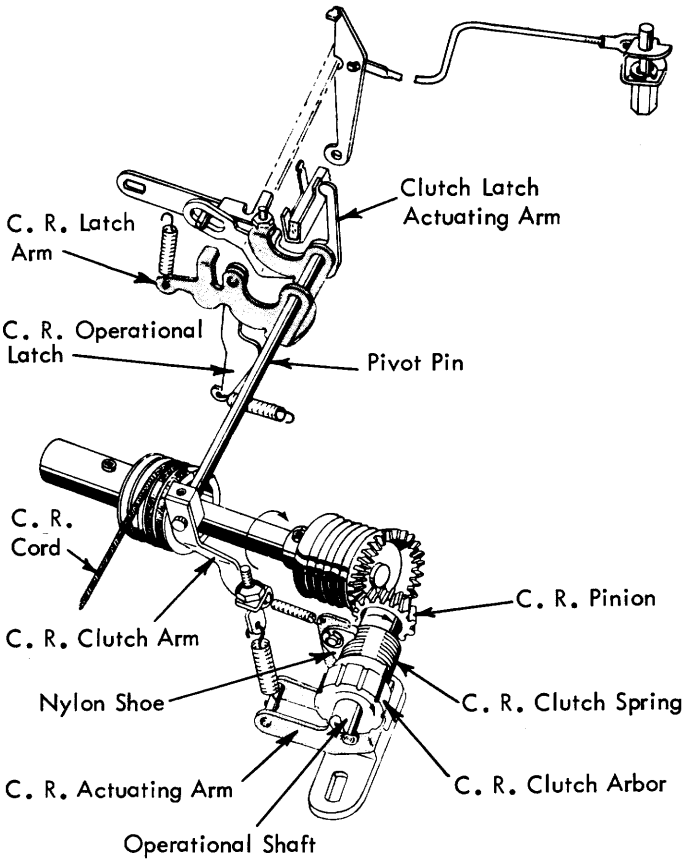


FIGURE 126. Carrier Return Mechanism

fits around the two hubs to complete the spring clutch. A steel band clamps the left end of the spring around its hub so that no slippage can occur at that point. The clamp causes the spring to rotate with the operational shaft. The shaft turns in the tightening direction of the spring; but no tightening occurs, because the pinion hub is smaller than the inside diameter of the clutch spring.

If the clutch spring is to tighten, friction must exist between the spring and the hub it is to drive. By pressing the loose end of the carrier return clutch spring against the pinion hub, friction is applied causing the spring to tighten around the hub and drive the pinion. The spring is pressed against the pinion hub by a nylon shoe just to the rear of the carrier return pinion (Fig. 126). The clutch spring decreases in diameter as it tightens around the pinion hub. The tension of the spring resists any change in size; therefore, when the pressure from the shoe is relaxed, the spring snaps back to normal size and ceases to drive the pinion.

In order to obtain a full carrier return, the carrier return shoe must press the spring against the pinion hub and hold it there until the carrier has reached the left margin. It must then release the clutch spring to end the carrier return operation.

The power to operate the shoe against the clutch spring is taken from the single lobed operational cam. Depression of the carrier return keylever sets the mechanism into operation. The keylever operates at the right side of the keyboard beside the backspace keylever. The keylever pivots around the fulcrum rod at the rear and operates in the keylever guide comb at the front. The limits of the keylever travel are the same as for the backspace lever.

When the keylever is depressed, the keylever pawl attached at the rear of the keylever releases the carrier return interposer to the rear (Fig. 111). The interposer causes the cam to be engaged and pushes the carrier return operational latch under the cam follower into position to be pulled down. When the cam operates, the cam follower pulls down on the latch. It also pulls down on the index pawl carrier link, attached at the rear of the follower, causing a line space operation.

The carrier return operational latch is mounted on the carrier return latch arm that pivots around a shaft on the operational latch bracket at the rear of the powerframe (Fig. 127). The shaft is called the pivot pin and also acts as a pivot point for the escapement trigger lever and the spacebar latch lever.

Attached solidly to the right end of the pivot pin is a bellcrank called the clutch latch actuating arm (Fig. 127). As the cam follower moves the operational latch down, an adjusting screw at the right side of the carrier return latch arm rotates the clutch latch actuating arm and pivot pin. This action causes three things to occur.

1. The top of the clutch latch actuating arm forces a lug of the escapement torque bar to the rear rotating it to remove the escapement and backspace pawls from their racks (Fig. 127). This prevents the pawls from dragging along their racks as the carrier is returned.

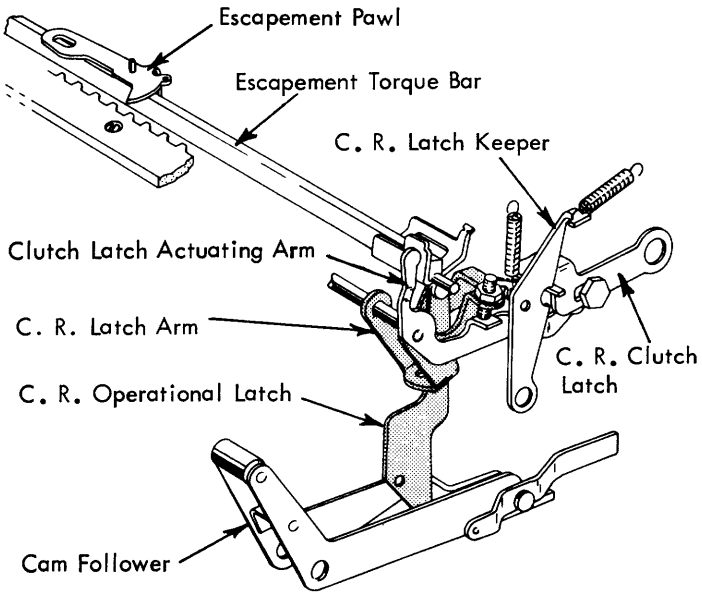


FIGURE 127. Carrier Return Latch Operation (721)

2. The carrier return clutch spring tightens around the pinion hub and drives the carrier return operation. An arm called the carrier return clutch arm, at the left end of the pivot pin pulls up on a heavy extension spring (Fig. 128). The lower end of the spring is connected to a bellcrank-like part called the carrier return actuating arm. The upward pull on the spring rotates the nylon shoe (at the top of the arm) against the clutch spring forcing the spring to tighten and drive the pinion gear. After the shoe is pressing against the clutch spring, the heavy carrier return actuating spring is extended slightly to maintain a constant pressure against the clutch spring.

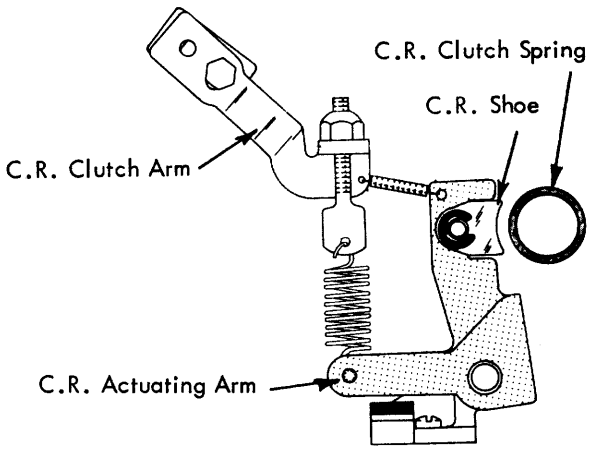


FIGURE 128. Carrier Return Clutch Actuating Mechanism

3. The clutch latch actuating arm is latched in the operated position to maintain pawl release and to continue the pressure of the shoe against the clutch spring (Fig. 127). The rear of the clutch latch actuating arm contains an elongated hole. An eccentric adjusting screw connects the arm to the carrier return clutch latch which pivots at the rear on the powerframe. As the actuating arm moves down the clutch latch is also lowered. Spring loaded against the forward edge of the clutch latch is a hook-like part called the carrier return latch keeper. When the clutch latch has been pulled down into its active position, the keeper hooks over the latch to hold it down (Fig. 127).

The carrier return mechanism remains latched in the active position until the carrier reaches the left margin. At that time the clutch is unlatched and the escapement pawl is restored to the escapement rack ready for a typing operation.

The margin rack is mounted between the side frames just in front of the carrier. The margin rack has a small amount of lateral movement. When the carrier is away from the left margin, a spring located at the left end of the rack loads the margin rack to the right. As the carrier moves to the left during a return operation, the carrier strikes the left margin stop forcing the margin rack to the left. The extreme right end of the margin rack contains a roll pin. Movement of the rack to the left causes the pin to operate the carrier return unlatching bellcrank that pivots on a stud outside the right side frame (Fig. 129). A link connects the bellcrank to the carrier return latch keeper at the rear. As the bellcrank operates, the unlatching link pulls the keeper forward releasing the clutch latch. The latch is restored to the rest position by its spring and the action of the escapement torque bar spring. A small spring connected near the carrier return shoe holds the shoe away from the clutch spring in the rest position (Fig. 128).

The clutch latch does not hold the cam follower in the active position during a return operation; therefore the cam and follower immediately restore to the rest position ready for the next operation. Depression of the keylever with the carrier in motion releases the interposer to cause another carrier return operation. Because the carrier is already in motion, this reduces to nothing more than another line space operation.

Depression of the keylever into the repeat position causes the repeat bail to yield and allows the rear lug of the keylever pawl to release the index interposer to the rear. This causes an index operation only, without operating the carrier return clutch mechanism. A repeat operation can be achieved either with the carrier at the left margin or as the carrier is moving toward the left.

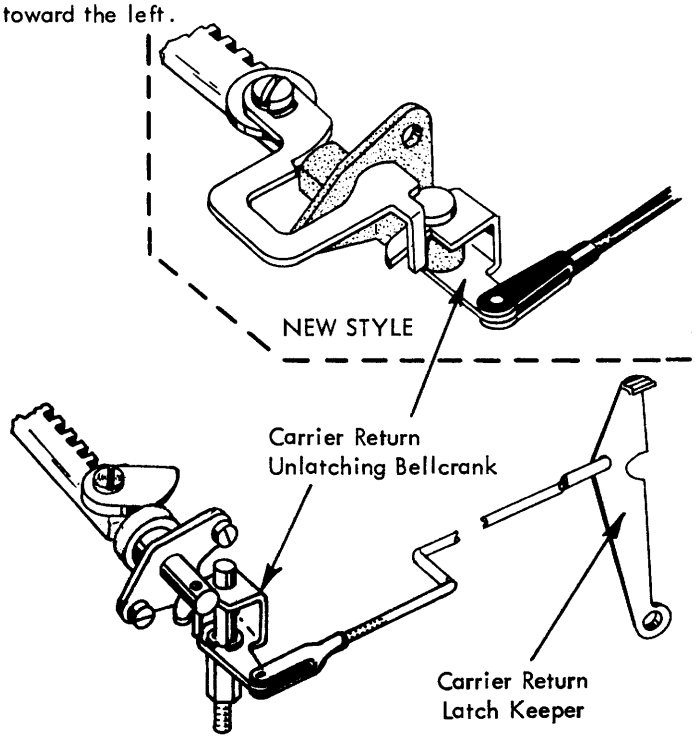


FIGURE 129. Carrier Return Clutch Unlatching Mechanism

If the carrier is already resting at the left margin when a carrier return operation begins, the clutch is prevented from latching. The cam is not prevented from operating, however, so a carrier return operation must occur. The platen is indexed and the carrier return spring clutch attempts to wind the carrier return cord onto the drum. The carrier cannot be pulled farther to the left, because it is already against the left margin. The pull continues to be exerted on the cord until the cam follower passes the high point of the cam at which time it restores and allows the shoe to move away from the clutch spring.

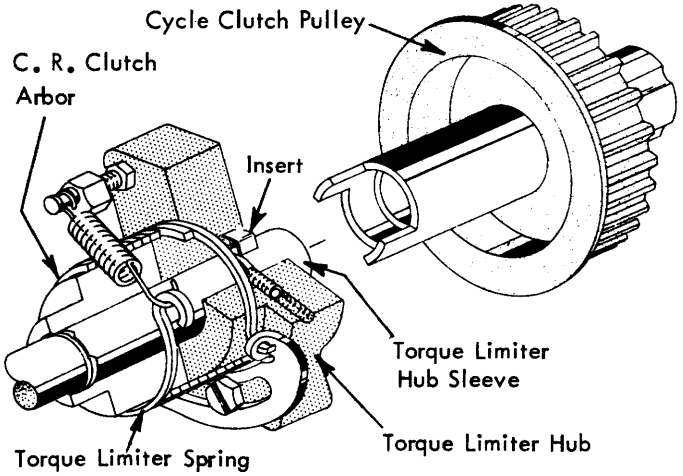


FIGURE 130. Torque Limiter - Cut Away View

During the time the cord is being pulled without being able to move the carrier, the carrier return clutch mechanism must be allowed to slip in order to reduce the strain and prevent breakage to the parts. The carrier return clutch arbor is indirectly driven by the operational shaft. The large shoulder on the arbor fits into a heavy clutch spring at the left called the torque limiter spring (Figs. 130 and 131). The left end of the spring is clamped to the torque limiter hub and the torque limiter hub is set screwed to the operational shaft. The carrier return clutch arbor is then driven by means of the torque limiter spring.

The operational shaft turns in the unwinding direction of the torque limiter spring. This tends to expand it allowing it to slip. The spring is heavy and considerably smaller than the carrier return clutch arbor over which it fits. The friction present between the arbor and the spring tends to drive the arbor even though it is in the unwinding direction of the spring. However, insufficient driving force is obtained from this arrangement.

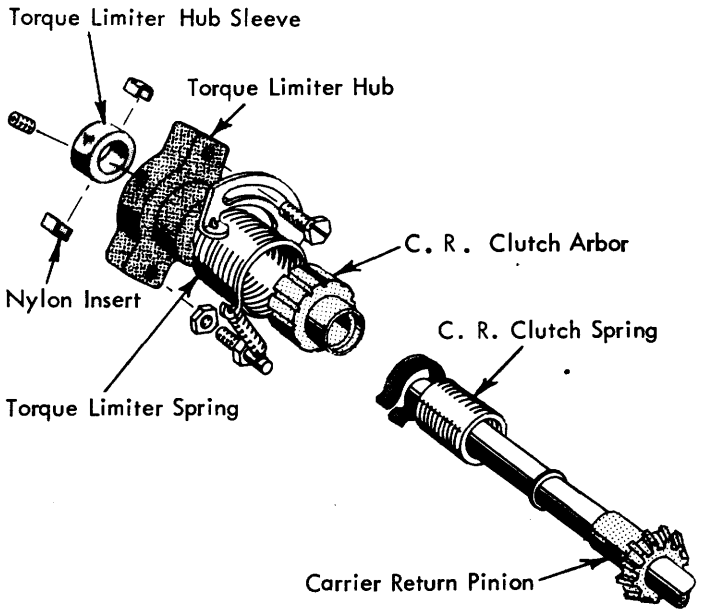


FIGURE 131. Torque Limiter - Exploded View

The right end of the torque limiter spring is formed into an eye to accept an extension spring connected from the eye to an eccentric adjusting stud on the torque limiter hub (Figs. 130 and 131). The extension spring increases the force required to unwind the torque limiter spring so that no slippage occurs during normal carrier return. The torque limiter spring slips when the carrier cannot move to the left. It also slips at the beginning of a carrier return operation to allow smooth acceleration and prevent a jerky start.

Carrier Return (723 and 725)

The following section concerns the operational section of the carrier return mechanism on the long carriage machines. This section of the mechanism is different than that of the 721 because of the longer carriage.

The carrier return operational latch is mounted on the carrier return latch arm which pivots freely about a pivot pin mounted in the right hand operational latch bracket at the rear of the powerframe. The carrier return latch arm straddles the carrier return lever (Fig. 132) which is tightly fastened to the same pivot pin by a bristo screw. The bristo screw tightens against a flat spot on the pivot pin. An adjusting screw threaded through the top of the latch arm contacts the top of the horizontal lug on the carrier return lever (Fig. 132). A spring between these two pieces loads them together. When the carrier return latch is pulled down by the cam follower, the latch arm forces down on the carrier return lever causing the pivot pin to rotate.

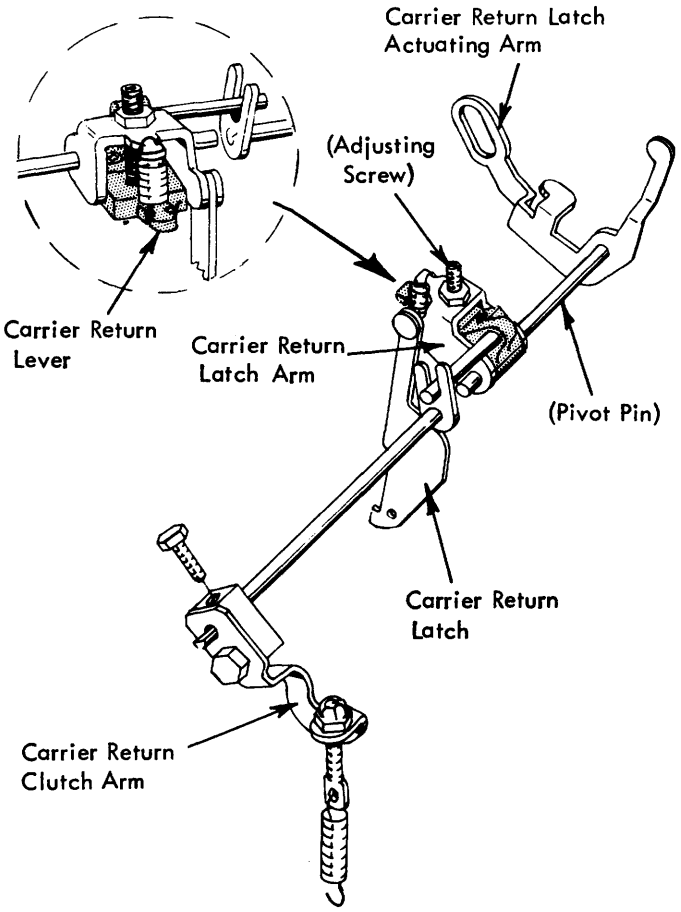


FIGURE 132. Carrier Return (723 & 725)

Riveted to the right hand end of the pivot pin and rotating with it is an arm called the carrier return latch actuating arm. This arm performs the same function on the long carriage machines as it does on the 721. It rotates the escapement torque bar (removing the escapement and backspace pawls from their racks) plus operates the carrier return clutch latch down to its latched position.

The carrier return lever has a pin that projects to the left into a forked arm located on the right hand end of the carrier return clutch arm pivot pin (Fig. 132). This provides a solid driving connection between the two pivot pins. Whenever the carrier return latch is pulled down, the carrier return clutch arm pivot pin rotates causing the clutch arm to produce a pull on the heavy spring attached to the carrier return actuating arm. From this point on the operation is the same on long carriage machines as on the 721.

INDEXING AND PLATEN VARIABLE

Indexing

The indexing mechanism operates to line space the paper vertically. An indexing operation can be obtained by depressing either the carrier return keylever or the special indexing keylever. Depressing the carrier return keylever also causes the carrier to move to the left margin; whereas depressing the index keylever causes a line space operation only.

The index selector lever, located to the rear of the right end of the platen, may be positioned so that the mechanism will space either one or two spaces during each operation. With the lever forward, single line spacing will occur. Double spacing will take place if the lever is to the rear.

Indexing is achieved by a pawl that engages and rotates a ratchet at the right end of the platen. The ratchet is locked to the platen so that the platen is also rotated. Two styles of indexing mechanisms have been produced.

Early Style Indexing

The index pawl pivots on a bellcrank called the index pawl carrier (Fig. 133). The carrier pivots just behind the platen on the right hand carriage plate. A small extension spring at the top of the index pawl spring-loads the pawl toward the platen ratchet. The index pawl is held clear of the ratchet in the rest position by a lug on the upper index stop located just to the rear of the pawl (Fig. 133). As the pawl carrier is operated down by the link attached at the rear, the index pawl moves forward away from the index stop. The spring is then allowed to rotate the pawl into engagement with the platen ratchet. Further movement of the carrier causes the index pawl to rotate the platen ratchet and platen.

The index detent lever located just below the platen has a roller attached at the front that seats between two teeth of the platen ratchet. The lever is spring-loaded to hold the roller into the bottom of the teeth. The roller insures that the platen will not creep out of position and that indexing will be accurate. As the ratchet is rotated, the detent roller moves from one tooth to the next. The detent lever spring forces the roller to the bottom of the teeth to insure equal spacing on each operation.

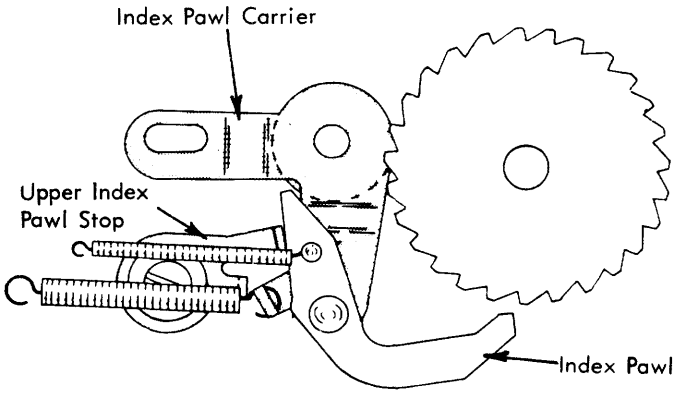


FIGURE 133. Index Pawl Rest Position

The index is operated by a link that extends down from the index pawl carrier to the rear of the operational cam follower (Fig. 134). When the single lobed cam rotates, the cam follower pulls down on the link to effect a line spacing operation.

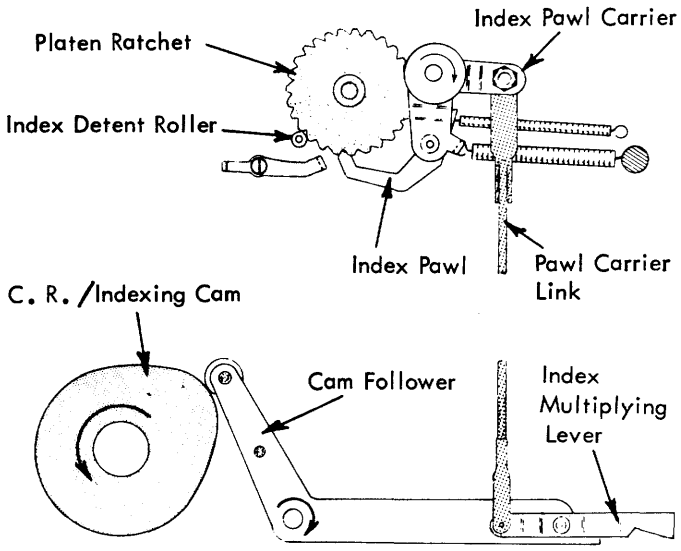


FIGURE 134. Single Space Indexing Operation

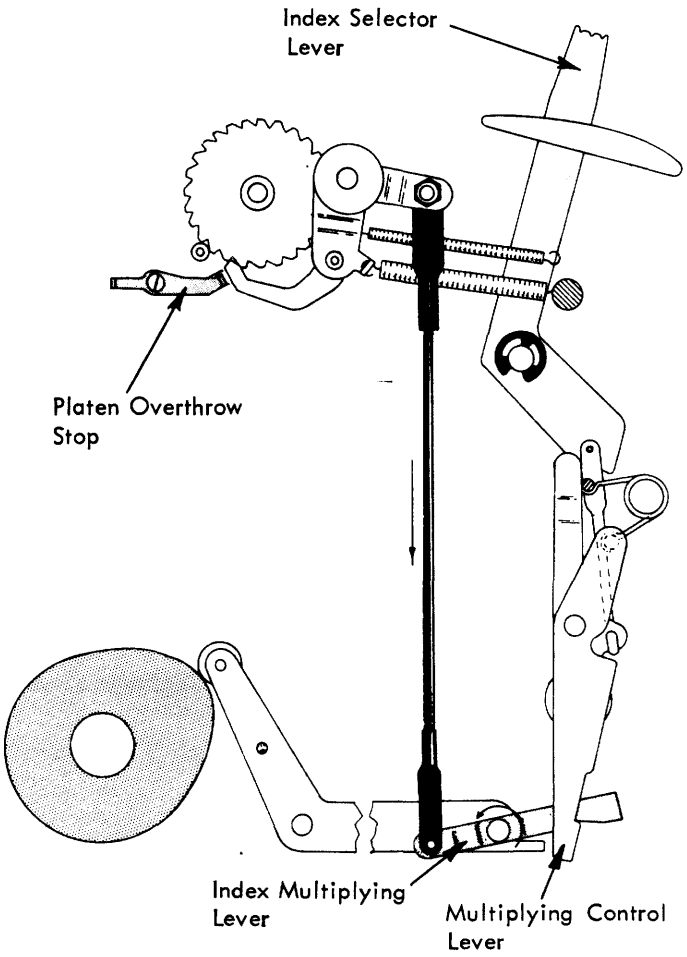


FIGURE 135. Double Space Indexing Operation

Whether a single or a double space operation occurs is determined by how far the index pawl carrier link is pulled. Because the cam follower always moves the same distance on each operation, a system of multiplying the leverage must be utilized. The index pawl carrier link is attached to the front of the index multiplying lever that pivots at the rear of the cam follower (Fig. 134). A lug at the rear of the cam follower prevents the multiplying lever from pivoting up in front. During a single line spacing operation, the multiplying lever moves down the same amount as the cam follower (Fig. 134).

When the index selector lever is pushed to the rear for a double space operation, it pivots the multiply control lever forward under the end of the multiplying lever blocking its downward movement (Fig. 135). When the cam follower operates, the multiplying lever pivots where it attaches to the cam follower. This causes the front of the multiplying lever to pull down on the index pawl carrier link sufficiently to cause two line spaces of movement to the platen (Fig. 135).

A double and single space operation require the same time lapse, because both are operated by the same rotation of the cam. The platen rotation must be accelerated for double spacing, because of the added distance it is to be rotated. This extra speed of rotation tends to cause the platen to space too far because of the momentum developed. To prevent overthrow, an index pawl stop is provided at the end of the double spacing stroke (Fig. 136). Upon contact with the stop, the index pawl is stopped and locked into mesh with the platen ratchet to block further rotation of the platen.

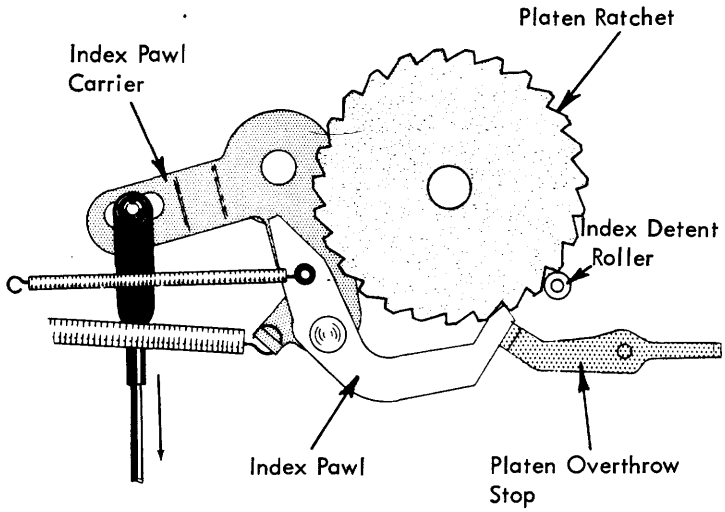


FIGURE 136. Platen Overthrow Stop

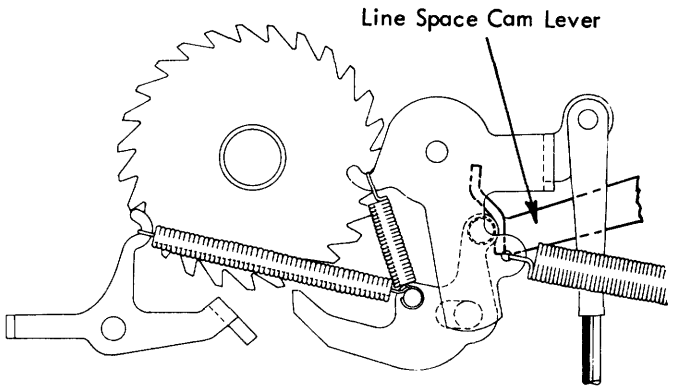
New Style Indexing

The new indexing mechanism is designed to improve the accuracy and reliability of the indexing operation. It differs from the earlier design mainly in the index pawl and the method of selecting single or double spacing.

The index mechanism is operated by the cam follower by means of the index pawl carrier link connected to the front of the index multiplying lever. The rear of the multiplying lever is always in contact with multiplying lever stop attached solidly to the powerframe (Fig. 137). As the cam follower operates, the multiplying lever pivots on the cam follower and pulls down on the index pawl carrier link (Fig. 138).

The index pawl carrier link always receives the same amount of motion each time it operates regardless of whether the mechanism is set for single or double spacing. The amount of travel is sufficient to cause double space rotation of the platen.

During a double space operation, the index pawl is allowed to enter the platen ratchet immediately. The index pawl then forces the ratchet tooth forward two spaces until the pawl contacts the platen overthrow stop. The overthrow stop wedges the pawl into the ratchet teeth to lock the platen in position (Fig. 138).



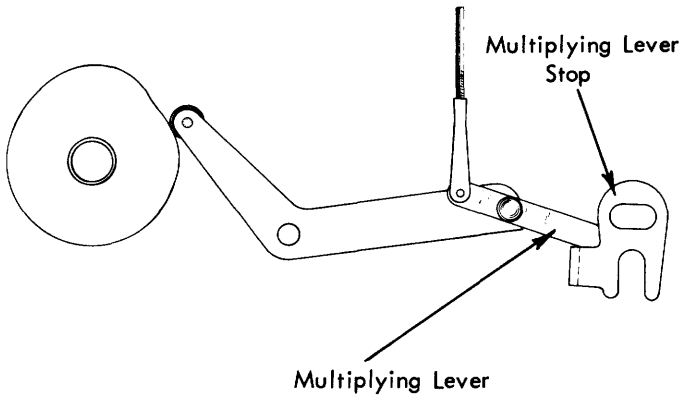
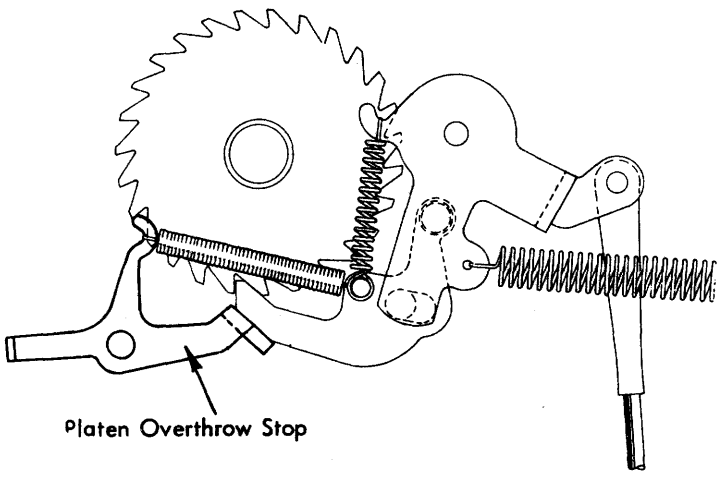


FIGURE 137. Index Mechanism - Rest Position

If only a single space operation is desired, the index pawl must be prevented from entering the ratchet until it has passed one tooth of the ratchet. The remaining travel after the index pawl enters the ratchet is only sufficient to cause one tooth of rotation to the platen. The index pawl contacts the platen overthrow stop at the end of the stroke as on a double space operation.

The index pawl entry into the platen ratchet is controlled by the line space cam lever attached to the index selection lever (Fig. 139). The cam lever has two steps at the forward end in position to contact a stud on the side of the index pawl. A small spring between the index pawl and the pawl carrier spring-loads the pawl toward the platen. With the index selection lever to the rear in the double space position, the index pawl stud contacts the lower step of the cam lever allowing



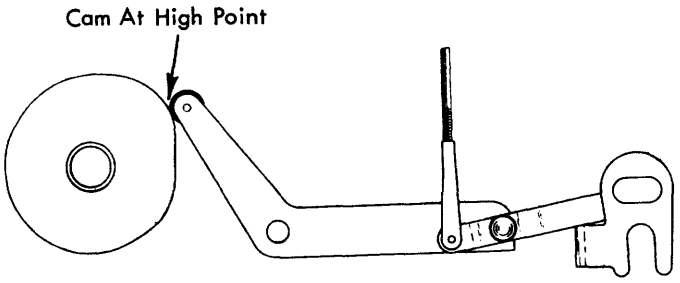
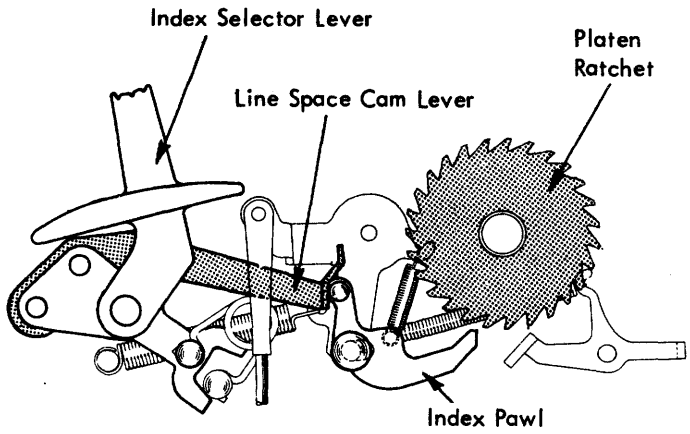


FIGURE 138. Index Mechanism - Active Position

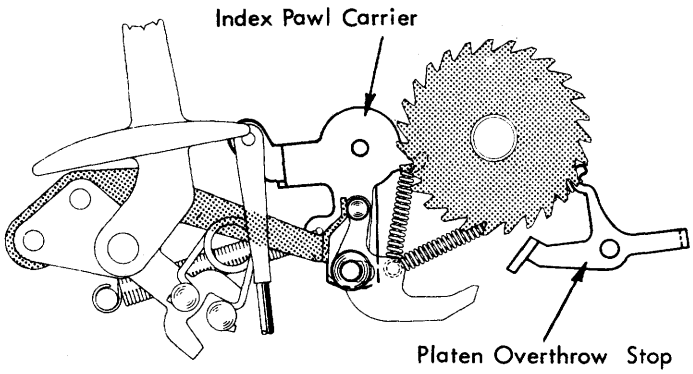
the pawl to rest near the platen ratchet (Fig. 139a). In the single space position, the index pawl stud contacts the upper step of the cam lever causing the index pawl to rest farther from the platen (Fig. 139b). Thus in the single space position, the index pawl stud maintains contact with the line space cam lever longer and delays the entry of the pawl into the platen ratchet.

The index selection lever is held in the single or double space position by a toggle hairpin spring (Fig. 139). Movement of the index selection lever is restricted by two extensions at the bottom of the lever that contact the hairpin spring mounting stud.



A. DOUBLE SPACE POSITION

FIGURE 139a. Index Selection Mechanism



B. SINGLE SPACE POSITION

FIGURE 139b. Index Selection Mechanism

The index pawl is designed with an elongated pivot hole so that it "floats" forward during a portion of the index stroke. The pawl is spring-loaded forward in the rest position by an extension spring between the pawl and a hook on the platen overthrow stop (Fig. 139). A heavier spring between the index pawl carrier and the base tie rod holds the mechanism in the rest position.

As the index mechanism operates, the pawl engages the ratchet tooth. There is then a slight delay until the pawl carrier reaches the end of the elongated slot in the index pawl. The pawl carrier is operated so sharply that it actually "kicks" the platen. The platen is thus caused to move ahead of the index stroke. Without the elongated hole in the index pawl, the platen ratchet would reach the final position ahead of the index pawl. With the floating index pawl, the pawl is spring-loaded forward against the ratchet tooth. As the ratchet moves ahead of the index stroke, the pawl is able to move with it and reach the overthrow stop at the same time the platen reaches the final position. The pawl is then able to wedge into the ratchet and block any further rotation due to the momentum of the platen.

Index Keylever Mechanism

An indexing operation occurs any time the cam operates the cam follower. The cam may be caused to operate by releasing either the index or carrier return interposer to the rear. The carrier return operation has been discussed in another section; therefore the index alone will be dealt with here. The index keylever pivots on the keylever fulcrum rod at the rear and extends toward the front only to the first row of keybuttons. An offset in the keylever places the end of the keylever and keybutton outside the right side frame (Fig. 140). A stud in the side frame fits through an elongated hole in the keylever to limit the overall travel of the keylever. An extension spring from a lug on the keylever to the stud restores the keylever to the rest position.

A small spring loaded arm called the index repeat lever operates in a slot in the side frame under the keylever and acts as a first limit for the keylever depression (Fig. 140). When the keylever is depressed to the first limit, the keylever pawl at the rear of the keylever depresses the index interposer to release it to the rear. This releases the cam allowing the operation to occur. If additional pressure is applied to the key-

button, the index repeat lever will yield allowing further depression of the keylever. The rear lug of the keylever pawl will then hold the interposer down so that it cannot relatch forward on the guide bracket. The interposer will continue to operate forward and back creating a repeat cam operation and a repeat indexing of the platen.

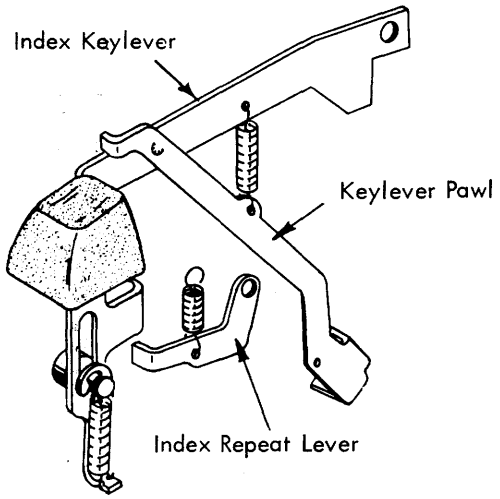


FIGURE 140. Index Keylever Mechanism

Platen Variable

The platen variable mechanism provides the operator with a means of rotating the platen to a position other than the normal writing line. The variable is used for typing above or below the writing line, locating the writing line after reinserting the paper, and for typing on lines of other than six lines per inch spacing.

The platen ratchet must remain stationary when selecting a new writing line so that the detent roller will be seated between two teeth of the ratchet at the new position. A clutch mechanism connects the ratchet to the platen so that it can be engaged for line spacing and disengaged for the variable operation. The clutch can be disengaged by pushing the left hand platen knob toward the right. As long as the knob is held to the right, the platen can be rotated freely while the ratchet remains stationary. When the knob is released, the clutch is automatically re-engaged by spring tension.

The left side of the platen ratchet contains two heavy lugs that form a channel (Fig. 141). The platen variable driver operates left to right in the slot and always turns with the ratchet. A compression spring between the ratchet and the driver loads the driver to the left so that serrations on the outer surface of the driver mesh with matching serrations inside the platen end plug (Fig. 141). The meshing of the serrations causes the platen, the driver, and the ratchet to be locked together and turn as a unit.

When the driver is disengaged from the platen end plug, the platen can be turned to the desired position. The driver can then engage different serrations and lock the platen in the new position. The left hand platen knob is mounted to a shaft that slides left or right inside the platen. A light compression spring holds the shaft toward the right to prevent free play. The shaft has two pins attached to the right end that operate through holes in the platen end plug (Fig. 141). The pins on the push rod extend through the end plug to the right and rest against the platen variable driver. Movement of the platen knob toward the right is transferred to the driver to disengage it from the platen end plug.

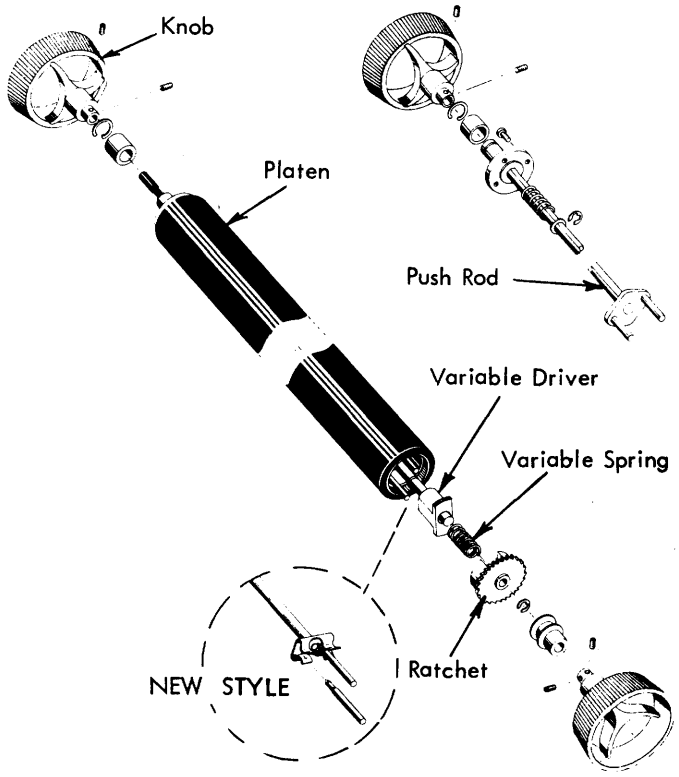


FIGURE 141. Platen Variable Mechanism

TABULATOR

The tabulator mechanism permits the operator to position the carrier quickly to a predetermined point on the writing line by depressing the tab keybutton one time. The tabulator is used in typing columns of figures, indenting paragraphs, or any other operation that requires positioning the carrier to a specific point each time.

In order for a tabulator operation to take place, several basic things must occur. The stopping point must be predetermined. The escapement and backspace pawls must be released to allow carrier movement. The pawls must be latched in the released position to continue the movement. The speed of the carrier must be controlled. And the pawls must be allowed to restore to their racks at the proper time.

Tab Set and Clear - Old Style

A rack of tab stops located just to the rear of the escapement rack, allows the operator to select the positions where the carrier will stop when the tab keylever is depressed. The tab stops operate friction tight in grooves of the tab rack -- one corresponding to each escapement position. The tab set and clear button located at the left of the keyboard may be rocked forward or back to set or clear a tab stop (Fig. 142). When the

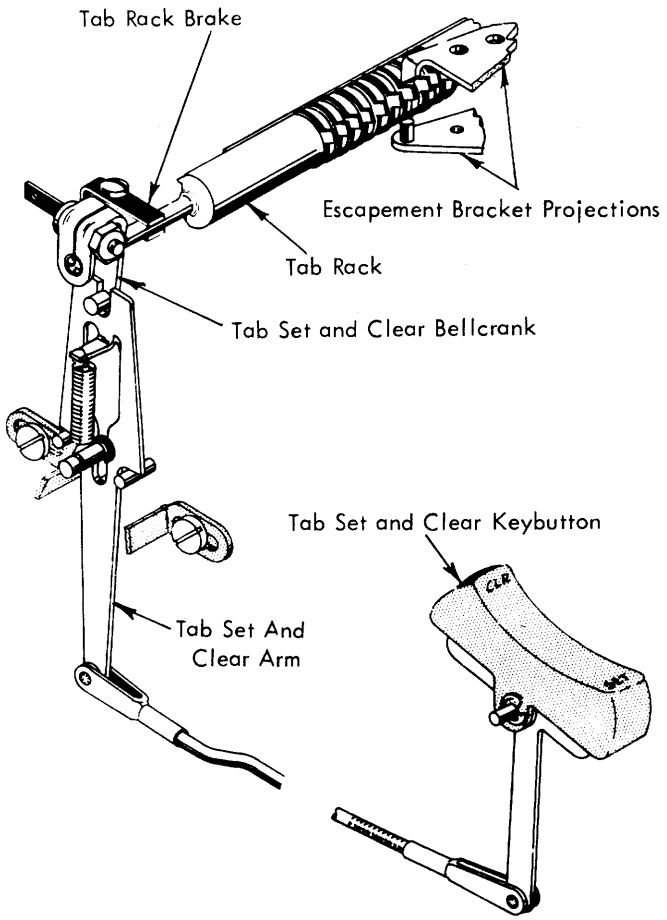


FIGURE 142. Tab Set And Clear Mechanism

front of the button is depressed, the set and clear arm at the rear operates to rotate the front of the tab rack up. A tab stop strikes a projection on the escapement bracket blocking the movement of the tab stop (Fig. 143a). As the tab rack continues to rotate, it is forced to rotate inside the tab stop. When the set and clear button is released, the tab rack is restored to the rest position leaving the set tab stop lower than the others (Fig. 143b). A reverse operation causes the tab stop to strike a stud at the bottom of the escapement bracket blocking its movement while the tab rack is rotated further (Fig. 143c). When the set and clear lever is released, the tab rack is restored to the rest position leaving the tab stop in the cleared position (Fig. 143d). Because it is the escapement bracket that sets or clears the tab stop, the carrier must be positioned to the desired tab stop before the set or clear action can occur. The stop at the extreme right end of the tab rack is the tab final stop and remains in the set position at all times to disengage the tab mechanism at the right hand limit of travel.

Tab Set and Clear - "Gang Clear"

Both the tab rack and the tab clear mechanism has been redesigned on late level machines so that the tab stops may be "gang cleared". Gang clearing tab stops is the term given to the procedure for clearing all of the set tab stops in one operation. This is accomplished by positioning the carrier at the extreme right hand margin, depressing the tab clear button and (while holding the tab clear button depressed) actuating the carrier return mechanism. As the carrier travels toward the left hand margin it will clear every set tab stop across the entire length of the tab rack.

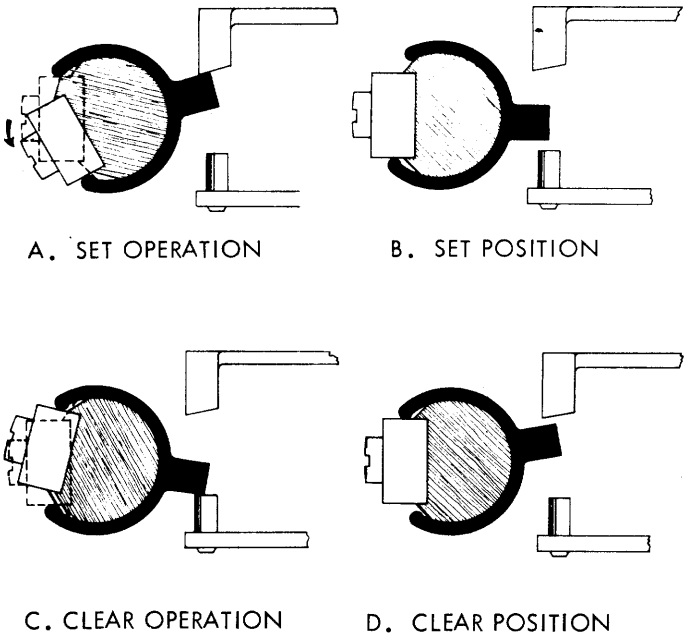


FIGURE 143. Tab Set And Clear Operation

Figure 144 illustrates a cross section of the tab rack. Notice that the tab stops encircle and operate freely about a round shaft that runs through the center of the tab rack. The tab rack is a slotted tube that mounts on shouldered bushings which are set screwed to each end of the shaft. The slots in the tube are guide slots for each individual tab stop (Fig. 145). Sections of spring fingers mounted across the entire length of the tab rack operate against small detent lugs (projections) on each tab stop. The function of the spring fingers is to detent each tab stop in either its set or cleared position.

The tab set operation on the gang clear mechanism remains the same as on the old style. The tab stop strikes a projection of the escapement bracket blocking the movement of the tab stop as the tab rack rotates top to the rear (Fig. 144a). The clear operation is slightly different. A gang clear finger mounted

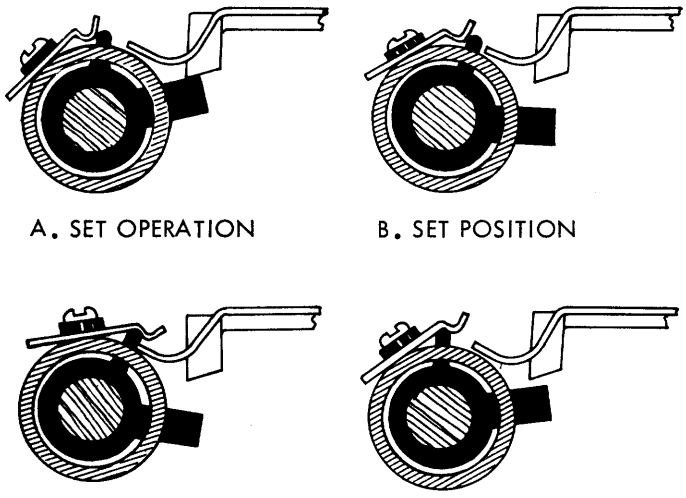


FIGURE 144. Tab Set and Clear Operation (gang clear)

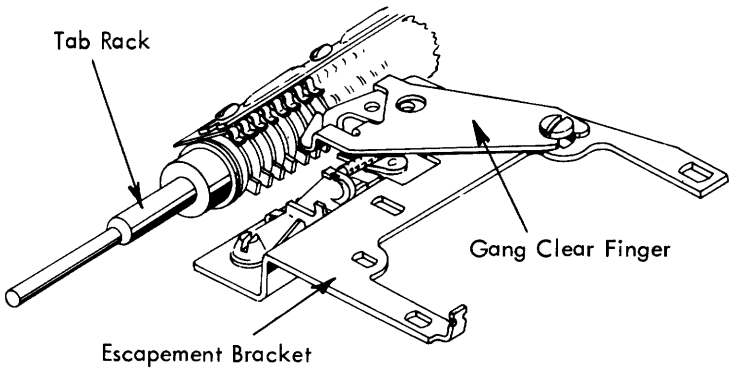


FIGURE 145. Gang Clear Finger

to the top of the escapement bracket (Fig. 145) projects to the rear just below the detent lugs on the tab stops (Fig. 144b). When the rack rotates top to the front during a clear operation, the gang clear finger which is in the path of the detent lug restricts the tab stop from rotating with the rack thereby causing the tab stop to rotate up to its cleared position (Fig. 144d).

During a gang clear operation the rack is held rotated in its cleared position as the gang clear finger slides along the rack (with the carrier) camming each set tab stop back to its cleared position. The angle on the left side of the tip of the gang clear finger provides the means for this camming action.

The tab rack is restored to rest from either the set or clear operation by an extension spring on the set and clear arm (Fig. 142). The spring pulls the arm down against two pins on the powerframe so that it maintains a vertical position. The tab rack is restored rather quickly when released and has a tendency to flip past the rest position. This could partially clear a stop that had just been set or partially set a stop that had just been cleared. To prevent the rack from restoring past the rest position, a leaf spring at the left end of the rack applies a slight braking action (Fig. 142).

Pawl Release

The main purpose of the tab lever is to remove the backspace and escapement pawl from their racks during a tab operation. The tab lever mounts at the rear of the escapement bracket on the same mounting stud as the backspace and escapement pawls do.

The tab lever is very easily operated to the rear by a manual process. The tab keylever operates at the left side of the keyboard the same as the letter keylevers. A lower extension makes the keylever operate as a bellcrank (Fig. 146). When the keylever is depressed, a link connected to the extension operates the tab bellcrank located on the powerframe at the rear. Through a vertical connecting link, the bellcrank rotates the tab torque bar. The tab torque bar is mounted the same as the escapement torque bar. It pivots at each end and operates just above the tab lever. The pivot point is near the top of the torque bar so that depressing the keylever causes the bottom of the bar to swing to the rear. The tab torque bar contacts a lug of the tab lever trigger located just above the tab lever. A lower lug of the trigger forces the tab lever to the rear as the keylever is depressed further (Fig. 147). As the tab lever pivots toward the rear, a lug at the front of the tab lever contacts the escapement and backspace pawls and forces them to

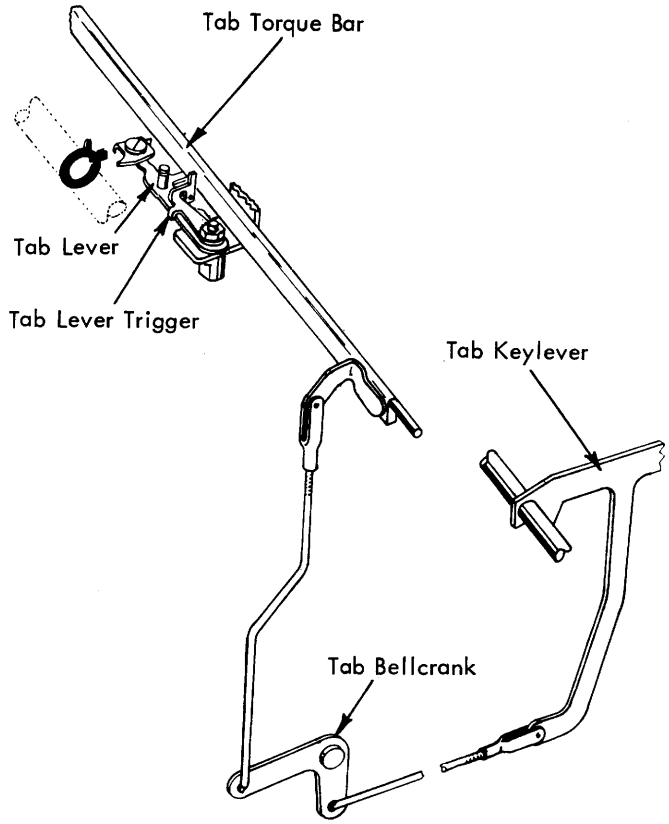


FIGURE 146. Tab Keylever Mechanism

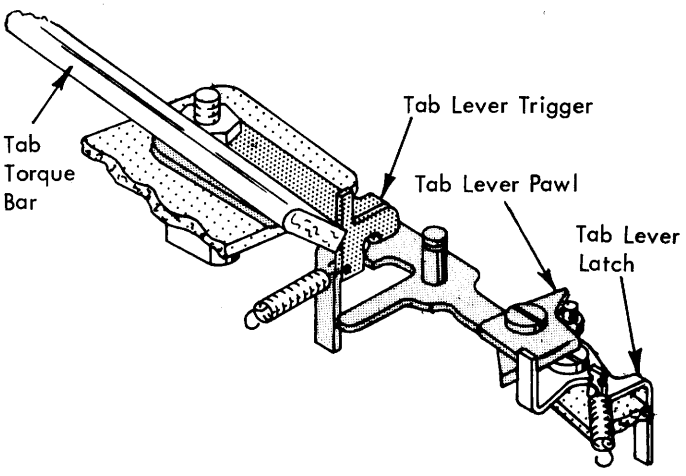


FIGURE 147. Tab Latch Operation

the rear out of mesh with their racks (Fig. 148). A small latch pivots on the escapement bracket at the right end of the tab lever. When the tab lever has moved far enough to the rear to release the pawls, the tab latch swings into a notch in the tab lever assembly to hold it to the rear thus latching the pawls out of their racks (Fig. 147). A tab lever overthrow stop is mounted on the escapement bracket. It extends to the rear and down behind the trigger (Fig. 149). The stop prevents the tab lever from being thrown into the tab rack if the keylever is struck hard.

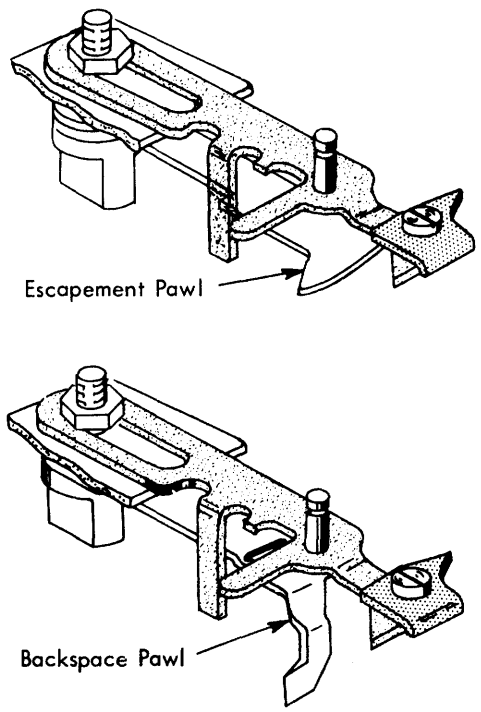


FIGURE 148. Pawl Release Operation

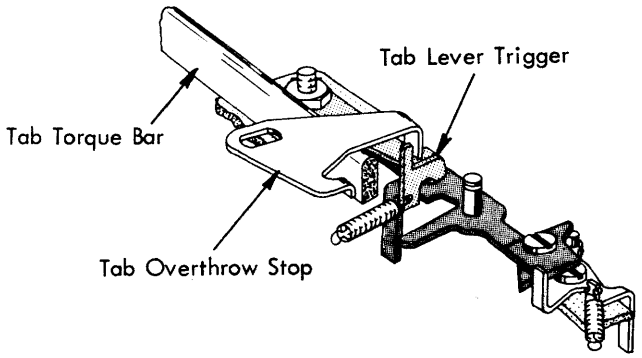


FIGURE 149. Tab Overthrow Stop

Tab Governor

The carrier speed during a tab operation must be controlled to insure an accurate tab, reduce the noise, and prevent excessive wear and shock on the components. During a tab operation, the carrier is pulled to the right by the tension of the mainspring as during an escapement operation. The tab governor operates by limiting the speed with which the escapement cord drum winds up the cord.

The beveled gear on the escapement cord drum meshes with the tab governor pinion located on the operational shaft to the right of the cord drum (Fig. 150). The pinion gear operates between two collars. The left collar and the pinion gear have hubs enclosed by a clutch spring. The left collar is set-screwed to the shaft and the pinion gear pivots freely on the shaft. The spring is wound so that it slips when the pinion is held stationary and the operational shaft is turning.

If the pinion gear is turned in the same direction as the operational shaft but at a faster rate of speed, the friction of the clutch spring causes it to tighten around the two hubs locking them together. During a tab operation, the cord drum drives the pinion gear in the same direction as the operational shaft. The mainspring tension causes the pinion to speed up and tighten the clutch spring. The mainspring then tries to accelerate the operational shaft. The mainspring does not have sufficient tension to drive the operational shaft, because of the drag present in the system. The shaft must be driven by the motor; therefore the speed of the tab governor pinion can be no faster than the normal speed of the operational shaft. The escapement cord drum can wind up the cord only as fast as the pinion will let it. The gear ratio between the pinion gear and the escapement cord drum allows the carrier to be moved at the proper speed during a tab operation. No governing action is obtained during approximately the first inch of carrier travel, because a short distance is required to tighten the tab governor clutch spring.

The tab governor pinion is the same size as the carrier return pinion gear. This makes the speed of the carrier the same for both tab and carrier return.

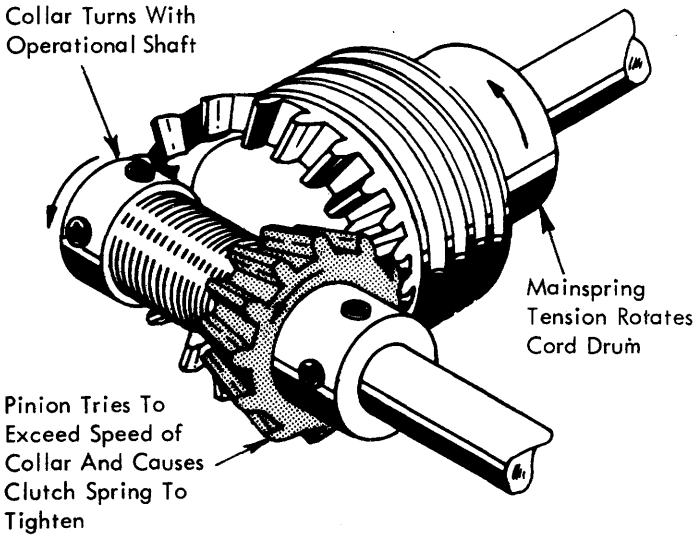


FIGURE 150. Tab Governor Mechanism

Tab Unlatching

When the carrier reaches the desired stopping point, the escapement pawl must be allowed to re-enter the escapement rack and stop the movement of the carrier. The tab lever is mounted in an elongated hole at its pivot point. An extension spring holds the tab lever to the right. As the carrier moves toward the right, the tip of the tab lever contacts the set tab stop and is prevented from moving further (Fig. 151a). The carrier continues to the right carrying the pawls and the tab latch with it. Movement is allowed by the elongated hole at the tab lever pivot. As the escapement pawl moves to the right in relation to the tab lever, a notch in the pawl allows it to drop off the lug of the tab lever and restore to the escapement rack (Fig. 151b). Further movement of the carrier moves the tab latch to the right out of the notch of the tab lever (Fig. 151c). The tab lever then restores and allows the backspace pawl to re-enter its rack.

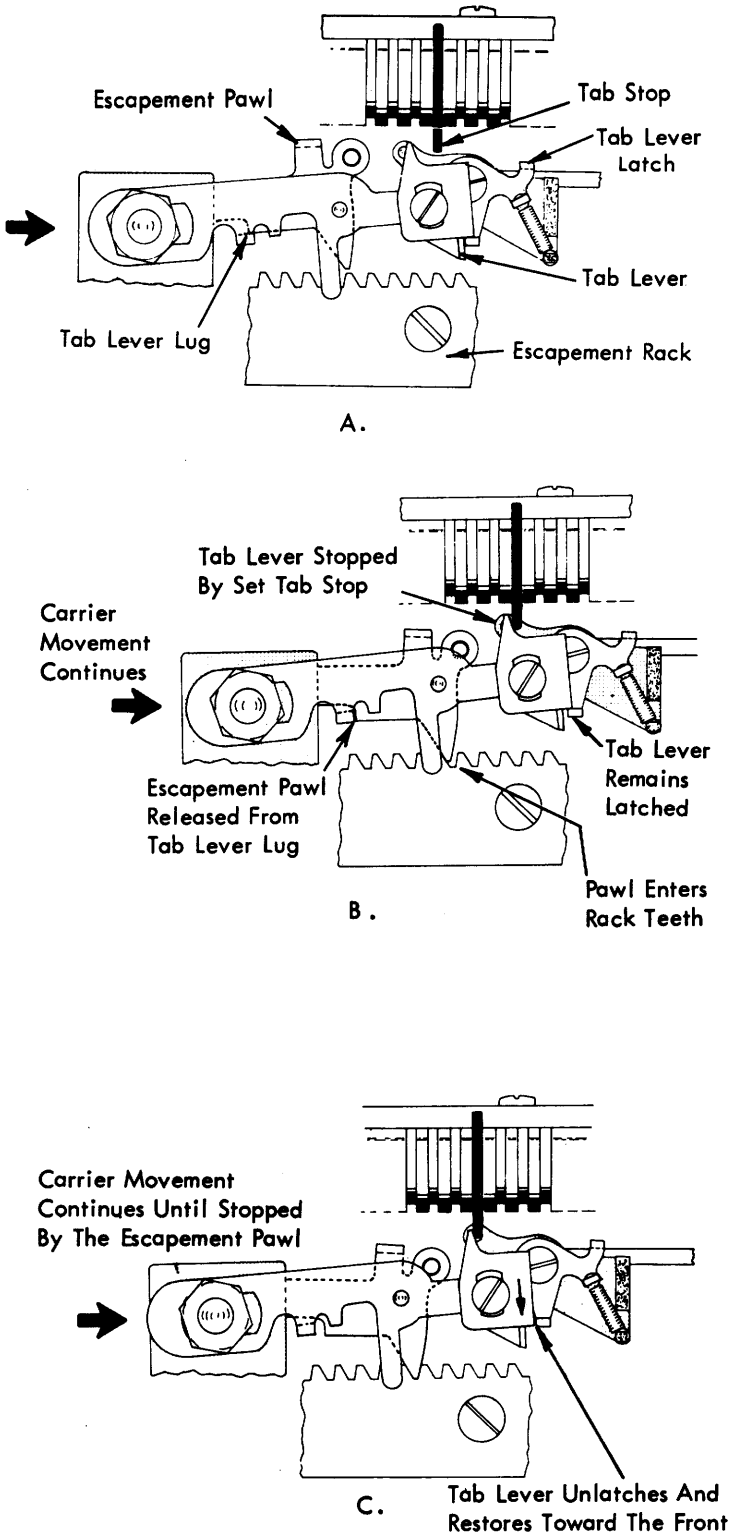


FIGURE 151. Tab Unlatching Operation

It should be noted that the escapement pawl is allowed to enter the rack before the backspace pawl. The escapement pawl must be allowed to enter early to insure that it will enter the correct tooth of the escapement rack. If the backspace pawl were allowed to enter at the same time, the adjustment of the backspace rack could allow the backspace pawl to enter its rack stopping the carrier slightly to the left of the desired point. Delaying the entry of the backspace pawl prevents this possibility.

During a rapid tab/typing operation, it is possible for the typist to delay releasing the tab lever until after a few characters have been typed. If this happened, the carrier could reach the right side of the elongated hole in the tab lever and be stopped by the tab lever against the set tab stop. To prevent blocking the carrier in this manner, the tab lever and tab lever trigger are designed to allow the tab lever to restore, even though the keylever is held depressed. The trigger moves to the right with the carrier during the unlatching travel of the carrier. At about the same time the tab lever is released by the tab latch, the tab lever trigger moves in front of a notch in the tab lever (Fig. 152). The tab lever is then allowed to move forward into the rest position. The tab lever is restored by the action of the springs on the tab lever and the backspace pawl.

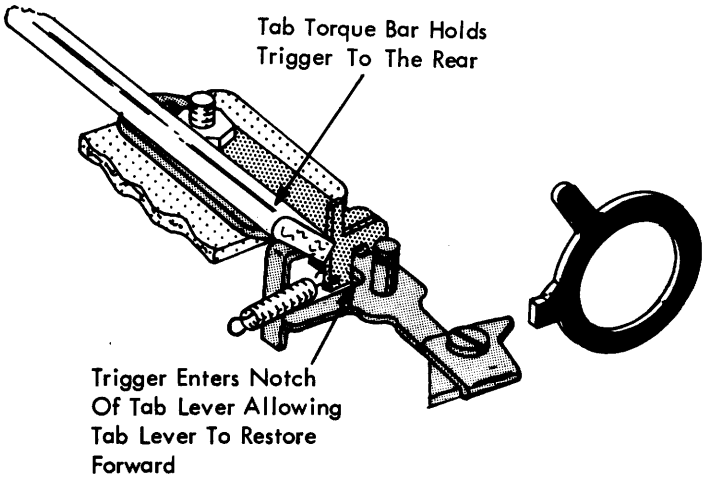


FIGURE 152. Tab Lever Trigger Operation

When the tab lever trigger restores, the tab lever is allowed to reset for the next operation. As the trigger moves out of the notch in the tab lever, the tab lever is snapped to the right by its spring into position to be operated by the trigger. At the same time, the tab lever lug resets to the right in front of the escapement pawl ready for pawl release on the next operation. A forward extension of the tab lever rests against the escapement bracket. A lug at the rear of the tab lever trigger rests against the tab lever to prevent the trigger from resting against the tab torque bar. Improper rest position of the tab lever can cause backspace problems if the backspace pawl is not allowed to mesh deeply enough into its rack. The tab lever will also fail to reset to the right in front of the escapement pawl, if the tab lever rests too far to the rear. The tab mechanism would then be inoperative, because no pawl release could be obtained.

Tab Interlock

The tab lever is prevented from latching to the rear during a carriage return operation. If the tab lever were allowed to latch, the tab lever pawl attached to the end of the tab lever would strike the right side of a set tab stop locking the carrier. The tab lever is prevented from latching by restricting the tab latch from rotating into its latching position. A lug at the rear of the tab latch extends down behind the escapement torque bar (Fig. 153). Whenever the escapement torque bar is operated, as during a carrier return, the tab latch is rotated counterclockwise away from the tab lever. Thus, the tab lever cannot latch.

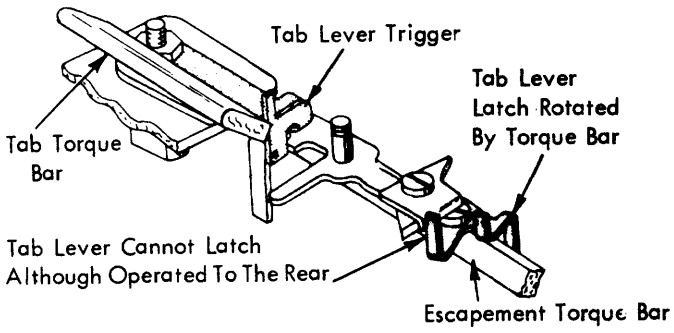


FIGURE 153. Carrier Return Tab Interlock

Carrier Return/Tab Interlock

The carrier return/tab interlock allows a tab operation to supersede or unlatch a carrier return operation. An operator can use this interlock feature to obtain a partial carrier return that will be followed by a tabulation operation to a desired set tab stop. This gives the operator a helpful short-cut when typing a column of figures or listing at the right side of the paper. The operation is achieved by depressing the tab keylever immediately after the carrier passes the desired set tab stop as the carrier is returning toward the left hand margin during a return operation.

This interlocking action is produced by a bellcrank, called the carrier return/tab interlock, which mounts to the side frame by a shouldered screw. The upper arm of the interlock extends behind a clip on the tab torque bar while the lower arm extends behind the carrier return latch keeper (Fig. 154). When the tab torque bar is rotated it produces a push on the upper arm of the interlock causing the lower arm to pull the keeper forward to unlatch the carrier return.

(The additional arm mounted on the same shouldered screw is for interlocking the carrier return whenever the top cover is raised. This is a "safety" interlock.)

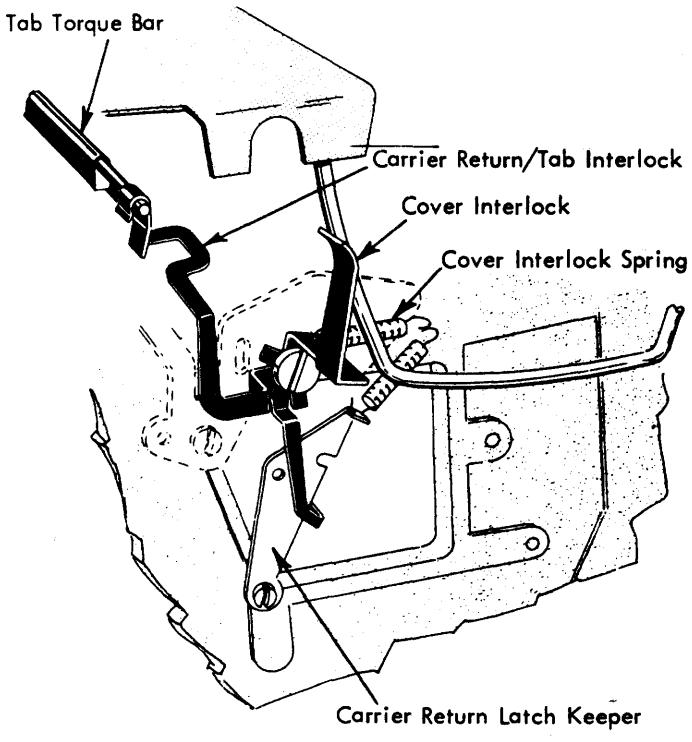


FIGURE 154. Carrier Return / Tab Interlock

MARGIN CONTROL

The term "margin" denotes the distance between the edge of the paper and the typewritten material. The left and right margins are determined by the position of the margin stops on the margin rack. The carrier travel is restricted by contacting the margin stops.

Margin Stops

The margin stops are mounted on the margin rack. The rack is positioned horizontally in the machine in front of the carrier. Each margin stop has a slider and pin assembly that meshes with teeth at the rear of the margin rack (Fig. 155). The number of teeth per inch in the margin rack corresponds to the pitch of the machine.

Each margin stop has a margin set lever attached to the slider and pin assembly. The margin set levers extend through a slot in the front case so as to be accessible to the operator. Either margin stop may be repositioned by pushing the margin set lever to the rear to disengage the pin from the rack and then by sliding the margin stop along the rack to the desired location. A scribe line on the margin set lever acts as a pointer to indicate the position of the margin stop in relation to the scale on the front of the case. A pointer on the front of the carrier indicates the position of the carrier.

The left hand margin stop controls the left margin on the paper. An extension of the stop is struck by the margin stop latch pivoted on a bracket attached to the carrier (Fig. 155). This action forces the margin rack to the left to unlatch the carrier return leaving the carrier resting at the left margin position.

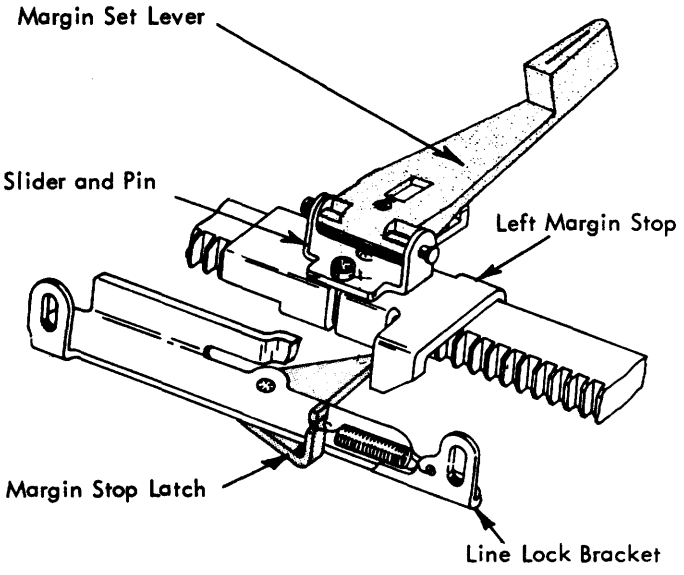


FIGURE 155. Left Margin Mechanism

Line Lock - (Old Style)

The purpose of the line lock mechanism is to lock the keyboard when the carrier has reached the right hand margin on the paper. With the keyboard locked, the operator cannot inadvertently continue typing at that position.

The line lock mechanism is operated by the right hand margin stop. In addition to operating the line lock, the margin stop also rings a bell several spaces before the line lock action occurs. The bell warns the typist that the carrier is near the margin stop. In most cases this allows sufficient space to complete a word or phrase before the line lock occurs. The typist can then return the carrier to begin the next line.

If the typist continues typing after the bell rings, the keyboard is locked after a few spaces to prevent typing on the margin. The keyboard is locked by forcing a special bellcrank into the selector compensator (Fig. 156). The keyboard lock bellcrank forces the steel balls of the compensator to shift in the tube and block the depression of any other interposer.

The keyboard lock bellcrank is actuated indirectly by an arm welded to the bellringer bail near the left end. A camming surface on the front of the line lock bracket forces the bail to pivot farther forward after ringing the bell (Fig. 156). The arm welded to the bail moves downward forcing down on the keyboard lock interposer. The keyboard lock interposer and the keyboard lock bellcrank pivot on the character interposer fulcrum rod. The keyboard lock interposer and bellcrank are connected at the rear by an extension spring. When the interposer is forced down, the extension spring pulls the bellcrank down into the selector compensator.

Because of the machine's character storage feature and the speed with which the interposers can be depressed, it is not always possible for the keyboard lock bellcrank to enter the selector compensator. If a character is in storage when the keyboard lock interposer is pivoted down, the keyboard lock bellcrank stops against the top of the steel balls (Fig. 157). The light spring connecting the interposer to the bellcrank is extended slightly.

To be sure that the keyboard is locked before another character can enter storage, the keyboard lock bellcrank is driven into position by the keyboard lock interposer. The interposer contains an elongated pivot hole the same as a character interposer. A stud in the interposer is in position just above a spring-loaded arm on the keyboard lock bellcrank (Fig. 157). The arm is relieved to permit downward movement of the interposer. When the keyboard lock interposer is down, it is positioned in the path of the filter shaft (Fig. 157). As the filter shaft operates the stored character interposer, the keyboard lock interposer is also driven forward (Fig. 158). The stud on the keyboard lock interposer cams the arm of the keyboard lock bellcrank down. The heavy extension spring between the arm and the bellcrank is extended and creates a strong downward pull on the bellcrank (Fig. 158).

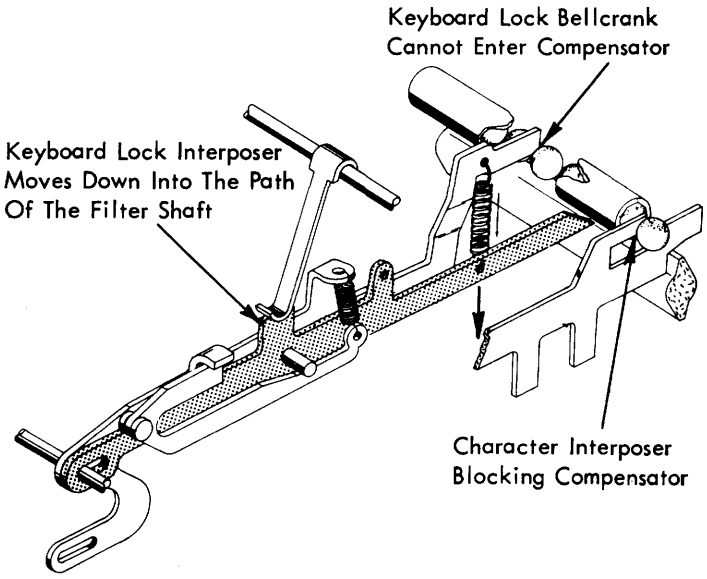


FIGURE 157. Line Lock Blocked By Stored Character Interposer

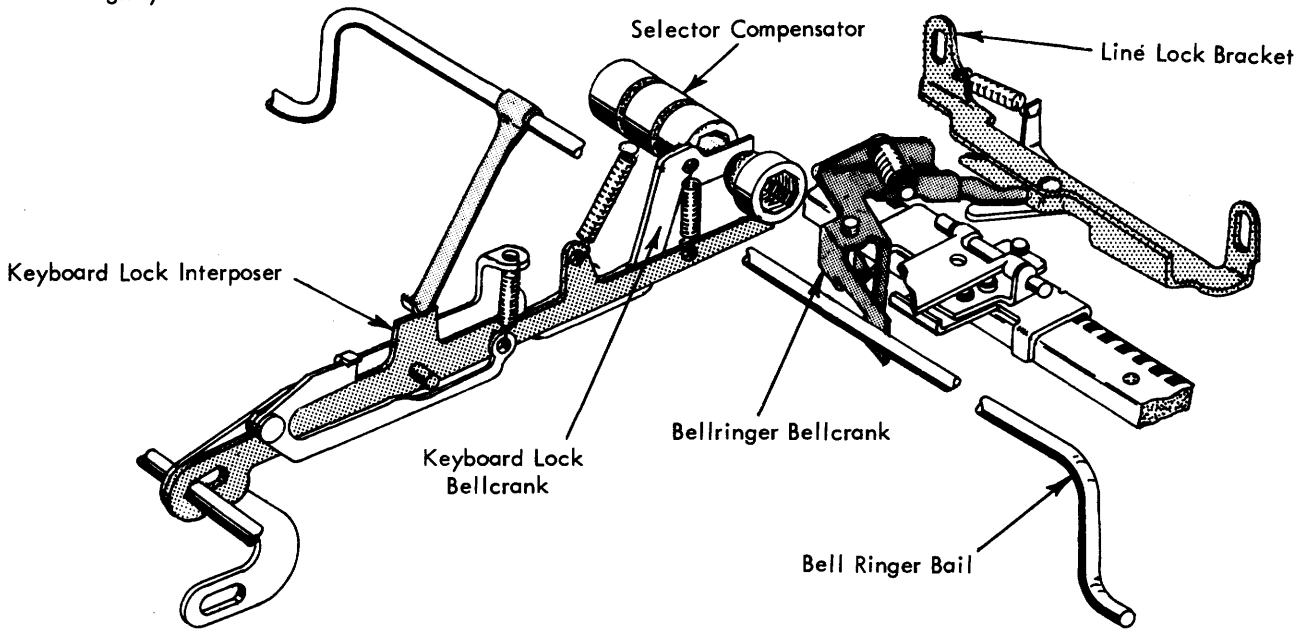


FIGURE 156. Line Lock Mechanism - Old Style

The keyboard lock bellcrank is then snapped into the selector compensator at the earliest possible instant. The keyboard is locked with the carrier at the margin stop position or one space later depending upon whether or not a character has been stored at the beginning of the line lock operation.

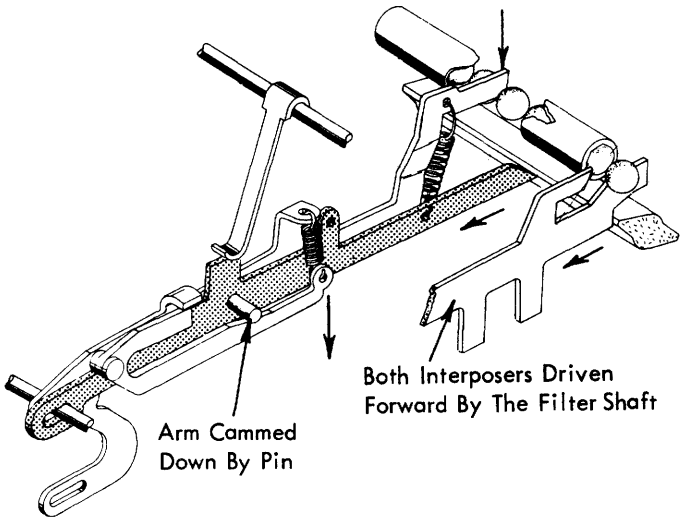


FIGURE 158. Keyboara Lock Interposer Operation

No type piling will occur as a result of a linelock failure, because the margin stop does not block the movement of the carrier. No operational keys are affected by the line lock. The carrier may be moved through the line lock zone by use of the spacebar, backspace, carrier return, or tabulator. The line lock is usually released by operating the backspace or carrier return or by depressing the margin release keylever. Moving the carrier to the left or right allows all the line lock parts to restore and unlock the keyboard. Depressing the margin release keylever rotates the margin rack to move the bellringer bellcrank out of the path of the line lock bracket. This allows the line lock parts to restore to rest unlocking the interposers.

The keyboard lock interposer is restored up and to the rear by an extension spring the same as a character interposer. The stud on the interposer extends to the left under the keyboard lock bellcrank in position to restore the bellcrank into the rest position.

Line Lock - (New Style)

On the new style line lock mechanism the keyboard lock interposer and its operation have been eliminated. It has been found that a sufficient amount of spring tension can be placed on the keyboard lock bellcrank, by the operation of the bellringer bail, to snap the bellcrank into the compensator tube at the earliest possible instant. The keyboard may still lock one space late after the carrier reaches the right hand margin because of the character storage feature.

The new line lock consists of two pieces, the keyboard lock lever and the keyboard lock bellcrank (Fig. 159). Similar to the old style, the bellringer bail operates the keyboard lock lever down (as the carrier approaches the right hand margin) causing the keyboard lock bellcrank to be loaded into the selector compensator tube by the tension of the spring between the lever and the bellcrank. The mounting of the bellcrank and lever remains the same as the old style.

Note in Figure 159 that the bellringer bellcrank has been redesigned and is now one piece as compared to the old style in Figure 156. The new style helps to reduce lost motion between the line lock bracket and the bellringer bail. Because of this new style bellringer bellcrank, the entire margin rack must tip each time the carrier is returned through the right hand margin. The rear extension of the bellcrank must ride up and over the camming surface on the line lock bracket causing the margin rack to rotate too.

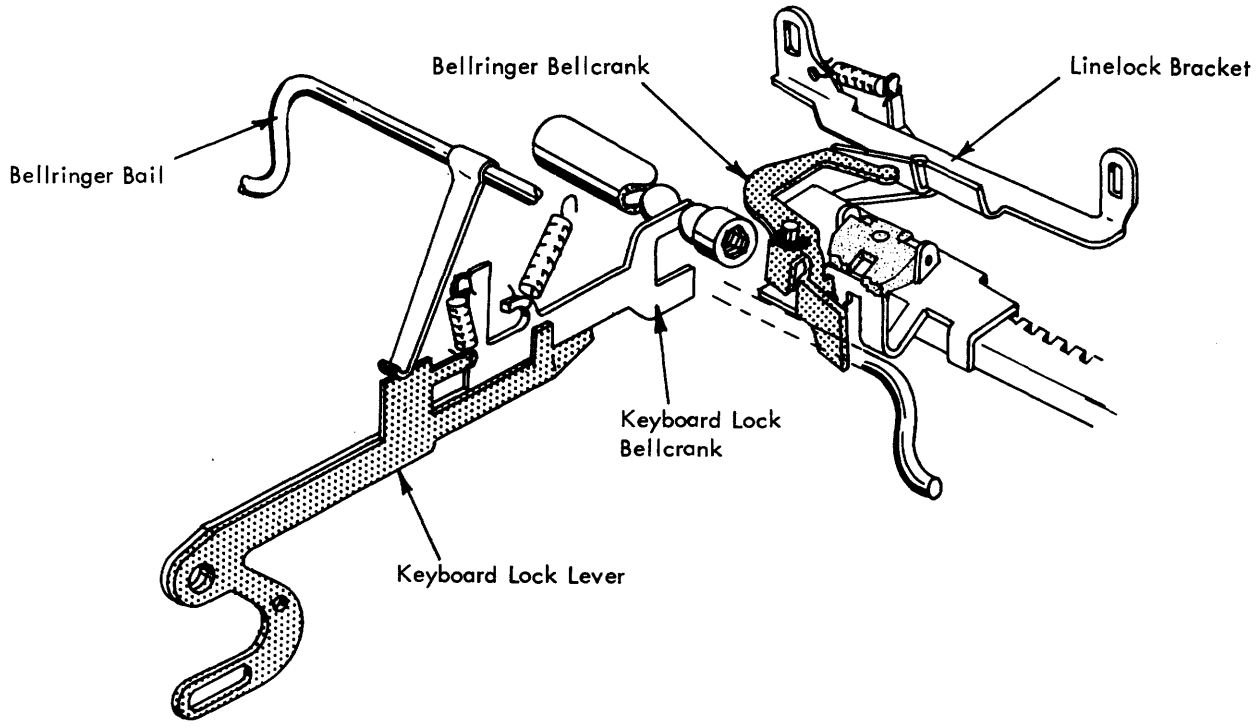


FIGURE 159. Line Lock Mechanism (new style)

The line lock bracket assembly has also been redesigned to simplify the margin release operation at the left margin. The mounting hole in the margin stop latch has been elongated. This allows the stop latch to float to the left under its restoring spring tension whenever the margin release is operated while the carrier is against the left hand margin stop. This eliminates the necessity of holding the margin release button depressed until the carrier is moved to the left.

Bell

The bell is located on the left side of the keyboard section and is rung by the bell clapper attached to a bellcrank above the bell (Fig. 160). The bell clapper bellcrank is operated by the action of the bellringer bail located across the machine just in front of the margin rack. The bellringer bellcrank pivoted on the right margin stop is contacted by the line lock bracket attached to the front of the carrier (Fig. 156). As the carrier moves to the right, the bellcrank pivots causing the bellringer bail to rotate forward. A small lever at the left end of the bail operates the bell clapper into the active position (Fig. 160). Further rotation of the bail causes the bail lever to slip off the bell clapper bellcrank allowing it to restore. An arm of the bellcrank contacts the bell mounting stud causing the bellcrank to stop suddenly. The momentum of the bell clapper causes it to spring over and strike the bell one time. When the bellringer bail is allowed to restore, the bail lever resets above the bell clapper bellcrank ready for the next operation.

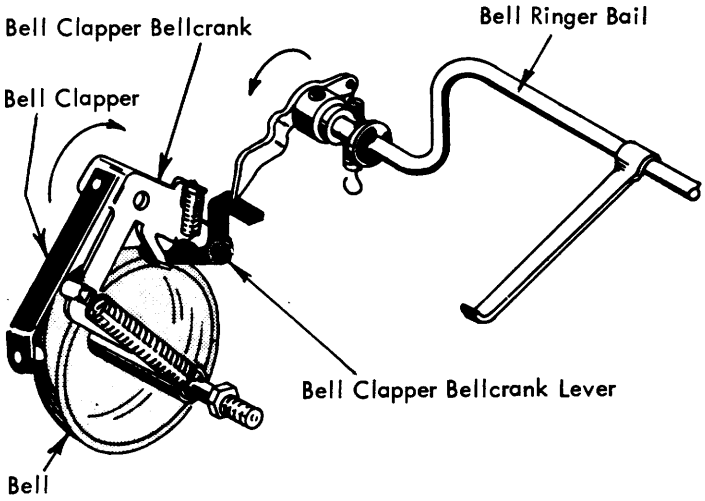


FIGURE 160. Bell Ringer Mechanism

Margin Release

The purpose of the margin release mechanism is to allow typing on either margin of the paper without repositioning the margin stops. The margin release operates by rotating the margin rack so that the margin stops move upward out of the path of the line lock bracket on the carrier.

The margin release keylever pivots at the left side of the keyboard (Fig. 161). A stud at the rear of the keylever operates in a slot in the margin release lever. The margin release lever is attached to the margin rack. Depression of the keylever causes the margin release lever to be raised. This action rotates the margin rack raising the rear of the margin stops. A lug on the left end of the margin rack remains in the path of the carrier to unlatch the carrier return if it is operated with the margin release keylever depressed.

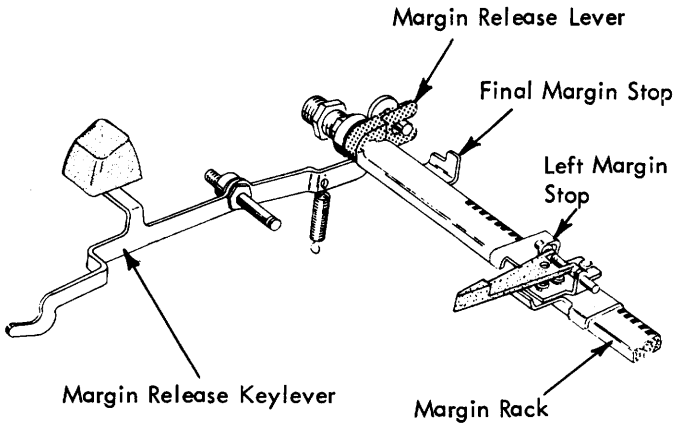


FIGURE 161. Margin Release Mechanism

An extension spring from the keylever down to a lug on the keyboard side frame restores the mechanism and holds it in the rest position.

PAPER FEED AND RELEASE MECHANISMS

Paper Feed

The purpose of the paper feed mechanism is to control both the horizontal and the vertical positions of the paper in the machine and to feed the paper vertically.

The paper feed operates by pressing the paper tightly against the platen so that it must move as the platen rotates. The paper is held against the platen by a front and rear feed roll assembly located beneath the platen (Fig. 162). Each feed roll assembly contains four rubber rollers equally spaced along the feed roll shaft and molded to the shaft.

The front feed roll shaft rests in notches of the front feed roll arms. The front feed roll arms pivot on the feed roll actuating shaft. A heavy extension spring from each front feed roll arm to the carriage tie rod supplies the pressure of the front feed roll against the platen. Various holes in the feed roll arms provide a means of adjusting the pressure.

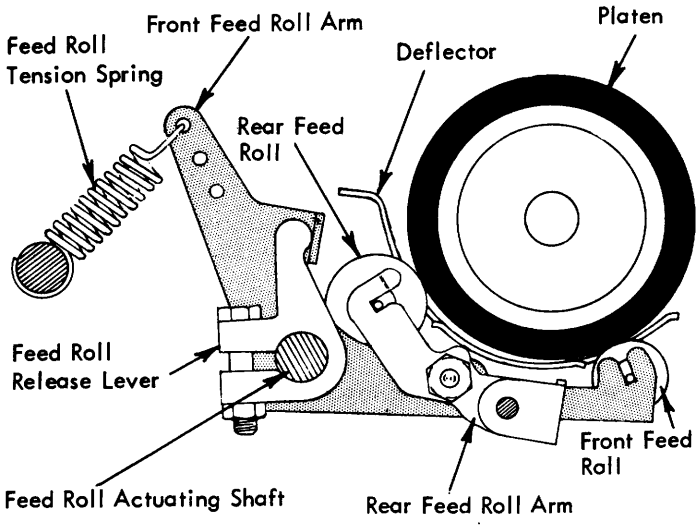


FIGURE 162. Paper Feed Mechanism

The rear feed roll shaft rests in notches of the rear feed roll arms. The rear feed roll arms pivot on studs at the front of the paper feed mounting arms that extend forward from the carriage tie rod (Fig. 164). A shoulder screw at each side connects the rear feed roll arms to the front feed roll arms. Pressure of the rear feed rolls against the platen is supplied by the front feed roll arms.

The paper deflector acts as a chute to guide the paper around the platen (Fig. 162). It is supported beneath the platen by the front and rear feed roll arms. A lug at each end of the deflector fits over a stud on the paper feed mounting arm to maintain the correct position of the deflector.

As the paper is inserted into the machine, an adjustable guide mounted on the case at the rear of the platen serves to position the paper for its left margin position. The paper deflector guides the paper between the rear feed roll and the platen. As the platen is turned, the paper is forced to move with the platen. The deflector guides the paper around the platen into position between the front feed roll and the platen.

As the paper is fed farther, the end of the paper is guided upward by the line gage card holder attached at the rear of the carrier (Fig. 163). The line gage card holder assists in holding the typing material against the platen in the printing area. A scale on each side of the holder assists the typist in reinserting material into the machine to a specific printing point. The marks on the scale indicate the middle of a character space and the horizontal line on the scale indicates the bottom of the writing line. A single mark in the middle at the top of the card holder indicates the middle of the next character to be typed.

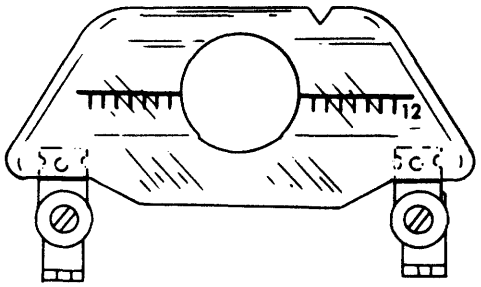


FIGURE 163. Line Gage Card Holder

Above the writing line the paper is engaged by two rubber rollers mounted on the paper bail. These rollers hold the paper against the platen above the writing line so as to reduce the possibility of overprinting on the paper. The rollers also feed the paper vertically after the bottom of the paper has left the front feed roll.

The paper bail is supported by a lever at each end that pivots front to rear on the side of the machine. A hairpin spring operates each bail lever as a toggle to hold the bail rolls either to the rear against the platen or forward in the released position.

The bail rolls are free to rotate around the bail shaft. A spring band inside each bail roll grips the bail shaft to restrict the lateral motion of the bail rolls. The bail rolls may be positioned along the bail shaft by exerting sufficient pressure to overcome the friction of the spring band on the bail shaft.

Paper Release

The pressure of the front and rear feed rolls is released from the platen to allow the operator to position the paper more accurately and to allow easier insertion and removal of the paper. Paper release is accomplished by pulling forward on the paper release lever located at the right end of the carriage (Fig. 164). The front of the paper release lever cams the top of the feed roll release arm forward to rotate the feed roll actuating shaft.

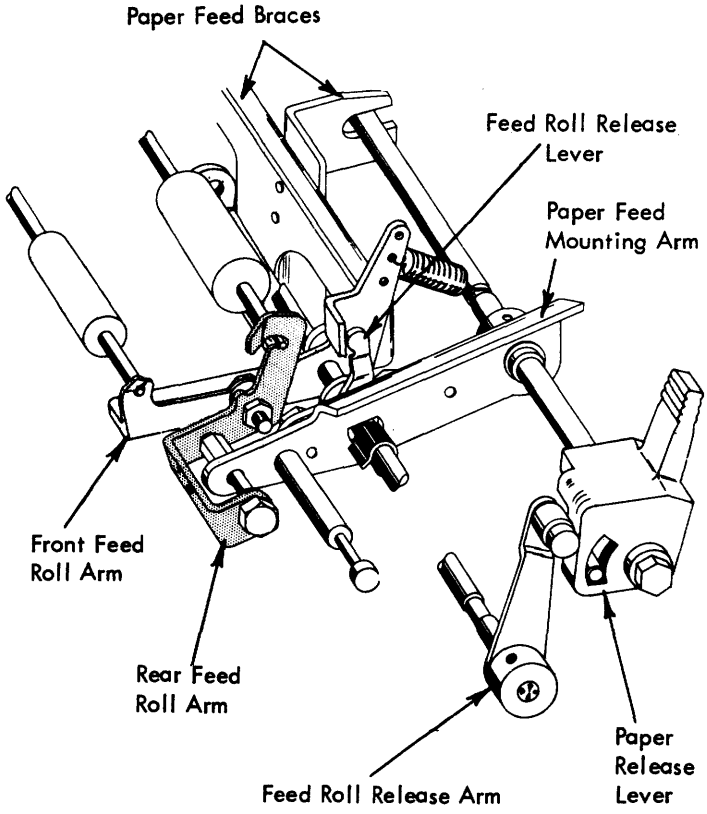


FIGURE 164. Paper Release Mechanism

Two feed roll release levers are clamped to the feed roll actuating shaft and rest behind a lug of each front feed roll arm. As the shaft rotates, the feed roll release levers rotate the front feed roll arms down, away from the platen. The rear feed roll arms are, in turn, forced away from the platen by their connection to the front feed roll arms. When the paper release lever has been pulled all the way forward, the end of the feed roll release arm detents over the point at the front of the paper release lever to hold it in the released position.

FABRIC RIBBON

The ribbon mechanism is divided into two separate distinct mechanisms. They are the ribbon lift mechanism and the ribbon feed and reverse mechanism. The ribbon lift mechanism raises the ribbon to the printing point before the type head prints and then lowers to allow a visible writing line. The feed and reverse mechanism moves the ribbon laterally past the printing point to provide an unused portion for the next typing operation. It also reverses the feeding direction when the end of the ribbon is reached.

The ribbon is a 9/16" fabric ribbon enclosed in a disposable cartridge unit for clean handling. The cartridge unit contains two spools on which the ribbon is wound. The ribbon is constantly fed from one spool to the other and back again until the ink supply has been depleted. Replacing the ribbon is a clean, effortless operation.

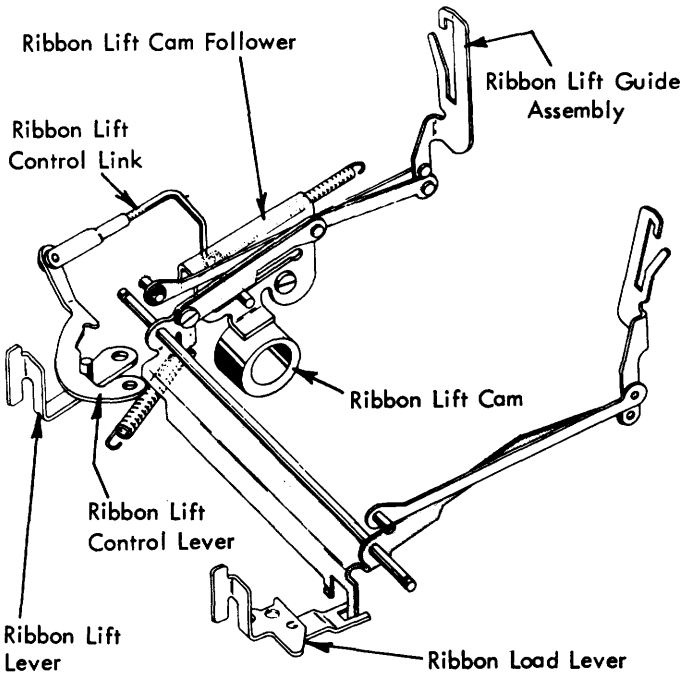


FIGURE 165. Ribbon Lift Mechanism

Located to the right of the pointer on the carrier assembly is the ribbon load lever (Fig. 165).

When pushed to the right, the load lever forces the ribbon lift guide into an extreme lift position for accessibility. The ribbon load lever is detented to hold the ribbon lift guide in the high lift position. The cartridge can be removed from the ribbon feed plate by simply lifting it off. The ribbon can then be easily removed from the guide without touching the ribbon.

Installing a new ribbon is just the reverse of the above. With the ribbon lift guide still in the extreme lift position, the ribbon can be inserted into the guide and the cartridge snapped into place all in one motion. Tapered lugs on the sides of the ribbon feed ratchet cores automatically guide the ribbon spools into the correct position. Guide lugs at each side of the feed plate maintain the lateral position of the cartridge. Retainer springs attached to the guide lugs hold the cartridge down to prevent vibration. After the ribbon is installed, the load lever is moved back to the left to allow the ribbon lift guide to restore to its normal position ready for a typing operation.

Ribbon Lift

The ribbon lift mechanism consists of a cam, cam follower, control mechanism, and the ribbon lift guide assembly (Fig. 165).

The mechanism is mounted to the carrier assembly and is transported by the carrier along with the type head. The ribbon lift cam is a single lobed cam set-screwed to the left end of the print sleeve. The cam has a punch mark on the right side that must line up with the print sleeve key-way to insure that the cam is not out of time with the print operation. Care must be taken if the cam is replaced, to insure that it is not installed in a reversed position.

The cam makes one complete revolution each time a cycle operation occurs. The ribbon lift cam follower pivots on the carrier casting above and to the rear of the cam (Fig. 165). Each operation of the cam raises the cam follower. The cam follower contains a long slot. In the slot is the end of the ribbon lift control link. The ribbon lift guide assembly rests directly above the control link. As the cam follower is raised, the control link forces up on the ribbon lift guide assembly. The guide assembly pivots on the carrier casting at the front causing the ribbon to be raised at the rear. A flat link from each side of the ribbon lift guide attaches to two pins at the front of the carrier to maintain the ribbon lift guide in a vertical position.

The height to which the ribbon will be raised is determined by the position of the ribbon lift control link in the slot of the cam follower. When the link is in the extreme rear of the slot, very little motion is obtained from the cam follower; consequently the ribbon lift guide is not raised at all. This is called the stencil position. It is so called because the ribbon is not used in typing stencils.

As the ribbon lift control link is moved toward the front in the slot, more and more motion is obtained from the cam follower. The link is also moved nearer the pivot point of the ribbon lift guide assembly so that the motion obtained from the follower is more effective in raising the ribbon. The nearer the front the link is moved, the higher the ribbon will be raised.

In addition to the stencil position there are three ribbon lift positions which may be selected by the operator. The ribbon lift control link is attached to the ribbon lift control lever pivoted under the front of the carrier casting (Fig. 165). The control lever is spring-loaded to the rear against a stud on the ribbon lift lever. The ribbon lift lever has a button located just to the left of the carrier pointer. Moving the button to the left causes the stud on the lever to force the ribbon lift control lever and link toward the front. The stud of the ribbon lift lever seats into notches in the control lever to maintain its position until changed by the operator.

The extreme right hand position of the button is the stencil position where no ribbon lift is available. The next position to the left is the low lift position used for typing on the top half of the ribbon. The third position allows typing in the middle of a single colored ribbon to get maximum life from the ribbon. The extreme left position of the button is the high lift position and is used for typing on the bottom half of the ribbon.

The ribbon lift guide assembly is spring loaded into the rest position to insure that it will restore rapidly and to prevent overthrow of the ribbon due to the momentum of the lift mechanism.

Ribbon Feed and Reverse

1. Ribbon Feed

The ribbon feed and reverse mechanism is a compact unit mounted at the top of the carrier just in front of the type head. The mechanism is detachable as a unit for repair or replacement purposes.

Each ribbon spool of the ribbon cartridge unit fits over the core of a nylon ribbon feed ratchet (Fig. 166). Rotation of a ratchet causes the spool to turn and wind up the ribbon. The ratchets are operated by a pawl that moves front to rear between them to force the ratchet teeth to the rear.

The position of the feed pawl determines which ratchet is fed as the pawl moves to the rear. The feed pawl pivots on a pin beneath the ribbon feed plate and extends up through a hole in the plate. The mounting of the pawl permits left to right as well as front to rear movement. An extension spring attached to the pawl restores the pawl to the rest position each time it operates (Fig. 166). The spring also holds the pawl left or right into engagement with the correct ribbon feed ratchet depending upon the direction of pull of the spring. The forward end of the spring is attached to a lever called the ribbon feed detent lever. The detent lever pivots on the ribbon feed plate.

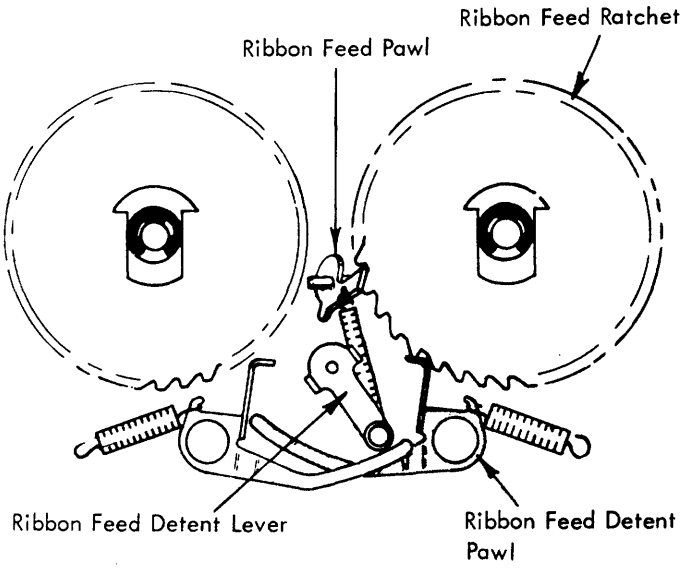


FIGURE 166. Ribbon Feed Mechanism - Top View

When the lever is moved to the right, the feed pawl is pivoted into engagement with the right hand feed ratchet (Fig. 166). When the detent lever is moved to the left, the feed pawl engages to feed the left hand ratchet.

The ribbon feed pawl is powered to the rear by the action of the ribbon feed cam located directly in the middle of the print sleeve. The cam is keyed to the sleeve and rotates one complete revolution on each operation. A sliding cam follower transfers the motion of the cam to the ribbon feed bellcrank which actuates the feed pawl to the rear (Fig. 167). Sufficient motion is available from the cam to cause a two teeth feed of the ratchet. The timing of the ribbon feed cam causes the ribbon feed action to occur early in the cycle operation. At the time the type head prints, the ribbon has completed its feeding operation except for the restoring of the feed pawl. Care must be taken in replacing the feed cam to insure that it is not installed in a reverse position. A V-shaped notch in the large part of the cam must be toward the right. Reversing the cam will change the timing of the ribbon feed.

As the feed pawl restores to the front, it slides along the teeth of the ratchet into the rest position. The drag of the pawl along the teeth tends to rotate the ratchet backward and unwind the ribbon. To prevent any backward rotation, a detent pawl is spring-loaded into the teeth of the ratchet to allow feed in one direction only (Fig. 166).

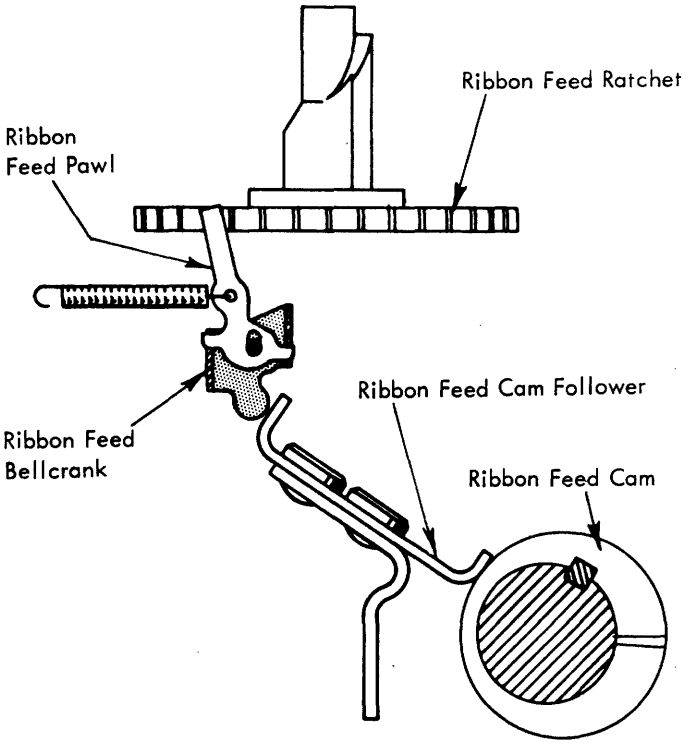


FIGURE 167. Ribbon Feed Mechanism - Side View

In order for the ribbon to wind onto one spool it must be unwound from the other spool. Each ratchet has a detent pawl to prevent rotation in the unwinding direction. The detent pawl must be disengaged from the supply side in order for ribbon feed to occur. Each detent pawl has a long curved extension resting against a roller on the ribbon feed detent lever (Fig. 166). When one detent pawl is engaged with its ratchet, the other is disengaged depending upon the position of the ribbon feed detent lever.

Two flat springs are mounted to the ribbon feed plate at the rear so that they rest against the ribbon feed ratchets. The slight drag applied by the springs prevents the jerk of the ribbon feed operation from spinning the supply spool and spilling off excessive ribbon.

2. Ribbon Reverse

The ribbon is fed from one spool to the other until the supply spool is emptied. The ribbon feed pawl must then be moved to the ratchet of the empty spool to begin feeding the ribbon in the opposite direction. To achieve a ribbon reversing operation, it is merely necessary to move the ribbon feed detent lever from one position to the other. The ribbon feed pawl spring then pulls the pawl into the reversed position. The reversing operation is the same for both sides except that the direction of parts movement is opposite.

The reversing operation requires a sensing mechanism to determine when the reverse is to take place and a means of powering the detent lever from one position to the other. Each ribbon feed ratchet core contains a small bellcrank called the ribbon reverse trigger (Fig. 168). As long as there is ribbon

around the spool, the ribbon holds the reverse trigger into the core in the inactive position. When the ribbon spool is emptied, a hairpin spring forces the reverse trigger out of the core through a slot in the ribbon spool. A portion of the reverse trigger pivots down through a hole in the ratchet into position below the ratchet.

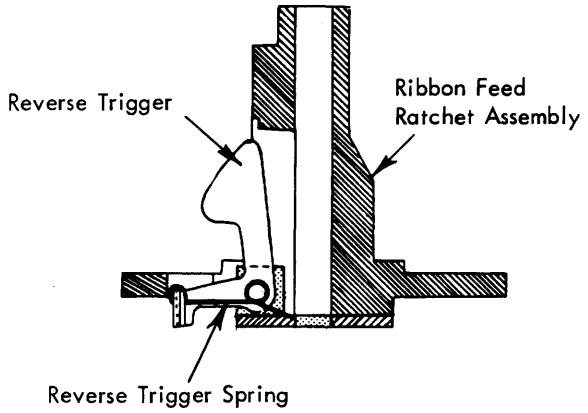


FIGURE 168. Ribbon Reverse Trigger

The empty spool rotates slightly farther causing the reverse trigger to contact and actuate the reverse lever which pivots just below the ratchet (Fig. 169). The reverse lever is connected, by means of a flat link, to an arm of the reverse yoke beneath the ribbon feed plate. The yoke is pivoted by operation of the reverse lever. A stud on the yoke at the rear of the pivot point extends up through the feed plate into a slot in the reverse interposer. Movement of the yoke positions the front of the reverse interposer left or right depending upon which ribbon spool is being emptied (Fig. 170).

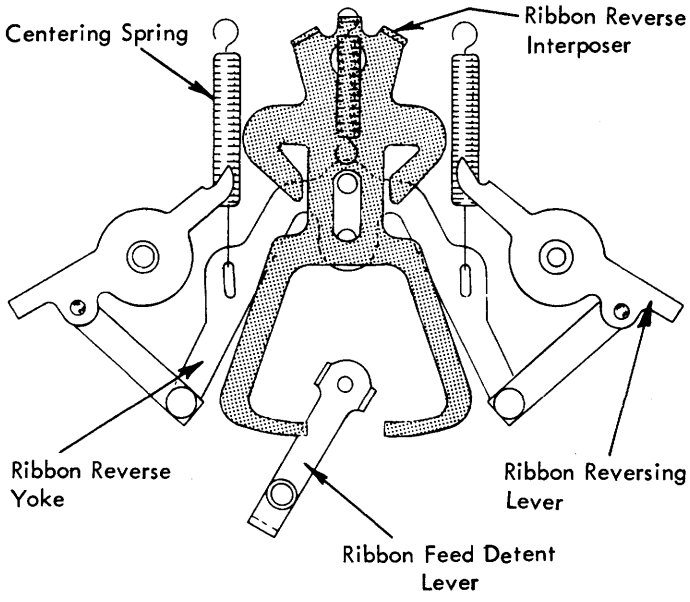


FIGURE 169. Ribbon Reverse Mechanism - Rest Position

Positioning the reverse interposer does two things. A hook at the front of the interposer hooks around a lug on the ribbon feed detent lever (Fig. 171). The interposer lever, mounted on the interposer, is positioned into the path of the ribbon feed pawl. The next operation of the ribbon feed cam causes the feed pawl to drive the reverse interposer to the rear. The hook at the front of the interposer pulls the lug of the detent lever

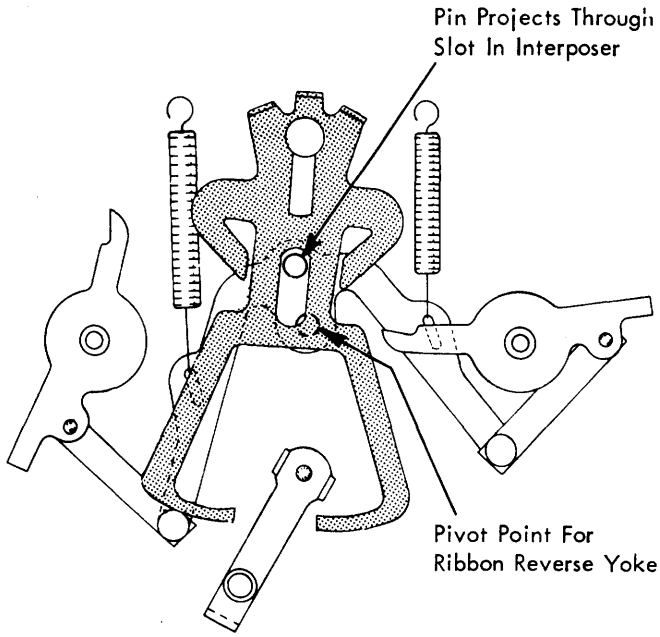


FIGURE 170. Ribbon Reverse Mechanism - Active Position

to the rear causing the detent lever to pivot to the opposite position. Movement of the detent lever disengages the detent pawl at the full spool and allows the pawl to engage the ratchet at the empty spool. As the feed pawl restores, its spring pivots it over into engagement with the opposite ratchet.

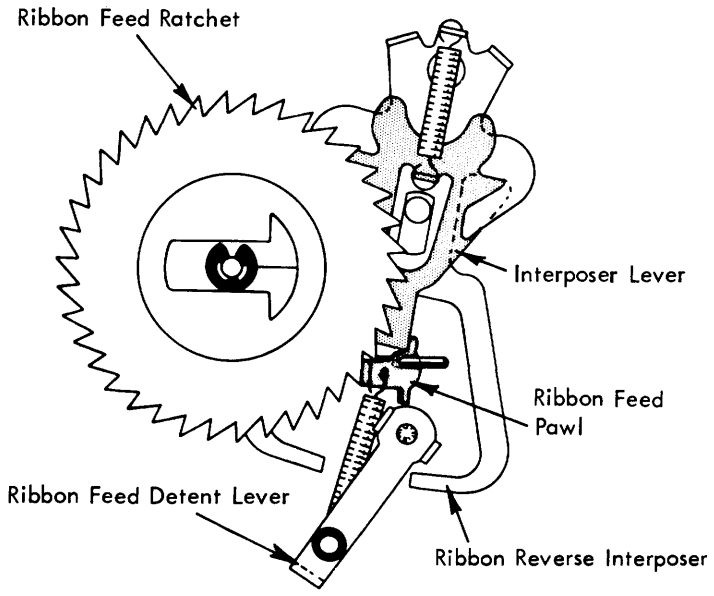


FIGURE 171. Ribbon Reverse Interposer and Lever

The purpose of the interposer lever is to prevent the motion of the ribbon reverse interposer from being choked off during a reversing operation by the ribbon feed pawl, if the feed pawl should become trapped between the ribbon feed ratchet and the interposer lever.

Stencil Lockout

When the ribbon lift lever is placed in its no lift or stencil position, a latch called the ribbon feed latch is allowed to interrupt and hold the ribbon feed cam follower from following the ribbon feed cam back to its low point (Fig. 172). This causes the ribbon feed cam follower to become crippled thereby locking out the ribbon feed operation. The ribbon feed latch mounts on a stud on the ribbon feed follower mounting bracket and is spring loaded into its active position by a small hairpin spring. The ribbon lift lever controls the position of the latch. When it is pushed to the right into the stencil position, it allows the latch to rotate against the top surface of a lug on the cam follower. As the follower is operated forward by the feed cam, the latch hooks behind the lug on the follower and prevents the follower from following the cam back to its low point. This cripples the feed operation because the cam follower no longer receives enough motion to operate the ribbon feed pawl effectively. Pushing the ribbon lift lever to the left (out of the stencil position) cams the latch into its inactive position, thereby releasing the follower so that it can be operated by the feed cam.

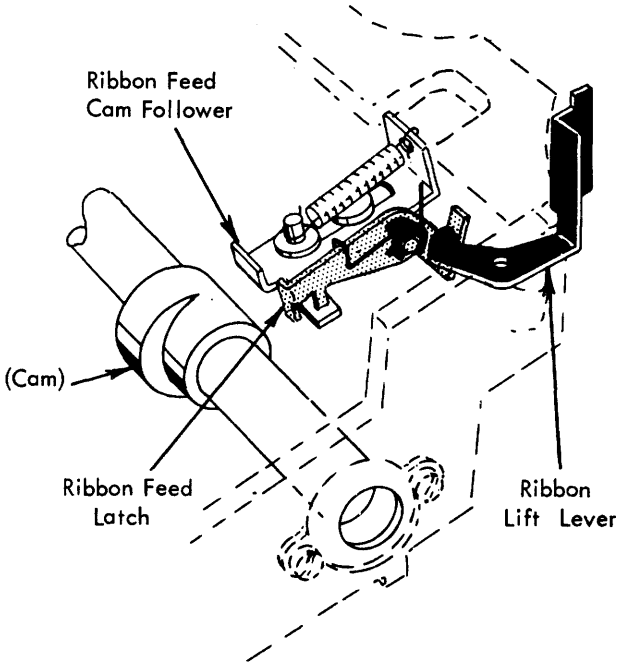


FIGURE 172. Stencil Lockout Mechanism

FILM RIBBON

The film ribbon mechanism in its entirety mounts on and moves with the carrier in a similar manner as the fabric ribbon mechanism. The spool of ribbon, which consists of approximately 240 feet of film ribbon wrapped tightly about a plastic core, mounts on a permanent supply spool on the left side of the carrier (Fig. 173). Matching flutes, between the plastic core of the spool of ribbon and the permanent supply spool, causes both the spool of ribbon and the permanent supply spool to rotate together during a ribbon feed operation.

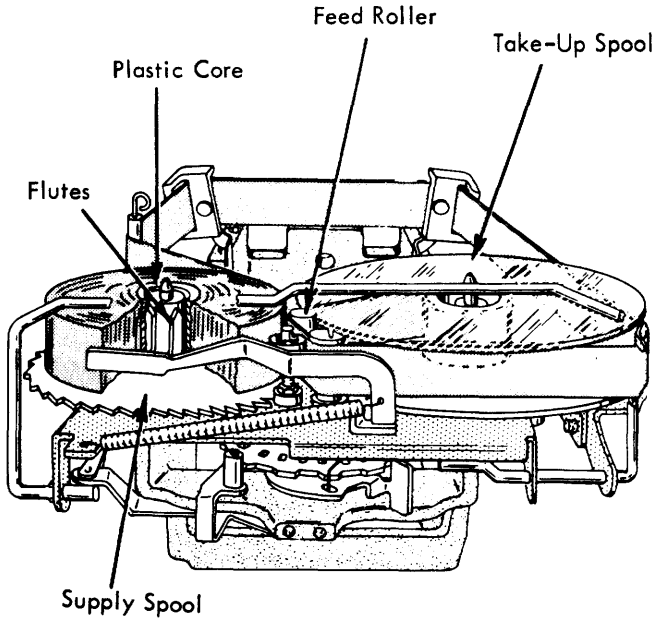


FIGURE 173. Film Ribbon Mechanism

As the ribbon comes off the supply spool it threads around the ribbon circuit to the take-up spool. The take-up spool is a disposable transparent spool mounted on the right side of the carrier (Fig. 173). Once the ribbon has been completely used and fed onto the take-up spool, both the take-up spool and the plastic core from the supply side are removed and discarded. The new spool of ribbon, to be installed, comes equipped with its own take-up spool fastened to the end of the clean leader.

Ribbon Feed

Clockwise rotation of the feed roller causes the ribbon to feed from the supply spool to the take-up spool (Fig. 173). The feed roller is directly connected to a star shaped wheel called the feed and lift wheel (Fig. 174). This wheel, which is lo-

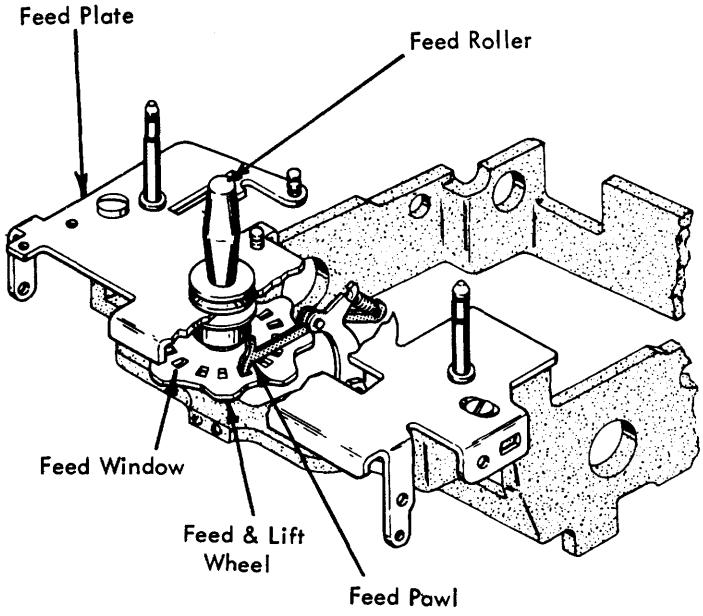


FIGURE 174. Ribbon Feed Operation

cated on the underside of the ribbon feed plate, fastens to the bottom of the feed roller by a bronze hexagon-headed screw that has a left handed thread.

The feed and lift wheel contains sixteen feed windows laid out in a circular pattern. A feed pawl operates in these windows in a manner similar to an index pawl operating in a platen ratchet. Each time the feed pawl is powered forward, it pushes on one of the feed windows causing both the wheel and the feed roller to rotate.

The ribbon feed cam located on the print sleeve supplies the motion to the feed pawl (Fig. 175). The motion from the cam is transmitted to the feed pawl through a cam follower. A barrel shaped roller mounted on the cam follower by an eccentric stud rides against the ribbon feed cam. The follower mounts on a bracket which is fastened to the front carrier casting by two hexagon headed screws. An extension spring, anchored to one of these screws, loads the follower against the cam. The feed pawl mounts at the top of the cam follower by a shouldered rivet and is spring loaded into engagement with the feed and lift wheel.

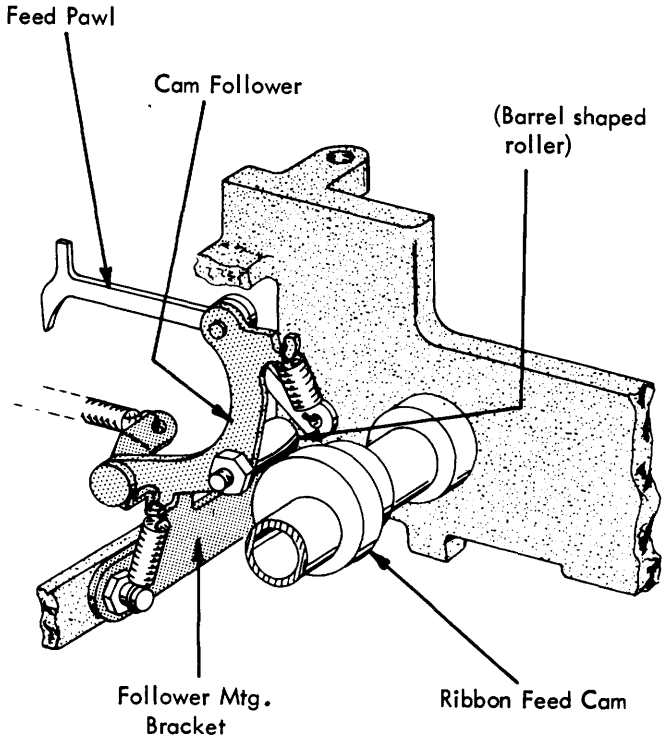


FIGURE 175. Ribbon Feed Mechanism

A ribbon feed operation occurs in the following manner: When the machine is at rest, the barrel shaped roller on the cam follower is on the high point of the feed cam (Fig. 176). In this position the feed pawl is all the way forward, and is engaged with one of the feed windows in the feed and lift wheel.

As the feed cam begins to rotate clockwise towards its low point, the feed pawl will begin to move toward the rear of the machine. This movement causes the feed pawl to be cammed up and out of the feed window. The pawl then slides along the top surface of the wheel towards the next feed window (Fig. 177).

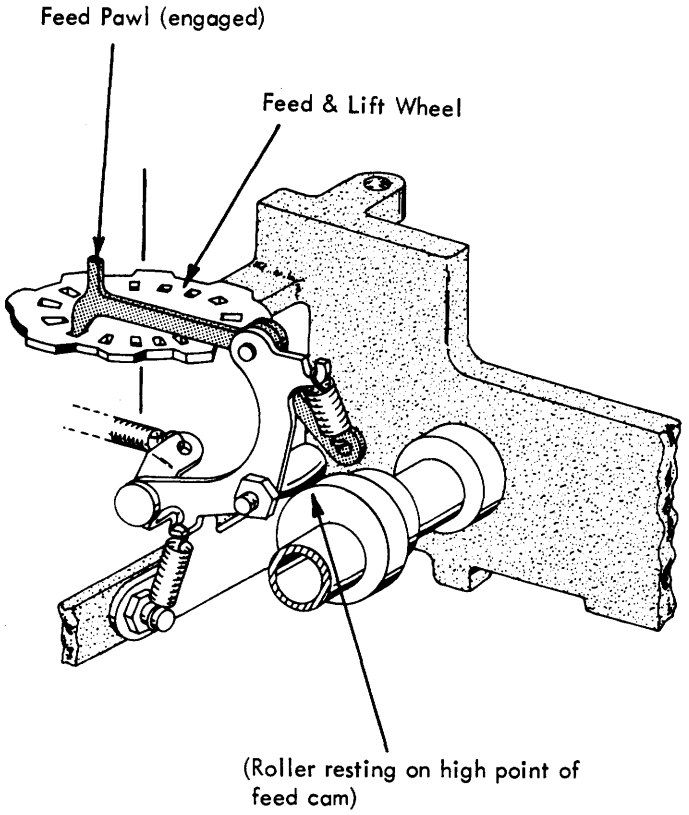


FIGURE 176. Feed Cam at Rest

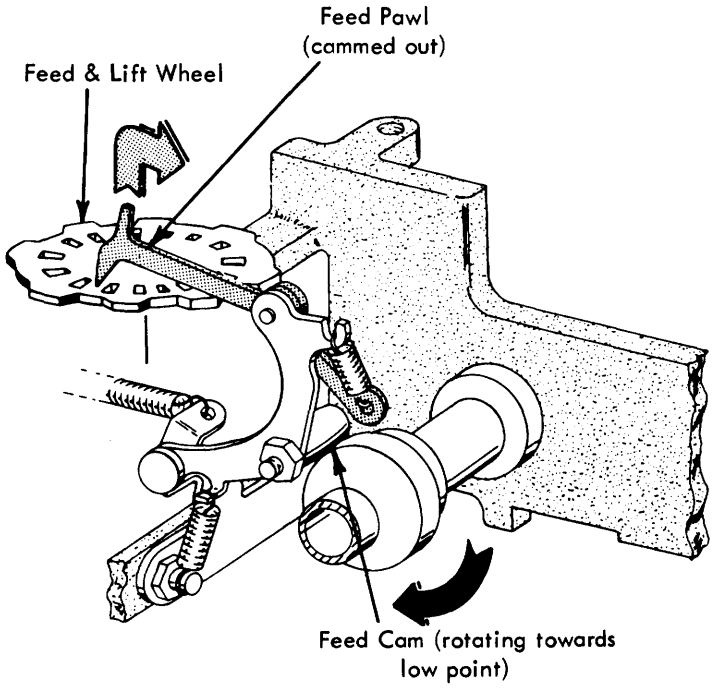


FIGURE 177. Follower Approaching Low Point

Slightly before the barrel shaped roller on the cam follower reaches the low point of the feed cam, the feed pawl drops into the next feed window (Fig. 178). Continued rotation of the feed cam back to its high point causes the feed pawl to rotate the feed and lift wheel 1/16 of a turn resulting in a ribbon feed operation.

Notice that the timing of the ribbon feed cam causes the ribbon feed action to occur late in a print cycle. At the time the type head prints the ribbon feed pawl has just begun to rotate the feed and lift wheel to cause a ribbon feed operation. By the time the print cycle completes the feed pawl is in its extreme forward position (Fig. 176) and the ribbon feed operation is accomplished.

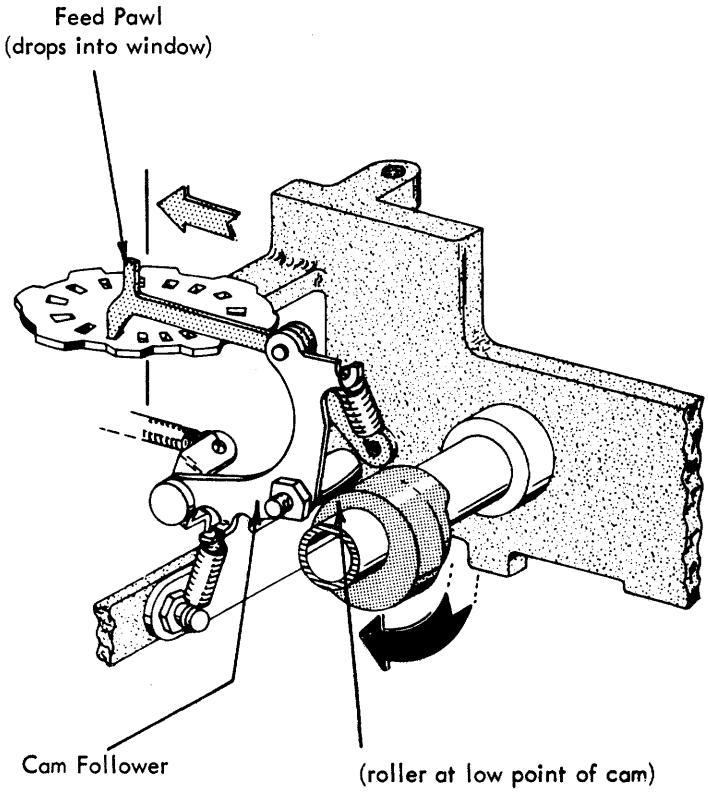


FIGURE 178. Follower at Low Point

During the early portion of a print cycle the feed and lift wheel tends to rotate backwards with the feed pawl when the feed pawl is being cammed out of the feed window as it travels toward the rear of the machine. To prevent the feed and lift wheel from rotating backwards the wheel is detented in position at the completion of each feed operation. This is accomplished by a detent spring that fastens to the underside of the feed plate and operates in the feed windows of the wheel (Fig. 179).

Beginning at the supply spool, let's trace the path of the ribbon to the take-up spool. As the ribbon comes off the spool of ribbon on the supply spool it immediately passes around a wire shock spring (Fig. 180). This shock spring is mounted to the top of the feed plate by a binding screw. Its purpose is to absorb the shock given to the ribbon during a feed operation plus eliminate all slack in the ribbon during a ribbon lift operation. The function of the black roller on the shock spring is to reduce friction as the ribbon slides over the shock spring.

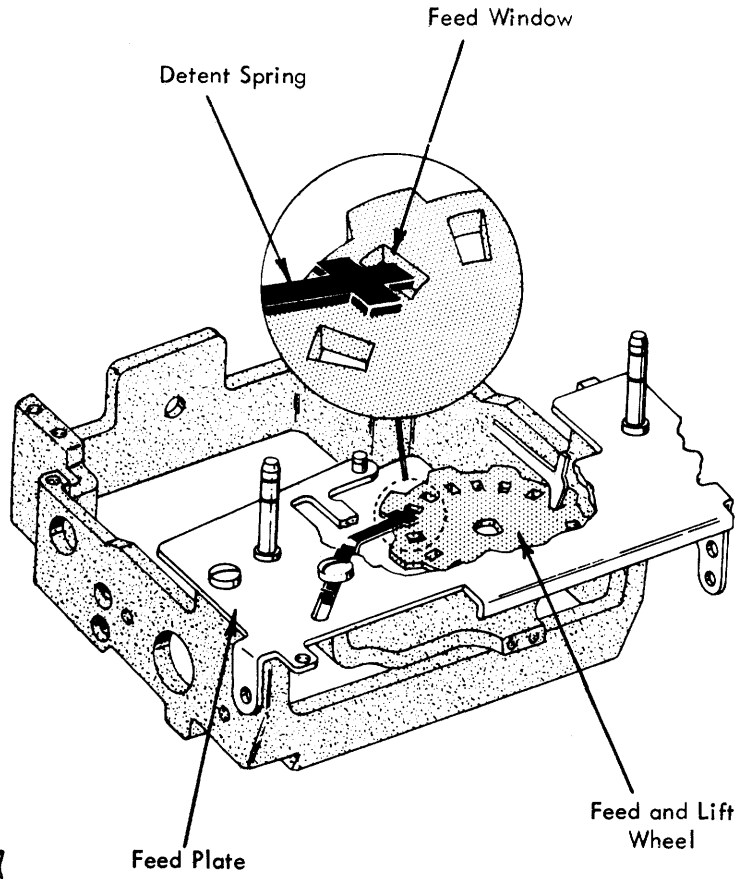


FIGURE 179. Feed & Lift Wheel Detent Spring

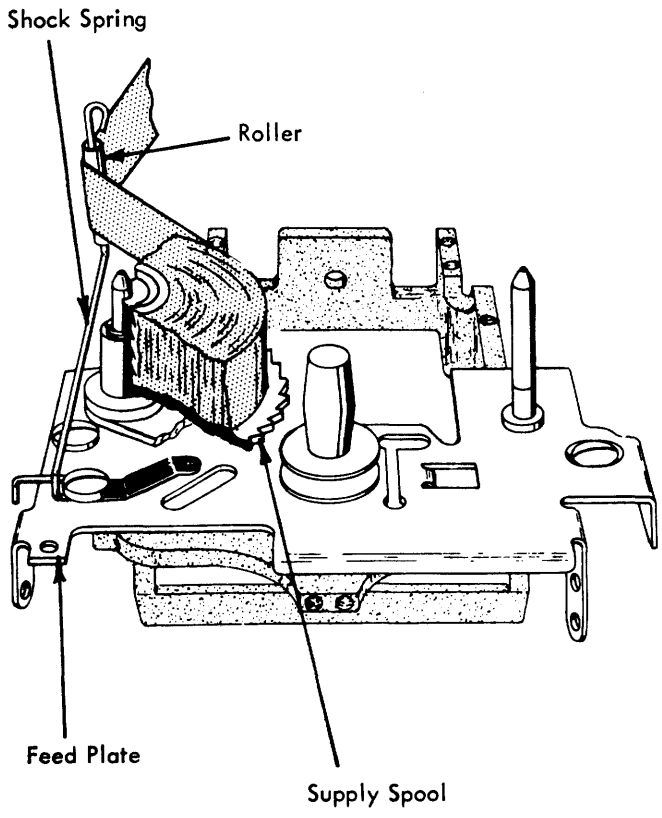


FIGURE 180. Ribbon Shock Spring

After passing around the shock spring the ribbon threads through two plastic guides on the ribbon lift guide assembly (Fig. 181). The guide assembly pivots about a fulcrum wire mounted at the front of the carrier casting. The function of the guide assembly is to elevate and guide the ribbon in front of the typehead during a print operation. A flat link running from each guide arm is anchored to a pin at the front of the carrier. These links maintain the plastic guides in a vertical position throughout a ribbon lift operation. The plastic guides, mounted on the guide arms, pivot freely on shouldered rivets. This type of mounting permits the tension on both the top and bottom of the ribbon to automatically equalize and remain equal throughout a ribbon lift operation.

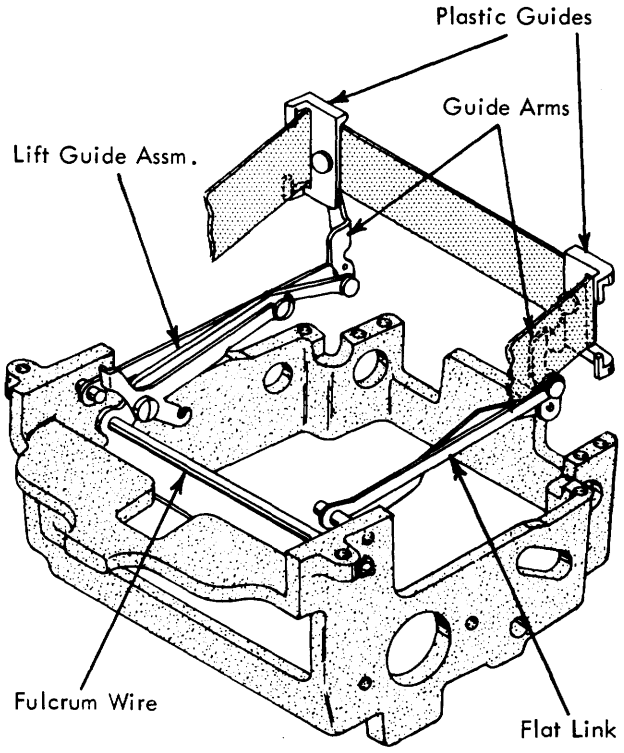


FIGURE 181. Ribbon Lift Guide Assembly

From the guide assembly the ribbon threads forward passing around the right hand corner guide (Fig. 182). This corner guide helps to dampen out the effects of the lift operation that is felt on the ribbon as it leaves the lift guide assembly. The corner guide stabilizes the ribbon before it enters the tracking post.

Before the ribbon enters the feed and pressure rollers it passes over a flanged post called the tracking post (Fig. 182). The purpose of this post is to make the ribbon engage the feed roller at the same angle throughout the use of an entire spool of ribbon. As you can see in Figure 182 the ribbon is being deflected between the corner guide and the tracking post by the used ribbon that has accumulated on the take-up spool. Without the tracking post, the angle that the ribbon enters the feed roller would be constantly changing as the diameter of the used ribbon on the take-up spool increased. This condition could cause ribbon tracking problems at the feed roller. The tracking post aids the system in maintaining a constant ribbon tracking characteristic at the feed and pressure rollers.

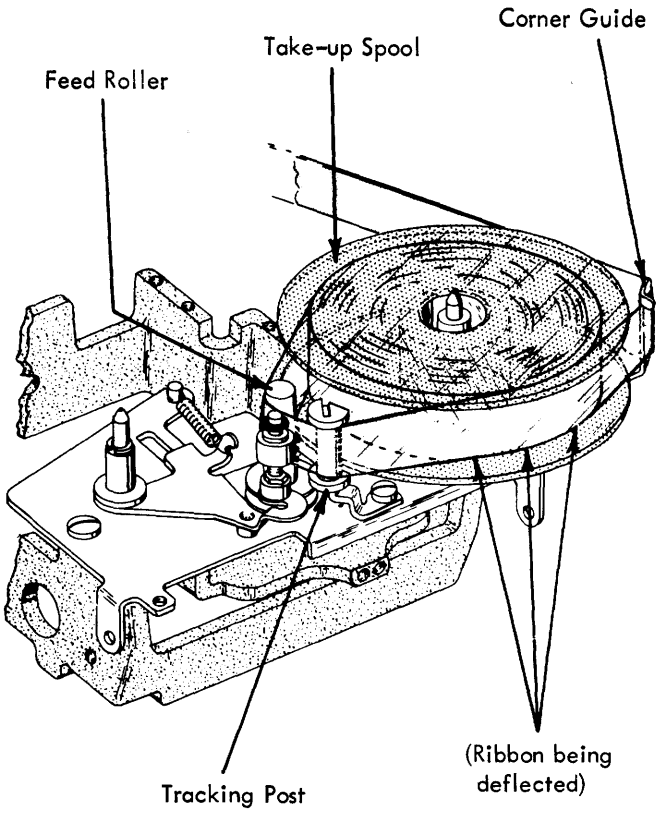


FIGURE 182. Corner Guide & Tracking Post

As the used ribbon leaves the tracking post and passes around the feed roller, a small rubber roller called the pressure roller loads the ribbon against the feed roller (Fig. 183). The function of the pressure roller is to insure that there will be no ribbon slippage at the feed roller during a ribbon feed operation.

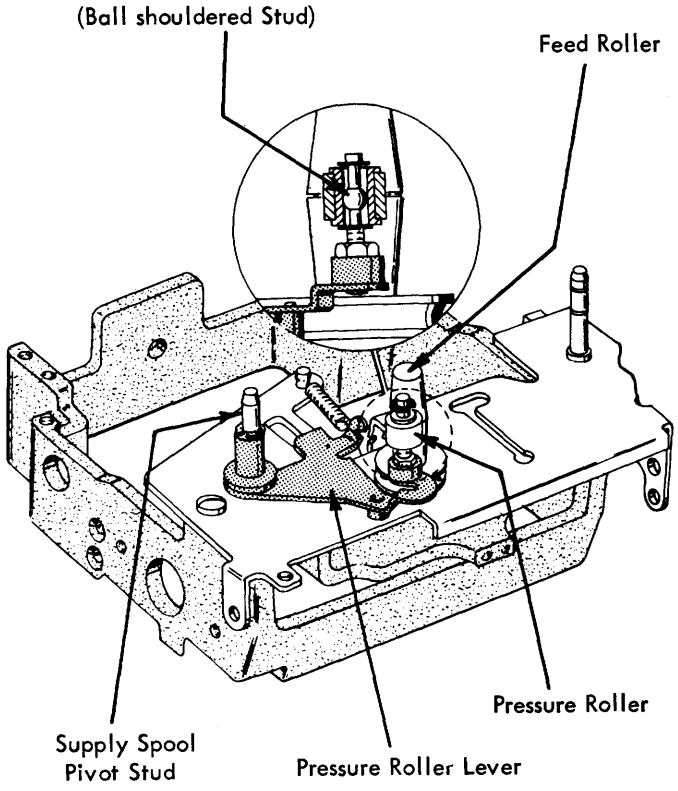


FIGURE 183. Pressure Roller Mtg.

The pressure roller mounts on a lever which pivots on the supply spool pivot stud. A heavy extension spring fastened to the lever loads the pressure roller against the feed roller. The pressure roller mounts freely on a ball shouldered portion of its adjustable mounting stud. This ball shouldered type mounting permits the rotational axis of the pressure roller to change as the roller rides over the embossed portions of the used ribbon during a feed operation. By giving the pressure roller the freedom to seek its own position, a better tracking characteristic is achieved at the feed roller.

Once the ribbon leaves the feed and pressure roller, the used ribbon is wound onto the transparent take-up spool. The take-up spool receives its motion from the feed mechanism by a friction type drive system.

A drive pulley located directly below the feed roller rotates with the feed roller during a ribbon feed operation (Fig. 184). This drive pulley supplies the motion to the take-up pulley by means of a drive spring. The take-up pulley, driven by the drive spring, rotates about the take-up spool pivot stud and is held in place by a "C"-clip. The shape of the belt groove in the take-up pulley is designed slightly different as compared to that of the drive pulley. This is to permit all of the necessary slippage of the drive spring to occur at the take-up pulley and not at the drive pulley. Two hooked lugs on the top face of the take-up pulley project into corresponding slots in the bottom of the transparent take-up spool. These lugs provide a locking type connection between the take-up pulley and the disposable take-up spool. Thus, a constant take-up drive is insured at the take-up spool.

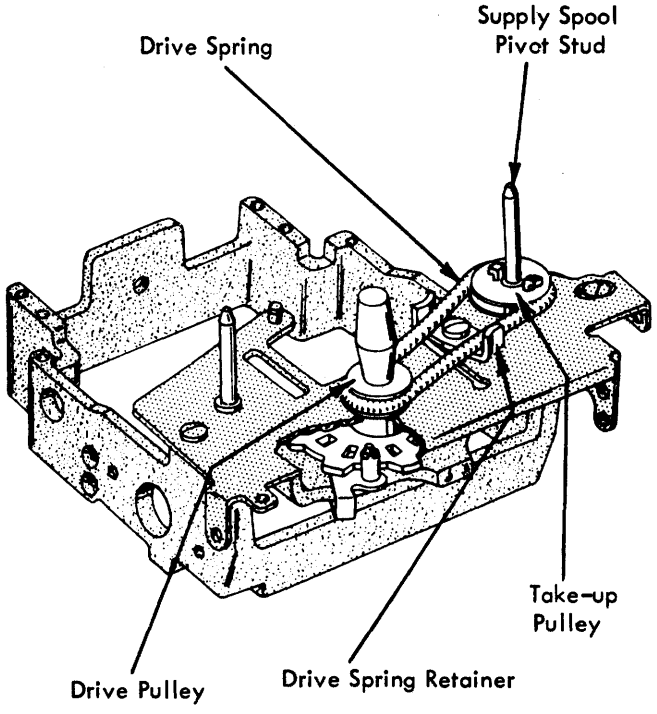


FIGURE 184. Take-Up Spool Drive

The function of the drive spring retainer is to dampen the oscillations of the drive spring during an operation. Without the retainer, oscillations of the drive spring would cause the spring to strike other components mounted nearby. This would result in damage to the drive spring which may lead to erratic take-up tension.

To maintain a stable ribbon tracking characteristic from the supply spool to the feed and pressure rollers, the ribbon must be kept slightly taut throughout the ribbon circuit. Any slackness in the system will affect the tracking of the ribbon. Maintaining the ribbon taut is accomplished by means of the supply drag lever in conjunction with the shock spring (Fig. 185). The supply drag lever is mounted on the feed plate by the same screw that mounts the tracking post. The drag lever pivots freely about a shoulder on this screw and is spring loaded against the spool of ribbon on the supply spool. Its function is to provide a constant drag on the outer wraps of ribbon. This drag plus the effects of the shock spring eliminate any slackness in the ribbon circuit during a ribbon feed operation.

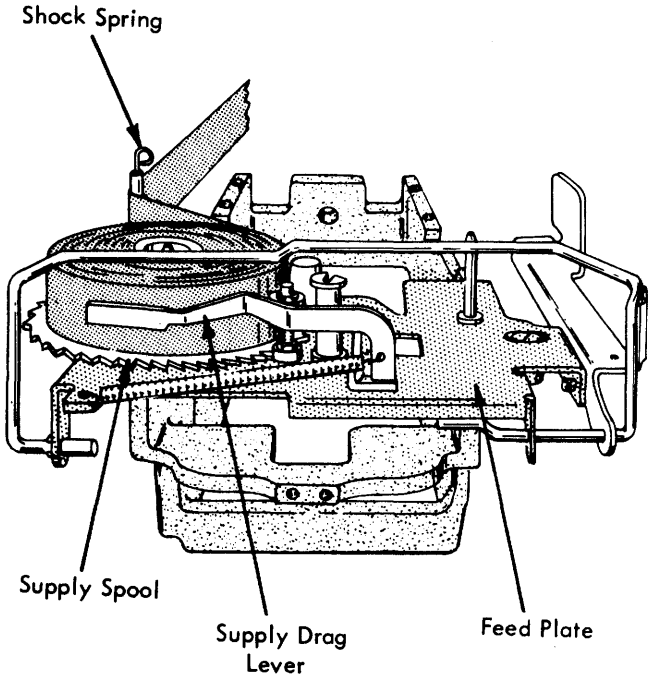


FIGURE 185. Supply Drag Lever

To keep the spool of ribbon tightly wrapped on the supply spool, a slight drag must be felt at the core of the ribbon supply. This is achieved by the supply spool drag spring which fastens to the feed plate by the same screw that anchors the shock spring. The drag spring provides a frictional drag to the underside of the ribbon supply spool. Without this drag, both the core and the inner wraps of ribbon would tend to turn within the outer wraps of ribbon on the supply spool. The end result would be a tendency for the spool of ribbon on the supply spool to grow in diameter which may eventually lead to ribbon spillage. This effect is mostly due to the momentum that is developed within the spool of ribbon during a ribbon feed operation.

The carrier vibrations that inherently occur during a repeat spacebar or backspace operation causes the ribbon supply spool to rotate in the unwinding direction, thereby permitting the ribbon to become slack in the circuit. To prevent this from occurring, a brake called the supply spool brake (Fig. 186) engages the ratchet teeth on the supply spool whenever the ribbon mechanism is not in a feed operation. The brake pivots on a stud on the feed plate and is spring loaded into engagement with the ratchet teeth on the supply spool.

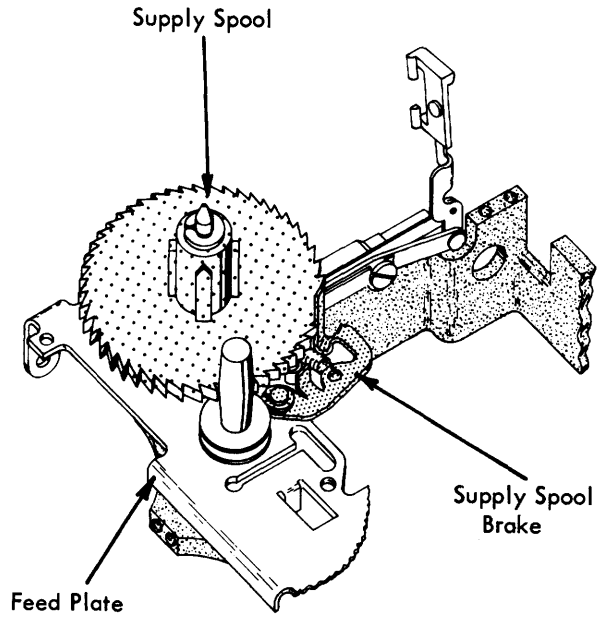


FIGURE 186. Supply Spool Brake

During a feed operation the supply spool brake must be disengaged from the supply spool. The mechanical motion required to disengage the brake is taken from the ribbon lift mechanism. A vertical lug, which is part of the lift guide plate on the lift guide assembly projects up through a window in the feed plate (Fig. 187). Each time the ribbon lift guide assembly is raised the vertical lug on the lift guide plate operates the brake actuating lever which in turn disengages the brake from the supply spool. Since the feed operation does not occur until well after the lift operation is underway, the brake will always be disengaged from the supply spool at the beginning of a feed operation.

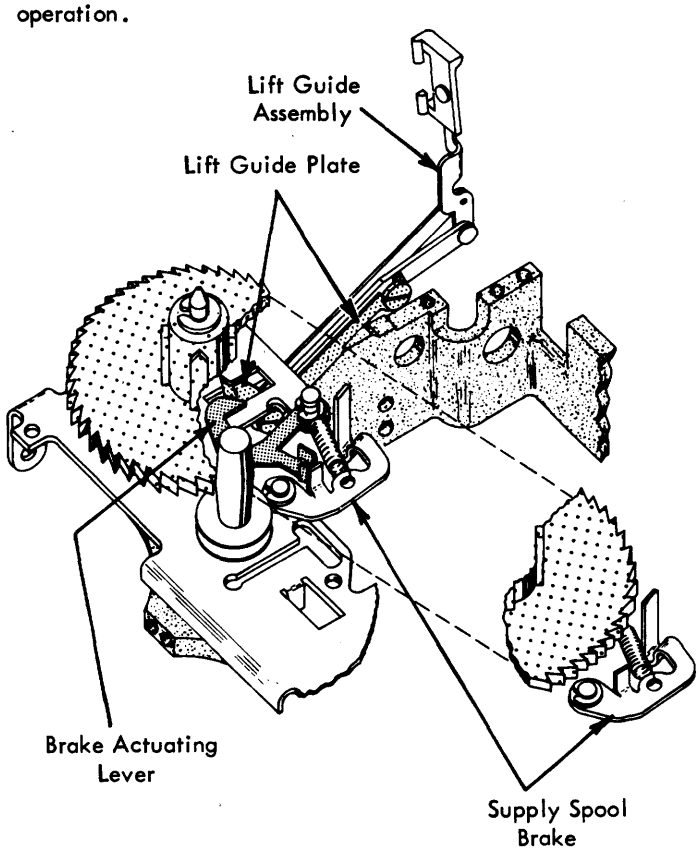


FIGURE 187. Brake Actuating Lever

Ribbon Lift

Because the entire ribbon mechanism is mounted on and moves with the carrier, the diameter of the supply spool is necessarily limited. To obtain the desired number of characters per spool of ribbon, a 9/16" wide ribbon is used. By using a wider ribbon and varying the ribbon lift position for each character (Fig. 188), a greater number of characters can be typed on a given length of ribbon.

The 3121 polyethylene film ribbon used on the Model "71" Selectric is a modification of the current 5121 ribbon used on the Model "C" typewriter. Character yield is approximately 52,000 characters per spool of 3121 film ribbon as compared to an average of 94,000 characters per spool of 5121 film ribbon on a Model "C" standard ET. This results in a consumption ratio of 1.8 spools of 3121 to each spool of 5121.

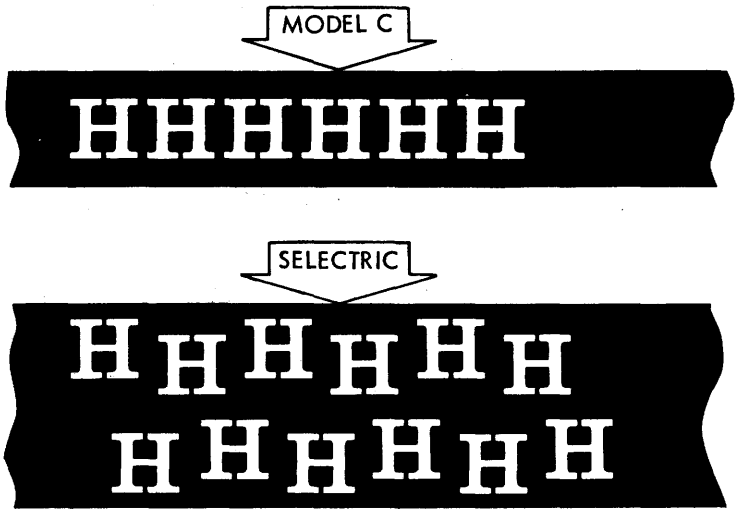


FIGURE 188. Lift Pattern

The lift mechanism for the film ribbon machines is similar to the lift mechanism on the fabric ribbon machines. That is, the lift motion is supplied by a cam on the print sleeve and the amount of lift that is produced is determined by the position of the control link in the slot of the cam follower (Fig. 189). The film ribbon lift mechanism differs only in that the lift position is automatically changed to one of four different lift positions during each print cycle.

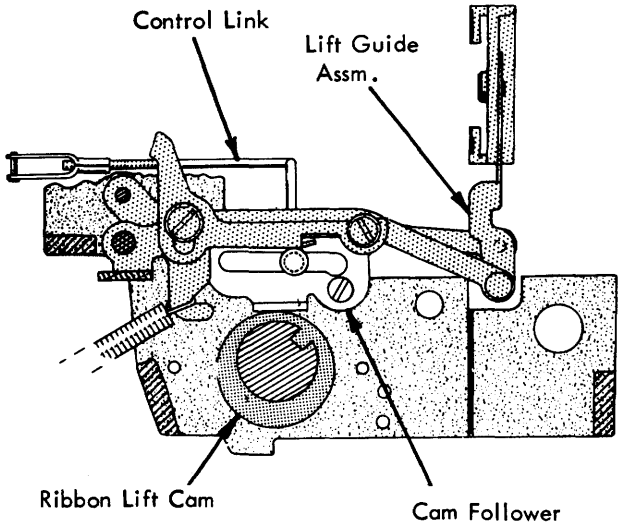


FIGURE 189. Ribbon Lift Mechanism

This is achieved by changing the location of the control link in the slot of the cam follower. The four lift positions obtained by moving the control link are designated by "A", "B" "C", and "D" (Fig. 190).

"A" and "C" are low lift positions while "B" and "D" are high lift positions. These four lift positions occur in a definite order during a typing operation. It takes four print operations to complate a lift cycle which is from "A" to "B" to "C" to "D" (Fig. 190). On the fifth print operation the lift cycle would begin all over again with lift position "A". Changing the location of the lift control link in the slot of the cam follower produces these lift positions.

The motion produced by the camming lobes on the feed and lift wheel is transmitted to the control link by the lift control lever (Fig. 192). The control lever is mounted to the front of the carrier by a shouldered screw and is spring loaded against the camming lobes of the feed and lift wheel. The steel roller "C" clipped to a pin on the control lever operates as a cam follower for the control lever.

The selected ribbon lift position for each print operation is always established by the ribbon feed operation from the previous print cycle. This is because the rotation of the feed and lift wheel (which produces the change in the ribbon lift position) does not occur until after the typehead has printed.

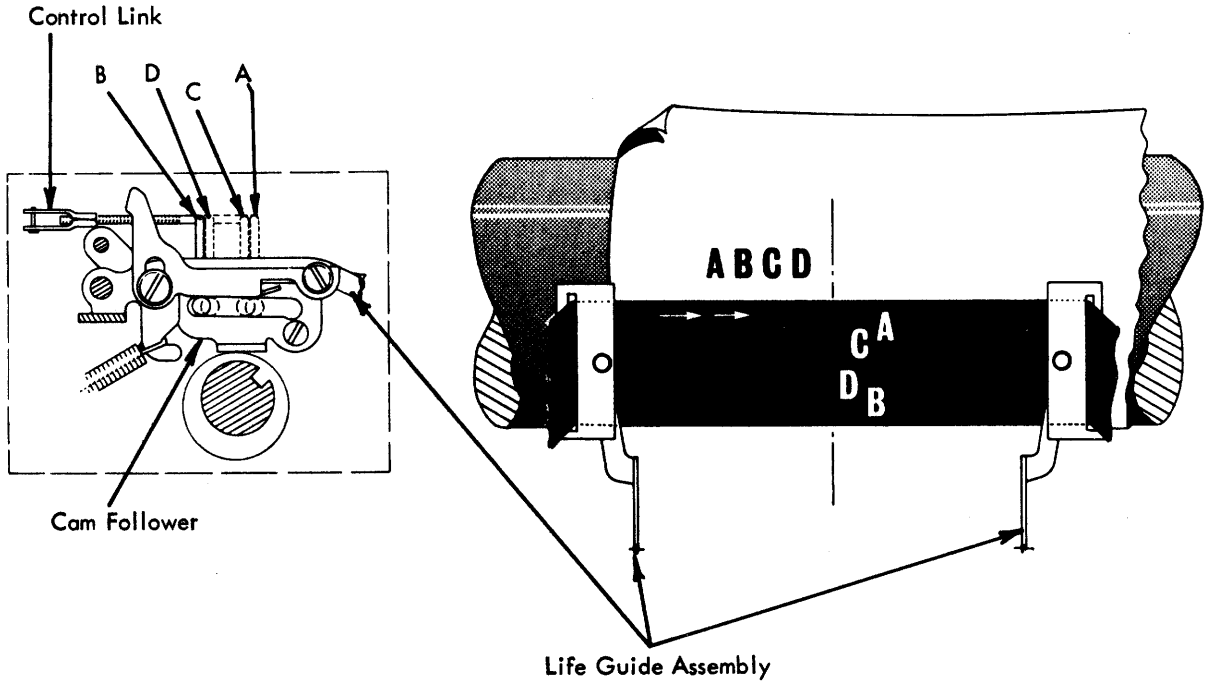


FIGURE 190. Ribbon Lift Positions

When the control link is positioned near the rear of the slot in the cam follower such as when it is in lift position "A", very little motion is obtained from the cam follower. Also, when the control link is in this position it is further away from the pivot point of the lift guide assembly than the other lift positions. Therefore, when the control link is in this position the least amount of motion is produced to the lift guide assembly and the character prints near the top of the ribbon.

Rotation of the feed and lift wheel during a ribbon feed operation produces the change in the lift operation. The camming lobes on the perimeter of the wheel governs the position of the control link during each ribbon feed operation (Fig. 191). These lobes which correspond to each feed window produce the four lift positions in a consecutive order as the wheel rotates 1/16 of a turn for each ribbon feed operation.

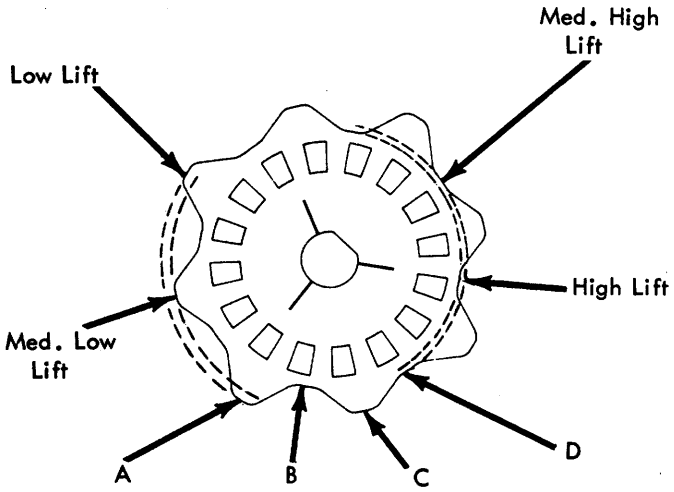


FIGURE 191. Feed & Lift Wheel

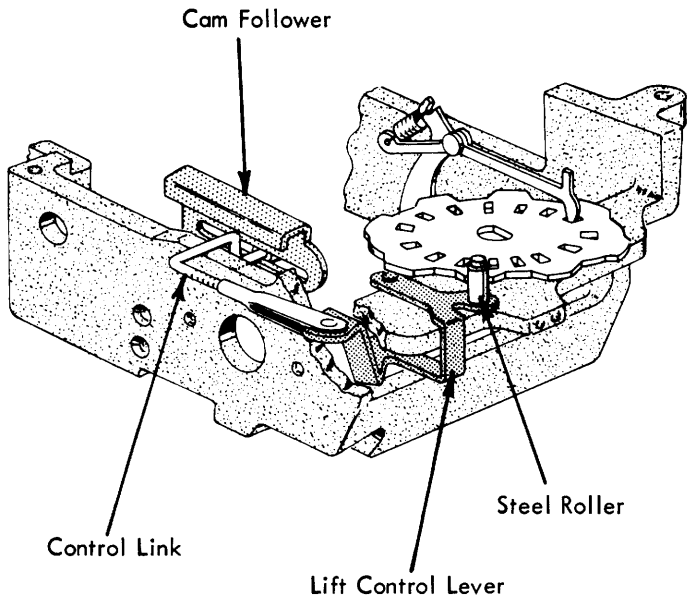


FIGURE 192. Lift Control Lever

Stencil Lockout

Whenever the machine is used for typing stencils, the ribbon feed and lift operation must be locked out. This is accomplished by a stencil lever located on the front of the carrier just above the carrier pointer. The stencil lever pivots about a shoulder on the feed and lift wheel mounting screw and is spring loaded into its rest position. When the lever is pushed to the left into its stencil position (Fig. 193) the feed and lift mechanism becomes inoperative.

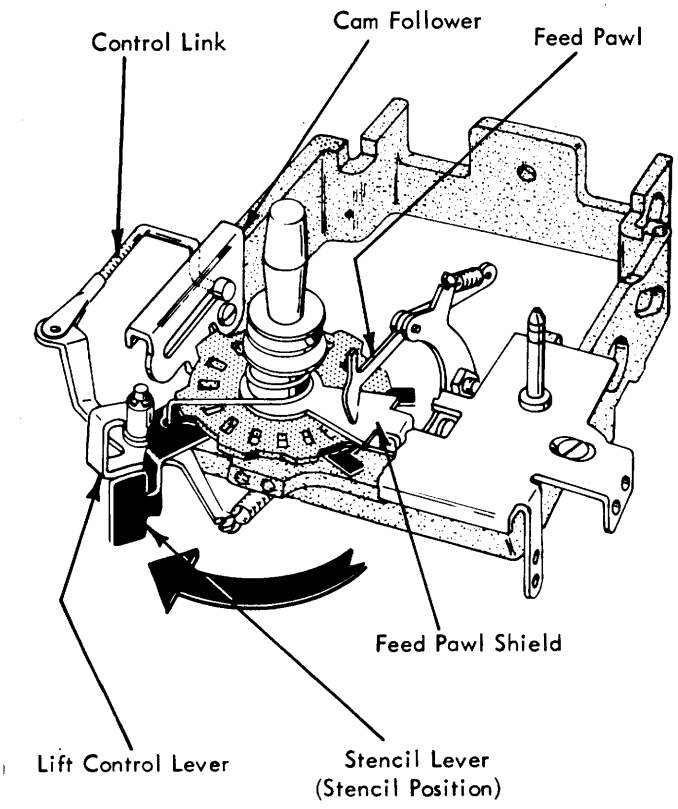


FIGURE 193. Stencil Lockout

The lockout of the feed mechanism is achieved through a feed pawl shield. Two lugs on the rear of the stencil lever position the shield over the feed windows in the area where the feed pawl operates during a ribbon feed operation. The feed mechanism becomes locked out because the shield prevents the pawl from dropping into the next feed window when the pawl is operated to the rear (as shown in Fig. 193) during a ribbon feed operation.

Lockout of the lift operation is achieved by a camming surface on the left side of the stencil lever. As the lever is pushed to the left into its stencil position, the steel roller on the lift control lever is cammed away from the feed and lift wheel (Fig. 193). This causes the control link to move to the rear of the slot in the cam follower where no lift motion will be produced to the ribbon lift guide assembly. The steel roller on the lift control lever detents the stencil lever in the stencil position even though a slight load is being applied to the lever in the restoring direction by the hairpin spring fastened to the front of the lever.

Ribbon Load Operation

When the operator desires to change the ribbon, she begins by pulling the load bail forward into its load position (Fig. 194). This causes the ribbon lift guide assembly to rise above the typing element so that the ribbon may be threaded through the guides with ease.

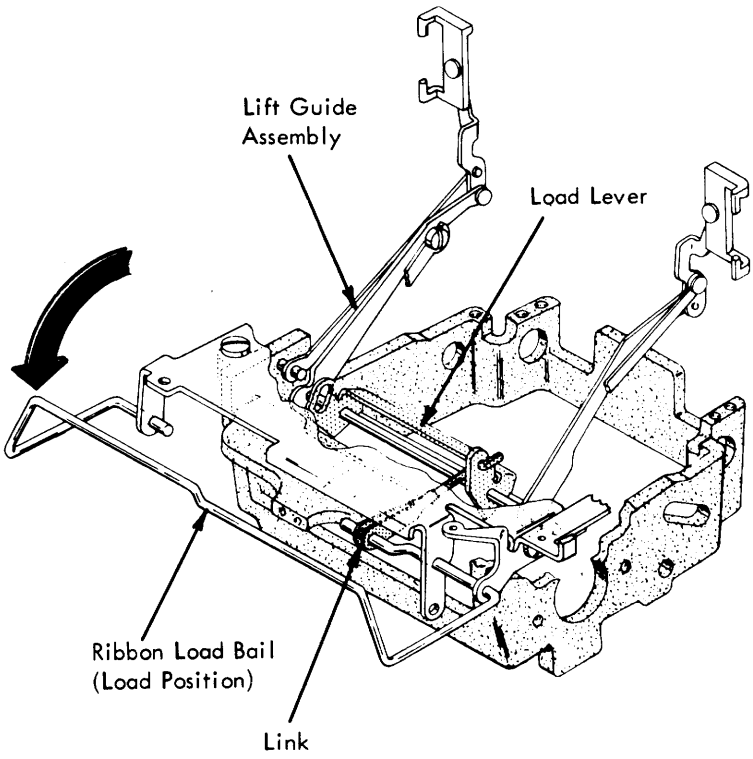


FIGURE 194. Ribbon Load Operation

Rotating the load bail forward into its load position creates a pull on the load lever link causing the load lever to rotate counterclockwise. The load lever then rotates the lift guide assembly into its elevated position. At the same time that the guide assembly rises, the supply spool brake is disengaged from the ratchet teeth on the supply spool. This action occurs in the same manner as it does during a normal ribbon lift operation (Fig. 187).

Pulling the load bail forward into its load position also causes the pressure roller and the supply drag lever to pivot out of the way as illustrated in Figure 195. A small arm called the pressure roller release arm fastens to the left end of the load lever. This arm extends up through a slot in the ribbon feed plate directly behind the pressure roller lever. Rotation of the load lever causes the release arm to push the pressure roller lever forward disengaging the pressure roller from the feed roller (Fig. 195). As the pressure roller swings forward, it forces the supply drag lever to pivot away from the supply spool. The operator may now install a ribbon with no obstructions.

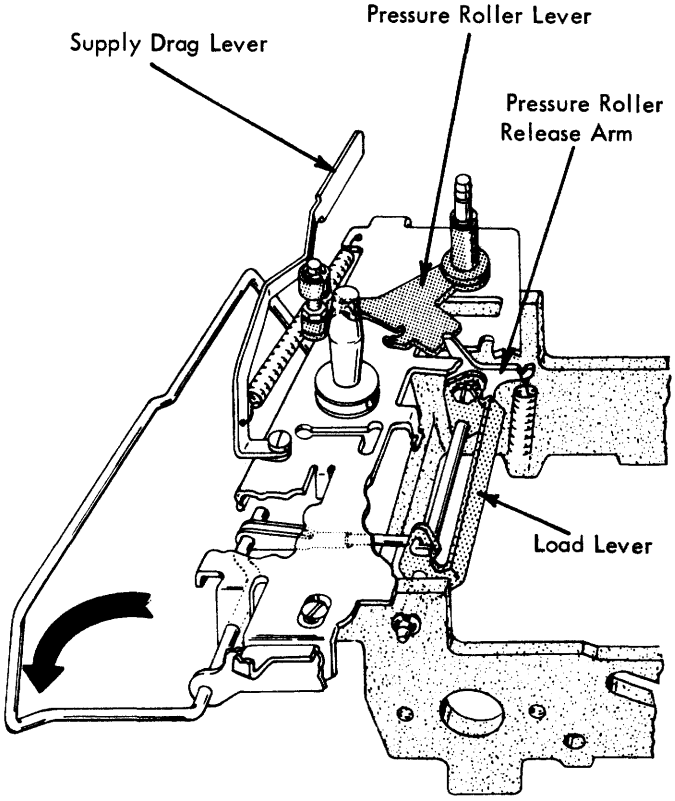
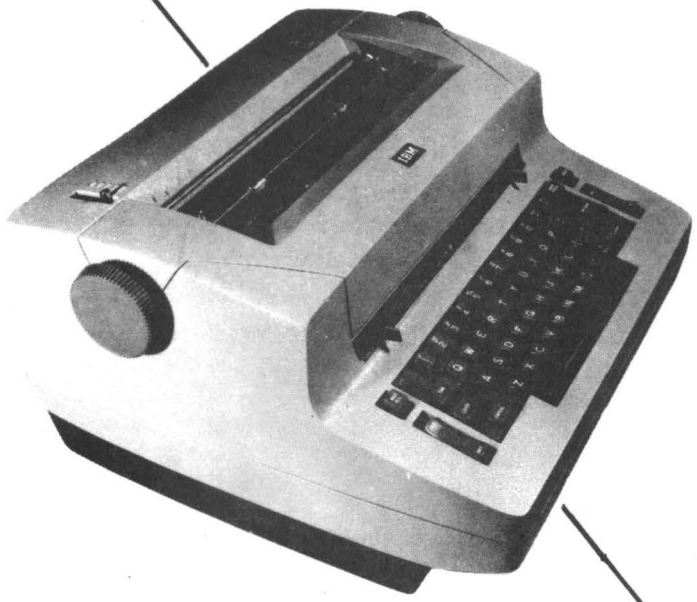


FIGURE 195. Pressure Roller Release

IBM®

ETD CUSTOMER ENGINEERING



Series 73 Selectric I/O Writer
INSTRUCTION MANUAL

Form/Part No: 241-5159-1

Printed in U.S.A.

Revised November 12, 1962

reprinted courtesy of IBM

Contents

INPUT - OUTPUT WRITER
73 SERIES

The basic machine is an IBM 72 Series typewriter adapted for input-output use, and called the 73 Series Input-Output Writer.

DESCRIPTION

The 73 Series typewriter can be used by an operator as a regular typewriter, or it can be used as an input device, an output device, or both.

It is a compact, light weight typewriter using a new principle of printing. A spherical type head moves along the writing line, while the carriage remains stationary.

MODELS

This machine is called the Model 731 Input-Output Writer. It is an 11" carriage machine with an 8 1/2" writing line. Two versions are available; Correspondence and BCD (binary coded decimal) machines. The correspondence machine has a correspondence type head and keyboard. This machine can be used by an operator as a correspondence typewriter. The BCD machine however, has a type head and keyboard arrangement that is designed primarily to be used in conjunction with a computer.

TYPE STYLE AND PITCH

1. Correspondence Machines -

   Any one of the six standard 721 typeheads may be used on a correspondence 731. Escapement is available in 10 or 12 pitch.

2. BCD Machines -

   The standard type style is Data #1 in a type head arrangement suitable for input-output use. 10 pitch escapement is standard.

RIBBON

The standard ribbon mechanism has three ribbon lift positions and a stencil position. The fabric ribbon and spool are contained in a snap-in cartridge to allow replacement without touching the ribbon.

A remote-controlled ribbon shift is available as an optional feature for special applications, to change ribbon color automatically.

COLORS

The standard case color is Sandstone Beige, however, any standard 72 Series case color is available. The keybuttons and platen knobs are the standard 72 Series gray.

KEYBOARD

The 73 Series Input-Output Writer uses two different keyboards. A correspondence 731 will have a standard 72 Series keyboard. The BCD 731 will use a BCD keyboard. This keyboard will contain special keylevers and buttons to meet the application and to match the type arrangement of the type head.

Non-repeat keys are being installed in all machines, because the primary use will be as an input-output writer. However, in installations where an operator will be using the machine to a greater extent as a typewriter, a Typamatic keylever may be desirable. In this case, they will be available upon customer request.

OPERATOR CONTROL

All keyboard and carriage controls (carrier return, tab set and clear, paper release, etc.) are located similar to the 72 Series typewriter. However, individual buttons may be changed to meet the application.

PRINCIPLES OF OPERATION

PRINT SELECTION UNIT

The print selection unit is used during input and output operations. When the 73 Series is being used as an output device, this unit selects the character to be printed. During input operations, which can be initiated manually or by remote control, contacts in the print selection unit transmit pulses that define the character being printed.

Print Magnet Assembly

Magnets are used in the print selection unit, and impulsing a magnet causes a character to be selected and printed.

Before the print magnet assembly is discussed in detail, some thought must be given to the selection of the selector latches.

The selector latch logic of the 73 Series is the same as the 72 Series typewriter. The selector latches always rest under the selector-latch bail. To select the proper character, it is necessary to move the unwanted selector latches away from the selector-latch bail. This leaves the correct selector-latch or latches to be operated by the latch bail.

In the 72 Series typewriter the selector latches are pulled forward by links. In the 73 Series typewriter they are pushed forward by pusher arms.

Magnet armatures in the print selection unit allow the pusher arms to move forward. For most selections, more than one latch will be pushed forward. This means more than one magnet will be energized.

An encoding network will select the correct print selection magnets and if needed, a code check magnet.

Code Check Magnet

The purpose of this magnet is to provide a means to determine if the proper number of magnets have been selected.

The name of the method used in the 73 Series typewriter to determine if the right number of magnets have been selected

is code check.

Code Check

In a machine where more than one magnet must be selected, it is possible, if a failure occurs, to pick up an extra magnet occasionally or it is possible to fail to pick one. By always selecting an odd number of magnets, any failure would be recognized immediately because an even number would result. This is called "odd bit" code check and is the type used in the Series 73 typewriter.

Code checking requires the addition of another magnet and selector latch. The selector latch has no mechanical function, other than to operate a contact. For example; to print a particular character, four magnets plus the code check magnet must be selected. If the decoding network selected the four magnets for the character selection and recognized the odd or check magnet was needed, the selection must have been correct. As another example; three magnets are needed to give the proper selection. This is now an odd number. Therefore, the code check magnet must not be selected. If a magnet is dropped or an extra one is selected, an even number will result and an error in selection will be indicated.

Although this does not indicate the proper character has been selected, it does insure that the proper number of magnets have been selected. Impulsing the code check magnet alone causes the full tilt and rotate character to print.

OPERATIONAL SELECTION UNIT

The functions that are controlled by the operational selection unit are carrier return, indexing, tabulation, backspace, and spacebar.

The mechanical operation of these functions are the same as the 72 Series typewriter except tabulation. Tabulation in the 73 Series typewriter is powered and will be discussed in the mechanical principles under the heading "Power Tab".

Operational Magnets

Impulsing one of the magnets in the selection unit will cause an interposer to be tripped without operating a keylever.

Operational Interposers

Each interposer serves two purposes in the input-output writer. One is to initiate the operation (carrier return, backspace, index, etc.) and the other is to allow a transmitting contact to transfer to indicate the specific function that is taking place.

Feedback Contacts

Contacts are operated by the operational unit to provide timed pulses to initiate and terminate magnet pulses.

KEYBOARD LOCK

The purpose of the keyboard lock is to prevent an operator from using the typewriter once it has been placed under remote control.

This is accomplished by a solenoid to lock all 44 character keys and the functional keys. The linkage can be assembled to lock the keyboard with the solenoid energized or de-energized. The application will determine which is used.

If the operator leaves the typewriter locked in upper-case, the keyboard lock will return it to lower case.

SHIFT

The Series 73 typewriter can be shifted automatically to upper or lower case. This is accomplished by impulsing a magnet. Once the typewriter has changed case, it will remain in that case until it receives a signal to change case.

A latch-type magnet assembly is used. Once the upper case magnet is energized, the magnet is mechanically latched in the energized position. This enables the typewriter to be "locked" in upper case without holding the magnet energized for a long period of time.

MECHANICAL PRINCIPLES

PRINT SELECTION UNIT

The print selection unit is mounted to the power frame in the lower left corner. It consists of three basic assemblies.

1. Selector magnet assembly
2. Selector latch pusher assembly
3. Selection contact assembly

Selector Magnets

There are seven magnets used in this unit, one for each tilt and rotate selection latch, plus one for code check (Fig. 1).

Trip Mechanism

Impulsing a magnet causes its armature to move to the rear. The operating end of the armature strikes the cycle clutch trip bail. As the bail is operated to the rear, it pulls a link that is connected to the latch lever (Fig. 2). The latch lever pivots about it's pivot point and the hook on the end of the latch lever disengages with the trip lever. The trip lever is spring loaded toward the cycle-clutch latch, so when the latch lever moves off the lug on the trip lever, the trip lever moves forward against the cycle clutch latch. The cycle-clutch latch restoring will restore the trip lever. At this time, the latching surface of the latch lever will restore behind the lug on the trip lever because of the restore spring on the latch lever and the mechanical restoration of the cycle-clutch trip bail.

Latch Pusher Cams

Two cams have been added to the cycle cam shaft assembly. They are located between the selector cams on the cycle shaft.

The purpose of the cams is to operate the selector-latch pusher bail.

As the cycle shaft begins to turn, the cam follower arms raise the latch pusher bail (Fig. 3). Further rotation of the cycle shaft causes the pusher bail to lower (Fig. 4). When the cycle clutch latches, the bail is restored to its rest position (Fig. 5).

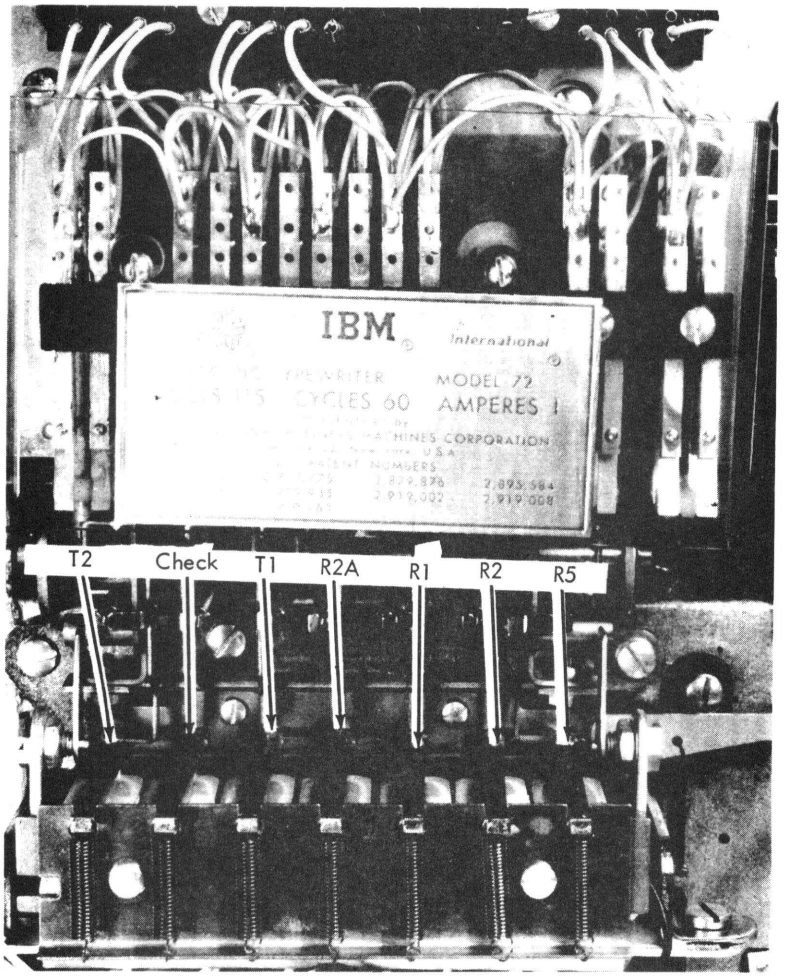


Figure 1. Selector Magnets

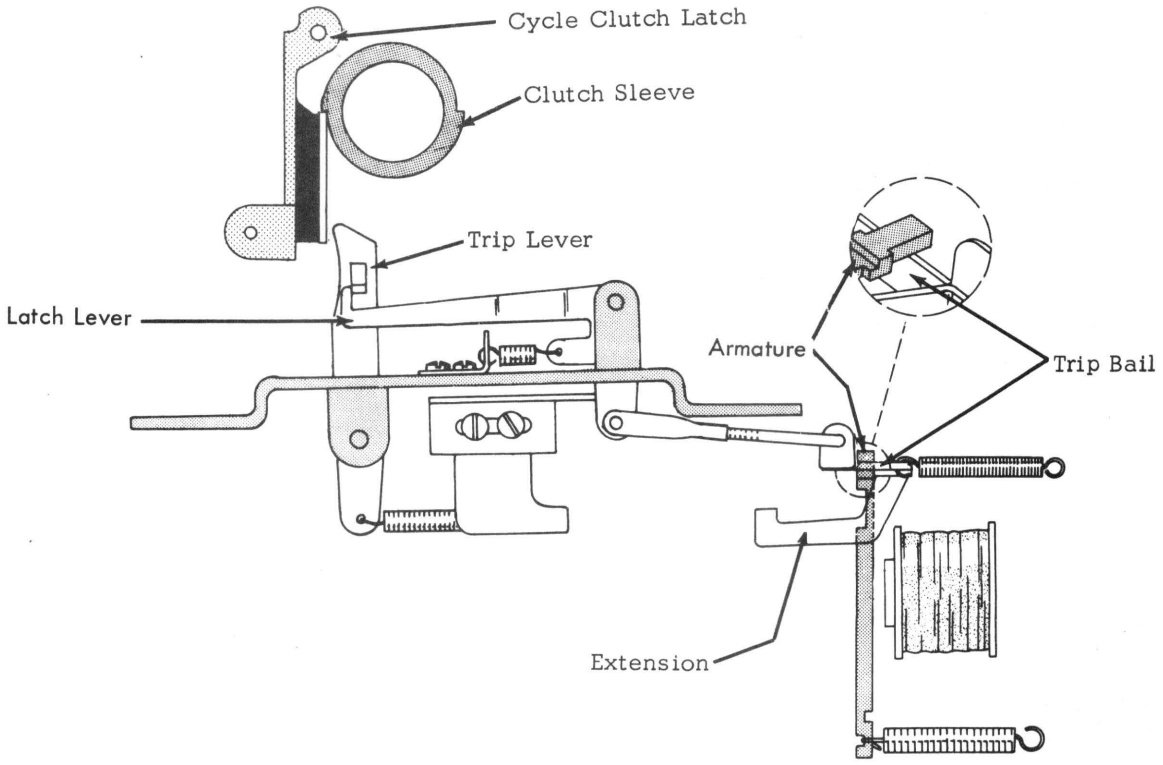
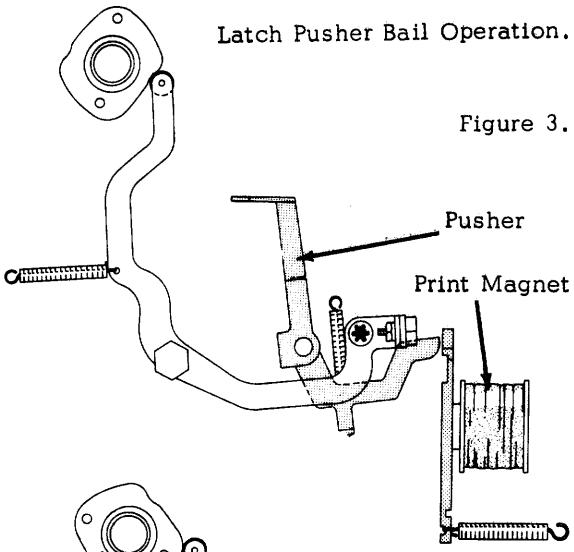


Figure 2. Cycle Clutch Trip Bail



Latch Pusher Bail Operation.

Figure 3.

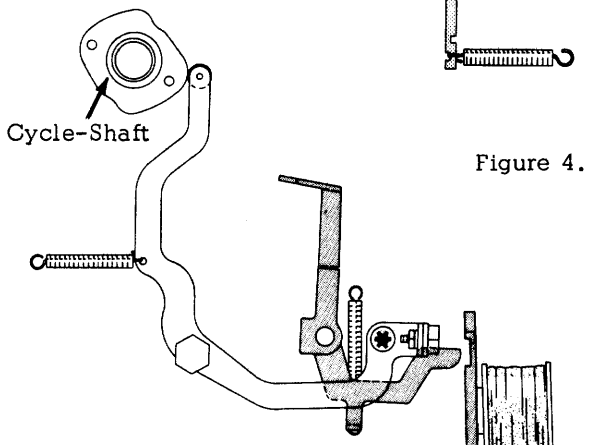


Figure 4.

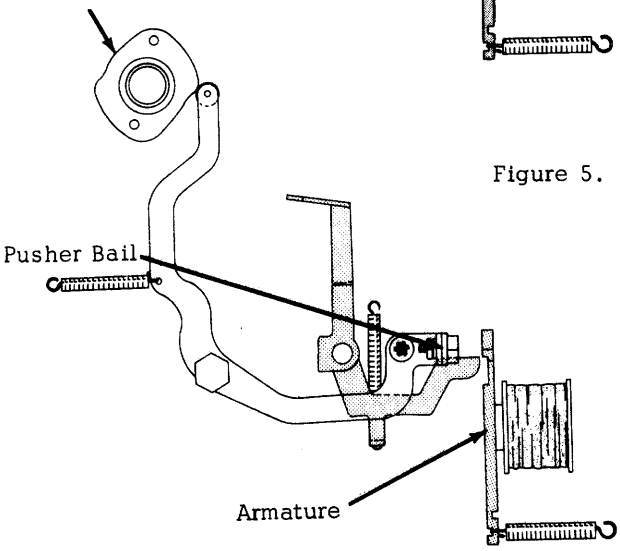


Figure 5.

Selector Latch Pusher

The selector latch pushers pivot on a shaft that is mounted to the power frame by a bracket.

There are seven latch pushers; one for each selector latch, including the code check latch.

The purpose of the latch pusher is to move the unwanted selector latches forward from under the selector-latch bail.

The latch pushers are spring loaded toward the pusher bail.

When the bail rises, the tail of the pusher follows the bail. As the pusher pivots on the shaft, the operating end or top of the pusher moves forward, contacting a selector latch. The latch will be pushed forward, away from the bail (Fig. 6).

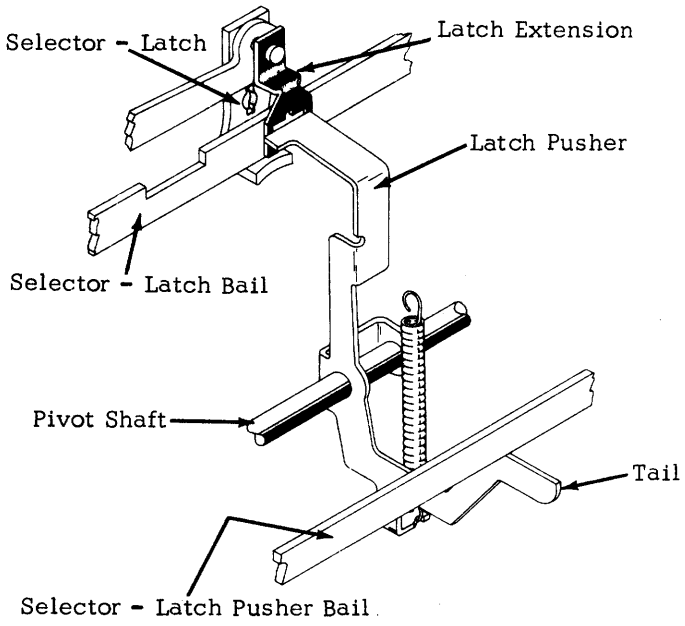


Figure 6. Selector - Latch Pusher

It is important to note that the selector armatures that were not attracted will keep their respective latches from being operated. This is accomplished by the tail of the pusher contacting the latching surface in the notch of the armature as the pusher starts to rise.

Because the armature kept the tail of the pusher from rising, the pusher could not push the latch away from the bail. Therefore, the selector latch is left under the selector-latch bail to be operated as the bail moves down.

Restoring the Selector Latch Pushers

The selector latch pushers are restored by the selector-latch pusher bail. The bail forces the tails of the pushers down, causing the pushers to pivot on a shaft. The top of the pusher moves to the rear, allowing the selector latches to return under the selector-latch bail.

The pusher bail must restore the pushers far enough to allow the armatures to fully restore. The tails of the pushers should be in the notch of the armatures. At rest, the tails of the pushers are against the pusher bail and should not be touching the armatures (Fig. 7).

Selector Armature Knock-off

Armature knock-off is provided to overcome any residual magnetism and to insure that the armatures restore quickly.

The selector-pusher bail provides a positive knock-off for the selection magnet armatures. An eccentric knock-off stud on the pusher-bail cam-follower arm restores the cycle clutch trip bail. The trip bail restoring imparts the knock-off to the selector armatures.

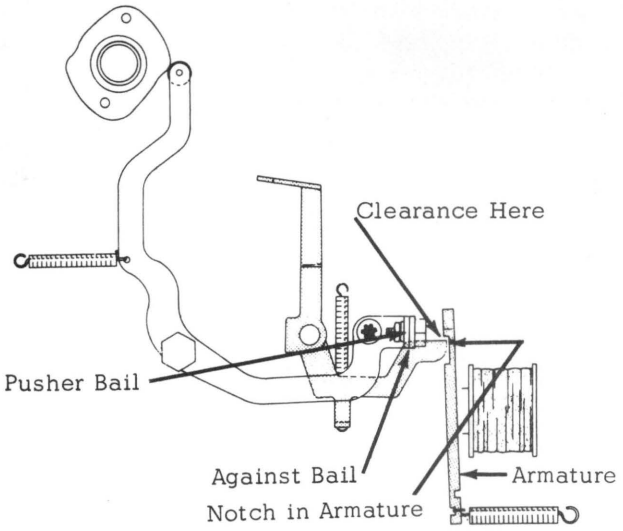


Figure 7. Latch Pusher Restoring

SELECTION CONTACT ASSEMBLY

The selection contact assembly is mounted to a bracket that is fastened to the power frame. It is located directly below the selector latches. The upper contacts are used for code check and the bottom contacts are the transmitting contacts (Fig. 8).

Contact Actuator

A contact actuator will be operated by the selector-latches that remain under the selector-latch bail during a selection.

The extension of the latch that is being carried down by the latch bail strikes the contact actuator. The actuator will be pushed down in its guide. The bottom of the actuator has two cross bars that operate the contacts (Fig. 9). When the selector-latch bail restores, the contact actuators rise under the spring tension of the contacts.

The five unit actuator is operated when the five unit latch is not pulled forward. Operating the -5 unit latch allows the five unit bail to rise. The contact actuator rises under the tension of the contacts (Fig. 9).

OPERATIONAL SELECTION UNIT

This unit is mounted to the power frame directly below the operational interposers and consists of a magnet unit and a contact assembly.

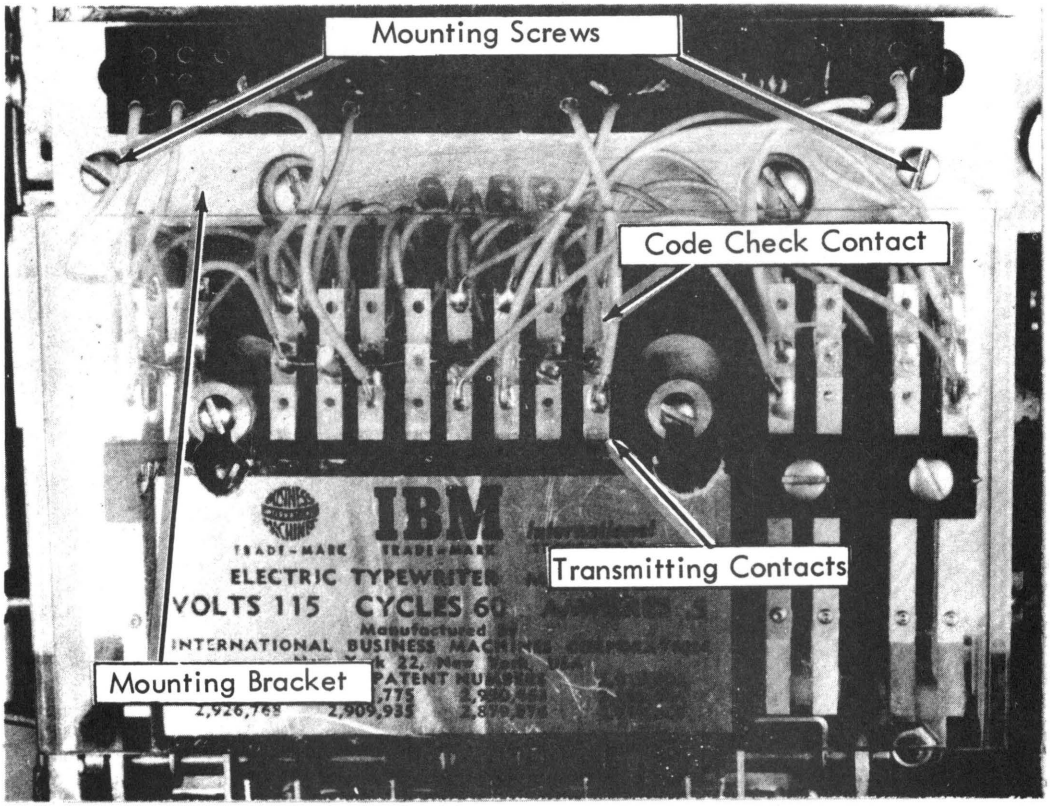


Figure 8. Print Selection Contact Asm.

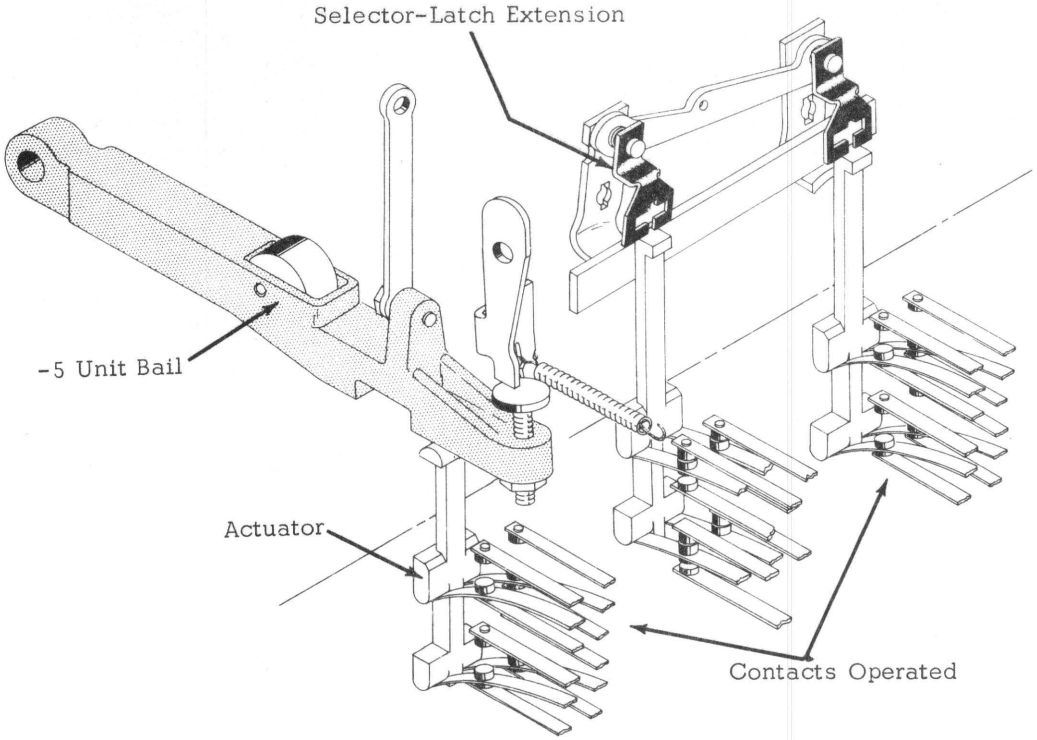


Figure 9. Contact Actuators

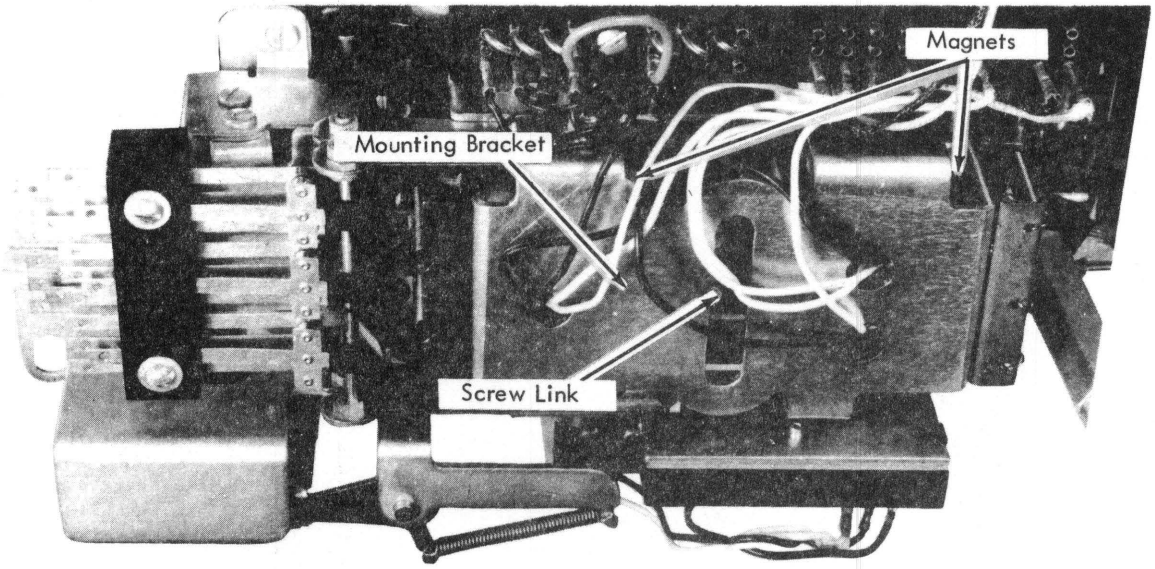


Figure 10. Operational Magnet Unit

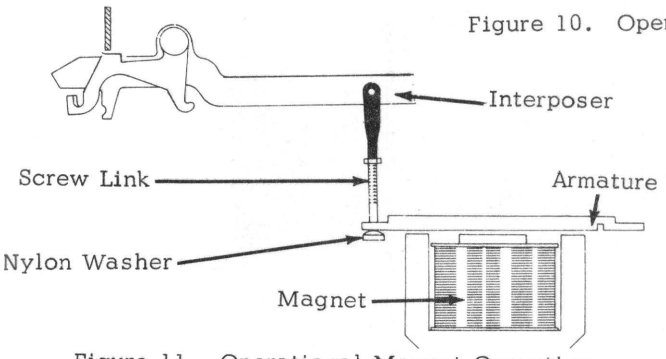


Figure 11. Operational Magnet Operation

Magnet Unit

The magnet unit contains five magnet assemblies, one for each operational interposer. The magnets are mounted to a bracket that is fastened to the power frame (Fig. 10).

When the armature is attracted, it moves away from the power frame pulling down on a link (Fig. 11).

This causes an interposer to be tripped without moving a key-lever. The slotted head on the lower end of the link provides

a means of adjusting the link without removing either end. The link extends through a hole in the armature, and the upper end is connected to the interposer by a clevis. A crowned nylon washer serves as a bearing point between the armature and the head on the link. This allows the top of the link to move front to rear with the interposer without changing the effective length of the link. Once the interposer is tripped, the mechanical operation of each function except tabulation, is the same as in the 72 Series typewriter. Tabulation will be discussed later in this text under "Power Tab".

Contact Assembly

This assembly is comprised of five sets of contacts, each having its own latch.

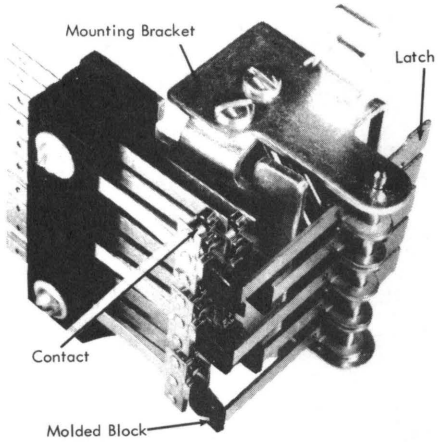


Figure 12. Operational Contact Assembly

The operating strap of each contact set has three contact points that make with three separate contacts (Fig. 13). The contacts are closed contacts held open by a bail. A block is molded on the end of the operating straps to provide an insulated operating surface for the bail and a latching surface for the contact latches.

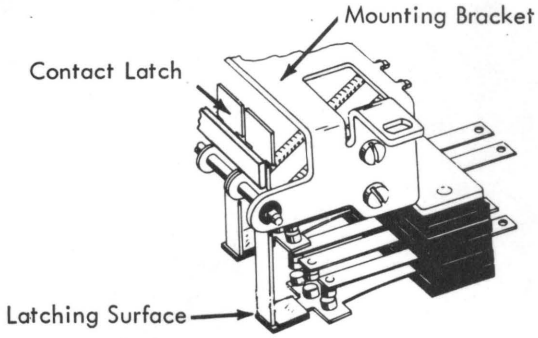


Figure 13. Operational Contact Assembly

Actuating Arms

There are two contact actuating arms. One is operated by the carrier return-index check pawl and called the right hand contact actuating arm. The other is operated by the tab-spacebar-backspace check pawl and called the left hand actuating arm.

The arms are mounted to the contact assembly by a shouldered screw. The shouldered screw provides a pivot point for the arms. The top of the arm has a pin riveted to it, and the lower end has two eccentric pins that serve as contact bails (Fig. 14).

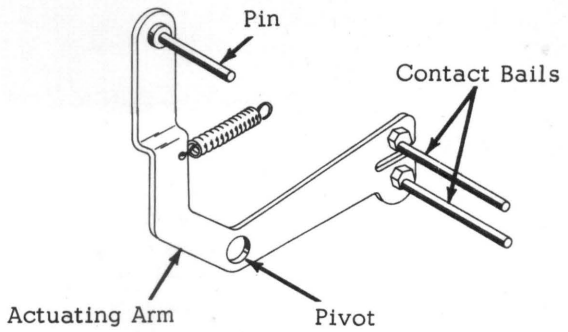


Figure 14. Actuating Arm

The pin on the upper end of the arm rides underneath the operational check pawl. As the cam rotates, the check pawl rides out of the notch in the check ring (Fig. 15). This causes the check pawl to move down, pushing the top of the actuating arm down. The actuating arm pivots on the shouldered screw, raising the lower end. As the bail rises, the contact that has its latch operated, will close. The other contacts will be held inoperative by their latches (Fig. 16).

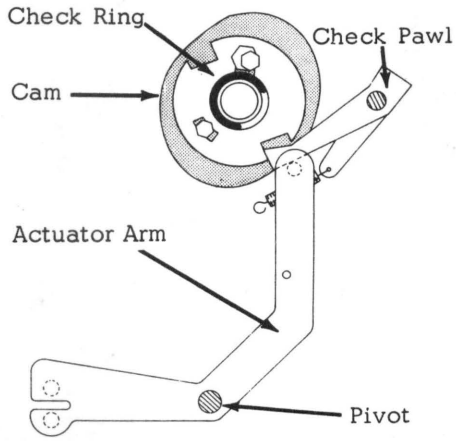


Figure 15.
Actuating Arm Operated By Check Pawl

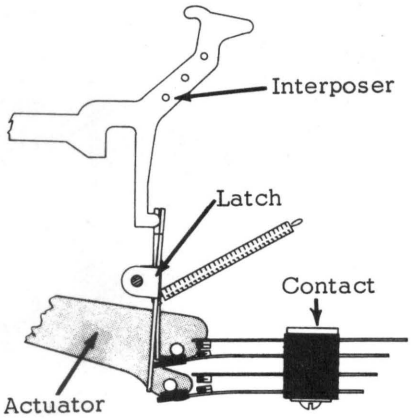


Figure 16.
Contact Bail and Latch Operation

When the check pawl returns into the notch in the check ring, the actuating arm will restore under its own spring tension. As the top of the actuating arm moves up, the lower end moves down. The contact bails lowering will restore the contact. With the actuating arm restored, the contact bail keeps the

contacts from resting against their latches. The purpose of this is to allow the latches to be moved away from the contact by the interposer without any drag.

Contact Latches

The latches rotate on a shaft that is mounted to the contact assembly by a bracket. The interposer moving to the rear, pushes the top of the latch to the rear. This is accomplished by an arm on the interposer that extends down in front of the latch. As the top of the latch is pushed to the rear, the latch pivots on the shaft causing the lower end of the latch to move to the front. As the interposer restores, the latch restores by its spring tension (Fig. 17).

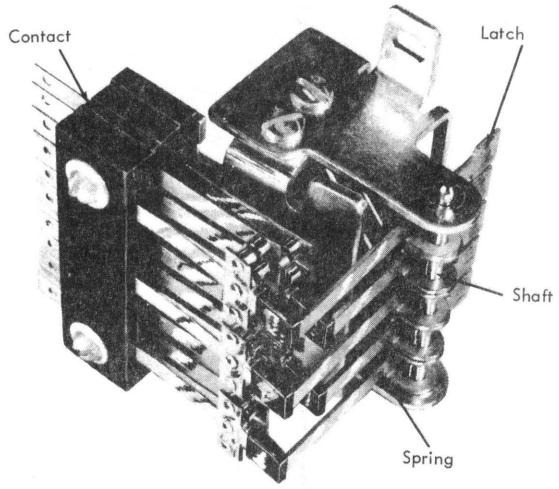


Figure 17. Contact Latch Operation

Feedback Contact

The feedback contacts are located in the right rear corner on top of the power frame. The contact on the left is the carrier return-index contact and the one on the right is the tab-space-bar-backspace contact (Fig. 18).

The carrier return-index contacts are operated by a tab on the cam follower lever. The tab-spacebar-backspace contacts are operated by an auxiliary cam follower lever.

As the cam follower lever is operated, the bail moves down, closing the contacts. The cam follower lever restoring allows the contact to open (Fig.19).

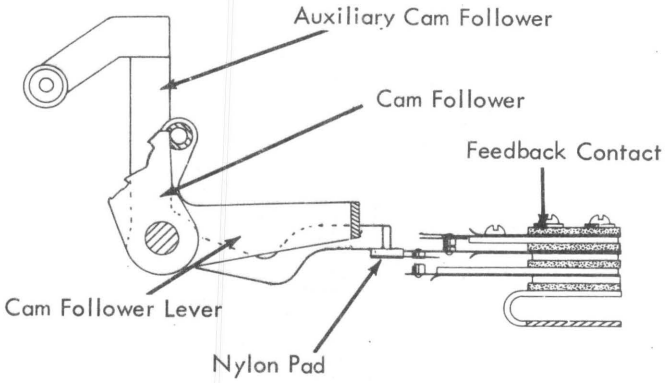


Figure 19. Feedback Contact Operation

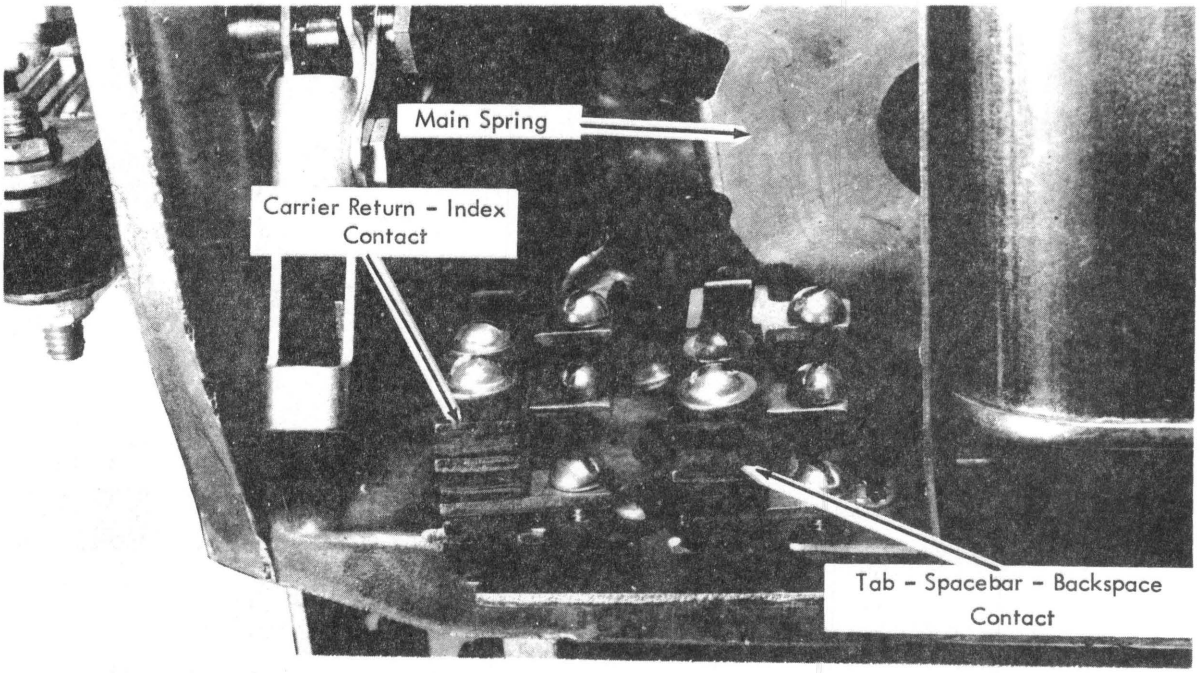


Figure 18. Operational Feedback Contacts

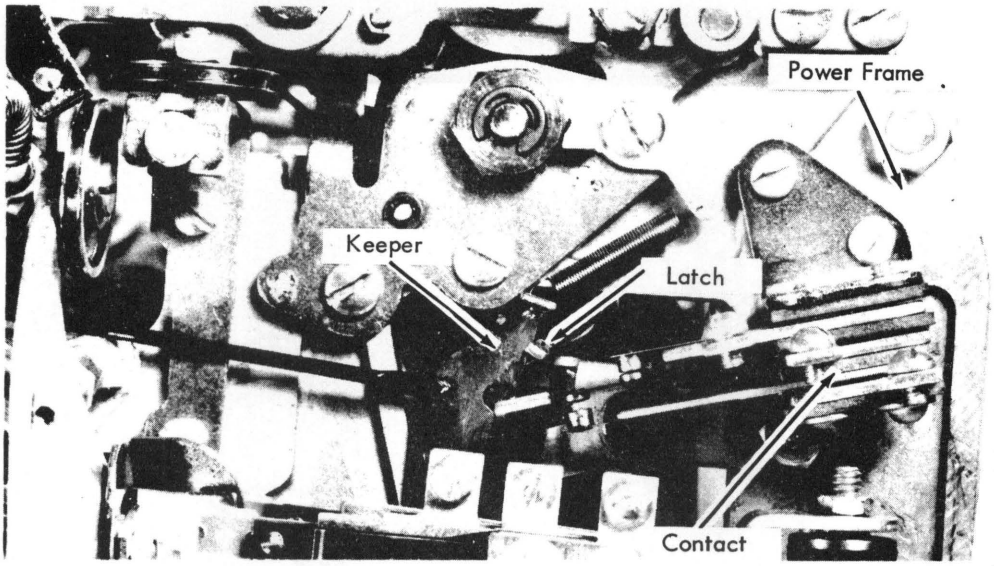


Figure 20. Carrier Return Interlock Contact

Carrier Return Interlock Contact

The carrier return interlock contact is mounted to the right side of the power frame in the upper corner (Fig. 20).

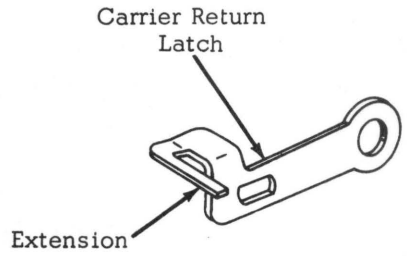


Figure 21. Carrier Return Clutch Latch

The interlock contact is operated by the carrier return clutch latch. An extension has been added to the latch for this purpose (Fig. 21).

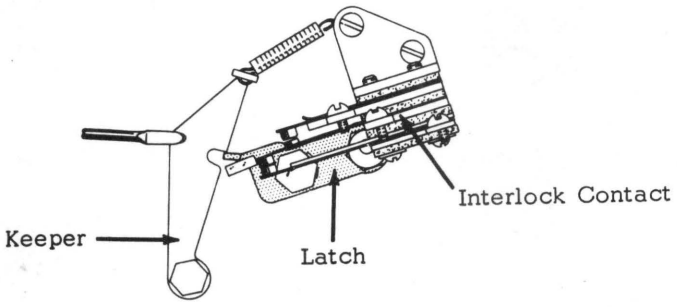


Figure 22. Carrier Return Interlock - Latched

With the latch in its rest position, the contact is in its normal position. As the latch is operated, the extension moves down, transferring the contact. In the latched position, the extension holds the contact transferred (Fig. 22). When the carrier return operation is completed, the latch is unlatched and the contact returns to normal (Fig. 23).

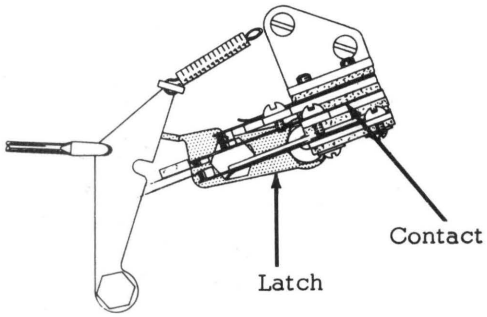


Figure 23. Carrier Return Interlock - Unlatched

POWER TABULATION

Tabulation in the 73 Series typewriter is basically the same as in the 72 Series typewriter. The major difference is the way the tab is initiated.

Keylever Assembly

The tab keylever operates a bail that extends from left to right above the shift bail (Fig. 24). The bail operates a keylever on the right side of the keyboard. This keylever does not have a stem for a keybutton and is used only to trip the tab interposer.

Power Tab Interposer

The tab interposer serves the same purpose as the other functional interposers. That is to trip the cam, to push the tab latch to the rear, and to operate the contact latch.

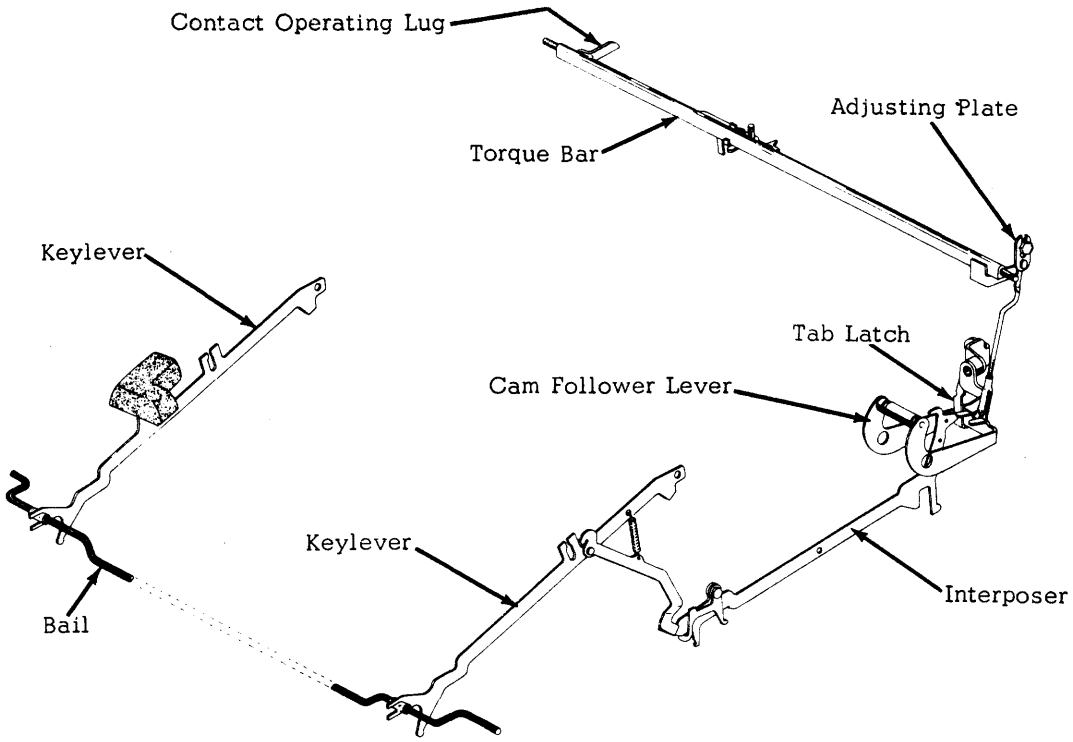


Figure 24. Power Tab Mechanism

Operational Cam

The power is furnished by the same cam that operates the backspace and the spacebar mechanisms.

Power Tab Operational Latch

Provision is made in the 72 Series machines for this latch. However, the latch is installed on input-output machines only.

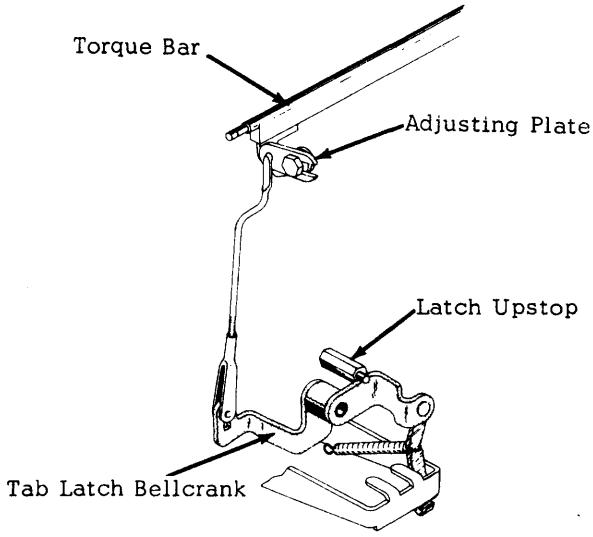


Figure 25. Power Tab Latch

The latch is connected to a bellcrank and, as the latch is carried down by the cam follower lever, the bellcrank is rotated clockwise (Fig. 25).

Power Tab Torque Bar

A lug has been fastened to the right end of the torque bar (Fig. 26). A link connects the torque bar to the latch bellcrank. Rotating the bellcrank raises the link. The link rotates the bottom of the torque bar to the rear.

Lockout Tab

The tab trigger has been eliminated on the 73 Series. The tab lever is pushed to the rear by the torque bar on the 73 Series. A torque bar lockout tab on the tab lever holds the torque bar to the rear as long as the tab lever is latched out. (Fig. 27).

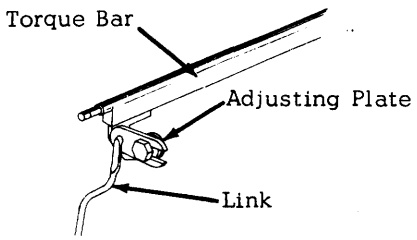


Figure 26. Torque Bar Operating Lug

Tab Interlock Contact

The tab interlock contact is a micro-switch. It is mounted to the right side of the power frame by a bracket (Fig. 28).

The tab on the left end of the torque bar has been altered to allow it to operate the interlock contact (Fig. 29).

With the torque bar latched out, the tab raises, allowing the contact to operate under its own spring tension. When the torque bar restores, the tab lowers opening the contact (Fig. 30).

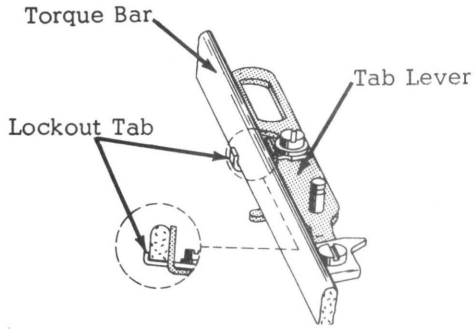


Figure 27. Torque Bar Lock-out Tab

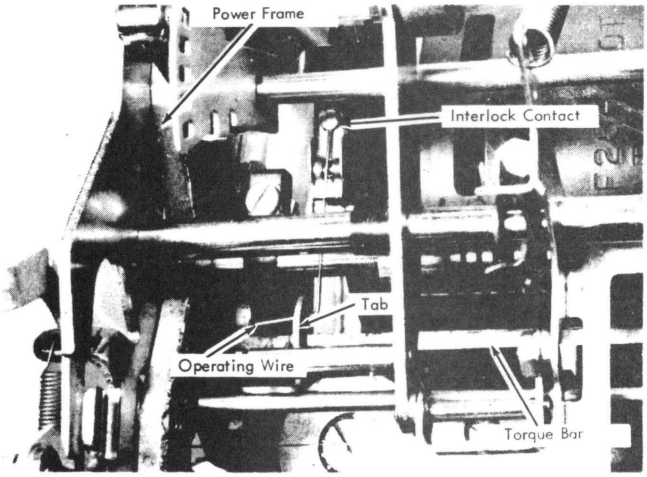


Figure 28. Tab Interlock Contact

KEYBOARD LOCK

Keyboard Lock Solenoid

The keyboard lock solenoid is mounted to the bottom side of

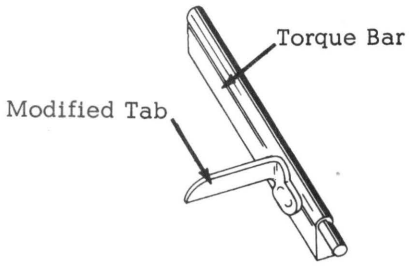


Figure 29. Torque Bar - Left End

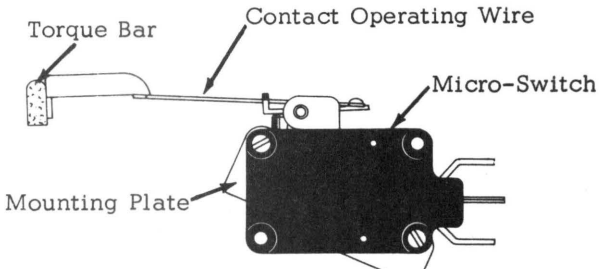


Figure 30. Tab Interlock Contact

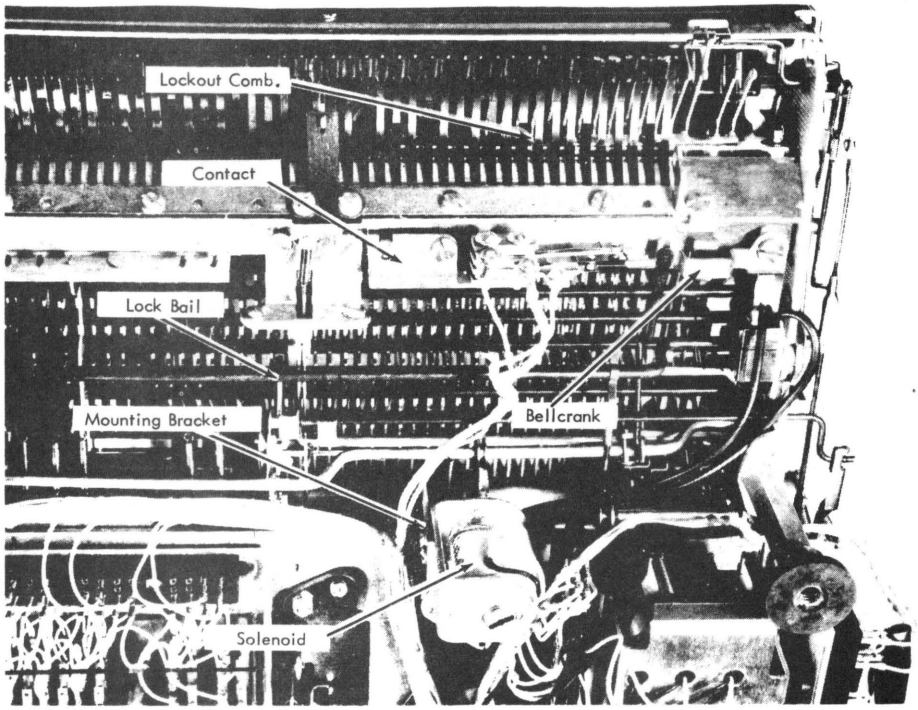


Figure 31. Keyboard Lock Mechanism

the power frame. The solenoid is held by a U-shaped bracket that is bolted to the frame (Fig. 31).

Lockout Lever

A bracket holds the lock-out lever in place. The bracket is fastened to the top of the power frame. The lock-out lever is fastened to the bracket by a shoulder pivot screw.

The lock-out lever is positioned front to rear. The front of the lock-out lever contacts the keyboard lock bail roller. A spring is fastened to the rear of the lever, pulling it down. This causes the front of the lever to move up anytime the solenoid is not energized.

The sloped surface of the lock-out lever contacts the bail roller. As the lever moves up or down, depending on which way it is installed, the roller will be cammed forward. Figure 32 shows the lever installed to lock the keyboard with the solenoid energized. Figure 33 shows the lever installed to lock the keyboard with the solenoid de-energized.

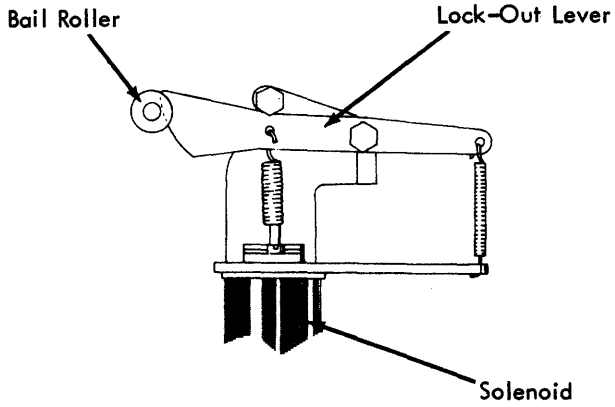


Figure 32. Lockout Lever - Solenoid Energized

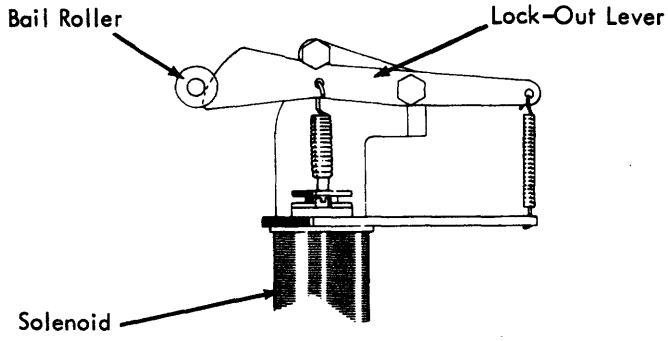


Figure 33. Lockout Lever - Solenoid De-energized

Keyboard Lock Bail

A bail is installed in the 73 Series typewriters to provide a means of locking the various functions. It is parallel to the line lock bail and is located to the rear of the line lock bail. The bail is operated by the bail roller, and as the bail is rotated to the front of the typewriter, three links fastened to the bail cause the following: (Fig. 34)

1. The functional keylevers are locked.
2. All 44 character keys are locked.
3. The machine will be returned to lower case if it is locked in upper case.
4. The keyboard lock contact is operated.

Functional Keylevers

The functional keylevers are locked by the link on the left end of the bail. The link is fastened to the bail by an adjustable clamp. The end of the link that extends to the front, fastens to a bellcrank. The bellcrank pivots around the screw that mounts it to the keylever support. As the link moves

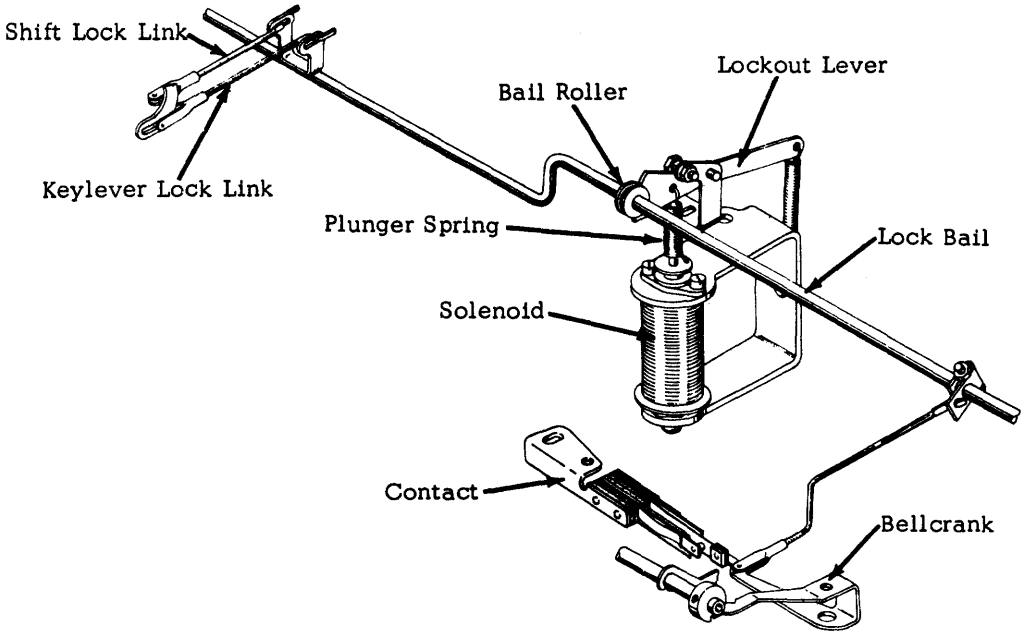


Figure 34. Keyboard Lock Mechanism

toward the front, the bellcrank is rotated in a clockwise direction. The arm on the bellcrank, that extends to the front, moves to the right. The bellcrank moves the lockout comb to the right.

The lugs on the comb move under their keylevers to prevent them from being depressed. The comb is spring loaded to the left so it will restore by its own spring tension when the key lock is removed.

Spacebar

Because the spacebar does not operate a key lever and cannot be locked with the comb, a lock collar is fastened to the spacebar shaft. When the keyboard lock bellcrank rotates, the arm on the bellcrank moves to the right under the step on the spacebar lock collar. This prevents the spacebar from being depressed (Fig. 35).

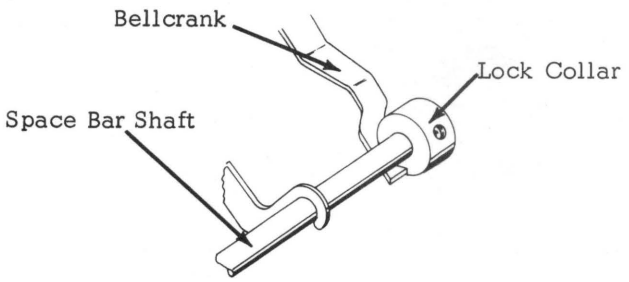


Figure 35. Spacebar Lock Collar

Keyboard Lock Contact Assembly

A contact is transferred when the keyboard lock is operated. This is accomplished by a tab on the bellcrank that is operated by the keyboard lock bail.

44 Character Keylever Lock

A bracket on the left end of the keylever lock bail operates the keylever lock. The keyboard lock bail pushes the link forward. The link operates the keylock interposer which locks the keyboard the same way as in the 72 Series typewriter.

Shift Lock Release

A bracket on the left end of the keyboard lock bail operates the shift lock latch. When the bail moves to the front, the shift lock link moves forward. The link operates the shift lock interposer.

The shift lock interposer moves front to rear. It is supported at both ends by a guide comb. As the link pushes the interposer to the front, the interposer contacts a stud on the lock. The lock is pushed away from the shift stop, releasing the shift key. The machine will return to lower case.

SHIFT MECHANISM

Shift Magnets Assembly

The shift magnet assembly is mounted to the right side of the power frame below the shift cam. It consists of two magnets and their armatures. (Fig. 36).

The upper case armature has a formed end that extends to the rear and serves as a latching surface (Fig. 37). The lower case magnet armature is spring loaded to the front against the formed end of the upper case armature (Fig. 38).

When the upper case magnet is energized, the formed end of the armature moves down. The lower case armature moves to the front, to its de-energized position. In this position, the lower case armature mechanically latches the upper case armature in the energized position, by moving forward into the path of the upper case armature's formed end (Fig. 39).

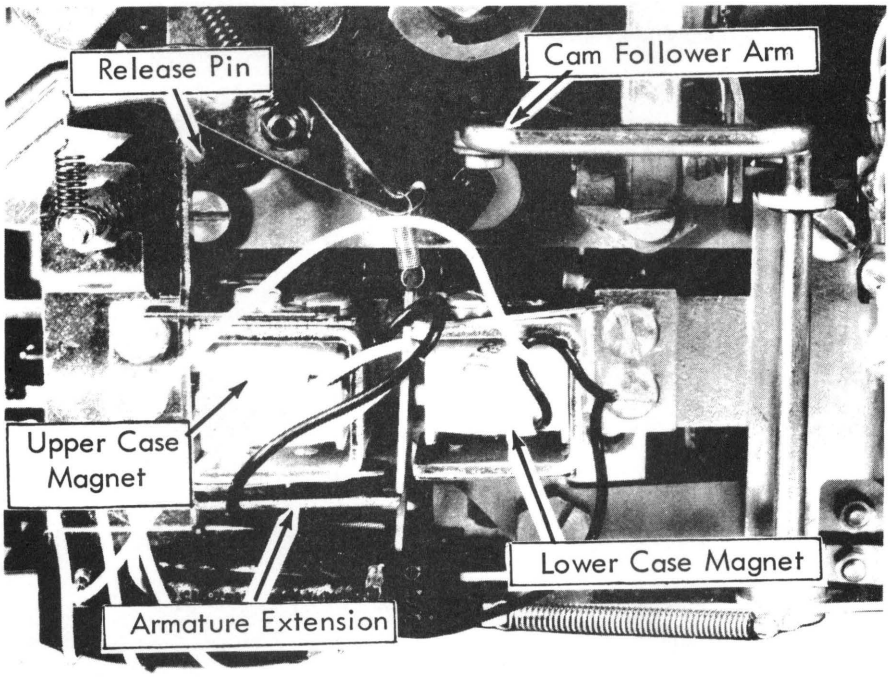


Figure 36. Shift Mechanism

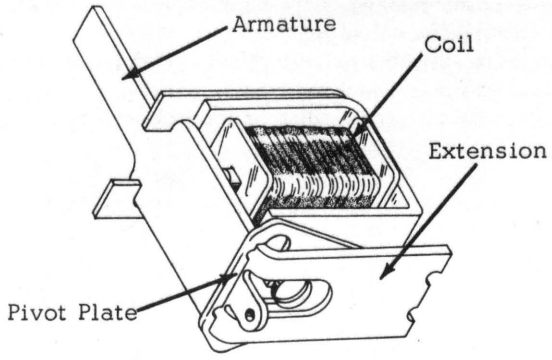


Figure 37. Upper Case Magnet

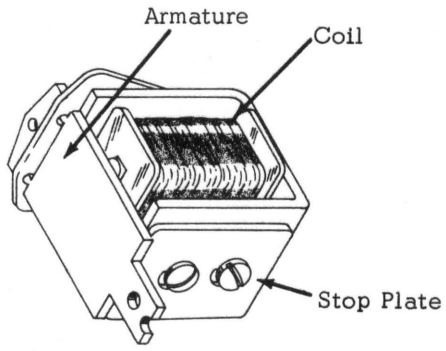


Figure 38. Lower Case Magnet

Attracting the lower case armature to the rear unlatches the upper case armature. The lower case armature moving to the rear, moves off the latching surface of the formed end of the upper case armature. This allows the formed end to snap up and the operating end to move forward.

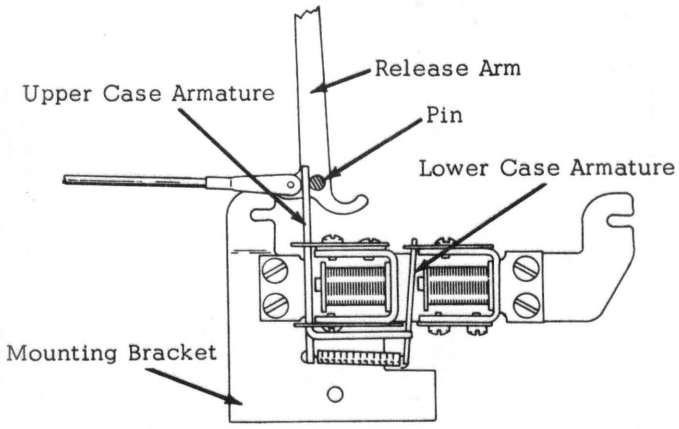


Figure 39. Shift Magnet Assembly

Shift Release Arm

The upper case magnet armature moving to the rear contacts a stud on the shift release arm pushing the arm to the rear.

The mechanics of the shift mechanism are the same as the 72 Series typewriter. Therefore, once the shift release arm is pushed to the rear, the shift takes place as normal.

Shift Contact Assembly

The shift contact assembly consists of a cam follower and two contact assemblies mounted to a plate. The plate is mounted to the right side of the power frame (Fig. 40).

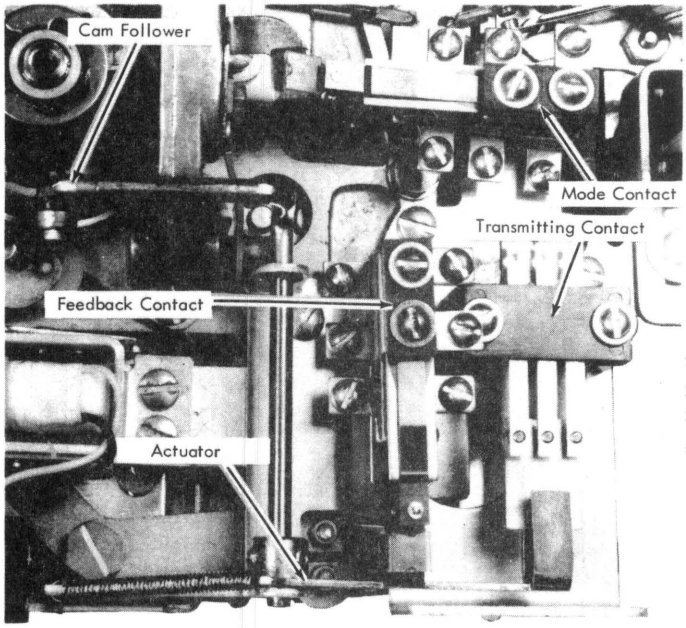


Figure 40. Shift Contact Assembly

Contact Cam Follower Actuating Arm

The cam follower arm rides the periphery of the shift cam. It is spring loaded toward and follows the contour of the cam.

The contact cam follower arm is positioned on the cam to lead the shift arm by 90°. In upper case, the shift cam is detented so the shift arm rests on the high point of the shift cam (Fig. 41). In lower case, the cam is detented so the shift arm rests on the low point (Fig. 42). Since the cam follower arm leads the shift arm by 90°, it will rest midway between the high and low point of the cam in either detent position. The contour of the

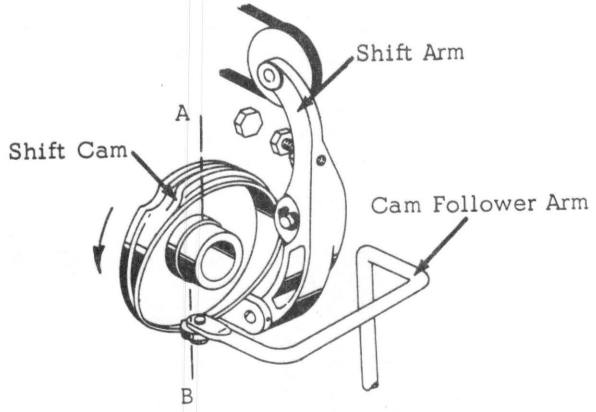


Figure 41. Shift Cam Detented Upper Case

cam is such that Point A and B (Figs. 41 & 42) are in the same plane. Therefore, the cam follower arm will be in the same relative position when the shift cam is detented in upper or lower case.

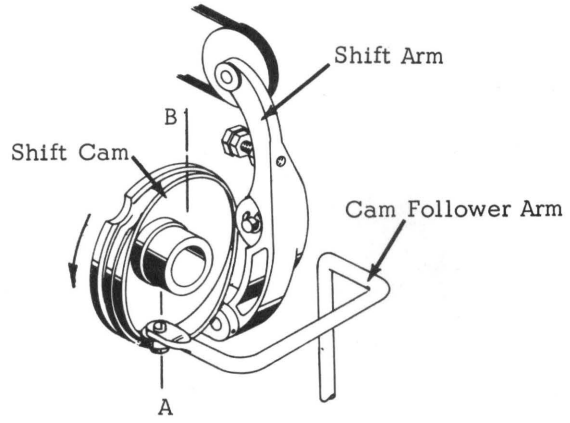


Figure 42.
Shift Cam Detented Lower Case

Assume a shift operation from lower to upper case. The shift cam rotates 180° and will detent with the shift arm on the high point. As the high point moves past the cam follower arm, the arm is moved to the right and back to its rest position.

Returning to lower case, the shift cam rotates 180° and will detent with the shift arm resting in the low dwell. As the shift cam rotates, the cam follower arm moves to the left into the low dwell and back to its rest position.

Contact Actuating Arm

The contact actuating arm is set-screwed to the lower end of the cam follower arm.

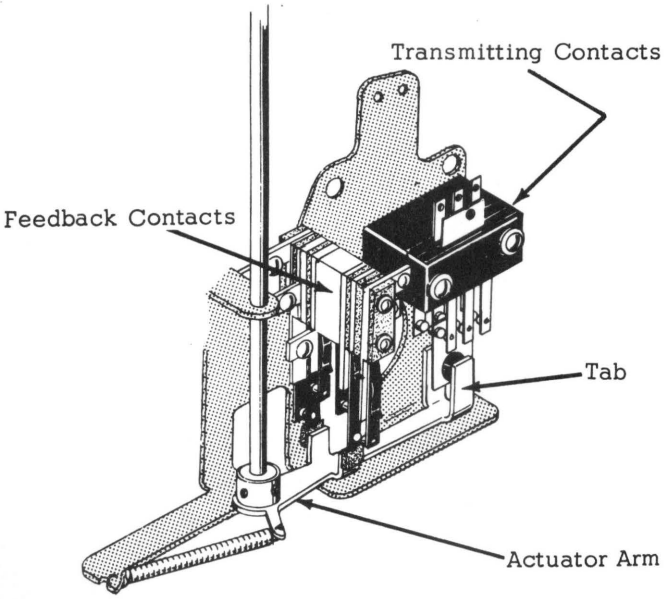


Figure 43. Shift Contact Assembly

The follower arm moving to the right, pivots about its fulcrum point, rotating the actuating arm to the left. As the follower arm returns to rest, the actuating arm returns to the center position. As the follower arm moves in and out of the low dwell, the actuating arm moves to the right and back to its center or rest position.

NOTE; the actuating arm operates only during the movement of the shift cam.

Shift Contact Assemblies

The contact assemblies operated by the actuating arm are feedback and transmitting contacts.

The assemblies nearest the front are the feedback contacrs. The assembly to the rear is the transmitting contact (Fig. 43).

Shifting to upper case, the inner set of feedback and transmitting contacts are operated by the actuating arm. Shifting to lower case, the outer set of feedback and transmitting contacts are operated by the actuating arm.

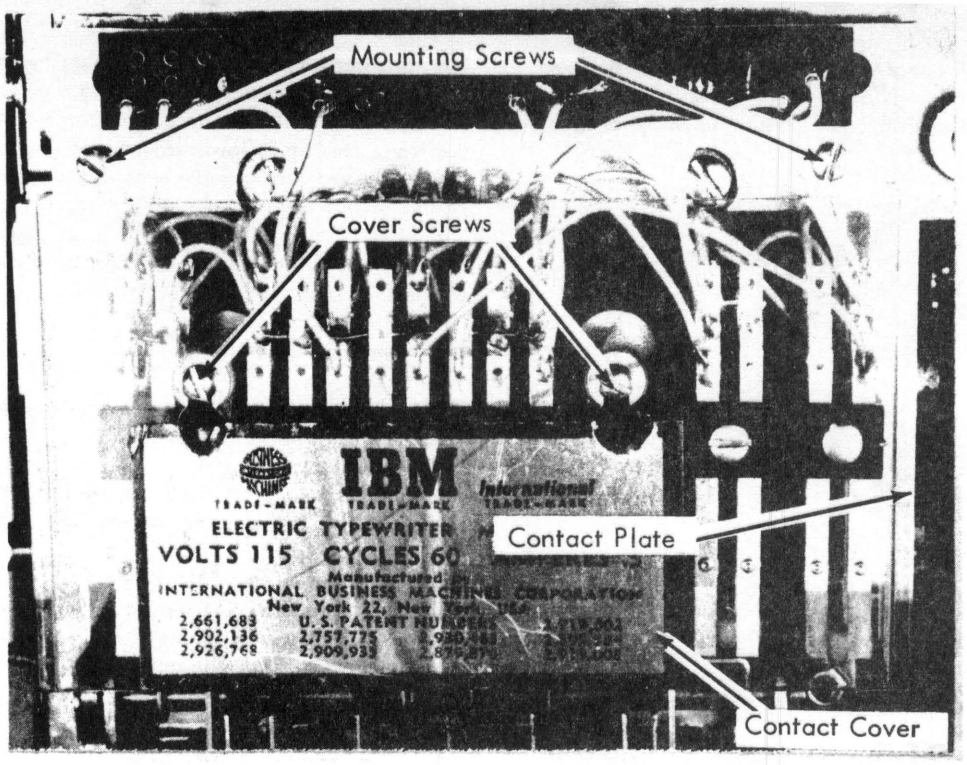


Figure 44. Contact Cover

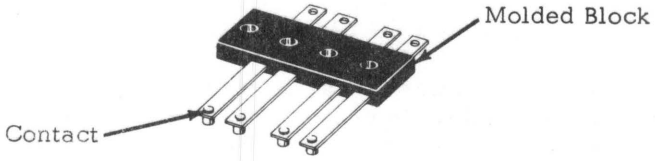


Figure 45. Contact Block

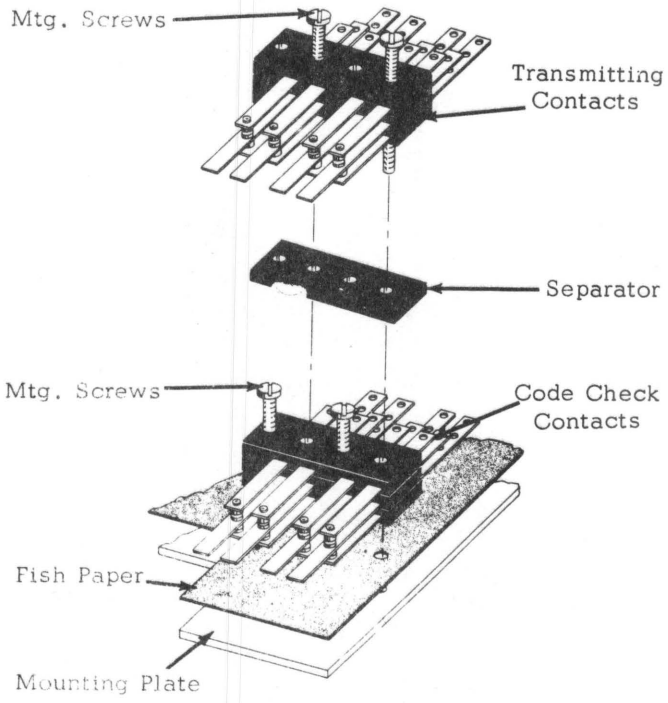


Figure 46. Transmitting Contact

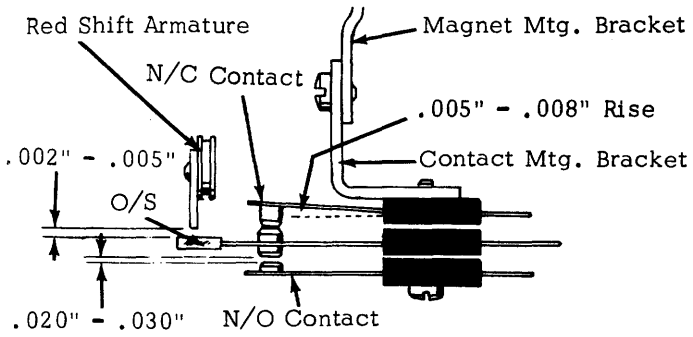


Figure 47 Ribbon Transmit Contact Asm.

SPECIFICATIONS

VOLTAGE AND CURRENT REQUIREMENTS

I. DRIVE MOTOR

a. Running Current - 1.0 Amperes at 115 volts 60 cycles.

II. MAGNETS - All except upper case shift and red ribbon.

a. Operating Voltage - 43 to 53 volts.
b. DC Resistance - 432 - 518 ohms.
c. Maximum Current - 125 ma.

III. UPPER CASE SHIFT MAGNET

a. Operating Voltage - 43 - 53 volts.
b. DC Resistance - 221 - 259 ohms.
c. Maximum Current - 300 ma.

IV. RED RIBBON SHIFT - One magnet (Optional)

a. Operating Voltage - 43 - 53 volts.
b. DC Resistance - 345 - 455 ohms.
c. Maximum Current - 160 ma.

V. RED RIBBON SHIFT - Two magnet (Optional)

Red Magnet
a. Operating Voltage - 43 - 53 volts.
b. DC Resistance - 221 - 259 ohms.
c. Maximum Current - 300 ma.

Black Magnet
a. Operating Voltage - 43 - 53 volts.
b. DC Resistance - 432 - 518 ohms.
c. Maximum Current - 125 ma.

VI. SOLENOIDS

a. Operating Voltage - 43 - 53 volts.
b. DC Resistance - 329 - 397 ohms.
c. Maximum Current - 160 ma.
d. Duty Cycle - 100%

VII. FEEDBACK, TRANSMITTING, & CHECKING CONTACTS (Gold Plated Silver)

a. Ratings:

1. 40 ma. at 10 volts (minimum).
2. 300 ma. at 48 volts (maximum).

VIII. FEEDBACK CONTACTS (Tungsten)

a. Ratings:

1. 1 amp at 53 volts (maximum).

TIMING REQUIREMENTS

I. MAGNET PULSES

a. All magnets must pick within 10 ms. at 48 volts. (Except red ribbon shift and keyboard lock solenoid).

1. Red Ribbon Shift must pick within 12 ms. at 48 volts.
2. Keyboard Lock Solenoid must pick within 55 ms. at 48 volts.

b. Magnet pulses for repeat cycles must begin no later than 18-20 ms. before the end of a current cycle.

II. MAGNET PULSE DURATION

a. See timing charts.

OPERATING SPEEDS (NOMINAL)

I. PRINTING CYCLE - 15.5 cycles per second minimum.

II. CYCLE TIME - 64.5 ms.

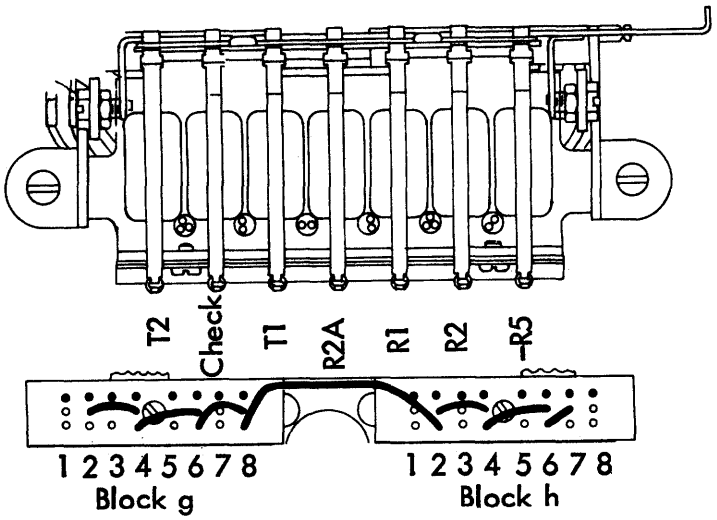
III. CARRIER RETURN AND TABULATION - 17 inches per second.

IV. SHIFT CAM AND CLUTCH - 15.5 cycles per second.

V. BACKSPACE - 15.5 cycles per second.

VI. SPACING - 15.5 cycles per second.

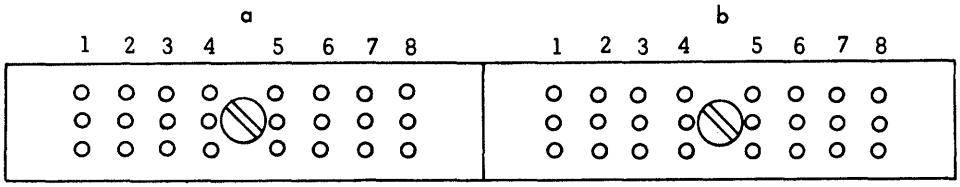
PRINT SELECTION MAGNET ASSEMBLY



WIRING CHART		
Coil	Black Lead	White Lead
T2	g-2	g-1
Check	g-4	g-3
T1	g-6	g-5
R2A	g-8	g-7
R1	h-2	h-1
R2	h-4	h-3
-R5	h-6	h-5

Figure 48

SELECTION TRANSMITTING CONTACT ASSEMBLY



SELECTION CONTACT TERMINAL BLOCKS

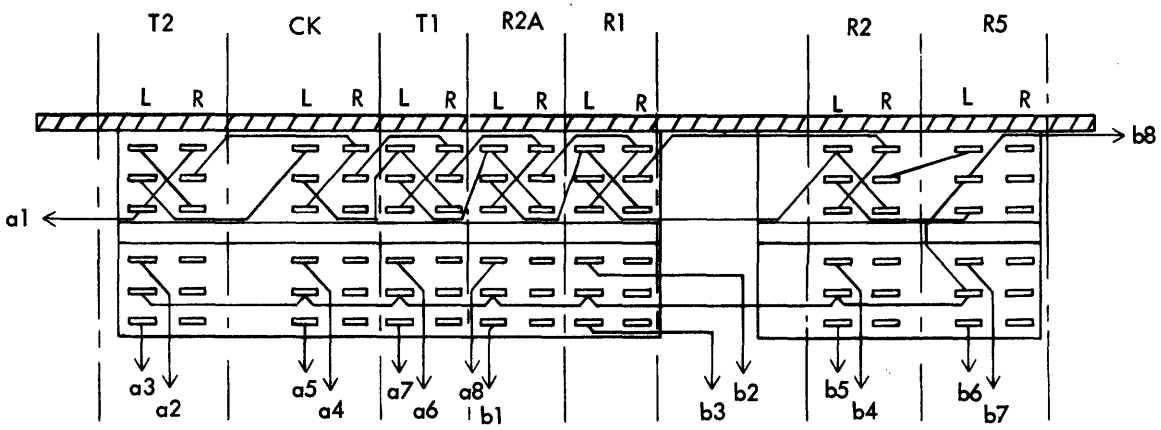


Figure 49. TERMINAL END VIEW OF SELECTION CONTACTS – CORRESPONDENCE

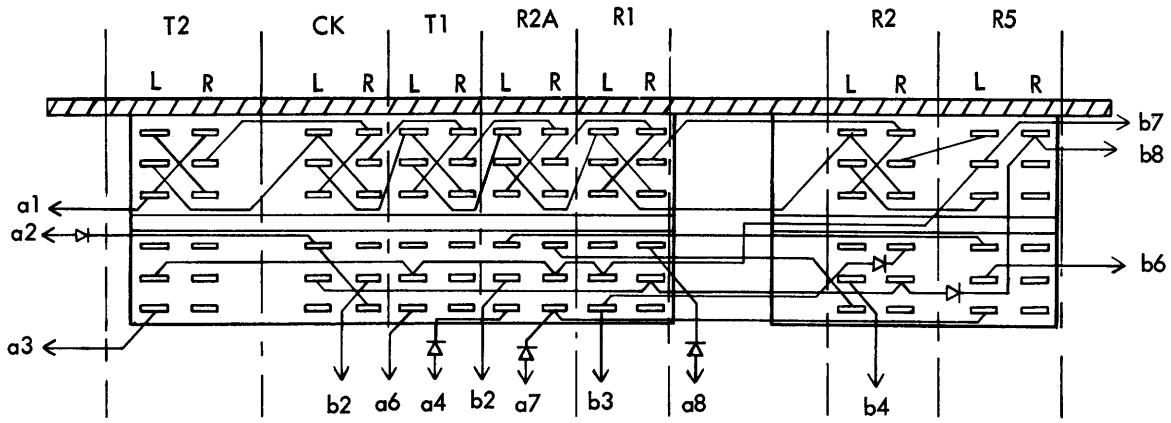
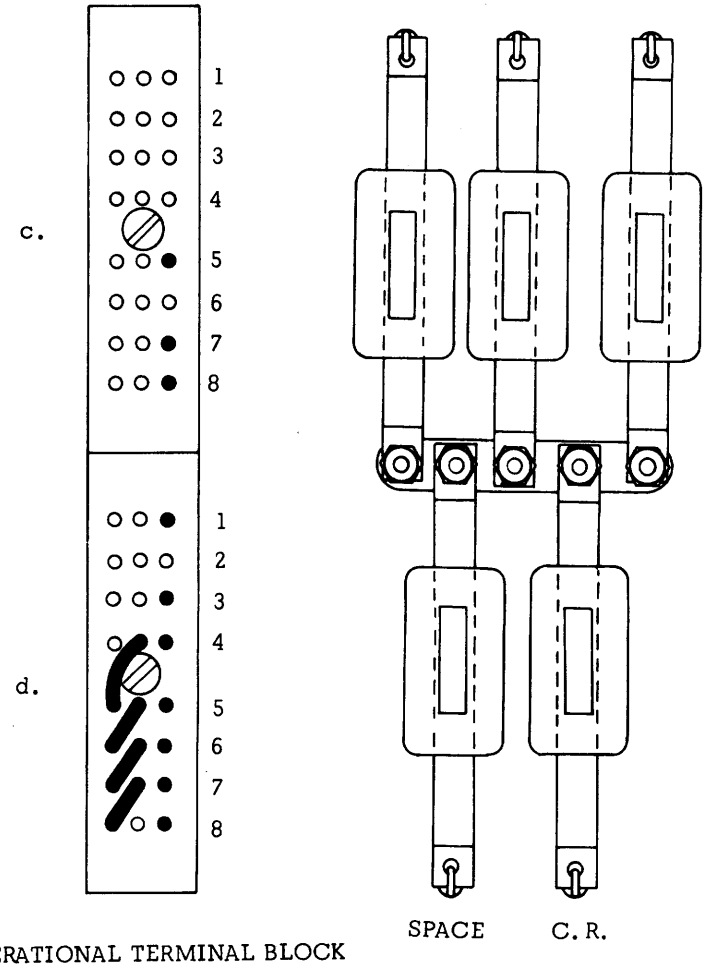
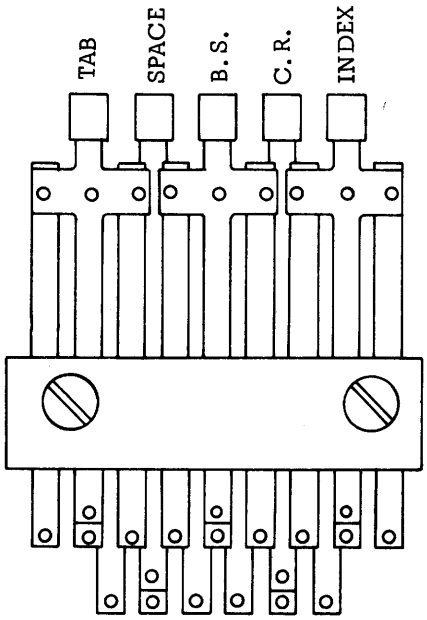


Figure 50 Terminal End View of Selection Contacts (BCD)



OPERATIONAL TERMINAL BLOCK

OPERATIONAL MAGNETS



OPERATIONAL MAGNET CONTACTS

Magnet Wiring Chart		
Coil	Black Lead	White Lead
Tab	d-4	c-5
Space	d-5	c-7
Backspace	d-6	c-9
Carrier Return	d-7	d-1
Index	d-8	d-3

Figure 51

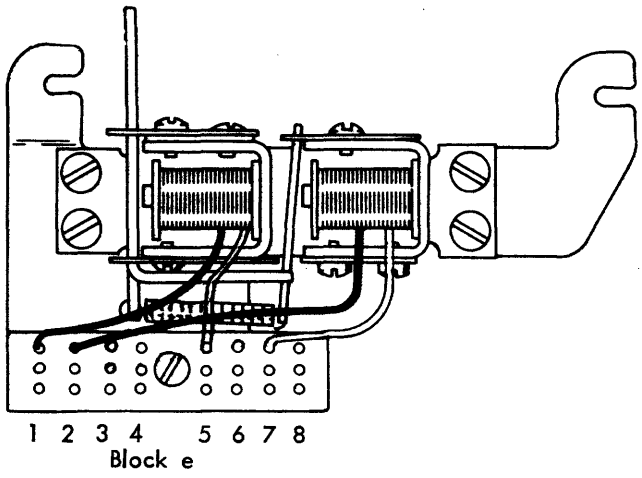


Figure 52. SHIFT MAGNET ASSEMBLY

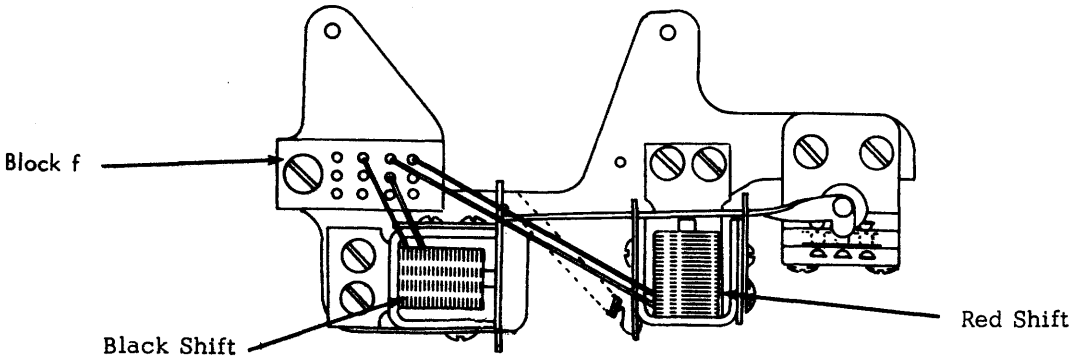


Figure 53. 2 Magnet Red Ribbon Shift Assembly

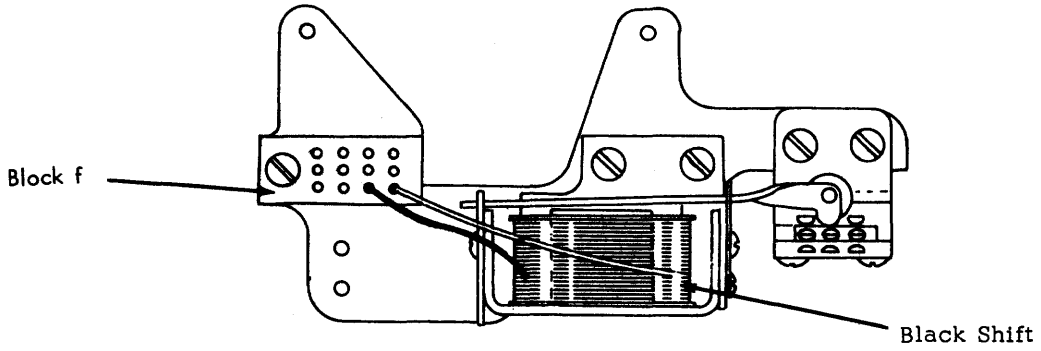


Figure 54. 1 Magnet Red Ribbon Shift Assembly

READER COMMENTS

The General Electric Company solicits your comments on publications covering Process Computer equipment. Please explain any "no" responses in the COMMENTS section. Your comments and suggestions become the property of the General Electric Company.

- Name of Manual: _____
- What is your computer application: _____

  _____
- How is this publication used:

  Familiarization ☐ Reference ☐

  Training ☐ Maintenance ☐

  Other (Explain) _____

	YES	NO
• Does this publication meet your requirements	☐	☐
• Is the material:		
1) Presented in clear text	☐	☐
2) Conveniently organized	☐	☐
3) Adequately detailed	☐	☐
4) Adequately illustrated	☐	☐
5) Presented at appropriate technical level	☐	☐

- Please provide specific text references (page number, line, etc.) with your comments.

NAME _____ DATE _____

TITLE _____

COMPANY NAME _____

AND ADDRESS _____

COMMENTS:

No postage necessary if mailed in the U. S. A.

Staple

Communications concerning Technical Publications should be directed to:

Manager, Technical Publications
Utility and Process Automation Systems Operation
2255 West Desert Cove Road
Phoenix, Arizona 85029

Fold Fold

FIRST CLASS
Permit No. 4091
Phoenix, Arizona

BUSINESS REPLY MAIL

NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY...

GENERAL ELECTRIC COMPANY
UTILITY and PROCESS AUTOMATION
SYSTEMS OPERATION
2255 West Desert Cove Road
Phoenix, Arizona 85029

Attention: Technical Publications

Cut Along Line

Fold Fold

Additional Comments:

4852B DUAL BULK CONTROLLER

TABLE OF CONTENTS

INTRODUCTION

This publication contains theory of operation information for the Model 4852B Dual Bulk Controllers. The controller's primary function is to control data transfer operations between the Core Memory and the system's Bulk Memory Units, referred to as BMU's.

The basic controller (4852BS011) provides a two-bulk-device capacity but a two-unit adder option (4852BS021) expands the limit to four devices per controller.

Controller operations are initiated by GEN II commands, addressed to it from the Central Processor. After it receives the command, the controller governs the functions necessary to complete the operation. It communicates with the BMU selected for the operation and handles data transfer operations between the BMU and Core Memory.

A data transfer operation is initiated using an OUT command addressed to the controller. The controller responds by requesting a pointer word from a controller pin-selectable core address. This word is used by the controller to address the first of three control words it will access from Core Memory. These control words inform the controller of the type of transfer (read from BMU or write on BMU) to be performed and the desired BMU to be used for the transfer, the starting Core Memory and BMU address for the transfer, and the number of transfers that are to take place.

The number of data transfers resulting from an OUT command is determined by the contents of Control Word 3, the last of three control words accessed by the controller. If the number is zero, no transfers will take place but the selected BMU will move its heads to place them over the desired starting address. This operation is known as a "seek-only" command. Since the drum unit's heads are fixed, this operation is only useful for directing disk units to seek a specific cylinder address.

Following control word accessing, and head movement if necessary, the actual transfer operation begins. Data are transferred parallel between Core Memory and the controller, and serially between the controller and the BMU.

The controller monitors various circuitry within itself and its connecting bulk devices for error conditions. Any detectable error is recorded and the operation, if one is in progress, is terminated. Termination occurs immediately after detecting the error unless there is a write operation in progress. Termination from errors detected during write operations await the beginning of the next 64-word group, referred to as a sector.

Other GEN II commands are available for controlling and testing controller and BMU operations. For example, an IN command can be utilized to transfer the status of the error detection flip-flops to a program-accessible register in the Central Processor. An Abort command is available to terminate controller operations conditionally or unconditionally, depending upon the command word coding. JNR and JNE test commands can be used for testing ready and error conditions, usually during troubleshooting routines.

An Activate command is utilized for initiating ready interrupts for the controller or the individual BMU's if they are not in use. (No interrupt results if the selected device is busy during the execution of the Activate command.) Ready interrupts are generated automatically by the controller and BMU's when they complete an operation.

The OPR command is decoded by the controller as a Return-to-Zero Seek operation. It is intended to be used in directing a disk unit to return its heads to a known location (i.e., cylinder zero) and clears the File Unsafe flip-flop in the bulk device.

Data organization differs between the drum and disk units. This organization is explained in the "Functional Description" portion of this publication.

Information contained in this publication is oriented primarily towards the controller's functioning during the different command operations. Separate theory of operation publications exist for the other functional modules, and for the drum and disk units.

This publication's text is organized into sequenced descriptions of the OUT command functions (read, write, verify, fill, and seek-only), and of the non-sequencing Abort, Activate, IN, JNE, OPR, and JNR commands. Fig. WHD 7.4 provides an overall generalized outline of OUT command functions. The sheet containing the flow chart is folded so that it can be extended out from the main body of the text and used as an outline while referring to the text, block diagrams, and timing diagrams that precede it.

All logic sheets referred to within the text and on illustrations of this publication are for the controller's logic, 70C180908.

REFERENCE DOCUMENTS

The following is a list of the technical publications, logic drawings, and the Dual Bulk Subsystem Test program, for the Dual Bulk Subsystem. Each document listed has a brief mention of its contents and a reference to where it may be located.

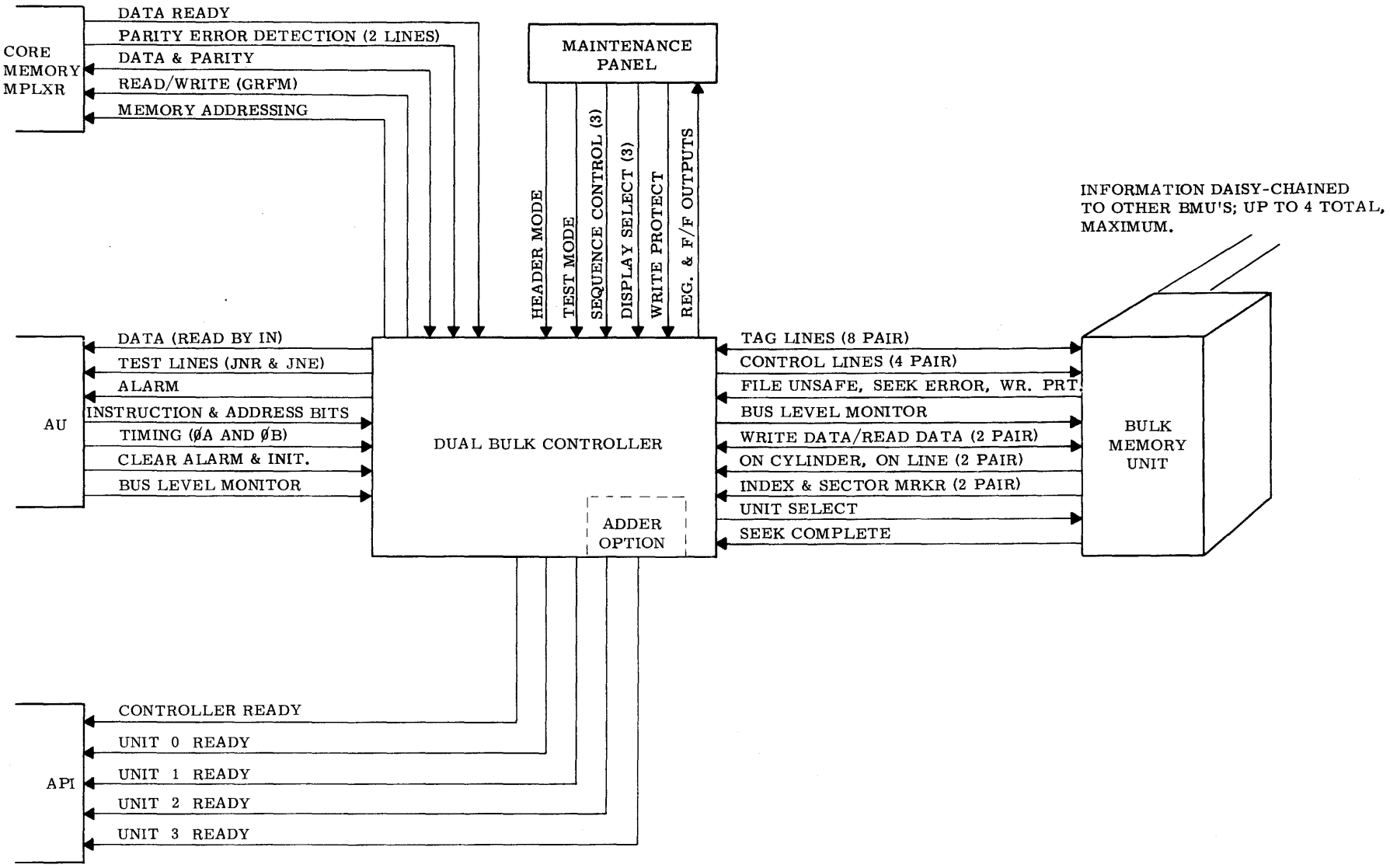


Fig. INT. 1 Simplified Block Diagram

Technical Publications

DISK

- MODEL 133 DISK DRIVE TECHNICAL MANUAL (Vendor)
  This manual is published by the disk unit's manufacturer, CDS, and is shipped with 4010B computer systems containing Model 133 disk drive units. The manual contains theory and maintenance information for the disk units.

- 4548C/4852B-M DISK MEMORY UNIT MAINTENANCE (GE)
  This publication is contained in the Disk section of the Computer Maintenance Manual for systems having a disk unit. The publication describes the type of disk unit implemented on the 4010B computer systems and contains maintenance information not otherwise covered by the Vendor manual.

DRUM

- 4521A-T DRUM MEMORY UNIT THEORY (GE)
  This publication is contained in Volume III of the Theory of Operation manual for systems having drum units. The publication contains a descriptive analysis of the drum memory's operation.

- 4521A/4852B-M DRUM MEMORY UNIT MAINTENANCE (GE)
  This publication is contained in the Drum section of the Computer Maintenance Manual for systems having a drum unit. It provides maintenance-related information for the drum.

Logic Drawings

The following logic drawings contain logic and/or schematic representations of the circuitry for the items listed. These drawings are included in the System Drawings book set.

- 4852B DUAL BULK CONTROLLER - 70C180908

- 4521A DRUM MEMORY UNIT
  Drum Memory Unit - 70C180293
  Drum Read Electronics - 70C180119

Test Program

The following test program can be found in the book set labeled "Test Instructions", if the system contains a Dual Bulk Subsystem.

- 1500A Dual Bulk Subsystem Test-70A100003

FUNCTIONAL DESCRIPTION

BASIC READ/WRITE THEORY

Drum and disk recording utilize the principle of inducing concentrated flux fields on magnetizable surface and retaining these fields for later pickup. The process of inducing the fields upon the disk is referred to as writing, while the pickup of these fields is referred to as reading. The recording is accomplished using a time sharing transducer, referred to as the read/write head.

In order to understand the recording principles involved, it is necessary to examine the theory of magnetism. All magnets have two poles, a north pole and a south pole. There is a magnetic field around the magnet consisting of imaginary lines of force. The lines of force emanate from the north pole and enter the south pole, returning to the north pole through the magnet itself, thus forming a closed loop. (See Fig. FNL. 1.) The entire quantity of magnetic lines surrounding a magnet is called magnetic flux, while the number of lines per unit area is referred to as flux density.

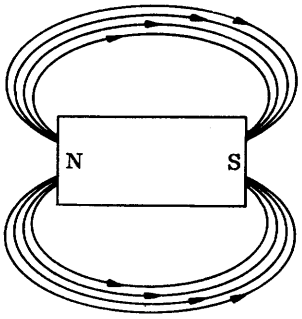


Fig. FNL. 1 Magnetic Lines of Force

When a magnet is bent to form a loop without the ends touching, as shown in Fig. FNL. 2, there is still a north and south pole, but the magnetic field is of shorter length and greater concentration than the bar magnet.

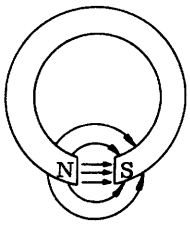


Fig. FNL. 2 "Horseshoe" Magnet Showing Concentration of Magnetic Field

One characteristic of the imaginary lines of force in a magnetic field is that they tend to take the path of least resistance, or in other words flow through the material that has the greater permeability (conductivity for magnetic flux). Air offers more resistance to the lines of force than a piece of iron or steel.

If a piece of iron is brought in close proximity to the gap of the horseshoe magnet,as shown in Fig. FNL. 3, the lines of force will tend to bend and flow through the iron. This is the principle of recording utilized, with an electromagnet being used as a magnet.

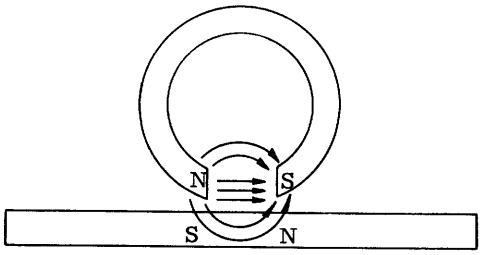


Fig. FNL. 3 Lines of Force Inducing Magnetism

The electromagnet sets up a north and south pole in relation to the direction of electron flow through its coil. With electron flow in the direction shown in Fig. FNL. 4, the magnet will set up a corresponding north/south pole relationship as shown. Reversing the direction of electron flow will, in turn, reverse the polarity of the magnet. As also indicated by the figure, the induced field will be of opposite polarity of the electromagnet. Thereby, information may be recorded by varying the direction of current flow through the electromagnet, or in this case, write head, as the recording medium passes beneath the head.

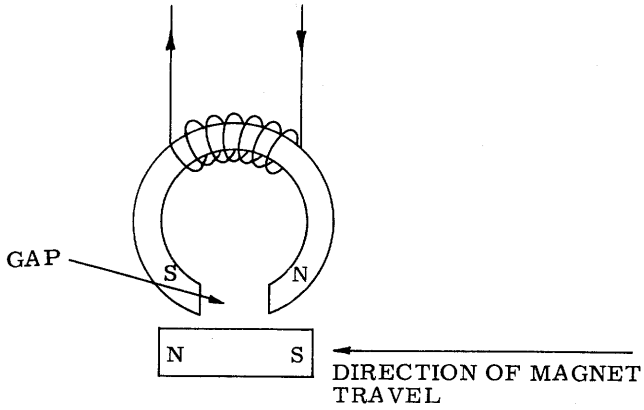


Fig. FNL. 4 Inducing Current in a Coil

Once recorded, the information remains until written on again. By removing current from the electromagnet, and allowing current to be induced into the coil by rotating the disk beneath it and having the recorded information induce current into the coil in relation to the manner of which it was recorded, we may read the information back (see Fig. FNL. 4). The signal is then amplified and shaped into pulses to be detected as a binary one or zero.

The write circuitry employes a non-return to zero method of recording, during which there is a constant flow of current through the write head. The

write current is alternately supplied from the two legs of the write amplifier, through the write head to the grounded center tap.

For disk units, an erase head is included as part of the head mechanism, preceding the read/write head as the disk spins beneath it. The erase head is enabled during a Write operation and erases previously recorded information by having a DC current passed through the erase head, thereby aligning the disk track flux in a constant direction. The write head, positioned behind the erase head physically, as well as in time, then writes data provided by the controller over the erased surface. Drum Units simply overwrite previous information and have no erase heads.

During the Write operation, pulses are sent to the write circuitry at regular intervals. These intervals are known as clock A's (clock 2's). At a period halfway between these clocks, another clock is also generated. This clock is referred to as clock B (clock 1). During this time, the data to be written on the disk is sampled to determine its value (binary "one" or "zero"). If the data to be recorded is a "one", the pulse is allowed to be recorded. If the data is a "zero", the pulse is not recorded. Whenever the data is read back, it is examined at clock B (clock 1) time, generated from the incoming data, to determine its value.

PROGRAM CONTROL

There are seven GEN II commands that can be used for initiating, initiating and controlling, terminating, and testing Bulk Memory Controller operations. They are as follows:

- Activate (ACT) $2501K_3K_2K_1K_0$
- Operate (OPR) $2502K_3K_2K_1K_0$
- Abort (ABT) $2503K_3K_2K_1K_0$
- OUT (OUT) $2504K_3K_2K_1K_0$
- IN (IN) $2505K_3K_2K_1K_0$
- JNR (JNR) $2506K_3K_2K_1K_0$
- JNE (JNE) $2507K_3K_2K_1K_0$

All of these commands can be used in an operating program. The JNR command, which can be coded via the K_1K_0 bits to test ready conditions for the controller or Bulk Memory Units, is used in test routines, and in operating routines following ready interrupts to verify that there was not a false ready interrupt.

K_3K_2 Addressing

There are four optional addresses available for the Dual Bulk Controller (DBC), determined by wiring at time of manufacture. They are K_3K_2 octal addresses 10, 11, 12, and 17.

ACT Command

An Activate command can be used to initiate an action involving the DBC by activating the controller, or one of the Bulk Memory Unit's, interrupt lines. The interrupt line, determined by the K_1K_0 coding of the ACT command, will not cycle if the specified device is busy. K_1K_0 coding of the ACT command to cycle the various lines is as follows:

S'=0, K = 000	Controller Ready
S'=1, K = 000	Seek Complete (Unit 0)
S'=1, K = 001	Seek Complete (Unit 1)
S'=1, K = 010	Seek Complete (Unit 2)
S'=1, K = 011	Seek Complete (Unit 3)

Note: S' is Bit 03 of the GEN II command

OPR Command

This command is decoded by the controller as a return-to-zero seek operation. Its purpose is to permit the control program to place a disk unit's head at a known location (i.e., cylinder 0). If the S' bit (Bit 03 of the OPR command) is a zero, the unit selected for the operation is the same as that accessed by the previous command. When S' equals a "one", the K_0 bits are used to decode the unit address, as shown in the ACT command.

ABT Command

An Abort command can be used by the control program to terminate a controller operation in favor of a higher priority operation. Two levels of abort are made possible by coding of the ABT command. If the S' bit is a "zero", the controller causes a conditional abort. When the S' bit is a "one", the controller unconditionally terminates the controller's operation upon receipt of the command, which may destroy data on the bulk device. (Refer to the Abort command description, found under "NON-SEQUENCING COMMANDS" later in this publication for further details on conditions for aborting.)

OUT Command

The OUT is the only command to cause the controller to enter a sequenced operation, necessary when performing all the functions required to transfer data. In addition to being the command used to initiate data transfer operations, the OUT can be used to initiate a "seek-only" command that will cause a disk unit to move its heads to a desired cylinder.

All of the necessary addressing information, for both Core Memory and selected Bulk Memory Unit (BMU),

and the transfer length are specified in three control words. These control words are accessed from Core Memory by the controller after it receives the OUT command. A pointer word is accessed first by the controller, from a core address preset into the controller's Core Address Register. This preset address is pin-selectable and can be 00_8, 11_8, 12_8, or 13_8. The pointer word informs the controller of the core location for the first of the three control words. Control words two and three are accessed from the next two consecutive core locations.

Fig. FNL. 5 illustrates the format of the pointer and control words. The contents of these locations must be supplied by the operating program prior to the execution of the OUT command.

If the contents of CW2 is all "zeros", a "seek-only" operation is performed and no data transfers will result from that OUT command.

Enabling the "Test Mode" switch on the controller's Maintenance Panel permits the OUT command to be utilized for writing "header" information on the bulk units, or for stepping through an OUT command sequence so that controller functions can be monitored on selectable indicators on the Maintenance Panel. Both of these operations can be executed in conjunction with the Dual Bulk Subsystem Test program, GE number 70A100003. The "Header" switch must also be enabled on the Maintenance Panel when performing a "Write Header" operation.

Stepping through an OUT command, referred to as an "Increment Sub-Operation", is enabled through the presence of S'=1 (Bit 03 of the OUT command) when the Test Mode switch is enabled. The accompanying test program enables incrementing to Sequence State 5, in one-sequence-state increments, upon depressing the "Demand" pushbutton on the Computer Maintenance Console. Depressing the Demand button during SS5 increments the operation to each control information transfer (see "OUT COMMAND - SS5"). At the end of SS5, depressing the button increments the sequence to the end of SS6. Subsequent stepping (now in SS7, the transfer sequence state) increments one sector transfer at a time.

IN Command

The IN command is normally used in response to an error detection (detected by a JNE command). This command permits the reading of various controller status, including error conditions, into the "A" register of the Arithmetic Unit. (Refer to "IN COMMAND", found under NON-SEQUENCING COMMANDS description in this publication, for further detail.

JNR Command

A JNR command permits testing of the controller or bulk devices availability. The controller is made unavailable, "busy", as a result of an OUT command. Bulk devices can become busy as a result of an OUT or OPR command. Operating programs are informed of the controller and device availability by ready interrupts.

This command is utilized in test programs where the use of interrupts is not desired. Selection of the line for sampling is determined by the coding of the JNR's S' and K_0 bits, as specified by the following:

S'=0, K_0 = any configuration	Controller Ready
S'=1, $K_0 = 000_2$	Unit 0 Ready
S'=1, $K_0 = 001_2$	Unit 1 Ready
S'=1, $K_0 = 010_2$	Unit 2 Ready
S'=1, $K_0 = 011_2$	Unit 3 Ready

If the sampled line is ready, the AU's Program Register will increment by one; if it is not ready, the Program Register will increment by two (jump).

JNE Command

The JNE is used to determine if any errors have been detected by the controller. This command usually precedes the execution of an OUT command to the controller in an operating program. If an error is present, the AU's Program Register is incremented by one - typically branching the operation into an error recovery or identification subroutine. If no error is present, the Program Register is incremented by two (jumps) - typically branching to an OUT command.

DRUM AND DISK DATA ORGANIZATION

Both drum and disk units contain read/write heads flying over a magetizable recording surface. The magnetic recording principle (see Basic Read/Write Theory) utilized by these heads permits recording and subsequent recovery of data. For both types of devices the heads cover circular areas of recording surfaces, referred to as "tracks", that are divided into 16 sectors of data. Each sector contains 64 24-bit data words. Therefore, a head covering a track of data encompasses 1024 data words (16 sectors x 64 words/sector).

The controller is synchronized, during data transfers, to the beginning of a data sector by the receipt of a "sector marker" from the unit. An additional pulse is received when the sector is the first for the track (Sector 00); this pulse is referred to as the "index marker".

Drum Units

Fig. FNL. 6 illustrates the head positionings for drum units. Drum unit heads are fixed, spaced evenly along the surface of the drum. Each head is individually selectable for reading or writing and the number of heads utilized is determined by the storage capacity of the particular drum unit.

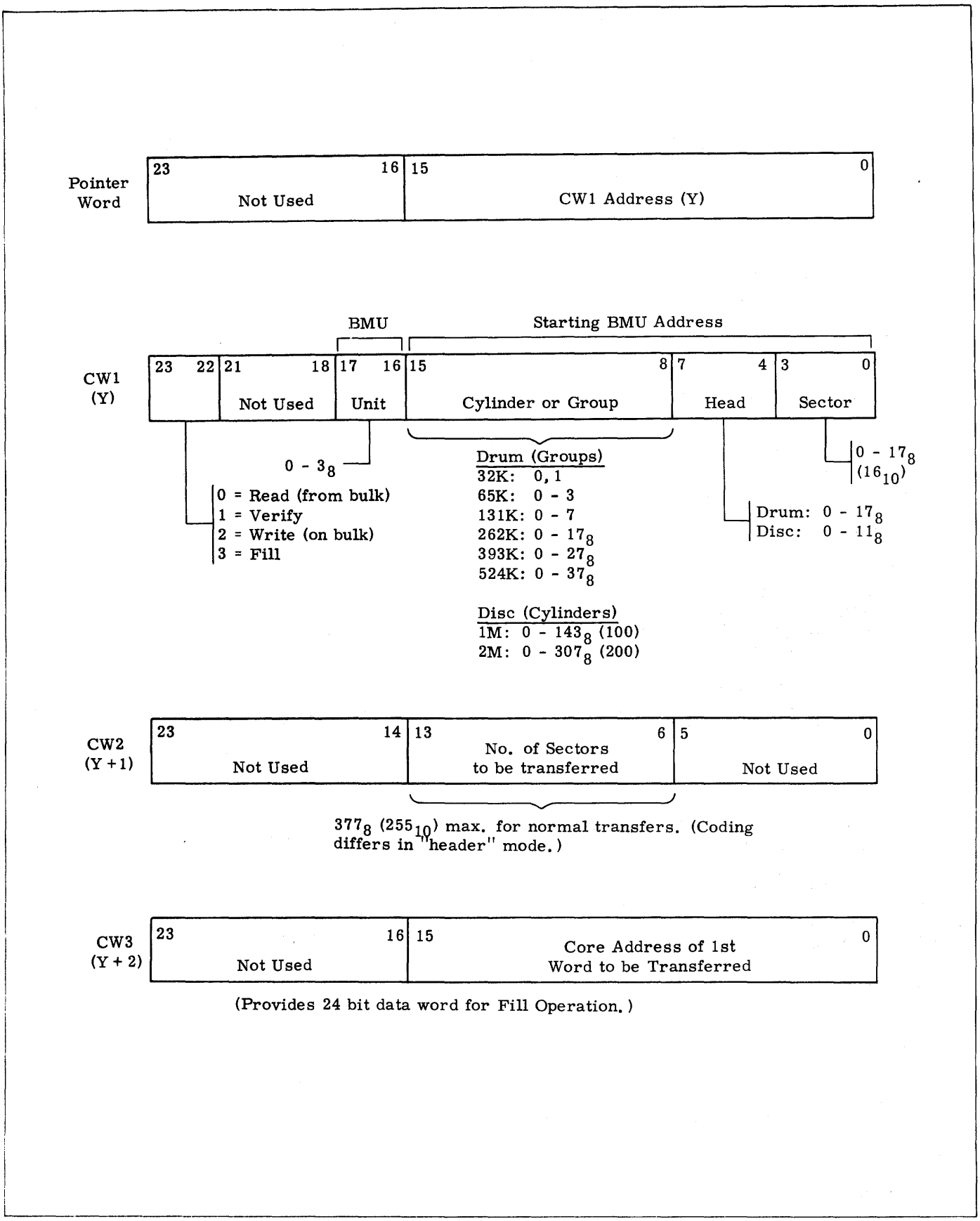


Fig. FNL. 5 Pointer and Control Words

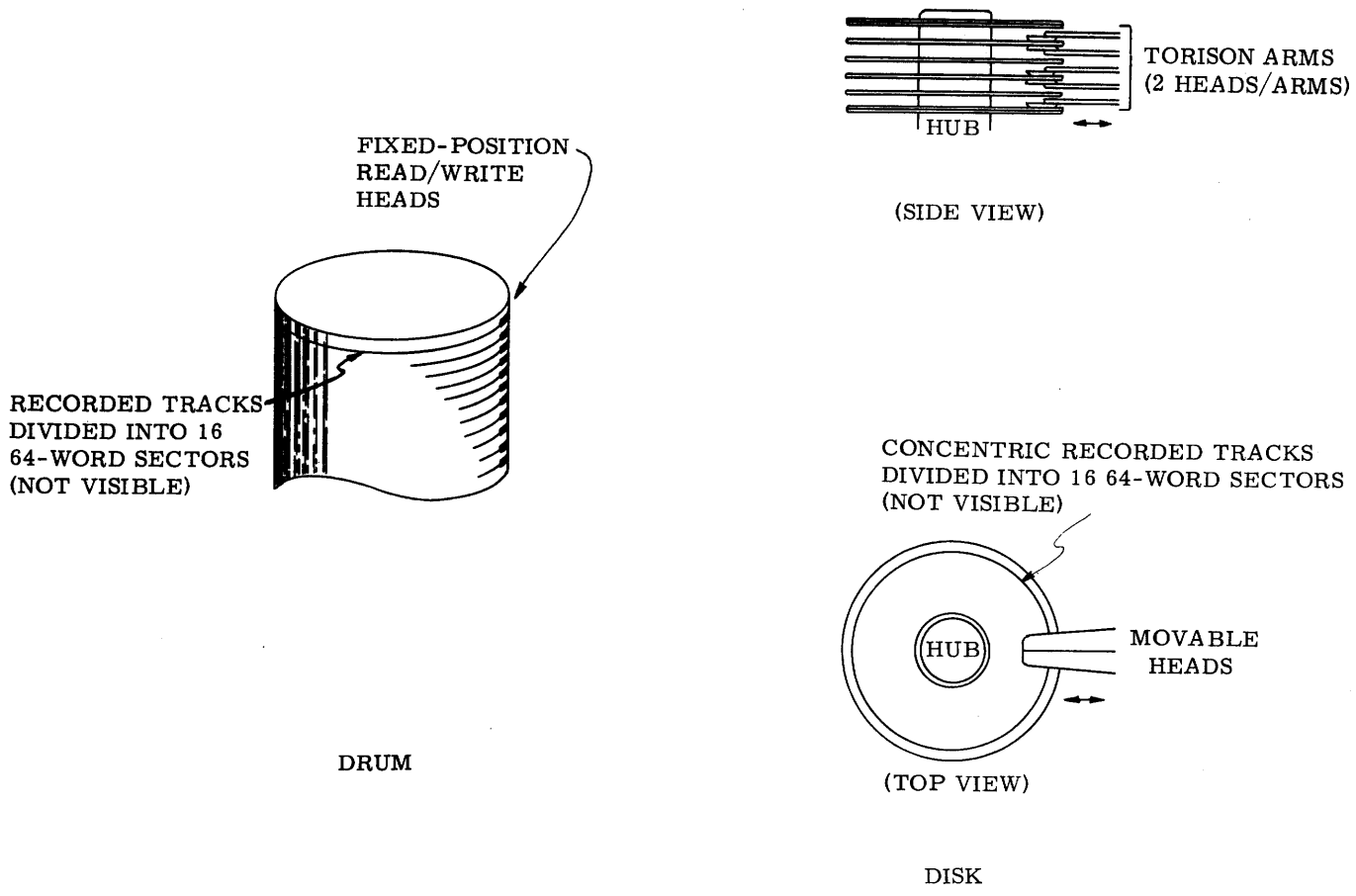


Fig. FNL. 6 Drum and Disk Head Positionings

A read or write operation on a drum can begin at any sector of any head, as selected by the controller. If the operation continues beyond the last sector of the head, the next subsequent head is selected and the operation continues at the first sector of that head. Information is addressed sequentially, sector by sector, incrementing to the next head at the end of the last sector for a head, until the required amount of data has been transferred. The amount of data to be transferred is determined by a count in the controller.

Disk Units

Disk units contain 10 read/write heads, fixed in a vertical alignment, that can be moved horizontally across the surfaces of the disks to rest over any one of the concentric tracks of information. At any one of these locations, the heads actually cover 10 tracks of information. These ten tracks, encompassed by the ten read/write heads, comprise a "cylinder" of data. Each cylinder of data contains 10,240 data words (10 heads x 16 sectors/head x 64 data words/sector).

The controller can specify that data transfers begin at any existing cylinder. (Disk units are available with 100 or 200 cylinder limits.) Unless the heads are already at the desired cylinder, this requires a seek operation to place the heads at the desired cylinder. Starting sector and head addresses are also specified by the controller.

When a data transfer extends beyond the last sector for a head, the operation continues at the first sector of the next sequential head. (Heads sequence from top to bottom, referred to as Head 0 through Head 9.) If the data transfer extends beyond the last sector of Head 9, the transfers must be temporarily suspended while a "cylinder advance" is performed. A cylinder advance, which merely means moving the heads to the next sequential cylinder, requires an exchange of cylinder address information between the controller and the disk. (Refer to the "Cylinder Update" description, located under OUT COMMAND - SS5).

ADDRESS SELECTION

The organization of Control Word 1, the first control word transferred to the controller following the execution of an OUT command, is shown on the following page.

23	22	21 18	17 16	15 8	7 4	3 0
R/W	V/F	——	U	CYLINDER/GROUP	HEAD	SECTOR

R = Read (Bit 23 = 0 and Bit 22 = 0)
W = Write (Bit 23 = 1 and Bit 22 = 0)
V = Verify (Bit 23 = 0 and Bit 22 = 1)
F = Fill (Bit 23 = 1 and Bit 22 = 1
U = Unit (0 through 3)

Information contained in this control word is stored in registers within the controller and used later in the command sequence to determine the starting BMU address for data transfers (See Fig. FNL. 7). Bits 15 through 8 are gated into the Cylinder/Group Register, bits 7 through 4 into the Head Register, and bits 3 through 0 into the Sector Register of the controller.

Disk Units

If the selected BMU is a disk unit, a subtract operation is performed in the controller using information from the controller's Cylinder/Group Register as the minuend and information from the disk unit's Cylinder Address Register as the subtrahend. The resulting difference is gated back to the disk unit and stored in its Decrement Counter. This count is used by the disk unit to determine the amount of head movement necessary to reach the desired starting cylinder.

During a data transfer command, the controller enters Sequence State 7 (SS7) when the heads reach the desired cylinder. The read cicuitry is enabled, and the information beneath the selected read/write head is gated back to the controller. (Head selection is determined by the contents of the disk unit's Head Address Register, loaded from the controllers Head Register during SS5.) The beginning portion of the sectors, containing the read synchronization bits and the address information for the sector, is read until the sector's address compares with the desired starting address contained in the controller's Cylinder/Group, Head, and Sector registers. When comparison is achieved, the data transfer operation begins.

Drum Units

The subtract operation is also performed if the selected BMU is a drum, but the operation has no functional significance. In this case, zeroes are read from the drum during the Read Group period. This is subtracted from the Cylinder/Group Register, leaving the original contents of this register as the difference. The difference is gated to the Difference Register, replacing the zeros read from the drum during the Cylinder/Group read period. This count is not accepted since the BMU is a drum unit and has no need of a head movement count.

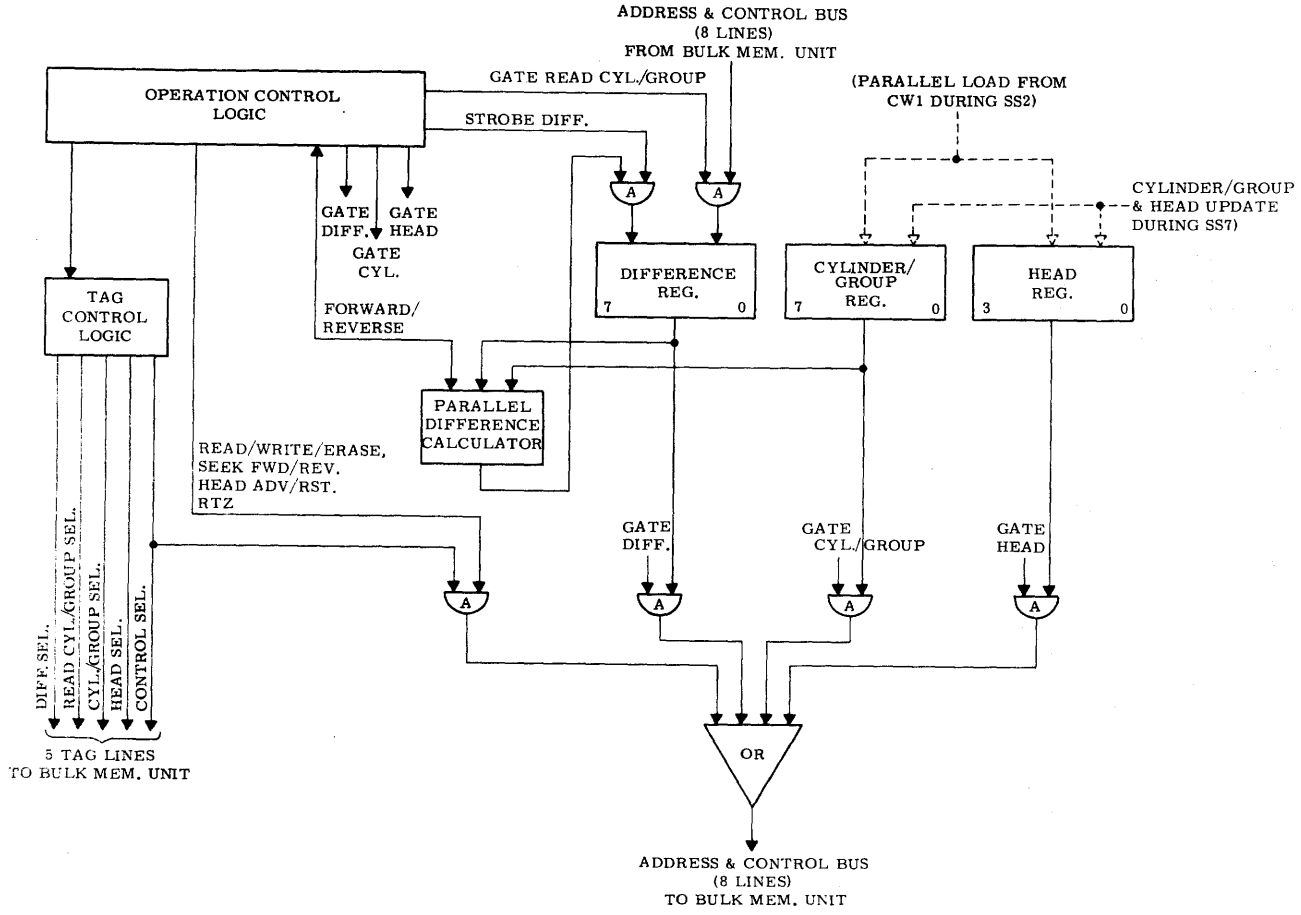


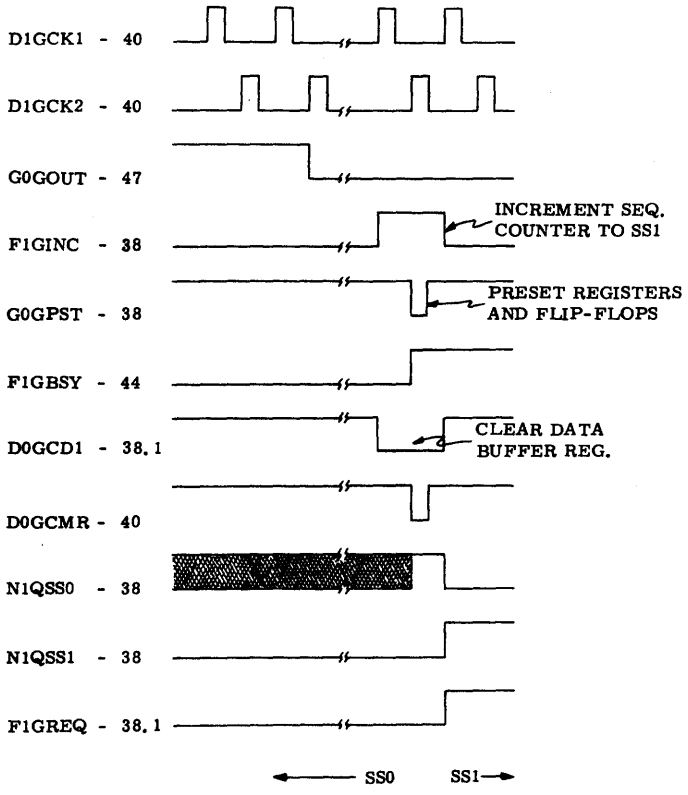
Fig. FNL. 7 BMU Control Information

OUT COMMAND - SS0

Sequence State 0 (SS0) is the controller idle state and defines the time during which data movement and status query commands may be received and processed. The controller remains in SS0 unless it is sent an OUT command (GEN II command with S bits equal to 4). An OUT command addressed to the controller causes it to increment to SS1 and sequence through its operation. The controller is returned to SS0 as a result of a successful completion of the operation, a detected error, or the receipt of an Abort command.

Upon entering SS0, the controller is timed from clock signals generated from a 6.92 MHz crystal oscillator. The signal from the oscillator is gated to a pair of flip-flops, F1RXCA and F1RXCB, shown on sheet 86 of the logic, which divide the frequency by four. Two clocks, CK1 and CK2, are generated from this circuitry and are used to time the events occurring up to SS6. If the selected Bulk Memory Unit (BMU) is a drum, the controller remains under the control of the same clock frequency. A 5 MHz clock is selected for use if the selected BMU is a disk unit.

Refer to Fig. OUT.0.1, for related timing diagram of this sequence state.



D1GCK1 = CLOCK 1

D1GCK2 = CLOCK 2

G0GOUT = GSEL · GGS4

F1GINC = GSTR = $\overline{\text{GISR}}$ · GCK1 · GPHB · GOUT · $\overline{\text{GBSY}}$ · $\overline{\text{PPER}}$
$\overline{\text{F1GINC}}$ = GSTB = $\overline{\text{OSTP}}$ · GBSY · GCK1

G0GPST = QSSO · GCK2 · GINC

F1GBSY = G0GPST

D0GCD1 = GPSD = GDIC = GROC · GDOC · $\overline{\text{GCAB}}$ · G1GREQ

D0GCMR = G1GCMR = GPIB · GPST

N1QSS0 = SEQUENCE STATE 0

N1QSS1 = SEQUENCE STATE 1

F1GREQ = G1GREQ · GBSY · $\overline{\text{RALM}}$ · GROC

Fig. OUT 0.1 Timing Diagram

OUT COMMAND - SS1

During SS1, the controller requests Core Memory for the transfer of the pointer word. The core address of the pointer word is preset into the controller's Core Address Register and will be one of four possible pin-selectable addresses. When the pointer word becomes available for transfer from Core Memory, it is read into the controller's Core Address Register and used as the starting core address for the subsequent accessing of the three control words.

SEQUENCE OF EVENTS

The basic sequence of events for this sequence state will be the requesting and receipt of the pointer word, its transfer through the controller to the Core Address Register, and the incrementing of the Sequence Counter to SS2.

The request for the pointer word is made with the setting of F1GREQ, shown on sheet 38.1 of the controller logic. F1GREQ sets on the trailing edge of G1GROC, which goes negative with the clearing of F1GINC at the end of SS0. F1GREQ clears, terminating the request, when the Data Ready pulse is received from the memory control logic.

Receipt of the Data Ready pulse is also used to set the Data Ready flip-flop, shown on sheet 39 of the logic. This enables a timing counter, consisting of flip-flops F1GTC1 and F1GTC2, to begin counting. During the count period, timing signals G0GTC1, G0GTC2, and G0GTC3 are enabled. The parallel transfers of the pointer word from the Data Buffer Register to the Accumulator Register and finally to the Core Address Register take place during the G0GTC1 and G0GTC2 pulses, respectively. Fig. OUT 1.2 shows the transfer path of the pointer word.

At the trailing edge of the N1GTC3 pulse, the Increment flip-flop, F1GINC, sets. It clears again on the following Clock 1 pulse, stepping the Sequence Counter to SS2.

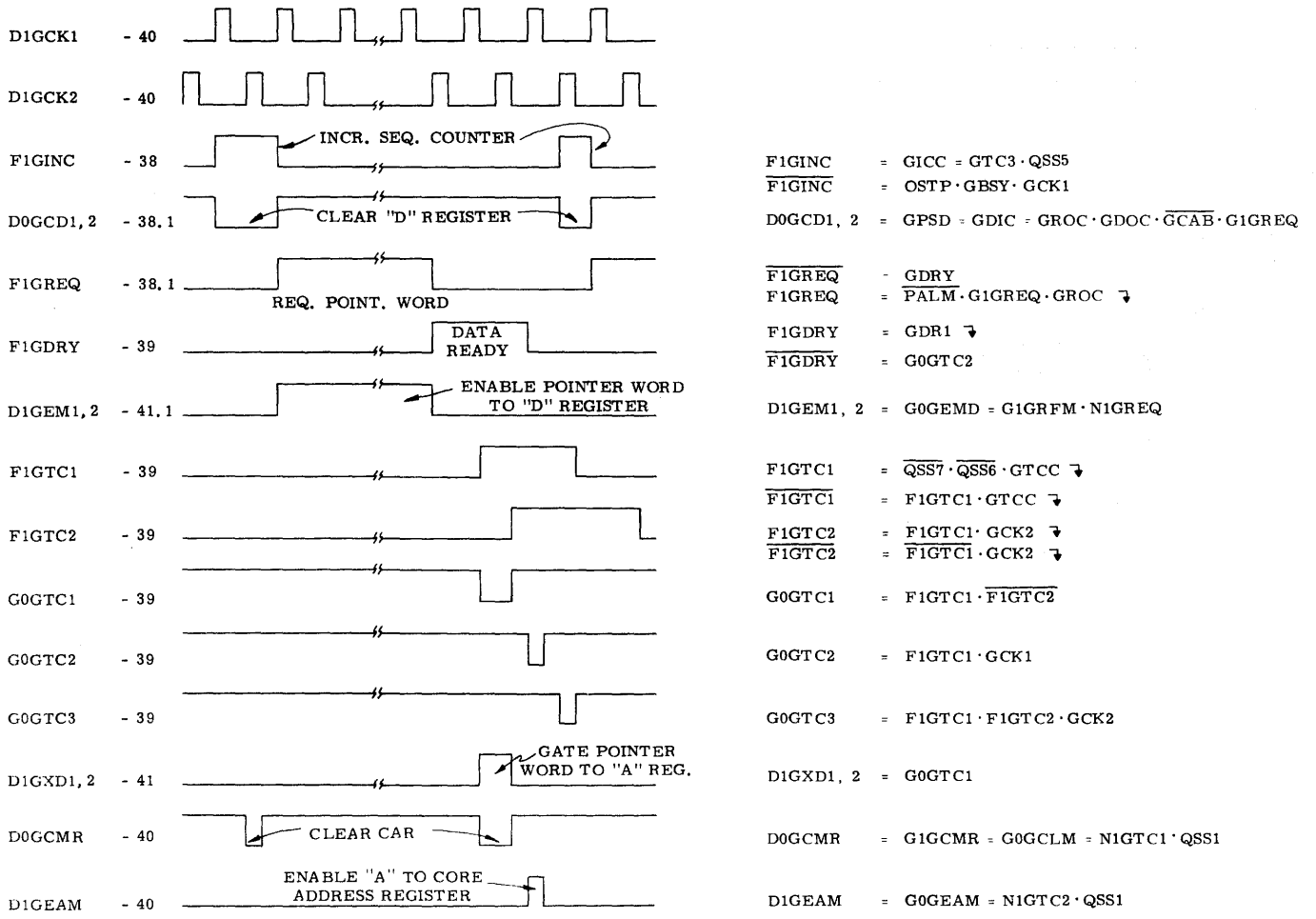


Fig. OUT 1.1 Timing Diagram

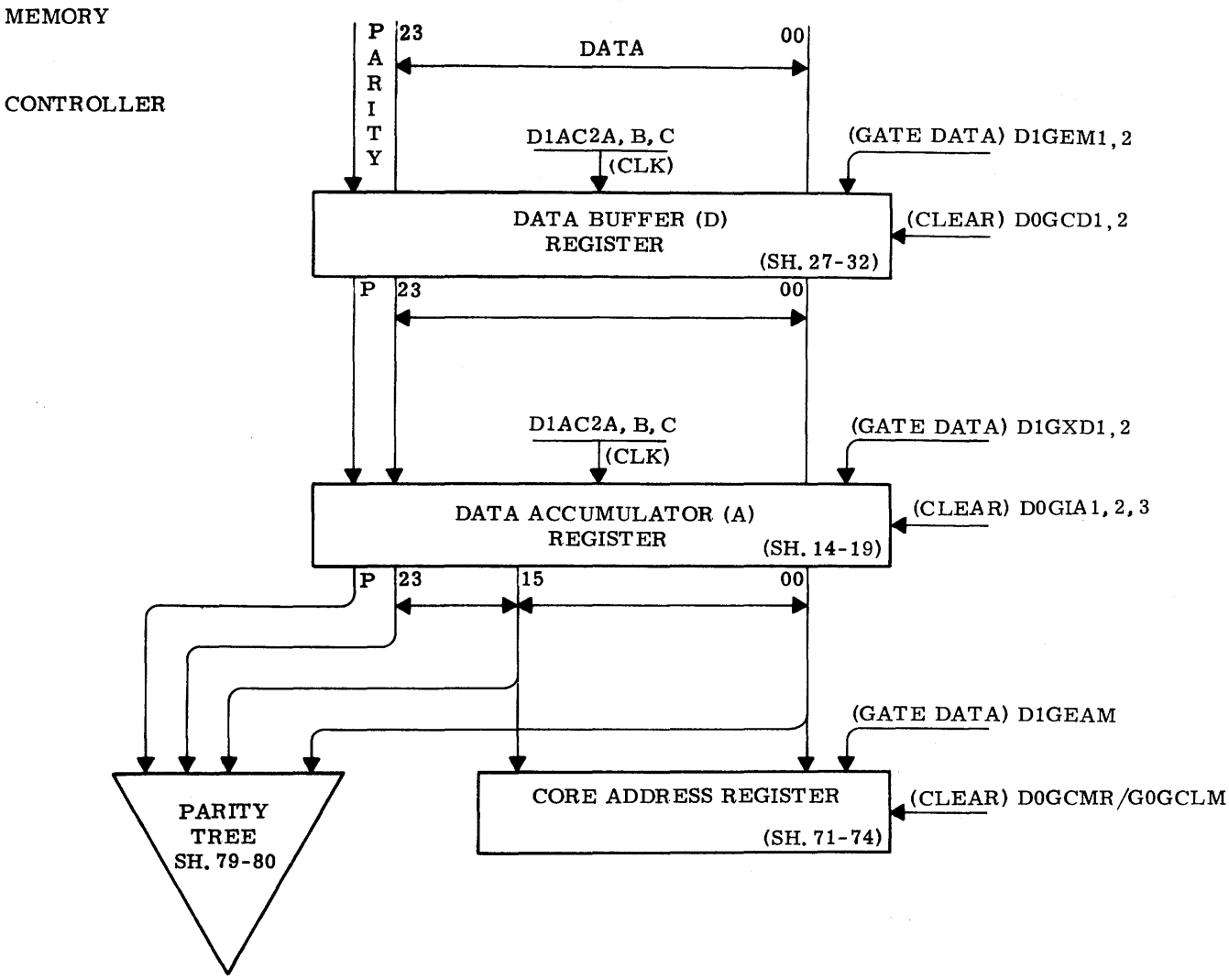


Fig. OUT 1.2 Block Diagram - SS1

OUT COMMAND - SS2

During Sequence State 2 (SS2) time, the controller requests and receives Control Word 1 (CW1) from the Core Memory. CW1 contains bulk memory addressing information (i.e., unit, cylinder or group, head, and sector) and designates the type of transfer that is to take place.

The timing of events for SS2 is shown in Fig. OUT 2.1. Fig. OUT 2.2 illustrates the general logic involved for requesting CW1 from Core Memory and transferring it through the controller to the various registers involved.

SEQUENCE OF EVENTS

Basically, the sequence consists of the request for CW1, the transfer of CW1 through the controller after it has been received from Core Memory, and the incrementing of the Sequence Counter to SS3.

The request for CW1 is initiated on the trailing edge of the G1GROC pulse by the setting of the Request flip-flop, F1GREQ. F1GREQ is cleared, terminating the core memory request, when the Data Ready pulse is received from the memory control logic.

Receipt of the Data Ready pulse is also used to set the Data Ready flip-flop in the controller. This enables a timing counter, consisting of flip-flops F1GTC1 and F1GTC2, to begin counting. During the count period, timing signals G0GTC1, G0GTC2, and G0GTC3 are enabled. The parallel transfers of CW1 within the controller take place during the G0GTC1 and G0GTC2 pulses. Fig. OUT 2.2 shows the transfer path of CW1, from the "D" register to the "A" register, and then to the various registers that are used to store the contents of CW1.

The Core Address Register is incremented by a count of one at the end of the G0GIMR pulse, which occurs coincidently with the G0GTC1 pulse. This provides the Core Address Register with the core address of CW2. F1GINC, the Increment flip-flop, sets at the trailing edge of G0GTC3 and clears at the next Clock 1 pulse, stepping the Sequence Counter to SS3.

SIGNIFICANCE OF CW1

The contents of CW1 will be used later in the sequence to determine what type of operation is to be performed, which bulk memory unit is to be accessed, and what the starting address will be for the operation.

Bits 23 and 22 are stored in the Unit Register of the controller and used during SS7 to determine which one of the four possible operations is to be performed. The operations and the corresponding bit configurations used to specify them are shown in the following:

OPERATION	BIT: 23	22
Read	0	0
Verify	0	1
Write	1	0
Fill	1	1

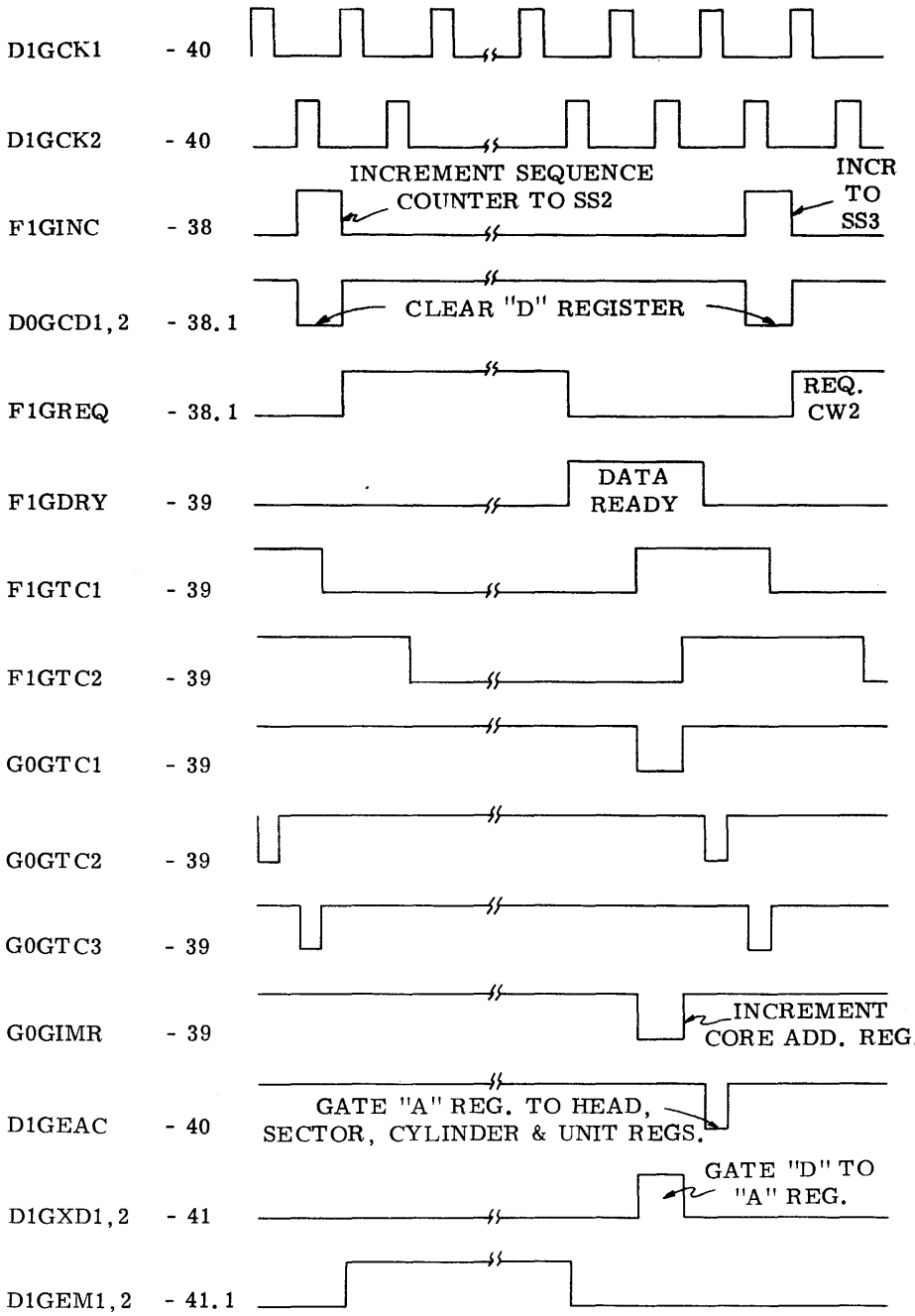
These operations are discussed in detail under "OUT COMMAND - SS7", located later in this publication.

Bits 17 and 16 are used by the controller for selection of the specified bulk unit to be used for the operation. They are gated to the Unit Register during this sequence state. The unit select decode logic provides selection of a single unit from these bits and the corresponding unit select line is enabled and sent to the device during sequence states four through seven.

Bits 15 through 8 are gated into the Cylinder/Group Register and used to select the starting cylinder (for disk units) or starting group (for drum units) for the selected unit. The contents of the Cylinder/Group Register are sent to the device during SS5. Although this information is accessed from the same register for transfer to either a disk or drum unit, it is regarded as "group select" when sent to a drum unit and "cylinder select" when sent to a disk unit. Refer to "Disk and Drum Specifications" within the INTRODUCTION portion of this publication for further information on cylinder and group identification.

Bits 7 through 4 are gated into the Head Register. The head select bits are transferred to the bulk memory unit (disk units only, drum heads are selected by "group select" information) during SS5.

Bits 3 through 0 are gated into the Sector Register and are used for comparison during SS7 to determine when the desired starting sector address has arrived under the read/write head.



F1GINC = GICC = $\overline{QSS5} \cdot N1GTC3$

$\overline{F1GINC}$ = GSTB - $\overline{\emptyset STP} \cdot GBSY \cdot GCK1$

D0GCD1,2 = GPSD = GD1C = GROC · GDOC · $\overline{GCAB}$ · G1GREQ

F1GREQ = $\overline{PALM} \cdot GBSY \cdot G1GREQ \cdot GROC\downarrow$

$\overline{F1GREQ}$ = F0GDRY

F1GDRY = GDR1↓

$\overline{F1GDRY}$ = G0GTC2

F1GTC1 = $\overline{QSS7} \cdot \overline{QSS6} \cdot GTCC\downarrow$

$\overline{F1GTC1}$ = F1GTC1 · GTCC↓

F1GTC2 = F1GTC1 · GCK2↓

$\overline{F1GTC2}$ = $\overline{F1GTC1}$ · GCK2↓

G0GTC1 = F1GTC1 · $\overline{F1GTC2}$

G0GTC2 = F1GTC1 · GCK1

G0GTC3 = F1GTC1 · F1GTC2 · GCK2

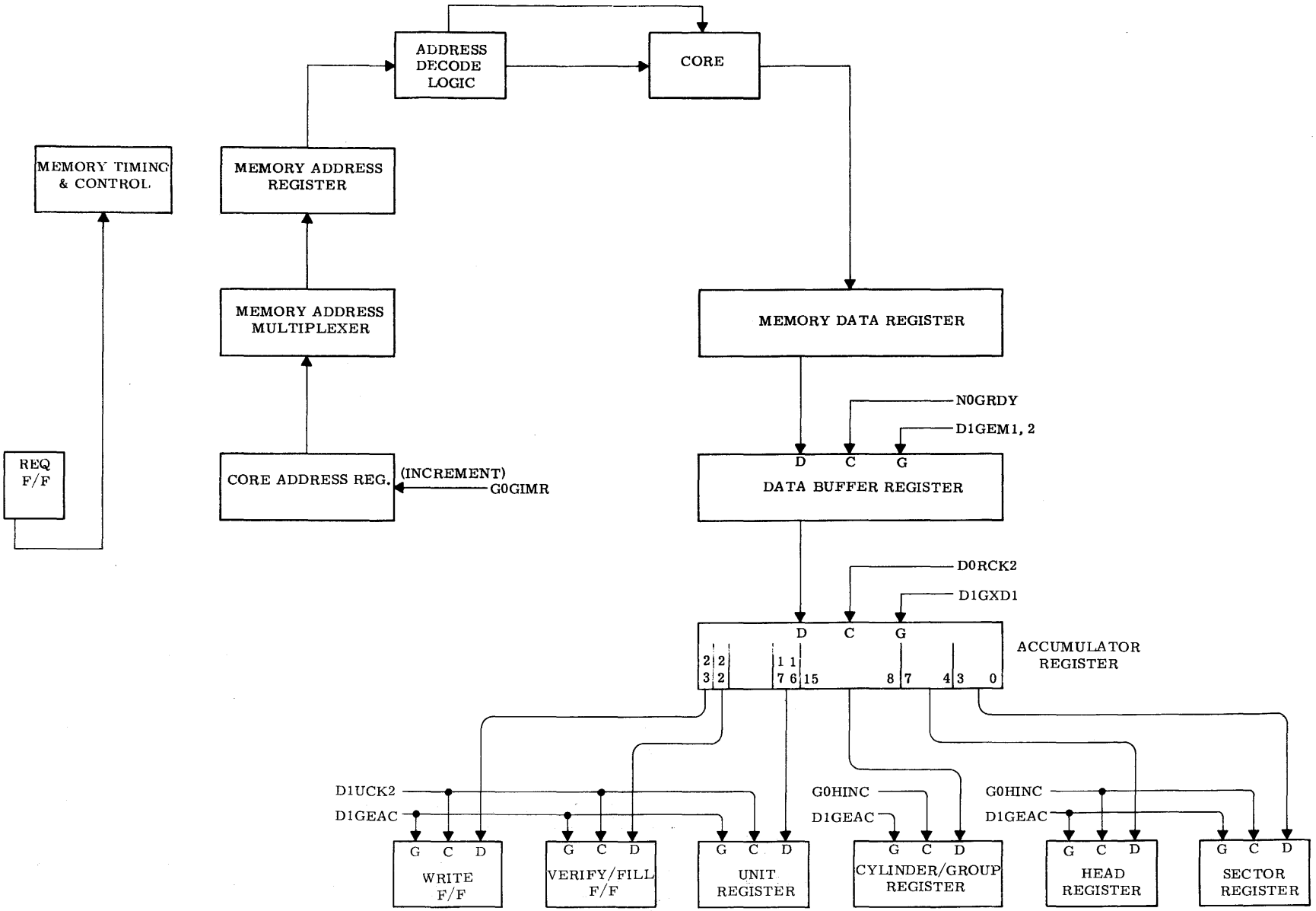
G0GIMR = QSS2 · N1GTC1

D1GEAC = G0GEAC - QSS2 · N1GTC2

D1GXD1,2 = G0GTC1

D1GEM1,2 = G0GEMD - N1GREQ

Fig. OUT 2.1 Timing Diagram



NOTE: The "G", "C", and "D" labeled inputs to the registers and flip-flops refer to "Gate", "Clock", and "Data", respectively.

Fig. OUT 2.2 Block Diagram - SS2

BLANK

OUT COMMAND - SS3

During Sequence State 3 (SS3) time, the controller requests and receives Control Word 2 (CW2) from the Core Memory. CW2 contains the block count for the data transfer operation. The block count specifies how many words (in 64-word increments) are to be transferred during the operation. This information is stored in the Block Count Register for use in determining when to terminate the operation.

SEQUENCE OF EVENTS

Essentially the same sequence occurs during the transfer of CW2 as took place when CW1 was "fetched" from Core Memory. The singular exception is that the complement of CW2 is gated, in part, to the Block Length Register from the "A" register. Accessing of CW2 from Core Memory is the same as CW1, except for the core address, and the gating to the "D" and "A" registers in the controller is identical.

The Core Address Register is incremented by one in the same manner as during SS2 and now contains the core address of CW3, which will be accessed during SS4. F1GINC is set and cleared in the same manner as during SS2, thereby advancing the Sequence Counter to SS4.

SIGNIFICANCE OF CW2

The contents of CW2 will determine how many sectors (each containing 64 24-bit words of data) are to be

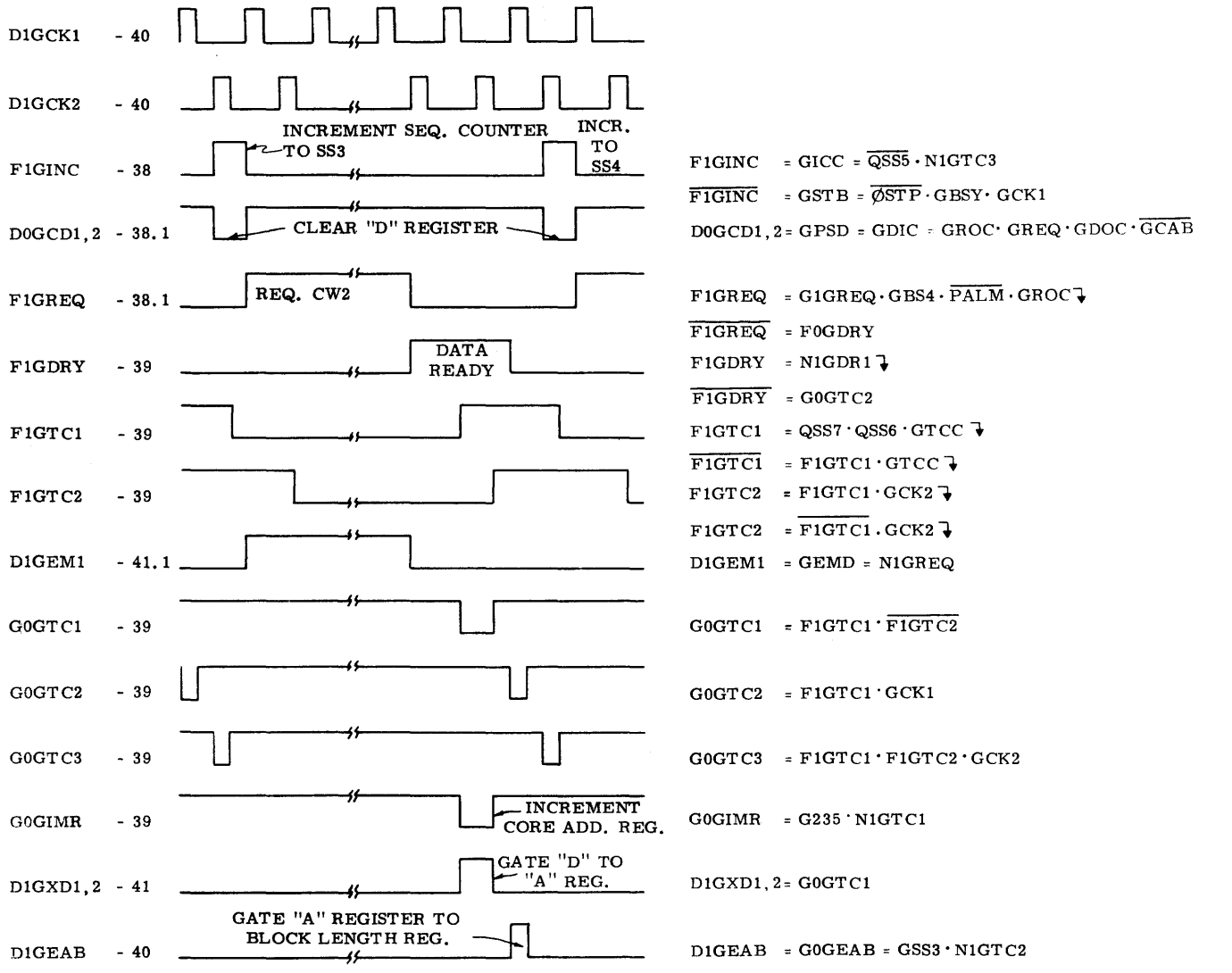
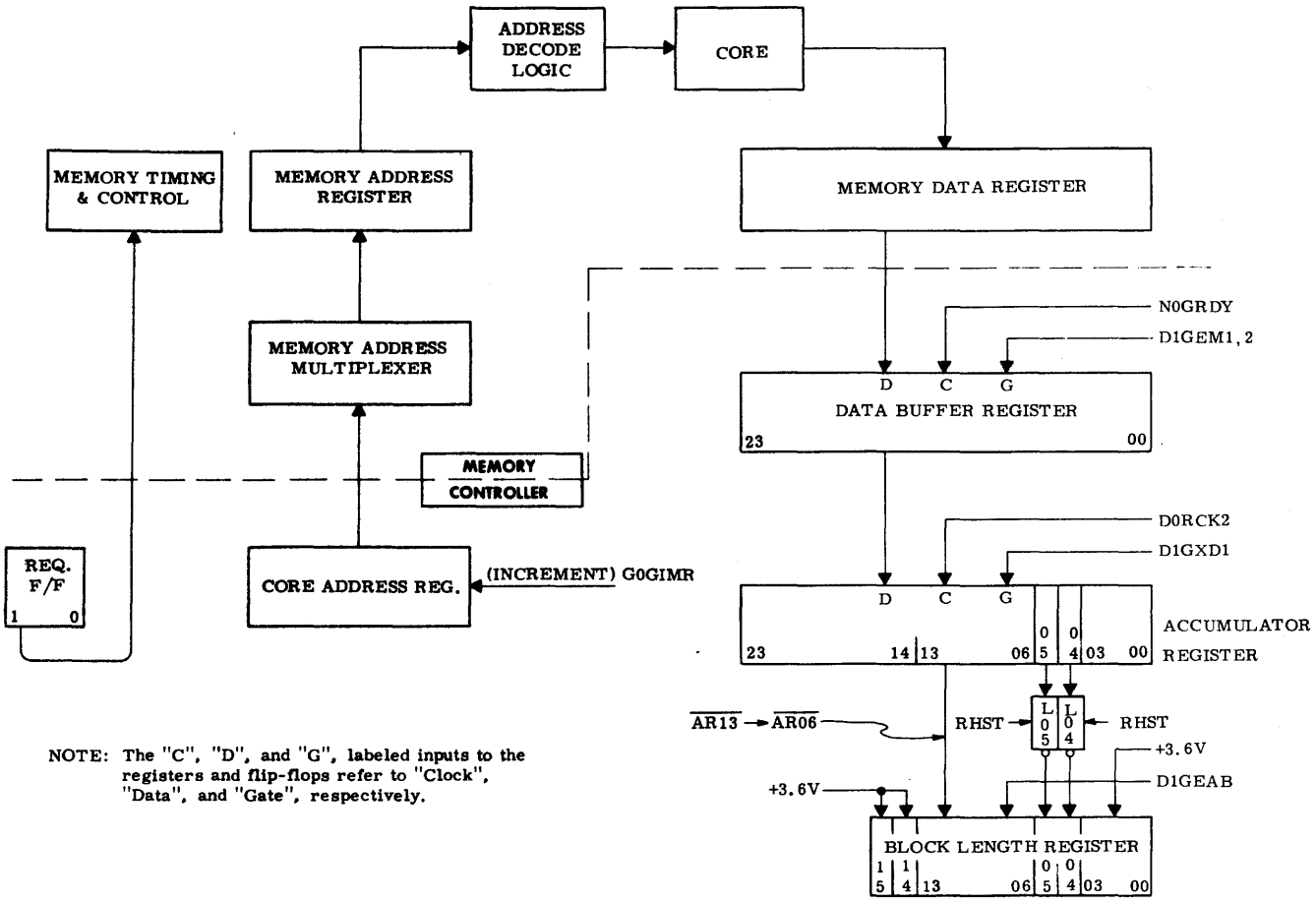


Fig. OUT 3.1 Timing Diagram - SS3

transferred. This is accomplished by taking the "one's" complement of the word count and incrementing it once for each data transfer that takes place until the Block Length Register contains all "ones".

The Block Length Register contains 16 bits, with Bit 15 being the most significant and Bit 00 being the least significant. During this sequence state the "one's" complement of bits 13 through 06 of the "A" register are transferred to the corresponding bit positions of the Block Length Register. Bits 15, 14, and 05 through 00 are automatically set to "ones" at the same time. Special gating is employed for bits 05 and 04 so that their complement can be transferred from the "A" register to the Block Length Register in the Test Mode of Write Header operations. However, they also are preset to "ones" during this type of operation.

Presetting bits 05 through 00 of the Block Length Register to "ones", which is equivalent to 64 binary counts, ensures that only full sector transfers take place. This means that the actual count transferred to the Block Length Register, represented by bits 13 through 06, indicates the sector (or block) count. The 64 counts of bits 05 through 00 must be fully incremented for each sector before Bit 06, the first bit actually representing the block count, is affected by the count.



NOTE: The "C", "D", and "G", labeled inputs to the registers and flip-flops refer to "Clock", "Data", and "Gate", respectively.

Fig. OUT 3.2 Block Diagram - SS3

OUT COMMAND - SS4

During Sequence State 4 (SS4) time, the controller requests and receives Control Word 3 (CW3) from the Core Memory. CW3 contains the starting core address for the data transfers that are to take place.

The timing of events for SS4 is shown in Fig. 4.1. Fig. 4.2 illustrates the general logic involved in requesting CW3 from Core Memory and transferring the control word through the controller to its Core Address Register.

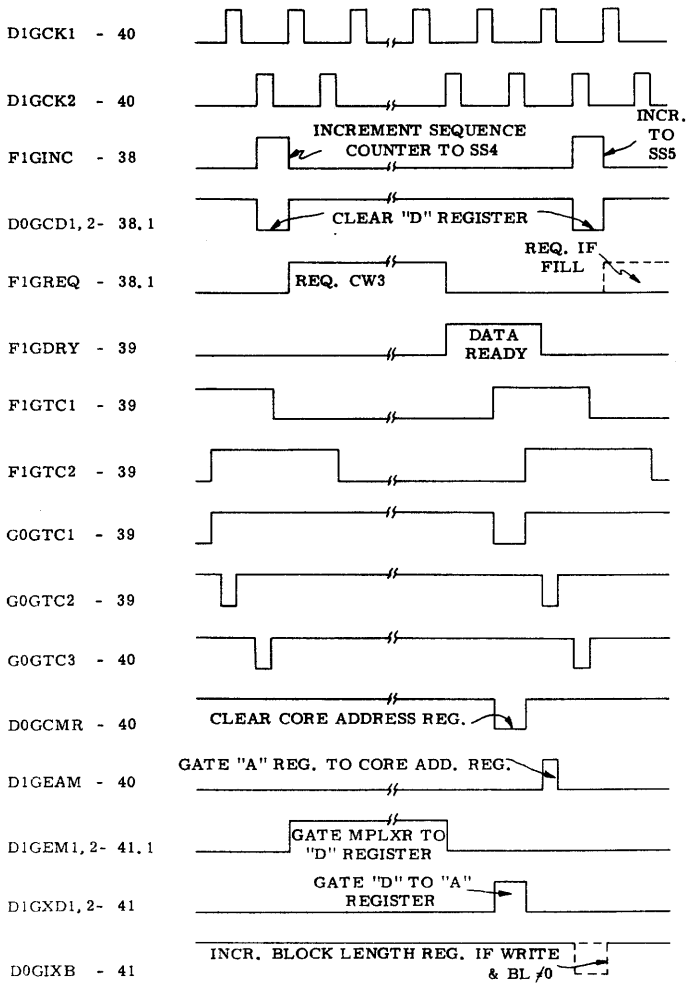
SEQUENCE OF EVENTS

CW3 is requested from Core Memory and, upon becoming ready, is gated to the Data Buffer (D) Register in the Bulk Memory Controller. The 25 bits (24 data bits and one parity bit) are parallel transferred from the "D" register to the Accumulator (A) Register. Bits 15 through 00 of the "A" register are then parallel transferred to the controller's Core Address Register. Bit 24 of the "A" register is gated to the Parity Tree logic and sampled along with the other 24 bits from the "A" register. Detection of a parity error (indicative of an invalid data transfer) would terminate the controller operation at this point. The absence of a parity error implies a successful transfer and the controller's Sequence Counter is incremented once to place the controller in Sequence State 5.

The request and transfer of CW3 is identical to the CW1 and CW2 transfers that took place in SS2 and SS3, except for the differing core address from which the control word was accessed and the differing final destination of the control word. In this sequence state the final destination of the control word's information is the controller's Core Address Register. This register is "cleared" during G0GTC1 time, prior to the DC gating of bits 15 through 00 from the "A" register to the "set" side of the Core Address Register's flip-flops; the actual gating is enabled by signal D1GEAM during G0GTC2 time.

SIGNIFICANCE OF CW3

The portion of CW3 that is loaded into the Core Address Register of the controller during SS4 is the starting core address for information transfers. If the controller operation (specified by bit 23 of CW1) is a "write", the first data word accessed from core to be written on the bulk device will be read from this core address. Conversely, if the operation is a "read", the first data word read from the device will be written into this core address.



F1GINC = GICC = GTC3 · $\overline{QSS5}$
$\overline{F1GINC}$ = G0GSTB = ØSTP · GBSY · GCK1
D0GCD1,2 = G1GPSD = GDIC = GROC · GDOC · $\overline{GCAB}$ · G1GREQ

F1GREQ = G1GREQ · GBSY · $\overline{PALM}$ · GROC ↓
$\overline{F1GREQ}$ = F0GDRY
F1GDRY = $\overline{F1GDRY}$ · GDR1 ↓
$\overline{F1GDRY}$ = G0GTC2
F1GTC1 = $\overline{QSS6}$ · $\overline{QSS7}$ · GTCC ↓
$\overline{F1GTC1}$ = F1GTC1 · GTCC ↓
F1GTC2 = F1GTC1 · GCK2 ↓
$\overline{F1GTC2}$ = $\overline{F1GTC1}$ · GCK2 ↓
G0GTC1 = F1GTC1 · $\overline{F1GTC2}$

G0GTC2 = F1GTC1 · GCK1

G0GTC3 = F1GTC1 · F1GTC2 · GCK2

D0GCMR = G1GCMR = G0GCLM = N1GTC1 · QSS4

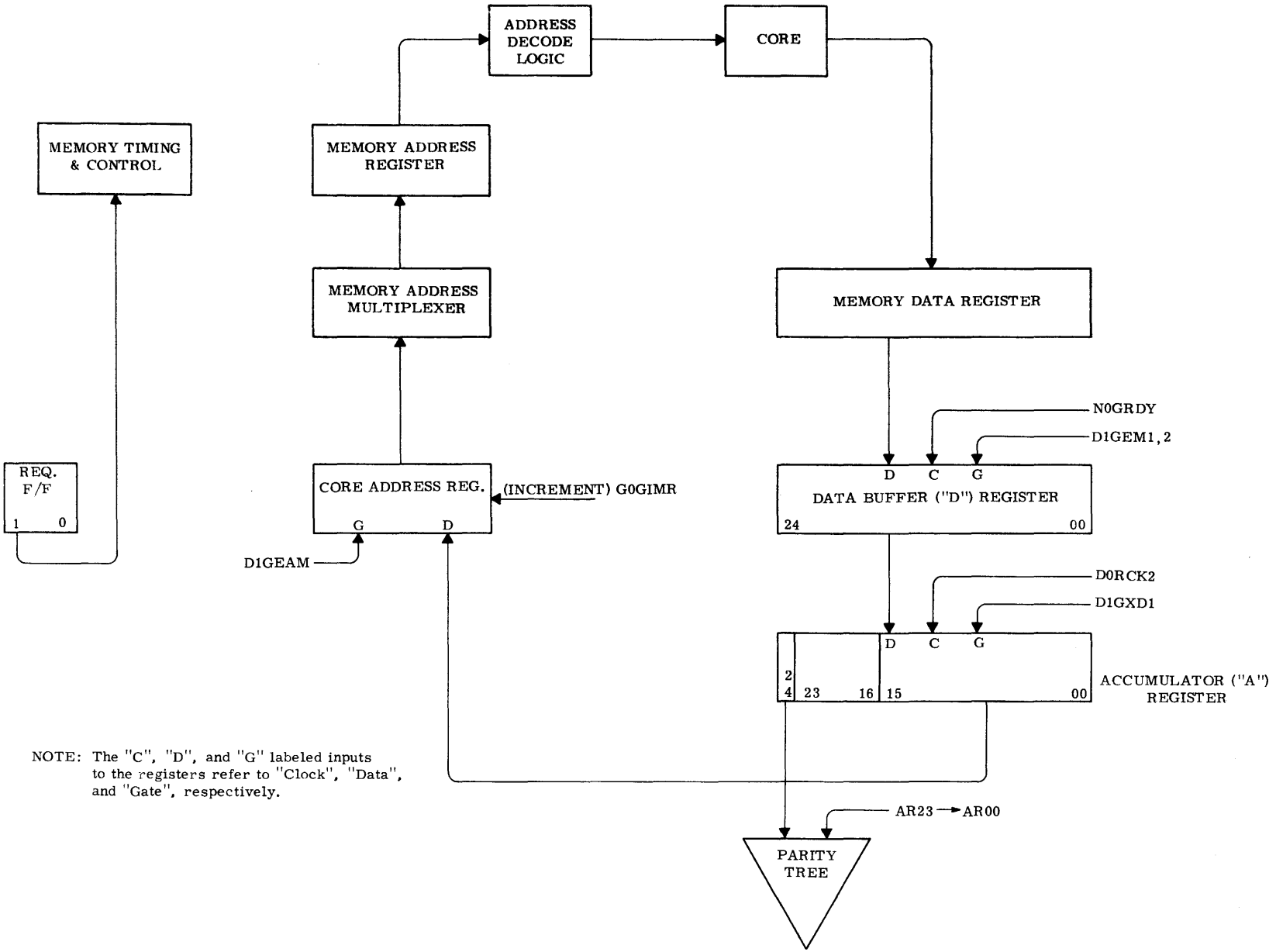
D1GEAM = G0GEAM = QSS4 · N1GTC2

D1GEM1,2 = G0GEMD = N1GREQ

D1GXD1,2 = G0GTC1

D0GIXB = GIBL = GWI3 = $\overline{GBLO}$ · QSS3 · GINC · UWRT

Fig. OUT 4.1 Timing Diagram - SS4



NOTE: The "C", "D", and "G" labeled inputs to the registers refer to "Clock", "Data", and "Gate", respectively.

Fig. OUT 4.2 Block Diagram - SS4

OUT COMMAND - SS5

This sequence state is used to transfer the necessary address information to the selected bulk device. For drum memory units, this transfer is only necessary once - during the initial sequence of the OUT command. However, disk units have movable heads and require a "cylinder advance" when the data transfer operation extends beyond the addressing limitations of the current cylinder. This means that the heads must be moved to the next sequential cylinder, and the new address information must be calculated and transferred to the disk unit. When this is necessary, the controller is returned from SS7 to SS4 for several clock pulses and then advanced to SS5 for address information transfers to the disk.

If CW2 contained a block count of zero, the controller sequence would terminate at the end of this sequence state. The necessary information would have been transferred to the bulk device and it would perform a "seek-only" operation, generating a seek complete signal when it reached the desired cylinder address. This is only meaningful for disk units which must seek a desired cylinder. The drum unit's heads are fixed and simply require the necessary time to select the desired head. However, if a seek-only operation is directed to a drum unit, it will generate the seek complete a few microseconds after receiving the Group Select information.

BULK ADDRESS INFORMATION TRANSFERS

Transfers of address and control information between the controller and the bulk devices during SS5 are accomplished using bi-directional bus lines. The information is sent in a series of parallel transfers over a group of eight bus lines connecting the controller and the bulk memory devices. There is also a group of "Tag Control" lines that are enabled in sequence to specify the significance on the information currently on the bus lines; this enables the bulk memory device to direct the information to the proper location.

Except for the "Read Cylinder/Group Select", all transfers during SS5 are from the controller to the device. The sequence of transfers, shown in Fig. OUT 5.1, is as follows:

1. Read Cylinder/Group Select - The controller transmits a "Read Cylinder/Group Select" signal to the controller's bulk memory units. If the selected Bulk Memory Unit (BMU) is a disk unit, the signal is used by the device to enable the contents of its Cylinder Address Register to be sent back to the controller. The controller compares the address with

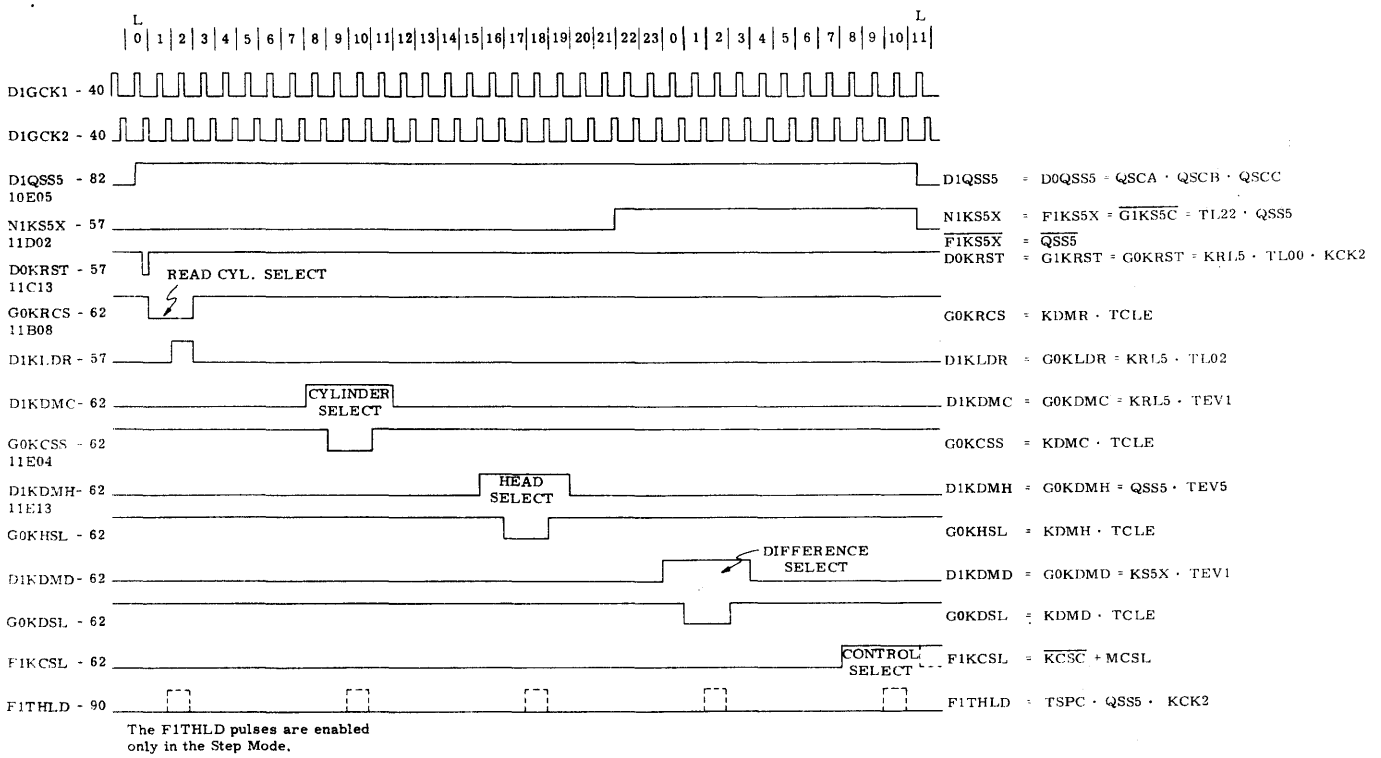


Fig. OUT 5.1 Timing Diagram - SS5

its desired cylinder address and computes the difference. The difference is returned to the disk (see transfer number 4 in this sequence) to be used as a count when moving the disk unit's heads.

If the selected BMU is a drum unit, the signal is merely used to "clear" its Head Register.

2. Cylinder/Group Select - The controller transmits a "Cylinder/Group Select" signal to the controller's bulk memory units. During this time the controller gates the contents of its Cylinder/Group Register onto the control bus lines. If the selected BMU is a disk unit, the information is gated into the Cylinder Address Register. When the selected BMU is a drum unit, bits 00 through 04 of the controller's Cylinder/Group Register are gated to bits 04 through 08 of the drum unit's Head Address Register.

   The disk unit stores the information so that the controller can determine where the heads are positioned after a "seek" operation. The drum unit uses the information as part of its head selection data.

3. Head Select - The controller transmits this signal to the controller's bulk memory unit during the time that it places the head select information on the bus lines. If the selected Bulk Memory Unit (BMU) is a disk unit, the information is loaded into the selected unit's Head Address Register. When the selected BMU is a drum unit, the information is joined with bits transferred during the "Cylinder/Group Select" and used in selecting the desired starting drum head. (The number of drum heads is dependent upon its storage capacity.)

4. Difference Select - The controller transmits this signal to the controller's bulk memory during the time that it places the difference information on the bus lines. If the selected Bulk Memory Unit (BMU) is a disk unit, the information is gated into a counter in the disk unit to be used for controlling the amount of head movement. When the selected BMU is a drum unit, the information is not used.

5. Control Select - The controller transmits this signal to the controller's bulk memory during the time that it places the control select information on the bus lines. If the selected Bulk Memory Unit (BMU) is a drum, only the "Write", "Read", "RTZ", and "Head Advance" signals are accepted. Disk Units utilize any of the eight bus lines' information.

If the operation continues past this sequence state (BL≠0), the control select signal remains enabled.

ADDRESS & CONTROL	TAG LINE FUNCTIONS				
BUS	READ CYL. OR GROUP SELECT	DIFF. SELECT	CYL. OR GROUP SELECT	HEAD SELECT	CONT. SELECT
BIT 0	1	1	1	1	WRITE GATE
BIT 1	2	2	2	2	READ GATE
BIT 2	4	4	4	4	SEEK FORWARD
BIT 3	8	8	8	8	RESET HD. REG.
BIT 4	16	16	16	NOT USED	ERASE GATE
BIT 5	32	32	32	NOT USED	SEEK REVERSE
BIT 6	64	64	64	NOT USED	RETURN 000
BIT 7	128	128	128	NOT USED	HEAD ADV.

Fig. OUT 5.2 Address and Control Select Information

OUT COMMAND - SS6

Sequence State 6 (SS6) is merely a period of time, following the control and address information transfers of SS5, when the controller awaits an "on-cylinder" signal from the selected Bulk Memory Unit (BMU).

If the selected device is a disk unit, flip-flop F1GDSK sets during this sequence state. This causes the controller's clock pulses to switch from a 6.96 MHz. rate to a 5.0 MHz. rate. If the selected device is a drum unit, the clock rate remains at 6.96 MHz.

The time delay of SS6 is dependent upon the amount of carriage movement that is necessary. If no movement is required (i.e., the disk heads are already at the desired cylinder or the selected BMU is a drum unit), the "on-cylinder" signal is enabled at L15 when the controller enters SS6. This would enable the controller to increment to SS7 on the following clock pulse. When the selected BMU is a disk unit and carriage movement is necessary, there could be a delay of up to 165 milliseconds before the controller steps to SS7.

BLANK

OUT COMMAND - SS7

It is possible to perform any one of five different types of operations during Sequence State 7. The one that is to be performed is determined earlier in the controller's sequencing for the operation. Four of the operations (Read, Verify, Write, or Fill) are specified solely by the status of bits 23 and 22 of Control Word 1, accessed from Core Memory during Sequence State 2. The remaining operation, (Write Header) is determined by the enabling of the controller console's "Test Mode" and "Write Header" switches, and the existence of a "write" bit (Bit 23 equal to a "one") in Control Word 1.

A Write Header operation establishes the non-data information within the sectors. (See Fig. WHD 7.1, under "WRITE HEADERS - SS7".) The Write/Fill and Read/Verify operations record and playback, respectively, the data portion of the sectors. A separate description is provided, for the Sequence State 7 portion of the Write Header operation, following this description of the Sequence State 7 portion of Read, Write, Verify, and Fill operations. Because of their similarity, the Read and Verify operations are described together, as are the Write and Fill operations. Prior to their descriptions, however, is that of the address comparison which must take place before any data transfer operation begins.

ADDRESS COMPARISON

Upon entering Sequence State 7, the bulk unit's selected head is resting over the desired track, which contains 16_{10} sectors. However, the data transfer is specified to begin at one of those sectors. Therefore, each sector must be checked by address comparison to determine if it is the correct starting sector before initiating the data transfer.

Each sector is checked by reading its address into the controller and comparing it against the desired address stored in the Cylinder/Group, Head Address, and Sector Address registers. The read is initiated by enabling Tag Control Line 1 - sent to all Bulk Memory Units (BMU's), but enabled only in the selected one. Tag Control Line 1, G0LTG1, is enabled by D0WXTL, and F1KCSL, which has been "true" since Sequence State 5. D0WXTL, the inverted output of the Crystal Clock flip-flop, is at +3.6V since the flip-flop was cleared by the receipt of a sector marker.

Clock Generator

The data are gated into the controller's read-recovery circuitry, which generates its own clock pulses from the data it receives (see Fig. OUT 7.1). Since F1WXTL is cleared, the read-recovery clock pulses replace those from the controller's crystal clock generator in enabling the G0RCK2 pulses. The D0RCK1 pulses are disabled at the clearing of F1WXTL, but D0RCK2 clocks from the controller's clock generator continue to be decoded - primarily for the error detection and sector marker generation circuitry.

Gap and Synchronization

The first information to appear following the sector marker is a gap of 72 "one" bits. F1Z0CT, the Zero Count flip-flop, sets on the first clock pulse generated

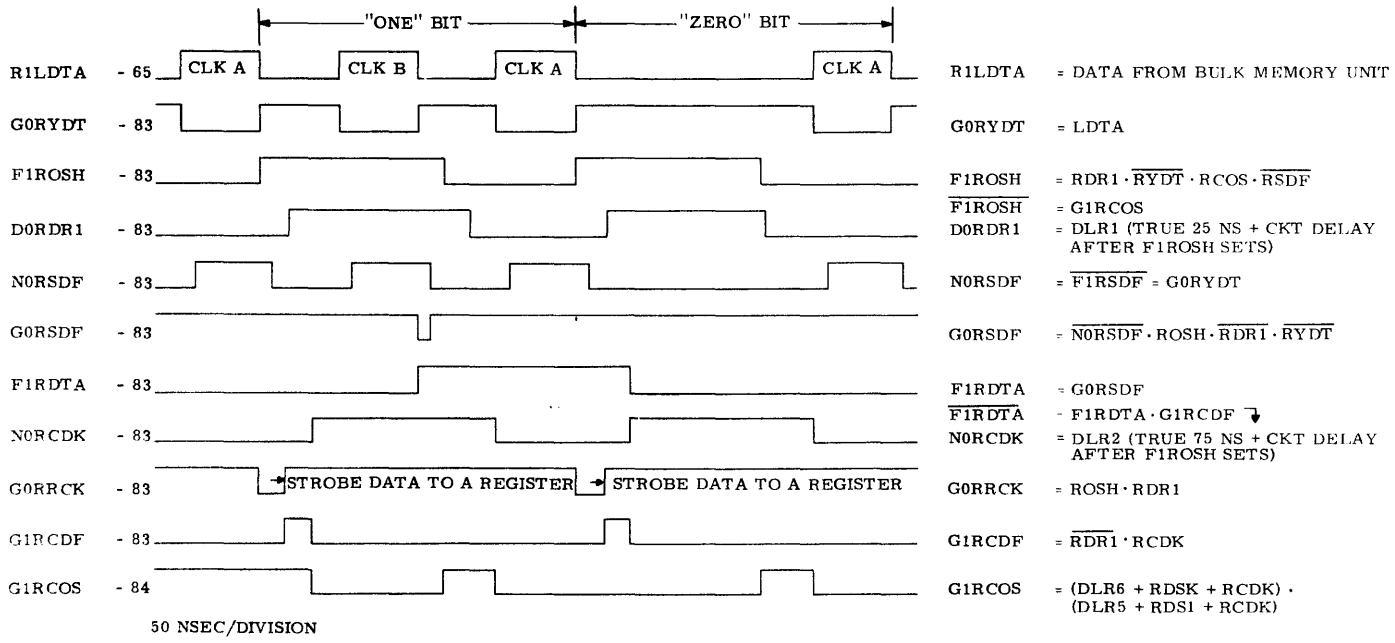
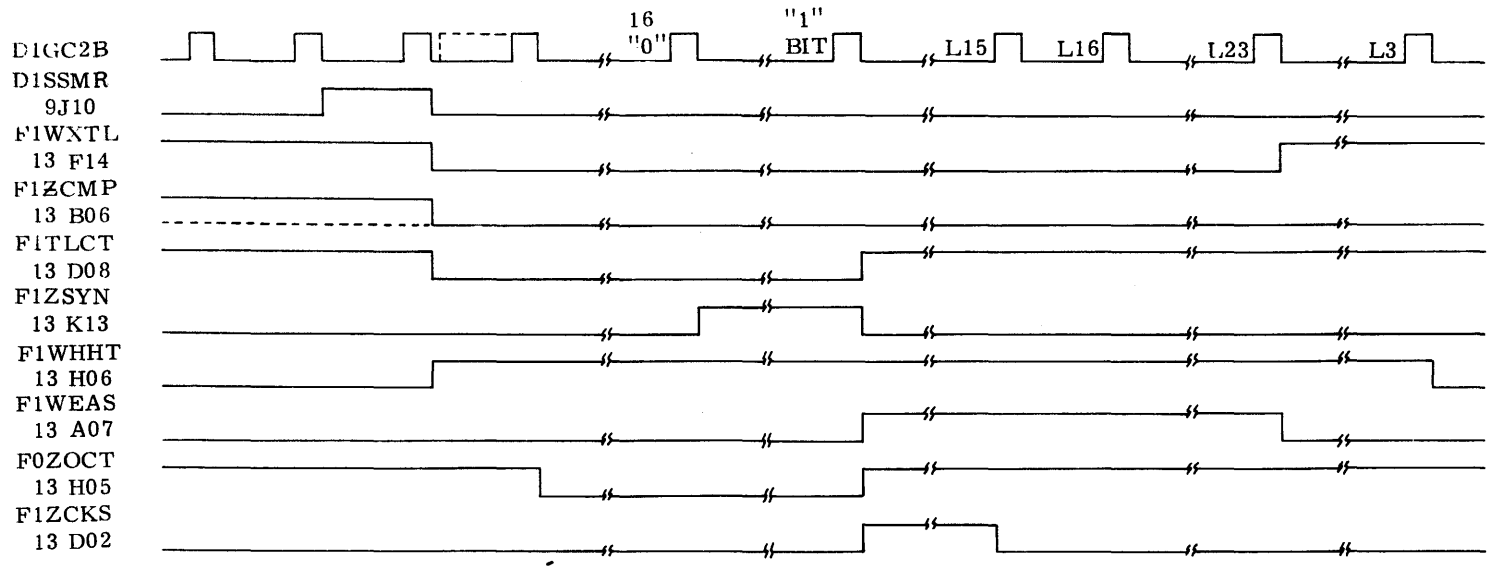
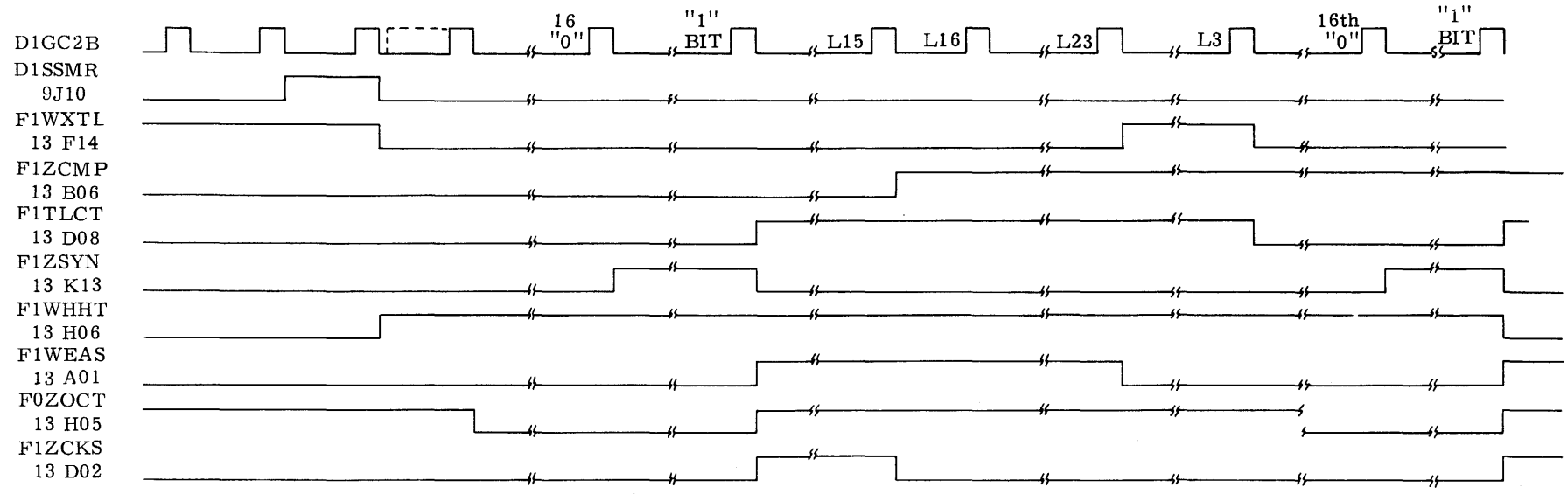


Fig. OUT 7.1 Data Recovery



a. Read Header/No Compare



b. Read Header/Compare

* TIMING THROUGH L3 IS THE SAME FOR READ OR WRITE OPERATIONS.

Fig. OUT 7.2 Address Comparison

from this data. During the period that F1Z0CT remains set, each zero bit received from the BMU enables gate G0ZZCT.

Twenty-three "zero" bits follow the 72 "one" bits of the gap. Each of these "zeros", by enabling G0ZZCT and consequently G1ZC00, permits the Zero Count Accumulator to increment once at the trailing edge of the clock pulse for each data period that contains a "zero". On the 16th "zero", the Sync flip-flop, F1ZSYN, sets.

A "one" bit follows the 23 "zeros" and enables the clearing of F1ZSYN. At the same time, F1TLCT, the L Count flip-flop, F1WEAS, the Enable A Shift flip-flop, and F1ZCKS, the Check-Sum flip-flop, set. They enable the controller to gate the next data word (address and address check-sum bits) into the Accumulator register and to count the "one" bits of the address portion in the Check-Sum register.

With the setting of the L Count flip-flop, F1TLCT, the L Counter increments once for each clock pulse. The increment periods are decoded as L pulses, ranging from L00 through L23. Incrementing past L23 returns the count to L00.

During the L00 through L23 periods of the read portion, data are gated serially into the controller's Accumulator register at the most significant end (Bit 23). From L00 through L15 of this Address and Address Check-Sum word time, the number of "one" bits contained in the data are counted in the Check-Sum register. This count is later compared with the check-sum gated into the Accumulator register during L16 through L23.

The address portion of the word is checked at the end of L16. At this time, Bit 00 of the Address and Address Check-Sum word is residing in Bit 12 of the Accumulator register and is about to be shifted to Bit 11 of the same register. If there is address comparison, the Compare flip-flop , F1ZCMP, sets at this time and the transfer operation takes place during this sector. Should there not be address comparison, the Crystal Clock flip-flop, F1WXTL, does not clear on the L03 of the following word time. The read-recovery circuitry would then be disabled until the next sector marker when F1WXTL again clears, enabling the read-recovery circuitry to again seek address comparison. (Refer to OUT 7.2 for Address Comparison timing diagram.)

WRITE OPERATION

Write operations, specified by bit 23 of Control Word 1 equal to a "one", are used to record data on the Bulk Memory Unit (BMU). The number of 24-bit data words to be written is specified by the count transferred to the controller, via Control Word 2 (CW2), during Sequence State 3 (SS3). Data is normally accessed from sequential Core Memory addresses for a write operation, beginning at the address specified by the contents of CW3.

An alternate method of writing data, referred to as a "Fill" operation, is also available. This type of write operation is specified when both bit 23 and 22 of CW1 are equal to a "one". The number of data transfers is still specified by the contents of CW2. However, the data is not accessed from Core Memory as during a "normal" write, but is accessed from the controller's Data Buffer Register which contains the contents of CW3. Thus, the specified number of sectors are filled with the data pattern from CW3 and no memory requests are made during the data transfer period.

Due to the otherwise identical aspects of operation, both the "normal" and "Fill" write operations are discussed together in the following description, which picks up at the point following address comparison.

Write Gap and Sync Word

Following address comparison the controller writes a gap of "one" bits, followed by a sync word containing 23 "zeros" and a "one" bit. This permits subsequent read operations for this sector to resynchronize the read-recovery circuitry in the interval between the reading of the Address and Address Check-Sum word, and the start of the data read. This is necessary since they were written during two unique write operations; the Address and Address Check-Sum word was written during a "Write Header" operation, and the data is written during a normal "write" operation.

The "one" bits of the gap and the one "one" bit of the sync word are written under the control of the Write 1's flip-flop, F1WWR1. F1WWR1 sets on the trailing edge of the L22 pulse for the Address and Address Check-Sum word. It remains set during the writing of the gap's "one" bits, controlled by the count sequence of flip-flops F1WRT1 and F1WRT2 which time the gap period. At the end of the gap, with F1WRT1 and F1WRT2 set, F1WRT3 sets and F1WWR1 clears. F1WWRT, the Write 1's flip-flop, sets again during L23 time of the sync word to enable the writing of a "one" bit, which follows 23 "zeros" in the sync word.

The gating of the gap and sync word begins with the setting of the Write Enable flip-flop, F1WWRE, at the end of the L03 following the Address and Address Check-Sum word. If an error had been detected in the check-sum, the Address Check-Sum Error flip-flop, F1EACS, would have set at the end of the preceding L count, L02. This would have disabled Address Compare driver D1ZACP, and F1WWRE could not have set.

Since F1WWRE cannot set until the check-sum is verified, at L02, the first four clock periods (L00 through L03) of the gap are not recorded during a normal write operation. However, this area would have been filled with "ones" during the "Write Header" operation that established the header information for the sector. The controller begins recording "ones" in the gap at L04 of the word-time immediately following the Address and Address Check-Sum word-time. The re-

maining portion of that word-time and the two subsequent word-times are filled with "one" bits to fill in the gap. This is followed by the sync word. Following this, the controller begins gating actual data to be written on the Bulk Memory Unit (BMU).

During the gap and sync word period, the information written on the BMU is determined by the status of the Write 1's flip-flop, F1WWR1. With F1WWR1 and the Write Enable flip-flop, F1WWRE, set, both Clock 2's and Clock 1's are gated through G0WDTA to be recorded on the BMU's recording medium. Each bit period (Clock 2 to Clock 2) that also contains a Clock 1 is regarded as a "one". If there are no Clock 1's during the intervals, as during L00 through L22 of the sync word-time when F1WWR1 is cleared, then the bit periods are regarded as "zeros".

Write Data

The information recorded in the gap and sync word was governed by logic within the controller, whereas the 24-bit data words are requested sequentially, as needed, from the Core Memory (or the Data Buffer Register during "Fill" operation). The data words are received from memory and gated through the controller's Data Buffer register to the Accumulator register by parallel transfers. Parity is checked on the information as it is received in the Accumulator register.

The first data word to be written on the BMU was requested during Sequence State 5 (SS5), and will be present in the Accumulator register when F1WRSA sets at the end of the sync word write. F1WRSA, set, enables the contents of the Accumulator register to shift serially toward the least significant end (Bit 00), one bit per L count time. As it shifts, the contents of Bit 00 is examined each L count time by gate G0WDTA and the status of Bit 00 determines what bit status (one or zero) will be written on the BMU for that clock period.

At each L01 period the Request flip-flop, F1GREQ, sets. (The request is blocked by G1GREQ during a "Fill" operation.) This requests a new data word from Core Memory as the current word is being shifted from the Accumulator register. When the requested word becomes available from memory, a ready pulse is sent to the controller. The trailing edge of this pulse triggers the setting of the Data Ready flip-flop, F1GDRY, and clocks the data from memory into the controller's Data Buffer register. F1GREQ, the Request flip-flop, clears as a result of F1GRDY setting. At the end of L23, as the last bit is being serially shifted from the Accumulator register to the BMU, the contents of the Data Buffer register is transferred parallel to the Accumulator register, and begins shifting to the BMU. This continues for the length of the word count, as determined by the BL register.

The Block Length register is incremented at each L01 while data is being written on the BMU. (It was also incremented once during Sequence State 3, before the first data word was requested from memory.) Therefore, following the L01 at the start of the 63rd data transfer, the Block Length register will have incremented to a count of 77_8. This enables N1WB77, which ANDs with D1GXD2 at L23 to set the End of Sector flip-flop, F1WESR. On the following L23, the end of the 64th data transfer, the Shift A flip-flop, F1WSRA, clears. The Write Enable flip-flop, F1WWRE, remains enabled however, for the writing of the check-sum bits.

Write Postamble

A count was kept in the Check-Sum register of the number of "ones" transferred to the BMU during the

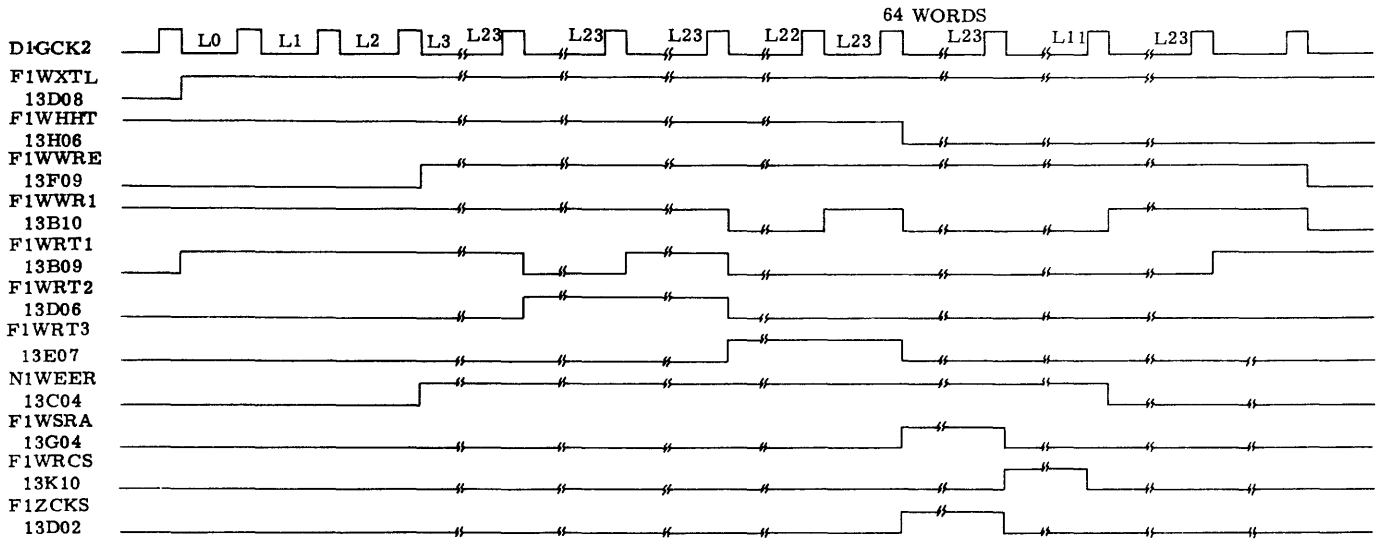
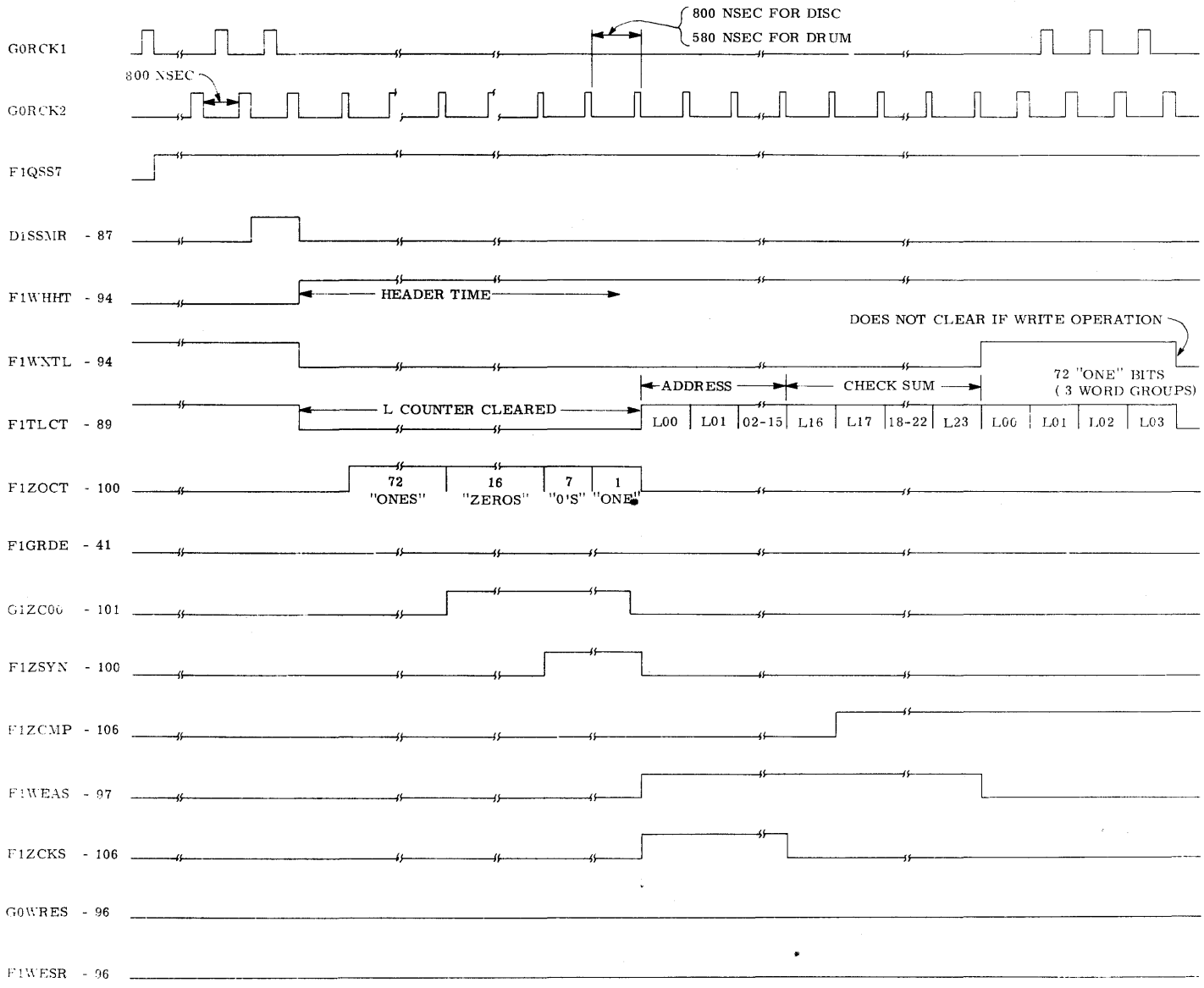


Fig. OUT 7.3 Write Timing Diagram


800 NSEC FOR DISC
580 NSEC FOR DRUM
800 NSEC
G0RCK1
G0RCK2
F1QSS7
D1SSMR - 87
F1WHHT - 94
HEADER TIME
DOES NOT CLEAR IF WRITE OPERATION
F1WXTL - 94
ADDRESS
CHECK SUM
72 "ONE" BITS
(3 WORD GROUPS)
L COUNTER CLEARED
F1TLCT - 89
L00
L01
02-15
L16
L17
18-22
L23
L00
L01
L02
L03
F1ZOCT - 100
72
"ONES"
16
"ZEROS"
7
"0'S"
1
"ONE"
F1GRDE - 41
G1ZC00 - 101
F1ZSYN - 100
F1ZCMP - 106
F1WEAS - 97
F1ZCKS - 106
G0WRES - 96
F1WESR - 96

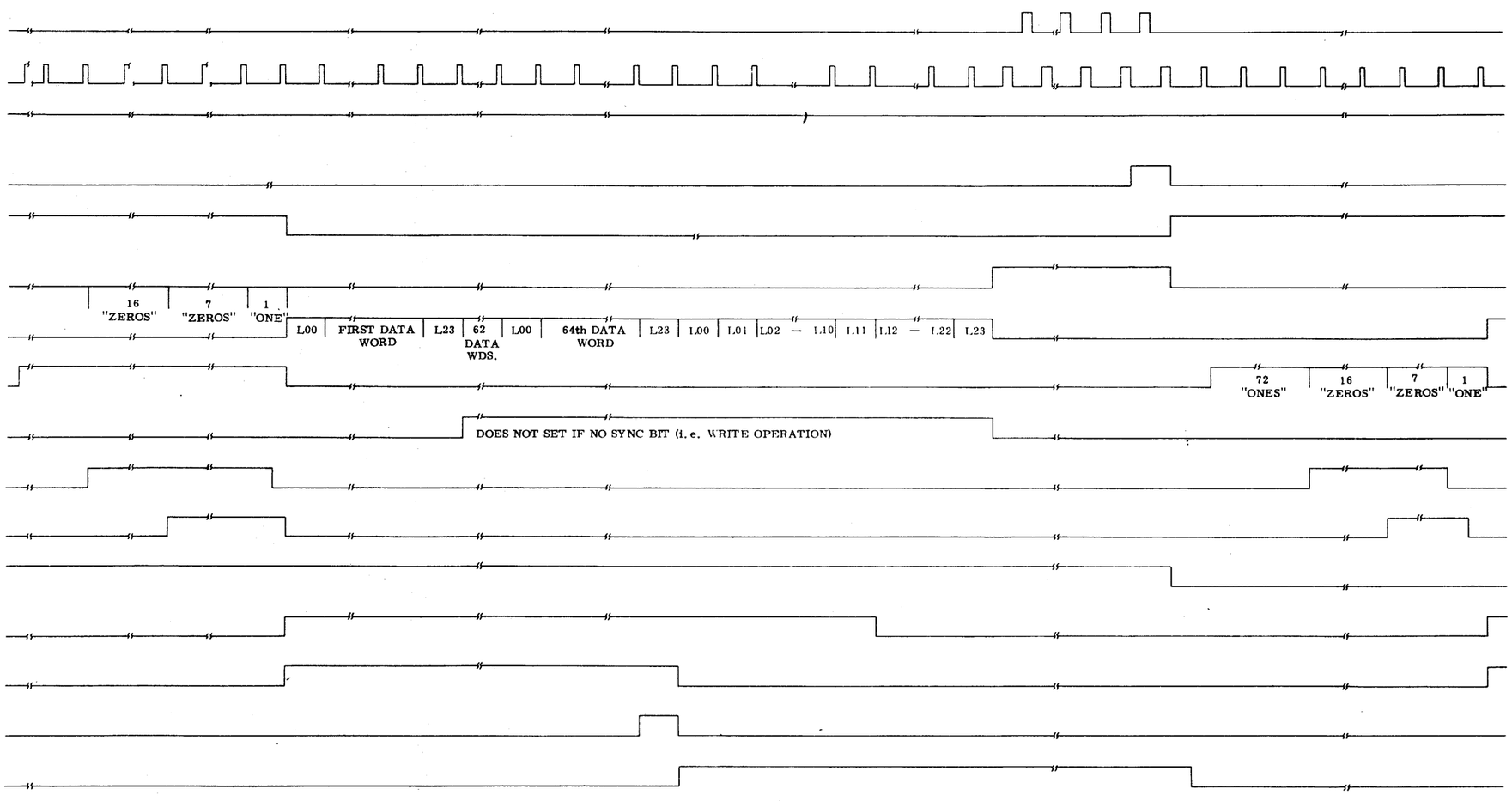


Fig. OUT 7.4 Read Operation (One Sector)

"Write Data" period. This count was recorded during the time that F1ZCKS, the Check-Sum flip-flop, was set, which corresponded to the "Write Data" period. Each "one" bit shifted from the Accumulator register during this time enabled G0ZCDS and, in turn G1ZC00. The Check-Sum counter was incremented during each L count of this period that G1ZC00 was "true", thus counting the number of "one" bits transferred to the BMU.

Upon completion of the "Write Data" period, the controller transfers this count to the BMU where it is recorded following the data. This provides a check-sum for use in validating the data during subsequent read operations. The recording of the check-sum occurs during the set period of F1WRCS, the Write Check-Sum flip-flop, "true" from L00 through L11 of the word-time following the "Write Data" period.

Immediately following the recording of the check-sum, the Write 1's flip-flop, F1WWR1, again sets. It remains set from L12 (of the check-sum word-time) through the following L00, enabling 13 "ones" to be recorded after the check-sum. These "ones" serve as a clock for subsequent reads, providing timing for checking the data check-sum after it has been read. Refer to Fig. OUT 7.3 for write timing diagram.

READ OPERATION

Read operations performed on the Bulk Memory Units (BMU's) may be either data reads or verify reads. Data reads result in the transfer of information from the BMU to Core Memory. The first transfer is stored in the core address specified by Control Word 3 (CW3), and each subsequent transfer is stored in a core address that is one higher than the last.

A verify read operation performs essentially the same sequence as the data read, but the data is not transferred to Core Memory. Verification is made by calculating a check-sum of the data read and comparing it against the recorded check-sum for each sector that is read following the actual data.

The sequence of operation discussed here begins at the point just following the reading of the Address and Address Check-Sum, discussed previously in the description for this sequence state. (Refer to Fig. OUT 7.4 for timing sequence.)

Following the reading of the Address and Address Check-Sum, the controller disables the BMU's read circuitry and returns the controller to its internal clock timing by setting F1WXTL, the Crystal Clock flip-flop, at the end of L23. F1WXTL clears again on the following L03 if there is no check-sum error from the reading of the address. This re-enables the BMU's read circuitry and the output data from the BMU again becomes the timing source for the controller.

F1TLCT clears on the same L03 that F1WXTL cleared, thereby disabling the L counter circuitry. The L counter remains cleared throughout the reading of the "Write" operation's Gap and Sync Word, allowing the controller to resynchronize its data recovery circuitry. This second synchronization, occurring just prior to the reading of the 64 data words, is for the information recorded by the actual "Write" operation. The initial read synchronization for the sector, which occurred prior to the reading of the address, was for information recorded during the "Write Header" operation. Synchronization is established in essentially the same manner in both cases. (See "Write Gap and Sync Word", located under WRITE OPERATION in this sequence state description.)

Following the reading of the pre-data Sync Word, F1TLCT again sets. This enables the generation of L count pulses for the reading of data. The L count pulses are incremented, at each CLK 2, from L00 through L23. L23 marks the end of a word time and the counter returns to L00 as the first data bit of the next word enters the controller.

F1WEAS also sets at the end of the pre-data Sync Word, permitting the data to shift serially into Bit 23 of the controller's Accumulator register under the timing of the L count pulses. The L count pulses are timed by clock pulses derived from the actual data being read back from the BMU.

F1GRDE (Read Envelope) sets at the end of L23, indicating the first word has been assembled in the Accumulator register. Drivers D1GXA1 and D1GXA2 are enabled during CLK 2 of the next L count (L00) to gate the data parallel from the Accumulator register to the Data Buffer register. At the same time, D1GXA2 enables the inputs to F1PD24 so that the calculated parity for the word can be established.

The actual request to Core Memory occurs with the setting of F1GREQ by the negative transition of signal G1GROC. G1GROC is enabled by D0GIXB, "true" during the CLK2 of each L01 while F1GRDE is set. (D0GIXB also causes the incrementing of the Block Length register.)

Upon becoming ready, the Core Memory returns a data ready pulse to the controller. The 24 data bits and the parity bit are then sent to Core Memory at the address specified by the controller's Core Address Register. Following the receipt of data ready, the Core Address Register is incremented one count by G0GIMR. The parity is checked for data validity. The data and a regenerated parity bit are then stored into core.

The same procedure is repeated for the remaining 63 data words of the sector. When the 64th data word has been gated into the controller from the BMU, F1WESR, the End of Sector flip-flop, sets. (F1WESR sets when bits 05 through 00 of the Block Length register are equal to a count of 62_{10}. This actually corresponds to the 64th data transfer since one count is lost because the counter starts incrementing from all "ones" instead of zero, and the other count is not recorded until the L01 following the 64th word of the sector.)

Data Check-Sum

A count was kept of the number of "one" bits shifted into the controller during the data read. This count, accumulated in the Check-Sum counter under the control of F1ZCKS, is compared with the recorded check-sum. The recorded data check-sum is read into the controller during the L00 through L11 period immediately following the reading of the 64th data word of the sector. The comparison is made during L count L12, just following the reading of the recorded check-sum. If there is not an identical correspondence, signal N1EDCS is enabled and the Data Check-Sum flip-flop, F1EDCS, sets on the trailing edge of the CLK2 pulse of L12. This, of course, would terminate the operation by clearing the controller's Busy flip-flop and the Central Processor would be informed of the error.

If there is correspondence of check-sums, the operation continues. F1WXTL would be set again at the end of the following L23, disabling the BMU's read circuitry and returning the controller to its internal clock timing, until the receipt of the next sector marker.

ADDRESS UPDATING

Address updating procedures differ somewhat as determined by the type of device (drum or disk) being accessed. Organization of data is not the same for both devices. (Refer to "Data Organization", under the heading PROGRAM CONTROL, located in the introductory portion of this publication for distinctions.)

Address updating is divided into "Head Update" and "Cylinder/Group Advance" descriptions in the following text.

Head Update

For both drum and disk units, data is divided into 16 sectors under each individual read/write head. Therefore, if the transfer operation is not complete (block length not equal to zero) following the data transfer from the 16th sector of a head, a head update is necessary.

Head update involves incrementing the controller's Head Register by one and transmitting an Advance Head Count signal to the selected Bulk Memory Unit (BMU). This signal causes the BMU's Head Register to be incremented by a count of one. Thus, both the controller and the BMU's Head Registers are incremented on a head update.

Cylinder/Group Update

Cylinders and groups refer to the organization of heads within a disk or drum, respectively. A disk is organized into cylinders containing 10 heads (addressed 00 through 11_8) each. Drum units are organized into groups, each containing 16 heads (addressed 00 through 17_8).

The starting cylinder (for disk units) or group (for drum units) address is loaded into the Cylinder/Group Register of the controller during Sequence State 2 (SS2). Updating of the Cylinder/Group Register and the corresponding register in the BMU is required when the addressing limit for a cylinder or group is exceeded during a data transfer operation. This happens at the end of the tenth head of a disk unit or the 16th head of a drum unit.

GROUP UPDATE

Updating the group address involves the incrementing of the Cylinder/Group Register in the controller and the Address Register in the selected drum unit by a count of one. Incrementing the Address Register in the drum unit, when the "head" portion (bits 03 through 00) contains a count of 17_8, automatically returns the "head" portion to zero and increments the "group" portion (bits 04 through 08) by a count of one. The controller remains in SS7 during the updating.

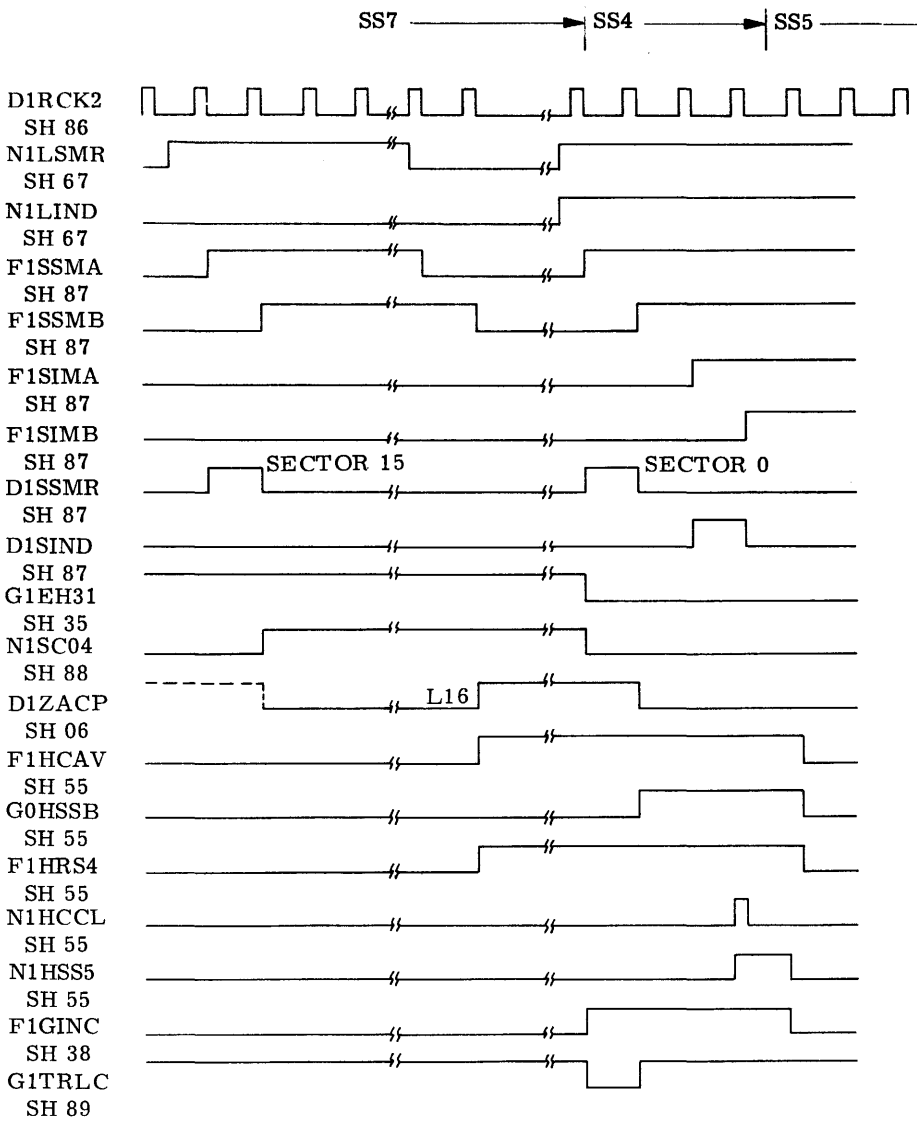
During the update, the controller's Cylinder/Group Register is incremented by the enabling of gating signal N1HCCL and clock signal G0HINC. N1HCCL is "true" because of the enabling of signal N1HC04 at gate G0HCCL when the controller's Head Address and Sector Address registers both contain a count of 17_8. D0HINC is enabled on the first CLK2 of the following sector marker.

Incrementing of the selected drum unit's Address Register is achieved by transmitting a Head Advance signal to the drum unit over Tag Control Line 7. Tag Control Line 7 is enabled during the set period of F1HHDV, "true" during a portion of the sector marker that follows the last sector of the final head for the group.

CYLINDER ADVANCE

Updating the cylinder address for a disk unit not only requires the incrementing of registers, but also involves the movement of the disk unit's heads to the next sequential cylinder. Fig. OUT 7.5 indicates the sequence involved. This requires the transfer of cylinder movement information, performed in SS5. The cylinder difference in this case would only be a count of one. (Refer to the description for OUT COMMAND - SS5 for the Cylinder Advance transfer sequence.)

HEAD UPDATE WITH CYLINDER ADVANCE - DISC - BLOCK LENGTH ≠ 0



N1LSMR = SECTOR MARKER FROM SELECTED DEVICE

N1LIND = INDEX MARKER FROM SELECTED DEVICE

F1SSMA = LSMR · CK2 ↓

$\overline{\text{F1SSMA}}$ = $\overline{\text{LSMR}}$ · CK2 ↓

F1SSMB = SSMA · CK2 ↓

$\overline{\text{F1SSMB}}$ = $\overline{\text{SSMA}}$ · CK2 ↓

F1SIMA = SSMB · LIND · CK2 ↓

F1SIMB = SIMA · CK2 ↓

D1SSMR = SSMA · $\overline{\text{SSMB}}$

D1SIND = SIMA · $\overline{\text{SIMB}}$

G1EH31 = HR00 · HR03 · GDSK

N1SC04 = SR03 · SR02 · SR01 · SR00

D1ZACP = $\overline{\text{EACS}}$ · ZCMP

F1HCAV = ZACP · SC04 · EH31 · CK2 ↓

$\overline{\text{F1HCAV}}$ = SIMB · CK2 ↓

F1HRS4 = SSMR · HCAV

F1HRS4 = SSMB · CK2 ↓

N1HCCL = HCAV

N1HSS5 = SIND · HCAV · CK2

F1GINC = GICC - HSS5 · GSS4

F1GINC = GSTB = $\overline{\text{ØSTP}}$ · GBSY · GCK1

G1TRLC = TLCT · $\overline{\text{QSS4}}$

G0HSSB = $\overline{\text{HOBL}}$. HRS4

Fig. OUT 7.5 Cylinder Advance Timing Diagram

WRITE HEADER - SS7

The Write Header operation is a "special case" OUT command, used to establish essential information (other than data) within the sectors of the Bulk Memory Units (BMU's). This is accomplished using the controller logic and information provided by the Write Header program. The pre-address sync bits, the address bits, and the address check-sum bits are provided by the program at times specified by the controller. The remainder of the sector, including the data area, is filled with "ones" by the controller logic.

While performing a Write Header operation, the "Test Mode" and "Header Mode" switches on the controller's maintenance panel must be placed in the enabled position. Enabling these switches informs the controller that the write operation is a "header" write instead of a "normal" write. Following the execution of a Write Header operation, the Write Header program initiates a Fill operation to establish pre-data sync bits (and also data check-sum bits) in the sector so that subsequent Read operations do not cause the controller to "hang-up" waiting for the detection of the pre-data sync bits.

Header information is recorded on a track-by-track basis (i.e., one Write Header operation records one track of information) for the purpose of restoring information that may have been damaged or destroyed.

Prior to the recording of the actual header information, the controller causes the entire track to be filled with "one" bits. Header information is then recorded on the second pass, with the controller requesting Core Memory twice each sector for the required header words. The two words requested during each sector are the Sync Word (23 "zeros" and a "one" bit), and the Address and Address Check-Sum Word. During this second pass, a gap of 72 "one" bits is also recorded prior to these two words in each sector.

The sector format, as it appears following a Write Header operation is shown in Fig. WHD 7.1. Fig. WHD 7.2 illustrates the sequence followed by the logic in performing the operation.

Special consideration must be given to the control words and data accessed from Core Memory, which is supplied by the test program used in writing the "headers". When a write header operation is performed, Control Word 1 (CW1) must contain the normal beginning bulk transfer address for the track, plus Bit 23 must be set to indicate a write operation. CW2 must equal 00000040_8. CW3 specifies the Core Memory location of the first word of the 32-word transfer which will supply items B and C of Fig. WHD 7.1 for the 16 sectors of the track.

After headers have been written, they may be verified on a per track basis by selecting the header mode. (See Fig. WHD 7.3.) CW1 will be the same except for Bit 23, which should be a "zero" to specify a read operation. Control word two must equal 00000020_8, which will enable the 16 Address and Address Check-Sum words to be read into Core Memory for verification by the test program.

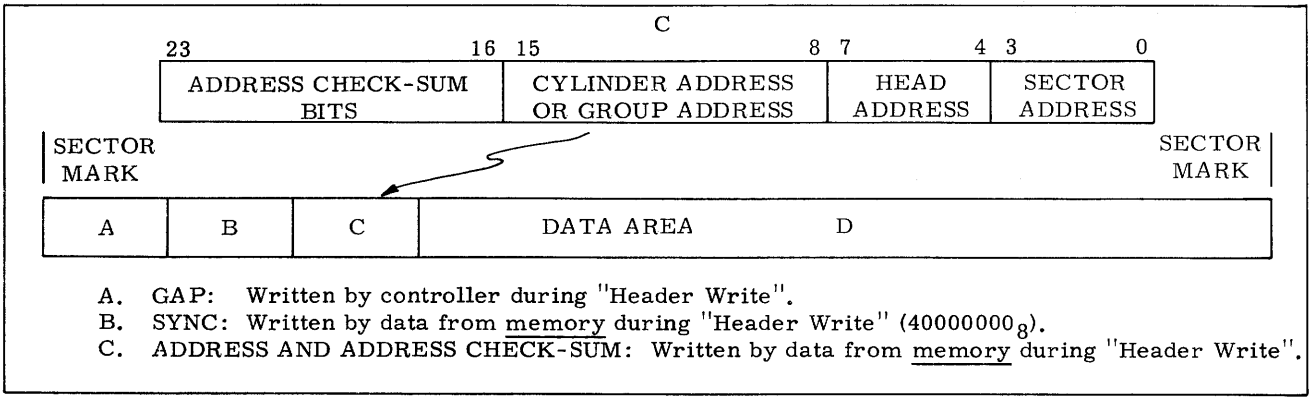
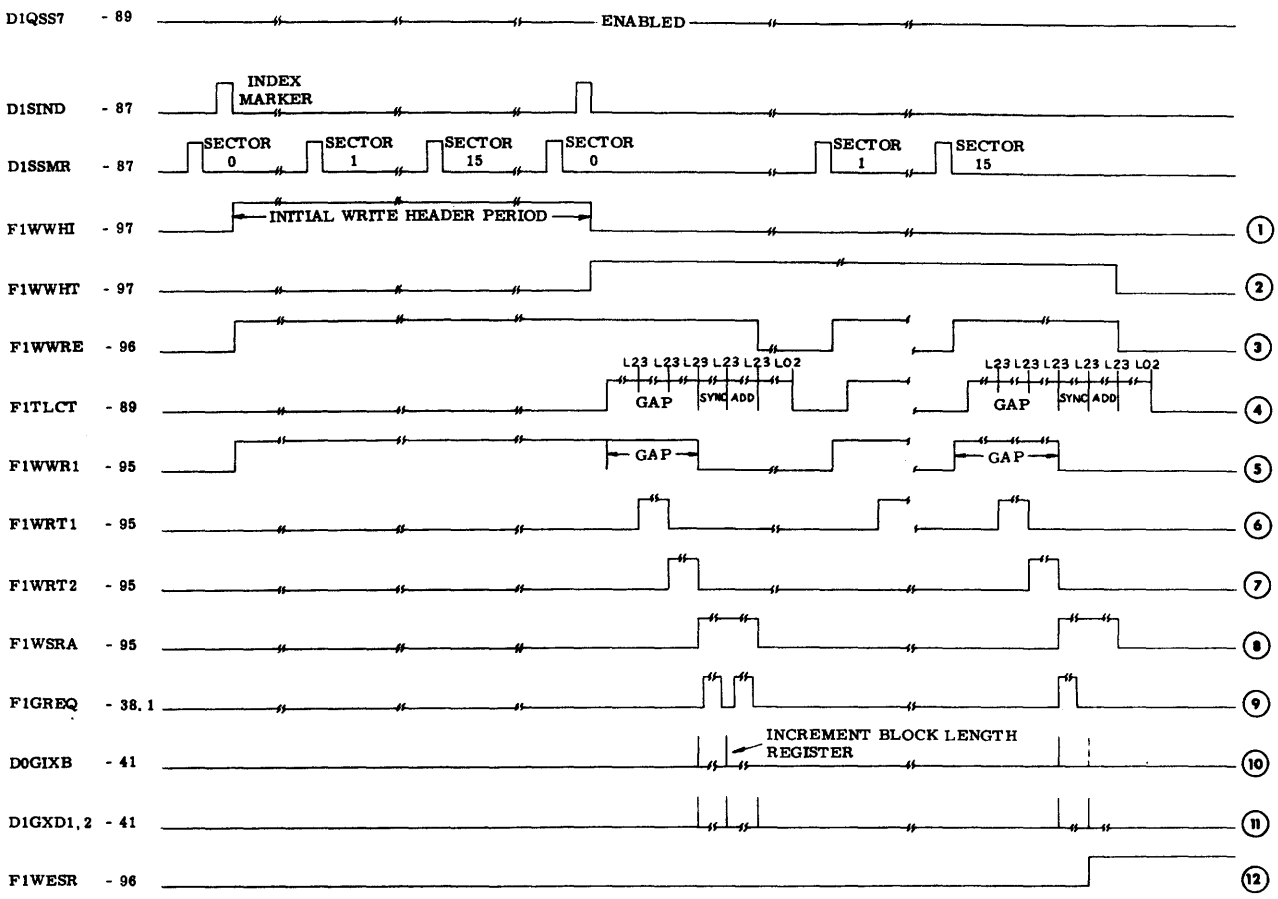


Fig. WHD 7.1 Sector Format after Write Header



(1) F1WWHI = ØHST · QSS7 · SIND · CLK2↓

$\overline{\text{F1WWHI}}$ = F1WWHI · SIND · CLK2↓

(2) F1WWHT = F1WWHI · QSS7 · SIND · CLK2↓

$\overline{\text{F1WWHT}}$ = WESR · BOBL · TL23 · CLK2↓

(3) F1WWRE = WHD1 · CLK2↓; WHD1 = (WHT1 · SIND · WWHI + SSMR · WHT1 · WHHT)

F1WWRE = WSWH · CLK2↓; WSWH = (WWHT · WSRA · TL23 · $\overline{\text{BR00}}$)

(4) F1TLCT = TWSI · CLK2↓; TWSI = (WWHI · SIND + WWHT · SSMR)

F1TLCT = TCLH · CLK2↓; TCLH = $\overline{\text{TWHD}}$ = WRHT · WWHT · $\overline{\text{WWRE}}$ · TL02

(5) F1WWR1 = WHD1 · TSS7 · CLK2↓

$\overline{\text{F1WWR1}}$ = WHDR · TL23 · CLK2↓

(6) F1WRT1 = WWR1 · WWHT · W232↓; W232 = TL23 · CLK2

$\overline{\text{F1WRT1}}$ = WWR1 · W232↓; W232 = TL23 · CLK2

(7) F1WRT2 = $\overline{\text{WRT1}}$ · WRT2 · W232↓; W232 = TL23 · CLK2

F1WRT2 = WRSA

(8) F1WSRA = WHDR · W232↓; W232 = TL23 · CLK2

$\overline{\text{F1WRSA}}$ = WHT1 · $\overline{\text{WB00}}$ · W232↓; W232 = TL23 · CLK2

(9) F1GREQ = G1GREQ $\overline{\text{PALM}}$ · GROC↓; GROC = D0GIXB

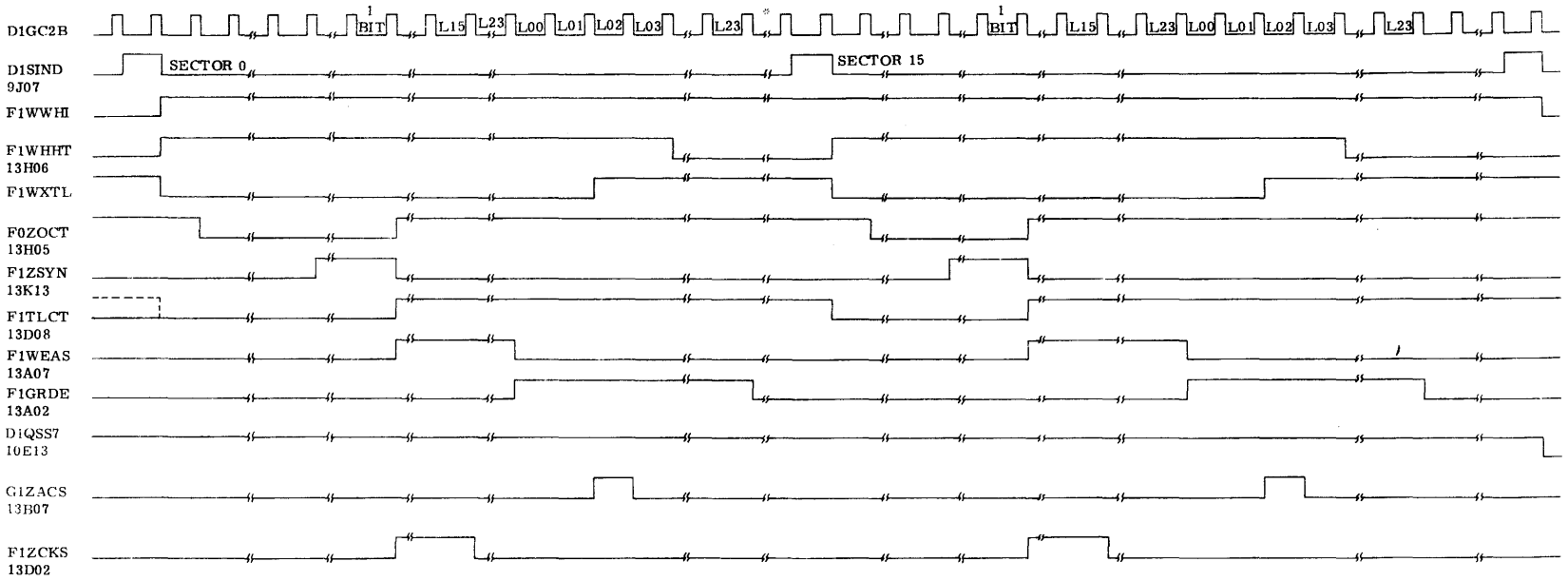
$\overline{\text{F1GREQ}}$ = F0GDRY

(10) D0GIXB = GIBL = $\overline{\text{GBLO}}$ · WSRA · CLK1 · TL01

(11) D1GXD1,2 = G0GXDA = TL23 · GEDA; GEDA = WSRA + WRT2

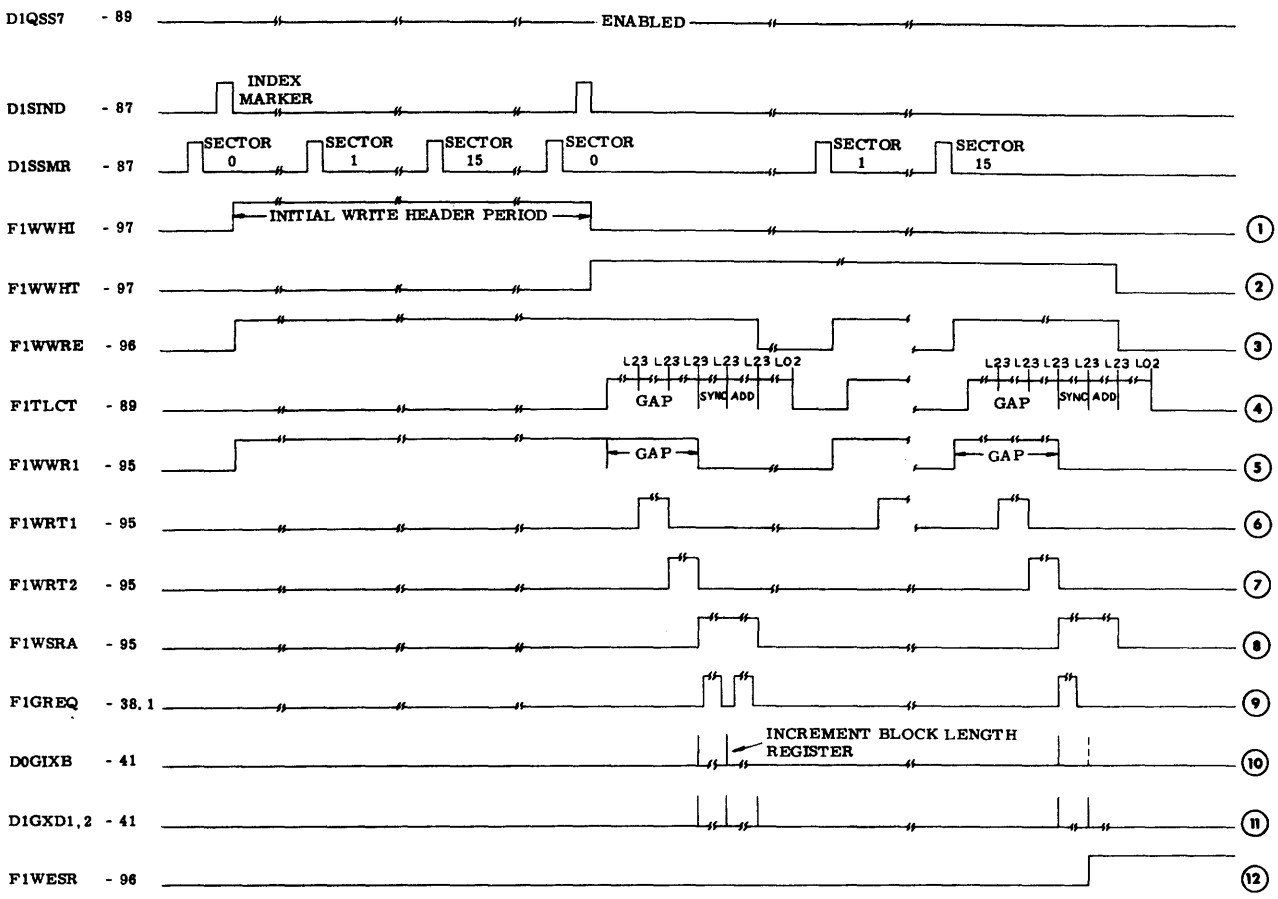
(12) F1WESR = WB77 · GXD2 · CLK2↓

Fig. WHD 7.2 Write Header Operation



* A TOTAL OF 16 SECTORS WILL BE TRANSFERRED.

Fig. WHD 7.3 Read Header Verify Timing Diagram



(1) F1WWHI = ØHST · QSS7 · SIND · CLK2↓

F1WWHI = F1WWHI · SIND · CLK2↓

(2) F1WWHT = F1WWHI · QSS7 · SIND · CLK2↓

F1WWHT = WESR · BOBL · TL23 · CLK2↓

(3) F1WWRE = WHD1 · CLK2↓; WHD1 = (WHT1 · SIND · WWHI + SSMR · WHT1 · WHHT)

F1WWRE = WSWH · CLK2↓; WSWH = (WWHT · WSRA · TL23 · $\overline{BR00}$)

(4) F1TLCT = TWSI · CLK2↓; TWSI = (WWHI · SIND + WWHT · SSMR)

F1TLCT = TCLH · CLK2↓; TCLH = $\overline{TWHD}$ = WRHT · WWHT · $\overline{WWRE}$ · TL02

(5) F1WWR1 = WHD1 · TSS7 · CLK2↓

F1WWR1 = WHDR · TL23 · CLK2↓

(6) F1WRT1 = WWR1 · WWHT · W232↓; W232 = TL23 · CLK2

F1WRT1 = WWR1 · W232↓; W232 = TL23 · CLK2

(7) F1WRT2 = $\overline{WRT1}$ · WRT2 · W232↓; W232 = TL23 · CLK2

F1WRT2 = WRSA

(8) F1WSRA = WHDR · W232↓; W232 = TL23 · CLK2

F1WRSA = WHT1 · $\overline{WB00}$ · W232↓; W232 = TL23 · CLK2

(9) F1GREQ = G1GREQ $\overline{PALM}$ · GROC↓; GROC = D0GIXB

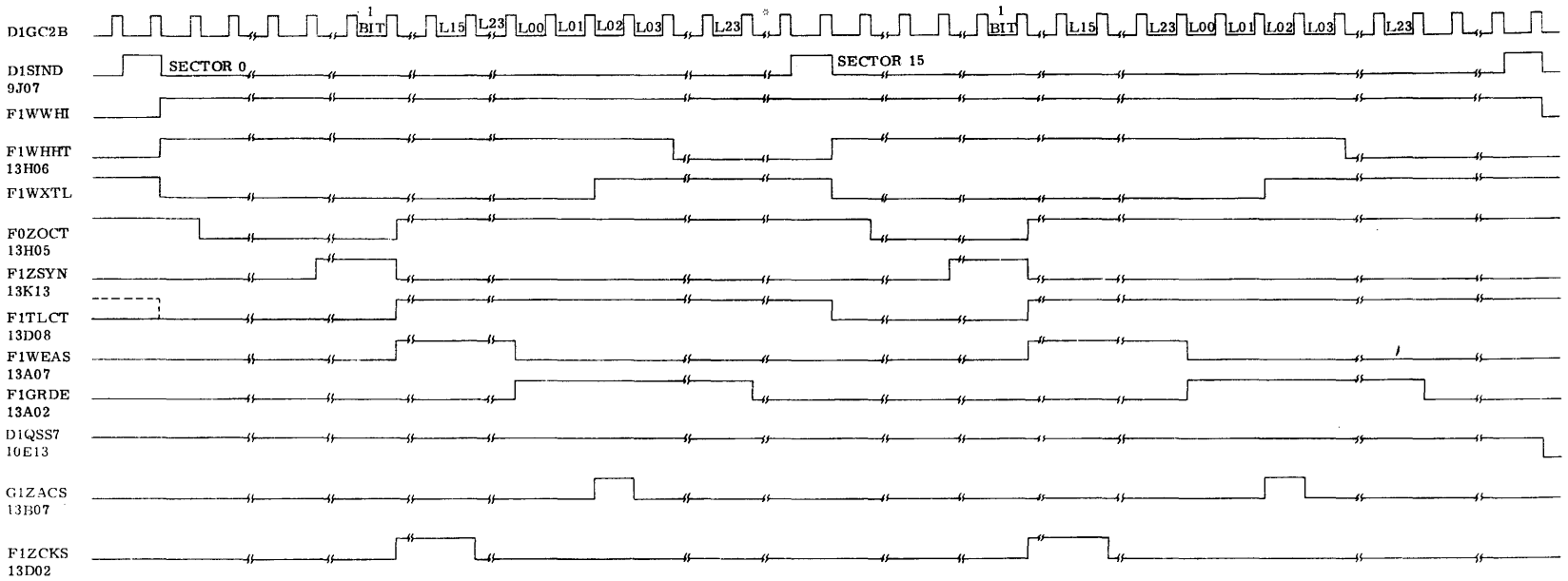
F1GREQ = F0GDRY

(10) D0GIXB = GIBL = $\overline{GBLO}$ · WSRA · CLK1 · TL01

(11) D1GXD1,2 = G0GXDA = TL23 · GEDA; GEDA = WSRA + WRT2

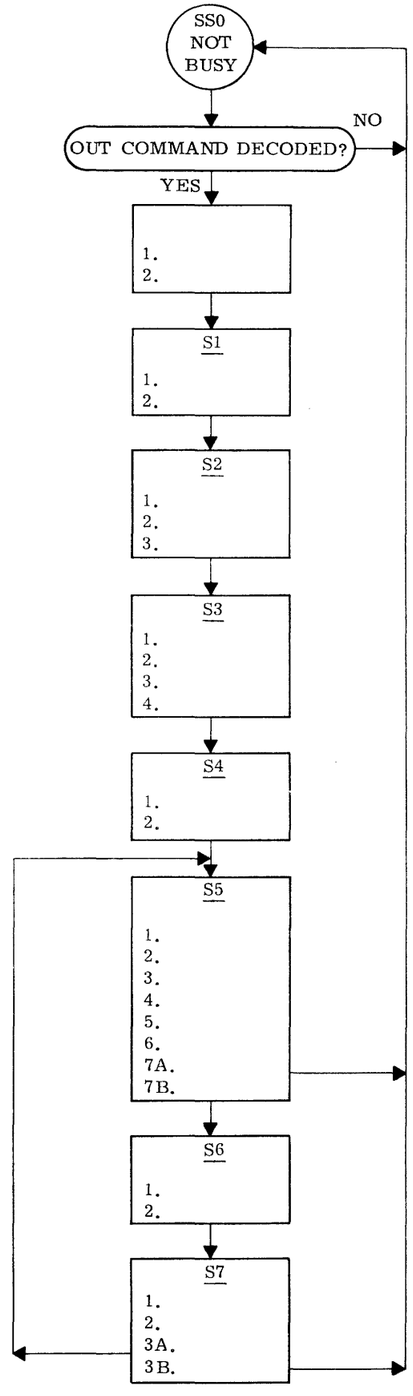
(12) F1WESR = WB77 · GXD2 · CLK2↓

Fig. WHD 7.2 Write Header Operation



* A TOTAL OF 16 SECTORS WILL BE TRANSFERRED.

Fig. WHD 7.3 Read Header Verify Timing Diagram



1. SET BUSY.
2. PRESET ALL REG.

1. REQUEST POINTER WORD.
2. LOAD POINTER WORD IN CORE ADDRESS REG.

1. REQUEST CONTROL WORD 1.
2. LOAD CONTROL WORD 1 IN DESIRED ADDRESS REG.
3. INCREMENT CORE ADDRESS REG.

1. REQUEST CONTROL WORD 2.
2. LOAD CONTROL WORD 2 IN BLOCK LENGTH REG.
3. INCREMENT CORE ADDRESS REG.
4. IF WRITE AND BL≠ 0, INCREMENT BLOCK LENGTH REG.

1. REQUEST CONTROL WORD 3.
2. LOAD CONTROL WORD 3 IN CORE ADDRESS REG.

1. IF WRITE AND BL≠0, REQUEST FIRST DATA WORD, INCREMENT CORE ADDRESS REG.
2. READ CURRENT CYLINDER ADDRESS.
3. TRANSMIT DESIRED CYLINDER/TRACK ADDRESS.
4. TRANSMIT HEAD ADDRESS.
5. CALCULATE CYLINDER/TRACK MOVEMENT PARAMETERS.
6. ENABLE CONTROL SELECT TO DEVICE.

7A. IF BL=0, GO NOT BUSY. (SEEK API WILL BE GENERATED WHEN ON CYLINDER.)

7B. IF BL≠0, GO TO S6.

1. SELECT DEVICE CLOCK.
2. WAIT FOR ON CYLINDER.

1. WAIT FOR ADDRESS COMPARE.
2. TRANSFER DATA.

3A. ON DISK, WHEN HEAD 10 AND SECTOR 15 AND BL ≠ 0. RETURN TO END OF S4.

3B. WHEN BL = 0, GO TO NOT BUSY.

Fig. WHD 7.4 Simplified OUT Block Diagram

NON-SEQUENCING COMMANDS

IN COMMAND

GEN II IN commands, executed by the Arithmetic Unit (AU) and addressed to the Bulk Memory Controller, select and transfer one of eight groups of controller status information to the "A" register of the AU. The specific group selected for transfer is determined by the status of the K_0 portion of the IN command.

An IN command's octal format when addressing the Bulk Memory Controller is 2505101X, where X refers to the K_0 bits that can be assigned binary configurations 000 through 111 to select one of eight groups for "read-in". The K_0 bits are decoded in the controller, as shown on sheet 46 of its logic, and enable one of eight signals (N1GKS0 through N1GKS7). (In order to enable the select signal, the K_1 bits of the IN command must equal 001_2.)

During the execution time of the IN command, signal D0GGIN (sh. 53) disables the decoding of the display select signals from switches SW6 through SW8 of the controller's maintenance console, while the K_0-decoded select signal enables one of the drivers, D1GDB0 through D1GDB7. The enabled driver permits the desired output information to be gated to gates G0OB23, through G0OB00, shts. 77 and 78, for transmission to the AU.

The information transmitted into the specific bit positions of the AU's "A" register, and the K_0 bits used to decode the select signal for its transmission, are shown by Table NSC. 1.

ACTIVATE COMMAND

An Activate (ACT) command can be used to cycle the controller ready line or any one of the four possible seek complete lines from the Bulk Memory Units (BMU's), if the sampled line is in a "ready" condition. In the event that an ACT command is executed when the specified line is "not ready", the line does not cycle and no consequent interrupt is generated.

All three K_0 bits and the least significant K_1 bit of the ACT command are used to determine which line is intended to be cycled. The specific lines intended for cycling are designated by the following K_1K_0 configurations:

$K_1 = 0$, $K_0 = 000_2$	Controller Ready
$K_1 = 1$, $K_0 = 000_2$	Seek Complete (Unit 0)
$K_1 = 1$, $K_0 = 001_2$	Seek Complete (Unit 1)
$K_1 = 1$, $K_0 = 010_2$	Seek Complete (Unit 2)
$K_1 = 1$, $K_0 = 011_2$	Seek Complete (Unit 3)

A successful cycling of the controller's Ready Line does not imply that any one or all of the BMU's are ready, but does indicate that the controller is available for receipt of a data transfer (OUT) command. However, successful cycling of the specified unit's seek complete line is an indication that the BMU is not seeking.

Sheet 52 of the controller logic shows the Controller Ready and Seek Complete activate lines. Gate G0GCAT will go to zero volts, during the ØB portion of the ACT command, if the busy line (G1GBSY) is "false" and Bit 03 of the ACT command is a "zero". This generates a positive pulse, from inverter N1GCAT, whose trailing edge is detected by the API logic as Controller Ready interrupt.

The status of the seek flip-flops, F1GSK0 through F1GSK3, is sampled during ØB of the ACT command by gates G0GPA0 through G0GPA3, also shown on sheet 52. These signals are inverted and the desired line is sampled by the presence of one of the K_0-decoded enabling signals, N1GKS0 through N1GKS3, at gates G0G0AT through G0G3AT. (The K_0-decoded enabling signals cannot be decoded unless Bit 03 of the ACT command is a "one" bit.)

If the seek flip-flop being sampled was "cleared", the enabled sampling line will provide a positive pulse that is inverted and sent to the API logic, generating an API when the signal goes negative. If the sampled flip-flop was set, no signal will be generated, indicating that the BMU has not completed its seek operation.

ABORT COMMAND

An Abort (ABT) command can be either conditional or unconditional, as specified by the status of bit 3 of the ABT command word.

A conditional ABT command (Bit 3 equal to a "zero") causes an orderly termination of any command in operation within the controller. The unconditional ABT command (Bit 3 equal to a "one") causes the controller to immediately terminate any operation in progress.

The conditional ABT can be used by typical operating programs to interrupt a bulk transfer operation without fear of losing or destroying information since the abort does not take place until the end of a sector transfer or, if a head movement is in progress, when the disk heads reach the desired cylinder. At this point, the controller's Busy flip-flop is reset and the desired operation can be performed before returning to complete the aborted operation.

Bit No.	K_0 Decode Group							
	7	6	5	4	3	2	1	0
23			A23(20)	D23(20)		RXE(41)	WRT(93)	SS7(82)
22			A22(20)	D22(20)		XTL(94)	VEF(93)	SS6(82)
21			A21(20)	D21(20)		WR1(95)		SS5(82)
20			A20(20)	D20(20)	DSK(51)			SS4(82)
19			A19(20)	D19(20)	SK3(51)			SS3(82)
18		ABC(47)	A18(20)	D18(20)	SK2(51)			SS2(82)
17		INC(38)	A17(20)	D17(20)	SK1(51)	WRE(96)	U01(93)	SS1(82)
16		DRY(39)	A16(20)	D16(20)	SK0(51)	EAS(97)	U00(93)	SS0(82)
15		REQ(38.1)	A15(20)	D15(20)	M15(71)	CMP(106)	C07(25)	BSY(44)
14		REV(57)	A14(20)	D14(20)	M14(71)	LCT(89)	C06(25)	STP(75)
13		HLD(92)	A13(20)	D13(20)	M13(71)	B13(21)	C05(25)	
12		TLA(92)	A12(20)	D12(20)	M12(71)	B12(21)	C04(25)	D24(79)
11	Z11(102)	TLB(92)	A11(20)	D11(20)	M11(72)	B11(22)	C03(26)	A24(79)
10	Z10(102)	TLC(92)	A10(20)	D10(20)	M10(72)	B10(22)	C02(26)	PAL(79)
09	Z09(102)	TLD(92)	A09(20)	D09(20)	M09(72)	B09(22)	C01(26)	USL(37)
08	Z08(102)	TLE(92)	A08(20)	D08(20)	M08(72)	B08(22)	C00(26)	MTM(37)
07	Z07(103)	KD7(61)	A07(20)	D07(20)	M07(73)	B07(23)	H03(56)	SEK(37)
06	Z06(103)	KD6(61)	A06(20)	D06(20)	M06(73)	B06(23)	H02(56)	OFL(37)
05	Z05(103)	KD5(61)	A05(20)	D05(20)	M05(73)	B05(23)	H01(56)	WPT(37)
04	Z04(103)	KD4(61)	A04(20)	D04(20)	M04(73)	B04(23)	H00(56)	ACS(37)
03	Z03(105)	KD3(61)	A03(20)	D03(20)	M03(74)	B03(24)	S03(88)	ADC(37)
02	Z02(105)	KD2(61)	A02(20)	D02(20)	M02(74)	B02(24)	S02(88)	FLL(37)
01	Z01(105)	KD1(61)	A01(20)	D01(20)	M01(74)	B01(24)	S01(88)	FUS(37)
00	Z00(105)	KD0(61)	A00(20)	D00(20)	M00(74)	B00(24)	S00(88)	DCS(37)

NOTE: The numbers shown in parentheses, following the mnemonics in the table, represent the sheet number of the logic where the signals are generated to drive the indicator specified and are included here for convenience in referencing.

Table NSC.1

Unconditional ABT's are normally used for test purposes only, since the status of controller flip-flops and the contents of controller registers are altered. In addition, data being transferred during the execution of the unconditional ABT may be lost or a "sector write" may be left uncompleted.

Controller logic for the ABT command decoding is shown on sheet 47. The conditional ABT is decoded at gate G0GABC, where signal D0GK1S will be enabled if Bit 03 of the ABT command is not a "one". The presence of D1GK11 at the unconditional ABT decode gate, G0GUAB, indicates that Bit 03 of the command must be a "one" bit. G0GUAB enables G1GAB1, the same gate enabled if the manual initialize button were depressed, to enable the initialize signals, D0GIA1 through D0GIA3.

JNR COMMAND

A JNR command can be used to determine the availability of the controller or the individual Bulk Memory Units (BMU's). Executing the JNR permits the desired line to be sampled at the Jump gate, G1GJMP, located on sheet 48 of the controller logic. The output of G1GJMP is ANDed with signal G0GFJP, which prevents the jump logic from indicating the BMU's are ready between the time the controller receives an OUT command and the time it has incremented to Sequence State 4 (SS4).

Coding of the K_1 and K_0 bits of the JNR command determines which line will be sampled during its execution. Each ready line and its corresponding K_1K_0 code is as follows:

$K_1 = 000_2$, K_0 = any configuration	Controller Ready
$K_1 = 001_2$, $K_0 = 000_2$	Unit 0
$K_1 = 001_2$, $K_0 = 001_2$	Unit 1
$K_1 = 001_2$, $K_0 = 010_2$	Unit 2
$K_1 = 001_2$, $K_0 = 011_2$	Unit 3

All of the test lines are ORed into G1GJMP. The controller test line, G1GJMP, is enabled if the controller Ready line (G1GRDY) is "true", Bit 03 of the ACT command is a "zero", and the ACT command has been addressed to the controller.

There are four seek flip-flops in the controller, one for each possible BMU. The period of time when one of these flip-flops is set indicates when the actual unit is busy with a "seek" operation. A flip-flop assigned to a specific BMU is set whenever a Return-to-Zero Seek or an OUT command is addressed to that unit, and cleared whenever any necessary seek has been completed. In the case of the drum units, this only requires a few microseconds since the drum's heads are fixed and no seeks are necessary. However, disk units may require more than 100 milliseconds to complete seek operations. In either case the unit sends back a "seek complete" signal that clears the corresponding Seek flip-flop in the controller

A JNR command that is addressed to test the "ready" status of a BMU actually samples the status of the Seek flip-flop in the controller that is assigned to that unit. The sampling gates, G1GJS0 through G1GJS3, are shown on sheet 48 of the controller logic. In order to enable one of these gates during a JNR command (N1GJPC at plus 5 volts), it is necessary that the specific K_1K_0 bits be decoded properly and the sampled seek flip-flop (F1GSK0 through F1GSK3) must be cleared. This would generate a zero volt signal from the sampled gate, enabling G1GJMP to go to plus 5 volts. This, in turn, would enable G1GJNR to go to zero volts, if G0GFJP were also at plus 5 volts, indicating that the unit is ready. G0GFJP goes to zero volts, disabling G1GJMP and indicating the device is not ready, during the period when the controller is receiving and transferring movement information to the BMU (i.e., SS0 through SS3).

The Ready line is actually sampled in the Arithmetic Unit (AU) during the execution of the JNR command. If it is "true" (G1GJNR at plus 5 volts), the AU's Program Register is incremented by a count of one. If G1GJNR is "false" (0 volts) during the test, the Program Register is incremented by a count of two. Incrementing the Program Register by one indicates that the BMU was ready, while incrementing it by two (considered as a "jump") indicates that the BMU was not ready.

JNE COMMAND

A JNE command can be used to indicate the error status of the Bulk Memory Controller. There are ten different alarm indicators (defined later in this command description) that are OR'ed together at the controller's error test line.

The error test line (G1GJNE, sheet 47 of the controller logic) goes to zero volts if there is an error detected when the JNE command is decoded in the controller. If the error line is at zero volts during the sampling, which takes place in the Arithmetic Unit (AU), the AU's Program Register is incremented by one. When there is no error present during the test, G1GJNE remains at plus 5 volts and the AU's Program Register is incremented by two. Therefore, an error present during the execution of a JNE command results in the AU's Program Register being incremented by one and no error causes it to be incremented by two (regarded as a "jump").

G1GJNE, the controller's error test line, samples the status of signal N1EALM during the execution of a JNE command addressed to the controller. N1EALM is an inverter driven by G0EALM, shown on sheet 34 of the logic. G0EALM is the gate where the actual OR'ing of the alarm signals to the test line takes place.

Error Definitions

The following is a list of the errors detectable by the controller and a brief description of the conditions that can generate them.

UNIT SELECT

This error can only be detected during Sequence States five through seven of the controller's operation. The error is detected if a unit select signal is returned from a unit other than that selected by the controller, if either the unit select from the controller or unit is lost, or if a connected unit becomes not ready.

MEMORY TIMING

Indicates that a core memory request was not acknowledged within the time restriction allowed by the controller.

SEEK

Indicates the selected disk unit was unable to successfully complete a head movement requested by the controller. The error is actually detected within the device and only applies to disk units; drum units do not have movable heads.

ON LINE

If the selected device is a drum unit, this error indicates the device is not available for on-line operation due to a BLM bad indication, loss of the device's read or write control voltages, the heads not flying, or the drum unit being switched to an off-line condition. For disk units, this error indicates the device has been switched off-line, the disk pack rotation is not up to speed, or the heads are not loaded.

WRITE PROTECT

Write protection is available through switches on the controller's maintenance panel. These switches enable selectable write protection for entire units, for specific cylinders (or groups, if a drum unit) in the cylinder (or group) 00 through 07 range, or for the area covering cylinders (or groups) 10_8 through 37_8.

The drum units have pin-selectable write protection circuitry that works independently or the controller's write protect. A write protect panel is provided at the drum unit containing holes for accommodating insertable pins. Part of the panel is reserved for unused pin storage, but the remainder is dedicated for protecting specific areas of the drum when a pin is inserted.

The panel is organized so that write protection is selectable, by insertable pins, for 4096_{10} word blocks in the first eight drum groups, which encompasses 131,072 words. In addition, a block consisting of the first eight groups and all subsequent blocks of eight groups can be individually selected for protection.

A write protect error is detected when the controller attempts to perform a write operation in one of these protected areas. This error could result from an oversight in leaving an area protected that can now be utilized, from a hardware failure, or from an error in software parameters.

ADDRESS CHECK-SUM

This error indicates that the address read from the device contained a different count of "one" bits than did its accompanying check-sum. This could result from a transient or permanent hardware failure, damage to the recorded area, or an improperly recorded address and address check-sum (recorded by a Write Header operation).

ADDRESS COMPARE

This error indicates that the selected head has read the entire track and has failed to locate the desired address. This could result, in the case of the disk units, from a logic or mechanical failure during a "seek" operation. In the case of a drum unit, where no "seek" operations are necessary, it could result from an attempt to locate a nonexistant drum address. For either type of device, it could result from a failure in reading the addresses from the device.

FILE LIMIT

This error is detected when the controller attempts to address beyond the limit of the selected device. This is specifically intended for accessed disk units, and the limit is pin-selectable in the controller to match the addressing limits (1,024,000 or 2,048,000) of the disk units assigned to the controller. However, the same error could be detected when accessing a drum unit if the controller attempts to address beyond the pin-selected limit.

FILE UNSAFE

If the selected device is a drum unit, this error is detected when the device's read or write control voltages are out of tolerance, or when both the read and write circuits of the device are selected simultaneously.

The error is detected under the following conditions if the selected device is a disk unit:

a. Inability of disk unit's head carriage to detent during First Seek or Return-To-Zero operation.

b. Loss of 30% disk rotational speed with heads extended.

c. Disk unit oscillator failure with heads extended.

d. Loss of any disk unit DC voltage.

e. Carriage hitting mechanical stop during seek operation.

f. Head select malfunction.

g. Write circuitry malfunction.

h. Read circuitry malfunction.

i. Erase circuitry malfunction.

The detection of any one of items a. through d. causes an emergency retraction of the heads.

DATA CHECK-SUM

This alert, if set, indicates either that a transfer between the core memory multiplexer and the DBC has incurred an error in parity or a "data read" by the DBC from one of the BMU's has incurred a discrepancy between the 12 bit check sum previously stored for the 64 word sector and that reconstructed during the readout.

On a parity error occurring during the Pointer or Control word accesses, the DC alert is set, DBC operation ceases with that memory access, the Ready indicator is set, and the Error indicator is set. On data transfers if the alert results from a parity error, the DC alert is set, the transfer is continued to the end of the current sector with fill words, the RDY indicator is set Ready, and the Error indicator is set.

If the alarm is set as the result of a data check sum error, the transfer halts (the error having been detected at the end of a sector), and the RDY and Error indicators are set.

OPERATE (RETURN TO NORMAL)

An Operate command (GEN II command with S=2) is decoded in the controller as a Return-to-Zero Seek (RTZ) operation. The primary intended function of an RTZ operation is to return the bulk device's heads to a known location (i.e., cylinder 00 for a disk unit).

During ØA of the OPR command, the controller sets the RTZ flip-flop and sets the controller Seek flip-flop that corresponds to the selected unit. The Control Select signal (X1LCSL, sh. 69) and Tag Control Line 6 (G0LTG6, sh. 63) are also enabled during this time, informing the device of the type of operation it is to perform.

Selection of the unit that is to perform the RTZ operation is accomplished in one of two manners, determined by the status of Bit 03 of the Operate command. If Bit 03 is a "zero", the unit selected is the one that was last accessed by the controller for a "seek-only" or data transfer operation. When Bit 03 of the OPR command is a "one" bit, bits 02 through 00 (K_0 bits) of the OPR are decoded to select the unit.

This command is not actually a useful command for drum units, although drum units receive the command when they are addressed and return a Seek Complete signal several microseconds later as a response. This clears the Seek flip-flop for the unit in the controller and generates a Seek Complete interrupt.

When addressed to a disk unit, this command initiates a return to cylinder zero movement of the heads. When the heads arrive at cylinder zero, a Seek Complete signal is returned to the controller, generating a Seek Complete interrupt and clearing the controller's Seek flip-flop.

READER COMMENTS

The General Electric Company solicits your comments on publications covering Process Computer equipment. Please explain any "no" responses in the COMMENTS section. Your comments and suggestions become the property of the General Electric Company.

- Name of Manual: _____
- What is your computer application: _____

  _____
- How is this publication used:

  Familiarization ☐ Reference ☐

  Training ☐ Maintenance ☐

  Other (Explain) _____

	YES	NO
• Does this publication meet your requirements	☐	☐
• Is the material:		
1) Presented in clear text	☐	☐
2) Conveniently organized	☐	☐
3) Adequately detailed	☐	☐
4) Adequately illustrated	☐	☐
5) Presented at appropriate technical level	☐	☐

- Please provide specific text references (page number, line, etc.) with your comments.

NAME _____ DATE _____

TITLE _____

COMPANY NAME _____

AND ADDRESS _____

COMMENTS:

No postage necessary if mailed in the U. S. A.

Staple

Communications concerning Technical Publications should be directed to:

Manager, Technical Publications
Utility and Process Automation Systems Operation
2255 West Desert Cove Road
Phoenix, Arizona 85029

Fold Fold

FIRST CLASS
Permit No. 4091
Phoenix, Arizona

BUSINESS REPLY MAIL
NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY...

GENERAL ELECTRIC COMPANY
UTILITY and PROCESS AUTOMATION
SYSTEMS OPERATION
2255 West Desert Cove Road
Phoenix, Arizona 85029

Attention: Technical Publications

Cut Along Line

Fold Fold

Additional Comments:

DRUM MEMORY UNIT 4521A

TABLE OF CONTENTS

INTRODUCTION

Drum Memory Units provide the storage and retrieval of bulk data and programs on rotating cylindrical surfaces. While drum memory does not provide the very large storage capacity of disc memory, data can be recorded and read from drum memory in less time, with high reliability. The Drum Memory Units employ fewer moving mechanical parts than Disk Units, and therefore require somewhat less maintenance.

The drum memory consists of a rotating drum surface with a special magnetic coating which is designed to produce an optimum combination of output signal level and data density (bits per inch). Data are recorded and read by small magnetic heads, whose position over the drum surface is fixed. When the drum is rotating at a proper speed, the heads are allowed to "fly" above the drum surface on a very thin film of helium gas. As the drum spins, a narrow "track" on the magnetizable surface passes below each head, and the binary data are recorded on the tracks by passing controlled currents through the heads. When reading data from the tracks, a head operates as a variable reluctance pick-up, which provides an analog output signal to the read-back electronics.

Drum Memory Units are available in two basic shell sizes. One is capable of storing up to 131, 072 24-bit words. The other is capable of storing up to 524, 288 24-bit words. The unit with the smaller basic capacity may implement logic and heads to store 32, 768, 65, 536, or 131, 072 words. The unit with larger capacity may implement 131, 072, 262, 144, 393, 216, or 524, 288 words. In spite of differing storage capacities, the two basic units have the same external dimensions.

REFERENCES

The following documents provide information useful in understanding the operation and maintenance of the Dual Bulk Memory Subsystem, of which, the Drum Memory Unit is a major component:

- GE-PAC* 4010 System Summary Manual, GET-6035, Section 4.
- GE-PAC 4010 General Description, GET-6074, Section 2.2.
- Dual Bulk Controller Theory, pub. no. 4852A or 4852B.

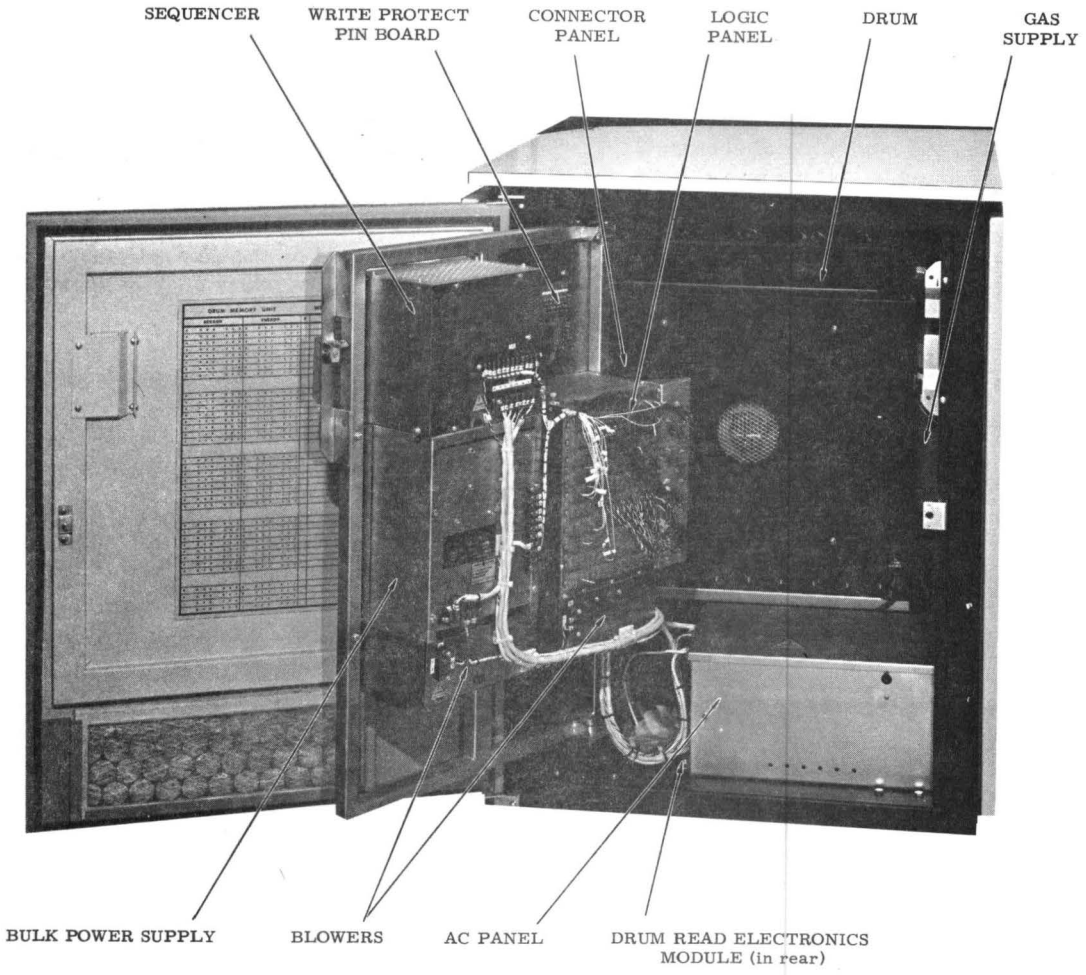


Fig. INT. 1 4521A Drum Memory Unit

*Registered Trademark of General Electric Company

- Logic, Drum Memory Unit, 70C180293.
- Logic, Drum Read Electronics, 70C180119.
- Logic, Dual Bulk Controller, 70C179882 (4010A) or 70C180908 (4010B).
- Computer Maintenance Manual, Section 9, Drum Memory Unit.

FUNCTIONAL DESCRIPTION

Included in the Drum Memory Unit (DMU) are the helium filled drum assembly, the voltage sequencer (VS1), the power supply (PS1), the basic logic panel (LP1), and the drum read electronics module (DRE).

The drum assembly is sealed, and gas pressure is maintained by a supply bottle and regulator assembly. The sealed drum assembly, including the magnetic drum, head pads, motor, and head and motor control electronics, is normally considered as a non-repairable item. If repair or modification of the drum assembly becomes necessary, in most cases, a temporary DMU exchange will be made, while the repair or expansion is made at the factory.

Each drum has three pre-recorded clock tracks, one active, and two spares. All data storage and clock tracks have 29,184 bit positions. The clock tracks contain "ones" in all bit positions, except for a "zero" which marks the start of each of sixteen sectors around the circumference of the drum. Three zeros mark the beginning of the first sector. Each of the single "zeros" marking the beginning of sectors 2 through 16 is referred to as a "sector mark". The three "zeros" marking the beginning of sector one are referred to as the "index mark".

The drums also incorporate 32, 64, 128, 256, 388, or 512 active read/write data heads. Each data head is positioned over a sixteen sector track, and each sector includes 64 24-bit data words. Each read/write head records or reads 1024 data words (16 sectors X 64 words). The heads are mounted in 18-head pads. Sixteen heads on each pad are active, two are reserved as spares, one of the spares is reserved for use by the basic drum vendor, and one for the Process Computer Department.

Each of the sixteen data sectors on all implemented tracks contains information in the form depicted on Fig. INT. 2. The address and data checksums are generated in the Dual Bulk Controller (DBC) during the recording process, and checked when reading from the drum. The sector address word is recorded when executing the "write headers" portion of the Dual Bulk Subsystem test program (70A100003).

A dual frequency, non-return to zero (NRZ) recording process is used. When the drum is rotating at its nominal speed, 3600 RPM, each bit position passes below its head in 578 nanoseconds. A "one" bit is represented by one polarity reversal in the magnetic field on the drum track within a bit position (one full cycle), and a "zero" is represented by one half-cycle per bit position. "Ones" are, therefore, written and read-back at a nominal frequency of 1.73 MHz and "zeroes" at 0.865 MHz.

The active clock track is continuously read while the DMU is operating and sector mark and index pulses are fed to the Dual Bulk Controller during any period when the controller has selected a DMU. Writing and read-back are also under the control of the controller. A DMU selected by the controller provides status and alarm information which may be monitored by the controller.

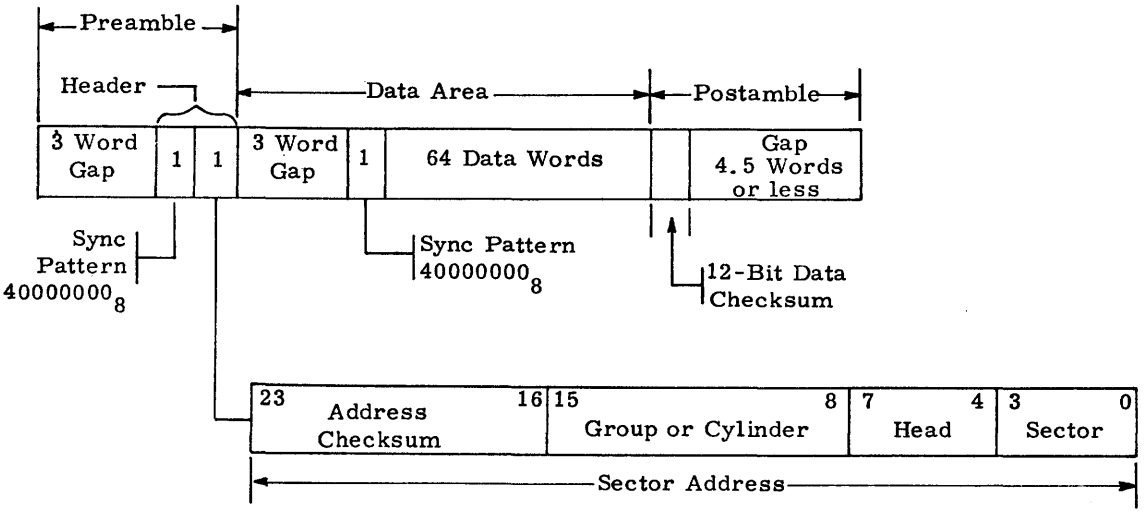


Fig. INT. 2 Sector Format

A Write Protect pin board is provided on the DMU to protect data and programs stored on the drum from being overwritten due to software errors or hardware faults. Pins are provided to bypass writing current directed to each 4-head block (4096 words) in the first eight head groups (16 heads per group) and to bypass current directed to each 8 group block (131, 072 words). Should the controller attempt to write on a protected track, a Write Protect Alarm signal is applied to the controller.

THEORY OF OPERATION

BASIC READ/WRITE THEORY

The read/write heads are small electromagnets formed by a center tapped winding on low reluctance core. The core has a very small gap in the point nearest the drum surface. When the head is in the operating position close to the drum surface, the drum surface forms a path for magnetic flux to travel across the gap in the core. This provides a closed loop for the flux which is generated by current applied to the winding. When reading from the drum, a small current is induced in the winding, as previously recorded information passes the gap.

Fig. THEORY. 1 illustrates the basic elements of a drum head. When writing information on the drum, current flow through the head winding is continually changing. Each half-cycle of write current is applied from the center tap to alternate half-windings. The rate at which the current and flux are reversed dictates whether a data bit is a "one" or a "zero". A "one" consists of one reversal (one full cycle) per bit position. A "zero" consists of one half-cycle per bit position.

When reading data from the drum, an analog voltage of about 10 millivolts peak-to-peak is developed across the entire winding and is applied to an amplifier in the Drum Read Electronics (DRE) module.

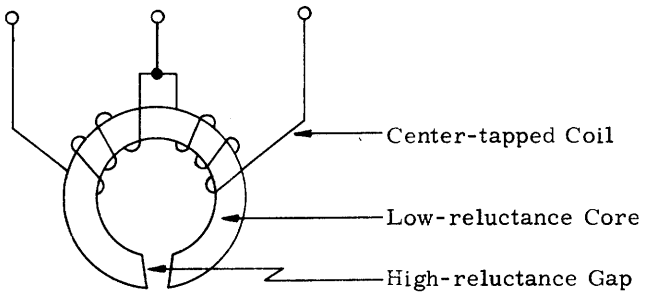


Fig. THEORY. 1 Read/Write Head

Fig. THEORY. 2 depicts idealized bit waveforms in three forms; (1) as they appear on the data line between the controller and the drum or disc units, (2) at the input to the write amplifiers in the Drum Memory Unit, (3) and the current in the read/write heads. The analog voltages produced by the heads when reading from the drum have an appearance similar to waveform 3 and are converted to appear like waveform 1 on the data line to the controller. The illustration does not show delays which occur in reading and writing.

INTERFACES

Each bulk memory unit is interconnected with the Dual Bulk Controller and any other bulk memory units in the system through a series of interconnecting cables which carry the same signals between the controller and all units. In addition, a second cable carries signals destined to, or from, each individual unit. Fig. THEORY. 3 illustrates the relation of the Drum Memory Unit to the controller and the other bulk memory units in the system. In the case shown, the Drum Memory Unit is the first unit in the system (unit 0). When operating as unit 1, 2, or 3, the cables connected to CP2-J02 and CP2-J03 are connected between the preceding unit and the next unit.

All of the signals carried by the interconnecting cables, with the exception of some DC voltages and power sequencing signals are transmitted on two wire signal lines. When the signal is present or "true", the line drivers drive one of the lines in the pair in a positive direction, and the other line in a negative direction. The line receivers are integrated circuit comparators, which respond only to a voltage difference on the input pair. This method of signal transfer is fast, features high noise rejection, and is highly reliable.

A schematic of the cable from the controller to the individual units appears on sheet 3 of the Drum Memory Unit logic, drawing no. 70C180293. The controller/inter-unit cable appears on sheet 4 of the logic.

The interface line drivers and line receivers appear on sheets 13 through 16 of the -293 logic. The terminations for the individual unit cable are on sheet 14. The remaining drivers and receivers terminate the inter-unit cables. Fig. THEORY. 4 is a schematic of a typical inter-unit signal path.

Transistor Q1, in the line driver, is conducting when the input signal is not present or "false". When the input goes "true", Q1 is turned off, allowing the base and emitter of Q2 to swing in a positive direction, and driving the collector of Q2 in the negative direction. Q3 and Q4 are common base amplifiers which provide power gain without inverting the emitter signal input. The comparator in the line receiver senses the voltage difference at its two inputs, and its output is driven in a negative direction. The comparator output is inverted by the M1Ixxx gate.

The example used in Fig. THEORY. 4 is typical of the circuits terminating the cables between the Dual Bulk Controller and the individual units. Line drivers and line receivers of the same type are used for signals exchanged between the controller and all units, but for signals going to the units, receivers in each of the units are connected in parallel; and for signals going from the units, drivers in each of the units are connected in parallel.

Fig. THEORY. 5 provides an example of a "Tag Line" signal path between the controller and the units. The Tag Lines are used to transfer control information from the controller to the bulk memory units. Disc Memory Units also return control information to the controller on the same Tag Lines, but Drum Memory Units do not return such information.

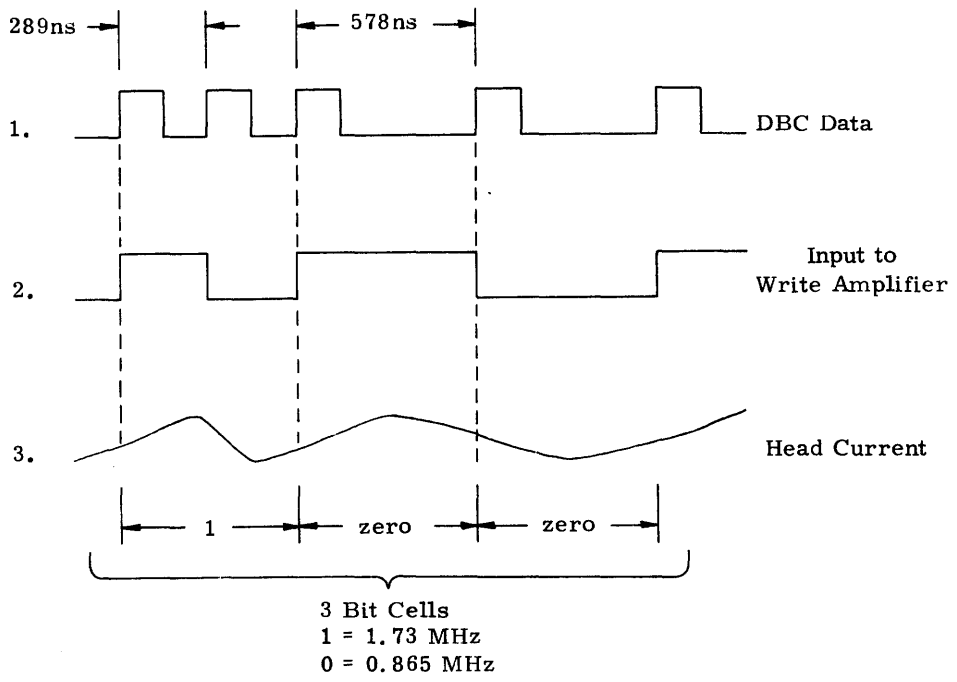


Fig. THEORY. 2 Bit Waveforms

DRUM MEMORY UNIT CONTROL

The control and operation of a Drum Memory Unit is relatively passive. The unit continuously reads clock track data, and if it has been selected by the Dual Bulk Controller, it decodes that data to generate sector and index marks for the controller. In addition, when selected and placed in the write mode by the controller, the drum unit writes on the data sectors indicated by the controller, for as long as the controller tells it to do so. When selected and placed in the read mode, the unit reads from the tracks indicated by the controller, for as long as the controller tells it to do so.

The operation of the drum units, therefore, can be understood without too much concern about the operating sequence of the Dual Bulk Controller. The drum unit operation is the same during normal read and write operations by the controller, and during special modes of operation implemented by the controller, such as header recording and playback, verify tests, and fill tests. Of course, an understanding of the overall operation of the Dual Bulk Memory Subsystem, requires study of the theory of operation of the Dual Bulk Controller. This is provided in theory publication no. 4852A, which is included in this section.

The responses to, or causes of, each of the control signals in the interface between the Dual Bulk Controller and the Drum Memory Unit are described in the following paragraphs.

Unit Select

Unit Select is generated by the Controller when it requires any operation by a unit. In a transfer sequence, the Unit Select signal for the unit selected by control word 1 will go "true" in sequence state 4 and remains true until the transfer is complete. Unit Select also goes true if an OPR instruction is addressed to the controller, to initiate a return to zero seek operation. The unit selected is dictated by the S' bit (bit 3) of the instruction word and by the K0 bits as follows:

S'	K0	Unit
0	0_8	As selected by last CW1
1	0_8	Unit 0
1	1_8	Unit 1
1	2_8	Unit 2
1	3_8	Unit 3

The Unit Select signal is transferred over the individual unit cable (sh. 3 of -293 logic). The line receiver appears on sheet 14 of the -293 logic. The select unit signal, M1ISLU, partially enables the unit on-line and unit selected gates, G0DUOL and G0DUUS, on sheet 17. With SLU "true", G0DUUS is enabled when the CSU bus level monitor (BLM) is "good".

$$\overline{\text{F1DBNG}} = \text{CSU BLM good}$$

$$\text{G0DUUS} = \text{SLU} \cdot \overline{\text{BNG}}$$

G0DUOL is enabled when the CSU BLM is good, the unit is selected and, F1DONL is set. F1DONL is set when the drum unit is on-line and the heads are loaded in the proper position. This circuit is described in more detail under the "Power Supplies and Controls" heading.

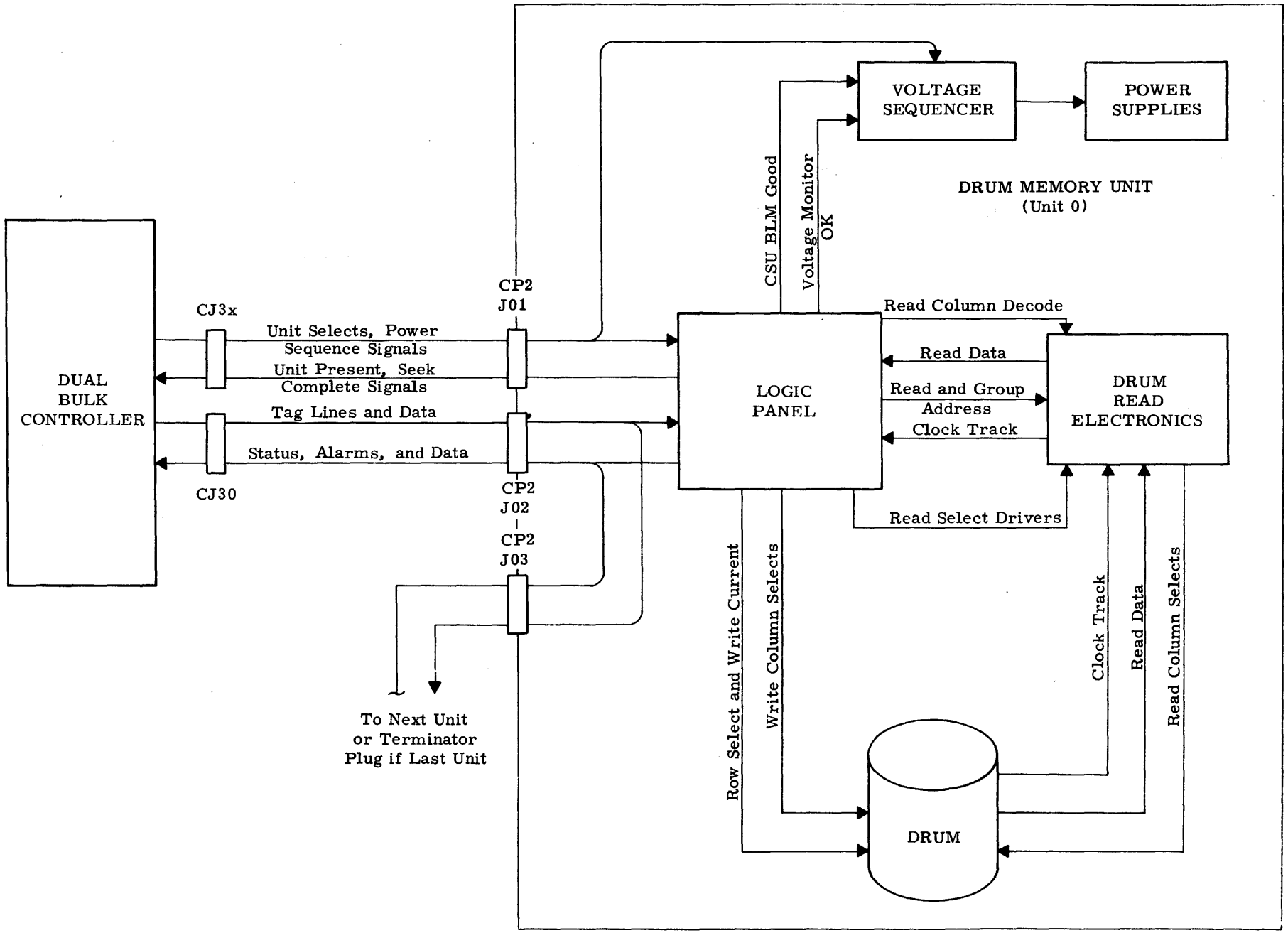


Fig. THEORY. 3 Simplified Block Diagram, Drum Memory

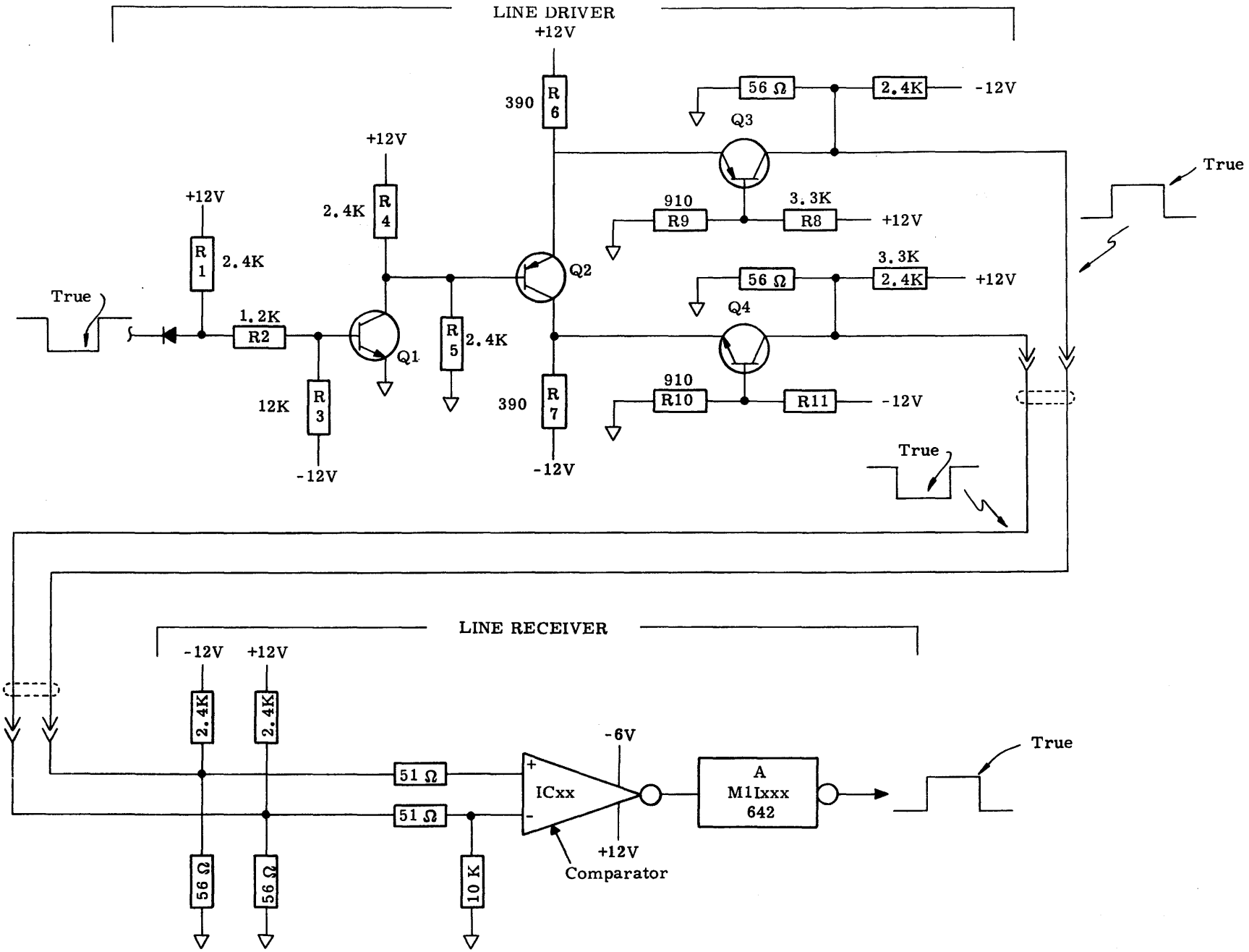
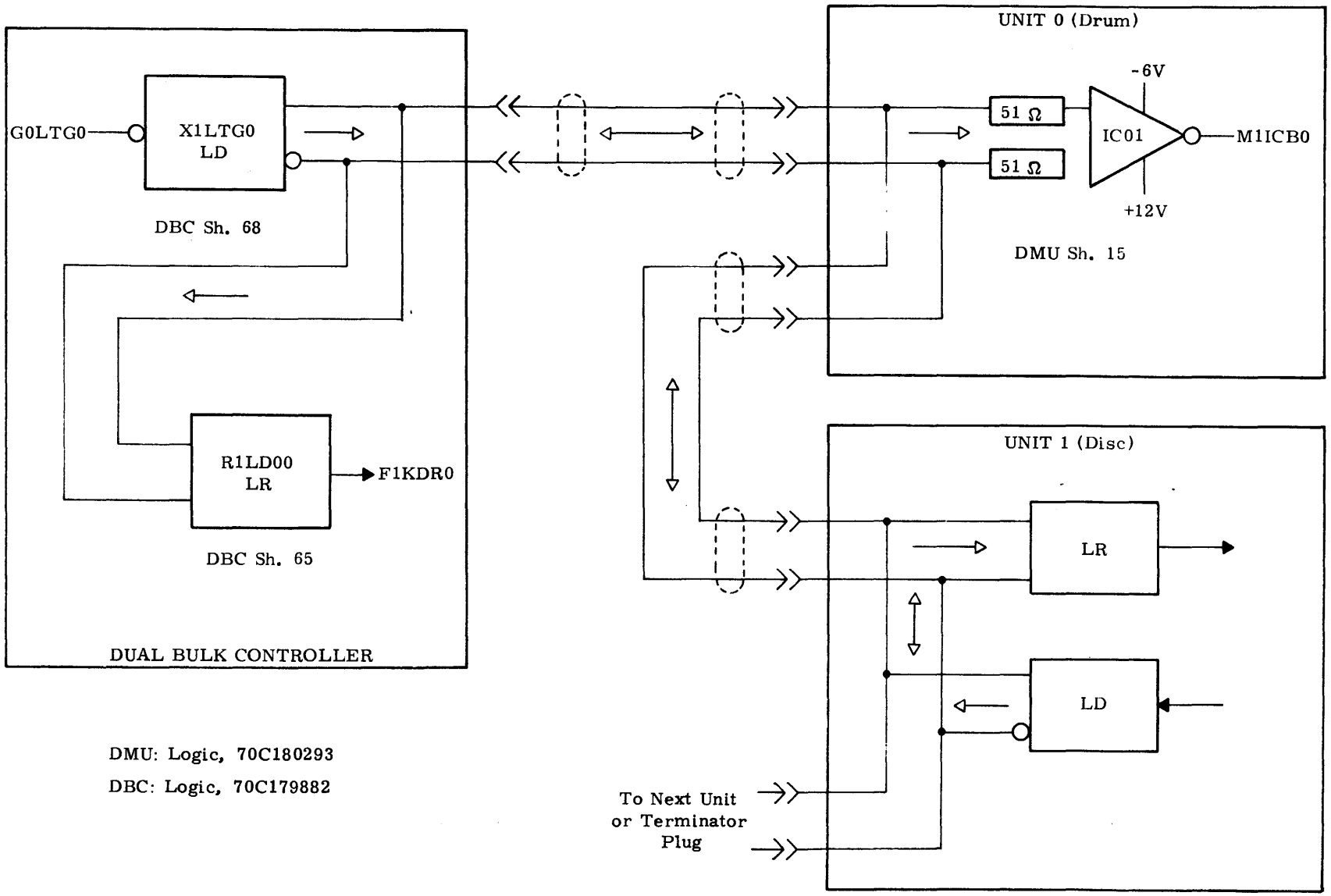


Fig. THEORY. 4 Two-Wire Signal Transmission



DMU: Logic, 70C180293

DBC: Logic, 70C179882

Fig. THEORY. 5 Example, Tag Line Interface

Unit Selected

Unit Selected is generated by the drum unit to indicate to the controller that the unit has responded to Select Unit. Gate G0DUUS (sh. 17 of -293 logic) is enabled when the unit is selected and the CSU BLM is good. The line driver appears on sheet 14. The signal is returned to the controller over the individual unit cable (sh. 3).

G0DUUS = $\overline{\text{BNG}}$ · SLU

BNG = CSU BLM not good.

ID Line

The ID Line identifies an attached unit to the controller as a drum or a disc unit. The line is on the individual unit cable. It is on pin 00H of the BJ3x connector on the Dual Bulk Controller. If the attached unit is a disc unit, the line will be at +12V. If it is a drum unit, the line is jumpered to -12V in the drum unit. Note on sheet 3 of the -293 logic that CP1-J01, pin 00H is jumpered to -12V from the controller at CP1-J01, pin 00J. This jumper also appears on sheet 13 of the logic.

Read Group (Cylinder) Select

Read Group Select resets the drum unit address register and disables the drum column selects. It is generated by the controller during sequence state 5 of a transfer sequence. For drum units, its purpose is to initialize the address register, prior to loading a new starting address.

This signal is transferred over the inter-unit cable. The line receiver appears on sheet 16 of the -293 logic. M1IRDG enables G0DURG (sh. 18), if the unit is selected, is on-line, and Control Select is not present.

G0DURG = $\overline{\text{UCS}}$ · OLS · RDG

$\overline{\text{G0DUCS}}$ = Control Select not present.

G0DURG is inverted by D1AURG (sh. 25), which in turn enables D0ARG1 and D0ARG2. The address register, F1AR08 through F1AR00 (sh. 24 and 25), is cleared through the DC reset inputs.

While RG1 and RG2 are "true" (0V), the column select enable drivers, D0CSE0, SE1, SE2, and SE3 (sh. 27-29) are disabled.

Group (Cylinder) Select

This signal gates a beginning group address into bits 08 through 04 of the drum unit address register, sets the seek complete flip-flop, and disables the On Group signal. It is generated by the controller during sequence state 5 of a transfer sequence, following Read Group Select.

The signal is transferred over the inter-unit cable. The line receiver appears on sheet 15 of the -293 logic. M1IGPS enables G0DUGS (sh. 18), if the unit is selected, is on-line, and Control Select is not present.

G0DUGS = GPS · OLS · $\overline{\text{UCS}}$

G0DUGS makes the disable drive return signal, G1DDDR, "true", which disables the column select enable drivers, SE0 through SE3 (sh. 27 - 29).

The G0DUGS pulse is inverted by N1DUGS which partially enables G0DSA8 through G0DSA4. SA8 through SA4 are enabled according to the data contained on tag lines CB4 through CB0, gating the group address into bits AR08 through AR04 of the address register (sh. 24 and 25), through the DC set inputs.

The unit On Group gate, G0DUOG (sh. 17), is disabled by G0DUGS. G0DUGS also sets the seek complete flip-flop, F1DCSD (refer to the "Seek Complete" heading).

Head Select

This signal gates the head address into bits 03 through 00 of the drum unit address register. It is generated by the controller during sequence state 5 of a transfer sequence, following Group Select.

The signal is transferred over the inter-unit cable. The line receiver appears on sheet 15 of the -293 logic. M1IHDS enables G0DUHS (sh. 18), if the unit is selected, is on-line, and Control Select is not present.

G0DUHS = HDS · OLS · $\overline{\text{UCS}}$

G0DUHS makes the disable drive return signal, G1DDDR, "true", which disables the column select enable drivers, SE0 through SE3.

The G0DUHS pulse is inverted by N1DUHS which partially enables G0DSA3 through G0DSA0. SA3 through SA0 are enabled according to the data contained on tag lines CB3 through CB0, gating the group address into bits AR03 through AR00 of the address register, through the DC set inputs.

Difference Select

This signal, which is generated by the controller during sequence state 5, following head select, has no effect on drum units. Difference Select is fed back out to the next unit, as shown on sheet 16 of the -293 logic.

Control Select

Control Select specifies the kind of operation to be performed by the bulk memory units. Control Select is generated by the controller near the end of sequence state 5, following Difference Select, and will affect only the unit dictated by the Unit Select signal. Control Select then remains "true" until the operation is complete. When Control Select is true, one or more tag line signals must also be true to specify the action to be taken. The actions are defined as follows:

- Control Select and Tag Line 0 specify a write operation. Unit control select gate, G0DUCS (sh. 18 of -293 logic), is enabled when the unit is on-line and selected and when Control Select is present at the output of M1ICTL. G0DUCS is inverted by N1DUCS, and UCS and Tag Line 0 enable the write driver, D0DWRT (sh. 19). D0DWRT partially enables the write amplifier enable gates (sh. 30 - 37), as well as other logic involved in a write operation.

  D0DWRT = CB0 · UCS · $\overline{\text{FUA}}$
  F0DFUA = No file unsafe alert.

- Control Select and Tag Line 1 specify a read operation. The unit control select signal and Tag Line 1 enable the read driver, D0DRED (sh. 19). D0DRED partially enables the read select drivers (sh. 30 - 37), as well as other logic involved in a read operation.

- Control Select and Tag Line 2 specify a seek forward operation for disc units and have no effect on drum units.

- Control Select and Tag Line 3 reset the head register in disc units and have no effect on drum units.

- Control Select and Tag Line 4 specify an erase operation in disc units and have no effect on drum units.

- Control Select and Tag Line 5 specify a seek reverse operation on disc units and have no effect on drum units.

- Control Select and Tag Line 6 specify a return to zero seek. On a drum unit with no read or write operation in progress, return to zero seek generates a Seek Complete signal for return to the controller. The unit control select signal and Tag Line 6 enable G0DRTZ (sh. 18), if neither a read nor write operation is in progress. The G0DRTZ pulse sets seek complete flip-flop, F1SCD (sh. 17), if it has not already been set by unit group select gate, G0DUGS (refer to the "Seek Complete" heading).

  G0DRTZ = CB6 · $\overline{\text{WRT}}$ · $\overline{\text{RED}}$ · UCS

- Control Select and Tag Line 7 specify a head advance at the next index mark. The drum address register is incremented by one at the next index mark, selecting the next head in the address sequence, so that a transfer of a length greater than one 1024-word track may continue. The unit control select signal and Tag Line 7 enable both D0DITA and G0DIAR (sh. 18). D0DITA remains enabled until Tag Line 7 goes "false", but IAR remains enabled only until capacitor C25 discharges sufficiently to allow ITA to disable IAR. This limits the IAR pulse width to about 100 nanoseconds. The D0DITA pulse disables column select enable gates SE0 through SE3, via G1DDDR and N1DDDR. They remain disabled long enough to allow the address register to increment. The 0V IAR pulse is inverted by D1AIAR, (sh. 24), and the positive IAR pulse is applied to the clock inputs of all address register flip-flops, incrementing the register at the trailing edge.

  D0DITA = CB7 · UIM · UCS

  G0DIAR = CB7 · UIM · UCS · $\overline{\text{ITA}}$

  F1DUIM = Unit Index Mark

File Unsafe Alert

This signal is generated by the drum unit to inform the controller when it is inoperative. As is described under "Power Supplies and Controls", the file unsafe flip-flop, F1DFUA (sh. 17), is set if either of the 24V power supplies is out of tolerance. If so, G0DAVA is enabled, which sets FUA. If both the read driver and the write driver have been turned on, G0DRWF (sh. 19) is enabled, which sets FUA. F1DFUA may be reset by the control voltage monitor, Y0VVMD, during a power-on sequence, or by a return to zero operation (UCS · CB6).

F1DFUA = AVA · RWF

G0DRWF = CB1 · WRT + CB0 · RED

Seek Alert

This signal is generated by disc units to indicate that a unit could not complete a seek. The signal from other units is jumpered through drum units as shown on sheet 16 of the -293 logic.

End of Cylinder

This signal is generated by disc units to inform the controller that the end of a cylinder has been reached. The signal is jumpered through drum units as shown on sheet 16 of the -293 logic.

On Group (Cylinder)

This signal is generated by a drum unit to inform the controller that it is not in the process of storing a new group address. It is generated by a unit which is selected and on-line, at any time when the Group Select signal is not present. The unit on group gate, G0DUOG (sh. 17 of the -293 logic), enables line driver X1IGOP. The signal is transferred over the inter-unit cable.

G0DUOG = OLS · $\overline{\text{UGS}}$

On Line

This signal informs the controller that the unit is on-line with heads loaded, is selected, and has not received a "BLM not good" signal from the controller

(refer to the "Power Supplies and Controls" heading). The unit on-line gate, G0DUOL (sh. 17), enables line driver X1IONL. The signal is transferred over the inter-unit cable.

$$G0DUOL = \overline{BNG} \cdot ONL \cdot SLU$$

Seek Complete

This signal informs the controller that the drum unit has responded to a group select or return-to-zero seek instruction. For a disc unit, seek complete indicates that the heads have completed the process of moving to a new location as the result of a normal seek resulting from an OUT instruction, or a return-to-zero seek resulting from an OPR instruction. Since the drum heads do not move in response to such instructions, seek complete is generated shortly after the receipt of the instruction. The seek complete signal resets the seek flip-flop, in the controller, which represents the unit selected. When it resets, the flip-flop generates a seek complete API at the level representing the selected unit and holds the corresponding line to the Arithmetic Unit jump logic in the "complete/ready" state for testing by JNR instructions.

If the seek complete signal from a drum unit is in response to an OUT instruction, it occurs when the unit group select signal, G0DUGS, sets the complete flip-flop, F1DSCD (sh. 17). Since the controller is still in sequence state 5, the unit control select signal is not yet present and the unit seek complete gate, G0DUSC, is enabled.

$$G0DUSC = SCD \cdot \overline{USC}$$

A return-to-zero seek instruction also sets F1DSCD. In response to an OPR instruction, encoded to select this unit, the controller first turns on Control Select and then Tag Line 6. As previously described, this enables G0DRTZ, which sets F1DSCD. Control Select remains on for about the duration of the OPR instruction. About 100 nanoseconds later, capacitor C9 on the UGS run to G0DUSC will have charged to a sufficiently positive voltage to enable the unit seek complete gate.

G0DUSC enables line driver X1ISKC which transmits the signal to the controller over the individual unit cable.

F1DCSD is reset by D0DRED when the controller next request a read operation. This will occur in the current or the next transfer sequence when the controller requests a read operation to find the desired sector on the addressed track.

SECTOR MARK AND INDEX MARK RECOVERY

Three timing tracks are pre-recorded at the factory on each drum. All of the 29,184 bits positions on these tracks contain "ones" except for three consecutive "zeros" which identify the index mark at the beginning of sector 0, and 15 "zeros" which mark the beginning of sectors 1_8 through 17_8. One of the timing tracks is active and the remaining two tracks are reserved as spares.

The outputs of the heads on these three tracks are continuously amplified by a pre-amplifier in the drum assembly. The three amplifier outputs are carried by a cable from timing connector J02 on the drum to the Drum Read Electronics (DRE) assembly. A timing amplifier board in the DRE is positioned in one of three slots to monitor one of the three clock tracks. Refer to the DRE logic, drawing no. 70A180119, sheets 9 through 11.

The analog timing track signal is demodulated by the DRE timing amplifier, and the demodulated signal is routed to the drum unit logic panel, where it is further decoded to develop drum unit clock pulses, and the sector mark and index mark pulses. The sector and index mark pulses are used by the controller to synchronize the controller logic with the rotating drum.

Clock Track Demodulator

The timing amplifier board in the DRE demodulates the analog timing track signal, converting it to digital pulses. The principal waveforms appear on Fig. THEORY. 6. A complete schematic of the circuitry involved appears on page 3.0 of the timing amplifier PWB drawing, PX1300XTTA1. The logic is on sheet 9 of the -119 drawing.

The analog signal is amplified by an integrated circuit amplifier which provides a voltage gain of about 10:1. It then passes through an amplifier whose frequency response is controlled by an RC filter in a feedback circuit. The second amplifier serves as a low pass filter, which provides good response to signals in the drum readback frequency range (1.7 - 3.5 MHz) but rejects high frequency noise signals.

The signal is then applied across a transformer with a center tapped secondary to an integrated circuit comparator, which operates as a zero crossing detector. The output of the comparator is a square wave whose up and down transitions correspond to each zero crossing by the analog signal (each time the signal goes through phase angles 0° and 180°).

The comparator output is then differentiated by another transformer with a center tapped secondary, and positive pulses corresponding to each zero crossing are applied to a line driver. The 0° and 180° pulses are ORed together by two diodes on the secondary which operate in the same manner as a full-wave rectifier.

Drum Unit Clock Generation

As "ones" are read from the active timing track, the zero crossing pulses received in the logic panel from the DRE occur at the rate of two pulses per bit time, or 578ns/2 = 289ns between pulses. When the "zeros" at the index and sector mark positions appear, the time between pulses is 578 nanoseconds. The logic

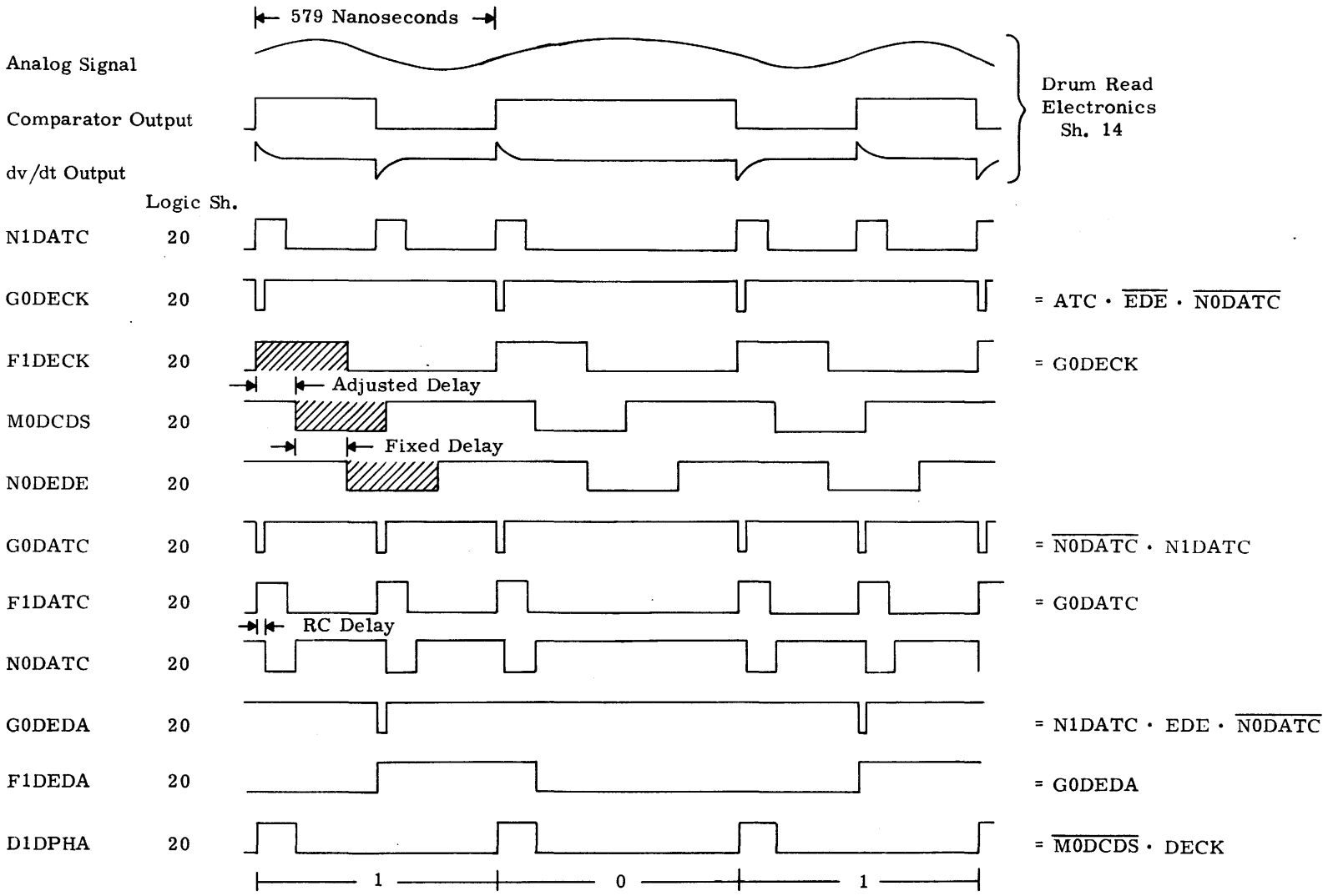


Fig. THEORY. 6 Clock Pulse Generation

on sheet 20 of the -293 drawing generates a "phase A" clock pulse (PHA) which corresponds to the first pulse in each clock track bit period. When the bit cell contains a "one", a second pulse appears at the middle of the bit period, and that pulse causes the logic on sheet 20 to generate an "envelope data" pulse (EDA). These two signals are used by the sector mark detection logic, as described under the next heading. Fig. THEORY. 6 illustrates the development of the PHA and EDA pulses.

The output of the clock track line driver in the DRE is applied to the "clock from drum" line receiver, M1DCFD, and positive going pulses which mark each 0° and 180° transition of the analog data appear at the output of the line receiver and N1DATC. The data envelope clock flip-flop, F1DECK, is set at the leading edge of the first ATC in each bit period. The DECK pulse enables G0DCDS because NODEDE is not then enabled, and delay line driver X1DCDS starts a pulse on its way through delay line DL1.

While the pulse is traveling through DL1, G0DPHA is enabled by the DECK pulse. The PHA pulse terminates in about 90 nanoseconds, when the delayed pulse enables MODCDS. The delayed pulse then enters DL2, and after an additional delay of 100 nanoseconds, plus the gate delay time, N0DEDE is enabled. The total delay of the EDE pulse is selected by a pin jack pair on one of the taps on DL1, to cause the EDE pulse to bracket the time position of the next expected ATC pulse. If the second ATC pulse appears in the correct time position, F1DEDA is set, indicating that the current bit is a "one".

Gate N0DATC controls the duration of the pulses which set F1DECK and F1DEDA. Several nanoseconds after F1DATC is set, N0DATC is enabled. N0DATC also remains enabled several nanoseconds after F1DATC clears. These delays are caused by C28, which must charge or discharge to each new level, through R11, before the state of N0DATC change. In effect, N0DATC is the F1DATC pulse, inverted and delayed.

Note that F1DECK is always prevented from setting at the middle of a bit period because the EDE pulse is present at that time, disabling G0DECK. Also note that F1DEDA is always prevented from setting at the beginning of a bit period because EDE is not true at that time, disabling G0DEDA.

Sector and Index Mark Detection

The sector and index mark pulses are developed on sheet 21 of the -293 logic. Two examples appear on the timing diagram, Fig. THEORY. 7; first the detection of a sector mark, then the detection of the index mark.

The unit sector mark flip-flop, F1DUSM, and the unit index mark flip-flop, F1DUIM, remain clear as ones are read because EDA brackets each PHA pulse. F1DUSM is set at the trailing edge of the first PHA when EDA is clear, and that occurs during each sector mark bit period.

F1DT00 and F1DT01 count the PHA pulses which follow the setting of F1DUSM. If the third bit period following the first "zero" also contains a "zero", F1DUIM sets.

If the drum unit is on-line and selected, G0DUSM will be enabled when a sector mark is detected, and both G0DUSM and G0DUIM will be enabled when an index mark is detected. These gates enable line drivers which transfer the sector and index mark pulses to the controller over the inter-unit cable.

ADDRESSING AND HEAD SELECTION

The program specifies the first track on which data is to be written, or from which data is to be read, by placing the address of that track in control word one (CW1), prior to initiating a transfer sequence. As is indicated on Fig. THEORY. 8, the starting track (head) address is contained in bits 12 through 4 of CW1. The remaining bits in CW1 are used by the controller, only, and are not transferred to the drum units.

The drum heads are arranged in an electrical matrix, consisting of up to 16 rows, and up to 32 columns. Any one of the heads may be selected by decoding the nine bits in the address register to select the row and column intersection to which the head is connected.

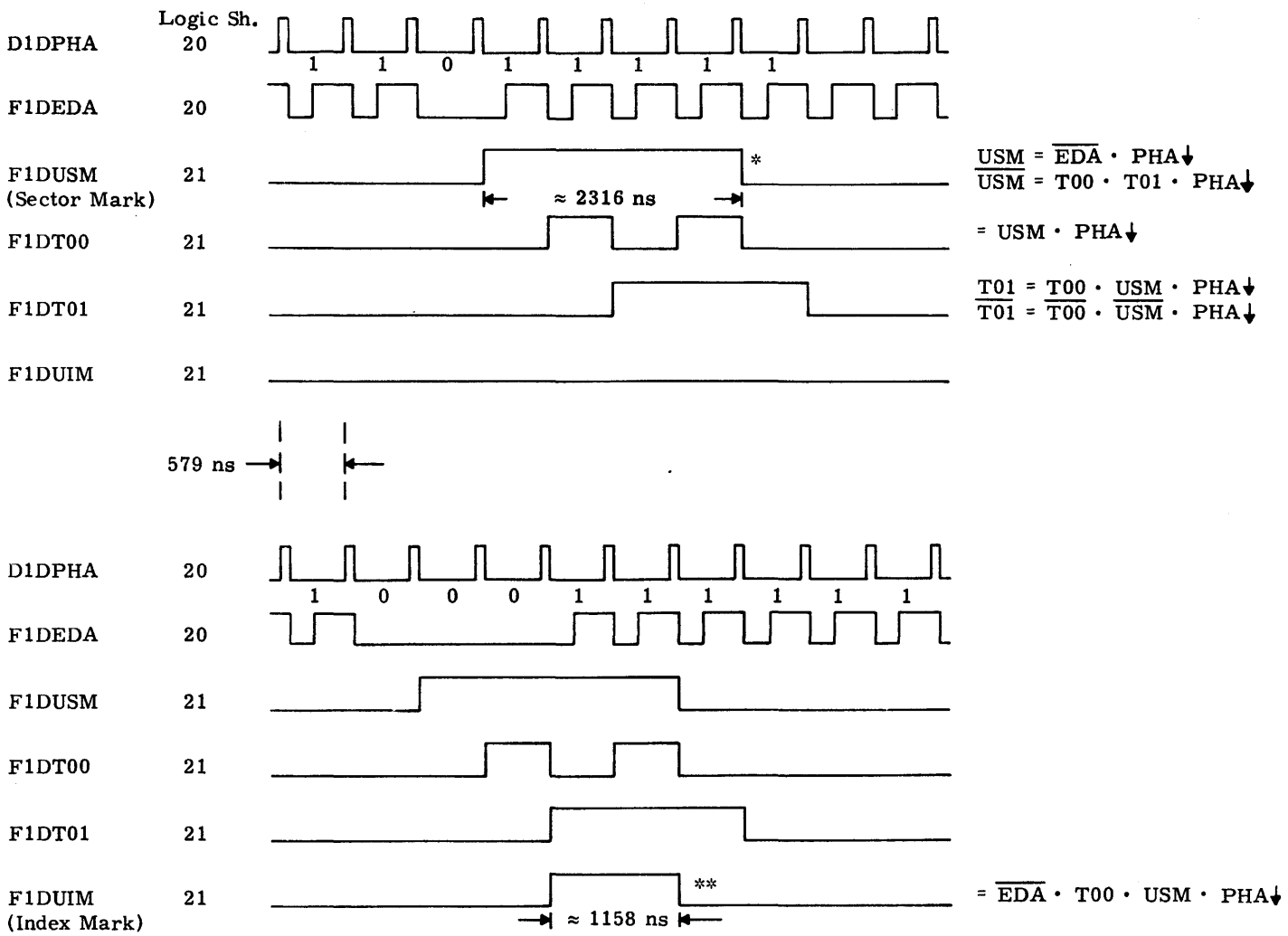
When writing on the drum, the row select logic directs the writing current to the addressed row, while the column select logic provides the ground return to the center tap on the selected head. When reading from the drum, the row select logic enables the appropriate read select drivers in the DRE, and the read column select drivers in the DRE provide the ground return from the center tap on the selected head.

Table THEORY. 1 provides several examples of the correlation between starting track addresses contained in CW1 and the column and row addresses of the corresponding heads. Note that bits 3 through 0 of CW1 have no effect on the head selection, as they contain the sector address, which is not transferred to drum units, but used by the controller to find the starting sector. Also note that, while CW1 provides for the specification of groups 0 - 37_8, heads 0 - 17_8, and sectors 0 - 17_8, the octal representation for group 37_8, head 17_8, sector 17_8, is 17777. This is due to the use of only 13 binary bits to represent the six octal digits (37, 17, 17_8).

The table also shows how the row and column select logic is implemented for the six drum storage sizes. Sheet 12 of the -293 logic indicates the row and column select PWB implementation scheme.

Address Register

The drum unit address register appears on sheets 24 and 25 of the -293 logic. As previously described, the 9-bit register (AR08 - AR00), is cleared early in a transfer sequence, as the Read Group Select signal



* 1.04 millisec. between pulses.
** 16.67 millisec. between pulses.

Fig. THEORY. 7 Sector and Index Marks

from the controller enables D0ARG1 and D0ARG2, which clear the register through the DC clear inputs.

The starting track (head) address is gated into the DC set inputs to the 9 stages, by the Group Select and Head Select signals from the controller. The group and head addresses are transferred on the tag lines as indicated on Fig. THEORY. 8.

When the controller requires a transfer of a length greater than one track (16 sectors or 1024 words), it issues a head advance instruction (Control Select and Tag Line 7) which makes D1AIAR (increment address register) go true at the next index mark (refer to "Control Select"). The index mark identifies the end of sector 17_8 on the current track and the beginning of sector 0. The address register, which is interconnected as a normal binary counter, is incremented at the trailing edge of the IAR pulse.

The incrementation of the address register can continue until the head with the highest row and column address, implemented by the unit, is selected (if less than row-17, column-37). If the controller tries to increment the address beyond the last head, an address compare alert should be detected in the controller, because logic does not exist to decode the row and column address, and no head can be selected.

On drum units implementing the maximum of 512 heads (including row-17, column-37), the address register can (but would not normally) be incremented after it has been set to 777_8, in which case, it will revert to 000 (row 0, column 0).

The maximum CW1 addresses, indicated in Table THEORY. 1, would allow for a transfer of the final sector (64 words) on the last track. In normal operation, the starting address is set much lower, and transfers of up to 255 sectors (16,320 words) are made.

The equations provided on Fig. THEORY. 8 can be used to decode any head address. For example, assume that CW1 calls for a starting address of 13760_8. CW1 bits 12 through 4 correspond to drum address register bits AR08 through AR00, so they will contain the following:

	12	11	10	9	8	7	6	5	4	3	2	1	0
CW1 or DBC	1	0	1	1	1	1	1	1	1	0	0	0	0

	8	7	6	5	4	3	2	1	0
Add. Reg.	1	0	1	1	1	1	1	1	1

The only address register bit not set is AR07. This provides a head address of row-13/column-37 (see Fig. THEORY. 8). The address register contents can be decoded mentally to determine the octal head number: 577. $577_8 = 383_{10}$. Counting head 000_8 as number one, then, this head becomes number 384. Head number 777_8 is number 512_{10}.

Write Column Selection

Address register bits AR06 through AR02 are decoded by the logic on sheets 26 through 29 on the -293 drawing to enable the write return drivers. The write return drivers provide the write current ground return from the selected head center tap as shown by Fig. THEORY. 9.

The enabling of write return driver, D0CS37 (sh. 29), is described in the following example. All other drivers are enabled in the same manner, per the logic equations shown on the logic drawing and on Fig. THEORY. 8.

Assume that address register bits AR06 through AR02 all equal one. If the address register is not being reset, and if no new address is being gated in, column select enable gate, D0CSE3 is enabled.

$$D0CSE3 = \overline{RG2} \cdot AR05 \cdot AR06 \cdot \overline{DDR}$$

D0ARG2 = URG = Read Group Select (resets add. reg.)

N0DDDR = Head Select + Group Select

$$G1CS37 = 304 \cdot 302 \cdot 301 \cdot SE3 = R04 \cdot R03 \cdot R02 \cdot SE3$$

G1CS37, then, enables write return line driver D0CS37, which provides a return path for the write current directed to one of the rows in column 37. A schematic of the discrete component line driver appears on sheet 51 of the -293 logic. It also appears in partial form on Fig. THEORY. 10. When enabled, transistor Q2 will be conducting, providing a path to ground for the current.

Write Row Selection

When the drum unit is in the write mode (Control Select and Tag Line 0), the write row selects enable one of 16 pairs of write amplifiers, which deliver write current according to data received on the write data line from the controller. The write row select logic appears on sheets 30 through 37 of the -293 drawing. Fig. THEORY. 9 provides a simplified diagram of the selection scheme.

The enabling of the row-00 write amplifiers (sh. 30) is described in the following example. All other write amplifiers are enabled in the same manner, per the logic equations shown on the logic drawing and on Fig. THEORY. 8.

Assume that address register bits AR08, AR07, AR01, and AR00 all equal zero. Also assume that the drum unit is in the write mode. Row-00 write amplifier drivers, D0R100 and D0R000 will be enabled by each data bit appearing on its corresponding write data line.

$$D0R100 = Write \cdot E00B \cdot \overline{R000} \cdot Data\ Line\ 1$$

$$D0R000 = Write \cdot E00B \cdot \overline{R000} \cdot Data\ Line\ 0$$

Number of 16-Head Groups Implemented	Rows	Columns	Heads (Rows X Columns)	Word Capacity	Maximum CW1 Address
2	00 - 03 (4)	00 - 07 (8)	32	32,768	00777
4	00 - 03 (4)	00 - 17 (16)	64	65,536	01777
8	00 - 03 (4)	00 - 37 (32)	128	131,072	03777
16	00 - 07 (8)	00 - 37 (32)	256	262,144	07777
24	00 - 13 (12)	00 - 37 (32)	384	393,216	13777
32	00 - 17 (16)	00 - 37 (32)	512	524,288	17777

CW1 (Bits 12 - 0)	Column	Row	Head (10)	CW1 (Bits 12 - 0)	Column	Row	Head (10)	Head (8)
17777	37	17	512	17170	31	17	488	747
17770	37	17	512	17070	30	17	484	743
17760	37	17	512	16770	27	17	480	737
17750	37	16	511	15770	17	17	448	677
17740	37	16	511	14770	07	17	416	637
17730	37	15	510	13760	37	13	384	577
17720	37	15	510	12760	27	13	352	537
17710	37	14	509	11760	17	13	320	477
17700	37	14	509	07760	37	07	256	377
17670	36	17	508	03760	37	03	128	177
17570	35	17	504	01760	17	03	64	077
17470	34	17	500	00760	07	03	32	037
17370	33	17	496	00020	00	01	2	001
17270	32	17	492	00000	00	00	1	000

Table THEORY. 1 Head Address Examples

$\overline{G0R000} = \overline{AR00}$

$G1RE00B = E00 = \overline{AR01} \cdot \overline{AR07} \cdot \overline{AR08}$

The row-00 write data shown entering the write amplifier input transformer on Fig. THEORY. 9 is the output of the row-00 write amplifier drivers. Fig. THEORY. 9 is a simplified schematic of the discrete component write circuit for row-00/column-01. All 16 write-row circuits and all 32 column selects circuits are similar. Detailed schematics of the discrete component circuits involved are provided on sheet 51 of the -293 logic, and on the PWB drawings, PX1000XRSA5 and PX1000XCSA5. A more detailed description is provided under the "Write Operation" heading.

Read Column Selection

The read column select logic enables the read return select drivers to provide a ground return path from the center tap of the selected head, for the read select current. The read select current is produced by the read row select logic. Fig. THEORY. 10 illustrates the read select scheme.

Read column selection involves logic on sheets 15 through 20 of the DRE logic, 70C180119, and on sheet 25 of the Drum Unit logic, 70C180293. The enabling of read select return driver D0SR07 is described in the following example. All other drivers are enabled in the same manner, per the logic equations on Fig. THEORY. 8.

Assume that address register bits AR06 and AR05 equal zero, and AR04, 03, and 02 all equal one. The column - 7 read select return driver, G0SR07 (sh. 17 of -119 logic) will be enabled as shown by the following equations:

$G0SR07 = R0X \cdot 004 \cdot 002 \cdot 001$

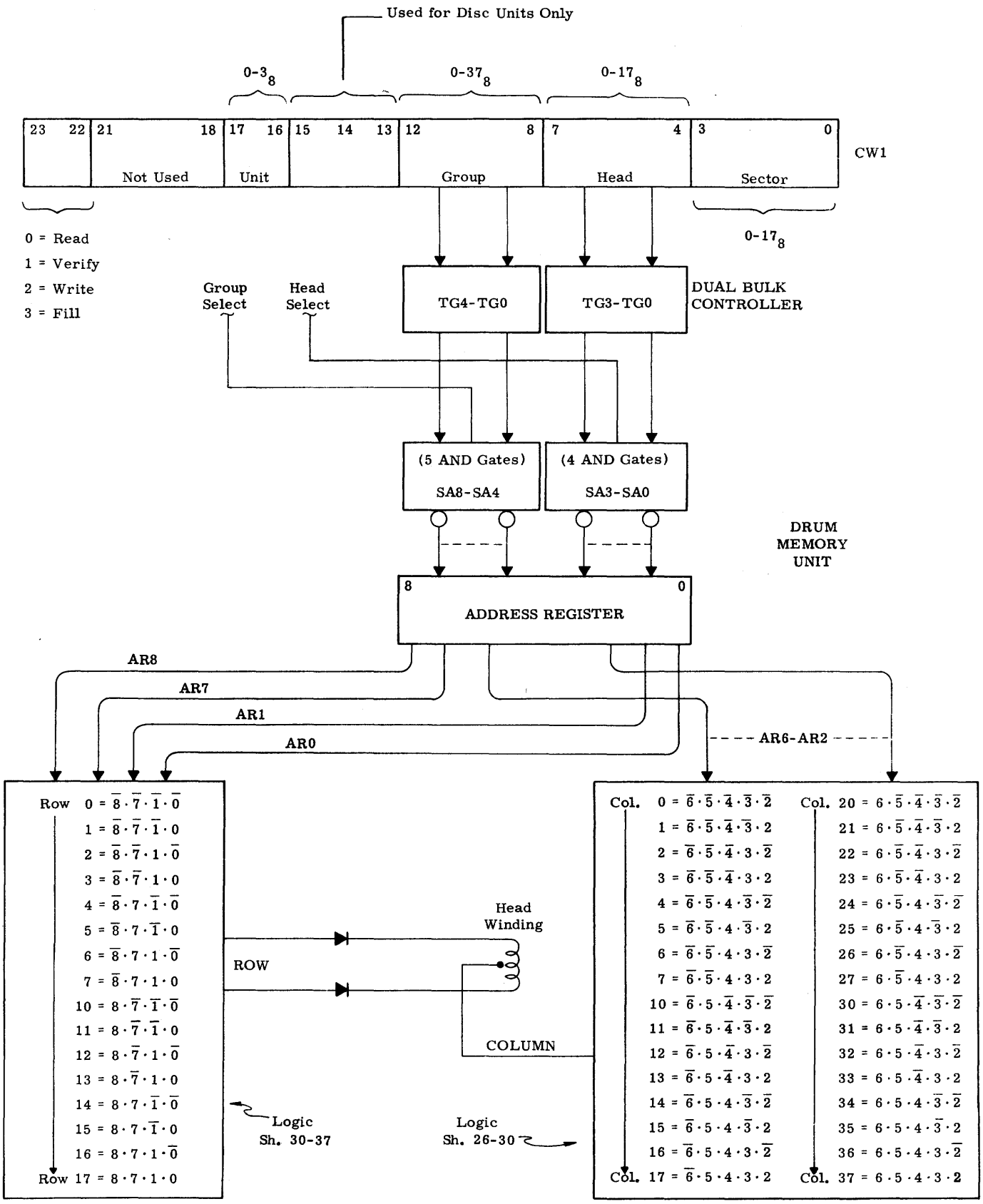


Fig. THEORY. 8 Head Addressing

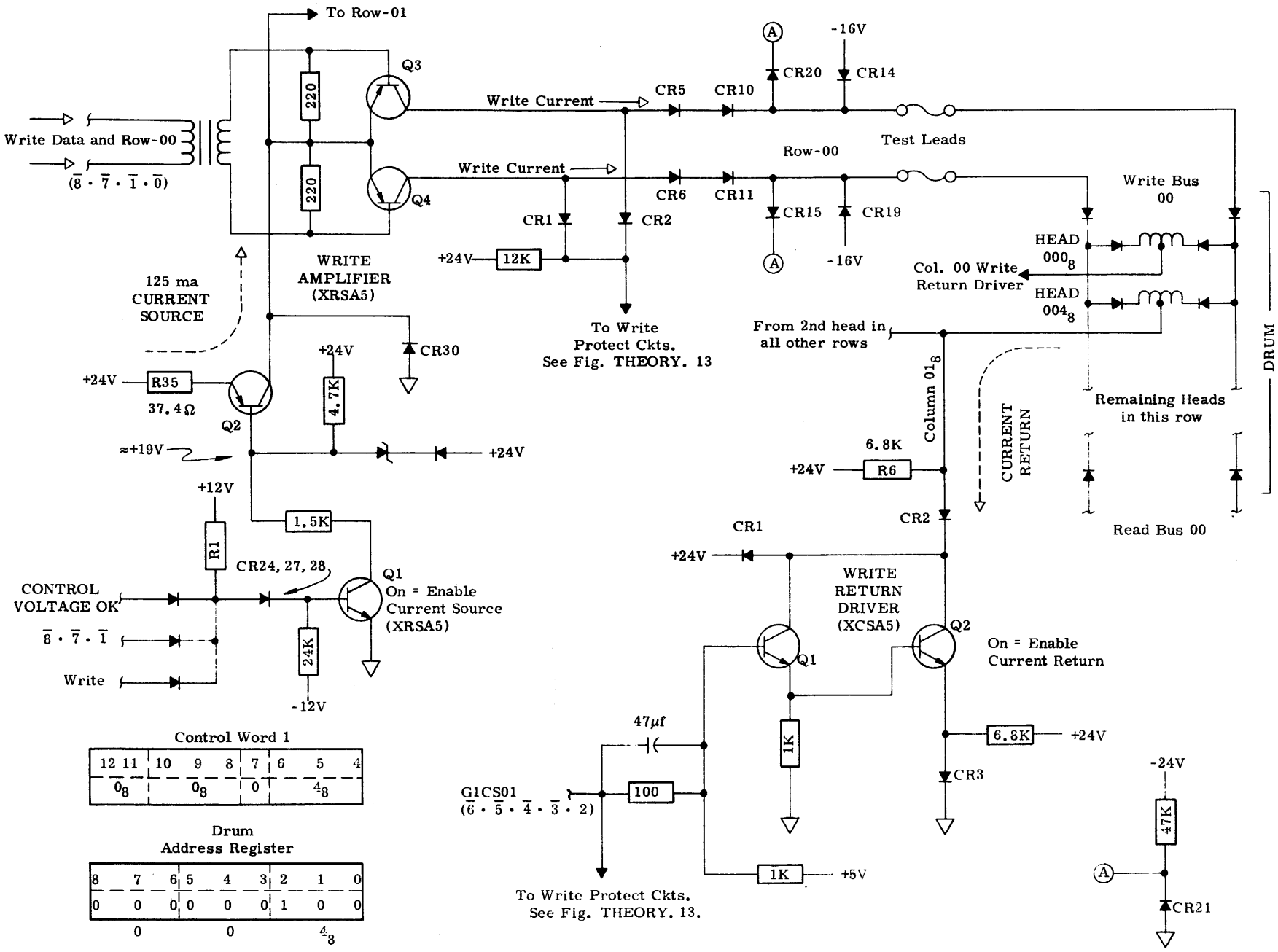


Fig. THEORY. 9 Row-00, Column- 01 Write Circuit

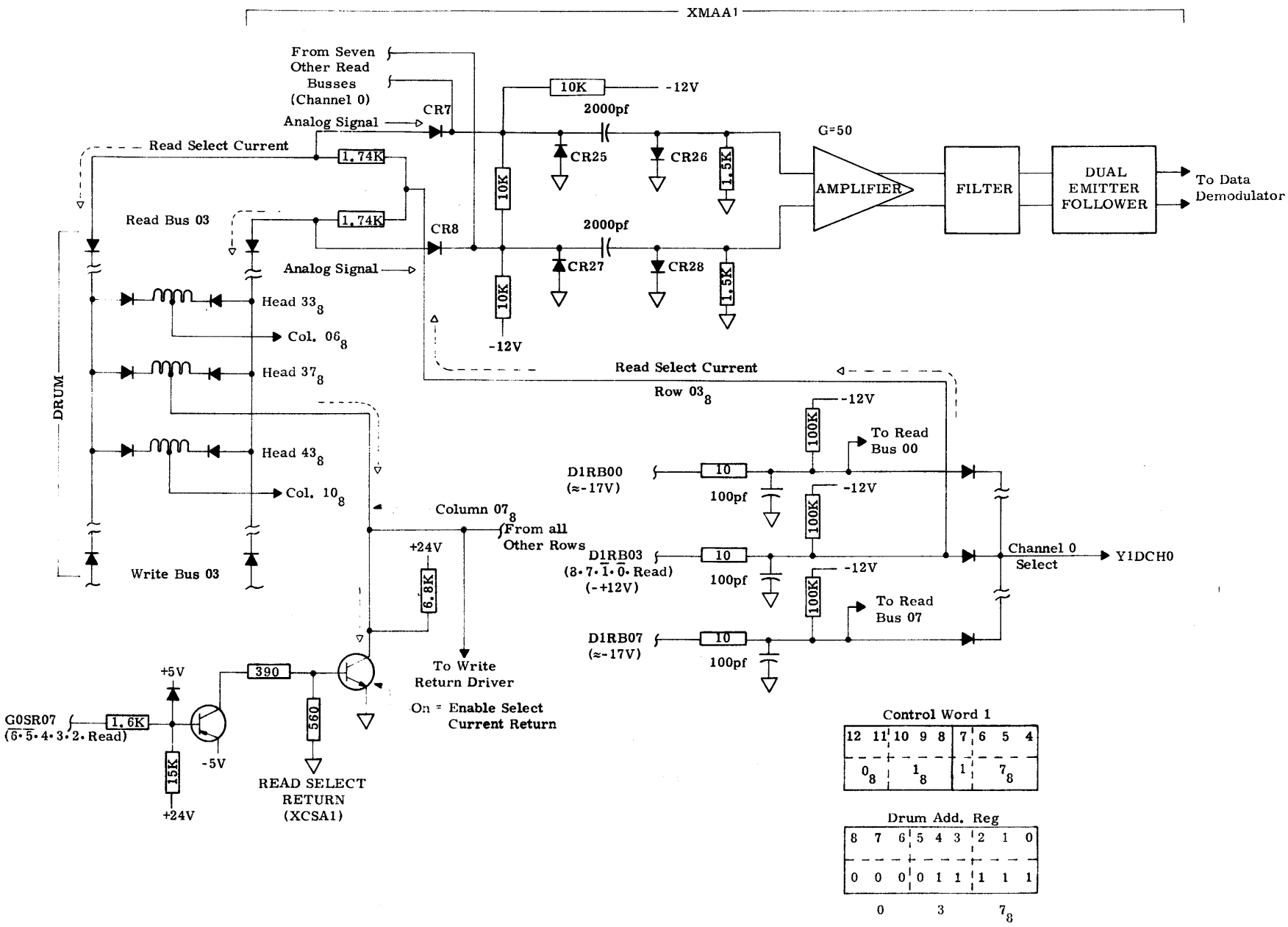


Fig. THEORY. 10 Row-03, Column-07 Read Circuit

D1SR0X = AEG0 = $\overline{AR06} \cdot \overline{AR05} \cdot$ Read (sh. 25 of -293 logic).

D1S004, 002, 001 = AR04; AR03, AR02, respectively.

The read select return circuit appears on Fig. THEORY. 10 and on sheet 21 of the -119 logic. The two transistors are normally not conducting, but when G0SR07 is enabled, the zero volt signal provides current to turn the PNP transistor on, and that provides +5V at its collector, which draws current to turn on the NPN transistor. The read select current then has a path to ground through the NPN transistor. The write return driver, which is connected to the same head-center taps, is turned on at the same time (Fig. THEORY. 9).

Read Row Selection

The read row select logic selects the analog signal output signal from the selected read bus and applies that signal to the data demodulation circuits. One head on the selected bus provides the analog signal, because the head matrix diodes on the drum, for that head only, are enabled by the read select current, which returns to ground through the read select return driver. Fig. THEORY. 10 provides a simplified schematic of the circuits involved in selecting the head.

The enabling of the row-03 read select driver is described in the following example. All other read select drivers are enabled in the same manner, per the equations on Fig. THEORY. 8 and on sheets 30 through 37 of the -293 logic.

Assume that address register bits AR08 and AR07 equal one, and AR01 and AR00 equal zero. The row-3 read select driver, D1RB03 (sh. 31 of -293 logic), will be enabled.

D1RB03 = 001 · E01A · Read

G1R001 = AR00

G1RE01A = AR01 · $\overline{AR07} \cdot \overline{AR08}$

The read select driver is a discrete component circuit, a schematic of which appears on sheet 51 of the -293 logic. When disabled, the output rests at about -17V, which reverse biases the diodes in the row select logic in the DRE. In the example used on Fig. THEORY. 10, D1RB03 is enabled, and applies +12V to the row-3 select logic.

The read row selection occurs on a PX1300XMAA1 board in the DRE. In units implementing up to 256 heads, only one of these boards with row selects 00 through 07 is employed. In larger drums, rows 10 through 17 are selected by a second XMAA1 board. The logic symbols for these boards appear on sheets 12 and 13 of the -119 logic. Detailed schematics appear on PWB drawing no. PX1300XMAA1. A simplified schematic of the row-3 select circuit appears on Fig. THEORY. 10.

The positive output of D1RB03, in the example shown, forward biases the read bus 03 diodes in the drum and the diodes on the one head in row-03 which is connected to an enabled read select return (column-07). Diodes CR7 and CR8 in the amplifier input circuit are also forward biased. The analog signal read from the drum track by head 037_8 can pass through the forward biased diodes and through the coupling capacitors in the amplifier input circuit to the amplifier. The capacitors decouple the DC read select voltage from the amplifier input.

If any read select driver for the first 8 rows is enabled, the channel 0 select line to the data demodulator goes "true". If the second 8-row channel is implemented, any enabled row in that channel will make channel 1 select "true".

WRITE OPERATION

When the controller calls for a write operation, by turning on Control Select and Tag Line 0, it places the data to be written on the write line in the inter-unit cable. The data as it appears on the write line from the controller, and the data as it is written on the drum are of differing forms.

Data generated by the controller consist of clock pulses one-quarter of a bit period in duration. When writing a zero, one clock pulse at the beginning of the bit period is placed on the write line. When writing a one, a second clock pulse is added which begins at the middle of the bit period.

The logic on sheet 19 of the -293 drawing converts the clock pulse write data to the dual frequency data written on the drum (0.865 MHz = zero; 1.73 MHz = one). Fig. THEORY. 11 is a timing diagram which illustrates the operation of the data conversion logic. The outputs of F1DWDL are applied to the write row selects (sh. 30 - 37).

As described under the "Write Row Selects" heading, the write data are applied to the write amplifier of the selected row. The current source for the two rows on each XRSA5 row select PWB is also enabled, as the AR08, AR07, and AR01 address register bits enable one of the inputs to AND gate, which enables the current source.

Fig. THEORY. 9 is a simplified schematic of the write circuitry involved in writing on head 004_8 (row-00/column-01). The operation of the circuit shown is typical of the write circuits for all heads.

The current source enable transistor (Q1 of Fig. THEORY. 9) is off unless selected by address register bits 8, 7, and 1. If the voltage monitor indicates that the drum unit voltages are OK, if the unit is in the write mode, and if one of the two rows on the PWB is selected, sufficient current is drawn across R1 to turn Q1 on. Q2 then, operates as a constant current source.

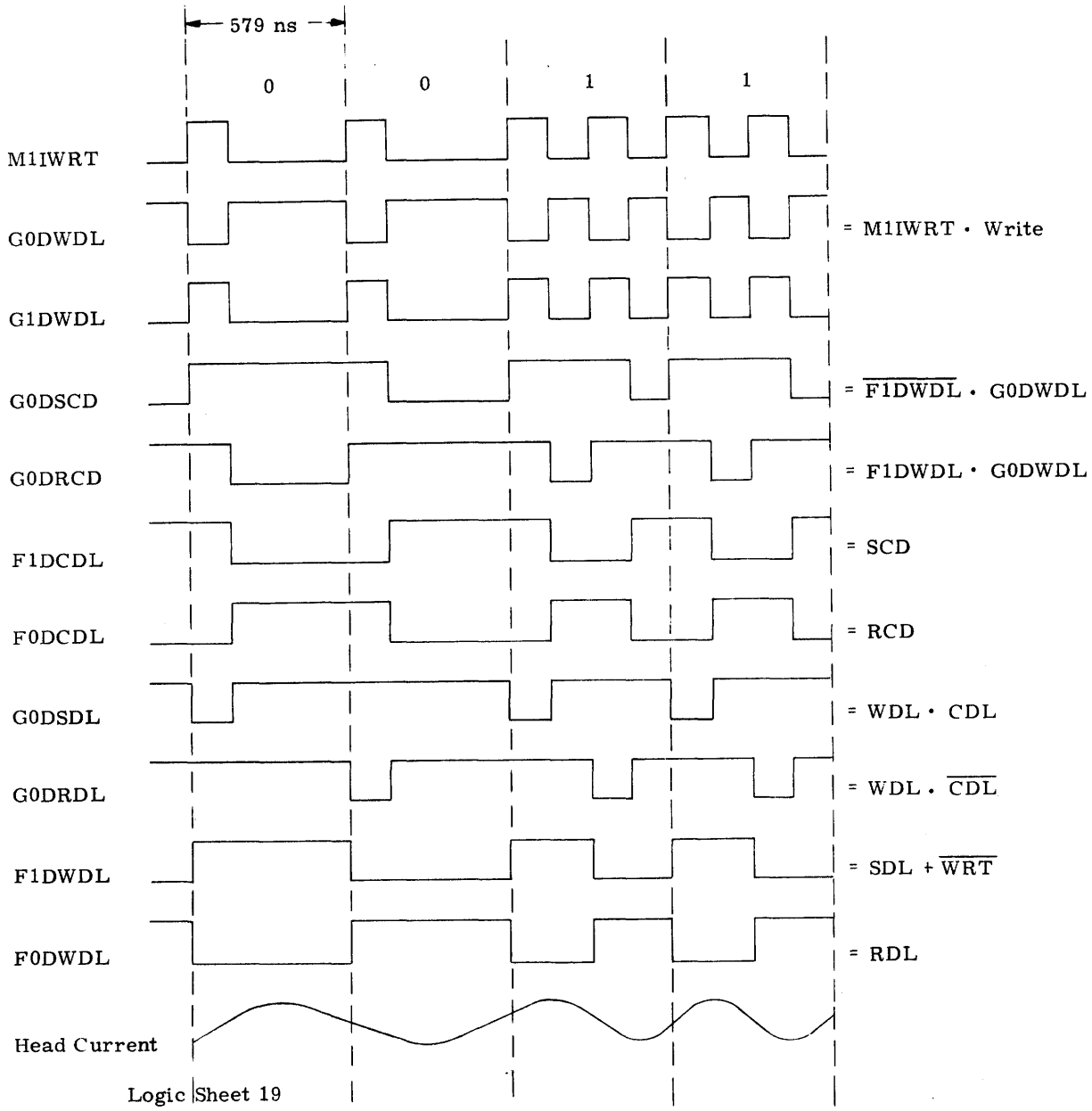


Fig. THEORY. 11 Write Data Conversion

If Q1 is on, enough voltage is placed across the 5.1V zener diode to cause it to go into reverse conduction. The zener diode holds the base of Q2 at about -19V DC, and the emitter of Q2 is held at a similar voltage. Since R35 has a constant voltage drop of about 5V, the current flowing across R35, through the emitter and collector of Q2 and through the selected write amplifier, is held at about 125 milliamperes over a wide range of load impedances. (5V/37.4Ω = 133.7 ma).

The write data from the two outputs of F1DWDL is applied to the input transformer of the selected write amplifier. The bases of write amplifier stages Q3 and Q4 are 180° out of phase, and they are turned on by alternate half-cycles of the dual frequency data, allowing the write current to pass through alternate halves of the one head in row-00 which has its center tap returned to ground by a write return driver. The resultant alternating flux, passing through the gap in the head core and the drum surface, records the data on the drum.

WRITE PROTECTION

In addition to the logical write protection provided by the Dual Bulk Controller, which is selectable through switches on the controller panel, drum units provide protection of selected drum storage areas by bypassing to ground any write current directed to protected heads. If any current bypass occurs, a write protect alarm signal is transferred to the controller on the inter-unit cable.

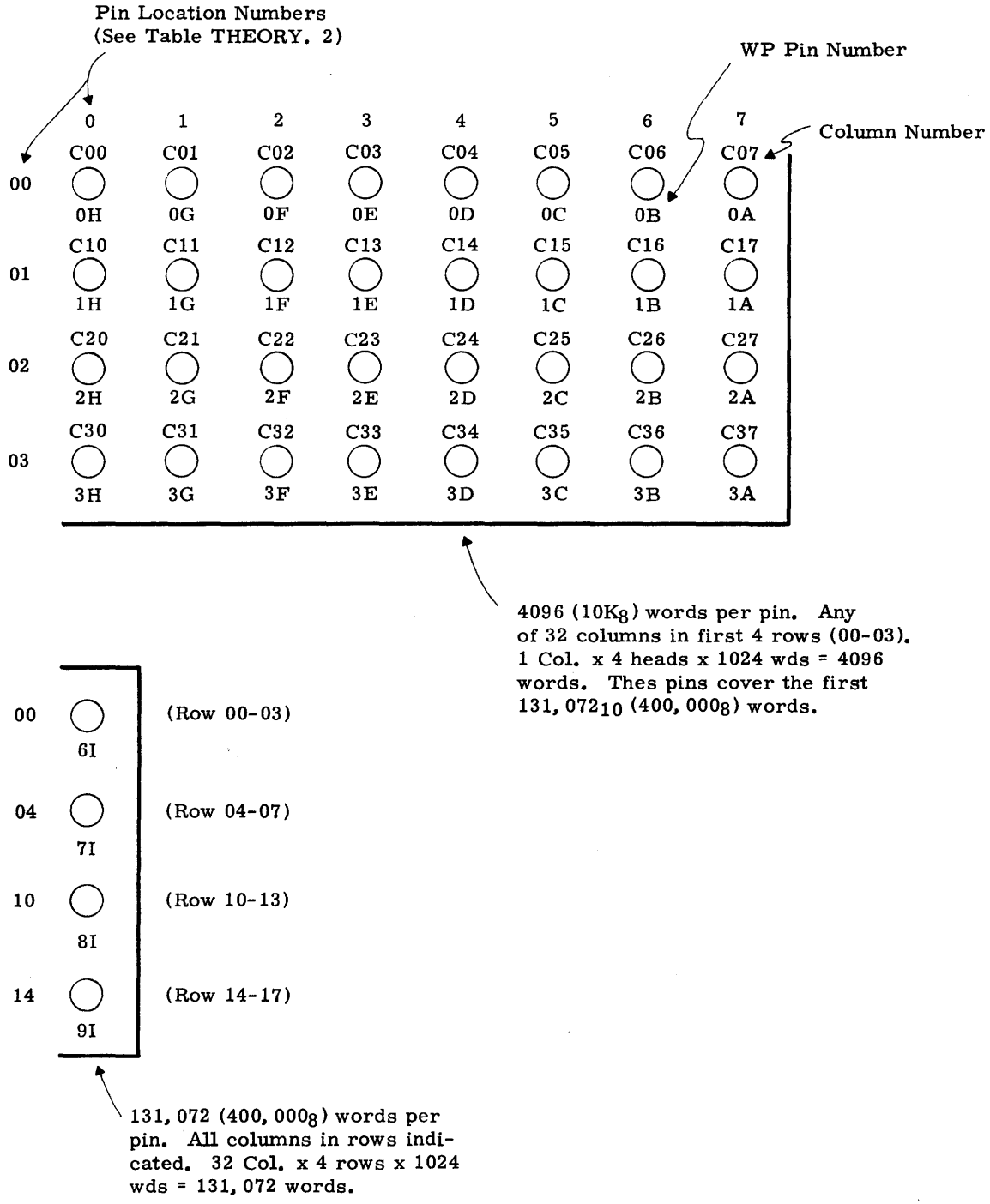


Fig. THEORY. 12 Write Protect Select Panel

4096 ($10K_8$) - Word Pins

Pin No. corresponds to column intersections protected in row-00 through row-03.

Pin No.	Drum Addresses (AR08-AR00)	CW1 or DBC Begin. Add. (12 - 00)	Pin No.	Drum Addresses (AR08-AR00)	CW1 or DBC Begin. Add. (12 - 00)
000	000 - 003	00000	020	100 - 103	02000
001	004 - 007	00100	021	104 - 107	02100
002	010 - 013	00200	022	110 - 113	02200
003	014 - 017	00300	023	114 - 117	02300
004	020 - 023	00400	024	120 - 123	02400
005	024 - 027	00500	025	124 - 127	02500
006	030 - 033	00600	026	130 - 133	02600
007	034 - 037	00700	027	134 - 137	02700
010	040 - 043	01000	030	140 - 143	03000
011	044 - 047	01100	031	144 - 147	03100
012	050 - 053	01200	032	150 - 153	03200
013	054 - 057	01300	033	154 - 157	03300
014	060 - 063	01400	034	160 - 163	03400
015	064 - 067	01500	035	164 - 167	03500
016	070 - 073	01600	036	170 - 173	03600
017	074 - 077	01700	037	174 - 177	03700

131,072 ($400K_8$) - Word Pins

Pin No. corresponds to first row protected in a group of four, at all 32 column intersections.

Pin No.	Drum Addresses (AR08-AR00)	CW1 or DBC Begin. Add. (12 - 00)
00	000 - 177	00000
04	200 - 377	04000
10	400 - 577	10000
14	600 - 777	14000

Table THEORY. 2 Write Protect Selection

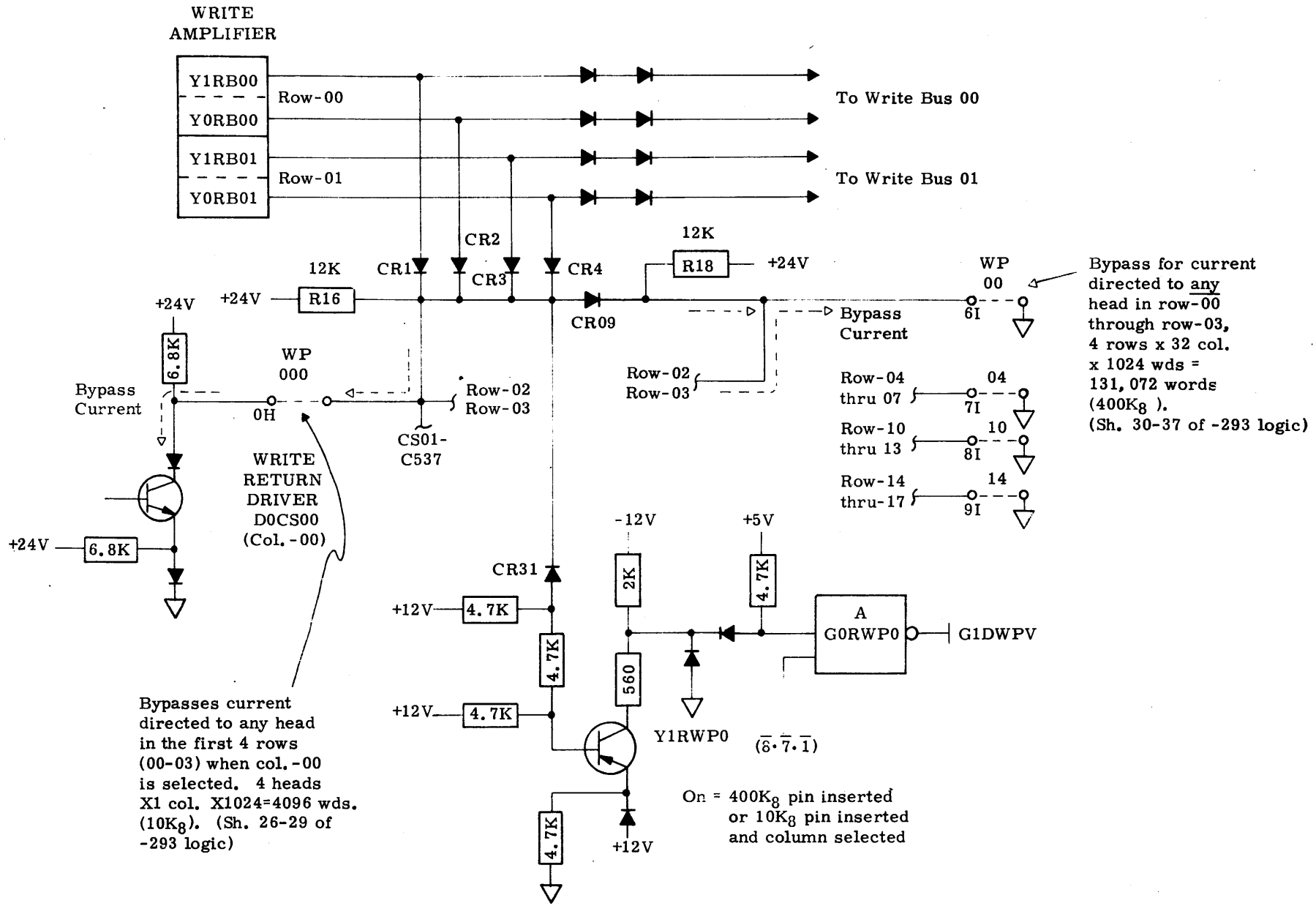


Fig. THEORY. 13 Write Protect Circuit

The areas to be protected are selected at the write protect select panel (WP1), which is above the logic panel, and accessable by opening the front door of the Drum Memory Unit. A picture of the panel appears on sheet 38 of the -293 logic, and on Fig. THEORY. 12. The scheme whereby any or all 4096-word blocks in the first 131,072 words of drum storage may be selected, or whereby all of the drum may be selected for protection in 131,072-word blocks, is shown on Fig. THEORY. 12.

Each of the 32 pin jacks, above and to the left of the heavy line at the top of the panel, may be used to select four consecutively numbered heads in the first four rows, at any of the 32 column intersections. These pins protect the first 131,072 words of bulk storage.

For drum units with a storage capacity greater than 131,072 words, the four pins at the bottom-left of the panel may be used to protect the first and each subsequent 131,072 word block.

Table THEORY. 2 relates the select panel pin positions with the starting head address of each area protected by the pins. Sheet 39 of the -293 logic and a chart attached to the inside of the front door on each drum unit provide similar information, however, these two sources state the beginning and ending address of each protected area in terms of words. As an example, the 524,288th word is at ending address $1\ 777\ 777_8$. Using CW1 terminology, this is group 37_8, head 17_8, sector 17_8, the 77_8th word.

The circuits involved in bypassing write current through pin no. 000 and no. 00 appear in simplified form on Fig. THEORY. 13. The operation of these circuits is typical of all others.

Assume that a pin is inserted in position 000. Should write current be applied to write buss 00 or 01, and should column-00 be selected through some hardware, software, or operator error, the current will find a lower impedance path to ground through the write return driver, than through the heads. For example, if the current is applied to row-00, it will pass through CR1 and CR2, through write protect pin-000, and through the write return driver bypassing head 000_8.

If a pin is inserted in position 00, any write current appearing on row-00 through row-03 will be bypassed to ground through the pin, regardless of the column selected.

The write protect detection circuit, Y1RWP0, is enabled whenever the cathode of CR31 is held near ground potential. This occurs when pin 00 is inserted or when pin 000 is inserted and column-00 is selected. The write protect alarm signal, however, is generated only when the unit is in the write mode, and when one of the two rows associated with the detection circuit is selected. When the unit write protect alarm gate, G0DUWP (sh. -19 of 293 logic), is enabled, the alarm signal is transferred to the controller over the inter-unit cable.

$$G0DUWP = WRT \cdot WPV$$

$$G0DWPV = WP0 + WP1 + WP2 + WP3 + WP4 + WP5 + WP6 + WP7$$

$$G0RWP0 = Y1RWP0 \cdot \overline{AR08} \cdot \overline{AR07} \cdot \overline{AR01}$$

READ OPERATION

The controller initiates a read operation by turning on Control Select and Tag Line 1. One head and one read bus will be enabled as described under "Read Column Selection" and "Read Row Selection", according to the content of the drum unit address register.

The analog signal from the selected head (1.73 Hz = zero; 3.46 Hz = 1) passes through two DC blocking capacitors to the input terminals of a dual integrated circuit amplifier on the multiplexer board, XMAA1 (Fig. THEORY. 10), which provides a voltage gain of about 50:1. This raises the signal level from several millivolts at the head to about one-half volt peak-to-peak. The signal passes through a dual emitter follower to the field effect transistor (FET) switches on the data demodulator board, XDDA1 (sh. 14 of the DRE logic, 70C180119).

Data Demodulator

If more than eight rows are implemented (storage capacity greater than 262,144 words), the input to the data demodulator will originate from one of the two multiplexer boards. The channel 0 enable signal, Y1DCH0, or the channel 1 enable signal, Y1DCH1, enables the appropriate FET switch, which then passes the signal through a single-ended emitter follower, and on to the demodulation circuits.

The demodulator detects each crossing from one-half cycle of the analog signal to the next, or each "zero crossing", and produces a short negative going pulse at each zero crossing time. The demodulation waveforms appear on sheet 8 of the -119 logic. After filtering to remove possible noise and harmonic components, the signal is amplified sufficiently to greatly overdrive two voltage limiting diodes. The clipped positive and negative half-cycles are coupled across a transformer to a dual comparator. The outputs of the comparators are square waves with steep leading and trailing edges, which mark the time position of each zero crossing.

The comparator outputs are differentiated at the input to the pulse width shaper, T0DDTA. T0DDTA utilizes three common-emitter amplifiers, the first of which is biased so that it responds only to the positive spikes in the differentiated pulse waveform. These are inverted three times, as the pulses are amplified and shaped, and the resultant output is one negative going pulse, whose leading edge corresponds to each zero crossing of the analog signal read from the drum. Zeros, then, are represented by one pulse per bit period (579 ns), and ones are represented by two pulses per bit period.

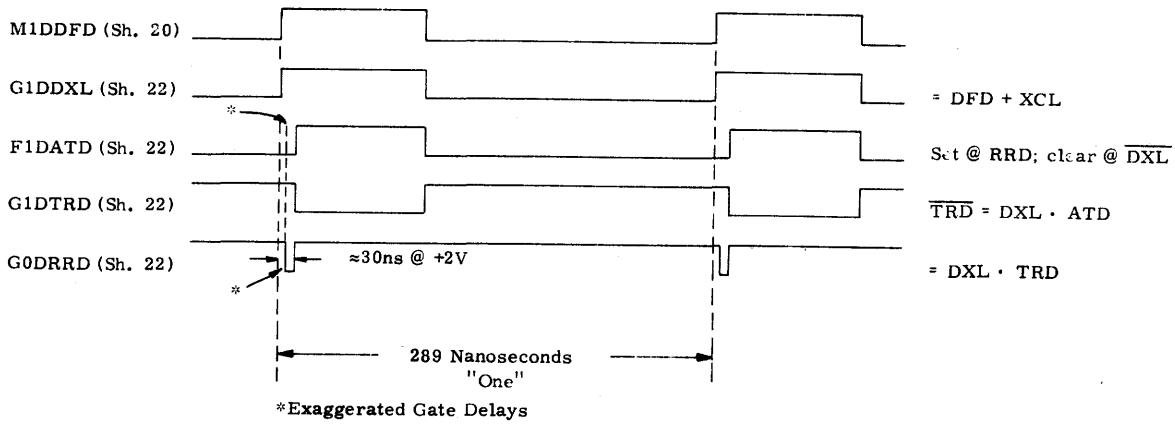


Fig. THEORY. 14 Read Data Shaping

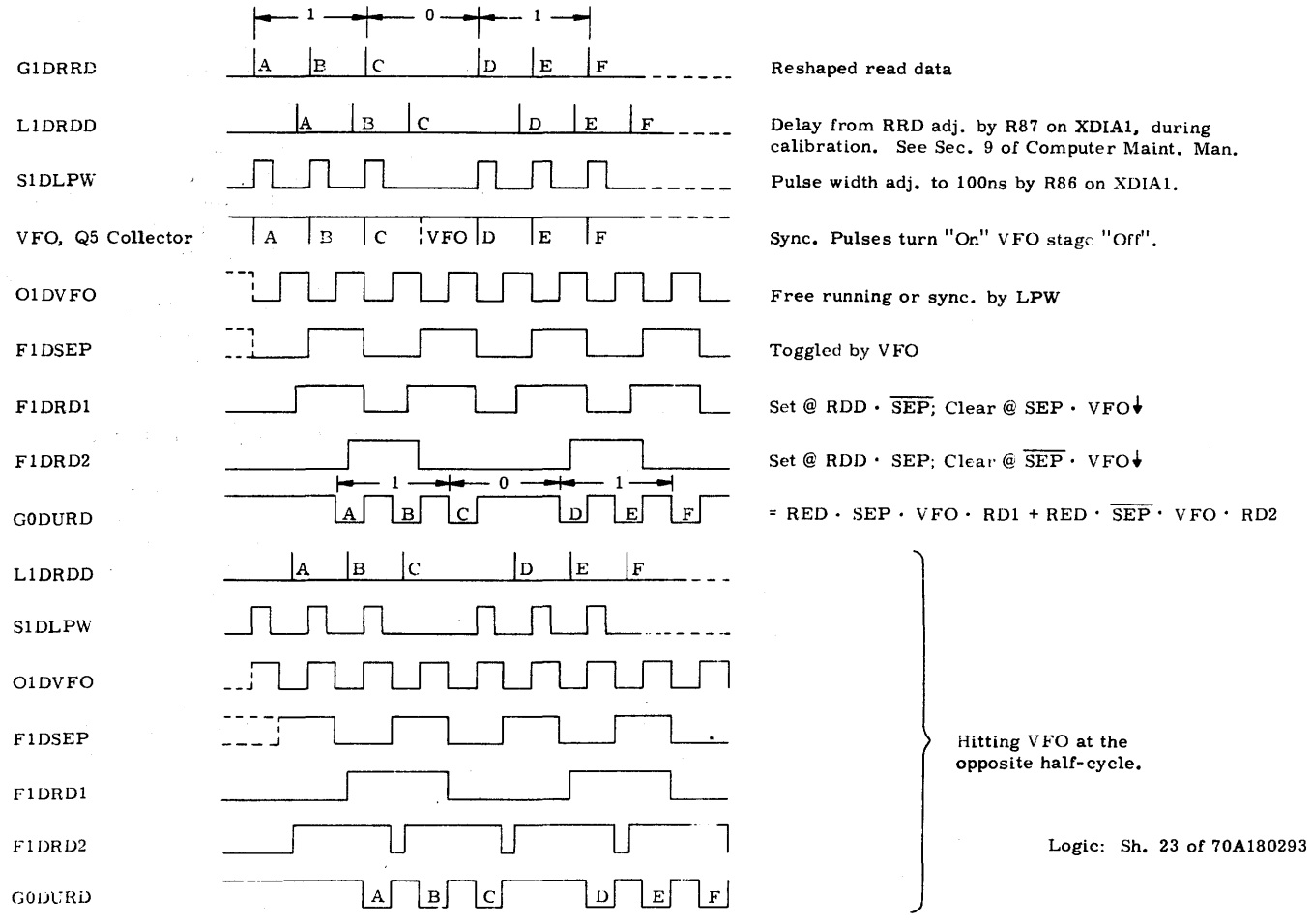


Fig. THEORY. 15 Read Data Recovery

The T0DDTA output is fed to the logic panel for further processing, by two-wire line driver, X1TDTA, which is located on the sector demodulator board, XTTA1 (sh. 9 of the -293 logic).

Data Recovery

Because of mechanical tolerances in the drum, and variations in gas pressure, temperature, rotating speed, etc., the data pulses from the DRE may vary slightly in width, time position, and amplitude. The data recovery logic converts the data read from the drum to solid, stable, digitized serial data for transfer to the controller.

The logic which appears on sheet 23 of the -293 drawing reshapes the DRE data pulses to very narrow pulses for use as data strobes by the recovery logic. The reshaping logic depends on the propagation delay through some of the gates to limit the width of the output pulses. A timing diagram is provided by Fig. THEORY. 14.

The 289 nanosecond period indicated on the timing diagram represents the first half of a bit cell containing a "one". 'If the cell contained a "zero", the second set of pulses would not be present. In the example shown, the OPR/CAL switch on the XDIA1 board is in the OPR position, and the DXL pulses correspond to the data from drum pulses, M1DDFD.

When the data pulse appears at DXL, it also appears in a few nanoseconds at the output of G0DRRD, because at that time, G1DTRD is true. In a few additional nanoseconds, F1DATD is set by the RRD pulse. With ATD set, TRD goes not true, and the RRD pulse is terminated.

The OPR/CAL switch is placed in the CAL position when performing maintenance procedures. Gates G0D6FB, 6XT, and 692 operate as a crystal oscillator. G0D6FB provides the regenerative feedback signal through R24. The 6.92 MHz square wave is divided by two in F1DXCL, which provides a 3.46 MHz square wave input to DXL. With the switch in the CAL position, the drum data is stopped and the XCL pulses pass, providing standardized "one" pulses at the output of RRD for calibration purposes. Refer to Section 9 of the Computer Maintenance Manual for the calibration procedure.

As has been explained, data in the controller is in the form of pulses which are one quarter of a bit period in duration. Two of these pulses per bit period represent a one, and one per period represents a zero. The logic on sheet 23 of the -293 drawing generates the controller data in accordance with the reshaped read data pulses from G1DRRD. This logic is arranged to produce data pulses of the correct width and time position, but the RRD pulses have no direct effect on the data pulse outputs, other than to dictate the number of them occuring in a bit period. The timing diagram provided by Fig. THEORY. 15 illustrates the operation of the logic on sheet 23.

S1DLPW, L1DRDD, and O1DVFO are discrete component circuits, which appear in schematic form on sheet 50 of the -293 logic. The RRD data pulses trigger single shot multivibrators in the LPW and RDD circuits. These are conventional single shots, which produce positive pulse outputs of adjustable duration, at the collector of the second transistor. The LPW pulse is adjusted for a duration of 100 nanoseconds at the +2V level. The RDD delay controls the time position of a short spike at the trailing edge of the single shot period.

The spike at the trailing edge of the RDD period is obtained from the AND of the inverted single shot pulse and another pulse delayed by an RC circuit, as shown by Fig. THEORY. 16.

O1DVFO is a free-running multivibrator, which operates at 3.46 MHz. The Q6 and Q7 stages form an astable multivibrator, which is synchronized by the leading edges of the LPW pulses, if read data is present. The operating point of the sync circuit is adjusted by R85 while the crystal controlled calibration signal is applied to G1DRRD through G1DDXL. When data from the drum is present, the oscillator is resynchronized by each data pulse. When a "zero" is read from the drum, no LPW pulse will occur at the middle of the bit period, but the oscillator will continue to run, with the sync being corrected, if necessary, at the beginning of the next bit period.

The remainder of the recovery logic is involved in gating appropriate VFO pulses onto the data line to the controller. The example used on Fig. THEORY. 15 is that of the recovery of the first three bits of a word. Note that each positive half-cycle of the VFO signal is one-quarter of a bit period in duration, and that the unit read data at the output of G0DURD is in the controller data format. G0DURD enables line driver X1IRED, which transmits the data to the controller on the inter-unit cable.

POWER SUPPLIES AND CONTROLS

The interface between the Dual Bulk Controller and the bulk memory units incorporates a power sequencing scheme whereby each unit is brought on-line in sequence, in order to spread the start up surge-current demand over a longer period of time. An additional feature of the interface is the plug interchangeability of Drum Memory Units and Disc Memory Units.

The power sequencing interface, is depicted on sheet 67.2 of the controller logic, 70C179882, and also appears in simplified form on Fig. THEORY. 17. While the interface circuits in the drum and disc units are not identical, they are similar and compatible. In both examples, unit 0 is designated as a drum unit. While a drum unit or disc unit can function in any of the unit positions, it is normal practice to assign drum units to the lower unit positions, when both drum and disc memories are employed.

When the controller receives a "Bus Level Monitor (BLM) good" signal from the CSU, relay driver

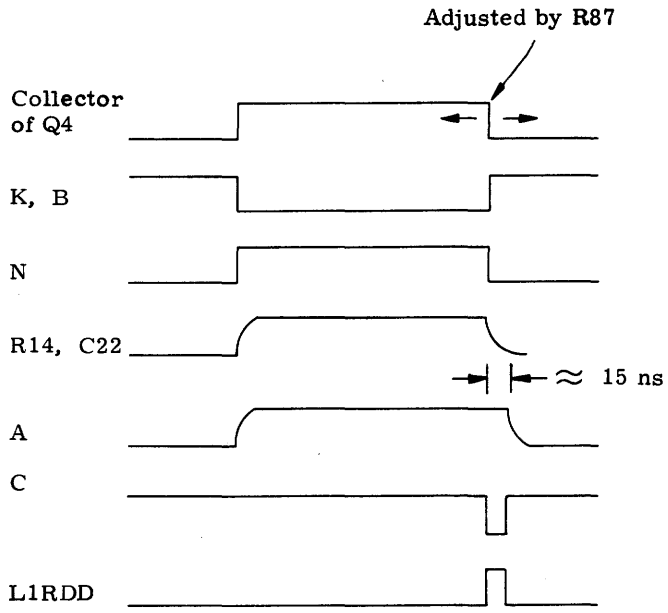


Fig. THEORY. 16 L1DRDD Timing

P1LBLM, in the controller, will be enabled, closing the circuit between pins R, T, and U on CP2-J1 at unit 0. +24V is thereby applied to the power on relay, K6, in the drum unit sequencer. As is described under the "Drum Unit Sequencer" heading, when the drum unit start-up sequence is complete, relay K4 is energized. As is shown on Fig. THEORY. 17, the K4 contacts in unit 0 apply the turn-on signal to the next unit. When that unit has come on, the next is turned on, and the process continues until the last unit implemented is on.

A unit ready signal, B1LUxR, is applied to the controller, as each of units 1, 2, and 3, comes on. The controller design presumes that at least unit 0 will be present and ready, so no unit 0 ready line is provided.

The unit ready line voltages will be +20V DC if the next higher unit is a disc unit or +24V DC if the next higher unit is a drum unit. If, however, the next higher unit is not connected or the control voltage is not on, the ready signal becomes +12V DC from the controller, which is supplied through a diode. +12V is applied to B1LU3R when unit 3 is ready, because it is the last unit possible. The diodes in series with the +12V lines from the controller are back biased when the control voltage from the next higher unit is on.

Drum Unit Sequencer

A normal power-on sequence is described in the following numbered steps. Sheet no. references are to the 70A180293 logic. A simplified schematic of the principal components involved is provided by Fig. THEORY. 18.

1. Each bulk memory unit is plugged into a standard industrial type socket, supplying single phase, 115VAC, 60 Hz power on three pins (high, neutral, and safety ground). This power is applied to the AC entrance panel (AC1) (sh. 48). If the main circuit breaker, CB1, is closed, the drum unit blowers will run, and the +24V DC control voltage power supply will be on (sh. 41; T2, CR11, CR12, CB6).

2. When either the controller closes the "BLM good" line, or the next lower unit comes on, the +24V control voltage energizes K6.

3. Initially K3 and K2 are de-energized, and the B1/S1 time delay switch contacts in the K2 ground leg are closed. K2 is energized by +24V, which passes through K6 contacts, the Reset BLM Alarm switch, and the normally closed K3 contacts. A pair of K2 contacts closes, applying +24V through CR2 to energize K3. The same contacts hold K2, through CR1. Another K2 contact set (sh. 47) applies 115 VAC to the B1/S1 time delay relay and to the bulk power supply (sh. 41). Note that the diode reference designators (CR_) are sometimes duplicated because they are mounted on separate but identical terminal boards.

4. +24V energizes the drum motor temperature relay K5 through the drum motor thermal switch contacts. Initially, CR2 and the B1/S1 contacts provide the ground return. Once energized, a set of K5 contacts holds the relay.

5. In a few milliseconds, the 5V and 12V logic power supply voltages settle within their tolerances, and the voltage monitor detector output from Y0VVMD goes to about +5V.

6. After about 6 seconds, the B1/S1 time delay contacts in the K2 ground leg open and the B1/S1 contacts in the ground leg of K1 close. Relay deiver #2 now provides the ground return for K2.

7. The B1/S1 contacts in the ground leg of K1 have now closed. S2 should be in the On-Line position, energizing K1. A set of K1 contacts applies power to the drum motor and the elapsed time meter (sh. 47). Within 5 minutes the drum motor will be up to speed and the head actuation mechanism releases the heads to allow them to "fly" on a film of helium gas formed at the rotating drum surface.

8. The heads loaded/unloaded switch in the drum goes to the loaded position. Since S2 is in the On-Line position, F1DONL (sh. 17) is set. N1DONL and the voltage monitor enable relay driver #1, which provides a ground return for K4, energizing K4, which then enables the power-on sequence in the next unit (Fig. THEORY. 17).

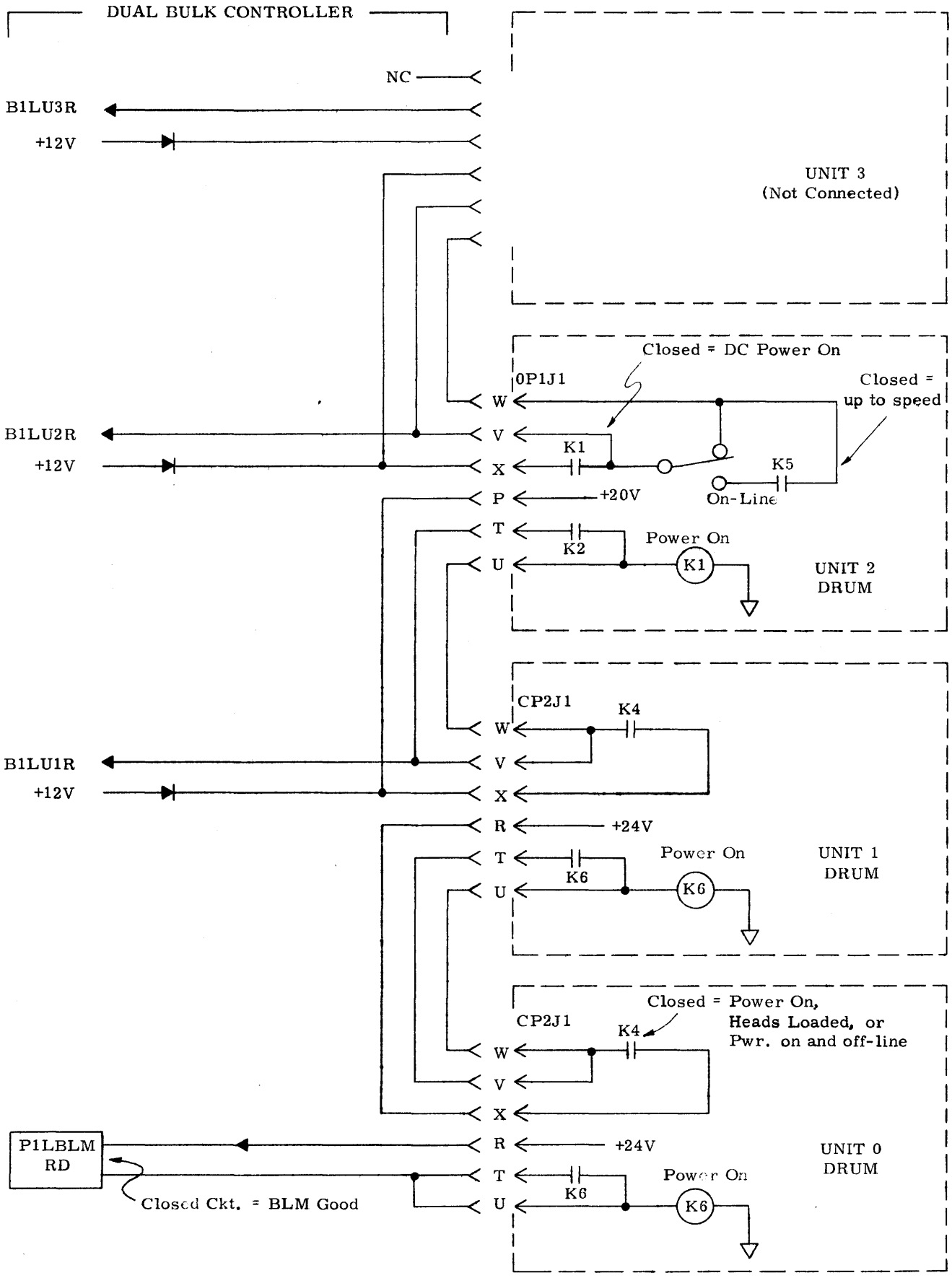


Fig. THEORY. 17 Unit Power Sequence Controls

The drum mechanism incorporates a speed monitor which retracts the heads to the unloaded position if the rotating speed drops below a safe level. The head switch in the drum will then clear F1DONL and the Unit On Line signal to the controller will go off.

Should the drum motor temperature exceed the safe limit, a thermal switch on the motor windings or another inside the motor housing opens, dropping K5. A set of K5 contacts drops K1, which disconnects AC power from the motor. Another set of K5 contacts closes, illuminating the Motor Temp lamps. Pushing the Reset Motor Temp switch will re-energize K5 and restart the motor if it has cooled enough to close the thermal switch.

Should one or more of the voltages monitored by Y0VVMD go out of tolerance, the VMD output will go to zero volts, disabling relay driver #2, and dropping K2. The K2 contacts in the +24V leg of K4 open, dropping K4. K3 remains energized, as +24V is applied through the Reset BLM Alarm switch and a K3 contact set. This same circuit lights the BLM Alarm Lamp. When K2 drops, AC power is removed from the bulk power supply and the time delay relay.

An attempt at restarting the power supplies may be made by pushing the Reset BLM Alarm switch. This will drop K3 and extinguish the alarm light. When the pushbutton is released, K2 will re-energize, re-energize K3, and reapply power to the bulk supply. If the DC supplies come within tolerance by the time the B1/S1 contacts open, relay driver #2 will hold K2. During this time, K1 remains energized, as its own contacts close the ground circuit.

The On-Line/Off-Line switch may be used to take the unit off-line without dropping K4 and thereby disrupting the operation of the next unit. When placed in the Off-Line position, S2 clears F1DONL, and the Unit On Line signal to the controller goes off. S2 provides a ground return for K4, and enables control of the drum motor with S3, Off-Line Motor Control.

The bulk power supply consists of a step-down transformer, T1, five full-wave rectifiers, and five circuit breakers (sh. 41 of -293 logic). The five unregulated DC voltage outputs are used by the voltage regulators and voltage monitors.

+5V Regulator

The +5V regulator (sh. 42 of -293 logic) filters and regulates the +10V DC bulk supply output to produce +5VDC ± 1% for use by the drum unit logic. Q5, Q5, Q6, and Q7 are series pass transistors, whose series impedance is adjusted by a control signal from the differential voltage amplifier, Q8 and Q9. Q19 is an emitter follower which couples the control signal Q5, Q6, and Q7.

The differential amplifier compares the output voltage with a stable +3.9V reference voltage which is developed at the base of Q8, across zener diode CR18. Q3 provides an additional control signal component which is developed by comparing the bulk +10V DC with a reference voltage. The Q3 output also aids in cancelling line frequency ripple in the +5V output, as the ripple appearing at the collector of Q3 is inverted with respect to that on the bulk supply.

As an example of the operation of the regulator, assume that due to an increase in line voltage or a decrease in load current, the +5V output voltage tends to increase. The increased voltage, increases the voltages across the voltage divider on the base of Q9, which includes a variable resistor for adjustment of the +5V output. The Q9 collector current increases, while the Q8 collector current decreases. The resultant increased voltage drop across the Q3 circuit drives the base of Q19 in a negative direction, increasing the series impedance of the pass transistors, and holding the +5V output within tolerance.

Q3 aids the effect described in this example. As the bulk +10 DC increases, the collector to emitter impedance of Q3 increases, increasing the voltage drop across the transistor, which drives the base of Q9 in a negative direction.

Stages Q1 and Q2 form a differential amplifier which serves as an undervoltage sensor. The amplifier compares the +5V output voltage with a reference voltage at wiper of variable resistor, R50, which serves as the trip point adjustment. When adjusted properly and operating normally, Q1 is conducting and Q2 is off. The sensor output to the voltage monitor, Y0VP05VM, then, rests at about +5V (the collector load resistor, R41, appears on sheet 43, and is returned to -12V).

Should the output voltage drop below +4.5V, Q2 turns on, and Q1 goes off. This allows Y0VP05VM to swing to about 0V, and trips the voltage monitor, which drops sequencer relay K2, shutting down the logic power supplies.

An overvoltage crowbar is also provided. Should the output voltage go above about +5.7V, zener diode CR22 goes into reverse conduction, pulling the gate of SCR1 sufficiently positive to fire the SCR. This places a virtual short across the +10V bulk supply, which trips CB5. The undervoltage sensor senses this event as an undervoltage, and AC power is removed from T1 in the bulk supply.

+12V and -12V Regulators

Functionally, the 12V regulators (sh. 44) are similar to the +5V regulator. The principal differences are the reduced number of series pass transistors, and in the case of the -12V regulator, the use of PNP transistors where NPN's are used in the +5V regulator, and an NPN where the +5V regulator uses a PNP.

Undervoltage sensors are provided for both regulators and these operate in the same manner as the +5V sensor. The operating points and component values of these sensors are adjusted to accomodate the differing voltages with which they must operate, but the output voltages at Y0VP12VM and Y0VM12VM are similar to those

of the +5V regulator (+5V = go; 0V = no go). The +12V sensor is adjusted to trip at +10.8V. The -12V sensor is adjusted to trip at -10.8V.

Overvoltage crowbars are provided for both 12V regulators. They function in the same manner as the +5V crowbar. The +12V crowbar fires at about +17V and trips CB2 on the bulk supply. The -12V crowbar fires at about -17V and trips CB3 on the bulk supply. The -12V crowbar fires when Q9 begins to conduct. During normal operation, the emitter and base are held at about -13V by zener diode CR8. When the -12V regulated output goes sufficiently negative, enough current is drawn across R26 to turn the PNP transistor on, and fire the SCR.

+24V and -24V Regulators

The +24V and -24V regulator outputs are the principal read and write circuit power supplies. The schematic is on sheet 45 of the -293 logic. The voltage regulators, undervoltage sensors, and crowbars operate as similar circuits previously described, with one significant difference.

The +24V and -24V undervoltage sensor outputs are not applied to the voltage monitor, but are used on logic sheet 17 to produce the File Unsafe Alert signal. In the case of the +24V sensor, should the output voltage drop below +21.6V, Q19 turns on and Q18 goes off. The Q19 collector swings positive, turning Q20 on. This lowers the collector of Q20 to about 0V, turning Q21 off. In the case of the -24V sensor, should the output voltage go below -21.6V, Q13 goes on, Q14 goes off, Q15 goes on, and Q16 off.

Note on logic sheet 17, that gate G0DAVA is normally disabled, as Y1VP24RD and Y1VM24RD are normally at 0V. If either undervoltage sensor trips, G0DAVA is enabled and sets the file unsafe flip-flop F1DFUA.

The +24V crowbar fires at about +27V and trips CB1 on the bulk supply. The -24V crowbar fires at about -27V and trips CB4 on the bulk supply.

Voltage Monitor

During the power on sequence, the voltage monitor (sh. 43 of -293 logic) monitors the outputs of the 5V and 12V undervoltage sensors, and allows the sequenced power to remain on if the voltages are in tolerance by the time the B1/S1 time delay relay opens the ground return line on sequencer relay K2 (Fig. THEORY. 18 and logic sh. 46).

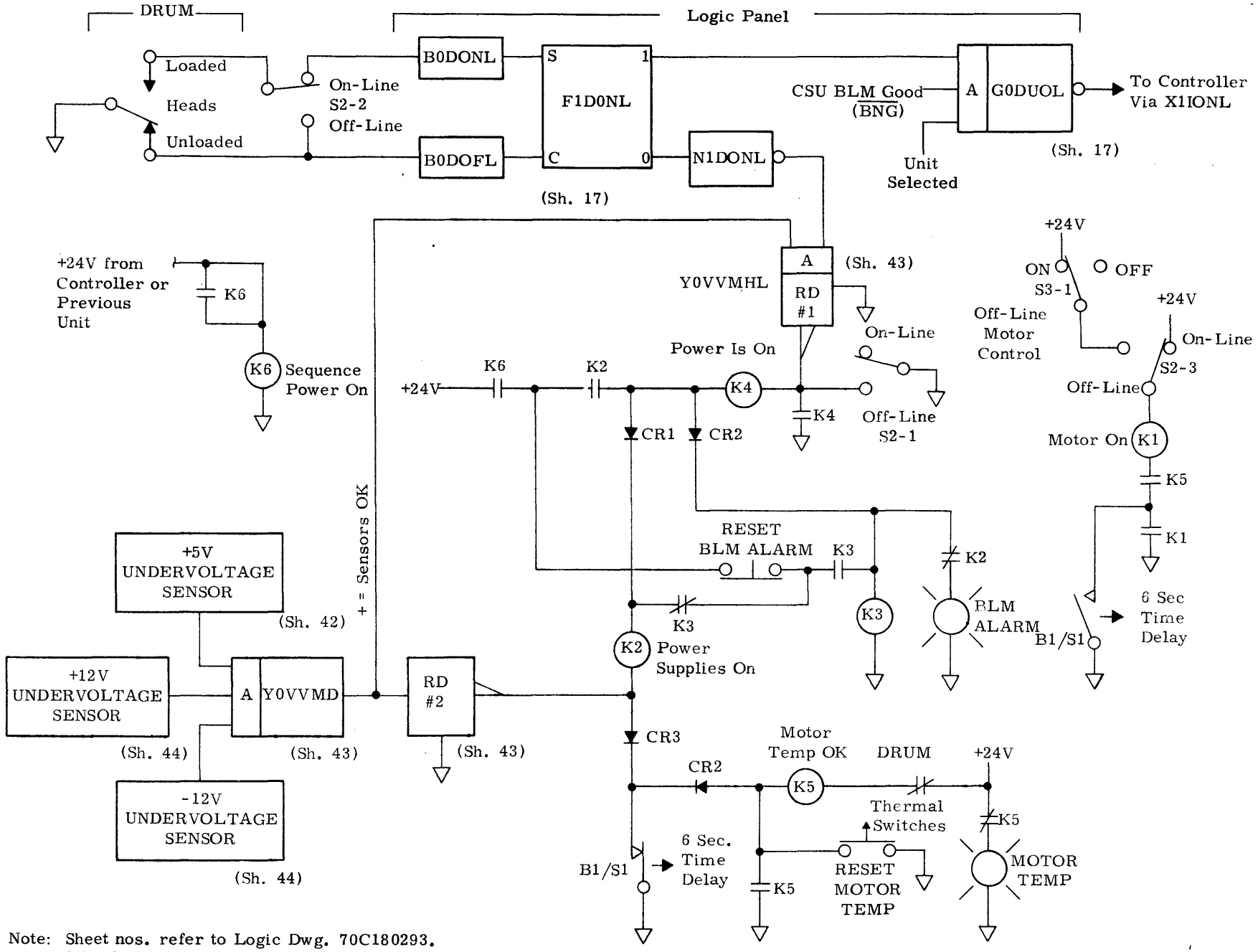
If, after the power on sequence is complete, any of the three logic voltages goes out of tolerance, the voltage monitor drops K2, which removes AC power from the bulk power supply.

CR1 through CR6 and Q10 comprise an AND gate, which requires that all six diodes be reverse biased by a positive voltage, in order to hold Q10 on. CR1, CR2, and CR3 are connected to the +5V, +12V, and -12V undervoltage sensor outputs, respectively and all three are normally at +5V. The remaining three diodes are connected to the +5V supply. Should any of the three undervoltage sensors be tripped, the sensor line will open, which allows the corresponding diode to be forward biased by -12V through a resistor. CR7 will also be forward biased and the base of Q10 will be held slightly negative.

With Q10 off, Q11 is turned on, which holds Q12 off. With Q12 off, relay K1 is de-energized and a K1 contact sets holds the Y0VVMD output at ground. Y0VVMD disables each of the relay drivers in the lower half of the schematic. Relay driver #2 is the one which holds sequencer relay K2 energized during normal operation. Y0VVMD also resets the BLM not good flip-flop, F1DBNG, resets the file unsafe alert flip-flop, F1DFUA, resets the seek complete flip-flop, F1DSCD, and disables all of the write amplifier current sources.

If the three logic supplies come within tolerance during the power-on sequence, Q12 will turn on and Q16 off, making Y0VVMD go positive, enabling relay driver #2, which provides the ground return for sequencer relay K2, holding sequenced power on.

Relay K1 in the voltage monitor circuit holds the Y0VVMD signal down for a few milliseconds during the power on sequence to allow the logic power supplies to stabilize before picking up the ground return for K2. It is possible for the regulated voltages to temporarily overshoot the "in tolerance" band when power is coming up, and to return momentarily to the undervoltage levels before stabilizing.



Note: Sheet nos. refer to Logic Dwg. 70C180293.
All relays and switches are on Sh. 46.

Fig. THEORY. 18 Power Sequencing and Control

READER COMMENTS

The General Electric Company solicits your comments on publications covering Process Computer equipment. Please explain any "no" responses in the COMMENTS section. Your comments and suggestions become the property of the General Electric Company.

- Name of Manual: _____
- What is your computer application: _____

  _____
- How is this publication used:

  Familiarization ☐ Reference ☐

  Training ☐ Maintenance ☐

  Other (Explain) _____

	YES	NO
• Does this publication meet your requirements	☐	☐
• Is the material:		
1) Presented in clear text	☐	☐
2) Conveniently organized	☐	☐
3) Adequately detailed	☐	☐
4) Adequately illustrated	☐	☐
5) Presented at appropriate technical level	☐	☐

- Please provide specific text references (page number, line, etc.) with your comments.

NAME _____ DATE _____

TITLE _____

COMPANY NAME _____

AND ADDRESS _____

COMMENTS:

No postage necessary if mailed in the U. S. A.

Staple

Communications concerning Technical Publications should be directed to:

Manager, Technical Publications
Utility and Process Automation Systems Operation
2255 West Desert Cove Road
Phoenix, Arizona 85029

Fold Fold

FIRST CLASS
Permit No. 4091
Phoenix, Arizona

BUSINESS REPLY MAIL
NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY...

GENERAL ELECTRIC COMPANY
UTILITY and PROCESS AUTOMATION
SYSTEMS OPERATION
2255 West Desert Cove Road
Phoenix, Arizona 85029

Attention: Technical Publications

Fold Fold

Cut Along Line

Additional Comments:

4291/2 VIDEO DISPLAY SUBSYSTEM

TABLE OF CONTENTS

INTRODUCTION

The Video Display Subsystem provides a television type display of graphic characters, including letters, numerals, punctuation marks, and special symbols. The characters may be transferred to the display from a GE-PAC* computer and from optional keyboards. The display subsystem provides a very direct and highly flexible interface between process operators, engineers, and other personnel, and the computer system. Messages of almost unlimited variety and format can be displayed for interpretation and response by the operators. When the optional keyboards are provided, the operators can communicate with the computer system, directing various computer and process functions, as the computer monitors and controls the process.

The Video Display Subsystem consists of a Display Controller Unit, one or two Display Terminals, a communications channel, and an interface in the GE-PAC computer.

This publication provides a description of the overall operation of the display subsystem, a block diagram analysis of the subsystem, and references to the detailed theory of operation publications and logic drawings. Detailed circuit and logic analyses of the Display Controller, Display Terminals, and of GE-PAC modules covered elsewhere are not provided in this publication, except as required to provide an understanding of the interrelations of several of the functional modules.

This publication describes display subsystems utilizing Display Controller models DCU-765 and DCU-775 (PCPD models 4291/4292). Refer to publication number DTN-760 for a description of display subsystems implementing Display Controller DCU-760. The operation of the 92-character per line Display Controller, DCU-785/786, is similar to the operation of DCU-775/776, except that DCU-785/786 can serve one Display Terminal only.

EQUIPMENT CONFIGURATIONS

Fig. INT. 1 is a simplified block diagram of the overall Video Display Subsystem. Where the Display Controller and the computer are directly connected by a multiconductor cable, they may be separated by up to 50 feet of cable. When two data sets are used, one at the computer site and one at the Display Controller site, the distance separating the sites is virtually unlimited.

Video Display Equipment Options

The video display equipment can consist of as few as two units, the Display Controller and a display monitor; or it may consist of a Display Controller with two display terminals, each of which may incorporate a keyboard. The functional units are often referred to by three-letter identifiers. These are described in the following subparagraphs. The actual equipment employed in a GE-PAC computer system can be determined by reference to the System Hardware Block Diagram (or System Configuration Diagram) and by physically inspecting the equipment.

- DCU - Display Controller Unit. The basic electronics cabinet for the display subsystem. One is required for each subsystem. The DCU includes the following functional modules. However, the physical components which make up the functional modules are situated throughout the DCU, and are not individual sub-units as in DCU-760.

  BC - Basic Controller.

  DLC - Data Line Controller. The communications line interface, which transfers data to and from the computer.

  DTS - Direct Timing Source. Used for timing of a directly connected synchronous communications channel (DCU-775 only). The DTS is not normally used in DCU-775's connected to a GE-PAC computer, as 4800 baud clock signals are available from the clock generator in the computer's I/O Buffer.

  PPA - Page Print Adapter option. Not supplied with GE-PAC systems to date, and not covered in this publication.

  TMU - Terminal Memory Unit. Stores all data displayed on one display terminal. TMU-A will be present and TMU-B may be present if the MEU option is implemented.

  MEU - Memory Expansion Unit. Provides a second delay line memory and necessary logic required for TMU-B. This option may be installed when the DCU is delivered or may be added in the field.

- Display Terminal - one for each TMU implemented in the DCU. The Display Terminals include:

  DMU - Display Monitor Unit. The TV type display. May be a 14 inch (diagonal) DMU-761 or a 23 inch (diagonal) DMU-765.

  EKB - Electronic Keyboard. The optional operator's keyboard. Two basic types are available (Fig. INT. 1): EKB-761 has a keyboard lay-out similar to an office typewriter. EKB-764 and 765 are operator oriented keyboards with the keys arranged in alpha-numeric order,

*Registered Trademark of General Electric Company

special Action Keys which initiate predetermined message transmissions are available, and transmissions of a single display line or part of a single line may be initiated by one key depression. The operator oriented keyboards can be used with DCU-775 only.

KAU - Keyboard Auxiliary Unit. Allows an EKB to be used with the 23 inch monitor, DMU-765, by providing a keyboard power supply and interface. The power supply and interface are built into the 14 inch DMU-761.

- LRU - Line Repeater Unit. Permits the distance separating the DCU and Display Terminal to exceed 1000 cable feet. One LRU extends the maximum to 3000 cable feet. A second LRU in the line will allow a 5000 feet maximum. The LRU includes:

  KVA - Keyboard Video Amplifier. Provides compensations to signals transferred between the DCU and Display Terminal.

- PLJ - Party Line Junction option. Allows two to eight Display Terminals to be connected on a party line to a TMU. Each terminal will display the same information. A keyboard at any of the terminals may have access to the TMU if no other keyboard on the line is in use.

Each TMU can serve one Display Terminal. The number of rows displayed on the monitor can be selected by means of jumper wires in the DCU. Normally, the maximum 22 rows are displayed. Displays of 16, 8, or 4 rows may also be selected. Memory partitioning, as is available with DCU-760, is not available with the 765/775 display controllers.

A polling option, which allows more than one DCU to be connected to the same communications line, is not normally used with GE-PAC systems and is not covered in this publication.

Display controllers in a table top configuration are also available. They are functionally and electronically identical to DCU-765 or DCU-775. They are identified by the addition of one to the model number: DCU-765 becomes DCU-766 and DCU-775 becomes DCU-776. For 92-char./line units, DCU-785 becomes DCU-786 in the table top version.

Communications Channels

Characters are exchanged between the GE-PAC computer and the display equipment in a bit-serial form using a USASCII character coding. The interfaces to the communications line both at the computer and the DCU are in compliance with EIA Standard RS-232C.

The communications channel interface in the computer is a functional module of the 4202 I/O Buffer. When synchronous character transmissions at 4800 baud (baud = bits per second) are used, the Synchronous Communications Unit (SCU) is employed[1]. When asynchronous character transmissions at 1200 baud are used, the Asynchronous Communications Drive (ACD) is employed.

In 4020/40/50/60 computers, the SCU is model 4202A/B/C466/566. In 4010 computers, the SCU is model 4820A/BS06. These modules are functionally identical and employ identical printed wire boards, PSTB1 and PSRB1.

In 4020/40/50/60 computers, the ACD is model 4202A/B/C461/561. In 4010 computers, the ACD is model 4820A/BS07. These modules are functionally identical and employ identical printed wire boards, PATA1 and PARB1.

In 3010 computers, DCU's -775 and -785 communicate through the computer's 3010AH14 Synchronous Drive Interface (SDI). For all GE-PAC systems, DCU's -775 and -785 communicate through a synchronous channel at 4800 baud. DCU-765 communicates with 4000 series systems through the asynchronous channel at 1200 baud.

In the majority of applications, the computer and DCU are connected directly through a multiconductor cable 50 feet or less in length. Digital data sets may provide an interface to leased telephone lines or through a dialing network. All communications are two-way half-duplex. Output-only communications from the computer to DCU are not used with these display controllers. Typical data sets are:

- Synchronous, 2000 or 2400 baud; 201A3 or 201B1 (American Telephone and Telegraph Co.).
- Asynchronous, 1200 baud - TDM-210 (General Electric Co.); 202C or 202D (American Telephone and Telegraph Co.).

[1] If synchronous data sets are used, the baud rate may be limited to 2000 or 2400 baud.

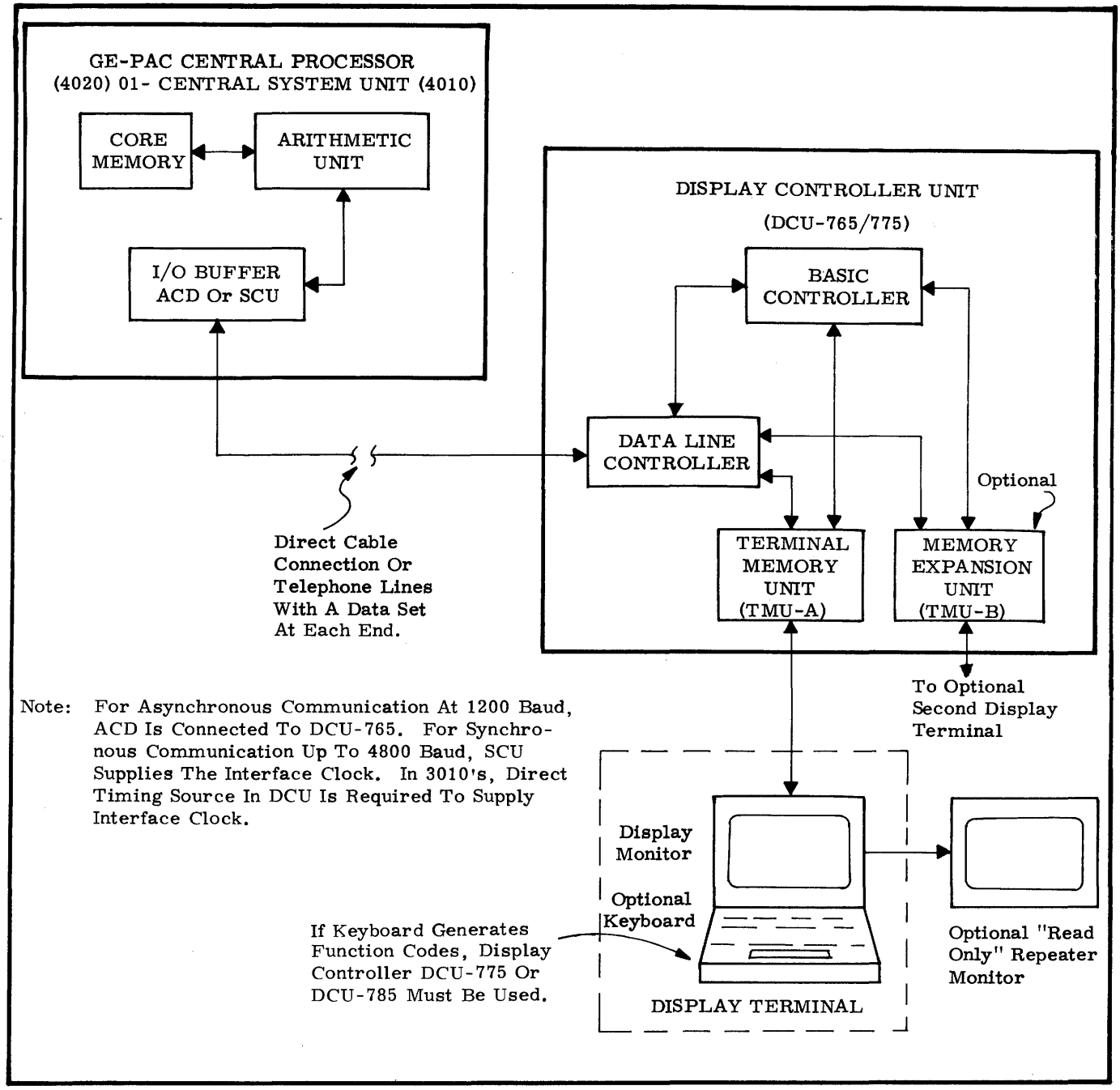


Fig. INT. 1 Video Display Subsystem

OPERATION

VIDEO DISPLAYS

The display monitors are an industrial version of a conventional television set, except that the receiver circuits are not included. Each display consists of 46 character positions on up to 22 rows, or a maximum of 1012 character positions, as shown on Fig. OP. 1. (DCU-785/786 generates 2024 character positions.)

A blinking entry marker is always at the top edge of one of the 1012 character positions. The entry marker indicates the position of the next character to be added to or changed on the display. The entry marker position may be controlled from the keyboard or by the computer.

When the operator initiates a transmission request, the entry marker indicates the first text character in the message to be transmitted to the computer. As the message transmission proceeds, the entry marker is incremented from left to right, row by row, until an end-of-text symbol is encountered, or until row 22, position 46 is reached. The text transmission is then completed and the entry marker is placed in position one of the next row[1].

The entry marker proceeds from left to right, row by row, as text messages are received from the computer, stopping in the position after the last text character received, unless the text is followed by one or more entry marker control characters[1].

The entry marker also moves in response to character entries from the keyboard, as described under the Keyboards heading.

If a computer or keyboard input updates the entry marker beyond the last character position on the display, the entry marker will appear at the first character position at the top of the display, and will continue to be updated as required from that point.

One or more ETX symbols, as depicted on Fig. OP. 1, may be placed in any of the 1012 character positions on the display by moving the entry marker to the desired position and entering the ETX symbol[2]. The ETX symbol is entered from the computer by the transmission of an EM (031_8) character. ETX entries from the keyboard are made by pushing the ETX key.

After the ETX symbol is entered, the entry marker is normally placed over the first text character to be sent in the next text transmission. A transmission request is initiated as described under the Keyboards heading. When the request is serviced, the text

[1] This differs from DCU-760, where the marker is updated after the transmission is complete.

[2] This differs from DCU-760 where the ETX symbol could be entered only at the end of a row, outside the text area, and only from a keyboard.

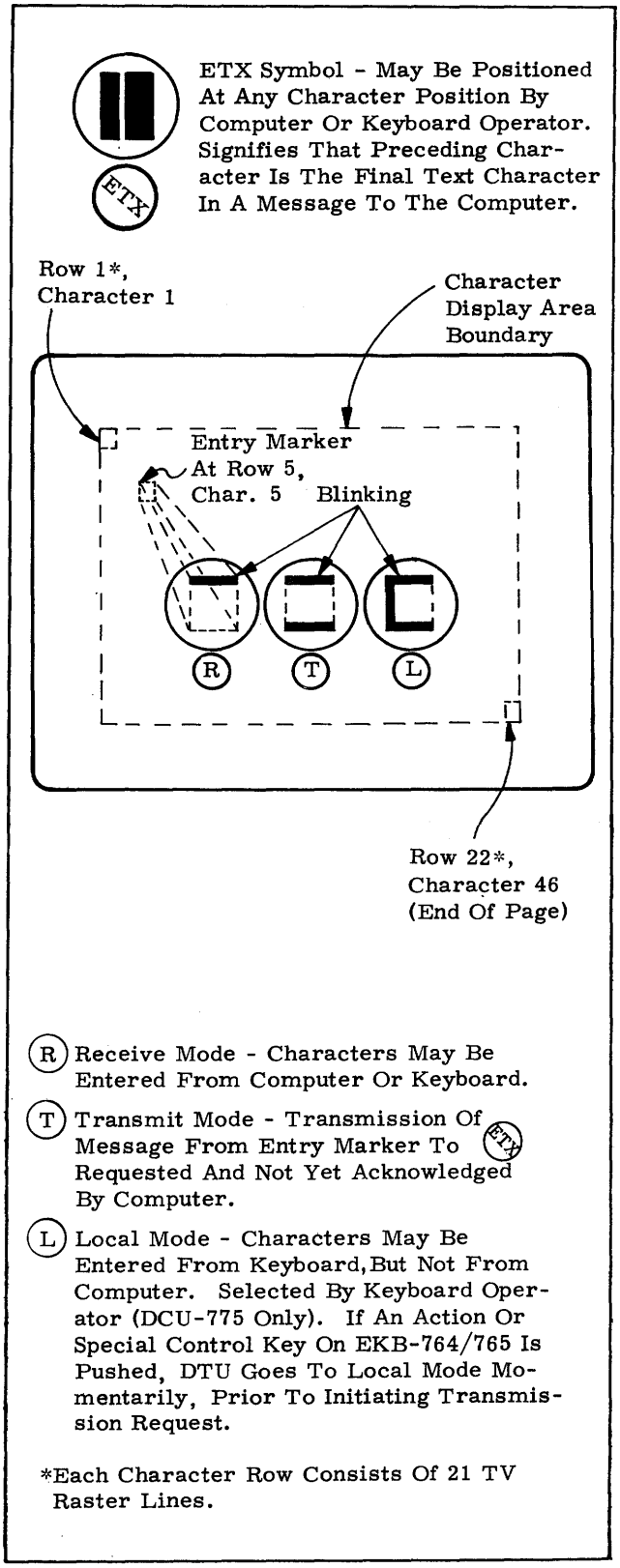


Fig. OP. 1 Video Display

portion of the messages begins at the entry marker position and continues through the transmission of ETX (003_8), when the first ETX symbol is encountered. Should the end of the page at row 22, position 46, be encountered, text transmission will end at that position, and ETX will be transmitted in the next character time.

The character position containing the entry marker also contains the display terminal mode indication, as indicated on Fig. OP. 1. The entry marker continues to blink, regardless of the mode, and the lines which indicate transmit and local modes do not blink.

When the keyboard operator enters function codes, they are stored in two function code registers (DCU-775 only) and do not appear on the display, as with DCU-760.

Table OP. 1 is a modified list of ASCII codes which are used in messages between the DCU and the computer, and which may be entered from the keyboard, if the key which generates the character is present. The graphic characters are those listed for octal codes 040 (space) through 137. If the "blink" code (136_8) is stored in a TMU, each of the characters following it, up to the next space or the end of the row, will blink. The "blink" code may be included in the text of messages from the computer, or may be entered by pushing the BLK key on keyboards containing such a key[1].

When new characters are added to a display, either by the computer or a keyboard, each character appears almost immediately. The rate at which characters added by the computer can appear is limited by the baud rate of the communications channel. The following is a list of the approximate times required to fill a 1012 character display at each of the baud rates in use with these display controllers (baud = bits per second):

Baud Rate	Char. per Sec.	Approx. Time
1200	120	8.5 sec.
2000	250	4.1 sec.
2400	300	3.3 sec.
4800	600	1.7 sec.

KEYBOARDS

Two principal keyboard versions are in use. EKB-761 has an appearance and lay-out similar to an office style typewriter. The other is the operator oriented version, which has the graphic symbol entry keys arranged in alphabetic and numerical order. The operator oriented keyboards, EKB-764 and EKB-765 have several Action Keys, arranged in three rows at the top of the board, which generate predetermined message transmissions to the computer with a single key depression.

[1]Two-color displays, and the selection of the display color with the "blink" code are not available with these controllers.

The operation of most of the keys is self-evident. When the Display Terminal is in the Local mode or the Receive mode, depressing any of the data entry keys causes the corresponding graphic character to be entered at the entry marker position on the display and to be stored in the associated TMU. The following paragraphs describe the functions of the remaining keys. Some of the keys described may not be found on individual keyboards, but where they are found, the functions are as described. On some keyboards, some of the markings indicated in the following paragraphs are found on the upper portion of several keys. In such cases, the SHIFT key must be pushed at the same time the key with two markings is pushed, to initiate the function.

Control Keys

BS — Backspaces the entry marker one space. If the entry marker is in character position one of a row, it will not move.

FS — Forward spaces the entry marker one space. If the entry marker is in position 46, it goes to position one on the next row. If in position 46 of row 22, it goes to row one, position one.

LF — Line Feed. Moves the entry marker down one row at the same character position. If the entry marker is on the last row of a display, it goes to the top row.

ETX — Enters the ETX symbol (Fig. OP. 1) at the current entry marker position.

RLF — Reverse Line Feed. Moves the entry marker up one row at the same character position. If the entry marker is on the first row of a display, the entry marker does not move.

LR — Line Return. Moves the entry marker to character position one on the same row.

FF — Form Feed. Erases the text area of the display and the associated TMU (or TMU segment). Moves the entry marker to row one, character position one.

TAB — Moves the entry marker from its initial position to the character position following the next vertical line symbol.

NEW LINE — Moves the entry marker to character position one on the next row. If the entry marker is on the last row of a display, it returns to row one, character position one.

Command Keys

LOC — Selects the local mode of the Display Terminal. Messages may be composed on the display and if the computer addresses the

Octal Code		Character	Octal Code	Character	Octal Code	Character	Octal Code		Character
000	(1)	NUL	050	(	102	B	134		———
001	(1)	SOH	051	)	103	C	135		|
002	(1)	STX	052	*	104	D	136		blink
003	(2)	ETX	053	+	105	E	137		⌐
004	(1)	EOT	054	,	106	F	140	(7)	(clear)
006	(1)	ACK	055	-	107	G	141	(7)	(A)
010	(3) *	BS	056	.	110	H	142	(7)	(B)
011	(3) *	TAB	057	/	111	I	143	(7)	(C)
012	(3) *	LF	060	0	112	J	144	(7)	(D)
014	(3) *	FF	061	1	113	K	145	(7)	(E)
015	(3)	LR	062	2	114	L	146	(7)	(F)
021	(3) *	RLF	063	3	115	M	147	(7)	(G)
022	(3) *	FS	064	4	116	N	150		(REC)
024	(3) *	PR	065	5	117	O	151		(LOC)
025	(1)	NAK	066	6	120	P	152		(TX)
026	(4)	SYN	067	7	121	Q	161	(7)	(1)
031	(5) *	EM	070	8	122	R	162	(7)	(2)
032	(6)	(NL)	071	9	123	S	163	(7)	(3)
040		space	072	:	124	T	164	(7)	(4)
041		!	073	;	125	U	165	(7)	(5)
042		"	074	<	126	V	166	(7)	(6)
043		#	075	=	127	W	167	(7)	(7)
044		$	076	>	130	X	170	(7)	(8)
045		%	077	?	131	Y	177		(TAB)
046		&	100	@	132	Z	177	(1) *	DEL
047		'	101	A	133	[			

Characters enclosed by ( ) are generated by keyboards only.
* These characters are not transmitted by the DCU.

(1) Control character used in messages between DCU and computer.

(2) Message control character, or generated by keyboards.

(3) Entry marker control characters.

(4) Four SYN's (synchronization characters), generated by hardware, precede all messages between DCU-775 and the computer.

(5) End of Medium - used by computer to place ETX symbol in current entry marker position.

(6) New Line - generated by some keyboards.

(7) Function Codes - transposed to graphic codes and included in FC1 and FC2 position in next text message to computer.

Table OP. 1 Video Display Subsystem Character Set

terminal while in local mode, the message is ignored. The local mode symbol (Fig. OP. 1) appears with the entry marker. Local mode is available with DCU-775 only. [1]

REC — Selects the receive mode for the Display Terminal. Characters may be entered from the computer as well as the keyboard. This is the normal quiescent Display Terminal mode. This key is effective only with DCU-775 when the terminal has been placed in local mode. In all other cases, the terminal goes to receive mode when initialized by the power-on sequence and when the last message transmission has been acknowledged by the computer. The single entry marker appears when the terminal is in the receive mode (Fig. OP.1). [1]

TX — Initiates a transmission request for the Display Terminal. They keyboard is locked and the double entry marker appears. When the request is serviced, all characters from the entry marker position to the ETX symbol are included in the text portion of the message to the computer. When the transmission is complete, the double entry marker will rest in position one of the row following the last message row. With EKB-761, TX causes spaces to be transmitted in the function code positions. Function keys also generate transmit requests. The function code marked on the key will be in the first function code position and a space in the second function code position. With EKB-764/765, TX causes spaces to be transmitted in the function code positions. When an Action or Special Control key is pushed, a transmit request is set and two predetermined function codes are automatically generated.

PRT — Print. Used to transmit a message to a teletypewriter through a Page Print Adapter. Not used with GE-PAC systems as of the date of publication.

Entry Marker Control Group

The Entry Marker Control Group (EMC) option permits simplified positioning of the entry marker, requiring the operation of only one key at a time.

◁ — Moves the entry marker to the left one character position. Repeats if held down. If the entry marker is in character position one of a row it will not move.

▷ — Moves the entry marker to the right one character position. Repeats if held down. If it is on character position 46 of a row, it goes to position one on the next row. If it is on character position 46 of the last row of a display, it goes to row one, character position one.

▽ — Moves the entry marker down to the same position on the next row. Repeats if held down. If the entry marker is on the last row of a display, it goes to row one.

△ — Moves the entry marker up to the same position on the next row above. Repeats if held down. If the entry marker is on row one, it does not move.

PR — Page Return. Returns the entry marker to row one, character position one.

ETX — Place the ETX symbol (Fig. OP. 1) in the current entry marker position.

Function Key Group

The Function Key Group (FKG) option provides a means of entering function codes (FC1 and FC2) into the header of messages to the computer. The computer software may interpret these function codes and take appropriate action. Where used, the meanings of the function codes are specified for each specific system. The keys are labeled A through G, and where the larger group is used, 1 through 8 are also provided. These keys are not provided on the operator oriented keyboards, but the Action Keys on the operator oriented keyboards provide a similar function. Only DCU-775 implements function code transmission capability.

When provided, the Function Key Group is located in a single row above the other keys on the keyboard (EKB-761, only).

Each time one of the FKG keys is pushed, the corresponding character is placed in Function Code Register 1, an ETX symbol is written in the current entry marker position, a page return is executed, and a transmit request is made. Function Code Register 2 always contains a space (040_8) when EKB-761 is in use. The function codes do not appear on the display as with DCU-760. The operator must take care in entering FC1, because a transmission request is made almost immediately. A suggested procedure:

1. Compose any text to be transmitted, starting at position one, row one. After the text is composed, the entry marker is resting in the position where the ETX symbol will appear.

2. Push the appropriate FKG key. The code is entered in FC1, an ETX symbol is automatically placed in the current entry marker position, the entry marker returns to row one, position one, and a keyboard transmit request is set.

3. When the DLC services the transmit request, the FC1 and FC2 portions of the message header will contain the code entered and a space, respectively. The text portion of the message will be the contents of the display from row one, position one, to the ETX symbol.

[1] A keyboard transmission request initiated by Action or Special Control Keys, changes the terminal from local to transmit mode until the request is serviced.

Special Controls

Two special Control Keys are provided on the operator oriented keyboards which allow the automatic transmission of an entire display row or part of a display row. A third special control on these keyboards is a key-lock switch which allows the keyboard to be locked off, preventing unauthorized entries. In the following descriptions, the first marking for each control appears on EKB-764 and the second appears on EKB-765. The functions for these keys are the same.

SGL LINE or EDIT — Transmits a message including all of the characters on the row containing the entry marker. When pushed, the terminal goes to local mode, the function code registers are cleared, a "B" is stored in FC1, a "1" is stored in FC2, a line return is executed, a transmission request is initiated, and Local Mode is cleared. When the request is serviced and the end of the row is encountered, the ETX character is automatically generated and transmitted. The message then has FC1 and FC2 equal to B and 1 respectively, and the text is the contents of the row.

PTL LINE or POINT SELECT — Transmits a message including all of the characters from the current entry marker position to the end of the row. When pushed, the terminal goes to local mode, the function code registers are cleared, an 'A' is stored in FC1, a '1' is stored in FC2, local mode is cleared, and a transmission request is generated. When the request is serviced and the end of the row is encountered, the ETX character is automatically generated and transmitted. The message then has FC1 and FC2 equal to A and 1 respectively, and the text is the contents of the row from the entry marker position to the end of the row. This differs from SGL LINE/EDIT in that no line return is executed.

> NOTE
>
> If an ETX symbol is on the line to be transmitted via either of these keys, the message will end at that symbol, rather than the end of the line.

Action Keys

Fifteen, thirty, or forty-five Action Keys may be provided on the EKB-764/765 operator oriented keyboards. These keys automatically store predetermined function codes in the function code register in the display controller and transmit text from row one of the display. The computer software interprets the function codes and text and takes appropriate action. The Action Keys are supplied with special labels which indicate the action to be taken in terms meaningful to the process operator.

When pushed, the terminal goes to local mode, the function code registers in the controller are cleared, the function codes are stored in the register as indicated on Fig. OP. 2, a page return is executed (entry marker goes to row one, position one), a transmission request is initiated, and local mode is cleared. When the request is serviced, FC1 and FC2 are included in the header of the message, and the ETX character is transmitted when the ETX symbol is encountered on the row or after the last character on the row is transmitted.

Space Deletion Feature

Of the two Display Controllers, only DCU-775 deletes most or all redundant spaces from the text of messages transmitted to the computer, when it is determined that there are no characters but spaces from the last graphic character to the end of the row. This reduces the length of messages, transmission times, and the computer core memory space required to store the messages. If the transmission request is due to the use of the EDIT/SGL LINE, POINT SELECT/PTL LINE, or Action Keys on the operator oriented keyboards, the space deletion does not take place.

The display controllers normally insert a line return character and a line feed character in messages to the computer after the last character on a row has been transmitted. When the DCU-775 determines that there are only spaces following the last graphic character on the row, line return and line feed are inserted in place of most or all of the trailing spaces. Since the Data Line Controller stores up to five characters prior to transmission, from one to five spaces may be stored in the DLC, and if so, cannot be eliminated by the space deletion feature.

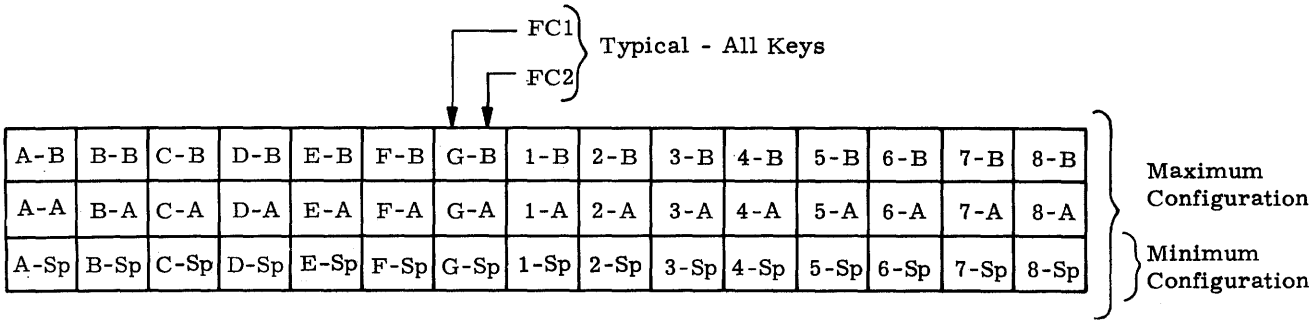
Keyboard Locking

Keyboards are locked (entries to the DCU prohibited) during the reception of a message addressed to the Display Terminal containing the keyboard, and from the time a transmission request is initiated at a keyboard, until the transmission is complete.

If a computer message addressed to a terminal is an information (NUL) message or an acknowledge (ACK) message, the keyboard is locked during the reception of the message. If the message is a negative acknowledge (NAK) message, the keyboard will remain locked until the retransmission is complete.

It should be noted that keyboards are locked from the transmission request until the transmission is complete. It is possible (but not probable) to have a double entry marker, indicating the transmit mode, after the transmission is complete. This is because an ACK or NUL message from the computer must be received to change back to the single entry marker.

If the double entry marker remains on for several seconds, it is possible that the computer is not replying to DCU messages, or that the computer's



Octal Keyboard Code		Octal Code Transferred to Computer
140	Space	040
141	A	101
142	B	102
143	C	103
144	D	104
145	E	105
146	F	106
147	G	107
161	1	061
162	2	062
163	3	063
164	4	064
165	5	065
166	6	066
167	7	067
170	8	070

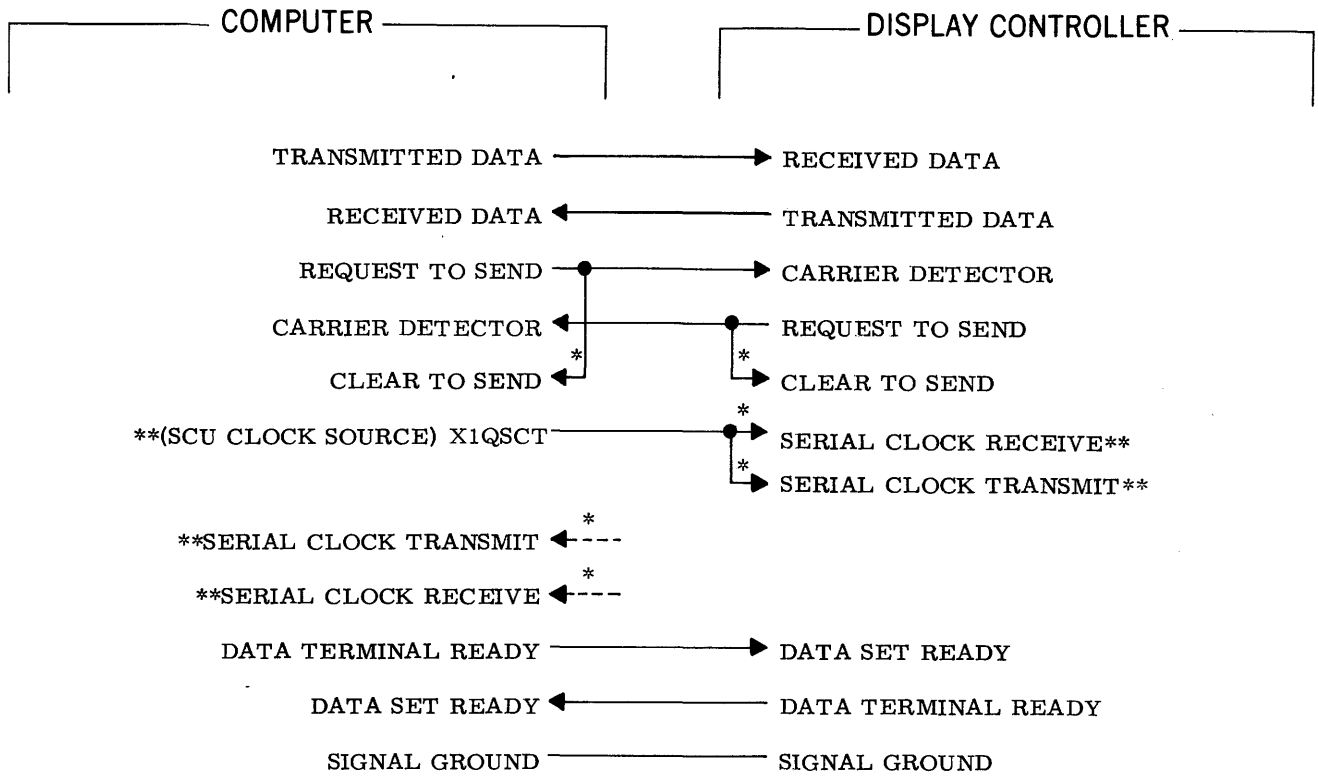
Fig. OP. 2 Action Key Function Codes

"Data Terminal Ready" signal is not on, which prevents transmission by the DCU. Data Terminal Ready from either of the two terminals cause Data Set Ready to be true at the opposite terminal. Of course, if the transmission is not completed, the keyboard will remain locked, and can be unlocked only by initializing the DCU (turn power off for about five seconds, then back on).

The preceding conditions hold for all keyboards with one exception. The operator oriented keyboards store the codes generated as the result of key depressions in a holding register. This allows EDIT/SGL LINE, POINT SELECT/PTL LINE, and Action Key codes to be stored while the keyboard is locked, to be serviced when the keyboard is unlocked. Should some key other than these automatic transmission keys be pushed, followed by the depression of an automatic transmission key, the code in the holding register will be the OR of the first key code and the automatic transmission key code, initiating an erroneous operation.

Party Line Monitor

The Party Line Monitor (PLM) option provides two combination key indicators which are required when a keyboard is used with a Display Terminal connected to a Party Line Junction (PLJ). This option is not used in GE-PAC computer systems.



Note: Connections shown are as with directly connected cable. When data sets are in use, the same sources and destinations hold. However, signals designated "*" originate in the data sets.

** These signals are used with Synchronous Communications Channels only. When directly connected, 4800 baud clocks originate in I/O Buffer.

Fig. OP. 3 DCU/Computer Interface

COMPUTER/DCU MESSAGES

Message exchanges between the computer and the display subsystem are in the form of 8-bit serial characters, several of which comprise a message. Either of these data terminals may transmit more than one message in a frame, where a frame is defined as the time during which the data terminal holds its carrier on. Fig. OP. 3 is a simplified schematic of the DCU to computer interface, which complies with EIA Standard RS-232C. The octal codes for all message characters are listed in Table OP. 1.

Asynchronous Communication

A start bit and at least one stop bit are appended to each character transmitted over the asynchronous communications channels. The start bit indicates the advent of a character and the stop bit represents the minimum amount of time during which the line must remain in the mark state before a new character may be transferred. Fig. OP. 4 illustrates the transmission of the letter "A" (101_8) over an asynchronous channel. The illustration shows an idealized waveform as would be observed at the transmitted data line output of either the Data Line Controller (DLC) in the display subsystem or the Asynchronous Communications Drive (ACD) in the computer.

In data communications terminology, a logical one is referred to as mark while a logical zero is referred to as a space. At the output of the DLC or the ACD, the line is in the mark state when at a negative voltage level and in the space state when at a positive voltage level. In the absence of any characters, the line remains in the mark state.

Jumper pins on the ACD boards are to be installed as follows: PATA1, F5 (black) and 1S, (green); PARB1, Ø (green) and S (black).

Synchronous Communication

The synchronous communications channels do not use start and stop bits but depend on the transmission of synchronization characters, SYN (026_8), before each message to synchronize the DLC in the display subsystem and the Synchronous Communications Unit (SCU) in the computer. The SCU hardware causes a minimum of four SYN characters precede each message

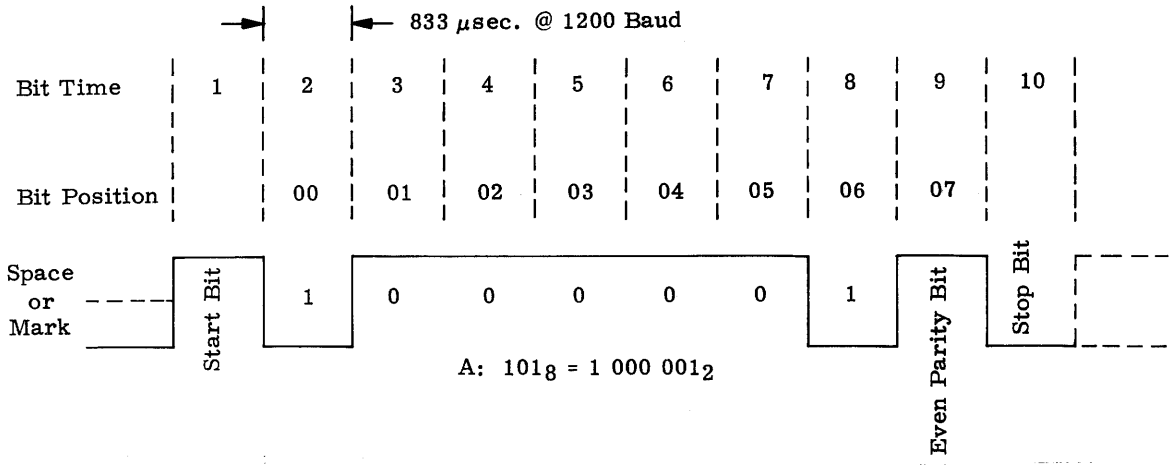


Fig. OP.4 Asynchronous Character Transmission

transmitted over the channel. The characters in a synchronous channel message must not be separated by any time interval. Once the DLC and the SCU are synchronized, they keep track of each bit time position until the message ends. There must be no time intervals separating characters in a synchronous message.

Fig. OP.5 illustrates the transmission of the letter "A" (101_8) as observed at the transmitted data line output of the DLC or the SCU. As with the asynchronous channels, a mark (one) is negative and a space (zero) is positive.

When data sets are used with synchronous communications channels, the serial clock pulses are normally supplied to the DLC and to the SCU by the data sets (a jumper pin must be installed in the black pin jack, FE, on both the SCU transmitter and receiver boards, PSTB1 and PSRB1).

When the SCU and DLC are directly connected by a cable, the serial clock pulses are supplied by the I/O Buffer at 4800 baud (a jumper pin must be installed in the black pin jack, FT, on PSRB1).

Proper operation of the lateral parity circuits in the SCU requires that the green pin jacks labeled "S" on both boards be jumpered.

Message Formats

The message formats for both communications channels are depicted on Fig. OP.6. Each of the characters listed is one 7-bit character plus a lateral parity bit. "Text" represents the graphic and entry marker control characters in the message. The characters enclosed in parentheses may or may not be in a message. The longitudinal parity character (LP) is always transmitted by the DCU, but is optional in computer messages (a hardware modification to the DCU is required to disable the LP character check. See Section 16 of the red Computer Maintenance manual or Section 19 of the blue Computer Maintenance manual). The EOT character should follow LP in the last message in a frame.

The significance of the message control characters is as follows:

SOH — Start of Header. The first character in any message. SOH indicates the header portion of the message is to follow.

ADR — The Display Terminal address. For the Display Terminal served by TMU-A, the address is 140_8. For the Display Terminal served by TMU-B, the address is 150_8.

NUL — Indicates that the message is an information message and is not a response to a message from the opposite subsystem.

ACK — Acknowledge. Error free reception of the last message from the opposite subsystem is acknowledged.

NAK — Negative Acknowledge. Indicates that a format or parity error was detected during reception of the last message from the opposite subsystem.

FC1
FC2 — The function codes. These will be "space" (040_8) unless function codes have been placed in the message through the use of the Function Key Group or Action Key options.

STX — Start of Text. Indicates that the next character is the first in the message. If there is no text, STX indicates the end of the header only.

ETX — End of Text. The preceding character was the last text character in the message and the following character will be LP. When received by the ACD or SCU in the computer, this character causes an end-of-record interrupt from the receiver at the end of the next character period. ETX also causes an end-of-record interrupt from the transmitter of the SCU, only, when the following character has been transmitted.

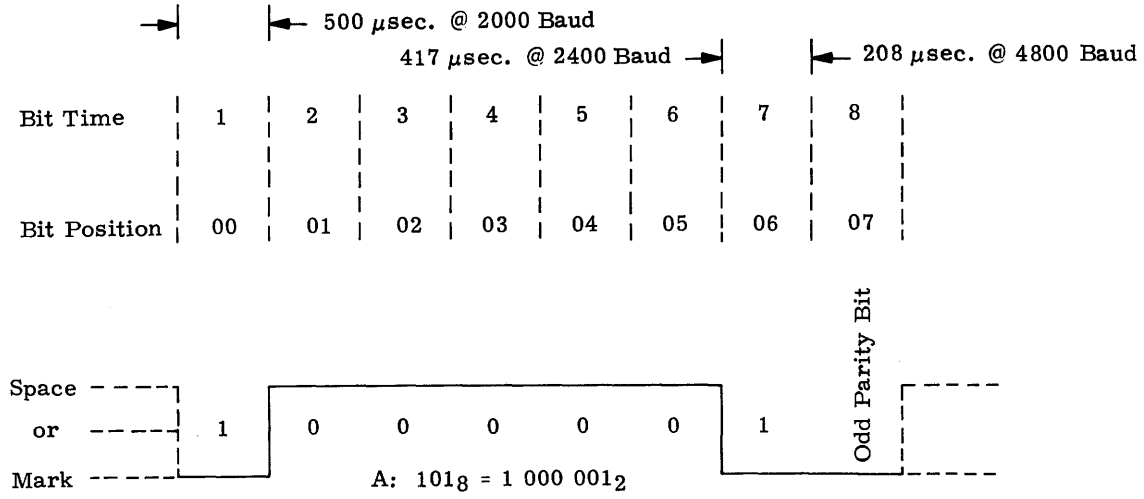


Fig. OP. 5 Synchronous Character Transmission

LP Longitudinal Parity. Refer to the Longitudinal Parity Check heading.

Message Sequences

Fig. OP. 6 indicates the basic sequence of messages between the DCU and the computer. Since communication is in the half-duplex mode (messages flow in one direction at a time), the number of "line-turn-arounds" is minimized if more than one message is allowed in each frame. When the DCU and computer are directly cable connected, very little time is lost in turning the line around. When half-duplex data sets are used with a two-wire communications line, turn-around time can become quite significant (turn-around time is minimized if four-wire communication lines are used).

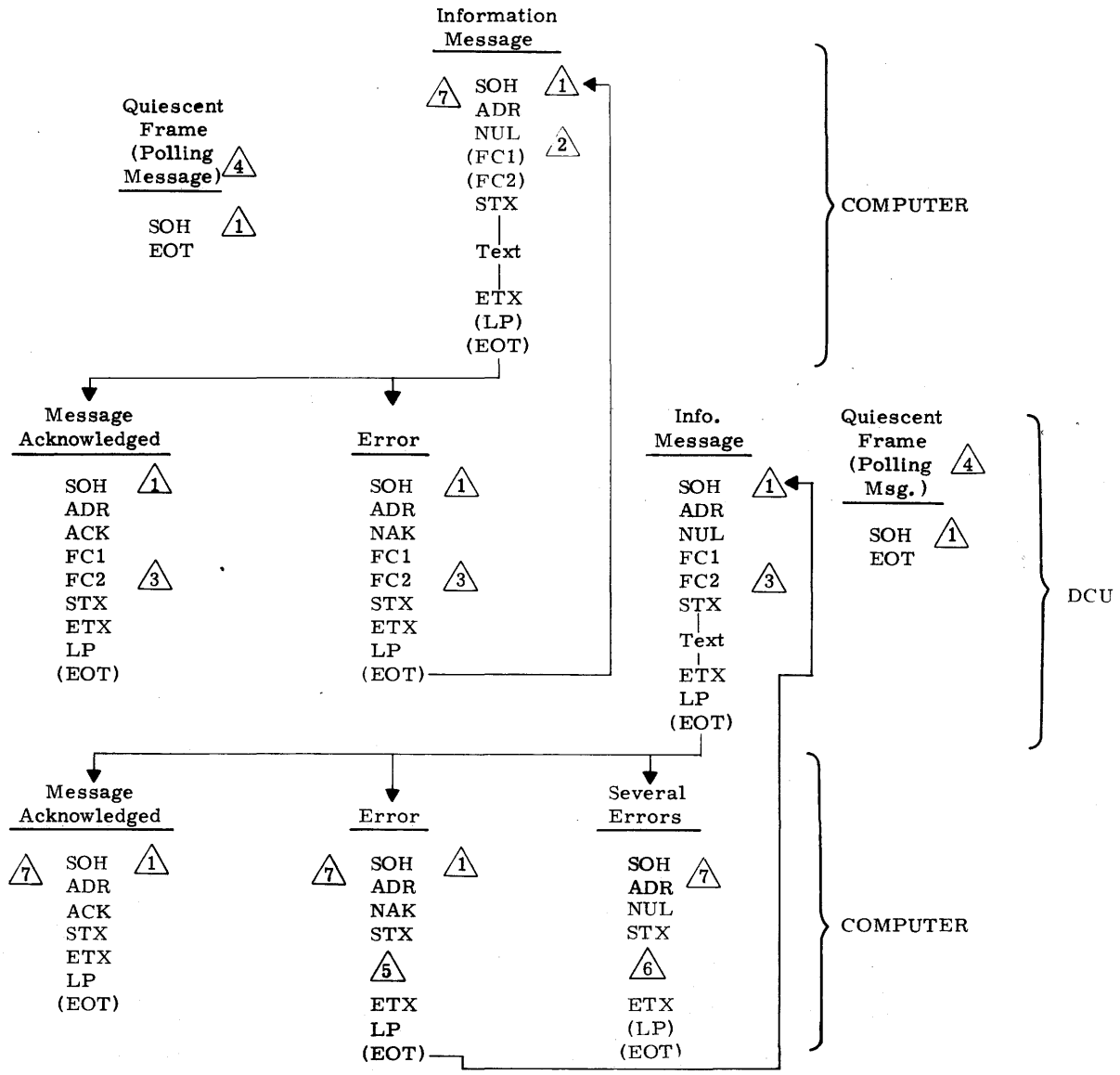
The DCU and computer message sequence flow charts, Fig. OP. 7 and Fig. OP. 8, indicate that more than one message can be transmitted in one frame. This requires that the DCU hardware be capable of properly sequencing transmitted messages and properly identifying received messages. The computer hardware and software cooperate in these tasks.

The primary factor in half-duplex message sequencing is carrier monitoring and control. The term "carrier" arises from the fact that when data sets are employed, the digital data modulates an audio frequency carrier, and the data sets employ a carrier detector which notifies the receiving data terminal when the distant data terminal has raised its carrier. Since the Request to Send signal from the transmitting end causes the transmitting data set to raise its carrier, the Request to Send signal is simply connected to the opposite Carrier Detector line, in the absence of data sets. This is shown on Fig. OP. 3. In half-duplex operation, a data terminal may not raise its carrier while the carrier from the opposite terminal is on. Further, when a data terminal drops its carrier, it is understood that it expects the opposite terminal to raise its carrier and send a reply.

Polling messages serve to confirm that the opposite data terminal is connected and operating. They also provide an indication that the terminal sending a polling message has no other message to send and is asking if the opposite terminal has any messages. If the DCU has no other messages to send, the hardware transmits a polling message in each transmission frame. DCU transmission frames occur on a regular basis: (1) Immediately after the computer drops its carrier if the last frame ended with EOT, and (2) four seconds after the computer drops its carrier if the last frame did not end with EOT. If the computer makes no replies to DCU polling messages, they will occur every four seconds. Polling message transmissions by the computer are optional.

If a negative acknowledge (NAK) message is received from the DCU, the computer program should retransmit the last message. If the computer detects an error in a DCU message and replies with a NAK message, the DCU will automatically retransmit the last message in the next transmission frame.

The standard PCPD software normally makes three attempts to retransmit messages for which NAK replies are received. If a third NAK reply is received, the program prints an error message on a system typer. If the computer must reply with NAK to several DCU retransmissions, the program may send an information message to the Display Terminal to print graphic characters on the display which inform the operator that the error condition exists.



1 When synchronous communication channels are in use, all SOH characters are preceded by at least four hardware generated SYN characters (026_8).

2 Function code characters may be included in computer messages if convenient, but are ignored by the DCU.

3 Function code characters will be "space" (SP, 040_8) from DCU-765, and from DCU-775, when messages without function codes are transmitted. FC1 and FC2 are "spaces" in virtually all ACK and NAK messages.

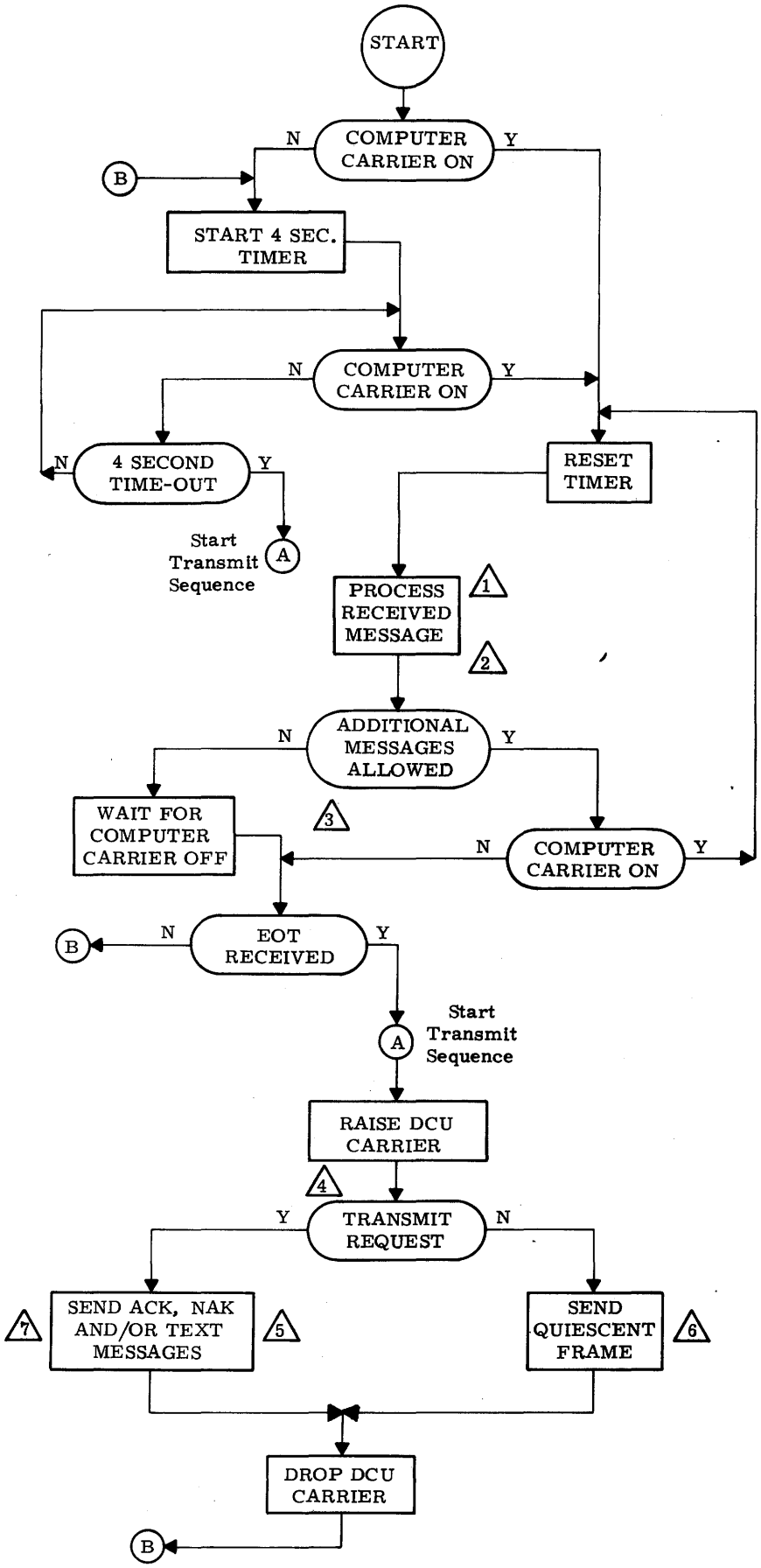
4 Quiescent Frames are optional from computer and are sent by DCU if no other message is to be sent in a transmission frame. In the absence of message exchanges, DCU sends quiescent frame each 4 seconds.

5 Move entry marker to start of message.

6 Graphic characters spell out error message to operator.

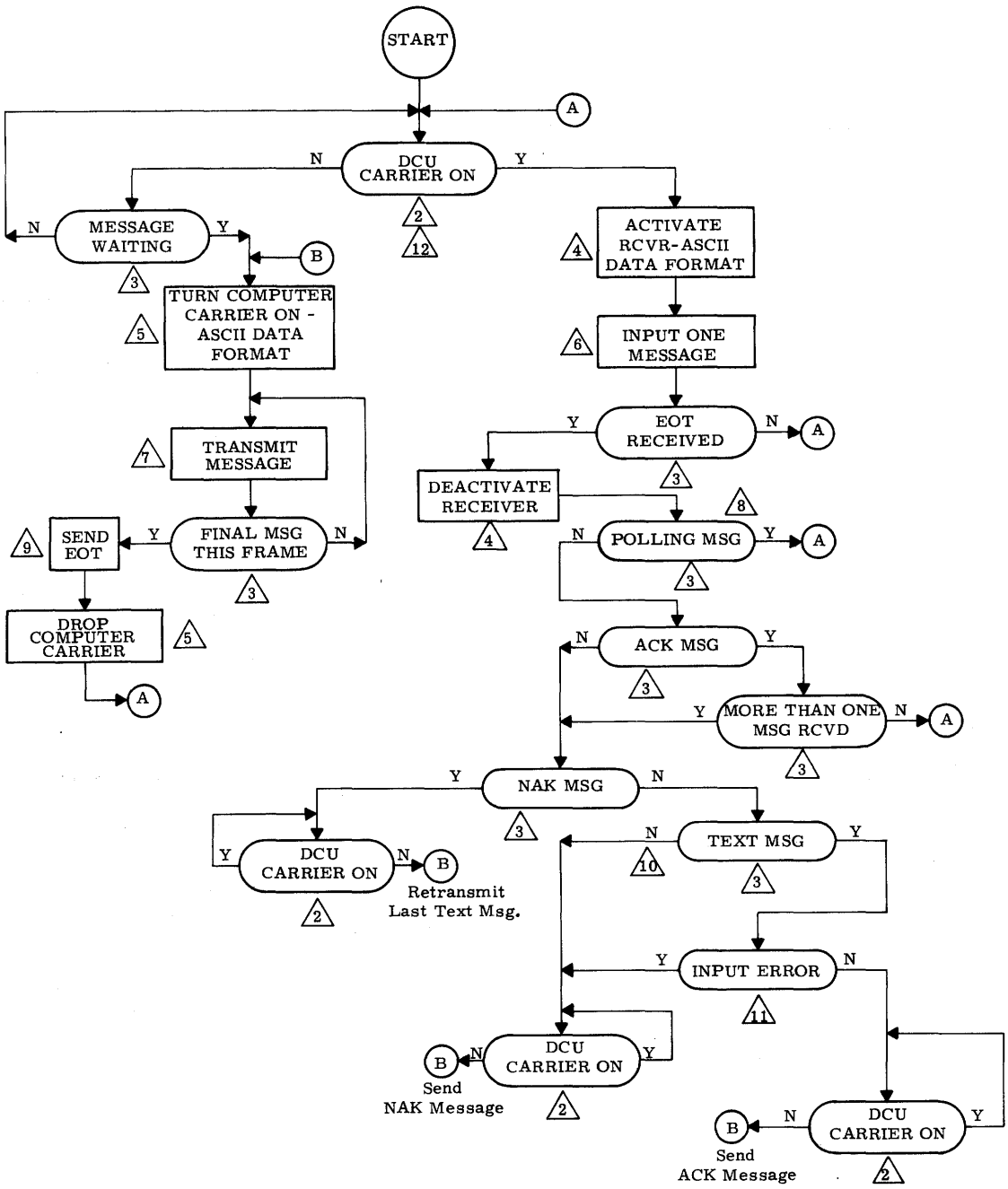
7 If valid SOH and ADR are not received by DCU, the entire message is ignored, and no ACK or NAK reply is sent.

Fig. OP. 6 Message Formats



1. Lock keyboard at STX time.
2. Unlock keyboard at LP time. If not a NAK message, replace double entry marker with single.
3. One NAK message allowed, or one ACK and one text message for each TMU are allowed.
4. May be a keyboard request or a retransmission request due to a NAK message reception. Keyboard locked until transmission is complete. Entry marker is double until non-NAK message (ACK or NUL) is received.
5. Any ACK or NAK message for TMU-A is sent first, then any ACK or NAK for TMU-B, then one text message if any are waiting. TMU-A and TMU-B messages are sent in alternate frames, if both are waiting.
6. SOH followed by EOT (preceded by four SYN's if DCU-775).
7. Terminal goes busy and computer messages are ignored from transmit request until transmission is complete.

Fig. OP. 7 DCU Message Sequencing



1. This flow chart is intended to illustrate generalized message flow. It does not chart any specific program.
2. OPR instructions addressed to receiver set up carrier on/off detection conditions. See ACD or SCU theory publications.
3. Determined by software.
4. OPR instructions addressed to receiver activate and deactivate data transfers. See ACD or SCU Theory publications.
5. OPR instructions addressed to transmitter turn carrier on, activate data transfers, and turn carrier off. Carrier is on when "request to send" signal from transmitter is "true". See ACD or SCU theory publications.
6. Characters are input via TIM function. JDR and IN instructions may also be used.
7. Characters are output via TOM function. JDR and OUT instructions may also be used. With SCU, hardware sends four SYN's prior to SOH.
8. Polling message is SOH followed by EOT.
9. EOT must immediately follow longitudinal parity character, or DCU reply or next message cannot be received until DCU timer runs out.
10. If message cannot be identified as one of the above, it must be in error.
11. Software detects longitudinal parity errors. Hardware detects lateral parity and other errors which may be detected with JNE. See ACD or SCU theory publications.
12. If no reply to last text message is received within a reasonable time, the Display Terminal was busy. Retransmit the last text message.

Fig. OP. 8 Computer Message Sequencing

If an unexpected question mark appears on a display during reception of a message from the computer, it is probably due to an error detected by the DCU during reception of the text portion of the message. "?" appears at the position of the entry marker when the error was detected, and informs the operator that the validity of the display is questionable.

If the DCU detects any error after the receipt of SOH and ADR, it will reply with a NAK message. If a valid SOH and ADR are not received, the message will be ignored. Messages addressed to a busy Display Terminal will also be ignored. A Display Terminal is busy when it is transmitting a message or when it is in local mode. Only Display Terminals associated with DCU-775 can be placed in local mode. DCU-765 does not feature local mode.

Errors detected by the DCU include lateral parity, incorrect message format, longitudinal parity, and on synchronous channels, the absence of a character in a character time position. On synchronous channels, the detection of any error by the DCU will cause a loss of synchronization. All messages from the computer must have at least four SYN characters preceding SOH, to re-establish synchronization.

Entry Marker Control Characters

Control of the entry marker position by the computer is effected by eight control characters which may be included in the text portion of information messages to the DCU. The octal codes for the entry marker control characters are listed in Table OP. 1, where they are identified by (3).

Because the DCU requires time to accomplish some of the entry marker control functions, some of the control characters must be followed by a number of DEL (177_8) characters at the higher baud rates, to allow time to reposition the entry marker. The control characters and the required number of DEL's at various baud rates used with GE-PAC systems are listed below. Failure to provide enough of these fill characters will result in an error detected by the DCU and a NAK reply.

PR — Page Return. Returns the entry marker to row one, position one. Must be followed by 4 DEL's at 4800 baud.

FF — Returns the entry marker to row one, position one and erases the display. Must be followed by 9 DEL's at 4800 baud.

LR — Line Return. Returns the entry marker to position one on the same line. Must be followed by 4 DEL's at 4800 baud.

LF — Line Feed. Moves the entry marker to the same position on the next row. No DEL's required at all baud rates.

RLF — Reverse Line Feed. Moves the entry marker to the same position on the preceding row. Must be followed by 4 DEL's at 4800 baud. Successive RLF's may substitute for one or more of the DEL's, e.g., if two RLF's are sent, the sequence at 4800 baud can be: RLF, RLF, DEL, DEL, DEL.

FS — Forward Space. Moves the entry marker forward one position. No DEL's are required.

BS — Back Space. Moves the entry marker back one position. Must be followed by 4 DEL's at 4800 baud. Successive BS's may substitute for one or more of the DEL's, e.g., if two BS's are sent, the sequence at 4800 baud can be: BS, BS, DEL, DEL.

TAB — Moves the entry marker to the position following the first vertical line encountered on the display. If there are no vertical lines on the display, the entry marker will not change. Must be followed by 1 DEL at 2000 baud, or 3 at 2400 baud, or 14 DEL's at 4800 baud.

No fill characters are required when LR is followed by LF. The DCU automatically inserts LF followed by LR, following the character contained in position 46, for each row or part of a row, in the text of messages transmitted to the computer. Where DCU-775 is involved, reference should be made to the Space Deletion Feature heading, earlier in this publication.

Longitudinal Parity Check

As is shown on Fig. OP. 6, a longitudinal parity character slot follows the ETX character in all DCU/computer messages. A character must be transmitted in the LP time position in all cases, and in most cases it will be a true longitudinal parity character. It is possible to delete the longitudinal parity check feature in the DCU. The only advantage gained thereby is a slight simplification in the computer program, and since the standard PCPD software includes the LP check and generation, the feature will not be deleted, in most cases. Where it is deleted, the computer should transmit DEL (177_8) in the LP slot. The DCU will transmit a true LP character in all cases (see Section 16 of the red Computer Maintenance manual or Section 19 of the blue Computer Maintenance manual for modification to defeat the DCU LP check).

The LP check provides an additional confirmation of accurate message transfer by accumulating a checksum which is representative of the total number of bits set in a message. The sum is accumulated from the character following SOH through the ETX character, and not including the parity bit. Each bit position in all characters is added, without carrying, as in the example in Table OP. 2. The LP character is checked by hardware in the DCU and by the program in the computer. A lateral parity bit, generated by hardware in both the DCU and the computer, is transmitted with the LP character.

Character	Bit Positions 6 5 4 3 2 1 0	Octal Code
Not Counted → SOH	0 0 0 0 0 0 1	001
ADR	1 1 0 0 0 0 0	140
NUL	0 0 0 0 0 0 0	000
FC1	1 0 0 0 1 0 1	105
FC2	0 1 1 0 0 1 0	062
STX	0 0 0 0 0 1 0	002
Text	Assumes even no. of bits set	
ETX	0 0 0 0 0 1 1	003
LP	0 (0 0 1 0 1 1 0) — LP Check Sum	026
Odd* Lateral Parity Bit		

*Parity will be odd on synchronous channels and even on asynchronous channels.

Table OP. 2 LP Character Generation

DOCUMENTATION

It is the intent of this publication to provide an understanding of the overall operation of the Video Display Subsystem and to provide a more detailed discussion of features, functions, and theory of operation which are not adequately covered elsewhere. Detailed theory of operation for the display controller and display terminal is provided in the vendor manuals which are shipped with the equipment. Detailed theory of operation for the computer including the basic I/O Buffer, the Asynchronous Communications Drive, and the Synchronous Communications Unit modules within the I/O Buffer, are provided elsewhere in the book set of which this publication is a part.

The publications and drawings which, when used in conjunction with this publication, provide an understanding of the detailed theory of operation of the entire Video Display Subsystem are described in the following paragraphs.

GE-PAC EQUIPMENT

Three book sets are provided with each GE-PAC computer system:

- Theory of Operation
- Computer Maintenance Manual
- System Drawings (model lists, logic, schematic, wire lists, cable drawings, and special system drawings)

The actual content of these books is determined by the functional modules which are provided in the system. Documents in these book sets which are pertinent to the Video Display Subsystem are listed in Tables DOC.1, DOC.2, and DOC.3.

TITLE	PUBLICATION NUMBER		VOLUME*
	4010	4020	
Input/Output Buffer	4820A	4202X-1	III
Video Display Subsystem	4291/2-T	4291/2-T	III
Asynchronous Communications Drive II	4202X-23 and -27	4202X-23 and -27	III
Synchronous Communications Unit II	4202X-26	4202X-26	III
SCU (In a 4010B)	4820B/ 4821A06-T		
SDI (In a 3010 or 3010/2)	3010AH14-T		

*Theory of Operation manuals for 4010 Computers. Servicing Instructions for 4020 Computers.

Table DOC.1 Computer Theory Publications

TITLE	PUBLICATION NUMBER	SECTION 4010	SECTION 4020
Asynchronous Communications Drives	4202X460/560/-461/561	20	17
Synchronous Communications Units	4202X465/565/-466/566	21	17
765/775 Video Displays	4291/2-T	19	16
SCU (In a 4010B)	4820B/4821B06-M		
SDI (In a 3010 or 3010/2)	3010AH14-M		

Table DOC.2 Computer Maintenance Publications

TITLE	DRAWING NO.	UNDER TAB
System Configuration Diagram or System Hardware Block Diagram	Special	System Drawings
Hardware Address Summary	Special	Installation Drawings
System API Logic	Special	API
Logic, ACD	70C179053 (4020)	I/O Buffer
	70C180347 (4010)	I/O Buffer
Logic, SCU	68C982325 (4020)	I/O Buffer
	70C180349 (4010)	I/O Buffer
Logic, Basic I/O Buffer	68C973839 (4020)	I/O Buffer
	70C180136 (4010)	I/O Buffer

Table DOC.3 Computer System Drawings

DISPLAY CONTROLLER AND TERMINAL

The vendor manuals supplied with the DCU and Display Terminal are arranged as follows:

- Volume 1 - System
- Volume 2 - Display Terminal
- Volume 3 - Display Controller

Volume 1 contains Product Performance Specifications and Design Descriptions, which are in effect, the theory of operation of the DCU and Display Terminal. In addition, Volume 1 contains installation instructions and a trouble isolation guide.

Volume 2 contains theory of operation for the display monitor and keyboards, schematics, timing diagrams, and parts lists.

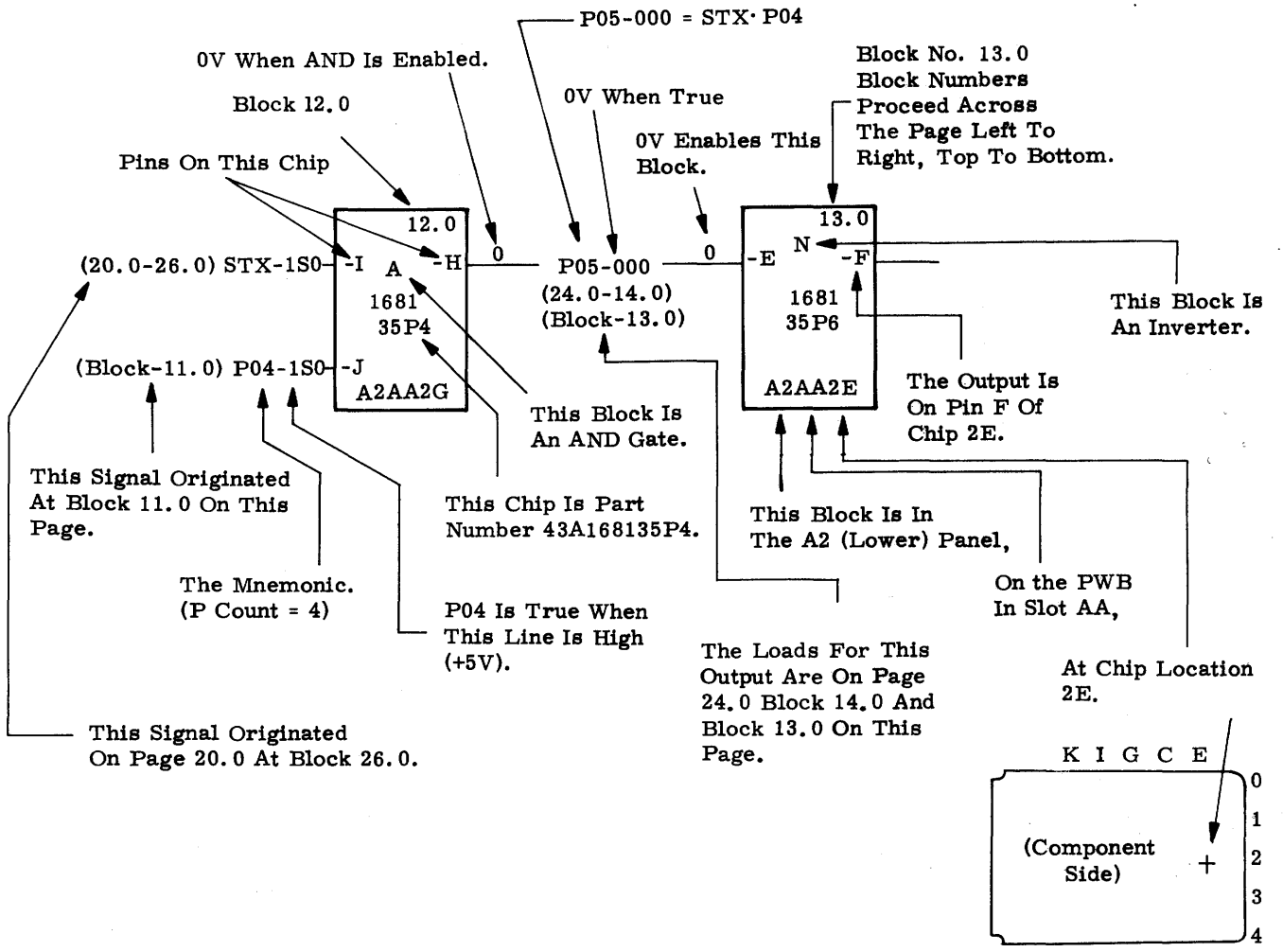


Fig. DOC. 1 DCU Logic, AND Gate And Inverter

Volume 3F contains the DCU logic, discrete circuit schematics, timing diagrams, detailed flow charts, a mnemonic dictionary, and parts lists for the DCU. The "F" in Volume 3F indicates that Volume 3 is the final book in the set. If any other vendor volume is found to have an "F" designation, the book set is nonstandard.

DCU Logic

The vendor documents most frequently referred to in this publication and probably the most used are:

- Logic, DCU-765; 59B201143.
- Logic, DCU-775; 59B202508.
- Logic, DCU-775 using ROM (vendor S/N 1000 and above); 59B203361.
- Logic, DCU-785; 59B203368.

Both DCU logic drawings are automated and printed by a computer on a line printer. At first glance, the symbols may appear foreign, but a little study will reveal that there are many similarities to the logic depicting GE-PAC modules. Most of the logic depicts the functions of dual in-line, flat pack, integrated circuits, similar to those used on GE-PAC PWB's. While the logic notations differ somewhat, the AND gate, OR gate, inverter, and flip-flop functions are virtually identical to those in GE-PAC modules:

- AND Gates; all inputs high (3.6V) hold the output low (0V).
- OR Gates; any input high holds the output low.
- Flip-Flops; JK flip-flops with internal steering are used.
- Wired Logic Functions; where the outputs of two gates are tied together by a conductor run, if the output of either gates goes low, it holds the outputs of both low. This is illustrated on Fig. DOC. 2.

The meaning of the symbols and notations used on the logic drawings is depicted on Fig. DOC. 1, DOC. 2, and DOC. 3. The logic is divided into functional groups designated by 4-digit numerals. The logic drawing pages proceed sequentially through these groups, starting with 1000 and ending with 5410 (DCU-765) or 5800 (DCU-775). PWB's, functions, and locations are listed on Tables DOC. 4 and DOC. 5.

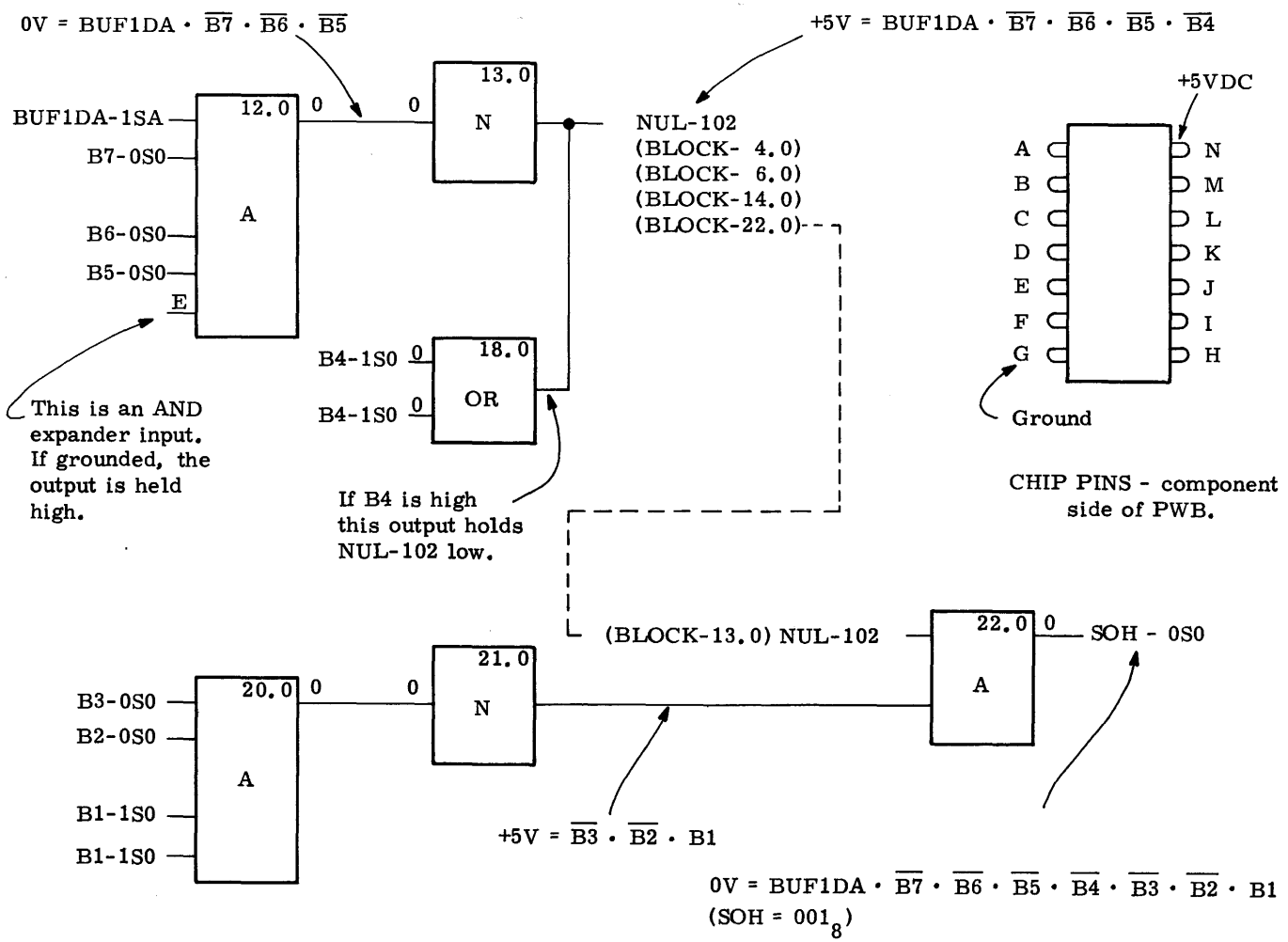


Fig. DOC. 2 DCU Logic, Derivation of Logic Equations

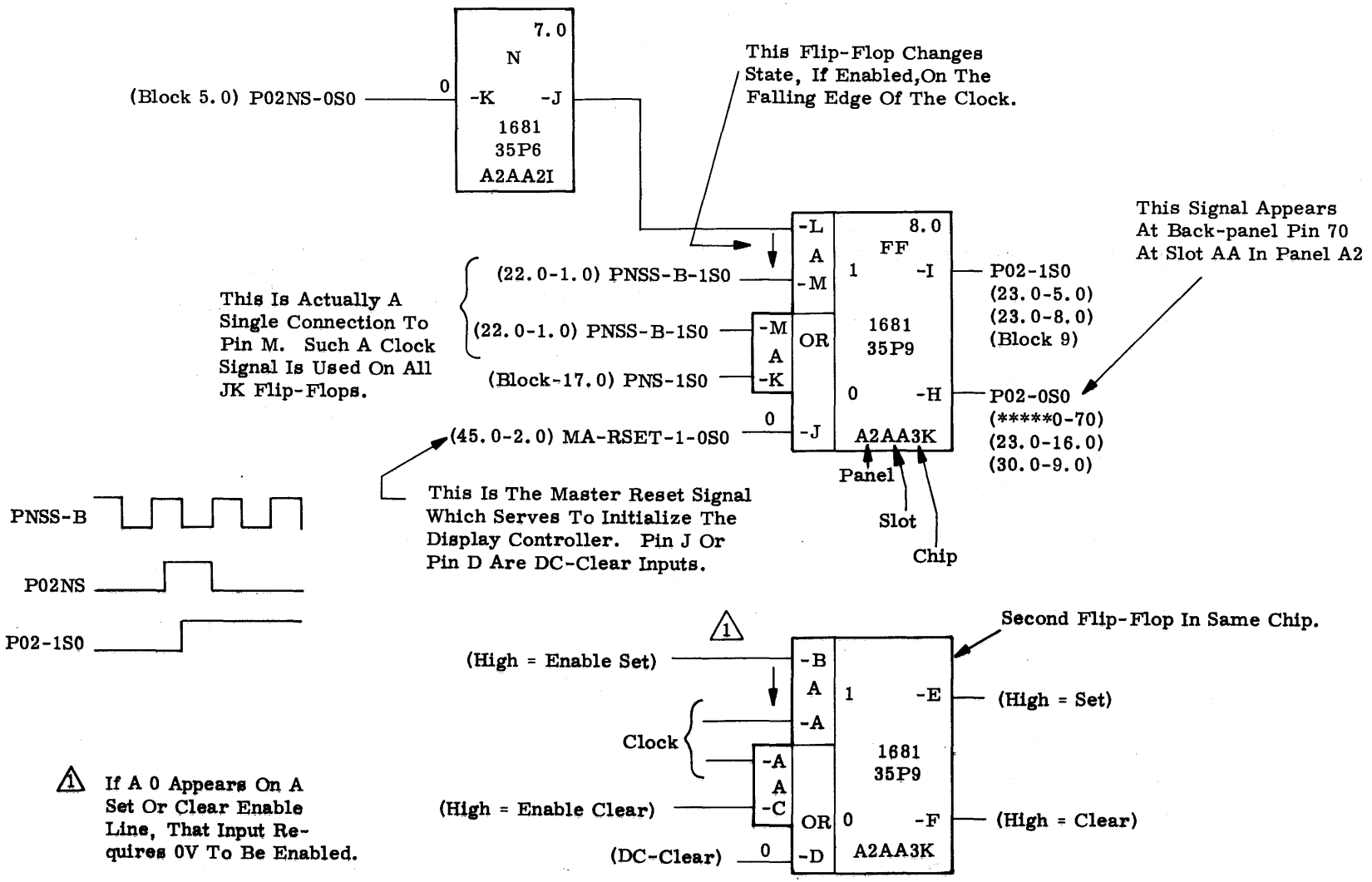


Fig. DOC.3 DCU Logic, Flip-Flops

FUNCTION	LOGIC GROUP	PWB PART NO.	PWB LOCATION
DLC1	1000	59D200510	A1AA
DLC2	1200	870	A1AC
DLC3	1300	880	A1AE
DLC4	1700	890	A1AG
DLC5	1400	900	A2AA
DLC6	1100	910	A2AC
DLC7	1500	920	A2AE
DLC8	1600	930	A2AG
BC1	3000	610	A2BD
BC2	3100	620	A2BB
BC3	3200	630	A2BC
BC4	3300	640	A2BE
BC5	3400	650	A2BF
BC6	3500	660	A2BG
BC7	3600	670	A2BH
BC8	2000	520	A1BD
BC9	2100	830	A1AT
BC10	2200	720	A2AV
BC11	2300	750	A2AX
BC12	2400	760	A1AR
BC13	2500	730	A1AN
BC14	2600	850	A1AX
TMU1	4100/5100*	840	A2AN/A2AR*
TMU2	4200/5200	800	A1AT/A1AV
TMU3	4300/5300	860**	A1AZ/A1BB
TMU4	4400/5400	530	A1BF/A1BH
TMU5	4000/5000	59D200940	A2AN/A2AR

*TMU-A/TMU-B **May also be 59D201070

Table DOC.4 DCU-765 PWB Complement

(Note for Table DOC.5 DCU 775 PWB Complement.)

NOTE

DCU-775's with vendor S/N 1000 and above use Read-Only Memory instead of BC1 through BC7 (5200-5800). Each TMU implemented has one ROM board. ROMA (5200) for TMU-A and ROMB (5300) for TMU-B. All DCU-785's and 786's use the ROM character generator.

FUNCTION	LOGIC GROUP	PWB PART NO.	PWB LOCATION
DLC1	1200/1300	59D200510	A1AB & A1AC
DLC2	1400	202150	A1AD
DLC3	1500	160	A1AE
DLC4	1600	170	A1AF
DLC5	1700	180	A1AG
DLC6	3300	270	A2AA
DLC7	3400	280	A2AB
DLC8	3500	290	A2AC
DLC9	3600	190	A2AD
DLC10	3700	300	A2AE
DLC11	3800	220	A2AF
DLC12	3900	202310	A2AG
BC1	5400	200610	A2BD
BC2	5200	620	A2BB
BC3	5300	630	A2BC
BC4	5500	640	A2BE
BC5	5600	650	A2BF
BC6	5700	660	A2BG
BC7	5800	670	A2BH
BC8	3000	200520	A1BE
BC9	5100	201810	A2AZ
BC10	5000	202110	A2AY
BC11	4900	200750	A2AX
BC12	4800	201890	A2AW
BC13	4700	201820	A2AV
BC14	2700	200850	A1AX
BC15	4600	202200	A2AU
TMU1	2500/2600*	202490	A1AV/A1AW*
TMU2	2300/2400	202210	A1AT/A1AU
TMU3	2800/2900	201070**	A1BA/A1BC
TMU4	3100/3200	200530	A1BG/A1BH
TMU5	4000/4100	202320	A2AL/A2AM
TMU6	4400/4500	230	A2AR/A2AS
TMU7	4200/4300	240	A2AN/A2AP
TMU8	2100/2200	59D202250	A1AR/A1AS

*TMU-A/TMU-B **May also be 59D200860

Table DOC.5 DCU-775 PWB Complement

THEORY OF OPERATION

BASIC FUNCTIONS

The major components of the Video Display Subsystem are depicted on Fig. INT. 1. As that illustration indicates, the subsystem provides an interface between a Display Terminal operator and the Arithmetic Unit and Core Memory in the GE-PAC computer. Data exchanges between the computer and the Display Controller Unit are accomplished over a serial data communications interface which complies with EIA Standard RS-232C. The connection between the DCU and the computer can be made by a direct cable 50 feet or less in length, or the computer and DCU sites can be distant, in which case a digital data set at the computer site, and another at the DCU site, provide communication via telephone lines.

The primary subsystem module within the computer is the Asynchronous Communications Drive or the Synchronous Communications Unit, in the I/O Buffer. The theory of operation of these modules is described under the I/O Buffer tab in this Volume.

Fig. THEORY. 1 illustrates the flow of data through the two primary subsystem modules which are external to the computer's Central Processor or Central System Unit. Frequent references to Fig. THEORY. 1 will be useful when using the material in this section, and when using the logic descriptions in the DCU and Display Terminal vendor manuals.

In its simplest form, the Video Display Subsystem can be considered as a memory capable of storing a number of characters by circulating the characters through the memory in a continuous stream. As the characters are transferred from the output of the memory back to the input, they are tapped off the memory stream, converted to a television signal, and displayed on a television monitor. Each 1 MHz delay line stores a full display - up to 1012 display characters.

The content of the memory stream may be changed from time to time by messages transferred over the communications channel from the GE-PAC computer. The memory stream content may also be changed from the keyboard.

The principal memory element is a 1 MHz delay line in each TMU. The data stored in the 1 MHz delay line is continuously recirculated, thereby refreshing the stored contents, unless new data is to be entered from the computer or the keyboard.

The Data Line Controller portion of the DCU processes messages received from the computer and provides temporary storage for text characters while they wait for the proper character slot in the memory stream to circulate through the 1 MHz delay line input gating logic. The DLC also generates messages transmitted to the computer, and when those messages contain text, extracts the characters from the memory stream, without erasing them, and places the characters in the proper sequence in the text portion of the message.

The MOS FET memory is an all solid state serial shift register memory, which stores and recirculates each 46 character display row while it is "painted" on the Display Monitor.

The character register and decoder are shared by the two TMU's. The TMU multiplexer gates data from the two TMU's into the character register on a time-shared basis. The video decoder generates appropriate video pulses in response to the current contents of the character register and the current display line - one of 21 lines per display row.

The horizontal and vertical sync and blanking pulses are added to the video wavetrain by the video driver. Where the keyboards are implemented, clear to send pulses are also added to the wavetrain by the driver. The clear to send pulses are used by the keyboard to enable the shifting of data onto the keyboard data line. They are stripped from the wavetrain by the keyboard, and the resultant display video is applied back to the monitor to generate the display picture.

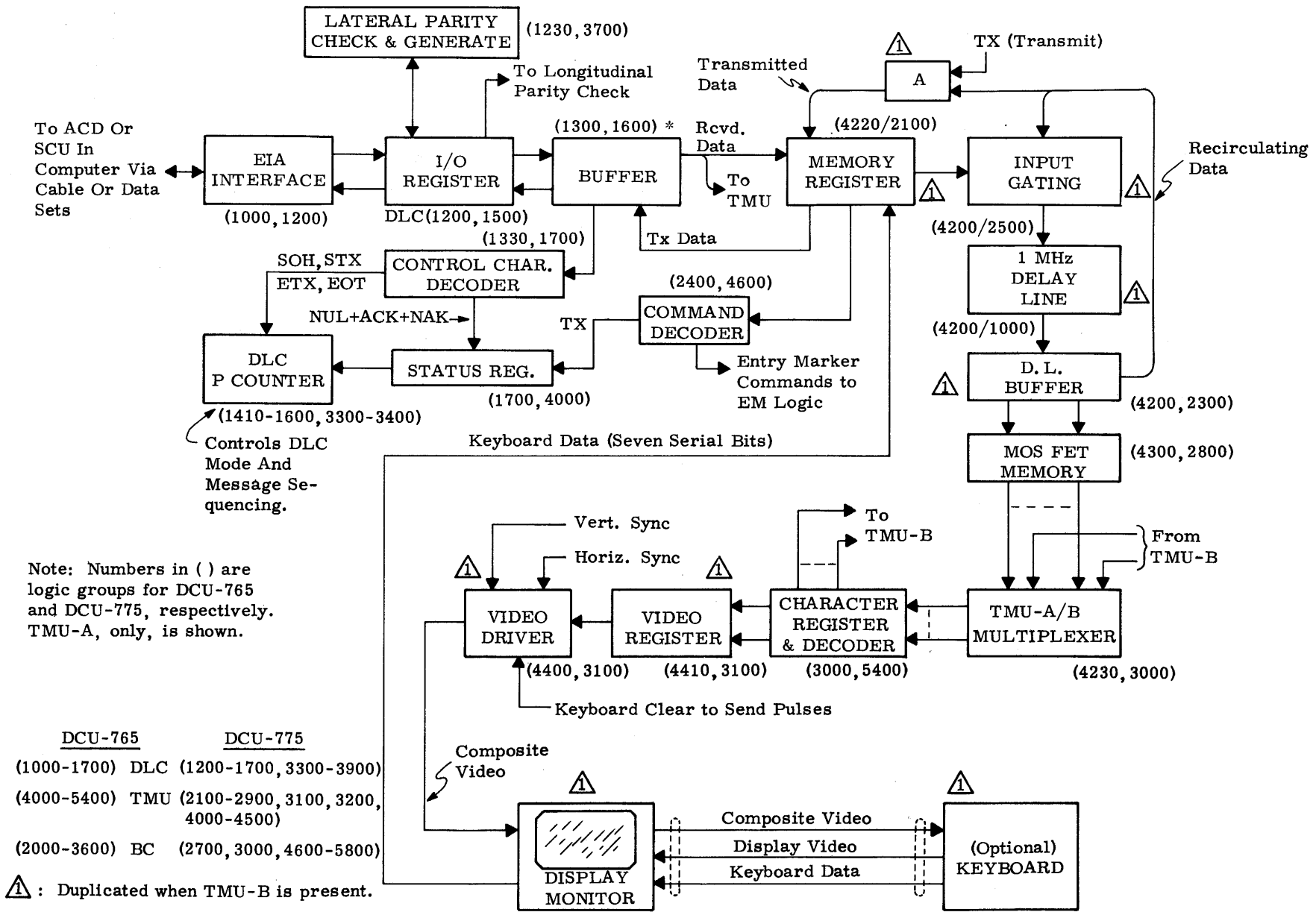
TV CHARACTER GENERATION

CRT Scanning

The Display Monitors use a standard 525 line television display, with a horizontal sweep frequency of 15,750 lines per second, and a vertical sweep frequency of 60 fields per second and 30 frames per second.

Fig. THEORY. 2 illustrates the development of the 525 horizontal lines on the cathode ray tube screen. As in home entertainment sets, interlaced scanning of the TV screen is used. One-half of each TV picture is displayed for each vertical sweep. Each half-picture field consists of all of the even numbered lines on the screen or all of the odd numbered lines on the screen. A complete picture, consisting of two fields is referred to as one frame.

A composite video signal, consisting of the graphic character video bits, the horizontal and vertical sync, and the blanking pulses, is transmitted from the TMU, through the keyboard, to the Display Monitor on a coaxial cable. The sync pulses are separated from the video and used to synchronize the horizontal and vertical deflection coils around the neck of the cathode ray tube (CRT). The magnetic field developed by the horizontal deflection coil moves the electron beam from left to right across the screen in a little less than 64 microseconds. At the same time, the field in the vertical deflection coil moves the electron beam from the top to the bottom of the screen in a little less



Note: Numbers in ( ) are logic groups for DCU-765 and DCU-775, respectively. TMU-A, only, is shown.

DCU-765		DCU-775
(1000-1700)	DLC	(1200-1700, 3300-3900)
(4000-5400)	TMU	(2100-2900, 3100, 3200, 4000-4500)
(2000-3600)	BC	(2700, 3000, 4600-5800)

1 : Duplicated when TMU-B is present.

*DCU-775 has five 7-bit parallel buffers, connected in series.

Fig. THEORY.1 Simplified Block Diagram, Display Subsystem

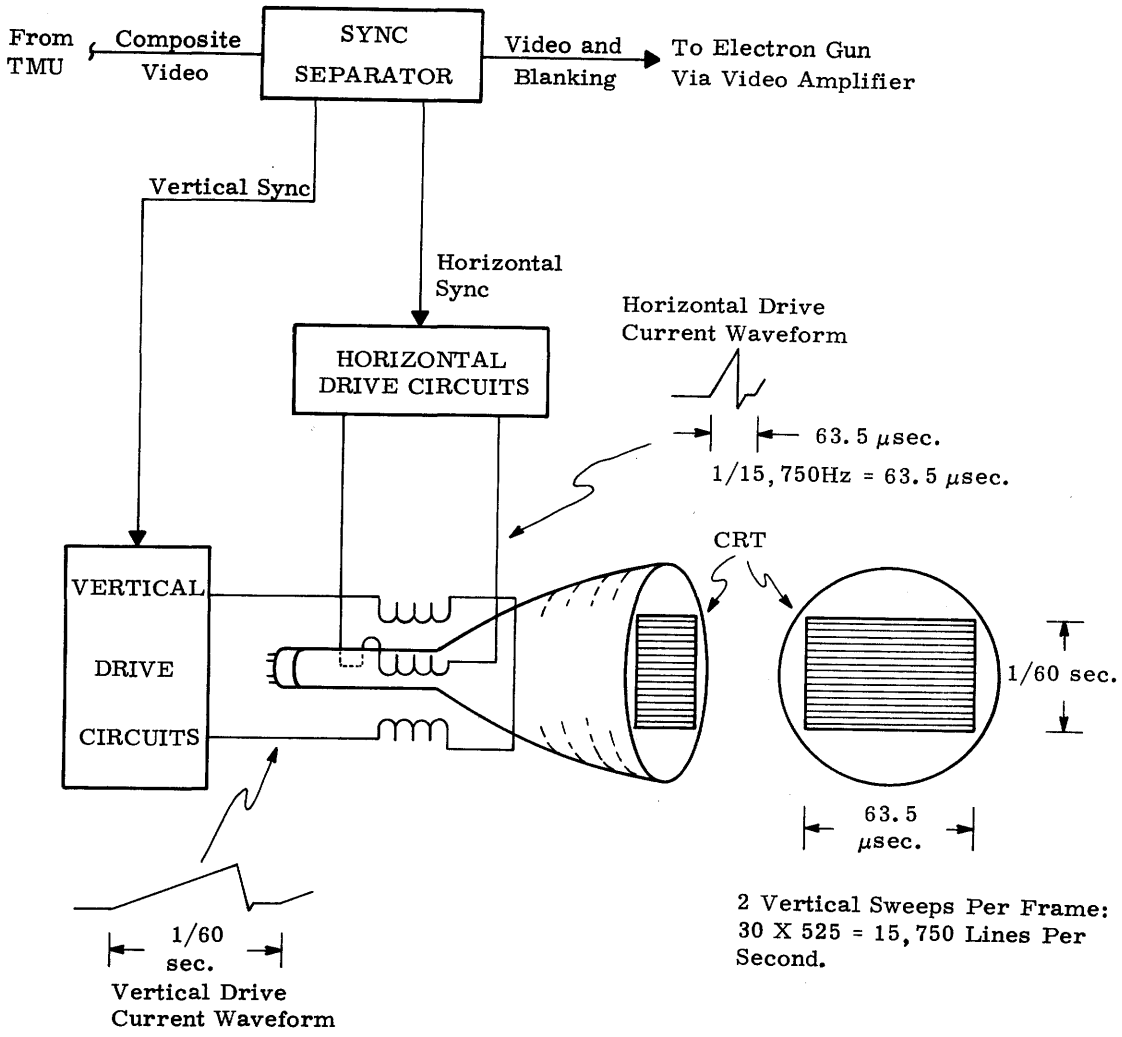


Fig. THEORY.2 CRT Scanning

than 1/60 second. The slower vertical deflection allows 262.5 horizontal raster lines to be traced across the screen each 1/60 second.

The deflection circuitry starts the second field in a frame one line lower than the point where the first frame started. The second vertical sweep fills in the remaining 262.5 lines, and the 525 line frame is completed.

At the completion of each sweep from left to right the electron beam returns very rapidly to the left of the screen. At the completion of each vertical sweep, the electron beam returns very rapidly to the top of the screen. The electron gun is virtually cut-off during these "fly-back" times, and the retrace cannot be seen. On a properly adjusted display monitor, the areas between characters should also be below the visible level.

Character Generation

Fig. THEORY.3 shows the form of the 22 row display. Each horizontal row is capable of displaying 46 characters and occupies 21 horizontal TV raster lines. It would appear, then, that 25 rows are available in the 525 horizontal lines (525/21 = 24.74). Vertical fly-back uses up the remaining 0.26 of a row. The TMU memory stream is 25 rows long and 22 of those rows store characters for display.

The lower portion of Fig. THEORY.3 illustrates the form of the characters on the display and the waveforms which are generated by the video decoder to write the letter "A" in a character slot.

Each row in the MOS FET memory and on the display contains 50 character slots and each character slot occupies 1.134 μs of horizontal display time. Two character slots on the left edge of the display and two slots on the right edge do not store any graphic characters, leaving 46 slots centered on the display to depict the characters. The horizontal line time then, is 50 X 1.134 = 56.7 μs. The remaining 6.8 μs of the 63.5 μs horizontal line time is used up in the horizontal fly-back to return the electron beam to the starting position at the left of the screen.

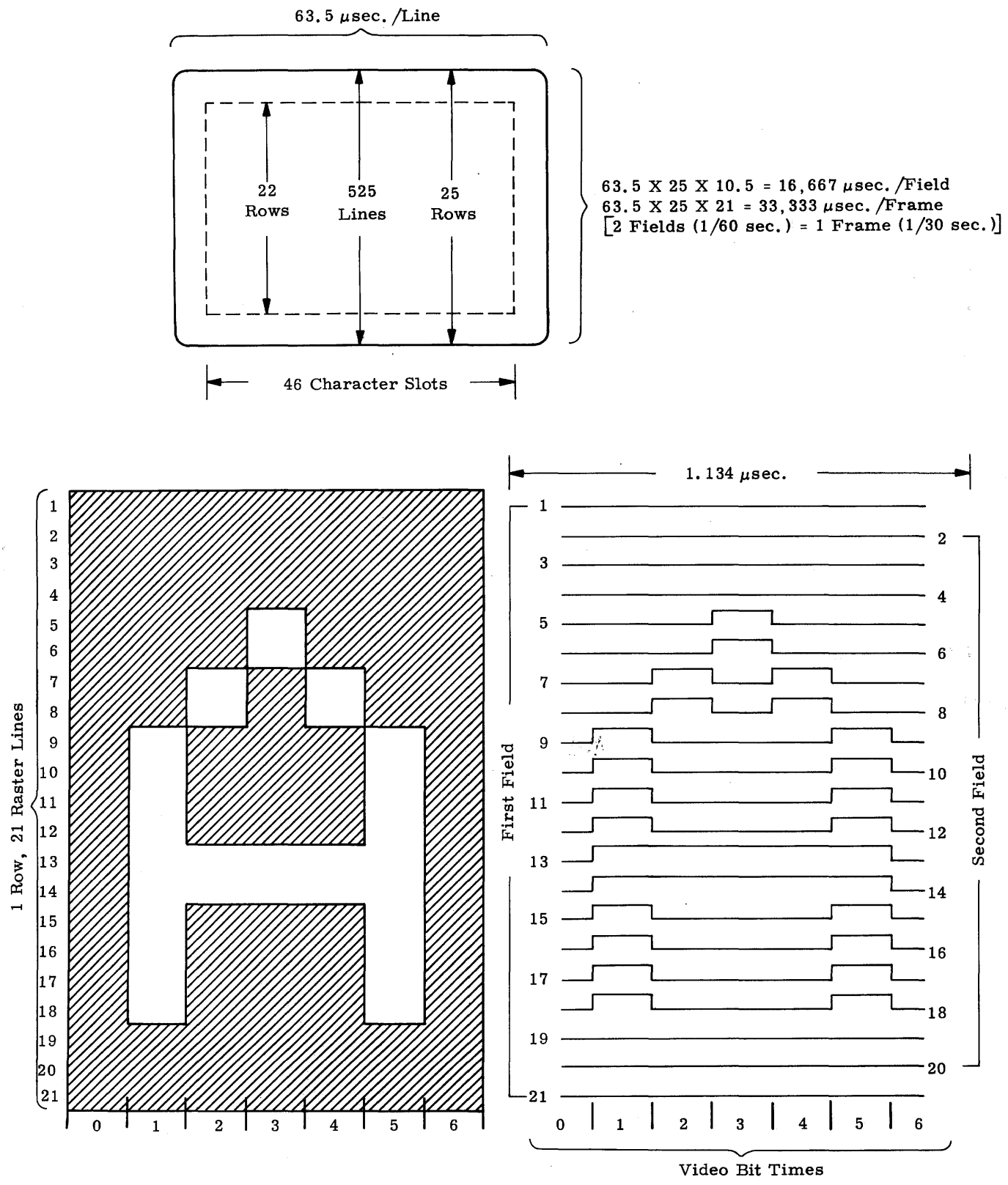


Fig. THEORY. 3 TV Character Generation

Each row is comprised of 21 horizontal display lines. Most of the graphic characters occupy lines 5 through 18 in the row and the remaining lines provide the separation between characters in adjacent rows.

Each 1.134 μs horizontal character slot is subdivided into seven video bits. Most of the graphic characters occupy bit positions 1 through 5 and the remaining bit positions provide separation between adjacent characters. Each video bit is on the cable to the Display Terminal 0.162 μs in duration.

The waveforms in the lower right-hand area of Fig. THEORY.3 are applied to the electron gun in the CRT and intensify the electron beam when positive. A comparison of the waveforms with the diagram of the letter "A" to the left should make the method of character generation apparent. One-half of each character is painted during each display field. The odd numbered lines are scanned during the first field and the even lines during the second. Of course, each character is not painted individually. Each raster line traces through 50 character time slots, of which 46 are in the display area.

The composition of all of the graphic characters in terms of horizontal lines and video bits is illustrated in Volume 1 of the display equipment vendor manuals on the following pages:

- DCU-765; 59A201699, page 28.
- DCU-775; 59A201703, page 68.

SUBSYSTEM TIMING

One of the more confusing factors in the study of any CRT display system is the timing relationships. In order to minimize this confusion, the timing in the Video Display Subsystem may be thought of in terms of three principal timing areas:

- Timing related to the TMU memory stream.
- Timing related to the CRT displays.
- Timing related to the communications channel.

Memory related timing and display related timing are under the control of a 12 MHz crystal oscillator in the Basic Controller. Since the end result of all Display Controller Unit timing, except for the communications channel, is the development of the CRT displays, it is necessary that the timing be such that the vertical sweeps on the CRT occur at 60 fields per second. This is necessary to minimize flicker on the TV picture which can occur if the picture field frequency is not at,or very near,the power line frequency. As will be shown, the count-down ratios used to arrive at the timing signal frequencies used in the DCU require a crystal oscillator frequency of 12.348 MHz.

Fig. THEORY.4 illustrates the generation of the DCU timing signals. Elsewhere in this publication and in the vendor manuals, the timing signals may be referred to as nominal values. As an example, logic term 1MCA refers to the nominal 1 MHz timing signal. The exact frequency is 1.029 MHz.

Memory Timing

Each display row consists of 21 TV raster lines, ten of which are "painted" during the even display fields. The remaining eleven lines are painted during the odd display fields. The memory timing is arranged so that the data contained in each memory character slot is passed into the video generation logic just before the time the CRT electron beam is tracing through that character position. This means that each character must pass into the video logic ten times during one row time, and eleven times during the next row time, in alternate fields.

The primary memory is the 1 MHz delay line, the delay line buffer, which is a 7-bit serial shift register, and the recirculation logic, which refreshes the delay line data in the absence of new data. The total delay of this loop is exactly as long as is required to paint one half of a field. It is necessary to recirculate the delay loop two times to paint one field. It is recirculated four times to paint one 2-field frame. Since the constantly shifting delay line loop cannot hold the characters in one position long enough to paint one ten or eleven line row, a secondary memory, consisting of two solid state shift-register delay networks, is provided. This MOS FET memory holds each row during its 10 or 11 line display time. Fig. THEORY.5 illustrates the relationships of the two memories to each other and the remainder of the display controller.

The delay line memory recirculation time is 8333.3 μs or 8-1/3 ms. Since two passes are required to scan all 25 rows (22 of which are in the display area), two recycle times correspond to the time for one display field: 16 2/3 ms (1/60 sec.). It is convenient, then, to arrange the memory rows in an alternating sequence, so that rows 1 through 13 are scanned in the first half of the field, during the first delay line pass, and rows 14 through 25 are scanned during the second pass.

Fig. THEORY.6 shows the memory row sequencing and indicates the timing relationships in the 1 MHz and MOS FET memories. Each time the row count is updated, flip-flop LD-MF1 is toggled. When LD-MF1 is set, each of the 49 characters in a row is shifted into the Accumulation Register in the MOS FET memory. Each clock pulse shifts one 7-bit character into the accumulator in parallel. Because the accumulator is loaded at alternate delay line row times, the rows are stored in the delay line loop in the order indicated at the top of Fig. THEORY.6. The full 25 rows will be shifted out of the Delay Line Buffer in two passes through the delay line loop, which takes 2 X 8333 μs = 16 2/3 ms. Since interlaced scanning is used, each

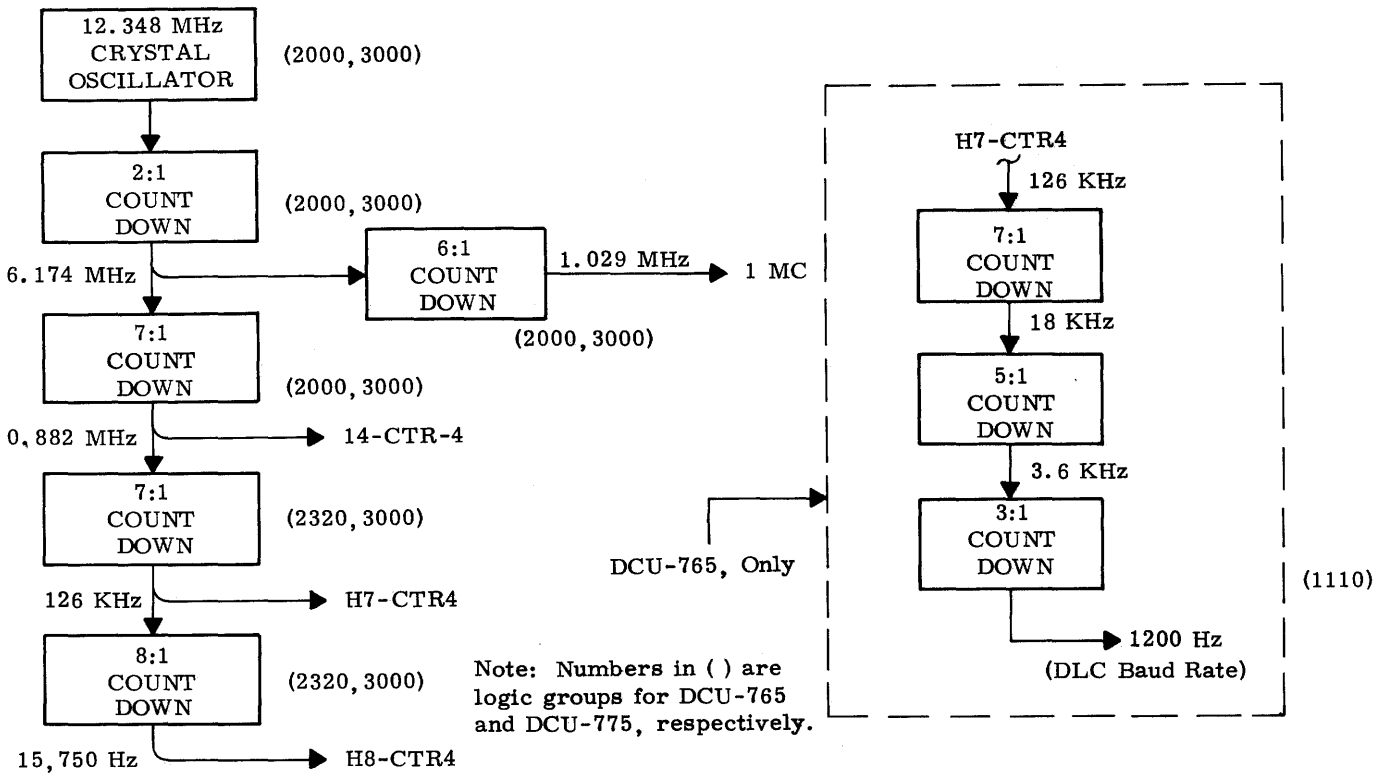


Fig. THEORY.4 DCU Timing Signals

row must be displayed twice in one TV frame, and the delay line loop must be recirculated four times to paint a complete two-field frame.

If new information is to be written into the delay line loop, the delay line write gate, DLWGT, goes true at the time the appropriate character slots are passing from the delay line buffer to the DL-DATA-IN flip-flop. When DLWGT is true, new data from the memory register is written into each character slot, rather than the recirculated character. As shown on Fig. THEORY.1, the memory register may contain new data from the computer or from the keyboard.

The MOS FET memory is contained on the TMU3 printed wire board. This board contains several MOS FET (Metal Oxide Semiconductor - Field Effect Transistor) integrated circuit cans, eight of which contain two 50-bit serial shift registers. Two interchangeable, but different, PWB's may be used for TMU3. Either PWB type may be found in the TMU3 slot, and where TMU-B is implemented, the TMU3 boards may be the same or different. The PWB drawings and schematic, which are provided in Volume 3F of the vendor manuals, are:

Part No. 59D200860, schematic 59D200861.

Part No. 59D201070, schematic 59D201071.

As is shown on Fig. THEORY.6, the MF1-CP1, CP2 clock signal runs at 0.147 MHz when LD-MF1 is true. These clock pulses are 6.8 μs apart and occur just as a complete character is held by the delay line buffer. They shift each 7-bit character into the MOS FET Accumulator Register. By the time LD-MF1 goes false, an entire row will have been shifted into the accumulator. Each of the 50 shift register stages then holds one bit of the 50 characters in a display row time. Forty-six of these characters may represent a graphic symbol to be displayed. The remaining characters, including any which are out of the text display area, will represent spaces (040_8).

When the accumulation is complete, the accumulator holds the row briefly. Just prior to the display time for the row, another burst of pulses occurs on the MF1-CP1, CP2 clock line. These are 0.882 MHz, or 1.134 μs apart. The second burst of clock pulses shifts the row into the Recirculation Register as the preceding display row recirculates for the last time - the tenth or eleventh time.

The MF1-TO-MF2 gate allows the transfer of the accumulator contents to the recirculator when true. When MF1-TO-MF2 is false, each recirculator output is fed back to the corresponding input, to refresh the recirculator contents.

The recirculator clock, MF2-CP1, CP2 also runs at 0.882 MHz, or 1.134 μs between pulses, which is the time required to paint one character position on the display. The MF2-CP1, CP2 clock stops for 6.8 μs during each horizontal retrace time, to hold the recirculator until the next display line begins.

When the character position corresponding to the current entry marker address is shifted into the accumulator, MFA-E-MKR (MFB-E-MRK in TMU-B) goes true. This signal remains true for 0.972 μs, and, therefore, sets the eighth bit in the MOS FET memory path. As shown on Fig. THEORY.5, the display characters occupy 7 of the eight parallel MOS FET paths, and the entry marker passes through the eighth path. During the character times when the entry marker bits pass by the output of the recirculator, the entry marker is painted at the top of the current display character slot.

Display Timing

The logic which develops the timing signals related to the television type display, provides timing control for several other areas within the Display Controller. If the display time relationships are understood, the analysis of the remainder of the Display Controller logic, especially the video generation logic and the entry marker control logic is simplified.

As has been mentioned, the Display Monitors utilize a scanning method which is identical to home entertainment TV sets. The display consists of 525 horizontal raster lines (523 of which are in the visible area), with 262.5 of these in each TV field, during which, one-half of the picture information is "painted".

Each field is scanned in 1/60 second, and each frame (or "page") in 1/30 second. The horizontal line frequency is 15,750 Hz (15,750/30 = 525 lines). Each line, then, must be blanked and re-synchronized every 63.5 μs. Each field must be blanked and re-synchronized every 16.667 ms.

During the display time, the horizontal synchronization logic generates pulses at a rate of 15,750 pulses per second, or 63.5 μs apart. These pulses are 4.5 μs wide, and occur during the time the MOS FET Recirculation Register is holding. They blank the CRT trace and synchronize the horizontal oscillator in the monitor, which generates the horizontal sweep current. These pulses are developed from a combination of counter outputs, as shown on Fig. THEORY.7.

At the end of each 16 2/3 ms field, the vertical synchronization logic generates pulses at a rate of 60 pulses per second. These pulses are 190.5 μs, or 3 horizontal line times wide. The V-SYNC-2 pulses are generated at the trailing edge of the QTR-FR1 and ODD-FLD gates as shown on Fig. THEORY.7. They blank the CRT trace and synchronize the vertical oscillator in the monitor, which generates the vertical sweep current.

Fig. THEORY.7 also indicates that the H-SYNC-2 and V-SYNC-2 pulses are combined before application to the driver which sends the composite video signal to the Display Monitor.

Because of the odd number of line times in each frame, each field must either start half of a line across the display area, or end half of a line across the display area (262.5 lines per field). Television monitors require a set of equalization pulses to control the correct start of each display field, and these are developed by the horizontal sync logic just before and during the V-SYNC-2 time.

The BEF-VS and V-SYNC-2 inputs to that logic control the development of the equalization pulses, part of which are shown on the lower timing diagram on Fig. THEORY.7. Twelve equalization pulses are developed prior to the start of each even field, and eleven are developed prior to each odd field. Since these pulses occur at twice the horizontal sync rate, they cause the monitor to reposition the CRT electron beam at the correct spot at the beginning of each field.

As is shown on Fig. THEORY.7, the SYNC-A-0S0 pulses are negative going. The vertical sync pulse is interrupted briefly by the equalization pulses which are developed on the H-SYNC-2 lines.

$$\text{SYNC-A-0S0} = (\text{H-SYNC-2}) \cdot \overline{(\text{V-SYNC-2})} + \overline{(\text{H-SYNC-2})} \cdot (\text{V-SYNC-2})$$

The vertical synchronization circuits in the monitor employ a low pass filter, which allows the 190.5 μs vertical sync pulse to pass, but blocks the higher frequency equalization and horizontal sync pulses.

Asynchronous Communications Channel Timing

Fig. THEORY.8 illustrates the asynchronous channel timing scheme and the proper PWB jumper pin selection at the ACD in the computer's I/O Buffer. The baud rate used on asynchronous channels is low enough that the stability of the timing oscillators used at each end may be relied upon to maintain accurate timing. The receivers at each end utilize the start bit which precedes each character to establish the time position of the received data clocks which shift each character bit into the input registers.

The display controller transmits only one stop bit following each character. The ACD may transmit either one or two stop bits, as selected by the green pin jacks on the PATA1 board, but the selection of one start bit per character, as shown on the illustration, will allow slightly faster message exchanges.

Synchronous Communications Channel Timing

Fig. THEORY.9 illustrates the synchronous channel timing scheme and the proper PWB jumper pin selection at the SCU in the computer's I/O Buffer. The higher baud rates used on the synchronous channels and the absence of any time between characters, dictate the synchronism of the baud rate clocks at each end.

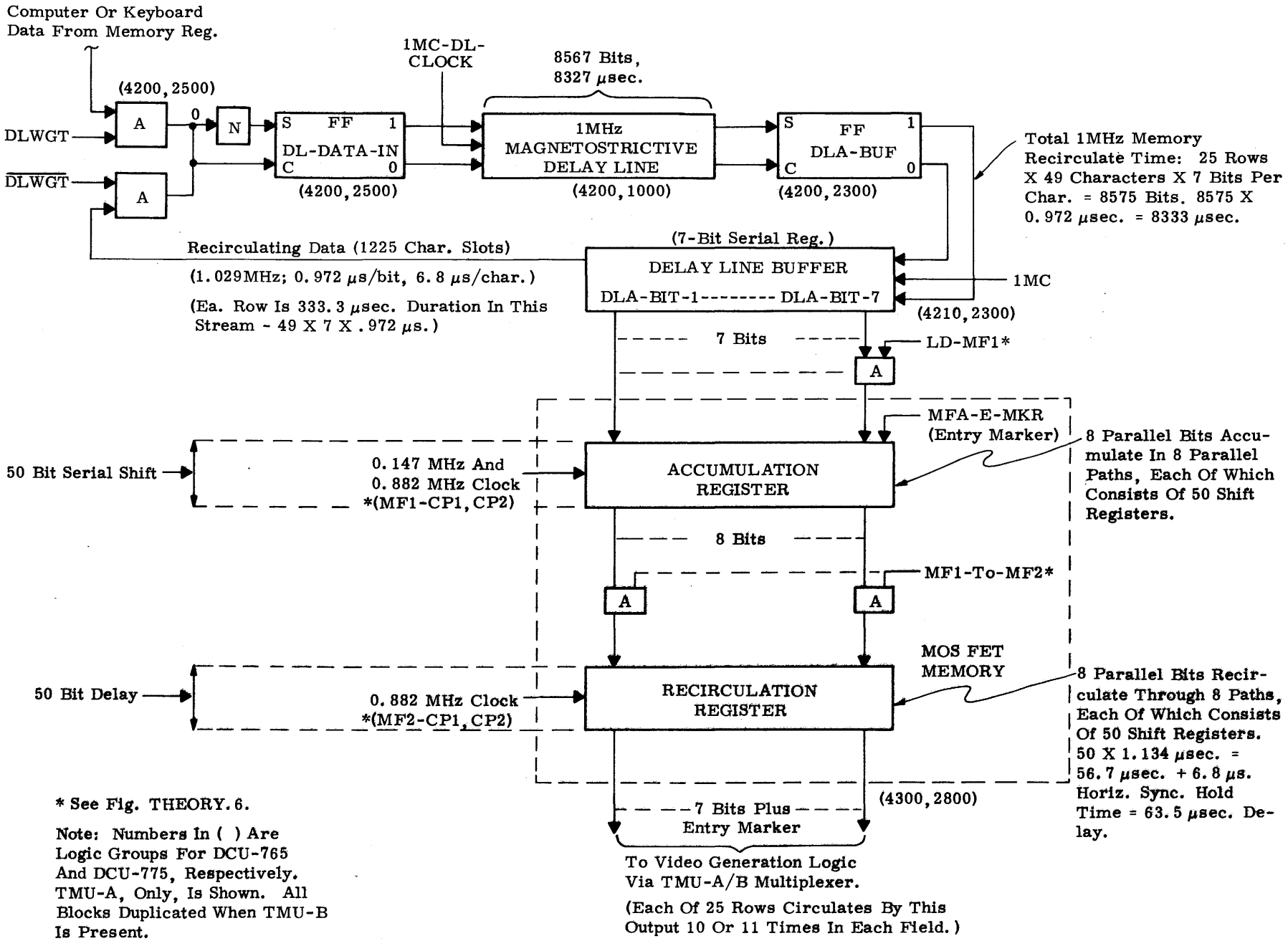


Fig. THEORY.5 Block Diagram, DCU Memory

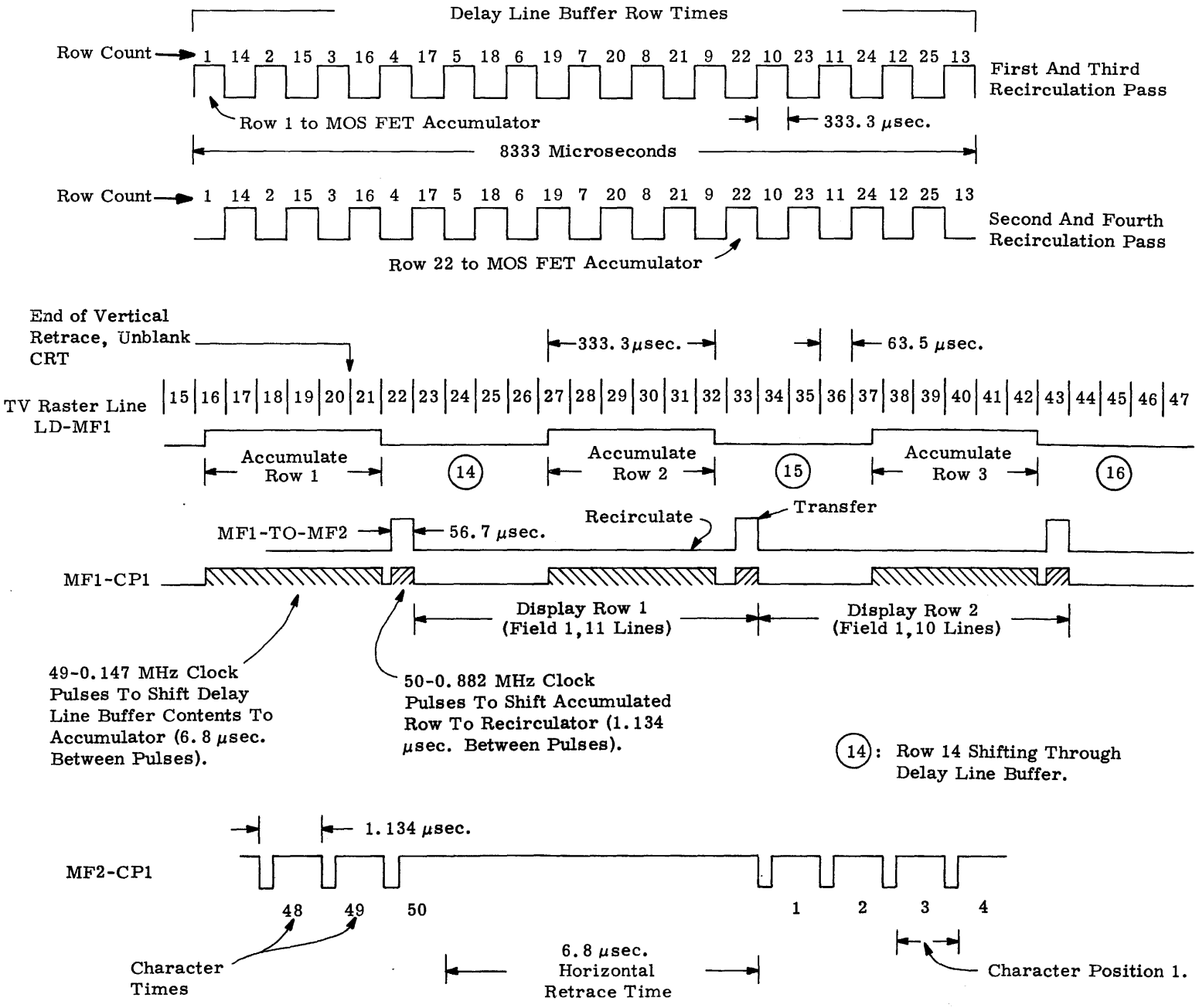


Fig. THEORY.6 Memory Timing

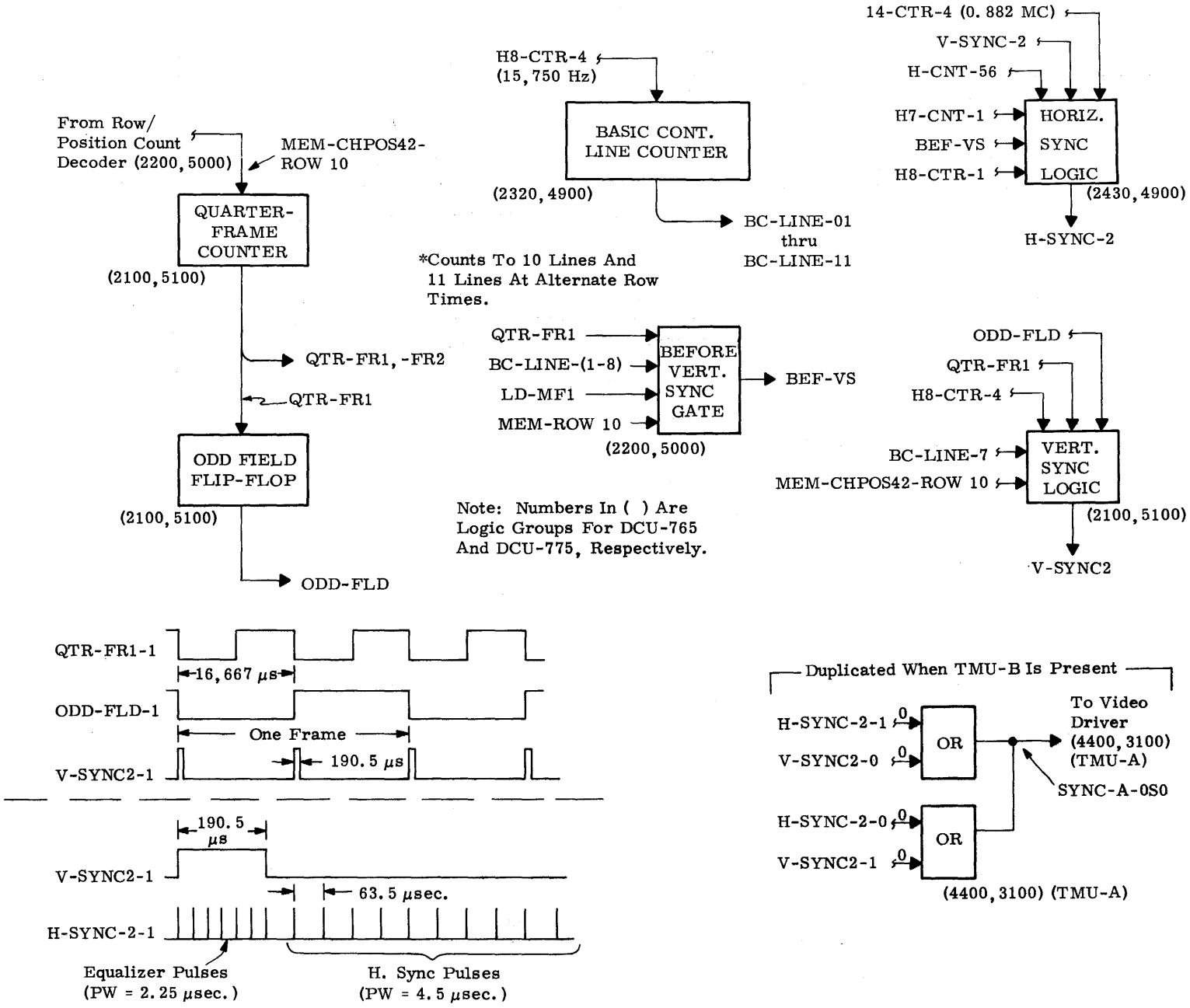


Fig. THEORY. 7 Display Timing

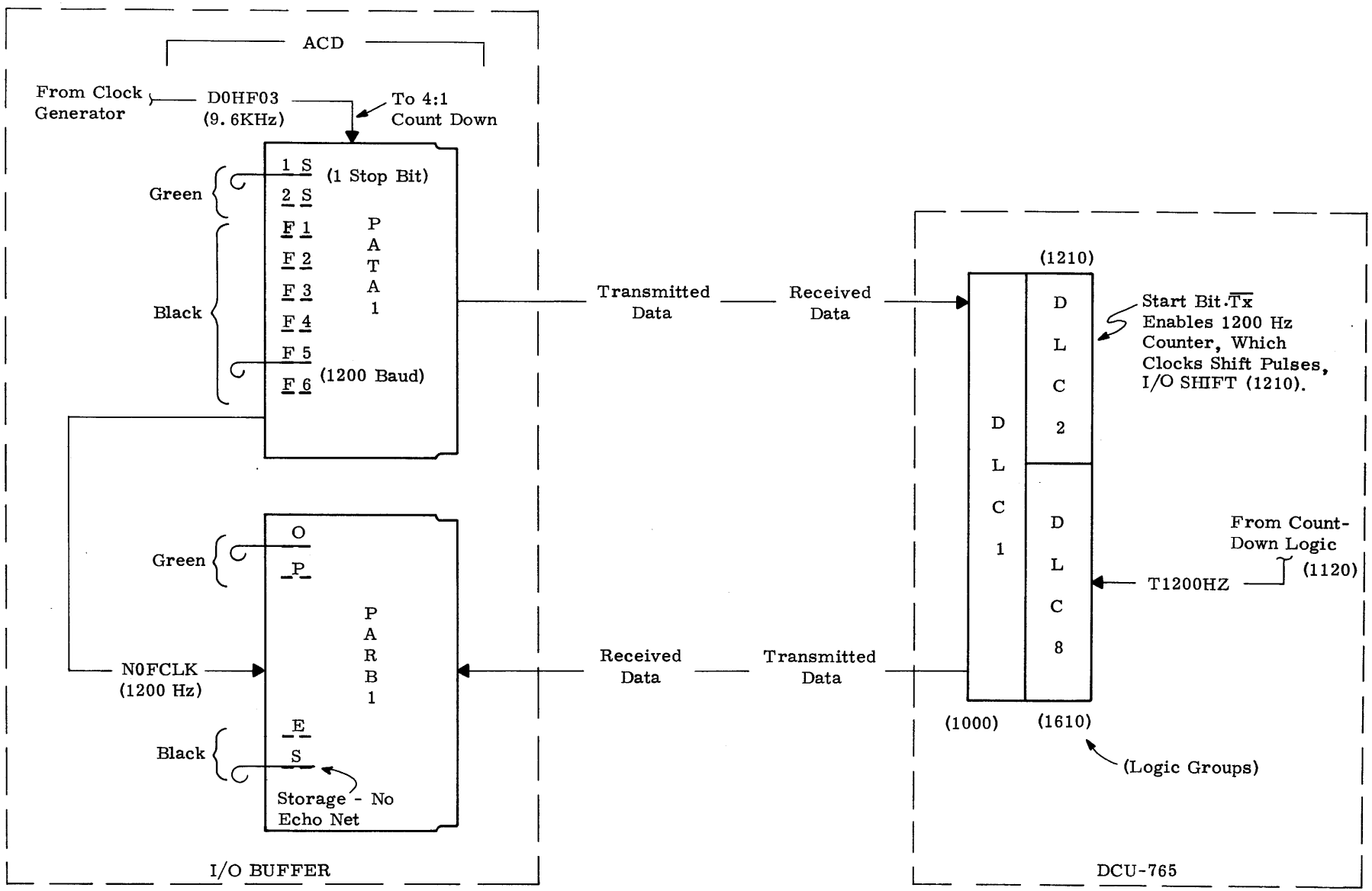


Fig. THEORY.8 Asynchronous Communications Timing

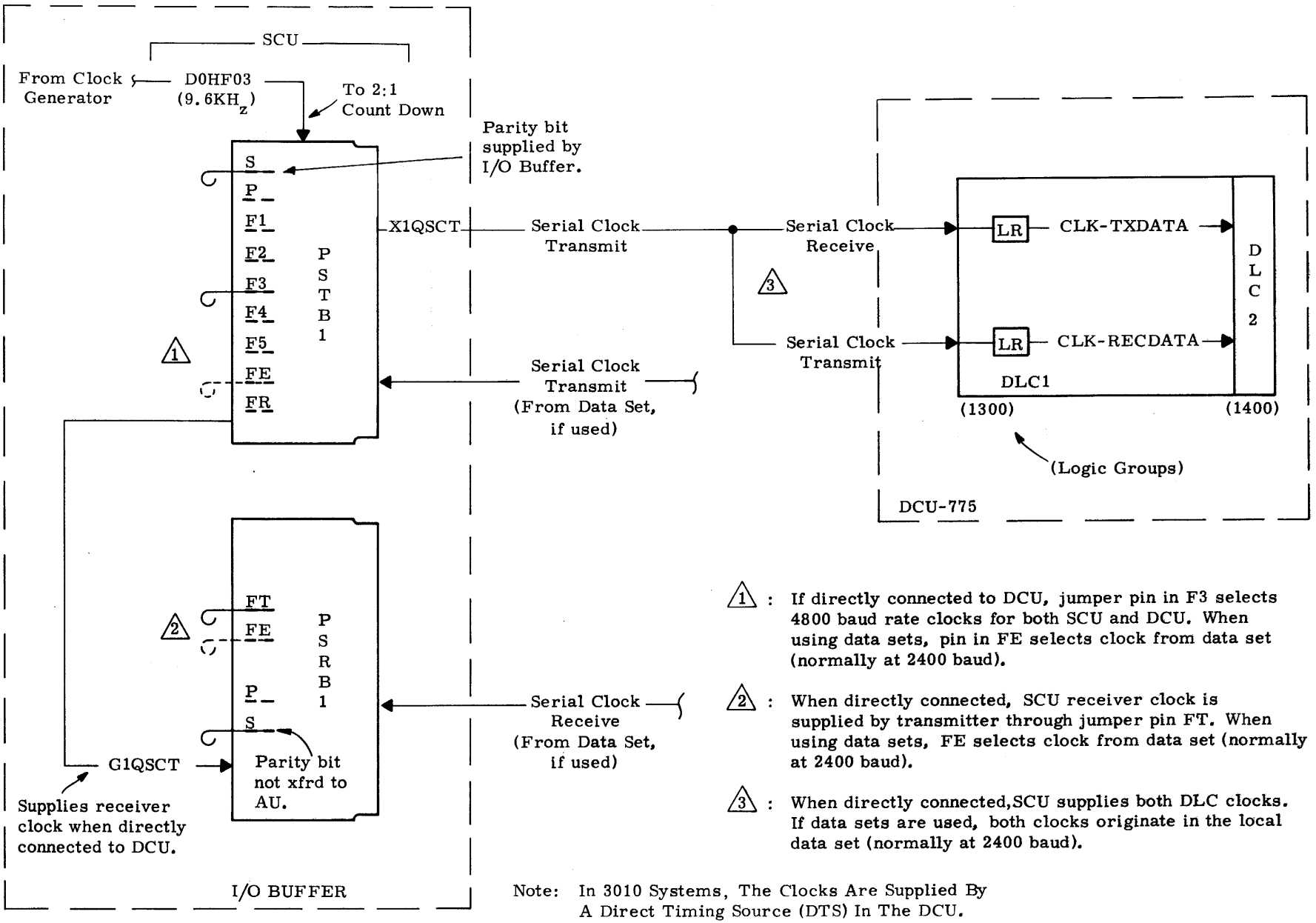


Fig. THEORY.9 Synchronous Communications Timing

Where data sets are employed, the data sets provide clock outputs which are in synchronism with the baud rate in use on the communication lines, both "serial clock transmit", which synchronizes the serial bit transmission at the local data terminal, and "serial clock receive", which synchronizes the receiver shift pulses.

When the SCU and Display Controller are directly connected by cable, the SCU transmitter supplies the baud rate clocks to both the DLC in the Display Controller and the SCU receiver. The Direct Timing Source option (DTS), which is available to supply clocks from the DCU, is not used, because of the availability of timing signals from the clock generator in the I/O Buffer.

VIDEO GENERATION

Video generation is accomplished by decoding each character, and the entry marker bit if present, as they are circulated by the output of the Recirculation Register in the MOS FET memory. At the appropriate line times, each video bit which is to be painted in the current character slot is loaded into a Video Register, and the contents of that register are shifted serially onto the video line to the Display Terminal. Fig. THEORY.10 is a block diagram of the logic involved in video generation.

The Character Generator Register and Upper Case Character Generator are time shared by TMU-A and TMU-B (if present). Since each character is held at the output of the MOS FET memories for 1.134 μs, there is sufficient time to gate one character from each TMU through the shared logic at alternate halves of the 1.134 μs period. Similarly, the outputs of the UC Character generator are gated back to the TMU's at alternate portions of the character period.

Output gate control signals, TMU-A-OGC and TMU-B-OGC, gate the characters into the multiplexer from TMU-A and TMU-B, respectively. The Character Generator Register holds each character while it is decoded by the UC Character Generator and the Video Bit Decoder. Each Video Bit Decoder in the two TMU's holds the same information, but their contents are gated into their respective Video Registers only when a character pertaining to the Display Terminal served by the Video Register is present. The Video Registers in TMU-A and TMU-B are loaded by 81 ns pulses LD-VRA and LD-VRB.

The upper case ASCII graphic symbols, which are represented by octal codes 041 through 132 (Table OP.1), are decoded by the UC Character Generator and gated in Video Bit Decoder, at appropriate character generator line times, to form the signals which set and reset appropriate Video Register bits, to form the parts of the character to be painted during the current line time and character slot. Other characters, such as the ETX symbol, the entry marker, the horizontal lines, vertical line, and corner marker, are not combined with the UC-LINE signals, but are loaded into the Video Register at the appropriate times by other logic in the character generator and the Video Bit Decoder.

Upper Case Character Generation

Fig. THEORY.11 illustrates the 21-raster-line X 7-video-bit character matrix which forms the graphic characters in each of the 1012 available character slots. The upper case characters are all formed by a 7 X 5 matrix within the larger matrix, during video bit times 1 through 5 and UC-LINE times 1 through 7, as indicated on the illustration. The illustration shows one example of the relationship of basic controller line times, to character generator line times, to UC-LINE times. The times alternate between the example shown and three other possibilities as dictated by the current field (odd or even) and the row count, which determine if ten or eleven raster lines are to be painted in the current field. In all cases, the UC-LINE count pulses occur when the CG-LINE count is higher by two (UC-LINE-07 = CG-LINE-09).

The letter "A" is used in the following example of video bit generation. Refer to Fig. THEORY.11 and note that during UC-LINE-02 in both the odd and even field, video bits 2 and 4 are to go true as the character slot is scanned.

1. The character is transferred into the Character Generator Register, setting TMU-BIT7 and TMU-BIT1, and resetting all other bits (101_8).
2. UC-A-0S0 (3100, 5200) goes true:

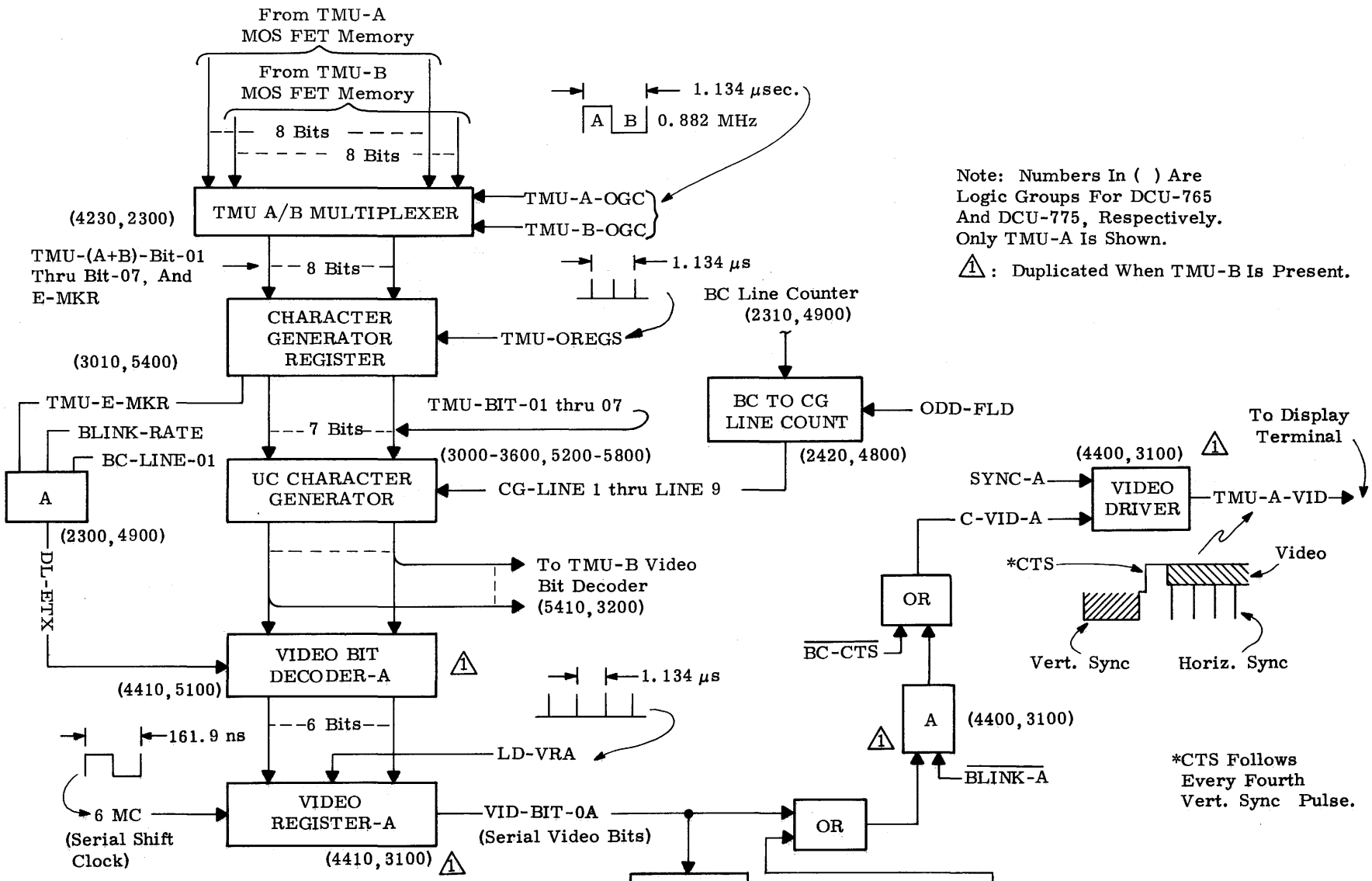
   $$\text{UC-A-0S0} = \text{UC-DLSD-01} \cdot \text{UC-DMSD-00}$$

   $$\text{UC-DLSD-01} = \overline{3} \cdot \overline{2} \cdot 1$$

   $$\text{UC-DMSD-00} = 7 \cdot \overline{6} \cdot \overline{5} \cdot \overline{4}$$

3. UC-QTS+A-0S0 goes true (3310, 5500), which makes the partial line-2 bit-2 signal, UC-PL2B2-1S0, true. During UC-LINE-02, this makes UC-PVB2B true (3620, 5800), and this makes DEC-VID-A-BIT-2A-1S0 true (4410, 5100).
4. UC-QTS+A-0S0 also makes the partial line-2 bit-4 signal, UC-PL2B4-1S0, go true (3310, 5500). During UC-LINE-02, this makes UC-LINE-(1+2+6)-B4-0S0 true (3600, 5800). This makes DEC-A-VID-BIT-4A-1S0 true (4410, 5100).
5. Just prior to the time the character slot is scanned during UC-LINE-02, LD-VRA-1S0 sets video bits 2 and 4 and resets all other Video Register stages. The next seven 6MCB clock pulses shift the contents of the register through VID-A-BIT-0A-1S0 onto the video line, making the CRT electron beam go bright during video bit times 2 and 4.

Analyses as in the preceding steps will indicate the process by which the appropriate Video Register bits are set prior to scanning through each character slot. Characters mentioned in the following paragraphs are afforded special treatment because they fill portions of the character matrix which are outside the 7 X 5 upper case matrix.



Note: Character Generator Register, UC Character Generator, And Video Bit Decoder Are Replaced By A Read-Only Memory In DCU-775 With Vendor Serial Numbers 1000 And Above. One ROM Is Provided For Each TMU - ROMA (5200) And ROMB (5300).

Fig. THEORY.10 Video Generation

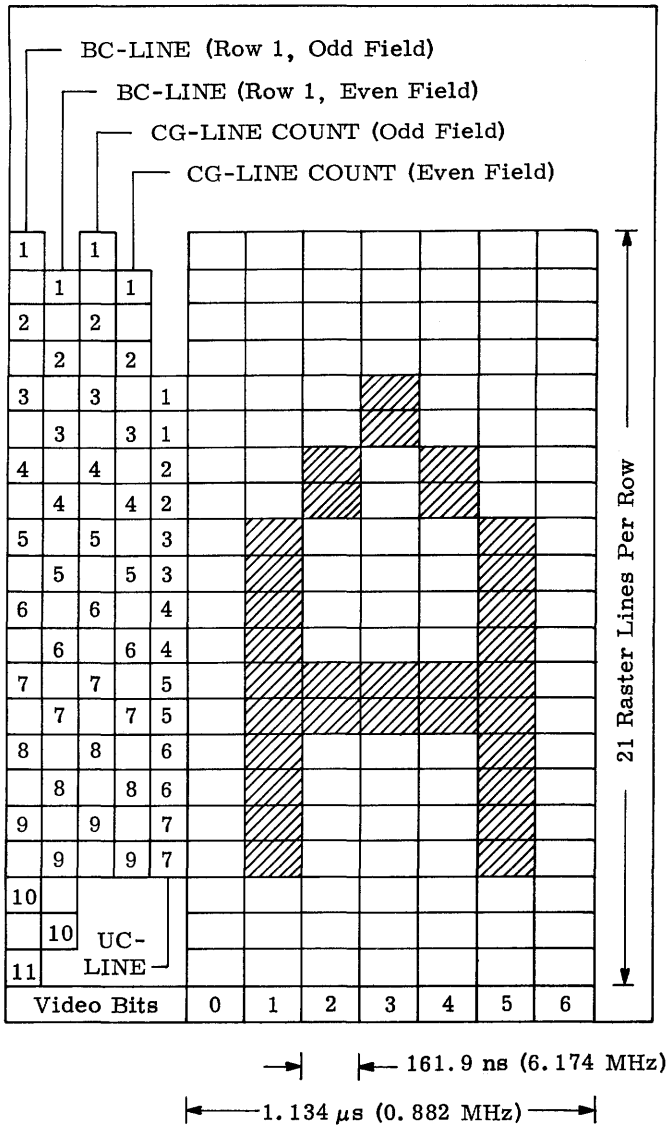


Fig. THEORY.11 Video Character Matrix

Entry Marker

The entry marker occupies the eighth bit of the MOS FET memory output and the multiplexer output. This bit will be set during the time the character corresponding to the current entry marker address is held at the remaining seven MOS FET outputs. During BC-LINE-01 time, TMU-E-MKR is combined with the BLINK-RATE signal to generate DL-ETX. The resultant DL-ETX pulse sets all Video Register bits at LD-VRA time, causing the entry marker to be painted across the top of the character slot. The entry marker is made to blink because it is not loaded into the Video Register when BLINK-RATE is false. (Video bit 2 is set by UC-PVB2A, which is enabled by DL-ETX when the entry marker is present.)

Blink Code

If a blink code (136_8) is decoded by the UC Character Generator, UC-BLK goes true. If BLINK-RATE is true at that time, UC-BLINK (2330, 4900) sets BLINK-A at LD-VRA time (4400, 5100). If set, BLINK-A will be cleared the next time a space is loaded into the Video Register. The BLINK-A signal blocks the serial video temporarily. This has the effect of making all graphic characters, from the position of the blink code in the memory stream to the next space or the end of a row, to blink.

Corner Marker

When a corner marker code (137_8) is decoded by the UC Character Generator, UC-CNR goes true (3100, 5200). This makes UC-CNR/LVN and UC-CNR/HORL both true (3400, 5600). UC-CNR/LVN sets video bit 0 at LD-VRA time, each time the corner marker code appears, thereby forming the vertical line at the left-hand edge of the character slot. UC-CNR/HORL produces a DL-ETX pulse during each BC-LINE-01, which sets all video bits, thereby producing the horizontal line across the top of the character slot, forming the remainder of the corner marker.

Horizontal And Vertical Lines

When the horizontal line code, 134_8, is detected, UC-CNR/HORL goes true. The left vertical line code, 133_8, makes UC-CMR/LVN true. The preceding paragraph describes the forming of these images in the character slot. In the case of the right vertical line code, 135_8, UC-RVRTLN goes true, which sets video bit 6 at each LD-VRA time, forming the right vertical line in the character slot.

ETX Symbol

When an ETX code, 003_8, is held by the Character Generator Register, DL-ETX goes true. This occurs when DL-ETX-100 is made true by one of its inputs which is derived from an and gate enabled by the ETX code.

$$\text{ETX} = \text{UC-DLSD-03} \cdot \overline{6} \cdot \overline{7} \cdot \overline{5}$$

In this case, the DL-ETX pulse duration is equal to the time the character is held by the Character Generator Register, and all video bits except VID-A-BIT-2A are set at each LD-VRA time. All bits and lines in the character slot except video bit 2 are set during each line time, forming the ETX symbol, ▮▮.

Composite Video Generation

The serial video output from the Video Register is combined with the keyboard clear-to-send pulses (BC-CTS) and the horizontal and vertical sync pulses as shown on Fig. THEORY.10. The CTS pulses and video rise in a positive direction from the baseline and the sync pulses are negative. The composite

video is fed to the monitor, via the keyboard, if present, as shown on Fig. THEORY. 1. The keyboard strips the CTS pulses from the composite video and returns the sync pulses and video to the monitor as display video. If no keyboard is present, a modification must be made to the Display Controller to remove the CTS pulse from the composite video waveform. The modification is described in the Computer Maintenance manual under Video Displays.

ENTRY MARKER CONTROL

The current entry marker address is maintained in two 7-bit serial-shift registers, the Entry Marker Position Register and the Entry Marker Row Register. Two similar registers maintain a count equal to the row and character position number of the character currently held by the Delay Line Buffer in the 1 MHz delay line memory loop. When the memory row and position count equals the entry marker row and position count, an entry marker compare pulse is generated, which enables either the reading of a character from the 1 MHz memory or the writing of a character into that memory. The compare pulse also causes MFA--E-MKR (4200, 2300) to load the entry marker into the eighth path through the MOS FET memory for display on the monitor.

The row and position counters and the two entry marker registers are preset to zero when the Display Controller is initialized during the power-on sequence. If no characters from the computer or the keyboard are fed into the Memory Register, the two entry marker registers will retain this preset count, which corresponds to row one, position one, and the entry marker will appear stationary at that position on the monitor. Fig. THEORY. 12 is a simplified diagram of the logic involved in entry marker control.

The position counter continuously cycles through character position counts 1 to 49. Each time the count equals 49 (after counting 0 - 61_8), the UPDATE-ROW-CNT1 flip-flop produces a 6.8 μs pulse, which updates the row counter. An ADDCOUNT signal, generated by the row and position decoder logic, determines whether the row count is to be updated by adding 13 or by subtracting 12. ADDCOUNT alternates with each UPDATE-ROW-CNT1 pulse, so the row count proceeds on the stagged row count sequence shown on Fig. THEORY. 6.

Because the entry marker indicates to the operator the position on the display where the next character may be entered, or the first text character in the next message from his Display Terminal, the entry marker address contained in the entry marker row and position registers must be updated as characters are entered on the display by the operator or the computer, and in response to entry marker control characters from either source. The entry marker address is also updated as text characters are read from the delay line memory for transmission to the computer.

As shown on Fig. THEORY. 1, each Memory Register may contain characters which originated in the computer or at the keyboard. These characters may be graphic characters which have just been read from the delay line memory or are waiting to be written into the delay line memory. If they are not graphic characters, they are control characters. Graphic characters are those with an octal code from 040 to 137, as shown on Table OP. 1. If the binary bits are assigned positions 7 through 1, each graphic character will have one, and only one, of bits 6 and 7 set. All other characters appearing on the table have neither of bits 6 and 7 set, or both bits 6 and 7 are set. Note that control characters from the keyboard or the computer are not written in the delay line memory but are held by the Memory Register while the entry marker control is effected.

When the control characters appear in a Memory Register, they are gated into a Command Decoder which is shared by the two Memory Registers. When either TMU detects a control character in its Memory Register, it requests the use of the entry marker control logic. If the entry marker and command decoder logic is not in use by the opposite TMU, access will be granted, and MRA-CONT from the requesting TMU goes true, allowing the control character to pass into the Command Decoder.

In addition to the Command Decoder, the TMU's also share the Adder/Subtracter control logic and the Adder/Subtracter. Since the character position and row counts in the two TMU's are identical, the position and row counters serve both. Of course, each TMU and Display Terminal must have independent entry markers, so each TMU must have its own entry marker position and row registers, and its own serial entry marker comparator.

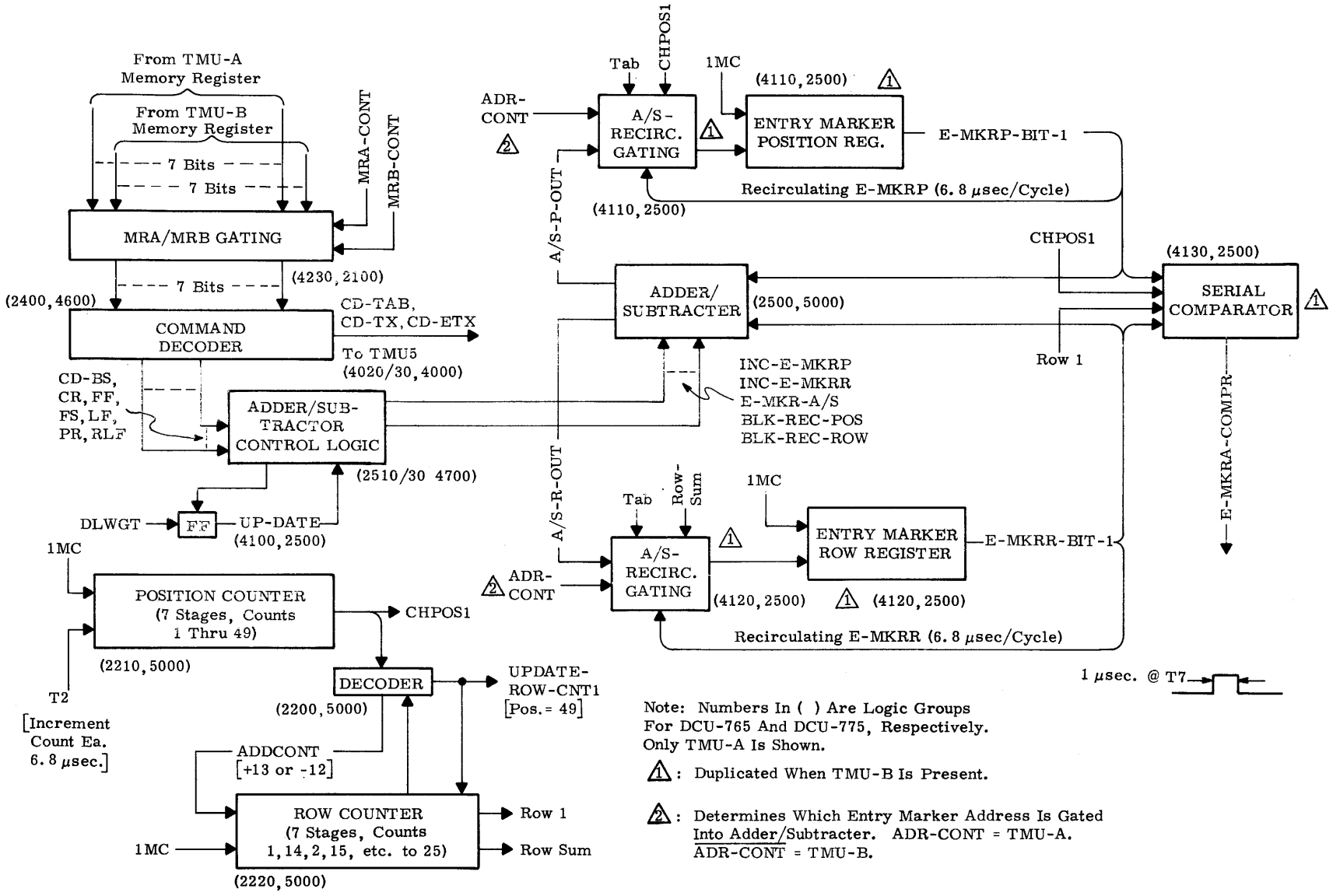
The Adder/Subtracter and control logic monitor the contents of the entry marker registers and automatically change their contents as required to sequence the entry marker through the display area, e. g., when the Entry Marker Position register acquires a count of 47 or more, E-MKR>POS-MAX goes true (2520, 4700), zeros are shifted into the Entry Marker Position Register, and the entry marker row count is increased by one. This has the effect of generating an automatic line feed, as seen on the display, when the marker is incremented beyond the last character position on a row.

If the Entry Marker Row register acquires a count of 23 or more, E-MKR>ROW-MAX (2520, 4700) goes true, and zeros are shifted into both entry marker registers, returning the marker to character position one, at the top of the page.

The following paragraphs describe the various conditions under which the entry marker registers and entry marker positions on the displays are updated. Detailed descriptions of the logic involved are provided in the BCU/TMU flow charts in Volume 3F of the vendor manuals.

Transmit Mode

The transmit mode is entered when the computer carrier has dropped following the transmission of an



Note: Numbers In ( ) Are Logic Groups For DCU-765 And DCU-775, Respectively. Only TMU-A Is Shown.

1 : Duplicated When TMU-B Is Present.

2 : Determines Which Entry Marker Address Is Gated Into Adder/Subtracter. ADR-CONT = TMU-A. $\overline{\text{ADR-CONT}}$ = TMU-B.

Fig. THEORY. 12 Entry Marker Control

EOT character at the end of the last computer message. If EOT was not received, prior to the computer's carrier drop, transmit mode is entered after the computer's carrier has been off for four seconds. These conditions are described in the flow chart on Fig. OP.7.

A text message transmission request may be present due to the detection of a TX code by the Command Decoder, which makes CD-TX true, or due to the receipt of a negative acknowledge (NAK) message from the computer. If text is to be transmitted BUF1-ETX in DCU-765 or STOETX in DCU-775 will be reset, and DLRGT-EN (1700, 3500) and the next entry marker compare will set the delay line read gate DLRGT (4130, 2500), which remains set while one text character is read out of the 1 MHz Delay Line Buffer.

When DLRGT resets, an UP-DATE pulse is generated (4140, 2500) which sets INC-E-MKRP and resets E-MKR-A/S (2510, 4700). The contents of the appropriate entry marker position and row registers are shifted through the Adder/Subtracter, the entry marker address is incremented by one, and the address is shifted back to the entry marker registers. Each time DLRGT sets and resets, the entry marker is incremented by one, and this process continues until BUF1-ETX (DCU-765) or STOETX (DCU-775) sets, indicating the end of text transmission.

Message Reception

Text characters received from the computer are gated into the Memory Register in the TMU addressed by the computer. If the characters are not control characters, they are shifted into the 1 MHz delay line at each entry marker compare, which produces a DLWGT pulse (4130, 2500). Each DLWGT causes an UP-DATE pulse to be produced, and that sets INC-E-MKRP and resets E-MKR-A/S (2510, 4700). The contents of the appropriate entry marker position and row registers are shifted through the Adder/Subtracter, the entry marker address is incremented by one, and the address is shifted back into the entry marker registers. Each time DLWGT sets and resets, the entry marker is incremented by one, and this process continues until the DLC has finished shifting characters into the Memory Register.

Back Space

When back space (010_8) appears in the Command Decoder, CD-BS goes true, which sets INC-EMKRP and resets E-MKR-A/S (2510, 4700). The entry marker address from the appropriate TMU is shifted through the Adder/Subtracter, the position count is decremented by one, and the new address is shifted back to the entry marker registers.

Back space characters addressed to a TMU in DCU-775 at 4800 baud must be followed by at least four fill characters to allow time for the entry marker to appear at its new position at the next entry marker compare. The fill characters may be either DEL (177_8) or successive back spaces. No fill characters are required at any other baud rates in use with GE-PAC computers.

Forward Space

When forward space (022_8) appears in the Command Decoder, CD-FS goes true, which sets INC-EMKRP and E-MKR-A/S. The entry marker address from the appropriate TMU is shifted through the Adder/Subtracter, the position count is incremented by one, and the new address is shifted back to the entry marker registers.

No fill characters are required following forward space.

Line Feed

When line feed (012_8) appears in the Command Decoder, CD-LF goes true, which sets INC-E-MKRR and E-MKR-A/S. The entry marker address from the appropriate TMU is shifted through the Adder/Subtracter, the row count is incremented by one, and the new address is shifted back to the entry marker registers.

No fill characters are required following line feed.

Reverse Line Feed

When reverse line feed (021_8) appears in the Command Decoder, CD-RLF goes true, which sets INC-EMKR and resets E-MKR-A/S. The entry marker address from the appropriate TMU is shifted through the Adder/Subtracter, the row count is decremented by one, and the new address is shifted back to the entry marker registers.

Reverse line feed characters addressed to a TMU in DCU-775 at 4800 baud must be followed by at least four fill characters to allow time for the entry marker to appear at its new position at the next entry marker compare. The fill characters may be either DEL (177_8) or successive reverse line feeds. No fill characters are required at any other baud rates used with GE-PAC computers.

Line Return

When line return (015_8) appears in the Command Decoder, CD-CR goes true which sets BLK-REC-POS (2530, 4700). The entry marker address from the appropriate TMU is shifted through the Adder/Subtracter, the position count recirculation is blocked which returns the position count to zero, and the new address is shifted back to the entry marker registers. An entry marker position count of zero corresponds to position one, so the entry marker appears at position one of the same line, at the next compare.

Line return characters addressed to a TMU in DCU-775 at 4800 baud must be followed by at least four DEL characters (177_8) to allow time for the entry

marker to appear at its new position at the next compare. No DEL's are required at any other baud rate used with GE-PAC computers.

Page Return

When page return (024_8) appears in the Command Decoder, CD-PR goes true, which sets BLK-REC-POS and BLK-REC-ROW (2530, 4700). The entry marker address from the appropriate TMU is shifted through the Adder/Subtracter, the position and row count recirculation is blocked, setting both counts to zero, and the new address is shifted back to the entry marker registers. Since an entry marker row count of zero and position count of zero correspond to row one, position one, the entry marker appears in that position at the next compare.

Page return addressed to a TMU in DCU-775 at 4800 baud must be followed by at least four DEL characters (177_8) to allow time for the entry marker to appear at its new position at the next compare. No DEL's are required at any other baud rate used with GE-PAC computers.

Form Feed

When form feed (014_8) appears in the Command Decoder, CD-FF goes true, which sets BLC-REC-ROW, BLK-REC-POS, and ERASE-A (or -B) (4130, 2500). The entry marker address from the appropriate TMU is shifted through the Adder/Subtracter, the position and row count recirculation is blocked, setting both counts to zero, and the new address is shifted back to the entry marker registers. Since an entry marker row count of zero and position count of zero correspond to row one, position one, the entry marker appears at that position at the next compare.

At the time of the compare which places the entry marker in its new position, ERS-WGT-A sets (4100, 2500), and blocks the recirculation of 1 MHz delay line. At the next compare time, spaces are written into the delay line by each T6 pulse, which sets bit 6 of each character position to load 040_8 into the entire 1 MHz memory. Finally, at the third compare time, ERS-WGT-A is reset. The display is now blank, except for the blinking entry marker in row one, position one.

Form feed addressed to a TMU in DCU-775 at 4800 baud must be followed by at least 9 DEL characters (177_8), to allow time for the execution of the form feed. No DEL's are required at any other baud rate used with GE-PAC computers.

Tab

When tab (011_8) appears in the Command Decoder, CD-TAB goes true, setting PTAB (4020, 2400). A sample tab flip-flop, SAM-TAB (4200, 2400), is continuously toggled at each row count update time, and therefore, is set during each of the 25 delay line row times, as the 1 MHz delay loop recirculates twice (see Fig. THEORY. 6). At the next entry marker compare which finds SAM-TAB set, TAB-A (or -B) sets. When TAB-A and SAM-TAB are set, the characters circulating through the delay line buffer are decoded to determine if the code for a vertical line is present. As shown on Table OP. 1, the codes which produce a vertical line are 133_8, 134_8, and 137_8. At the time any of these codes is detected, TAB-E-MKR-WGT sets (4100, 2300) and TAB-A is reset.

TAB-E-MKR-WGT blocks recirculation of the entry marker registers for the TMU involved, and allows the current row and position counter contents to be shifted into the entry marker registers. At the next compare, the entry marker will appear in the character position following the one which contained a vertical line. If no vertical line is found, TAB-E-MKR-WGT cannot set, and the entry marker will not move.

When transmitted by the computer, tab must be followed by DEL characters (177_8) at the higher baud rates, to allow time for the vertical line search (16 2/3 ms, max.). At 1200 baud, no DEL's are required. Tab must be followed by 1 DEL at 2000 baud, 3 at 2400 baud, and 14 at 4800 baud.

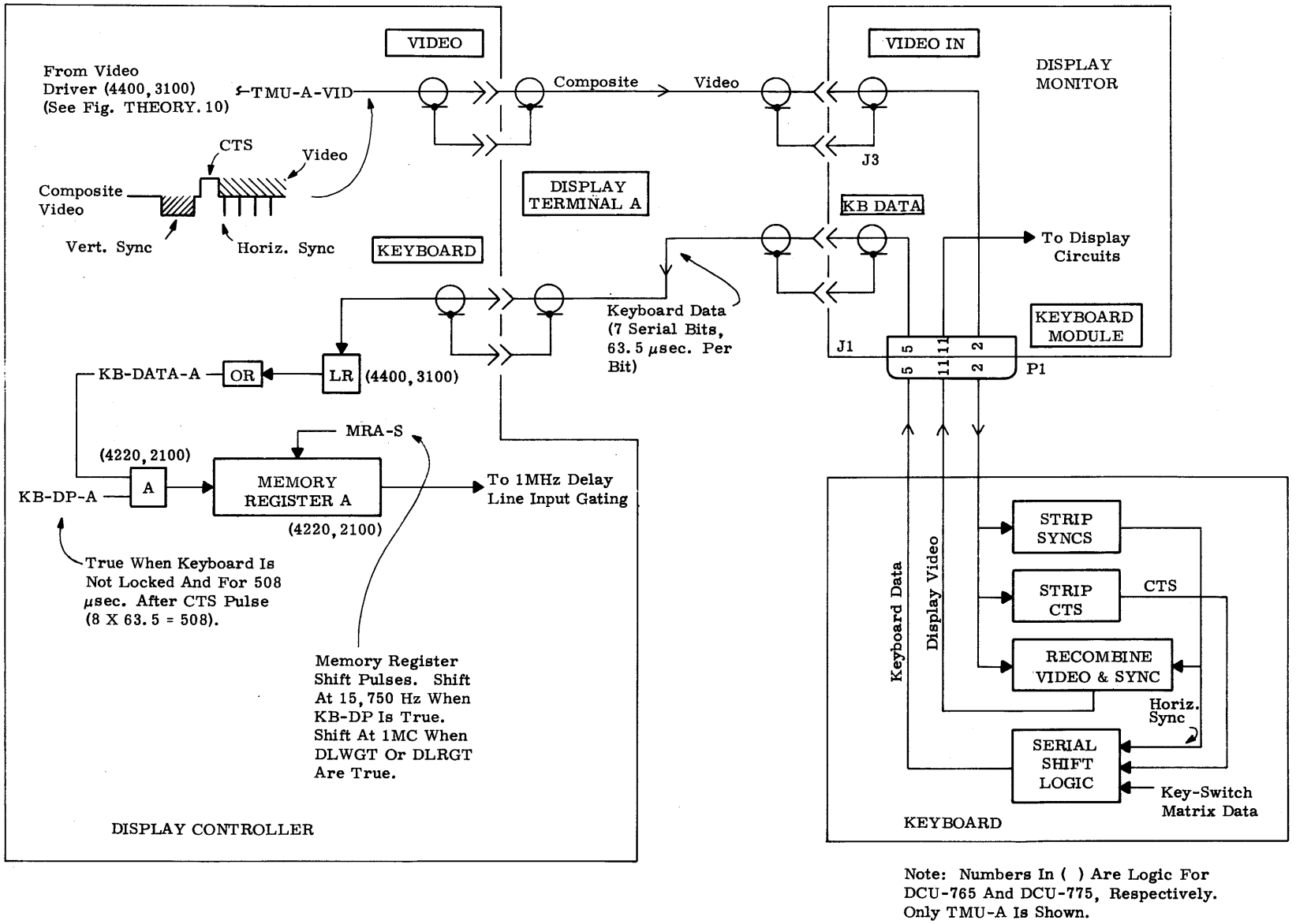
KEYBOARD INPUTS

When a keyboard key is pushed, a single pair of contacts generates a seven bit graphic character in a key-switch matrix, and the character is stored in a holding register in the keyboard, until the character is allowed to enter the Display Controller by a clear to send (CTS) pulse in the composite video signal. Fig. THEORY. 13 is a simplified diagram of the logic involved in transferring characters from a keyboard to the Memory Register in the Display Controller.

If a graphic character from a keyboard is held in the Memory Register, it is shifted into the 1 MHz delay line memory at the next entry marker compare time, which will cause the character to appear at the entry marker position, and the entry marker to be incremented one position. If the character is determined by the Command Decoder to be a control character, the DCU responds appropriately to the character.

The CTS pulses are added to the composite video waveform immediately following every fourth vertical sync pulse, or every 66. 67 ms. The 63. 5 μs pulses are generated in the display timing area of the Display Controller (2100, 5100) and are added to the composite video by the video output logic in each TMU (see Fig. THEORY. 10). The composite video waveform is transferred to the keyboard on coaxial cable, as indicated on Fig. THEORY. 13.

The keyboard logic strips the horizontal sync, vertical sync, and CTS pulses from the composite video waveform, and recombines the horizontal and vertical sync pulses with the video for return to the Display Monitor as display video. The CTS pulses are used to enable the transfer of the contents of the keyboard holding register to a shift register, and the horizontal sync pulses are used as shift register clocks to shift the key characters out on a coaxial cable, as 7-bit serial characters.



Note: Numbers In ( ) Are Logic For DCU-765 And DCU-775, Respectively. Only TMU-A Is Shown.

Fig. THEORY. 13 Keyboard Inputs

When a serial keyboard character appears at the input to the Memory Register in the TMU serving the Display Terminal containing the keyboard, and if keyboard data is not locked out (DLC is not busy receiving or transmitting a message), the character is shifted into the Memory Register.

If the Memory Register logic determines that the character is a control character (MRA-CONT, 4030, 4000), access to the Command Decoder and the entry marker Adder/Subtracter logic is requested (see Fig. THEORY. 12). If the character is not a control character, it is written into the 1 MHz delay line at the next entry marker compare, as DLWGT goes true (4200, 2500).

Function Codes

Two function code character slots are included in the header portion of all messages from Display Controllers to the computer (see Fig. OP. 6). In DCU-765, the DLC always loads spaces into the FC1 and FC2 character slots. In DCU-775, the contents of the two function code registers in the TMU originating a message to the computer are loaded into the function code slots.

The office type keyboard, EKB-761, may incorporate an optional Function Key Group (FKG) which provides keys which may be used to load the function code registers in DCU-775 with operator-determined function codes, and automatically initiate a message transmission. The operator oriented keyboards, EKB-764/765 incorporate Action Keys and Special Control keys which automatically load the function code registers with predetermined function codes (see Fig. OP. 2).

The function code registers (4400/4500) are loaded with the contents of the memory register, which are serially shifted into FCR1 or FCR2 when CD-FN-CODE from the Command Decoder goes true, indicating that the Memory Register is holding a function code. CD-FN-CODE identifies function codes by determining that the Memory Register holds a code from 140_8 through 147_8, or from 161_8 through 167_8 (see Table OP. 1). As the keyboard characters are shifted into the registers, they are transposed to the corresponding graphic character codes.

The function code register which will be loaded is determined by two flip-flops in each TMU, FCA-x-NEXT and FCB-x-NEXT (4400/4500). If both of these flip-flops are reset, FCA-x-NEXT sets on receipt of the first function code from the keyboard. If an EKB-761 keyboard is connected, the FCA-x-NEXT-002 input on the set side of the FCB flip-flop is grounded, prohibiting its setting and maintaining a space code in the FCR2 function code register.

When an EKB-764/765 keyboard is connected to the TMU, the FCA-NEXT-002 input to the FCB flip-flop is not grounded but is derived from the LD-x-FCA term, which is true when FCA-x-NEXT is set (see block 17.0 on logic sheet 128). In this case, FCA-x-NEXT sets at the first function code input and loads FCR1, and FCB-x-NEXT sets at the second function code input and loads FCR2.

When the DLC is ready to transmit the contents of FCR1 or FCR2, it turns FCA-x-OUT or FCB-x-OUT true, which enables the 1 MHz I/O shift pulses on the function code register stages, and shifts the contents into the appropriate header character slots via the I/O Buffers and the I/O Register, in the DLC.

When the computer replies to a message with a reply other than a negative acknowledge (NAK) reply, the function code registers are cleared (loaded with spaces). This is accomplished by AFCLR (4400) or BFCLR (4500), from the DLC.

Transmission Requests From EKB-761

When the office style keyboard, EKB-761, is connected to DCU-765, only the TX key can initiate a transmission request. When an EKB-761 with the Function Key Group option is connected to a TMU in DCU-775, the TX key and each of the function code keys initiate transmission requests.

All transmission requests set KBxTX (4020, 4000) which locks keyboard data out until the transmission is complete (DTxKBLOCK), and sets DISMKR which causes the video generator to produce a double entry marker until the computer replies to the transmission. Also, KBxTX informs the DLC that this terminal has a text message ready to go the next time that SERVDT1 or SERVDT2 goes true (1730, 3600), indicating that the DLC is ready to service the transmission request.

In the case of a request from the TX key, KBxTX is set when CD-TX in the Command Decoder detects the character. In DCU-775, CD-TX clears the function code register when an EKB-761 is connected, causing spaces to be sent in the FC1 and FC2 slots (FCR-x-CLR-100, 4400/4500).

When the request is initiated by a function key (DCU-775, only), first FCR1 is loaded with the function code, then, because a function code has been detected, REM-REL goes true (4400/4500), turning CD-ETX true, setting WRT-ETX, which writes an ETX symbol in the current entry marker position the next time a compare occurs and DLWGT goes true. Next, because REM-REL is true, when DLWGT goes false and UP-DATE goes true, TX-PR goes true, making CD-PR true (4700/4800), which causes the entry marker control logic return the entry marker to row one, position one. TX-PR also sets KBxTX, initiating the transmission request.

When the DLC services a transmission request initiated by a function key, the message text starts at the beginning of the displayed page and continues until the EXT symbol is encountered, at the position where the entry marker was, at the time the key was pushed.

Transmission Requests From EKB-764/765

The operator oriented keyboards can be connected to TMU's in DCU-775/776 or DCU-785/786. The TX key (if present) can initiate transmission requests, and the Action Keys and Special Control Keys initiate transmission requests, after loading the function code registers and positioning the entry marker.

All transmission requests set KBxTX (4000/4100) which locks the keyboard data out until the transmission is complete (DTxKBLOCK). KBxTX also sets DISMKR which causes the video generator to produce a double entry marker until the computer replies to the transmission. Also, KBxTX informs the DLC that this terminal has a text message ready to go the next time that SERVDT1 or SERVDT2 goes true (3600), indicating that the DLC is ready to service the transmission request.

In the case of a request from the TX key, KBxTX is set when CD-TX in the Command Decoder detects the character. In virtually all cases, the function code registers will contain spaces, because they were cleared by the last non-NAK message from the computer.

Each of the Action and Special Control Keys, when pushed, causes a group of characters to be transferred to the DCU. While the DCU's response to each key depression is similar, it is useful to bear in mind the code sequence generated by each key depression, when determining the sequence of events. Each single key depression causes the following sequence of characters to be transferred to the DCU at each CTS time (each key depression causes 5 or 6 characters to be shifted out):

- Action Keys (see Fig. OP. 2);

  Bottom row - FC(CLR), FC1, PR, ETX, TX.

  Middle row - FC(CLR), FC1, FC(A), PR, ETX, TX.

  Top row - FC(CLR), FC1, FC(B), PR, ETX, TX.
- SGL LINE or EDIT - FC(CLR), FC(B), FC(1), LR, ETX, TX.
- PTL LINE or POINT SELECT - FC(CLR), FC(A), FC(1), ETX, TX.

The DCU-775 responds to the Action and Special Control Key codes in virtually the same manner as if they were generated by individual entry marker or control keys, but since these keys are used to generate automatic transmissions, with the entry marker positioning, ETX entry, and transmit request all generated by a single key depression, the controller goes into a single-line mode when it detects an input from one of these keys.

Text transmission in response to an Action Key depression starts at row one, position one, and continues to the end of row one or to an ETX symbol if there is one on the row. SGL LINE or EDIT keys call for text transmission from position one of the row containing the entry marker to the end of the row or to an ETX symbol if there is one on the row. PTL LINE or POINT keys call for text transmission from the current entry marker position to the end of the row or to an ETX symbol if there is one on the row. Note that in each case, a single row or less is transmitted, hence the name, "single-line mode".

The Display Controller responds to the Action and Special Control Keys as follows:

1. The FC (clear) character (140_8) is loaded into both function code registers and sets SGL-A (4200) or SGL-B (4300).

   SGL-A-1S0 = CD-F-CLR· MRAB-CONT· 764

   The term, 764, is true when the back panel keyboard option patch plug has selected operation with EKB-764/765 (see the Video Display section in the Computer Maintenance manual).

   SGL-A-101 places the terminal in local mode by setting DT1LOC (4200, 4300). It will remain in local mode until KBxTX is set, to prevent computer messages from moving the entry marker or resetting FCR1 or FCR2 while the single-line mode message is being set up.

2. The next two function code characters load appropriate codes into FCR1 and FCR2. In the case of the bottom row of Action Keys, only FC1 is entered, FC2 remains clear (contains 040_8, space).

3. The next character received from the keyboard is either an entry marker control character or ETX. If it is ETX, go to step 4. If not, the entry marker control logic responds to PR or LR.

4. The next character received is ETX. CD-ETX does not cause an ETX symbol to be written on the display because it is blocked from the set side of the WRT-ETX flip-flop (4000, 4100) by SGL-A (or -B). CD-ETX does set SGL-x-TX (4200, 4300). SGL-x-TX disables the space deletion feature (SP-DETD, 2300, 2400), and partially enables TX-ENDPG (2500, 2600).

5. The next character received from the keyboard is TX. CD-TX sets KBxTX. When the DLC services the request, FC1 and FC2 in the message header contain the contents of the function code registers. Text begins at the entry marker position determined in step 3 and continues until an ETX symbol is read out of the delay line memory or until the end of the current row is encountered, which makes TCR/LF true (4600). TX-ENDPG, which was partially enabled in step 4, is now fully enabled, setting DDLRGT which disables the delay line read gate, stopping text transmission and causing the DLC to send ETX to the computer.

DATA LINE CONTROLLERS

The DLC's provide the interface with the computer or a data set, respond to the message control characters in the header of messages from the computer, transfer the text portion of received messages to the addressed TMU, generate messages transmitted to the computer, and control the sequencing of transmitted messages. The overall operation of the DLC's is depicted by the message sequencing flow-chart on Fig. OP. 7.

The logic in the DLC in DCU-765 and the DLC in DCU-775 is similar in the areas related to message sequencing, longitudinal parity and lateral parity checking and generation, header formation, command decoding, and the EIA-RS-232C interface to the computer and data set. Because the DLC in DCU-765 receives and transmits bit-serial asynchronous characters at 1200 baud, while the DLC in DCU-775 receives bit-serial synchronous characters at baud rates as high as 4800 bits per second, the message reception and transmission logic differs greatly.

Detailed DLC flow charts are provided in Volume 3F of the vendor manuals. The flow charts are especially useful in locating the logic involved in message sequencing and control character responses. The remainder of this discussion is intended to provide an understanding of the principals of operation and the data flow through the two DLC types, so that the logic and flow charts in the vendor manuals will be more meaningful. A review of the discussion on message sequencing in the Operation section of this publication will also be useful in understanding the DLC operation.

Both DLC's use a set of program count or "P" count flip-flops to control message sequencing. Table THEORY. 1 lists the "P" counts and indicates what each DLC does at each "P" count time.

Asynchronous DLC

A block diagram of the asynchronous DLC used in DCU-765 is provided on Fig. THEORY. 14. The I/O Register and the Buffer 1 Register handle serialized message characters during both transmission and reception. The I/O Register accumulates received characters at 1200 bits per second and shifts out transmitted characters at 1200 bits per second. Exchanges between the I/O Register and the Buffer are accomplished by serial shifting at 1 MHz. The Buffer, therefore, provides the transition from communications line timing to memory timing.

During text transmission, one character is read out of the 1 MHz delay line every 8-1/3 ms. During text reception, one character is stored in the 1 MHz delay line every 8-1/3 ms. Since the transmission or reception of each 10-bit character (Fig. OP. 4) at 1200 baud requires at least 8-1/3 ms plus the time required to shift in or out of the memory register, the delay line loop will always have time to recirculate once before each character is read or written.

During text reception, the data is directed to the appropriate Memory Register by REC-TMU-A or REC-TMU-B. These terms are enabled when the address character is received in the message header.

During text transmission, MR-TX-DATA is shifted into Buffer 1 from the appropriate TMU, as determined by TX-TMU-A or TX-TMU-B.

Synchronous DLC

Fig. THEORY. 15 is a block diagram of the DLC in DCU-775. The I/O Register and Buffers 1 through 5 handle the serialized message characters during both transmission and reception. The I/O Register accumulates received characters at up to 4800 bits per second and shifts out transmitted characters at up to 4800 bits per second. The five buffers are required to store received characters waiting to enter the delay line memory and to accumulate characters for transmission at 4800 baud.

The 1 MHz delay line loop recirculates every 8-1/3 ms. Because characters may be transmitted or received at a faster rate than would be possible if only one character was read from or written into the delay line loop, during each recycle of the loop, the five buffers accumulate blocks of up to five characters for transmission, and store blocks of up to five characters while waiting for access to the 1 MHz memory.

At 4800 baud, each bit time is 208. 3 μs and each 8-bit character is shifted in or out in 1. 666 ms (Fig. OP. 5). The five buffers allow characters to be accumulated for 5 X 1. 666 ms = 8. 330 ms at 4800 baud, or just slightly less than one delay line recycle time. If characters are coming in or going out at this maximum rate, the five buffers provide just enough storage time to allow the delay line to recycle to the next entry marker compare before a new block of characters is shifted in or out of the memory.

The rate at which characters are shifted through the I/O Register and from one buffer to the next is controlled by Serial Clock Transmit during transmission and Serial Clock Receive during reception. These clocks are derived from the SCU in the computer's I/O Buffer, if directly connected, or from the local data set, if not directly connected.

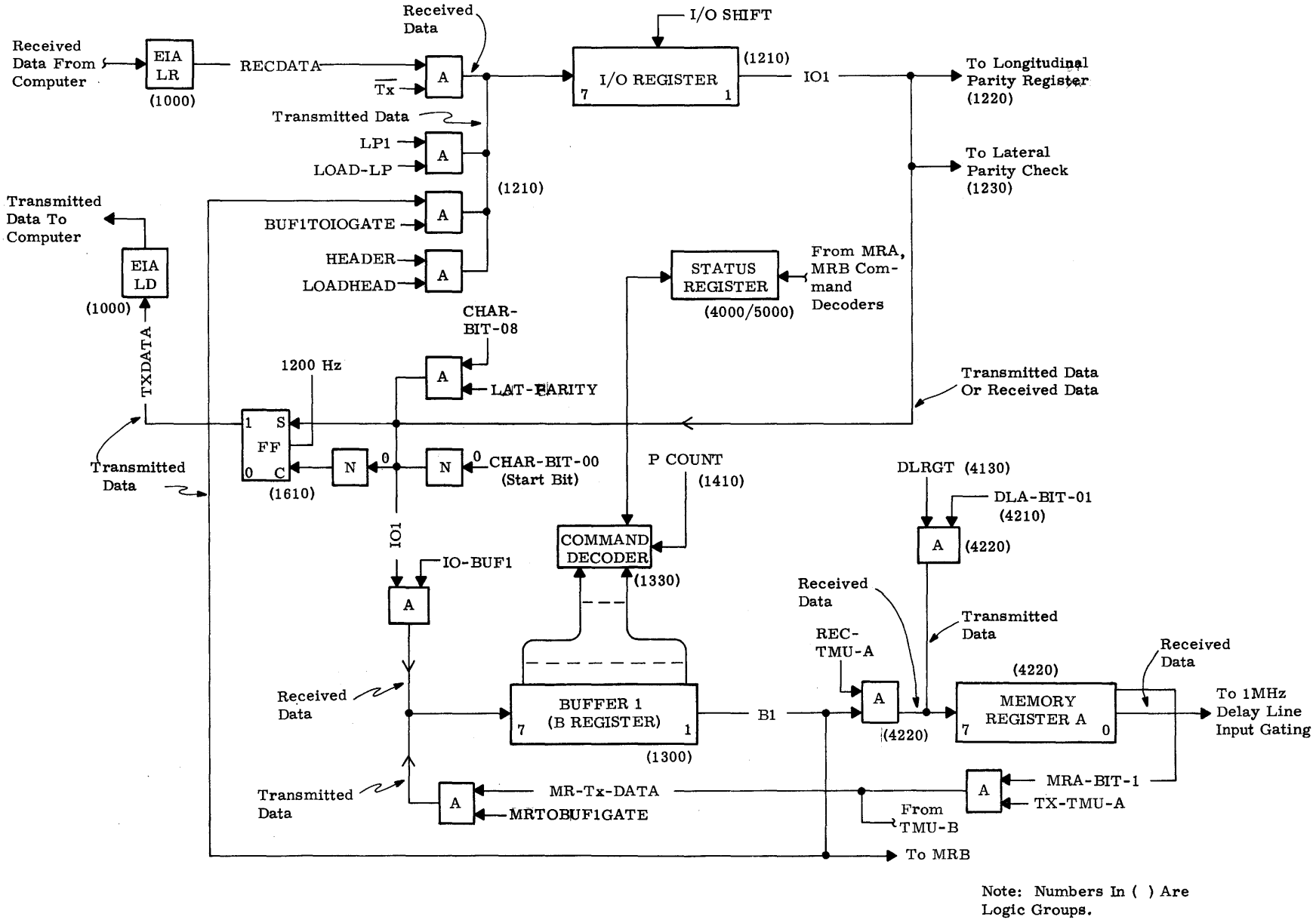
During text reception, the data is directed to the appropriate Memory Register by LD-MRA or LD-MRB. These terms are enabled when the address character is received in the message header.

During text transmission, the parallel load data for Buffer 1, PARLD-BUF1, is derived from the appropriate Memory Register, as determined by MRA-FULL (2100) or MRB-FULL (2200), which indicate that the Memory Register has a complete character, ready to go.

Program Count		DCU-765	DCU-775
RECEIVE MODE	P00	No such flip-flop, but when all counters reset, DLC is looking for computer carrier and timer is running.	Looking for computer carrier and if on, sync search starts. After IN-SYNC, go to P01. Timer is running while computer carrier is off.
	P01	Looking for SOH.	Looking for SOH.
	P02	Looking for EOT or ADR. If EOT, start transmit sequence after carrier drops.	Looking for EOT or ADR. If EOT, start transmit sequence after carrier drops.
	P03	Looking for ACK, NAK, or NUL.	Looking for ACK, NAK, or NUL.
	P04	Looking for STX.	Looking for STX.
	P05	Receiving text and looking for ETX.	Receiving text and looking for ETX.
	P06	Receiving LP. Looking for EOT.	Receiving LP.
	P07	(No such count)	Looking for EOT or SYN's at beginning of next message.
	P08	Received SOH, but ADR was illegal, look for ETX.	(No such count)
TRANSMIT MODE	P09	Checking LP.	Start TX, raise RTS, wait for DSR and CTS from computer or data set.*
	P10	(No such count)	Message search (ACK, NAK) and TMU text message search.
	P11	Start TX, raise RTS, wait for DSR and CTS*, send SOH.	Transmit SOH, and if polling message, EOT.
	P12	Message search (ACK, NAK) and for TMU text message.	Reset LP reg., load ADR character.
	P13	Load status (NUL, ACK, NAK).	Load status.
	P14	Load FC1 (space).	Load FC1.
	P15	Load FC2 (space).	Load FC2.
	P16	Load STX.	Load STX.
	P17	Send text.	Send text.
	P18	Load ETX.	Load ETX.
	P19	Load LP.	Load LP.
	P20	Any more messages, go to P13. If not, load EOT.	Any more messages, go to P13. If not, load EOT.
	P21		Load a NUL character.
	P22	Reset KBxTX request.	Reset KBxTX request.
	P23	End TX, drop RTS.	End TX, drop RTS.

*See Fig. OP. 3 for the meaning of DSR, RTS, and CTS.

Table THEORY. 1 DLC P-Count Sequencing



Note: Numbers In ( ) Are Logic Groups.

Fig. THEORY. 14 Data Line Controller, DCU-765

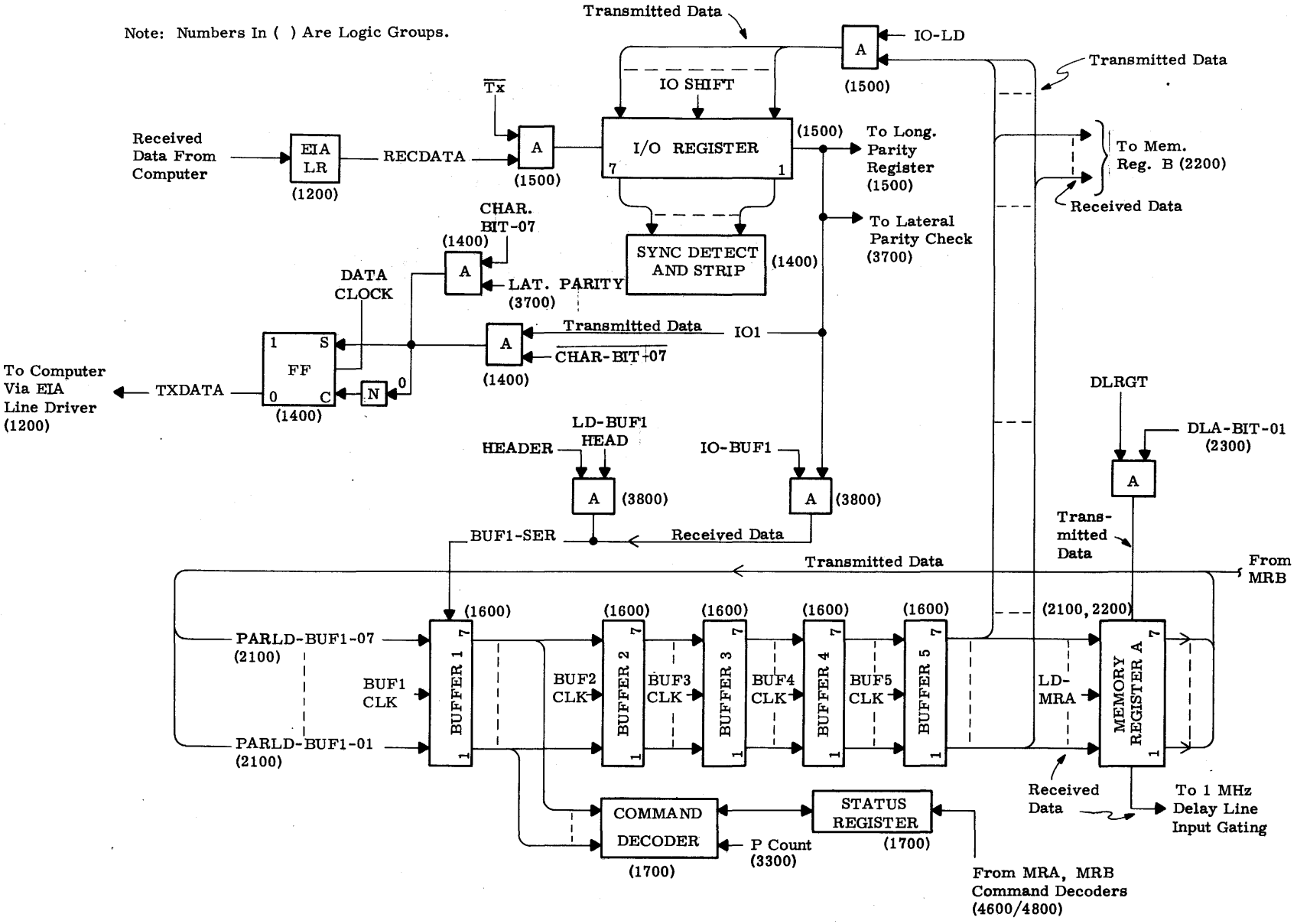


Fig. THEORY.15 Data Line Controller, DCU-775

READER COMMENTS

The General Electric Company solicits your comments on publications covering Process Computer equipment. Please explain any "no" responses in the COMMENTS section. Your comments and suggestions become the property of the General Electric Company.

- Name of Manual: _____
- What is your computer application: _____

_____

- How is this publication used:

  Familiarization ☐ Reference ☐

  Training ☐ Maintenance ☐

  Other (Explain) _____

	YES	NO
• Does this publication meet your requirements	☐	☐
• Is the material:		
1) Presented in clear text	☐	☐
2) Conveniently organized	☐	☐
3) Adequately detailed	☐	☐
4) Adequately illustrated	☐	☐
5) Presented at appropriate technical level	☐	☐

- Please provide specific text references (page number, line, etc.) with your comments.

NAME _____ DATE _____

TITLE _____

COMPANY NAME _____

AND ADDRESS _____

COMMENTS:

No postage necessary if mailed in the U. S. A.

Staple

Communications concerning Technical Publications should be directed to:

Manager, Technical Publications
Utility and Process Automation Systems Operation
2255 West Desert Cove Road
Phoenix, Arizona 85029

Fold Fold

FIRST CLASS
Permit No. 4091
Phoenix, Arizona

BUSINESS REPLY MAIL
NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY...

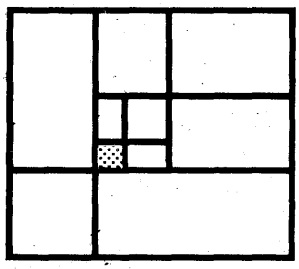
GENERAL ELECTRIC COMPANY
UTILITY and PROCESS AUTOMATION
SYSTEMS OPERATION
2255 West Desert Cove Road
Phoenix, Arizona 85029

Attention: Technical Publications

Fold Fold

Cut Along Line

Additional Comments:



GENERAL ELECTRIC