CONTROL DATA
8092 TeleProgrammer
REFERENCE MANUAL

IDP 106a
4/15/64

RECORD OF CHANGE NOTICES

C. N. NO.	DATE ORIGINATED	DATE ENTERED	INITIALS	REMARKS

Address comments concerning this manual to:

Control Data Corporation
IDP Publications
9549 Penn Avenue South
Minneapolis 31, Minnesota

CONTENTS

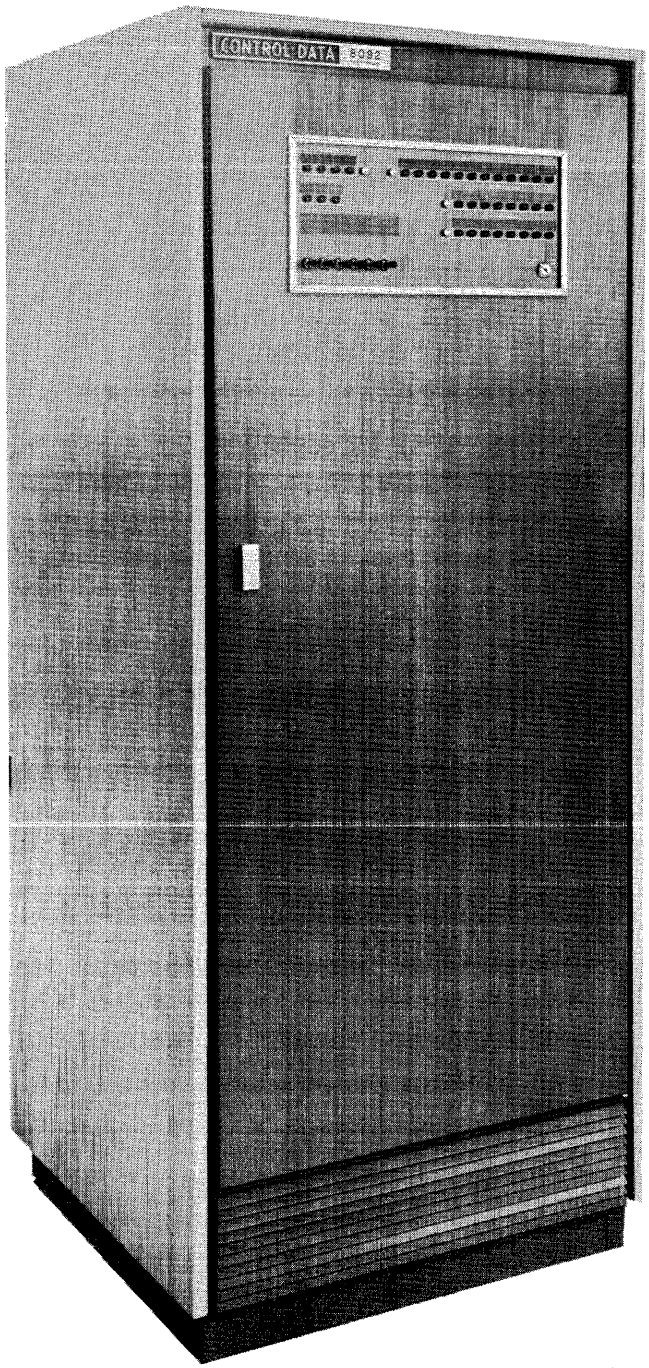
CHAPTER ONE - DESCRIPTION

CHAPTER TWO - OPERATION

CHAPTER THREE - PRINCIPLES OF OPERATION

FIGURES

TABLES



Control Data Corporation 8092 TeleProgrammer

CHAPTER ONE

DESCRIPTION

The CONTROL DATA* 8092 TeleProgrammer is a highly flexible, multi-purpose, stored program data processor and converter. It employs high speed (3 megacycle clock frequency) transistor amplifier circuits, diode logic, and an extremely fast magnetic core matrix memory. Word construction is 8 binary digits, parallel throughout, programmable to multiple-precision and to alpha-numeric and binary-coded decimal.

The basic memory of the CONTROL DATA 8092 TeleProgrammer consists of magnetic core storage with a basic capacity of 2048 8-bit words and an optional 4096 8-bit word memory with a storage cycle time of 4.0 usec. The internal program can directly address all 4096 words by the use of three 4-bit Tag registers. The TeleProgrammer has a total of 42 instructions, executed in one to four storage cycles; the time varies between 4.0 and 16.0 usec. The average program execution time for an instruction is approximately 10.0 usec.

---

* Registered Trademark.

A general purpose input channel and output channel system is provided for attaching a variety of input/output devices. Input and output transmissions are by 8-bit characters. Standard input/output is by optional punched papertape read and punch units. Optionally, other input/output devices, such as on-line typewriter, magnetic tape handlers, punched card units, digital communication units, analog-to-digital converters, or similar equipment may be added.

A buffer input/output channel permits the TeleProgrammer to continue high-speed computation while communicating with external equipment. Any peripheral unit connected to the buffer channel may also be addressed using the normal input/output channel.

Four interrupt lines, including a manual interrupt, allow more effective use of computer time through sophisticated programming.

Operation of the 8092 TeleProgrammer is sequenced by an internally stored program. This program, as well as the data being processed, is contained in the high-speed, random-access storage. An instruction is an 8-bit word comprising a 6-bit function code, F, and a 2-bit code Tag register reference, T. Program modifications are accomplished and operands are manipulated by memory or indirect addressing through use of T.

The 8092 is constructed in a stand-up cabinet 68 3/4 inches high. Operating requirements are a maximum of 850 watts of 110 volt, 60 cycle power, and a normal room temperature not exceeding 100 degrees F.

SUMMARY OF CHARACTERISTICS

Type of Machine	Digital, stored program
Mode of Operation	Parallel
Word Length	8 bits
Memory	2048 8-bit words expandable to 4096 8-bit words.
Type of Instruction	Single address
Address Features	No address, memory and indirect addressing.
Instruction Word	8-bit word (6-function code, F; 2-bit Tag reference, T).
Input/Output	Interrupt feature on all input/output channels Normal and Buffered input/output channels.
Instruction Repertoire	42 instructions for arithmetic and manipulative operations.
Type of Arithmetic	Binary; one's complement.
Control Provisions	Complete manual control from control panel.
Assembler	The ability to use the OSAS or OSAS-A assembler for those who have a 160 or 160-A computer.

DESCRIPTION OF REGISTERS

The CONTROL DATA 8092 TeleProgrammer uses six operational and eight transient registers. Operational registers are referred to implicitly in the individual program steps, and may be modified by manually inserting quantities into them. The transient registers, important only to the internal operation of the machine, are not available for modification from the control panel.

OPERATIONAL REGISTERS

A Register	A is the principal arithmetic register for most arithmetic operations. A operates as a subtractive accumulator. The quantity zero is represented by all zeros.
Z Register	As a buffer register, Z receives the word read out of storage and holds the word to be written into storage. For addition and subtraction, the contents of Z are added to or subtracted from the contents of A.
P Register	P, the Program Address register, contains the address of the current instruction. At the beginning of each instruction the contents of P are increased by one to provide the address of the instruction; if a jump is called for, a jump address is entered in P. For certain instructions, the contents of P are advanced by 2.
T Registers	The T registers hold the upper 4 bits of the 12-bit storage address used in direct or indirect addressing.

TRANSIENT REGISTERS

BFR	During a buffer operation the Buffer Data register holds the word of information being transferred to or from storage.
BER	During a buffer operation the Buffer Entrance register holds the address to or from which information is being transferred. It is advanced by one with each data-word transfer.
BXR	The Buffer Exit register contains the terminating address (LWA + 1) for the current buffer operation. After each data-word transfer, a comparison is made between BER and BXR to determine whether another word should be transferred.
A' Register	A' is an auxiliary arithmetic register. The results of arithmetic operations are first formed in A' and then transmitted to the A, Z, S, or P registers.
S Register	S functions as the Storage Address register. Prior to any storage reference, the address word is entered in S, and the contents of S are used to select the storage location involved in the reference.
F Register	F holds the lower 6 bits and f' the upper 2 bits (Tag register reference) of an instruction word (the function code) throughout the execution of the instruction. The execution of an instruction is under control of the quantity if F.
PSR	The P Storage register is used in interrupt operations. During such operations, the address in P is transferred to PSR, thereby releasing P and enabling the TeleProgrammer to jump to the interrupt address specified by the interrupt level. PSR is then transferred to P at the end of the interrupt routine.
O Register	The O register holds the upper 6 bits of a 12-bit external function code used for controlling external equipment.

All registers, but PSR, S, F, A' and O may be displayed by setting lever switches on the panel.

ADDRESSING MODES

Maximum flexibility of the 6-bit address is provided by three addressing modes:

No Address - N	Using the memory location immediately following the instruction as an 8-bit operand permits the performance of arithmetic and logical operations with an 8-bit constant associated with the instruction. This eliminates the need for entering many constants into memory.
Memory Address - D	Refers to an 8-bit operand in any storage location addressed by the memory location immediately following the instruction and the Tag register referenced by T.
Indirect Address - 1	Provides for operand references. Instructions employing indirect addressing use the memory location immediately following the instruction to refer to one of the first 256 storage locations. The contents of this address are used along with the Tag register (referenced by T) as the address of the operand.

INPUT/OUTPUT

Seven instructions deal with input/output operations. Instructions 70 and 72 initiate input. Output data is sent by instructions 71 and 73. External function codes for directing the operation of the input/output equipment are sent by instruction 75. Instruction 76 provides an 8-bit input to A, and 74, an 8-bit output from the memory location immediately following the 74 instruction.

Communication with external equipment, effected in the same manner for either normal or buffer input/output channels, is accomplished by first executing instruction 75; the operand obtained is the external function code.

This code selects the particular equipment and specifies the operation mode. Only one input or one output equipment is active at a given time on the normal input/output channel.

The input or output sequence of transmission is as follows: The address designated by the instruction (immediately following instruction) is placed in the A register and used for storing the input word or for reading a word for output. After each transmission the address in A is advanced by one and compared with the contents of the address immediately following the starting address. If the comparison shows inequality, another transmission of a data word is initiated. If the comparison shows equality, the program continues with the instruction at the address following the ending address. The 74 instruction initiates a single output operation with the 8-bit quantity from the memory location immediately following the 74 instruction being placed in the output channel.

The 70 and 71 initiate buffer input and buffer output, respectively. Prior to executing these, however, the buffer entrance and exit registers must be set and a device on the buffer channel selected by a 75 instruction. A complete list of instructions arranged in order of instruction codes is shown in Table 1-2. External Function codes and Status Responses for TeleProgrammer peripheral equipment are shown in Table 1-3.

TABLE 1-1
THE 8092 TeleProgrammer
INSTRUCTION REPERTOIRE

Functions	Rel. Code	Octal Code	Cycle Time *
LOADS:			
Load A (No.)	LDN	20	2
Load A (Mem.)	LDM	21	3
Load A (Ind.)	LDI	22	4
Load Comp. (Mem.)	LCM	25	3
Load Comp. (Ind.)	LCI	26	4
Tag Reg. to A	TTA	03	1
BER to A	BER	06	1
STORES:			
Store A (Mem.)	STM	41	3
Store A (Ind.)	STI	42	4
A to Tag Reg.	ATT	02	1
A to BER	ABR	04	1
A to BXR	ABX	05	1
JUMPS:* **			
If A = 0	ZJP	60	2
If A ≠ 0	NZP	61	2
If A ≥ 0	PJP	62	2
If A < 0	NJP	63	2
Unconditional	UJP	64	2
SHIFTS:			
A left 1 bit	SHA	01	1

Functions	Rel. Code	Octal Code	Cycle Time *
ARITHMETICS:			
Add (No Adr.)	ADN	30	2
Add (Mem.)	ADM	31	3
Add (Indirect)	ADI	32	4
Subtract (No)	SBN	34	2
Subtract (Mem.)	SBM	35	3
Subtract (Ind.)	SBI	36	4
Replace Add (Mem.)	RAM	51	4
Replace Add 1	RAO	55	4
LOGICALS:			
Log. Prod. (No)	LPN	10	2
Log. Prod. (Mem.)	LPM	11	3
Log. Prod. (Ind.)	LPI	12	4
Log. Sum (No)	LSN	14	2
Log. Sum (Mem.)	LSM	15	3
Log. Sum (Ind.)	LSI	16	4
IN-OUT:			
Input Normal	INN	72	**
Output Normal	OUT	73	**
Input Buffer	IBI	70	2
Output Buffer	IBO	71	2
Input to A	INA	76	2
Output No. Adr.	OTN	74	2
CONTROLS:			
Ext. Function	EXF	75	3
Clear Interrupt	CIL	13	1
Clear Buffer	CBC	07	1
Error Stop	ERR	00	-
Halt	HLT	77	1

* Cycle Times; each cycle = 4 microseconds.

** 3 + 2(X + 1) + terminate time. Where X = No. of words.

*** (Jump cycle times above are 1 less cycle if jump is not made.)

TABLE 1-2 LIST OF INSTRUCTIONS
Numerical Listing with Comments

Octal Code	Mnemonic Code		
00(X)*	ERR	Error Stop	This is an illegal instruction -- as such, it can be used as an Error Stop.
01(X)	SHA	Shift A Left One Bit	Shift the contents of A left -- end around -- 1 bit position. Bits coming off the left end of the A register go to lowest bit position on the right end of the register.
02(Y)	ATT	A to Tag Register	Transfer the lower 4 bits of the A register into the designated Auxiliary Tag register.
04(Y)	ABR	A to Buffer Entrance	Transfer the contents of A to the lower positions of the Buffer Entrance register. The lower 2 bits of Tag register 3 become bits 8 and 9 of BER. If the buffer is busy a jump occurs to the combined address contained in the designated Tag register of the first word and the contents of the second word of the instruction set. If not busy, control continues to the next instruction set in sequence.
05(Y)	ABX	A to Buffer Exit Register	Transfer the contents of A to the lower 8 bits of the Buffer Exit register (BXR). If the buffer is busy a jump occurs to the combined address contained in the designated Tag register of the first word and the contents of the second word of the instruction set. If not busy, control continues to the next instruction set in sequence.
06(X)	BER	Buffer to Entrance Register to A	Load the A register with the lower 8 bits of the Buffer Entrance register.

Octal Code	Mnemonic Code		
07(X)	CBC	Clear Buffer Controls	This instruction has the effect of sending a zero to buffer control and thus putting that device in a "ready state". If this instruction is used during a buffer operation, it will stop the buffer.
10(X)	LPN	Logical Product (No Address)	Form in A the Logical Product of the contents of A and the contents of the second word of the instruction set.
11(Y)	LPM	Logical Product (Memory Address)	Form in A, the Logical Product of the contents of A and the contents of the memory location whose address is the combined contents of the designated Tag register, and the second word of the instruction set. The initial contents of the memory location remains unchanged.
12(Y)	LPI	Logical Product (Indirect Address)	Form in A the Logical Product of the contents of A and the contents of the memory location whose address is the combined contents of the designated Tag register and the contents of one of the first 256 decimal locations. The address of this decimal location is given in the second word of the instruction set. The initial contents of the memory location remain unchanged.
13(X)	CIL	Clear Interrupt Lockout	This instruction clears the interrupt lockout flip flop and it must be programmed at the end of every routine which is initiated by the interrupt. To return to main program after clearing interrupt lockout, the Tag designation must be 1. If zero, control continues in sequence.
14(X)	LSN	Logical Sum (No Address)	Form in A the Logical Sum of the contents of A and the second word of the instruction set.
15(Y)	LSM	Logical Sum (Memory Address)	Form in A the Logical Sum of the contents of A and the contents of the memory location whose combined address is given in the designated Tag register and the second word of the instruction set.

Octal Code	Mnemonic Code		
16(Y)	LSI	Logical Sum (Indirect Address)	Form in A the Logical Sum of the contents of A and the contents of the designated Tag register and one of the first 256 (decimal) locations. The location of one of these 256 locations is given in the second word of the instruction set.
20(X)	LDN	Load A (No Address)	Load the A register with the contents of the second word of the instructional set. A quantity of 8 bits can be entered into A by this instruction.
21(Y)	LDM	Load A (Memory Address)	Load the A register with the contents of the of the memory address whose lower 8 bit digits are given in the second instruction word and whose upper 4 bits are contained in the designated Auxiliary Tag register.
22(Y)	LDI	Load A (Indirect)	Load A with the contents of the address whose lower 8 bits are contained in one of the first 256 (decimal) addresses, and whose upper 4 bits are contained in a designated Auxiliary Tag register. The location in the core (one of the first 256 decimal addresses) is given in the second instruction word. The Auxiliary Tag register is indicated in the first word.
25(Y)	LCM	Load Complement to A (Memory)	Load the A register with the complement of the contents of the memory address whose lower 8 bits are given in the second instruction word and whose upper 4 bits digits are contained in the designated Auxiliary Tag register.
26(Y)	LCI	Load Complement to A (Indirect)	Load A with the complement of the contents of the address whose lower 8 bits are contained in one of the first 256 (decimal) addresses and whose upper 4 bits digits are contained in the designated Auxiliary Tag register. The location in the core is given in the second instruction word. The Auxiliary Tag register is indicated in the first word.

Octal Code	Mnemonic Code		
30(X)	ADN	Add (No Address)	Add to the A register the 8-bit number given in the second word of the instruction set. The sum is left in A.
31(Y)	ADM	Add (Memory Address)	Add to A the contents of the combined address given in the designated Tag register and the second word of the instruction set.
32(Y)	ADI	Add (Indirect Address)	Add to A the contents of the combined address contained in the designated Tag register and one of the first 256 decimal locations indicated in the second word of the instruction set.
34(X)	SBN	Subtract (No Address)	Subtract from the A register the number contained in the second word of the instruction set. The difference is left in A register.
35(Y)	SBM	Subtract (Memory Address)	Subtract from the contents of A, the contents of the combined address contained in the designated Tag register and the second word of the instruction set.
36(Y)	SBI	Subtract (Indirect Address)	Subtract from the contents of A, the contents of the combined address contained in the designated Tag register and the location of one of the first 256 decimal registers, indicated by the second word of the instruction set.
41(Y)	STM	Store A (Memory Mode)	Store the contents of the A register whose address is equivalent to the combined contents of the designated Tag register and the second word of the instruction set.
42(Y)	STI	Store A (Indirect Mode)	Store the contents of the A register whose address is equivalent to the combined contents of the designated Tag register and the contents of one of the first 256 decimal core registers. The location of which is given in the second instruction word.

Octal Code	Mnemonic Code		
51(Y)	RAM	Replace Add (Memory Address)	Add the contents of the A register to the contents of the memory address formed by the contents of the designated Tag register and the second word of the instruction set. The sum thus formed, remains in A, and replaces the initial contents of the memory address.
55(Y)	RAO	Replace Add One	Add 1 to the contents of the memory address indicated by the combined contents of the designated Tag register and the second word of the instruction set. This sum is performed in A and remains in A at the end of the instruction.
60(Y)	ZJP	Zero Jump	If the contents of A equals zero, jump to the combined address contained in the designated Tag register and the second word of the instruction set. If the contents of A are not zero, continue in sequence with next set of instructions.
61(Y)	NZP	Not Zero Jump	If contents of A are not zero, jump to the combined address contained in the designated Tag register and the second word of the instruction set. If the contents of A are zero, continue in sequence with the next set of instructions.
62(Y)	PJP	Positive Jump	If the contents of A are positive (equal or greater than zero), jump to the combined address contained in the designated Tag register and the second word of the instruction set. If the contents of A are not positive, continue in sequence.
63(Y)	NJP	Negative Jump	If the contents of A are negative, jump to the combined address contained in the designated Tag register and the second word of the instruction set. If the contents of A are not negative, continue in sequence with the next set of instructions.

Octal Code	Mnemonic Code		
64(Y)	UJP	Unconditional Jump	Jump to the combined address contained in the designated Tag register and the second word of the instruction set.
70(Y)	IBI	Initiate Buffer Input	This instruction initiates an input buffer cycle. If the buffer channel is not busy, control goes to the next instruction following the second word of the instruction set. If the buffer channel is busy, a jump occurs to the memory location whose combined address is contained in the designated Tag register and the second word of the instruction set.
71(Y)	IBO	Initiate Buffer Output	This instruction initiates the output buffer cycle. If the buffer channel is busy, a jump occurs to the combined memory address given in the designated Tag register and the second word of the instruction set. If the buffer channel is not busy, control goes to the next sequential instruction following the instruction set.
72(Y)	INN	Input Normal	Input a number of words to memory starting at the memory address contained in the designated Tag register and the second word of the instruction. Last word address +1 is contained in the 3rd word of instruction set.
73(Y)	OUT	Output Normal	Output a number of words to memory starting at the memory address contained in the designated Tag register and the second word of the instruction set. Last word address +1 is contained in the 3rd word of instruction set.
74(X)	OTN	Output No Address	The instruction outputs one word. This word is the second word of the instruction set.
75(X)	EXF	External Function	This instruction is used to select an external input or output device to communicate with the 8092 TeleProgrammer.

Octal Code	Mnemonic Code		
76(X)	INA	Input to A	This instruction inputs one word from a previously selected input device to the A register.
77(X)	HLT	Program Stop	This instruction is used to bring the program to a halt.

NOTE:

*(X)-instructions followed by X means that this instruction ignores the upper two bits and the operand is in the next sequential memory location following the instruction or the operation is performed in 1 cycle.

*(Y)-instruction followed by Y means that this instruction uses the upper two bits to reference one of three 4-bit registers to establish a twelve bit address made up of 4 bits from the Tag register and 8 bits from memory.

TABLE 1-3

EXTERNAL FUNCTION CODES AND STATUS RESPONSES FOR PERIPHERAL EQUIPMENT

All External Function Codes and Status Responses are given in octal code.

1. 8098 TALLY READER

   A. External Function Codes

      4102 Select Reader

   B. No Status codes

2. 8096 TELETYPE MODEL 33 PAGE PRINTER

   A. External Function Codes

      4211 Select Input
      4221 Select Output

   B. No Status codes

3. 8099 350 PAPER TAPE READER

   A. External Function Codes

      4102 Select Reader

   B. No Status codes

4. 8091 BRPE-11 PAPER TAPE PUNCH

   A. External Function Codes

      4104 Select Paper Tape Punch

   B. No Status codes

5. 161 INPUT/OUTPUT TYPEWRITER

A. External Function Codes

4210 Select Typewriter Output
4220 Select Typewriter Input
4240 Request Typewriter Status

B. Status Response Codes

0000 Typewriter Ready
0004 Typewriter Power Off
0010 Typewriter not in Computer Status
0020 Input Character Ready
0040 Output in Use

NOTE: If a second typewriter is added, the master octal bits will be 43.

6. 8093 MAGNETIC TAPE SYNCHRONIZER

A. Function Codes

(X) is designated by Unit Select Switch on top of 603 Tape Handler Cabinet. Operator shall use positions 0 through 3.

"Binary" means odd parity is generated and checked on tape.
"Coded" means even parity is generated and checked on tape.

1X02 Status Request 1
1X03 Status Request 2
1X04 Set Interrupt Lockout
1X05 Clear Interrupt Lockout
1X07 Programmed Clear
1X10 Write, Binary, Low Density
1X11 Write, Binary, High Density
1X12 Write, Coded, Low Density
1X13 Write, Coded, High Density
1X14 Write File Mark
1X20 Search Forward to File Mark
1X21 Search Backward to File Mark
1X22 Rewind to Load Point
1X24 Search Forward One Record
1X25 Search Backward One Record
1X26 Rewind Unload
1X30 Read, Binary, Low Density
1X31 Read, Binary, High Density
1X32 Read, Coded, Low Density
1X33 Read, Coded, High Density
1X34 Read

B. Status Responses

Status Request 1

Bit Location	Octal	Description
0	001	Coded (Even Parity, no Odd Parity)
1	002	Transport Not Ready
2	004	Parity Error
3	010	Program Error
4	020	End of File
5	040	Tape Mark (Load Point or End of Tape)
6	100	High Density (Not Low Density)
7	200	Busy (Tape Motion)

Note: Illegal BCD will bring bits Ø, 2, and 3 (Octal Code 015)

Status Request 2

Bit Location	Octal	Description
0	001	High Density
1	002	Even Parity
2	004	End of File
3	010	Load Point
4	020	Write Not Ready
5	040	Searching for File Mark
6	100	Writing File Mark

7. 162 MAGNETIC TAPE SYNCHRONIZER

A. External Function Codes (6-bit mode, X = 1)(12 bit mode, X = 2)

X110 thru X117 — Write if OUT is given. Write End-of-File Mark if no OUT is given.

X120 thru X127 — Backspace one record if INA is given. Seach backward to End-of-File Mark if no INA is given.

X120 thru X137 — Read forward if INPUT is given. Search forward to End-of-File Mark if no INPUT is given.

X140 thru X147 — Request Status

X150
thru Rewind Unload
X157

X160
thru Rewind Load
X167

X171 Set tapes to odd parity

X172 Set tapes to even parity

2100
thru High density
2107

1100
thru Low density
1107

B. Status Response Codes

0000 Odd parity selected - no errors
0001 Even parity selected - no errors
0002 Tape X not ready
0004 Parity error
0015 Illegal BCD detected on Write
0020 End-of-File Mark read
0040 End-of-Tape or Load Point sensed
0100 High density
0200 Tape X busy

NOTE: The last octal bit designates one of the four (eight) 60X's. The master octal bits 12, 13, 22, and 23 are used for second and third tape control. If the tape transport is a 606, a 6-bit high density selection is illegal (a programmer consideration).

6. 166-2 LINE PRINTER

A. External Function Codes

0700 Asynchronous print
0710 Synchronous print
0740 Check status
072X Advance forms

B. Status Response Codes

0000	166-2 ready
0001	Buffer busy
0002	Out of paper
0004	Paper moving
0010	Drum stationary
0020	Off-line

9. 167-2 CARD READER (Hollerith Facility)

A. External Function Codes

4500	EF clear
4501	Free run read
4502	Single cycle read
4504	Negate translate, H BCD
4505	FRR, H BCD and pack
4506	SCR, H BCD and pack
4540	Check status

B. Status Response Codes

0000	Card reader ready
0001	Hopper empty
0002	Stacker full
0004	Feed failure
0010	Program error
0020	Amplifier failure
0040	Motor power off

10. 170 CARD PUNCH CONTROL UNIT

A. External Function Codes

3002	Punch
3040	Check Status

B. Status Response Codes

0000	170 ready
0200	MS in 1604 position
2000	Punch not ready

11. 177 CARD READER

A. External Function Codes

4500	EF clear	
4501	Free run read	
4502	Single cycle read	
4505	Negate translate, H	BCD, free run read
4506	Negate translate, H	BCD, single cycle read
4510	Gate card	
4540	Status request	

B. Status Response Codes

0001	Input tray empty
0002	Primary or secondary stacker full
0004	Feed failure
0010	Late input request
0020	Pre-read error
0040	Manual on or motor power off
0100	Read comparison error
0200	End of file
0400	Ready

12. 8094 PERIPHERAL ADAPTOR(Required when 12-bit interface is needed for peripheral equipment)

A. External Functions

6301	Select
6302	De-Select

B. No Status Codes

13. 8060 SERIES DIGITAL COMMUNICATIONS TERMINALS

A. External Function Codes

36X0	Select stop send
36X1	Select send
36X2	Select data input
36X3	Select status input
36X4	Select

B. Status Response

0000	Computer has cleared to send
0002	Carrier is on and computer sending

14. 8095 RECORD TRANSMISSION CONTROL PANEL

A. EXF Codes

35XX Select to input data
3501 Status Request 1
3502 Status Request 2

B. Status Responses #1

The Control Panel will respond with the status responses when the TeleProgrammer sends a 3501 external function. These status responses are, in effect, the way in which the terminal operator communicates with the TeleProgrammer.

Octal Response	Meaning
001	Send End of Message
002	Send Come to Phone
004	End of Message Not Acknowledged
010	Come to Phone Not Acknowledged
020	Stop (Not Ready)
040	- - - - - -
100	Translate
200	Send Mode (Not Receive)

Response #2

The Control Panel responds with the status responses shown in Table 3 when the TeleProgrammer sends a 3502 external function. This status response indicates to the TeleProgrammer which peripheral device to select for the input (send) or output (receive) operation it is to perform.

Select Send Switch Position	Octal Response	Select	For
1	0X0	Magnetic Tape	
2	0X1	Punched Cards	
3	0X2	Paper Tape	Sending
4	0X4	I/O Writer	
5	1X1		Equipment
6	1X2		
7	1X4		

Select Send Switch Position	Octal Response	Select	For
1	00X	Receive Device as Called for in Header	Receiving
2	01X	Magnetic Tape	
3	02X	Punched Cards	Equipment
4	04X	Line Printer	
5	21X	Paper Tape	
6	22X	I/O Writer	
7	24X		

CHAPTER TWO

OPERATION

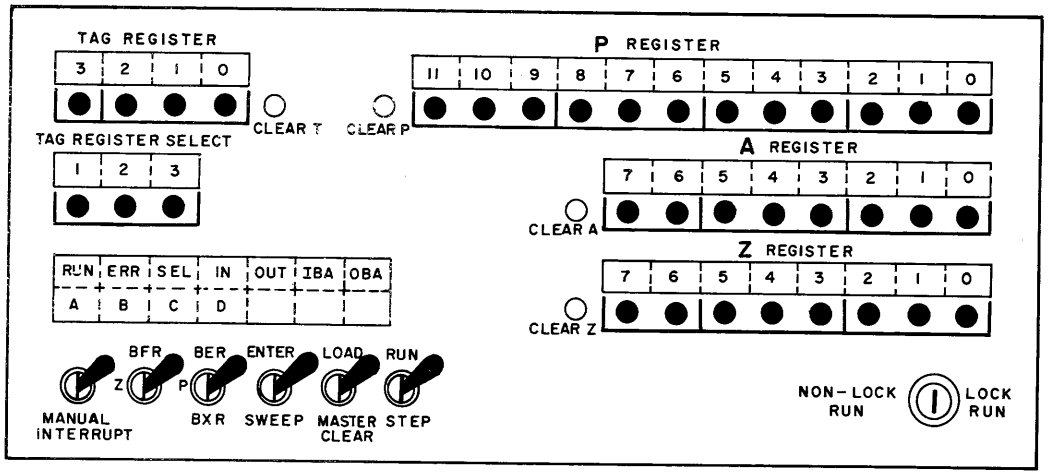


Figure 2-1 8092 Operator's Panel

TeleProgrammer OPERATOR's Panel

The 8092 TeleProgrammer Operator's Panel consists of several displays and switches necessary for the operation of the TeleProgrammer. The panel (see figure 2-1) contains six display windows, six switches, and a lock switch. Four of the display windows can display in binary the contents of nine 8092 registers. Buttons beneath these displays clear and enter data into the P, A, Z, and Tag registers (the only registers into which data may be entered or cleared). A fifth window contains information as to which Tag register has been selected. The sixth window contains the operating lights which indicate the status of operation

of the TeleProgrammer. At the bottom of the panel are located all the operating and mode switches. The operation of these switches is explained below:

SWITCHES

Manual Interrupt -Momentary depression causes the Teleprogrammer to enter an interrupt routine to determine the nature of the interrupt.

BFR,Z -This 3-position switch chooses the register that is to be displayed in the 8-bit Z register display.

Up - Displays the last word processed during the last buffer operation (BFR register).

Center - Shows the current contents of the Z register (Z register).

Down - Not assigned.

BER, P, BXR -This 3-position switch chooses the register to be displayed in the 12-bit P register display.

Up - Displays the address of the last word transferred out, the next word to be transferred in on the buffer channel (BER register).

Center - Displays the address of the current instruction (P register).

Down - Displays the LWA + 1 of the last buffer operation (BXR register).

ENTER/SWEEP

-Sweep is used to display the contents of core storage locations. Enter is used for entering information into core storage from the console.

LOAD/MASTER CLEAR

- LOAD position allows specially prepared paper tapes to be read into storage by the paper tape reader.

Master CLEAR performs a TeleProgrammer master clear which

a. Clears the registers

b. Clears the control flip flops

c. Clears all waiting interrupts and removes interrupt lockout.

Note: The master clear does not alter core storage.

RUN/STEP

Up - In RUN position a program is executed at high speed starting at the location specified by the P register.

Center - Center position stops the computer program. If the switch is in RUN and an ERR or HLT instruction is executed, the switch must be returned to neutral and then placed in RUN to continue computation.

Down - In STEP position, one storage cycle of an instruction is executed each time the switch is set; a program may be executed one instruction at a time for debugging.

NON-RUN LOCK
RUN LOCK

In the Lock position all other switches are disabled and the TeleProgrammer is locked in the RUN position.

In the non-lock position, the console switches are enabled and the TeleProgrammer programs can be operated and modified from the console.

DISPLAYS

Z REGISTER

-This display known as the Z register group displays the Z and BFR registers in accordance with the setting of the BFR, Z switch.

A Register

-Displays the current contents of the A register.

P Register

-This display known as the P register group displays the BER, P, and BXR registers in accordance with the setting of the BER, P, BXR switch.

TAG REGISTER

-This display indicates the Tag register currently being referenced by an instruction. The contents of any Tag register may be displayed by depressing one of the buttons directly below the select indicators. Depressing one of the select buttons also enables the Tag registers to be manually set or cleared.

STATUS INDICATORS

RUN

-Indicates that the TeleProgrammer is in RUN status. This does not necessarily indicate that instructions are being executed.

ERR

-Indicates that a timing fault has occurred.

SEL

-Displayed each time an EXF instruction is executed; remains until selection is completed. A constant display of SEL with no apparent input/output action usually indicates the TeleProgrammer has attempted an illegal selection.

IN	-Displayed during all normal input operations. A constant display of IN with no apparent input action usually indicates that input was attempted without proper unit selection. IN is also displayed when the TeleProgrammer is waiting for an external device to supply data.
OUT	Displayed during all normal output operations. A constant display of OUT with no apparent output action usually indicates that output was attempted without proper unit selection.
IBA	Displayed during all buffer input operations.
OBA	Displayed during all buffer output operations.
A, B, C, or D	Indicates which storage reference cycle will be executed at the next operation of the Run/Step switch. When a master clear is performed, D is displayed indicating that the next operation to be executed, when the Run/Step switch is operated, will be to fetch the instruction from memory at the address indicated by the P register.

STARTING THE 8092 TeleProgrammer

1) Be sure the TeleProgrammer is plugged into proper power source and room temperature is within the prescribed limits.

2) Turn Power switch on power supply to ON.

3) Master clear by momentarily pressing Load/Clear switch to Clear.

4) When the ERR light goes out, the TeleProgrammer is ready to operate. If repeated master clears do not turn the Red ERR light off, turn off the 8092 and call maintenance.

LOADING A PROGRAM OR DATA

Paper Tape Load Format

1) Master Clear

2) Turn on reader

3) Insert paper tape in reader

4) Set P to starting location

5) Set Load/Clear switch to LOAD

6) Set Run/Step switch to RUN. Paper tape will load and TeleProgrammer will stop.

ENTERING DATA FROM THE TeleProgrammer Panel

1) Master clear. Set Enter/Sweep switch to ENTER

2) Set P to location into which data is to be entered.

3) Enter one word of data into the A register.

4) Set Run/Step switch to STEP, once. At this point A is clear and the data word is in storage and in Z.

5) If data is to be entered into consecutive locations, go to step 3 and P will be advanced by one on step 4. If data is to be entered into non-consecutive locations, clear P. Go to step 2.

EXAMINING THE STORAGE CONTENTS

1) Master clear. Set Enter/Sweep switch on SWEEP

2) Set P to location to be examined.

3) Press Run/Step switch to STEP, once. The contents of the location specified by P will appear in Z.

4) To examine consecutive locations, go to step 3 and P will be advanced by one on step 3. To examine non-consecutive location, clear P, go to step 2.

CHAPTER THREE
PRINCIPLES OF OPERATIONS

The Principles of Operation section emphasizes circuit logic rather than electrical operation. The TeleProgrammer is constructed from a great many circuits and stages which are electrically similar. The functions of the circuits are described in this chapter.

The TeleProgrammer may be explained conveniently by discussing it in four major categories:

Control Section	Determines how the TeleProgrammer is to respond to a stored program and implements the necessary action.
Storage Section	Holds the pertinent program steps and operands used in problem solving.
Arithmetic Section	Performs the arithmetic processes.
Input/Output	The means by which information is usually entered into the TeleProgrammer, and the only method of communication to peripheral units for storage of processed data on magnetic tape, paper tape, or typewriter copy.

The functional descriptions contained herein are supplemented by:

Logic Diagrams	A graphic presentation of the logical relationships.
File of Equations	The complete and ultimate source of all information concerning the logic of the computer.

The information in these two volumes is so necessary for a complete understanding of the logic circuits that they are not called out during every functional description; rather, it is assumed that the reader will automatically refer to them.

Three additional sources will be found useful from time to time:

Appendix A	Instruction timing charts pinpointing the steps involved in execution of each instruction.
Appendix C	Information for installing the TeleProgrammer; includes weights, power consumption, and so forth.

A block diagram of the TeleProgr ımmer is shown in figure 3-1.

CONTROL SECTION

All TeleProgrammer logic, that is, the transfer of information from one logic element to another, is performed at a rate determined by a series of oscillator cards. Initiation and control of the logical operations or commands necessary for the execution of an instruction, however, are governed by a main timing chain. In the following paragraphs, each of the components in the basic control mechanism is discussed.

REGISTERS

The TeleProgrammer has fourteen registers. Six of these, A, P, Z, T_1, T_2, and T_3, are operational in that their contents may be modified by manipulation of the associated bit-stage push buttons. The eight transient registers A', S, F, PSR, BFR, BER, BXR and O are important only to the internal operation of the TeleProgrammer. The contents of all registers but PSR, A', S, F and O may be displayed on the console.

A Register (8 Bits)

The A register is the principal arithmetic register. For most arithmetic operations, A, together with the adder, operates as a 8-bit subtractive accumulator.

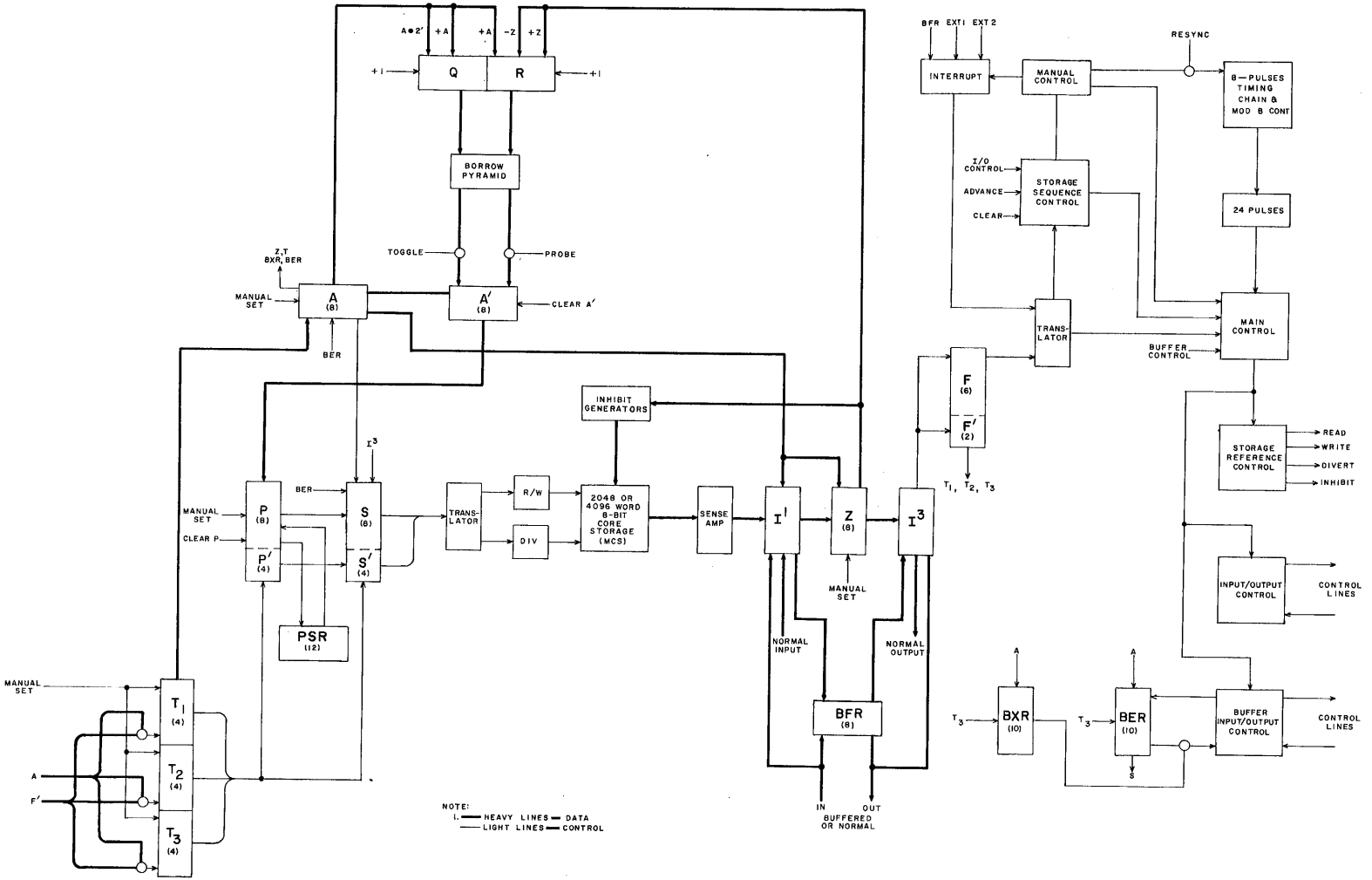


Figure 3-1. Block Diagram of TeleProgrammer

The arithmetic processes of addition and subtraction are performed modulus 2^8-1. The quantity zero is represented by all zeros.

Each stage of A incorporates a flip-flop (FF) with inputs from the corresponding stage of the A' register. The set side of the FF has an input from the bit-stage push button on the console control panel; the clear side has an input from the Clear A function. Additional forced (set/clear) inputs come from the Buffer Entrance register and Tag registers.

Outputs from A go to the adder through R and Q inverters, and to second-level control for sensing A = 0. The final stage in A sends its output to the function translator for determining, under certain conditions, whether the quantity in A is positive (A^{11} = "0") or negative (A^{11} = "1"). Outputs from A also go to Z, BER, BXR, S, T_1, T_2 and T_3.

Z Register (8 bits)

The Z register performs a variety of functions. It acts as a storage restoration register for non-buffered data read-out of memory, and it holds one of the operands in arithmetic operations. It communicates with external equipment via the input/output channels, and is one of the two means of communication with core storage.

Because the number of transmission paths through Z exceeds the input and output pin capacity of any one FF card, each stage of Z has four inverters. Two inverters lead to the set input, and the other two come from the clear and set outputs of the Z stage FF.

P Register (12 bits)

The Program Address register, P, holds the address of the current instruction and is a double-rank counter. At the beginning of each instruction, P is increased by one to provide the address for that instruction. If a jump instruction is called for, P is forced to the jump address. In addition to the manual set and clear inputs, P has one set input from the PSR register, and a forced input from A' and the three Tag registers to provide the jump address. The output from P goes directly to the S register.

A' Register (8 bits)

The results of all operations involving the adder are formed in the auxiliary arithmetic register, A'. The contents of A' move under program control to A, or P. The state of each stage of A' during adder operations is determined by one or more levels with the adder. Functions controlling A', labeled Clear, Toggle, and Probe, are explained in the Arithmetic Section.

S Register (12 bits)

The Storage Address register, S, holds the address for any ensuing storage reference for an operand, another address, or an instruction word. Each stage of S is a single FF with inputs from A, Z, P, BER, T_1, T_2, and T_3. Outputs go to the storage translators for implementing the actual address selection.

F Register (8 bits)

The Function register holds the upper 6 bits of the instruction word in its lower 6 bits throughout the execution of that instruction. The 2 bits of the Tag register reference are held in the upper 2-bit positions. The inputs to F come from

Z via the 1^3 inverters. Outputs from F go to the Function Translator. Outputs from F' go to the Tag register select circuits

PSR Register (12 bits)

The 12-bit P Storage register is temporary storage for the address at which the TeleProgrammer was interrupted. On recognition of an interrupt, information in P is forced into PSR. Each stage of the PSR consists of one FF which has inputs from each corresponding stage of the P register. Outputs from PSR go to corresponding stages in the P register.

T_1, T_2, T_3 Registers (4 bits)

The three 4-bit T registers hold the upper four bits of a 12-bit address. These are referenced or selected by the upper two bits of the instruction word. Inputs to the Tag registers are from an inverter selection matrix which is fed from the lower 4 bits of the A register. Manual inputs also come through this inverter matrix. Outputs from the Tag registers go to an inverter selection matrix that selects which Tag register content is to be transferred to the upper 4 bits of the P or S register.

O Register (6-bits)

The 6-bit O register is temporary storage for the upper 6 bits of a 12-bit External Function code. Inputs to O are from the lower 6 bits of the Z register. Outputs feed the line driver output cards.

BFR

The Buffer Data register holds the word of information being transferred to or from storage during a buffer operation.

Each stage of BFR consists of one FF. During a buffer input cycle, BFR receives data from the buffer input cable and sends an output to the IO-4 inverters for writing the input word into storage. For buffer output, BFR receives the buffer output word from the IO-2 inverters and puts it on the buffer output cable.

BER

The Buffer Entrance register holds the storage address to which, or from which, the current data word is to be transferred during a buffer input/output operation. Each BER stage has a forced input from the corresponding stage of the A register.

The contents of BER are incremented by 1 during each buffer cycle except the first, since for this case the first word address (FWA) has already been entered into BER by an ABR (X04) instruction. Outputs from BER go to a network for comparing the current buffer word address with the last word address held in the Buffer Exit register, and also to the S register.

BXR

The Buffer Exit register holds the terminating address (LWA + 1). Forced inputs to each single FF stage of BXR come from the A register during the ABX (X05) instruction which precedes each buffer I/O operation. The output from BXR goes to a network for comparing BER with BXR during each buffer cycle.

MASTER CLOCK

The master clock which provides the timing pulses used throughout the TP consists of four interconnected oscillators, each of which is contained on a type 02A card. Each oscillator operates at 3.0 megacycles and produces two sine waves, even and odd, 180 degrees out of phase with one another. Each oscillator card has two logic symbols. For example, C--0 could represent the even phase, and C--1, the odd phase. The odd phase is always the higher order symbol.

One pin on each of the four oscillator cards connects all even phases together. The odd phases are similarly connected, thereby assuring that the oscillators are synchronized. Five even and five odd output pins are available on each 02A card. Since these outputs draw some current, each oscillator is loaded as symmetrically as possible.

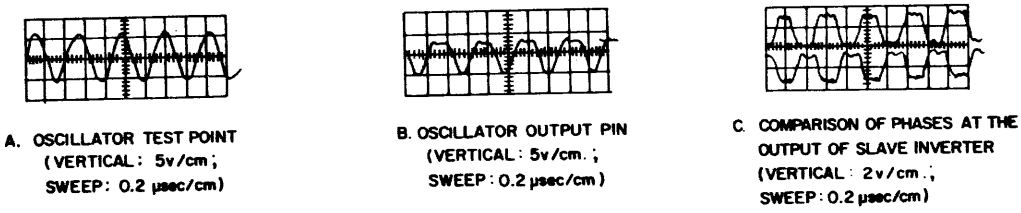


Figure 3-2. Master Clock Oscillator Waveforms

Figure 3-2 shows the waveforms of the oscillator card and the square wave produced after that wave passes through one inverter. The asymmetrical shape of the wave at the oscillator output pin is due to the bias used in clipping the sine wave peaks. Only the negative (top) portion of the wave is used in gating. Therefore, the clock pulses in the TeleProgrammer are 0.167 in duration.

Two oscillators drive a resynchronizing system which times asynchronous signals, such as those developed by manual switches and external equipment communication lines, to the TeleProgrammer. This system is discussed in the section dealing with input/output control.

MAIN TIMING CHAIN

The main timing chain is the ultimate source of control for all arithmetic, transfer and housekeeping functions. The timing chain is inactive during normal input/output operations since the external equipment is asynchronous with the computer. However, any operation using these devices must be initiated by the timing chain.

The chain consists of eight control delays (top of figure 3-3). For each input (start) pulse, the chain emits eight pulses (as evidenced by V00-outputs), spaced 0.167 usec apart. A recirculation control allows the V007 pulse to re-enter H000, thereby extending the number of timed pulses emitted. A string of 24 pulses (three passes through the timing chain) is produced for each start pulse. Such a train of pulses is called a storage reference cycle, and is the basic timing unit within the TeleProgrammer.

Timing Chain Excursions Counter

A record of the number of times a given pulse has traversed the timing chain is kept by the excursions counter. This is a two-stage, double-rank, reflected binary counter (figure 3-3 lower left). Excursions counter outputs go to the primary timing control.

Primary Timing Control

The primary timing control translators reflect the state of the excursions counter. The three passes of each storage reference cycle are decoded by the inverters in Figure 3-3, lower right). The passes are designated 1, 2, 3, early 2nd or early 3rd. The early quarter begins at time 5(V005) on the previous pass (figure 3-4). The part these cards play in TeleProgrammer timing is described more complete in the section on storage control.

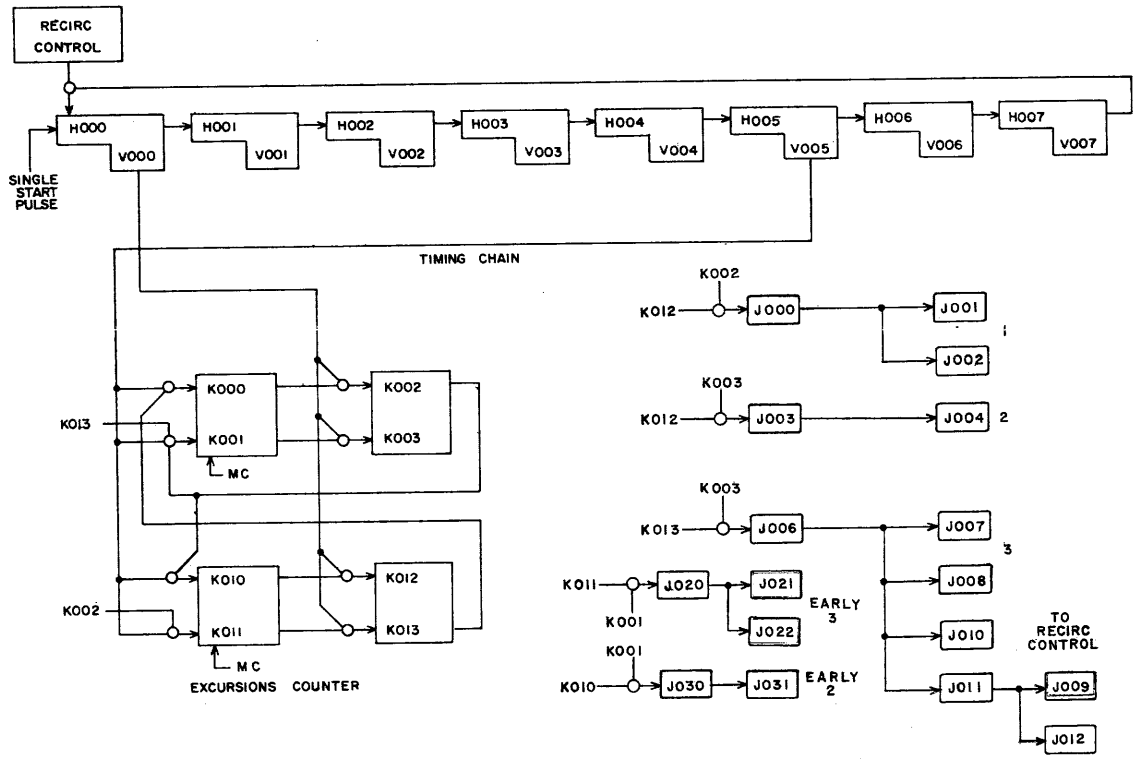


Figure 3-3. Main Timing Controls

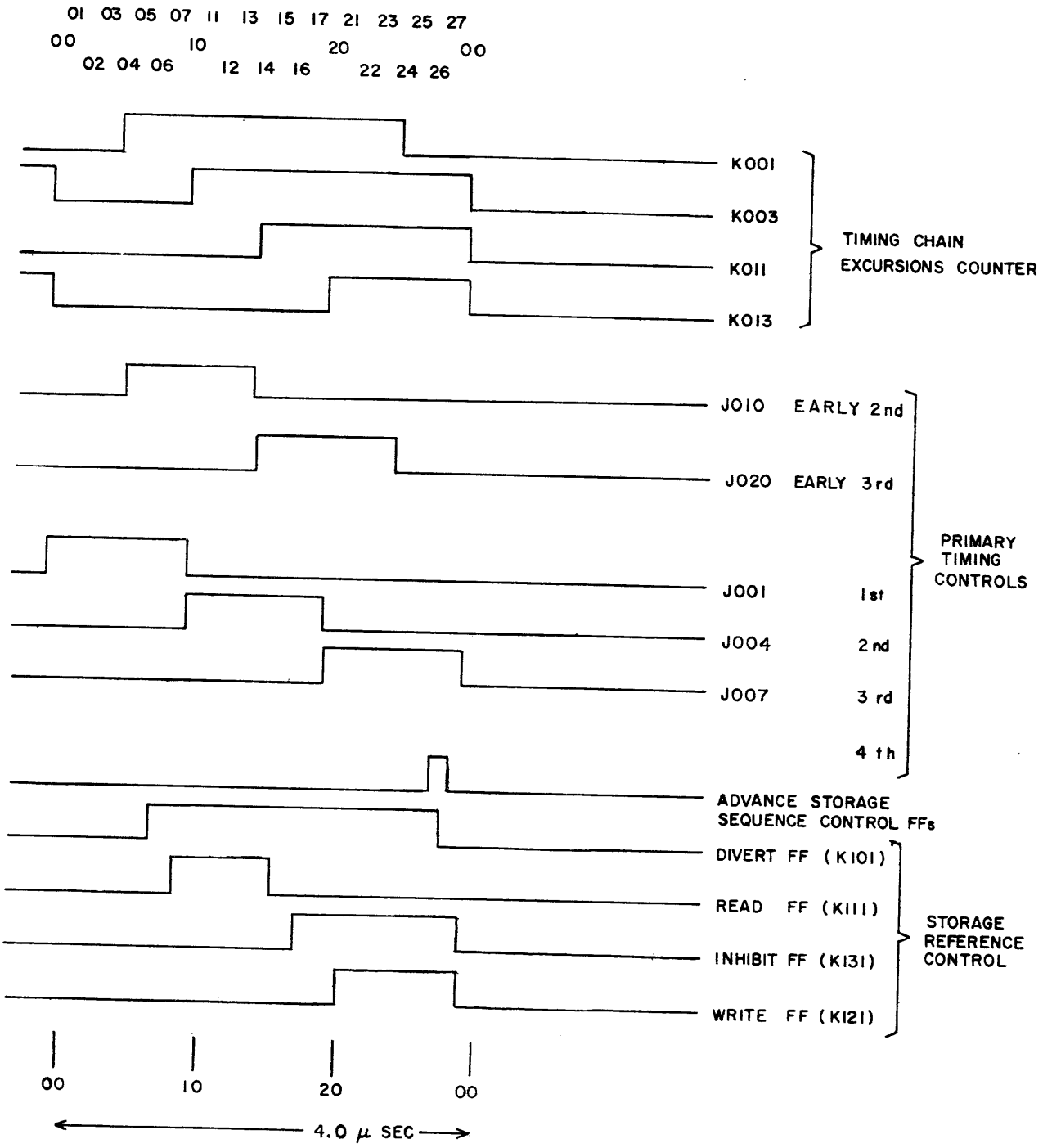


Figure 3-4. Main Timing

STORAGE REFERENCE CYCLE

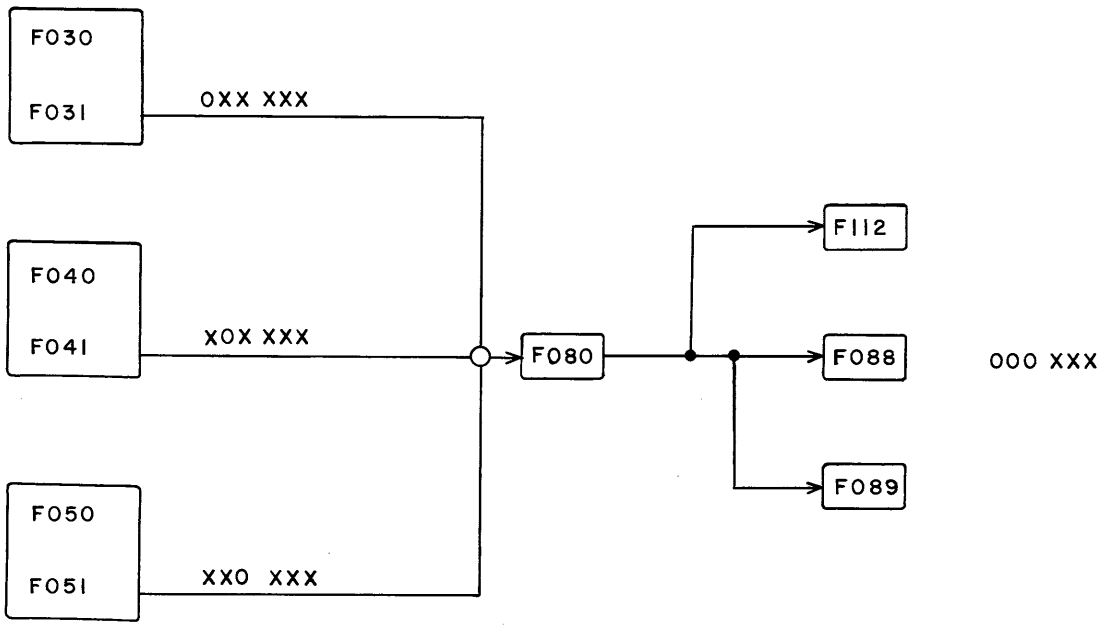
The storage reference (SR) cycle is the basic timing unit for the TeleProgrammer; all operations depend upon the execution of at least one such cycle.

The storage reference cycle consists of four phases: divert, read, write, inhibit (described in the storage section). At the bottom of figure 3-4 is shown the timing of the four Storage Reference Control FFs which regulate the duration of each of the SR phases. These FFs will be energized during each SR cycle, whether it be A, B, C or D. The decision as to which type of SR cycle is to be executed is the province of the Storage Sequence Control.

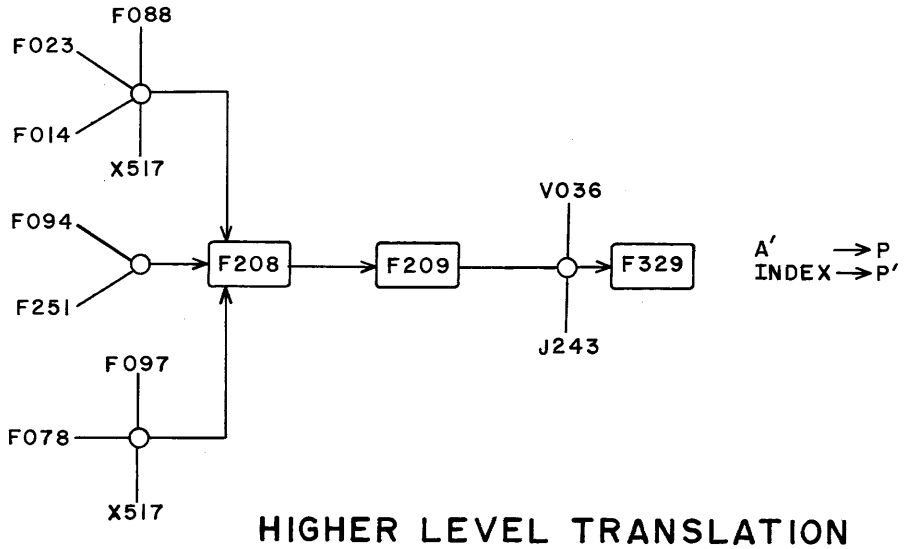
FUNCTION TRANSLATOR

The function translator performs several levels of translations upon the operation code held in the F register in order to determine what machine commands must be fulfilled during that instruction. A typical translation is shown in figure 3-5. Other translations may involve fewer elements.

The first-level translators consist of two 3-bit groups which translate the F register into octal notation. As an example consider the element F080. F080 has inputs from the upper 3-bit group of F030 through F051 and has a normal "1" output except when those three contain zeros. F080 is thus energized when a code 000XXX is in the F register (i.e., an octal 0X where X denotes insignificant digits). In other words, F080 translates 0X octal. From these first level translations, higher level translations are performed which, then combined with primary timing elements, control the actions and reactions of the TeleProgrammer to various instructions. An example of higher level translation



1ST LEVEL FUNCTION TRANSLATION



HIGHER LEVEL TRANSLATION

Figure 3-5. Typical Function Translation

is F208 and F209. Here if any one of the AND inputs is satisfied F208 will be energized and F209 will go to a "1", which in turn satisfied the AND to F329 thereby accomplishing the A'⟶P function.

F' Register

The F' register (F100/F101 and F110/F111) holds the tag bits which reference one of the three Tag registers. This is necessary to obtain a 12-bit address for addressing 4096 locations. An 8-bit word will address only 256 locations. If F' is "0" no Tag register will be referenced and the upper 4 bits of the address are "0"s. If F' is 1, 2, or 3, the corresponding Tag register will be referenced.

OPERATING CONTROLS

Three lever switches for controlling the TeleProgrammer are provided on the switch panel of the control console.

Run/Step Switch

The Run/Step switch (figure 3-6) starts the TeleProgrammer timing chain. When put in the locking RUN (up) position, the machine will run continuously. When the switch is moved to the momentary STEP (down) position, the Tele-Programmer executes one storage reference cycle and then stops. With the switch in the center, or neutral position the M card input to the Neutral FF is grounded, providing a "1" pulse to set the FF.

RUN — As soon as the Run Step switch is raised to RUN, the M card input to the Run FF is grounded. Since the Neutral FF is still set, the Run FF will be set with the first even synch pulse. This disables the AND input to the start card to produce a "1" output signal. Assuming that the load mode is not in effect, the timing chain is pulsed when the ensuing odd sync pulse occurs.

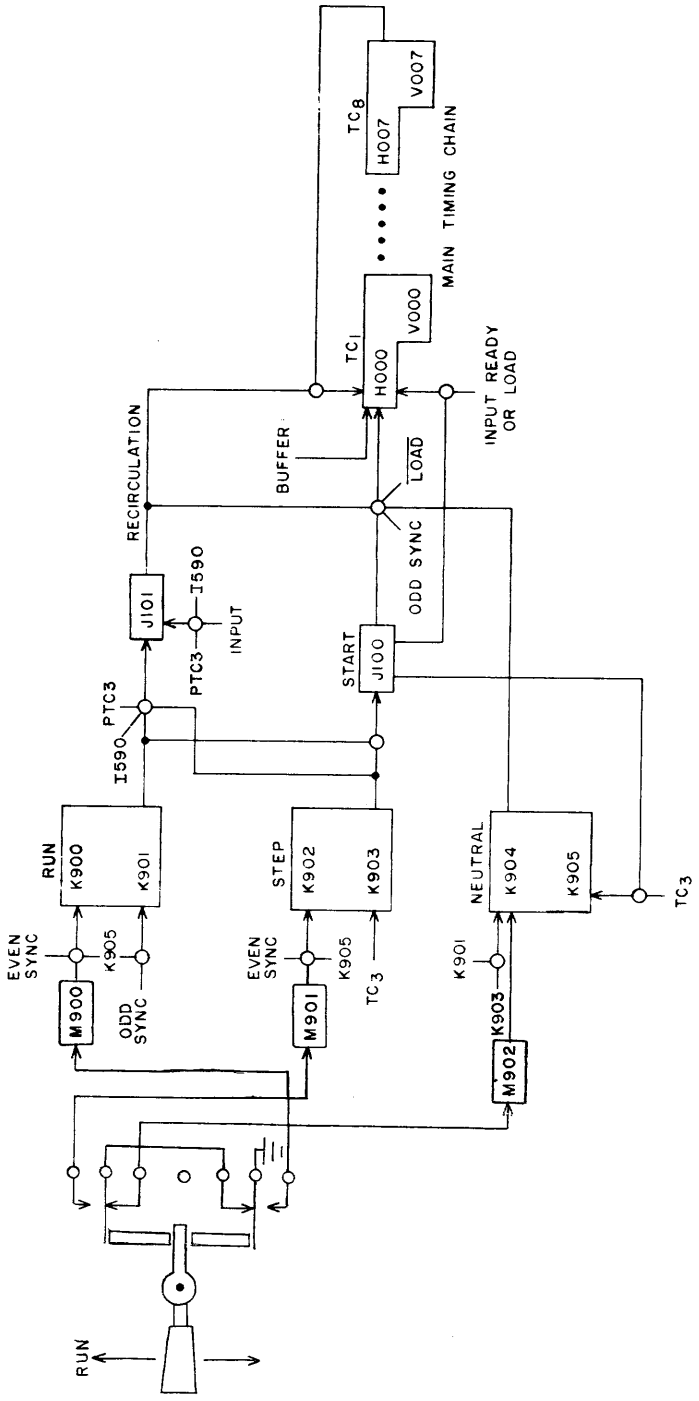


Figure 3-6. Run/Step Controls

To prevent double pulses to the timing chain, the Neutral FF is cleared by the output from third element in the chain (TC_3) which occurs 0.5 usec after initially starting the chain, or considerably before the succeeding odd sync pulse.

Recirculation in the timing chain is maintained by recirc (J101) which remains in the "1" state as long as the run FF is set Returning the Run/Step switch to neutral resets the Neutral FF permitting the Run FF to be cleared by the next odd sync pulse. This enables the left-hand AND input to the recirculation card, which is de-energized when the timing chain begins its third excursion. In this manner the chain is stopped upon the completion of the current storage reference cycle.

STEP — The initial conditions outlined for run pertain to this mode also. Putting the switch in STEP sets the Step FF, and the timing chain is actuated as explained above. TC_3 prevents double pulses by clearing both the Neutral and Step FFs. The left-hand input to recirc remains partially enabled in this mode, and the chain stops after three excursions (one storage reference cycle). One SR cycle is completed for each time the switch is set to STEP.

Load/Clear Switch

In the locking LOAD (up) position, this switch conditions the TeleProgrammer for loading from the paper tape reader, and paper tape is read as soon as the Run/Step switch is activated (figure 3-7). In the CLEAR (momentary down) position, a master clear signal is produced, provided that the TeleProgrammer is not in the run mode.

LOAD — When LOAD is in effect, the Not Load elements produce "0's", and the load elements are forced to a "1".

CLEAR — This position is effective only when the TeleProgrammer is not running. In addition to clearing the operational registers A, P, and Z two general master clear signals are provided (figure 3-7). One signal is used internally; the other is sent through an L card to external equipments.

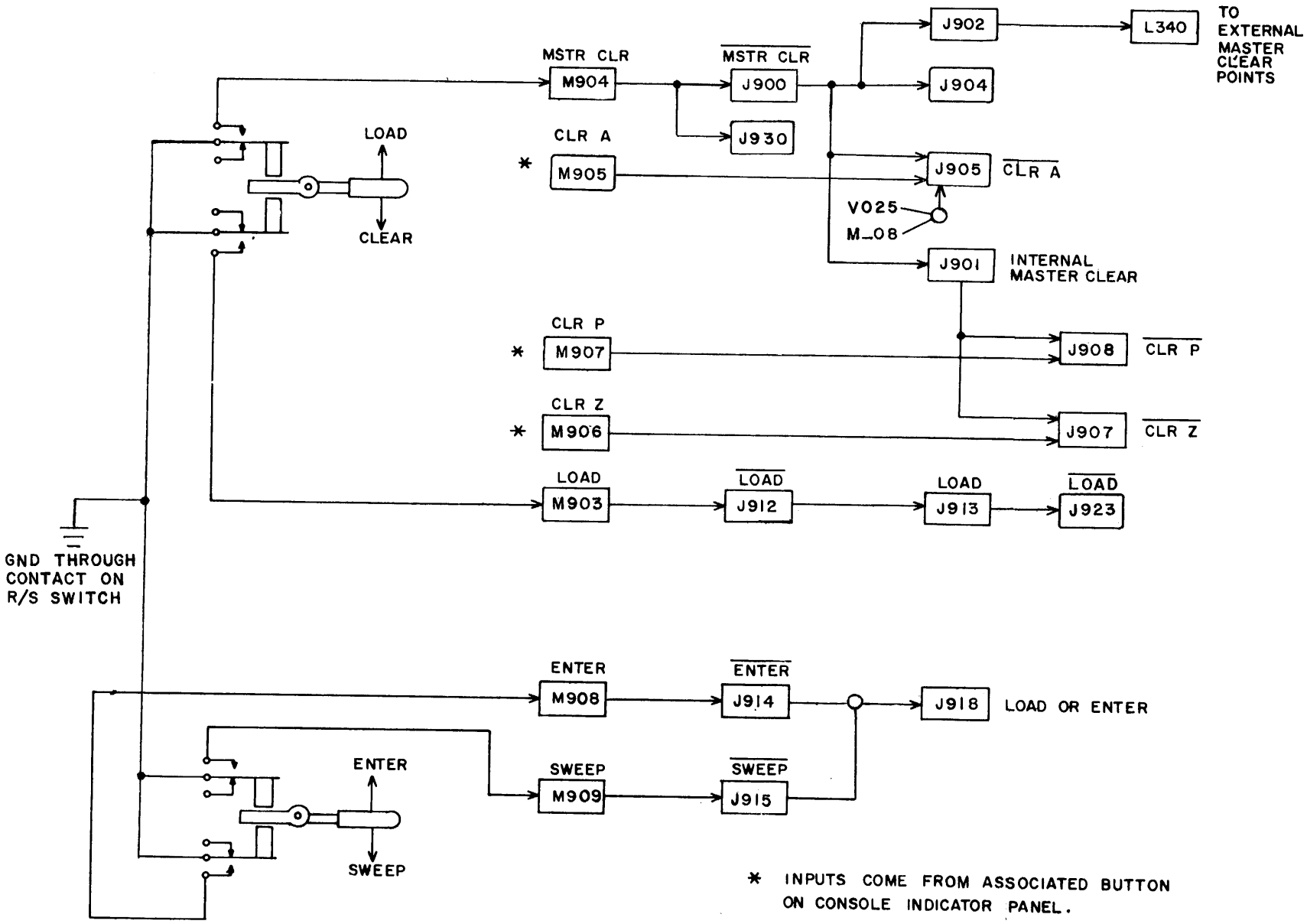


Figure 3-7. Load/Clear, Enter/Sweep Controls

Enter/Sweep Switch

Both ENTER (up) and SWEEP (down) are locking positions (figure 3-7).

ENTER — In this position, quantities may be manually inserted into the A register (usually preceded by a master clear). Putting the computer in either run or step loads the quantity at A into Z and storage at the address specified by P. Then A is cleared and P is advanced by 1 in preparation for the next manual entry into A. If entries are made without clearing P, the storage address of the entry will be one greater than that shown in P at the time of entering.

SWEEP — In this position, the entire contents of storage are examined sequentially by being transferred to Z. The position is operative in either run or step, but will only step once for each operation of the switch.

NON-VOLATILE STORAGE

The non-volatile storage feature prevents accidental destruction, or volatilization, of the contents of storage. Such a condition could occur with the decay of memory drive line current following a power interruption while the computer is running, or be due to a spurious pulse in the timing chain after power application. The logic and electrical circuits which are shown in figure 3-8 will detect a 3/4 volt change in the -20dc current.

If power fails, or if the main power switch should be turned off inadvertently, the computer will be stopped and MTF set. This FF disables the storage control circuits and assures that, when power is restored, no storage currents will flow until MTF has been cleared by a manual clear.

A second delay at the input to J215 operates upon application of power by holding that input to "0" for 5 ms, thereby ensuring the setting of MTF. Full d-c power is realized approximately 1 ms after power is turned on, but nearly 45 ms are required for the supply voltage to drop to zero.

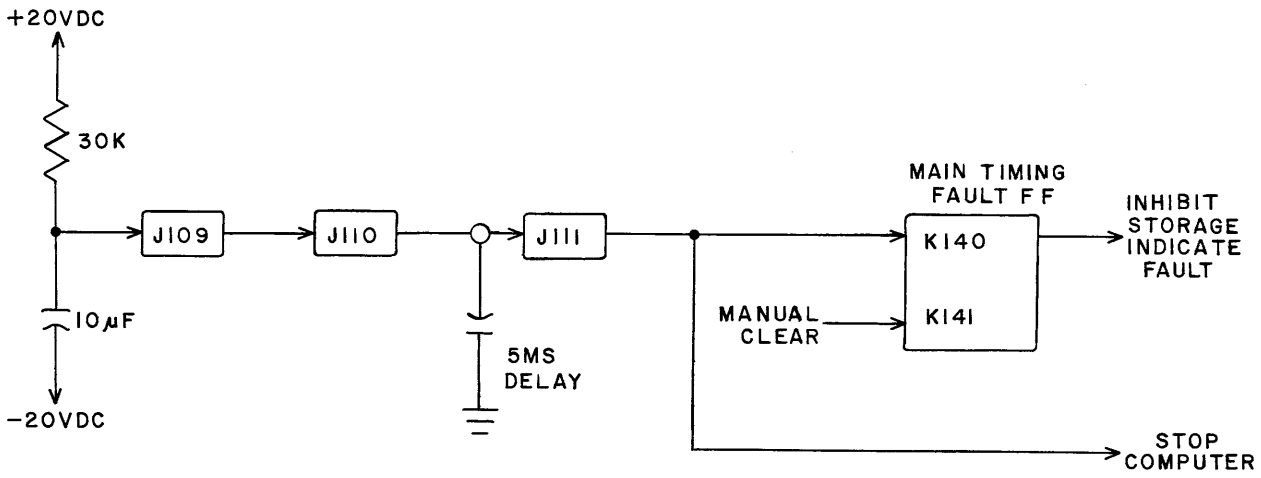


Figure 3-8. Non-Volatile Storage Circuits

STORAGE SECTION

The storage section consists of a 2048 or 4096-word, 8-bit memory employing magnetic toroids. The toroids, or cores, have an essentially square hysteresis curve. The memory (storage) cycle is 4.0 usec long. The read phase, which takes 0.83 usec, transfers the information from storage to the Storage register (Z) or the Buffer Data register (BFR). The information in the selected cores is destroyed, but is automatically restored by rewriting during the write phase, which is initiated 2.83 usec after the start of the cycle and lasts 1.8 usec.

In the 4096 memory, each bit of the 8-bit word is represented by a separate memory plane containing 4096 cores in a 64 x 64 array. The 8 planes are stacked to constitute a memory plane assembly.

Each core has four continuous wires running through it (figure 3-9). Two of the wires, intersecting the core at right angles to each other are used to select one of the cores in the array by a system of horizontal and vertical coordinates. In this manner, all 8 bits of one word (the 8 cores at the corresponding coordinate location in each of the memory planes) are addressed simultaneously.

The drive lines are connected to a bi-polar source (read/write drivers) which enables the lines to be driven in one direction for writing data into the core, and in the opposite direction for reversing the state of that core, thereby producing a pulse which is recognized during the read phase. The read pulse switches a core from the "1" state to the "0 " state, but does not affect cores already in the "0" state. The write pulse switches a core from "0" to "1".

The third wire, the inhibit line, is in parallel juxtaposition with the vertical lines and threads all cores in the corresponding bit-planes of both banks. Since the drive lines influence all cores in a selected storage location (address) simultaneously, it is necessary to negate the effect of those lines in any bit position which must remain in the "0" state. The 8 inhibit lines are connected to 8 inhibit generators which are controlled by the state of a corresponding bit position of Z or BFR. If, for example, Z contains a "0" in bit 04, the appropriate inhibit generator is energized. Current through the inhibit lines is approximately equal to that in the horizontal drive line, but opposite to it in polarity. As a result, the effect of the H and V write currents acting on the selected core in the inhibited bit-plane is not sufficient to change the state of that core from "0" to "1". (The current in the inhibit line, or the H or V line, is commonly called a half-current, because it is half the value needed to shift the state of a core from "0" to "1".)

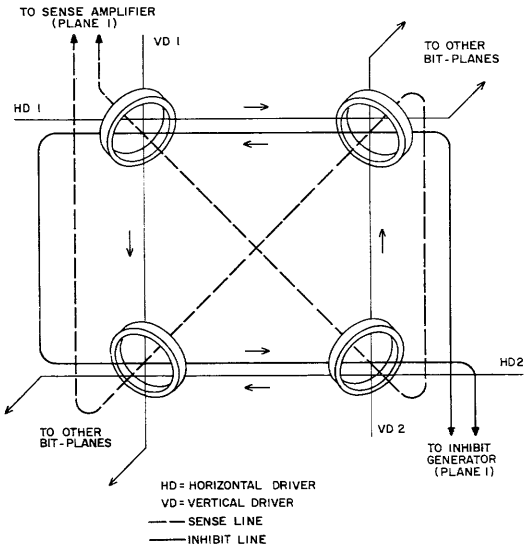


Figure 3-9. Four-Bit Magnetic Core Matrix

The fourth wire, intersecting all cores of corresponding bit-planes is the sense line. Its purpose is to pick up the pulse created when the core in that plane shifts from "1" to "0" during the read phase. The outputs from the sense windings transfer the word to Z or BFR, where it is held for restoring memory. Writing new information into storage is accomplished by blocking signals from the sense windings to Z or BFR. Then the write pulse will store the new word in memory instead of restoring that which was read out by the read pulse.

The H and V drive lines thread across the memory board horizontally and vertically, terminating in tabs at either side of the edge of the frame. A wire connected to a front tab on one edge of the frame terminates in a rear tab on the opposite edge. This feature allows close spacing of the wires. The end of each I (inhibit) wire terminates in a corner of the frame from which connections are made to the inhibit generator. Sense amplifier connections terminate in another corner of the frame.

The 8 memory planes comprising the memory module, or stack, are held together by bolts passing through the four corners of each. An aluminum plate at the back and front protect the stack from physical damage and hold the connectors which connect the stack into the main logic.

PROPERTIES OF THE MAGNETIC CORE

The magnetic properties of a core are represented by its hysteresis diagram (figure 3-10), which plots magnetic flux density (B) as a function of the field intensity (H). If current flow sufficient to cause a field intensity of $+H_m$ is

applied to the dirve lines, the flux density increases to saturation ($+B_s$). When current is removed, the flux density drops to the residual positive value ($+B_r$), the "0" state, and remains there. Another pulse of ($+H_m$) would merely shift the core to ($+B_s$) again; and after the pulse is removed, it would drop back to ($+B_r$). Application of current flow sufficient to cause a field intensity of $-H_m$ reverses the flux density to $-B_s$ and, when current is removed the flux density drops to the residual negative value ($-B_r$), the "1" state.

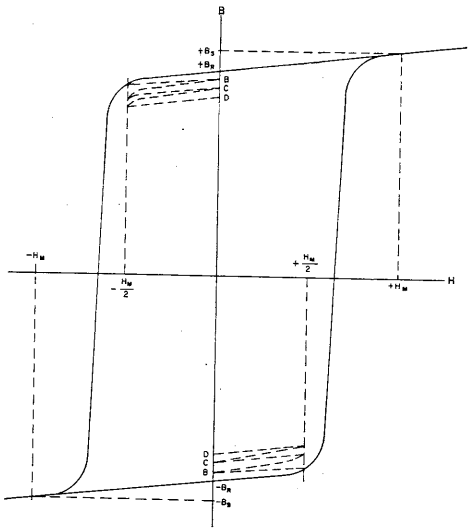
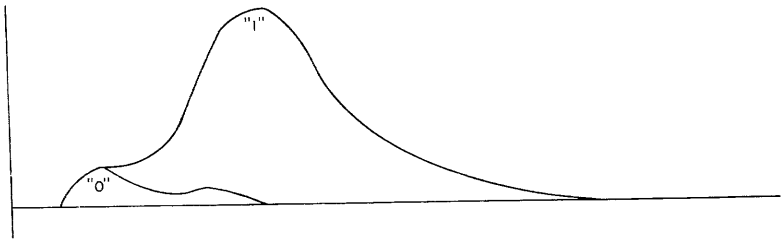


Figure 3-10. Typical Hysteresis Diagram

The basic memory cycle is composed of half-amplitude pulses capable of producing a field intensity of $H_m/2$. Since a half-amplitude pulse is insufficient to switch the core, the flux density returns to the residual value or slight lower after the pulse is removed. The coincidence of two half-amplitude pulses, one on the H drive line and the other on the V drive line of a core, produces a net field of H_m sufficient to switch the core. When a half-amplitude pulse drives

the flux density toward the knee of the hysteresis loop, the flux travels up or down the knee and then returns to a slightly lower residual value (B). Since the core is operating on a smaller loop, further half pulses reduce this remanent flux again and this effect soon reaches a limit, as at point D.

Any change in the magnetic state of a core causes a change in the total flux linking the core and winding to produce a voltage output on the sense winding (see drawing below). While H is applied, the voltage is sampled to see if the core switches. If a large voltage is sensed, the core was in the "1" state and has switched. If only a small voltage is sensed, the core was in the "0" state and has shifted from $+B_r$ to $+B_s$ and back again.



STORAGE CONTROL

The storage control system supervises the operation of each storage reference (SR) cycle and implements the selection of one, two, three or four storage references during each instruction execution.

Storage Reference Cycle

The fundamental purpose of the SR cycle is to select a word from memory and deliver it to the logic circuits of the TeleProgrammer. A detailed analysis of the circuits and logic appears under Address Selection.

The SR cycle consists of four phases: read, write, inhibit, and divert. These phases overlap one another during the 24-pulse cycle. During the read phase, two drive lines (one H and one V line) are energized in such a direction as to shift the selected toroids to the "0" state. During the write phase, the drive lines attempt to set each of the 8 selected cores to a "1", and do so, except where prevented by the inhibit line for that plane. Therefore, the inhibit phase must occur at the same time as the write phase. The divert operation which permits a final selection of two drive lines out of a possible 128 is necessary because the read/write drivers enable the H and V lines in groups of eight. Since the drive lines must remain selected throughout read and write, the divert phase encompasses both of these.

In the basic SR cycle (see next page), only operations common to all cycles appear. EK is the excursions counter. Specific operations occurring during cycles A, B, C, and D are shown in the Maintenance manual for each of the 42 instructions.

BASIC STORAGE REFERENCE CYCLE

00		Set EK_2 to 0	
01			
02	Clear S, Adv. P_1 by 1		
03			
04			
05	P ⟶ S	Set EK_1 to 1	
06	Set Divert FF		Divert
07			
10	Set Read FF	Set EK_2 to 1	Read
11			
12			
13			
14			
15	Clear Read FF	Set EK_1 to 2	
16	Set Inhibit FF		Inhibit
17			
20	Set Write FF	Set EK_2 to 2	Write
21			
22			
23			
24			
25		Set EK_1 to 0	
26	Clear Divert FF		
27	Clear Inhibit and Write FF		

Storage Reference Control

The three phases of a SR cycle are timed by storage reference control. Each phase is initiated (or terminated) by setting (or clearing) a corresponding FF (figure 3-11). Each set or clear input consists of an AND from primary timing control elements. In this manner, the exact pulse time for that operation is pinpointed. The timing of each phase may be understood by referring to the chart at the bottom of the figure.

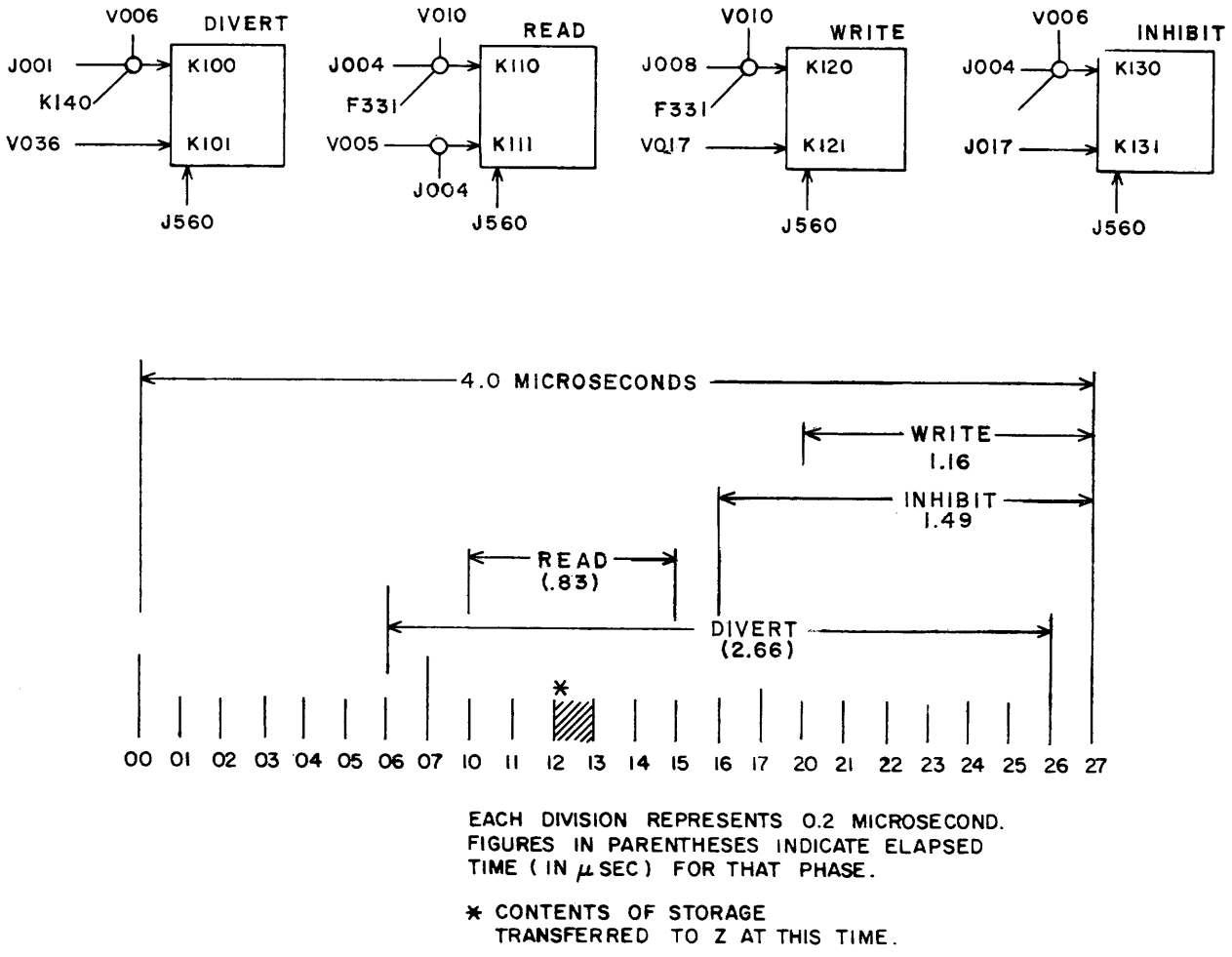


Figure 3-11. Storage Reference Control

Storage Sequence Control

If each of the 42 instructions in the computer repertoire is interpreted in terms of the different SR cycles necessary to complete the execution of that instruction, it is found that all instructions may be accounted for by arranging the four cycles in six patterns, called storage reference sequences, listed below.

D	
DC	No Address
DBC	Direct Address
DABC	Indirect Address
DBCC'	Replace Instruction, Direct Address
DCC'	External Function

The six sequences function in the following manner:

D	Perform the indicated operation using no operand a transfer or control instruction.
DC	Perform the indicated operation using the operand stored at address immediately following the address of the instruction.
DBC	Perform the indicated operation, using the operand stored at the address specified by the contents of the intermediate address immediately following the instruction and the Tag register referenced by the tag bits of the instruction.
DABC	Perform the indicated operation using the operand stored at address specified by an intermediate address immediately following instruction and the Tag register referenced by the Tag bits of the instruction.
DBCC'	Perform the indicated function, using the operand stored at the address specified as under the DBC sequence, then replace the operand with the results.
DCC'	Read out two halves of 12-bit external function code and place it on output lines. Upper 6 bits immediately follow instruction, lower 6 bits immediately follow upper 6 bits.

From these patterns it is possible to extract eight progressions.

1) D $\longrightarrow$ C

2) D $\longrightarrow$ D

3) D $\longrightarrow$ A

4) A $\longrightarrow$ B

5) B $\longrightarrow$ C

6) D $\longrightarrow$ B

7) C $\longrightarrow$ C'

8) C $\longrightarrow$ D

A ninth progression, (C $\longrightarrow$ B), is necessary when storing incoming data, and must be added to complete the list.

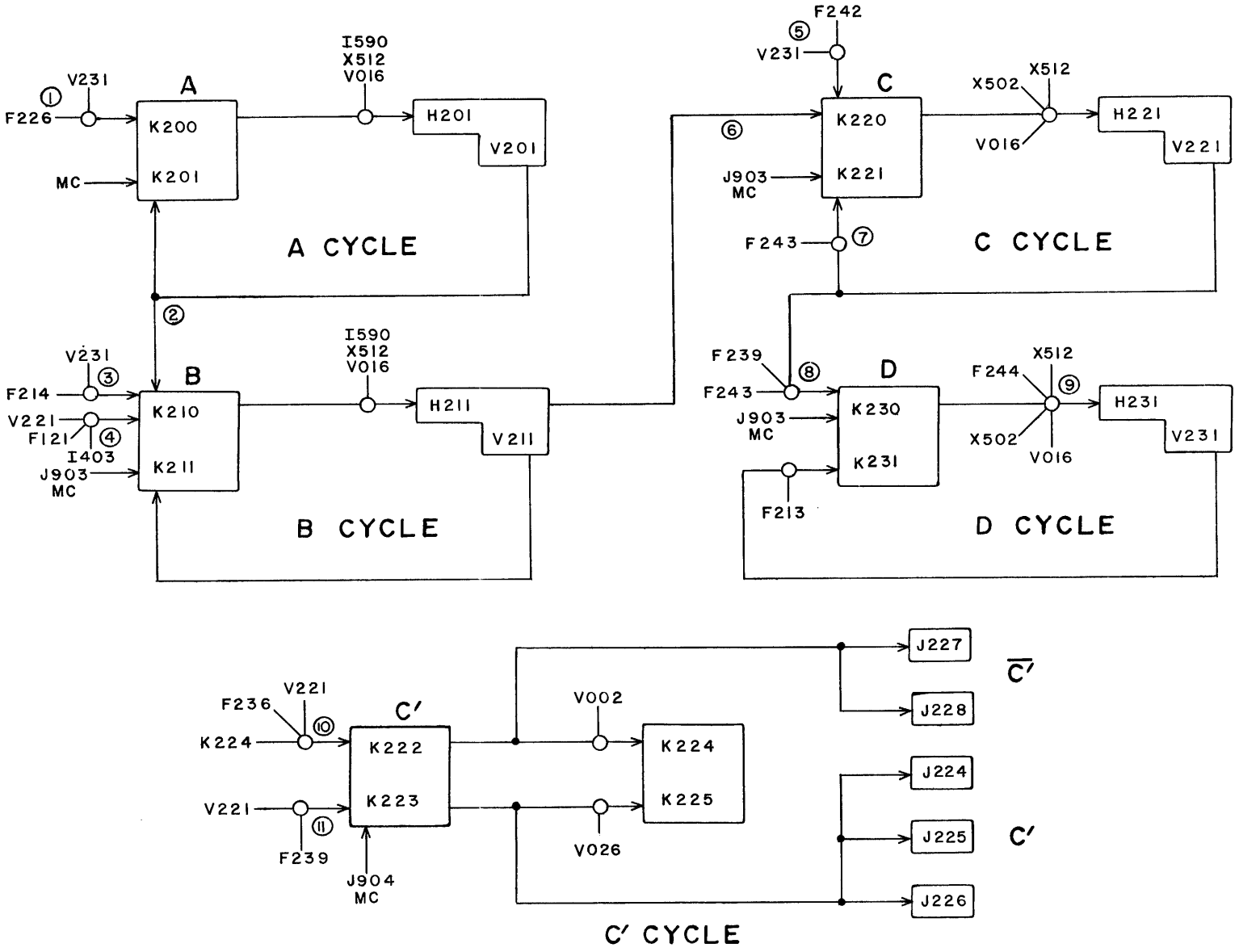


Figure 3-12. Storage Sequence Control

The Storage Sequence Control (figure 3-12) provides the method of selecting the proper progression, or series of progressions, for each instruction. Circled numbers on the drawing refer to inputs to the various sequences.

Input ① to K200/K201 (sequence A) controls the D ⟶ A progression. The two AND terms are:
V231, D sequence control delay
F226, indirect addressing mode

Input ② to K210/K211 (sequence B) controls the A ⟶ B progression and is an unconditional progression from A ⟶ B.

Input ③ to K210/K211 (sequence B) controls the D ⟶ B progressions.
V231, D sequence control delay
F220, direct addressing mode

Input ④ to K210/211 (sequence B) controls the C ⟶ B progression. This is used only on I/O instructions.
V221, C sequence control delay
F121, input or output instruction
I403, A ≠ 0 (i.e. starting addressing ≠ ending address)

Input ⑤ to K220/221 (sequence C) controls the D ⟶ C progression.
F242, no address operation
V231, D sequence control delay

Input ⑥ to K220 is an unconditional progression from B ⟶ C.

Input ⑦ to K221 (F243 V221) clears the C sequence FF except on instructions involving a C' cycle.

Input ⑧ to K230/K231 controls the C ⟶ D progression.
V221, C sequence control delay
F239, C' cycle on instruction involving C'
F243, and I/O sequence complete

Input ⑨ to H231 is made up of a 5 AND input which controls the D ⟶ D progression.

Input ⑩ to K222/K223 controls the C ⟶ C' progression. F236 is the controlling element and is up for instructions involving C'.

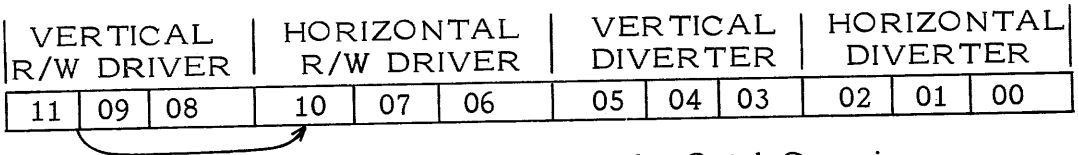
Input ⑪ to K223 clears the C' FF at the end of a C' cycle via F239.

ADDRESS SELECTION

In the 4096 word memory there are 4096 cores in each memory plane, and each core may be addressed by a discrete combination of one of the 64 horizontal and one of the 64 vertical drive lines. The 64 horizontal drive lines are connected, in groups of eight, to eight horizontal drivers; the 64 vertical drive lines are similarly connected. Current from the drivers may be directed (diverted) to one of the eight associated drive lines by means of eight diverter cards. The combination of eight drivers and eight diverters allows for a selection of any one of 64 drive lines, either horizontal or vertical.

Storage Address Register

The information for selecting one out of 4096 possible cores in each memory plane resides in the Storage Address register at the start of every SR cycle. As shown below, the 12 stages of the S group controls inputs to translators which select the vertical read/write driver, the next lower group provides horizontal R/W driver selection, and the two remaining octal groups provide selection for the vertical divertors and horizontal diverters.



Bit 10 is associated with bits 06 and 07 in the Octal Grouping.

Storage Translators

The storage translators interpret the contents of the four octal groups of S in order to provide inputs to the correct driver and diverter elements for each H and V selection. The storage translators will be discussed more fully under the Address Selection paragraphs dealing with the elements they affect.

R/W Driver Selection

Each bi-polar read/write driver is connected to eight drive lines. The direction of the current in these lines depends upon whether the associated driver is in the read or write state, as determined by storage reference control. Production of this current is discussed under Electronic Theory of Memory Circuits.

The method of selecting one of the eight R/W drivers is shown in figure 3-13. Note that the R/W drive selection for a given state of the three pertinent stages of the S register allows a total of three translators to be energized. Only two of these will be energized at any one time, however, since one translator solely depends upon the translation of S, whereas the other two are conditioned by the read and write phases of the SR cycle. It is the conditional translator which ultimately decides whether the current through the drive line shall produce a read or write pulse.

The current source cards provide d-c which the selected driver draws the read/write current. A current of 650 ma is needed to shift the state of the memory core. Each current source card provides 325 ma of this current. The half current of 325 ma needed for the H or V line is delivered to the winding by the transformer on the driver card.

The 64 H or V drive lines do not progress naturally from 00 to 77 (octal), but are interlaced after the pattern shown in figure 3-13. Note that an even-numbered driver has its eight drive lines interlaced with those from the successively higher odd-numbered driver. Thus, G0-0 is interlaced with G0-1, G0-2 with G0-3, G0-4 with G0-5, and G0-6 with G0-7. This does not affect the

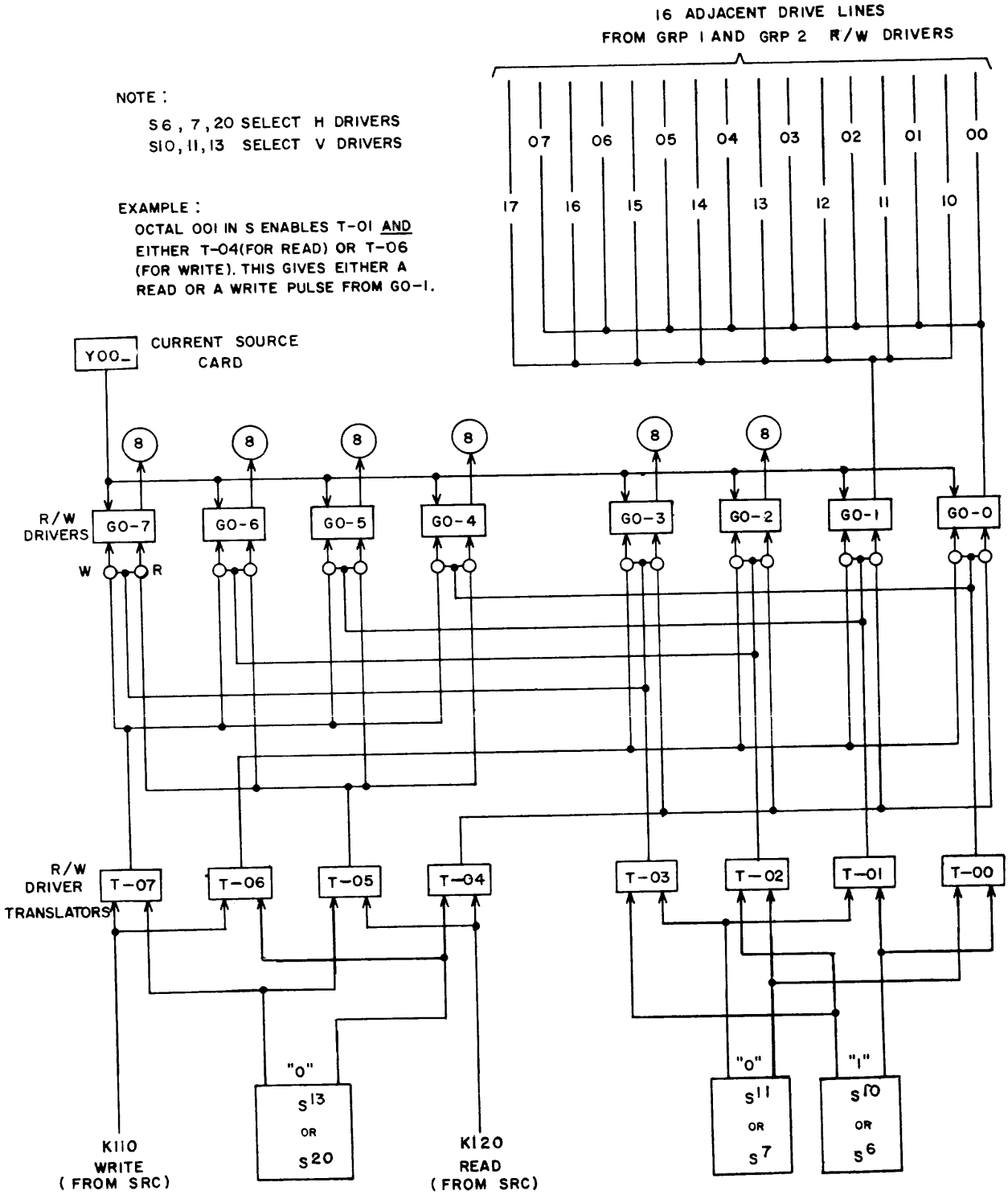


Figure 3-13. R/W Driver Selection

selection of any chosen core, but merely the physical location of that core within the matrix.

The T3 -- translators select the horizontal R/W drivers (G01-), while the T2 -- cards perform the vertical R/W driver selection (G00-).

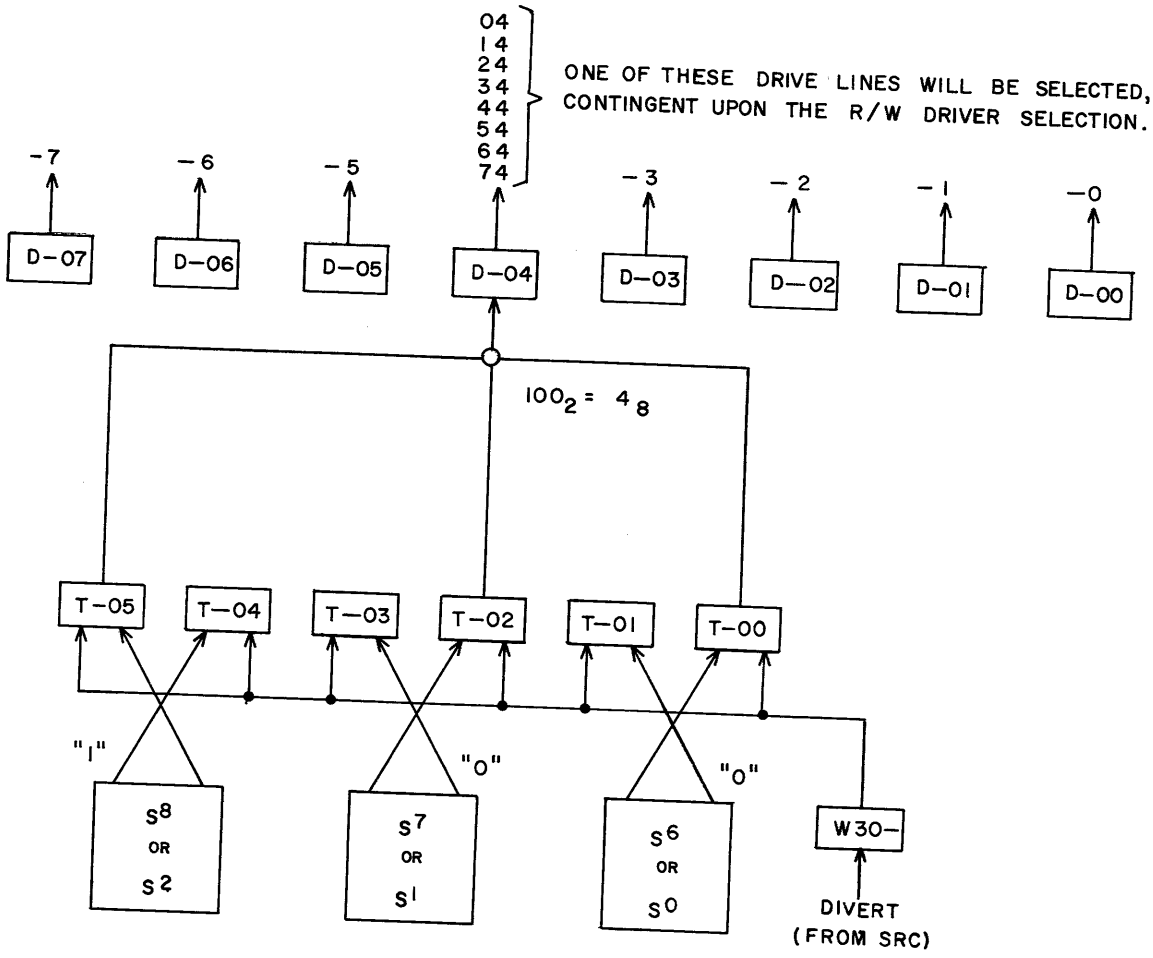


Figure 3-14. Diverter Selection

Diverter Selection

The diverters determine which of the eight drive lines is to carry the R/W current through the 8-bit planes. Stages S0, 1, 2 supply the information utilized by translators T100 through T105 to determine the horizontal drive

lines selected; S3, 4, 5 are used in conjunction with T000 through T005 to select the vertical drive line. The eight horizontal diverters (D10-) and the eight vertical diverters (D00-) are selected by a straightforward translation of the three bits of the S register which govern that particular set. Figure 3-14 shows the diverter selection. A translation choosing diverter four (D-04) is represented as an aid to understanding the octal translation.

Inhibit Driver Selection

An inhibit driver must be enabled if the selected core in the memory-plane associated with that driver is to remain in the "0" state; that is, if the core is not to be forced to a "1" by the write-current pulse (figure 3-15). The stage of 1--4 controlling the inhibit translator must be a "0" if the translator is to be energized, and also, the inhibit phase from storage reference control must be in effect.

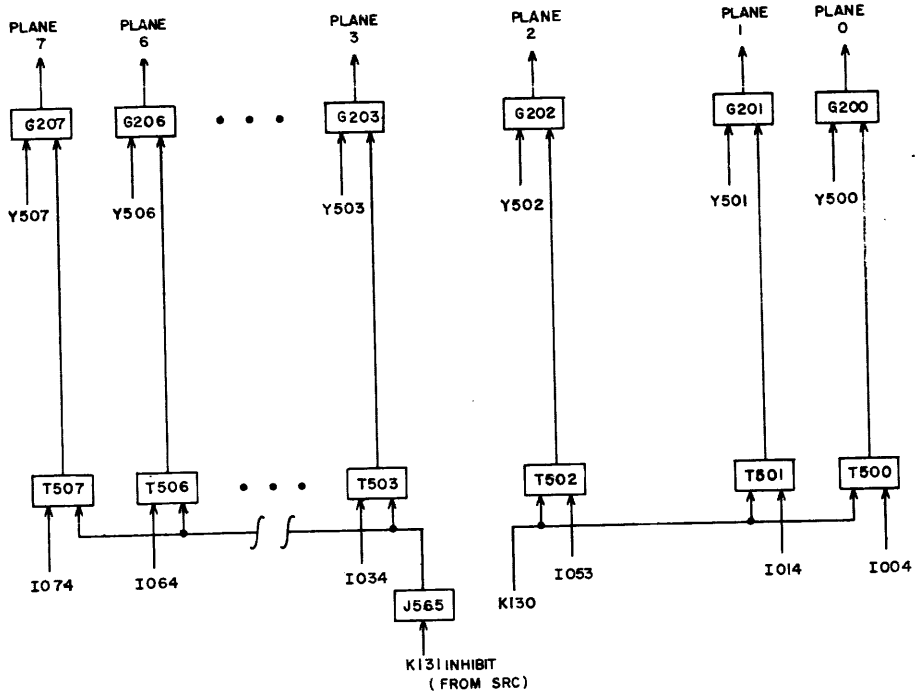


Figure 3-15. Inhibit Driver Selection

ELECTRONIC THEORY OF MEMORY CIRCUITS

The storage section performs non-logical functions such as amplification, pulse generation and switching. The card types are:

drive generator (type G10)	current source (type G13 and G14)
diverter (type 52)	inhibit generator (type G11)
selector (type 53)	sense amplifier (type G12)

The following paragraphs discuss the electronic theory of each of these card types and describe the interconnections of the cards to perform specific functions within the storage section. (Refer also to Equipment Diagram Volume.)

Drive Generator (Type G10)

The drive generator (figure 3-16) develops the R/W current which is applied to the selected H and V wires. Two identical channels feed opposite ends of the primary winding of transformer T01. Each channel consists of transistor Q01, connected as a common-emitter amplifier, and transistor Q02, connected as an emitter follower. The input signal is an AND combination of two selector outputs. A -0.5v input results in approximately 0v at the base of Q01, thus Q01 is cut off and approximately -20v appears at the base of Q02 which is also cut off. Consequently, no current flows in the primary of T01.

A -12v input signal causes Q01 to conduct, the conduction being held below saturation by feedback diode CR01. The negative voltage developed across R05 is applied to the bases of Q02, causing this transistor to conduct. Current flows from ground through the collector of Q02 to the emitter, through the primary of T01, to the current sources and back to -20v. The current pulse from the

secondary of T01, is applied to the H and V wires of the memory plane assembly. It then flows through the selected diverter card and back to the secondary of T01.

The polarity of the output current from T01 is determined by the direction of the current flow in the primary. The direction of current flow, in turn, is determined by the selected channel. For example, if channel A receives a -12v input, a read pulse is generated; a channel C input generates a write pulse. The CR03, R06 combination clamps the emitter so the voltage does not exceed 25v.

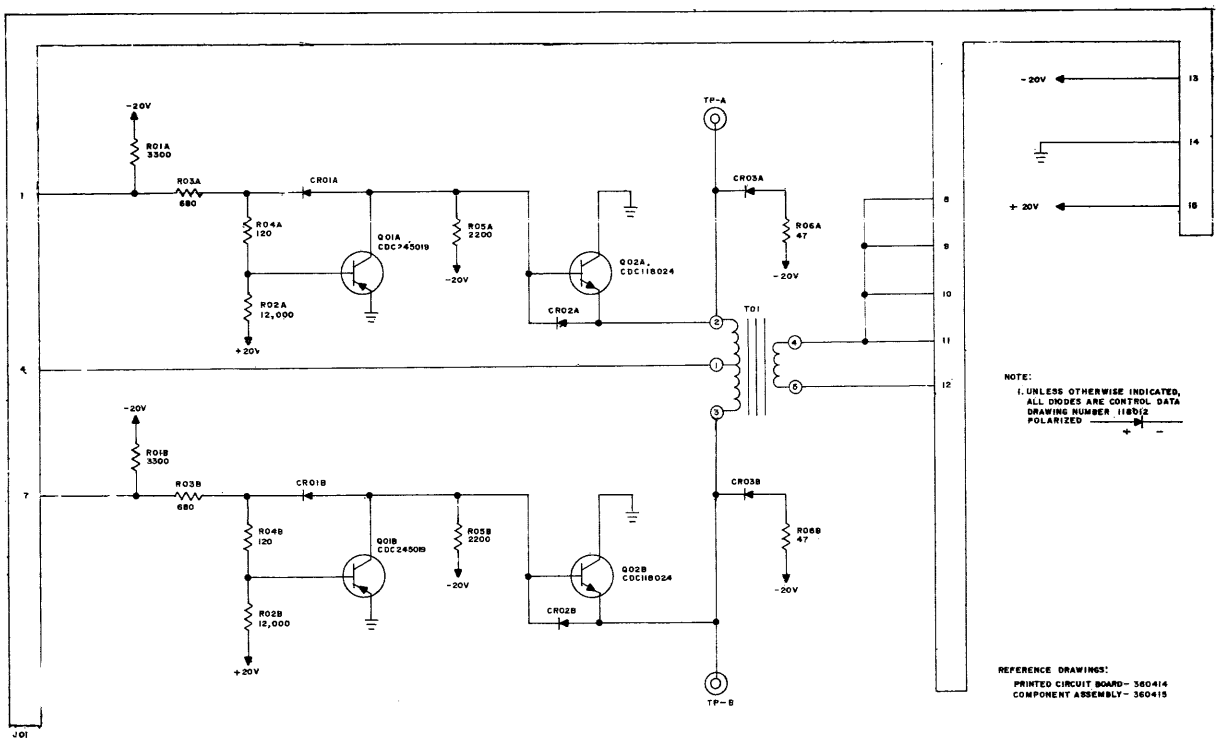


Figure 3-16. Drive Generator (G10)

Diverter (Type 52A)

The diverter circuit (figure 3-17) serves as an electronic switch for bi-polar current in series with an H or V wire of the memory plane assembly. The diverter consists of transistor Q01 (connected as an emitter-follower). Q02 and Q03. Either Q02 or Q03 passes the current pulse on the H or V wire to which the diverter is connected, depending on the polarity of that pulse.

A positive pulse passes one of the pairs of diodes CR03 and CR04, CR07 and CR08, CR11 and CR12, etc., depending upon the driver selection, and through Q02. A negative pulse passes through Q03 and one of the pairs of diodes CR01 and CR02, CR05 and CR06, etc. In either case, the current pulse is returned to the R/W driver through a common bus to which pin 12 is connected. The bleeder networks of all the diverters are connected in parallel via terminal 11. The diverter is selected through pin 10.

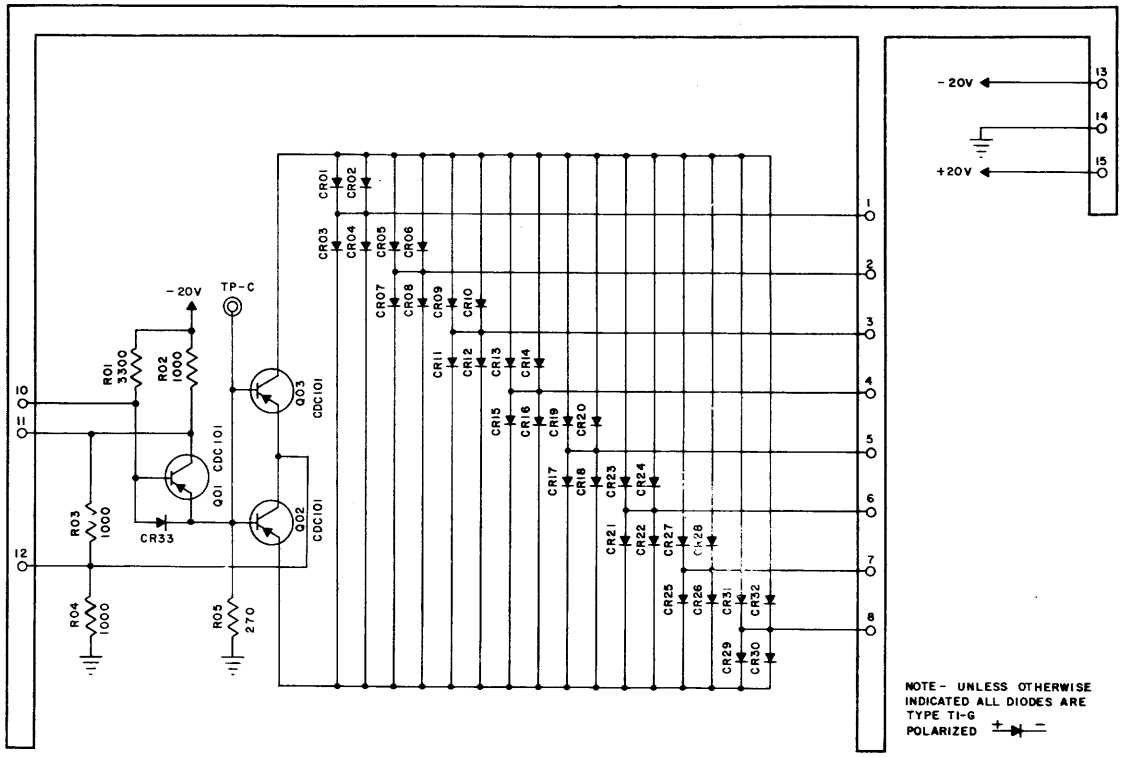


Figure 3-17. Diverter (52A)

Selector (Type 53)

The storage translators drive type G10, G11 and 52A cards. Each presents a different impendance level input than the standard logic card. For this reason, all storage translators employ the type 53 card. Each selector card consists of two identical selector circuits. A selector circuit (figure 3-18) is similar to the standard inverter except that the resistance results in output signal levels of -0.5v and -12v. Each selector circuit has two input diodes (CR01 and CR02) and four output diodes (CR09, CR10, CR11 and CR12).

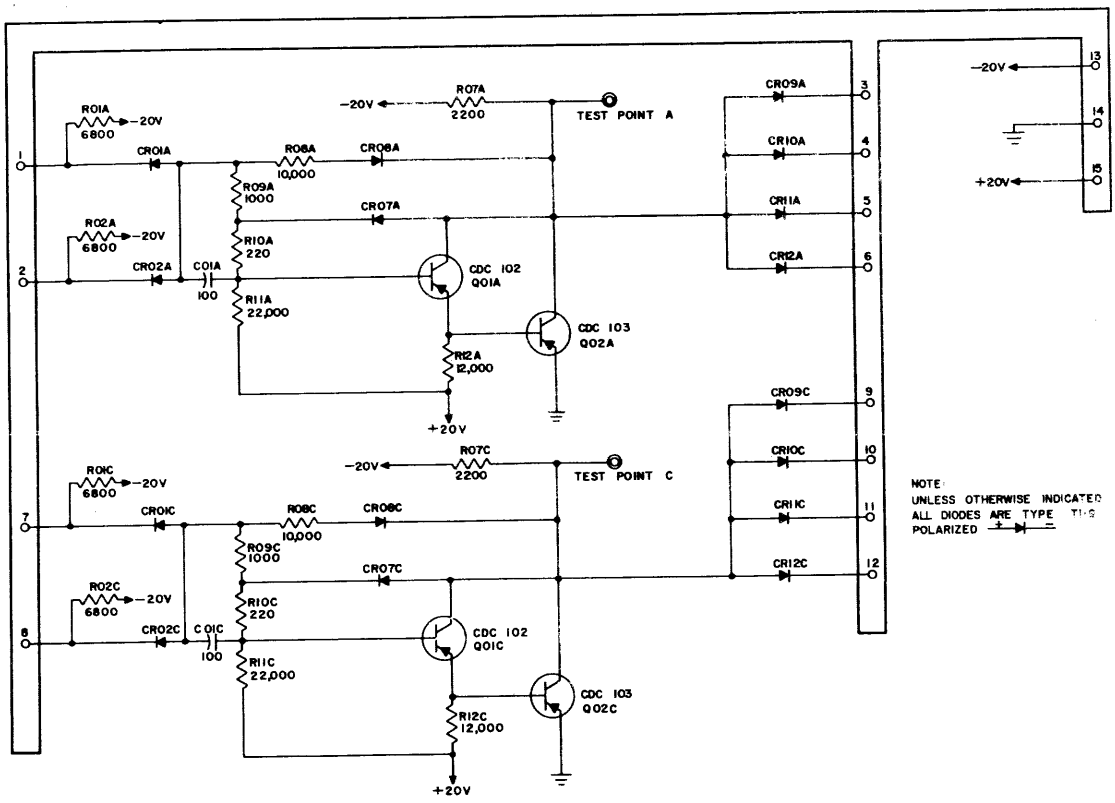


Figure 3-18. Selector (type 53)

Current Source (Type G12 and G13)

The current source cards (figure 3-19) consists of two banks of parallel resistors. The effective resistance of each bank is 49 ohms in the G12 and 64 ohms in the G13. One end of each bank is connected to the -20v output of the power supply. The G12 cards supply current to the H and V R/W drivers and the G13 cards supply current to the inhibit generators. The difference between G12 and G13 cards is the value of resistor which is 390 ohms on the G12 and 510 ohms on the G13.

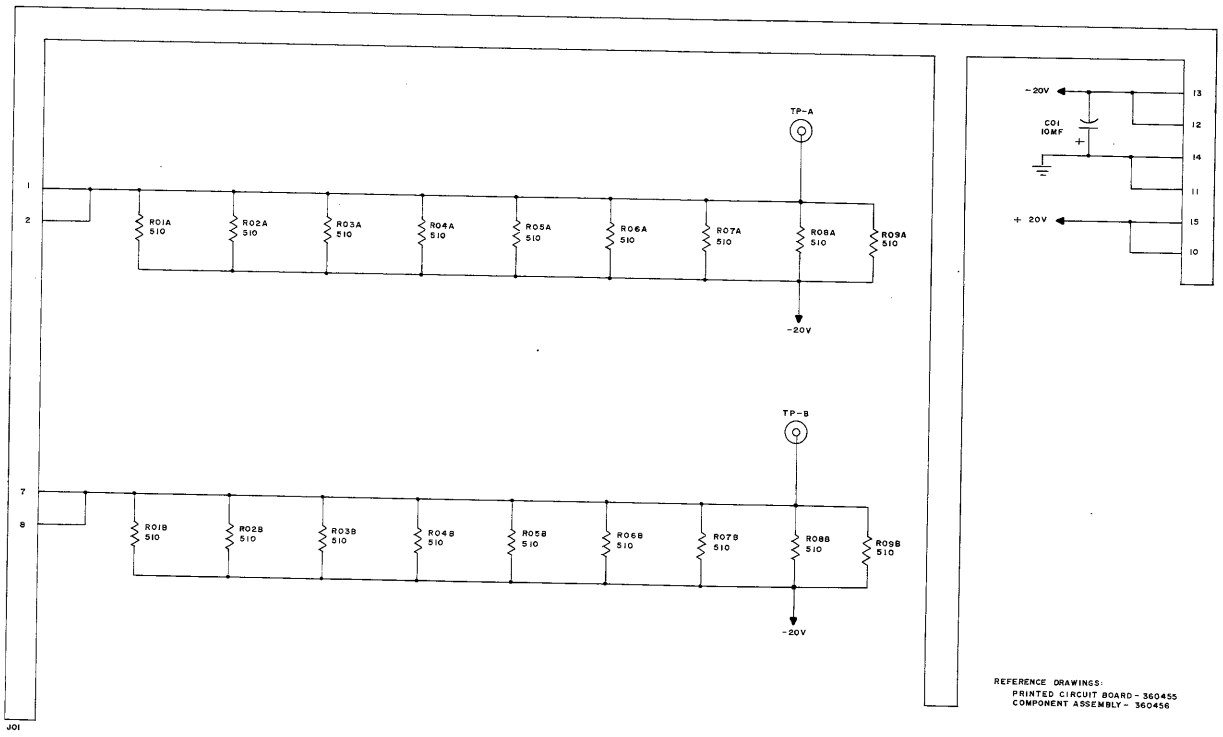


Figure 3-19. Current Source (Type G12 & G13)

Inhibit Generator (Type G11)

Each inhibit generator card (figure 3-20) has two generator circuits which are similar to the type G10 drive generator channels except for the absence of an output transformer. The output of each channel is independently connected to a terminal of the card.

A -12v input signal to either generator of a type G11 card causes Q01 to conduct and thus enable Q02. Current from the external source connects to the generator via terminal 6 or 12 and passes through Q02 to ground.

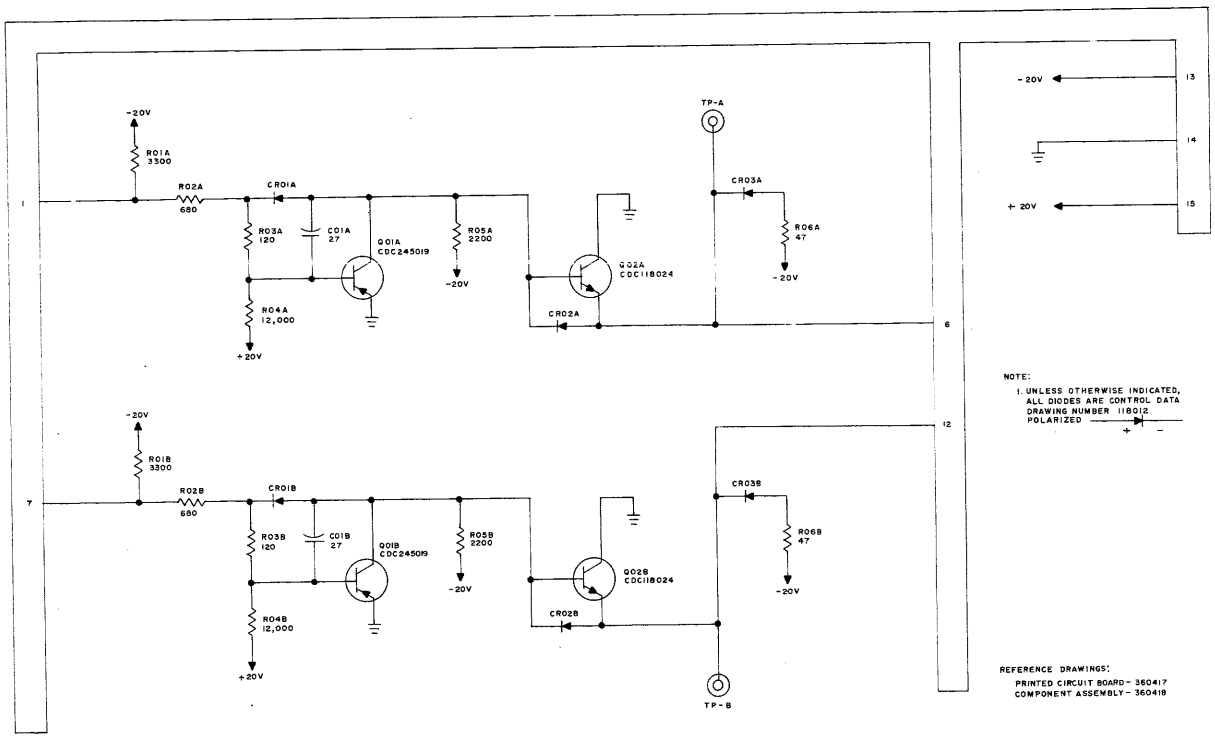


Figure 3-20. Inhibit Generator (G11)

Sense Amplifier (Type G14)

The sense amplifier (figure 3-21) amplifies the signal from a memory plane (pins 2 and 4) as the result of a read pulse. Transistors Q01 and Q02 form a differential amplifier which feeds T01. The secondary of T01 is connected to a bridge detector so that the signal polarity at Test Point C is always the same. The Q05 network is a standard inverter circuit providing normal output logic levels of -3.0v and -05v.

Gain from the differential amplifier for the common mode component of the input signal is about 2; for the differential mode component, across C01, gain is about 100. Pin 1 is connected to -20v. Pin 6 is connected to the diverter bus (-7.0v) to provide bias for input transistors Q01, Q02. The output from Q03, as the result of a "1" signal on the sense lines, is -0.5v.

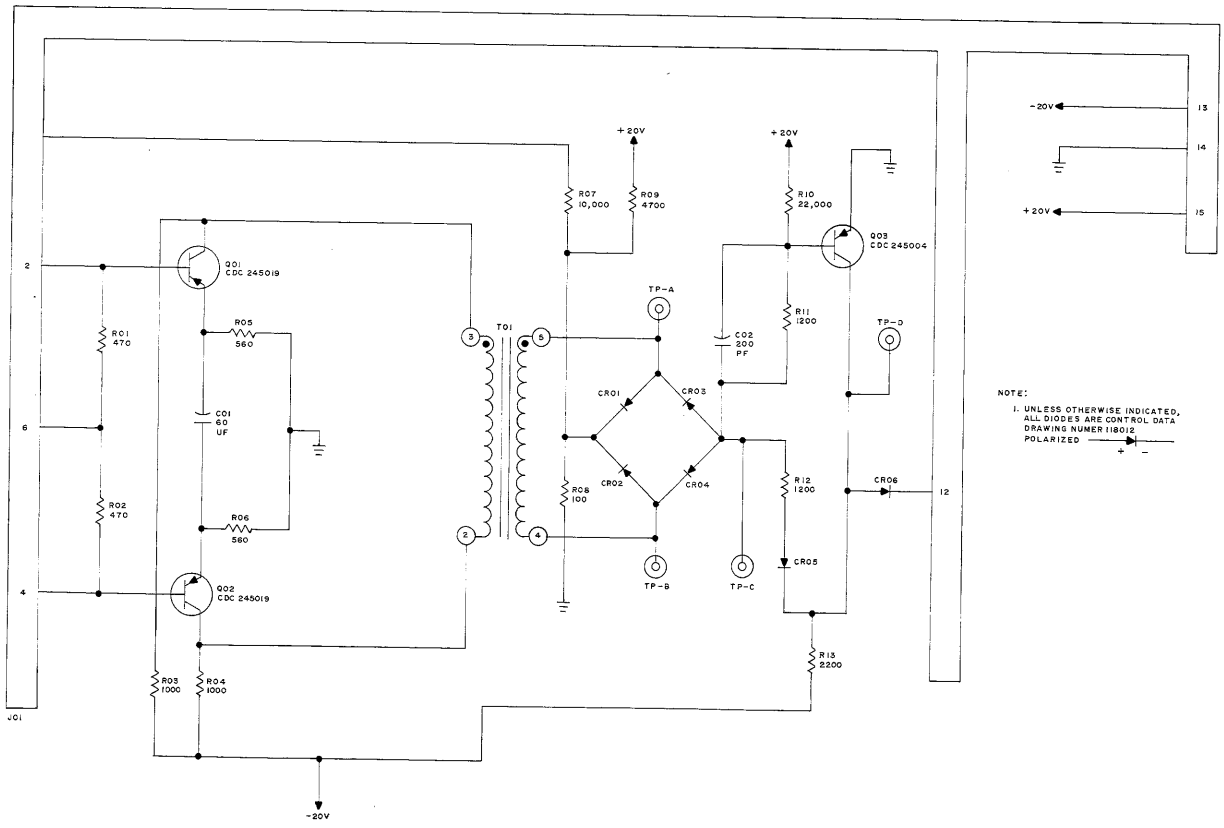


Figure 3-21. Sense Amplifier (G14)

ARITHMETIC SECTION

BINARY ARITHMETIC

The binary number system is the basis for the representation and manipulation of all information within the TeleProgrammer. Only two digits, "0" and "1", are used in this system. This characteristic is fundamental to computer operation since only two conditions (in this case, voltage levels) are necessary to encode data in binary form. A voltage of -3.0v represents a "1"; a voltage of -0.5v represents a "0".

A binary number uses the digit 2 as its basis of notation (radix) in the same manner that a decimal number uses 10. To illustrate, the decimal number 653 breaks down as follows:

$$6 \times 10^2 + 5 \times 10^1 + 3 \times 10^0 = 600 + 50 + 3$$

Similarly, a binary number such as 1011 can be analyzed:

$$1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 8 + 0 + 2 + 1$$

which is equivalent to decimal 11.

Binary numbers are added together according to the following rules:

$0 + 0 = 0$ $\qquad$ $1 + 0 = 1$

$0 + 1 = 1$ $\qquad$ $1 + 1 = 0$ with a carry of 1

The addition of two binary numbers proceeds as follows (the decimal equivalents verify the result):

Augend	0111	(7)
Addend	+0100	+(4)
Partial Sum	0011	
Carry	1	
	1011	(11)

TABLE 3-1. DECIMAL AND BINARY EQUIVALENTS

Decimal	Binary	Decimal	Binary
0	00000	8	01000
1	00001	9	01001
2	00010	10	01010
3	00011	11	01011
4	00100	12	01100
5	00101	13	01101
6	00110	14	01110
7	00111	15	01111
		16	10000

Subtraction may be performed as an addition. These decimal examples illustrate this method:

```
 8 (minuend)        or      8 (minuend)
-6 (subtrahend)            +4 (ten's complement of subtrahend)
 2                          2 (difference - omit carry)
```

The second method shows subtraction performed by the "adding the complement" method. This process is proved in the following identities:

$$8 - 6 = 2$$
$$8 + (10-6) - 10 = 2$$
$$8 + 4 - 10 = 2$$

The omission of the carry in the illustration has the effect of reducing the result by 10.

The method of complementing a binary number, known as the one's complement, is formed by subtracting each bit of the number from 1. For example:

```
 1111
-1001  (9)
 0110  (one's complement of 9)
```

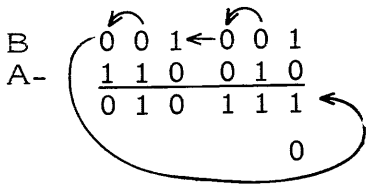
The one's complement of a binary number may also be formed by substituting "1's" for "0's" and "0's" for "1's" in the number.

ARITHMETIC OPERATIONS

Arithmetic in the computer is accomplished through a subtractive adder. That is, it adds A to B by subtracting the complement of A from B. For example, adding the binary numbers

```
B    0 0 1 0 0 1  (9 decimal, 11 octal)
A+   0 0 1 1 0 1  (13 decimal, 15 octal)
     0 1 0 1 1 0  (22 decimal, 26 octal)
```

is actually accomplished by subtracting, as shown below.



The long arrow indicates that an end-around borrow must be made from the first stage, since the last stage was unable to satisfy the borrow made when 1 was subtracted from 0. The final total is

$0\ 1\ 0\ 1\ 1\ 0$ (22_{10}) or $26_8)$

In the above example, the curved arrows indicate that the borrows must be made from the next higher-order stage. The arrow between the two octal groups represents the existence of a group borrow, where the last stage in a group cannot satisfy a borrow made upon it (or within that stage itself), and so

must propagate the borrow to the next higher group.

Subtraction is performed by subtracting the true value of the subtrahend (not its complement) from the minuend, as shown below.

minuend	0 1 0 1 1 0	(22 decimal, 26 octal)
subtrahend	0 0 1 0 1 0	(10 decimal, 12 octal)
	0 0 1 1 0 0	(12 decimal, 14 octal)

The curved arrow again indicates a stage borrow.

THEORY OF THE ADDER

The adder is a device used to generate and recognize the borrow signals. These are pyramided from each stage level to the octal group level and back to the stage level so the final state of each bit in the answer can be decided.

Shown below are the four possible states of a 1-bit minuend and subtrahend. A bit-by-bit subtraction results in the remainders indicated.

minuend	1 1 0 0
subtrahend	1 0 1 0
difference	0 1 1 0

Note that in the bit-by-bit subtraction, only one case in four finds the minuend unable to satisfy the subtrahend. This is where the minuend equals 0 and the subtrahend equals 1. Thus, it may be stated that whenever 1 is subtracted from 0, a borrow is generated, and this borrow must be propagated to the next higher-order stage.

Two cases produce a remainder of 0. This means that, after the initial bit-by-bit subtraction, that stage will be unable to satisfy a borrow made upon it by some lower-order stage, and must pass the borrow on to the next higher-order position. These cases exist whenever 1 is subtracted from 1, or when 0 is

subtracted from 0. The act of transferring an unsatisfied borrow signal to the next higher stage is termed an "enable". Thus it may be stated that an enable is generated whenever the bit in the minuend and the bit in the subtrahend are the same. In the remaining case, where 0 is subtracted from 1, the minuend is able to satisfy the subtrahend and also a borrow, if necessary.

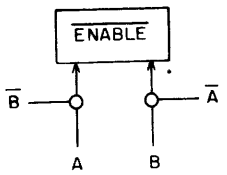
Stage-Generated Signals

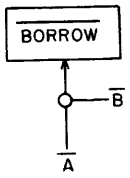
The above rules have been concerned with the subtrahend, which is the complement of the number actually involved in the arithmetic process. Building these rules around the addend, they become:

1) When corresponding bits in the two operands are both zero, a borrow is generated.

2) When corresponding bits in the two operands are unlike in value, an enable is generated.

These functions are illustrated below. Note that the quantities A and B may have two values (in binary notation), either 1 or 0. It is convenient to represent the presence of a quantity, e.g., the quantity B, by the symbol "B", and to show that the quantity does not exist (is 0) by the symbol $\overline{B}$. The latter is the Boolean algebraic expression meaning, in fact, NOT B.

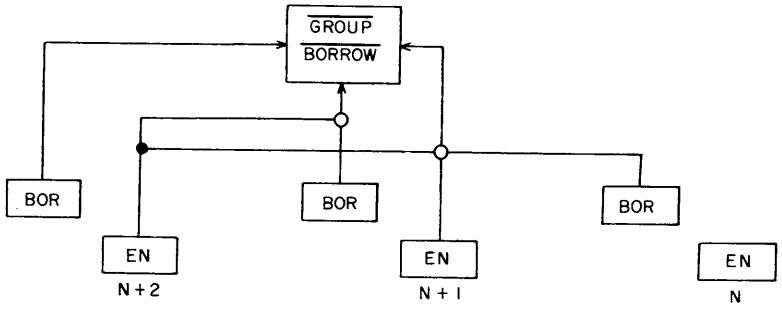




The small circle at the juncture of the symbol leaders and the arrow shaft indicates the logical AND function, which will be a 0 whenever either of the terms shown is a 0.

Group Borrow Signal

By definition, a group borrow signal is generated whenever higher-order stages (if any) in an octal group cannot satisfy a borrow generated within that group, and must pass the borrow signal on to the next group. The conditions creating a group borrow are shown.

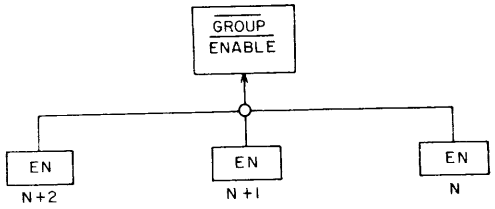


The leftmost input to the group borrow block is satisfied when the highest-order stage in that group (N + 2) generates a borrow signal. The middle input is completed whenever the second stage (N + 1) generates a borrow and the last stage has an enable signal present. The right-hand input shows that a borrow from the lowest stage (N) will produce a group borrow only when it is accompanied by enable signals in both higher-order stages. A general rule covers these cases:

> A group borrow is generated whenever a stage borrow is accompanied by a stage enables in the higher-order positions (if any).

Group Enable Signal

If each and every stage in a group contains an enable signal, a group enable is generated as shown below.



The rules presented earlier apply, regardless of the number of stages or groups involved, for a subtractive adder. The three-stage (octal) group is suggested because it preserves the identity of the octal group associated with the binary number system.

ADDER OPERATION

Three signals are used in conjunction with each adder operation to set the proper value for the final total into the A' register FFs. The first of these signals, CLEAR, sets every stage of A' to "0". The second signal, TOGGLE, sets a stage to "1" if an enable exists in that stage due to R and Q being unlike. The third signal, PROBE, may clear a toggled stage, or set an initially cleared stage, or leave a stage in its latest state, depending upon certain conditions existing within the adder. The logic involved in these three signals can best be appreciated by some concrete examples of addition.

In the following examples, all stages are labeled as generating enables (E), borrows (B), or no enables (NE). The borrow condition is really a no enable case, but B is used for distinction. The prime question to decide in resolving

the final answer, due to the necessity of initially clearing each stage of A', is this: How does the final answer in the A' register differ from all "0's" in that register? Stages which contain "1" are marked by an asterisk.

Example 1

B	B	E	B	NE	NE	B	E		
0	0	0	0	1	1	0	0	012	(014_8)
0	0	1	0	1	1	0	1	045	(055_8)
0	0	1	1	1	0	0	1	67	(071_8)
		*	*	*			*		

Example 2

B	E	E	E	E	E	E	NE		
0	0	1	0	1	0	1	1	043	(053_8)
0	1	0	1	0	1	0	1	085	(125_8)
1	0	0	0	0	0	0	0	128	(200_8)
*									

If each stage indicated by an asterisk in the above examples is examined to determine what conditions originally existed in that stage and also, what conditions act upon that stage, two general rules may be evolved:

1) A stage generating an enable signal is left in the "0" state unless acted upon by a borrow (B) or a passed borrow (PB); in which case the stage is set to "1".

2) A stage not generating an enable signal is set to "1" unless acted upon by a borrow or passed borrow.

The term "passed borrow" is used to express the condition where a borrow is not generated in an adjacent lower-order stage, but is propagated from some remote lower-order stage as the result of one or more intervening enable positions.

Rule 1 means that some provision must be made for complementing all stages which generate enable signals, since those stages were set to "1" by the toggle function, unless they are acted upon by a B or PB. Rule 2 means that all stages generating NE signals must be set to "1" (or complemented, since they are still in the cleared state) unless acted upon by a B or PB. A third all-inclusive rule is apparent:

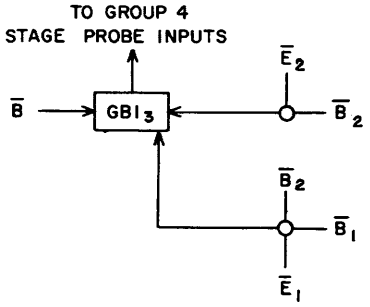
3) During the Probe function, all stages are complemented unless acted upon by a borrow or passed borrow signal.

It is now merely necessary to determine whether or not a particular stage has a B or PB signal acting upon it. This fact is determined by the group borrow input and stage probe input cards.

Group Borrow Input

A group borrow input card is associated with each of the three 3-stage octal groups constituting the adder. The philosophy of the group borrow was examined earlier, and is the same here. Once again, the group borrow input is realized if a borrow is generated in an adjacent lower-order group, or in some remote group if accompanied by intervening group enable signals.

Because of the inverter logic used in the adder, if an enable or borrow signal (group) exists, outputs from the group enable and group borrow cards are "0". To implement this logic, the following device is used.



An output to the group stage probe inputs will be "1" only when all GBI_3 inputs are down. Examination of the four situations which prompt a borrow from group 3 reveals that in every case, at least one of the terms causing the borrow is present at each of the inputs to GBI3:

Borrow from group 2, identified by $\overline{B}_2$, $\overline{B}$

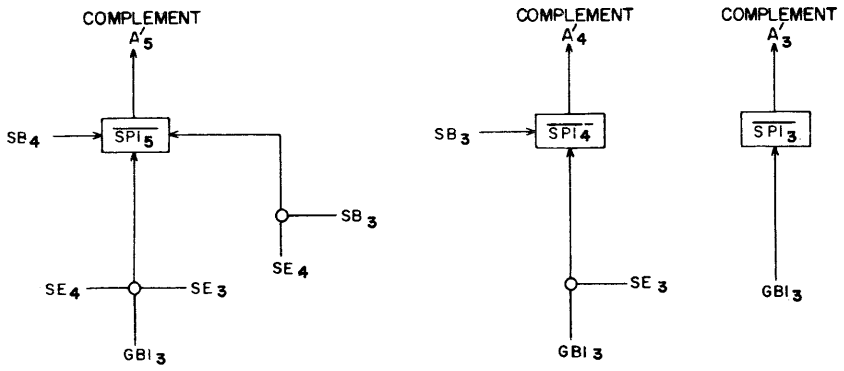
Passed borrow from group 1, - $\overline{E}_2$, $\overline{B}_1$, $\overline{B}$

End-around borrow from group 3, - E_2, E_1, B_3, B

The same analysis may be applied to any of the other group borrow inputs.

Stage Probe Input

As explained previously, outputs from the stage probe input cards are used to complement the corresponding A' register FF when no borrow exists. A borrow, then, must force the SPI card to a "0". The diagram below shows the three SPI cards for the second octal group (stages 3, 4, 5) of the A' register.



ADDER CONTROL

The operands are introduced into the base of the adder through two sets of inverters designated R--- and Q---. The basic functions of the adder are arithmetic, transfer, or shift. One R and one Q inverter is associated with each of the 12 adder stages. Inputs to these inverters may be combined to produce 21 adder-control commands (table 3-2).

STAGE NO.	7	6	5	4	3	2	1	0	
TOTAL	1	0	0	0	0	0	0	0	
PROBE	A170	A161	A151	A141	A131	A121	A111		*
TOGGLE		A160	A150	A140	A130	A120	A110		*
CLEAR	ALL STAGES OF A' SET TO ZERO								
STAGE PROBE INPUT	E507	E506	E505	E504	E503	E502	E501		
GROUP BORROW INPUT						E400			
GROUP BORROW GEN	E302		E301			E300			
GROUP ENABLE	E200		E202						
NO STAGE ENABLE	E071							E001	
STAGE ENABLE		E060	E050	E040	E030	E020	E010		
STAGE BORROW	U070								
+Z → R	0	0	1	0	1	0	1	1	
A → Q	0	1	0	1	0	1	0	1	
STAGE NO.	7	6	5	4	3	2	1	0	

PROBLEM:

ADD	0	0	1	0	1	0	1	1	(R)	43	decimal (053_8)
	0	1	0	1	0	1	0	1	(Q)	85	decimal (125_8)
	1	0	0	0	0	0	0	0	(B)	128	decimal (200_8)

NOTES:

1. THE ELEMENTS SHOWN PRODUCE "1" OUTPUTS EXCEPT WHERE OTHERWISE INDICATED.
2. * THESE SYMBOLS REPRESENT "1" INPUTS, NOT THE STATE OF THE OUTPUTS.

Figure 3-22. Operation of Adder

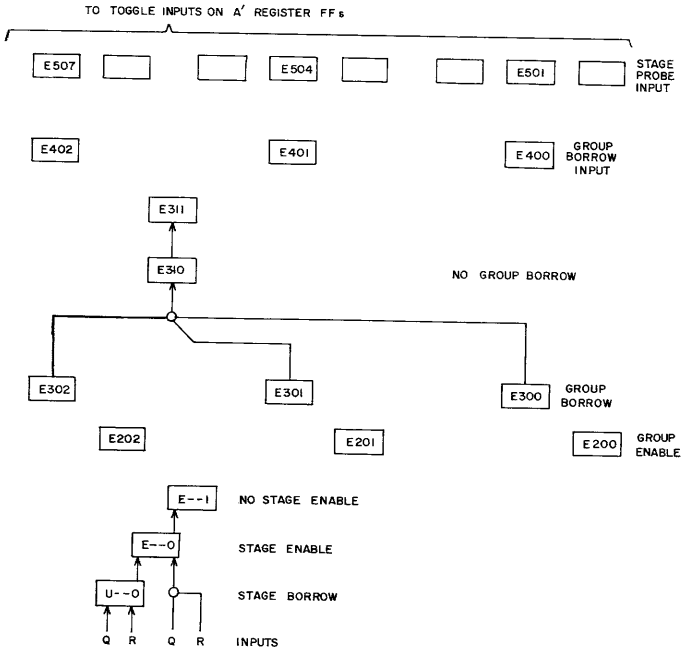


Figure 3-23. Adder Structure, Simplified

TABLE 3-2. ADDER COMMANDS

Type	Command	R	Q
Arithmetic	A + 1	+1	A
	A + Z	+Z	A
	A - Z	-Z	A
	Z + 1	+Z	+1
	Logical Product		Z,A
	Selective Complement	Z	A
	Replace Add One	+1	Z
Transfer	A $\longrightarrow$ A'		A
	-Z $\longrightarrow$ A'	-Z	
	+Z $\longrightarrow$ A'	+Z	
Shift	A left 1	$A \cdot 2^{1}(L)$	

Figure 3-24 describes the inputs to the R and Q inverters. First-level control for each of the functions resides in the W--- element. The control element acts to bring the entire contents of a register into the adder.

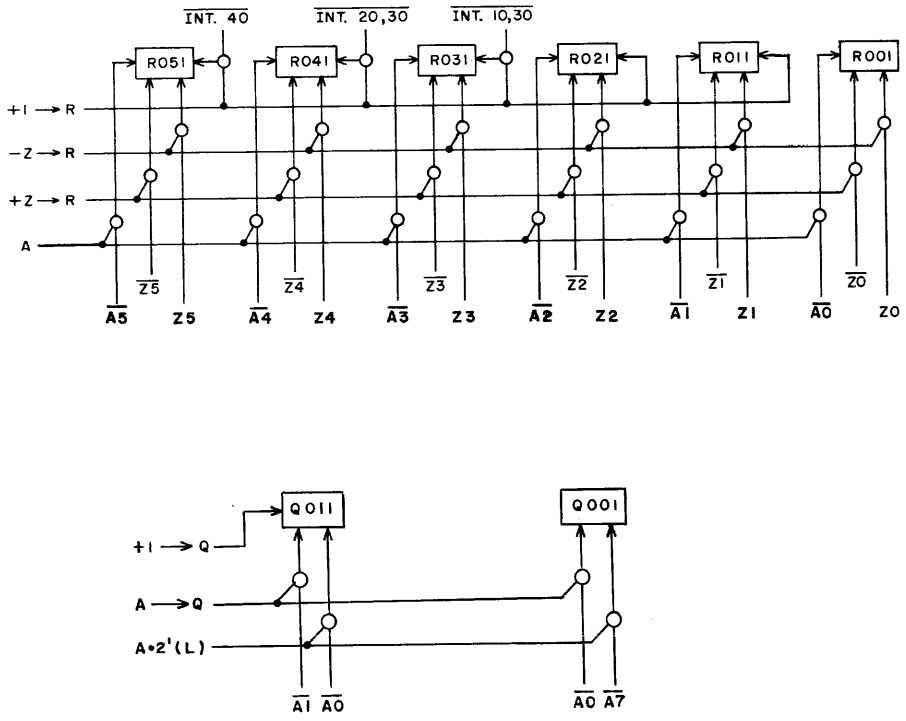


Figure 3-24. Add Input (R and Q Inverters)

The inversion in R and Q must be taken into account when selecting inputs to those elements. For example, a + 1 ⟶ R signal will force "1's" into all stages of R except the first, thereby providing the proper +1 input to the adder.

SHIFTING

The one shift provided in the TeleProgrammer is executed during cycle D by energizing the proper shift inputs (i.e. $(A \cdot 2^{1} \longrightarrow Q)$ to the adder (R and Q inverters).

X01 left shift 1

The shift input is energized during D_3.

INPUT/OUTPUT SECTION

The input/output section of the TeleProgrammer provides the methods for data exchange and for proper control of information transmission between the TeleProgrammer, the various external equipments, and the reader and punch.

Four cables, two input and two output, link the external equipment to the TeleProgrammer. All information from the externally located peripheral equipments must enter or leave the TeleProgrammer through one of these cables.

Information passes between the TeleProgrammer and the external equipment as a block of information at a word-by-word rate or as a single word input or output. The speed of the particular equipment in communication with the TeleProgrammer determines the data exchange rate.

INFORMATION CABLES

Each information cable contains 23 twisted-pair information or control lines and one common ground line. Line assignment for the cables appears in table 3-3. Since the cables are identical in construction they are interchangeable with respect to physical connection to the TeleProgrammer as well as to the function they perform. Table 3-4 outlines the function of each wire of the input and output cables.

Data appears on the communication lines as one of two d-c voltage levels: the binary "1" condition is 0v, the binary "0" condition is -16v. All binary digits of a word appear simultaneously on the wires of the output or input cable.

Double connectors in each external equipment allow more than one device to be connected to the TeleProgrammer. Select codes sent by the TeleProgrammer determine which equipment attached to a cable shall communicate with the Tele-Programmer.

TABLE 3-3. PIN ASSIGNMENTS, INPUT/OUTPUT CABLES

Input Cable	Pin	Output Cable
Bit 0 Input Status and Information	A	Bit 0 Output Function and Information
1 ↑	B	1
2	C	2
3	D	3
4	E	4
5	F	5
6	H	6 (Output Function)
7 ↓	J	7 (Information)
	K	8
	L	9
	M	10
	N	11
	P	
Input Ready	R	Information Ready
Input Request	S	Output Resume
	T	Function Ready
	U	Master Clear
Input Disconnect	V	
	W	
	X	
	Y	Interrupt 30
	Z	Interrupt 40
	a	
Ground	b	Ground

TABLE 3-4. INPUT AND OUTPUT CABLE LINES

INPUT CABLE LINES

Input Data and Input Status (8 lines)	Dual Purpose: 1) As data lines, they hold equipment input register contents which the 8092 may sample. 2) As input status lines, they indicate equipment's response to status request interrogation.
Input Ready (1 line)	Indicates that the external equipment contains information which the 8092 may sample.
Input Request (1 line)	Indicates to external equipment that TeleProgrammer desires an input word. This line is turned off by input ready.
Input Disconnect (1 line)	Indicates to 8092 that input device has no more data to deliver. The 8092 is then free to resume main program with no further delay. (Generally the input instruction establishes a storage field block of greater capacity than the anticipated input information block.)

OUTPUT CABLE LINES

Output Data (8) and Output Function (12 lines)	Continuously monitored by all equipment. Dual purpose: 1) As output data lines they hold output word which the external device may sample. 2) As output function lines they carry external function (EF) codes to select or sense a condition within the equipment. Function ready alerts the equipment to sample EF code.
Function Ready (1 line)	Accompanies EF code; turned on by instruction 75 and causes the equipment to examine EF code. It is turned off by an output resume from the external equipment.
Information Ready (1 line)	This signal accompanies the output data word from the 8092 and is turned on when the 8092 has a word of information ready for the external equipment. It is turned off by an output resume from the equipment.
Output Resume (1 line)	This signal is turned on when the external device has accepted the output word or EF code.
Master Clear (1 line)	This signal clears all external equipment. It occurs when the Load/Clear switch is in the CLEAR (down) position.
Interrupt 30 (1 line)	If no Interrupt Lockout, store P in PSR and obtain next instruction from address (r) 0030.
Interrupt 40 (1 line)	If no Interrupt Lockout, store P in PSR and obtain next instruction from address (r) 0040.

EQUIPMENT SELECTION

The TeleProgrammer controls the operation of its external equipment by issuing 12-bit function codes. (See code listing in chapter 1.) This process initiated by the EF (75) instruction, selects a 12-bit code located in the next two sequential memory locations, and places it on the output lines together with a function ready signal. Although each external device examines the code, only the particular unit directed by the code responds to it. After responding to the code the external equipment returns an output resume signal to the TeleProgrammer.

The upper 6 bits of the function code (unit designator) select a specific external equipment and the lower 6 bits (function designator) specify the function. The unit designator provides the TeleProgrammer with 64 distinct combinations for selecting various external control units. The function designator tells the selected unit what action is requested of it.

Status Request Responses

The interrogated equipment replies to a status request with a 8-bit status response signal on the input cable. The computer receives this response through an input instruction (72 or 76) followed by program control.

If more than one status request response code appears, the resultant code recognized by the TeleProgrammer is the sum of the transmitted codes. For example, if the magnetic tape unit makes the following responses:

004	00	000	100
020	00	010	000
040	00	100	000
	00	110	100
	0	6	4

The resultant code, 064, must be analyzed by the programmer to determine which codes responded and what conditions exist in the unit.

Selection Priority

If the buffer and normal channels are to be used concurrently, it is necessary to set up and initiate the buffer operation first. The reason is that an equipment selection on the buffer channel is not dropped by a subsequent EF instruction, providing the buffer is busy, but any EF code drops all previous selections on the normal channel. This, however, does not hold for all peripheral equipment. (i.e., one that must be de-selected by an EF code).

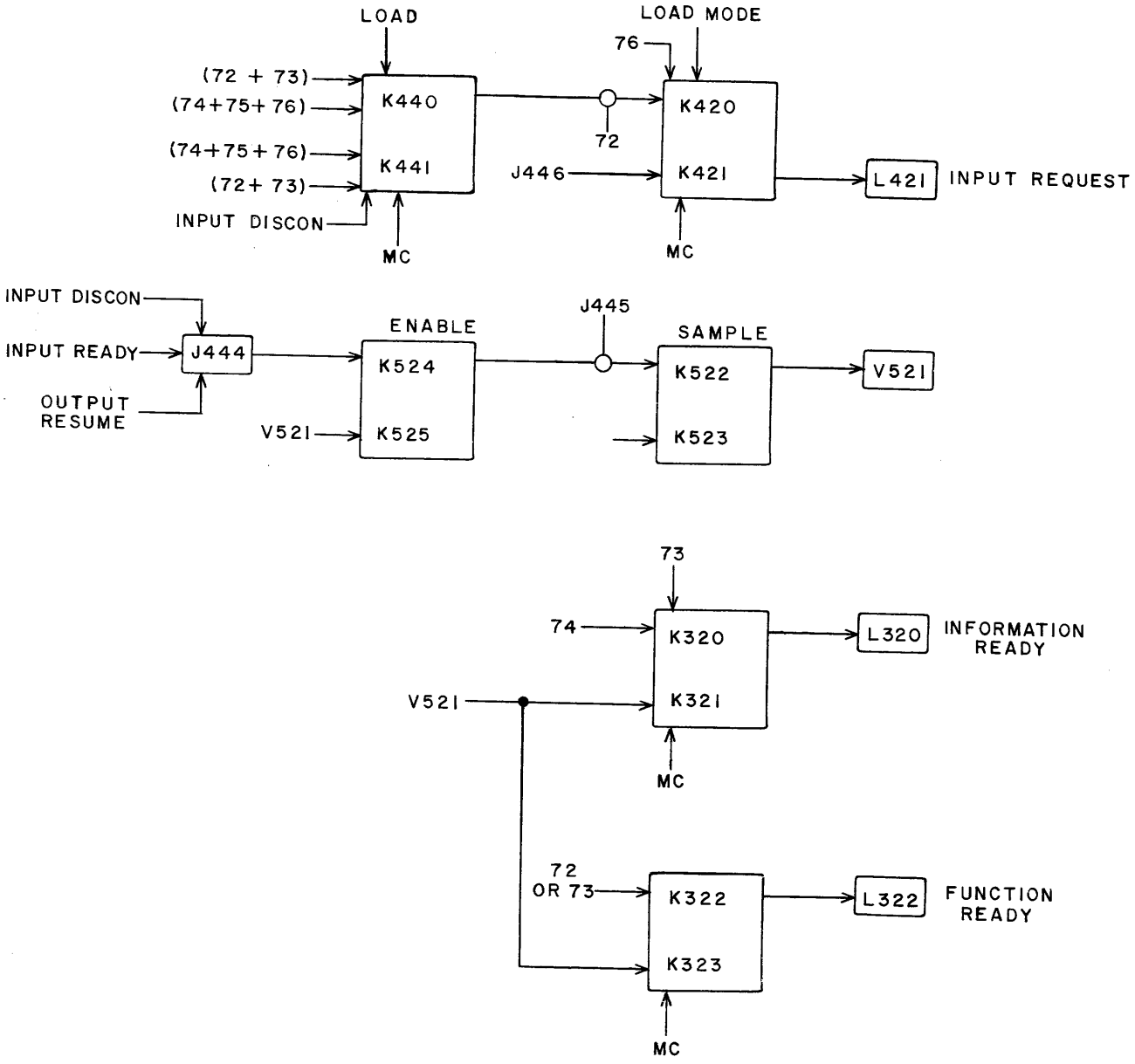
INPUT/OUTPUT CONTROL

The input/output control system (figure 3-25) consists of I/O Sequence Control, Wait and Function Ready FFs, and the resync circuits which handle signals from external equipment.

Resynchronization System

The resynchronization system permits a signal generated outside the logic circuits of the TeleProgrammer to be synchronized with those circuits. This is possible because the incoming signal is of long duration, compared to the logic pulses. The purpose of resynchronizing is to convert a signal which is not synchronized with the TeleProgrammer to one that is timed by the TeleProgrammer clock and to resolve runt pulses.

Network: The resync network (figure 3-26) consists of a two-stage, double-rank counter and decoder network. Timing is derived from two inverters driven directly from the master clock; the inverters present a square clock pulse to the FF inputs.



K440/441=I/O SEQ CONTROL FF
K420/421=I/O WAIT INPUT FF
K320/321=I/O WAIT OUTPUT FF
K524/525=ENABLE FF
K522/523=SAMPLE FF
K322/323=I/O FUNCTION READY FF

Figure 3-25. Input/Output Controls

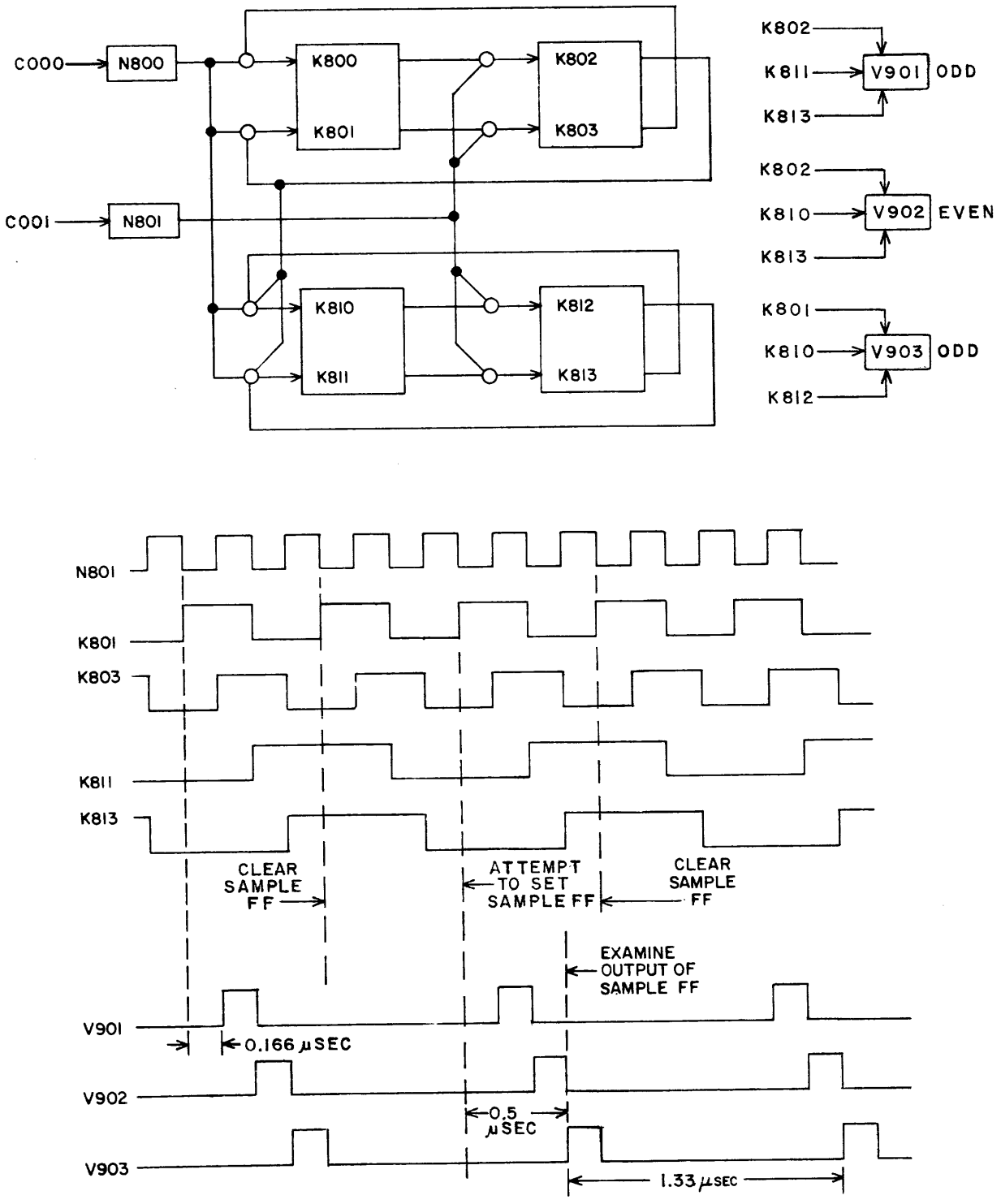


Figure 3-26. Resynchronizing Network

Concurrent with every fourth even clock pulse, V901 is energized, and V901 produces an output pulse, followed at relative times 0.167 and 0.33 usec by V902 and V903 (figure 3-26). There is a lapse of 1.33 usec between successive outputs from any one of the decoding inverters.

Circuit: In the typical resync circuit (figure 3-27), an external control signal (normally -16v) must, upon going to ground (0v), be timed with the Tele-Programmer logic. Ground on a cable is a "1", a -16v signal is a "0". When no external signal is present (circuit input -16v) the Sample FF is cleared; the Enable FF is set. When the external signal changes to 0v, the Sample FF is set. A synchronized output pulse which occurs 0.66 usec later is sent to the other logic circuits and also clears the Enable FF. After the output of the Sample FF is examined, it is cleared and cannot be set again until the Enable FF is set. The result is a single synchronized pulse.

The Enable FF remains cleared until the signal returns to -16v (normal static state of the circuit). Once circuit input has gone to 0v, it must return to -16v to enable the path to set the Sample FF.

Runt Pulses: A runt pulse is a pulse of such low power as to be uncertain of triggering an input circuit. In I/O networks, such pulses are most always the result of a synchronized pulse (or gate) being ANDed with an asynchronous input signal, such as the input to the SET side of the Sample FF (figure 3-27).

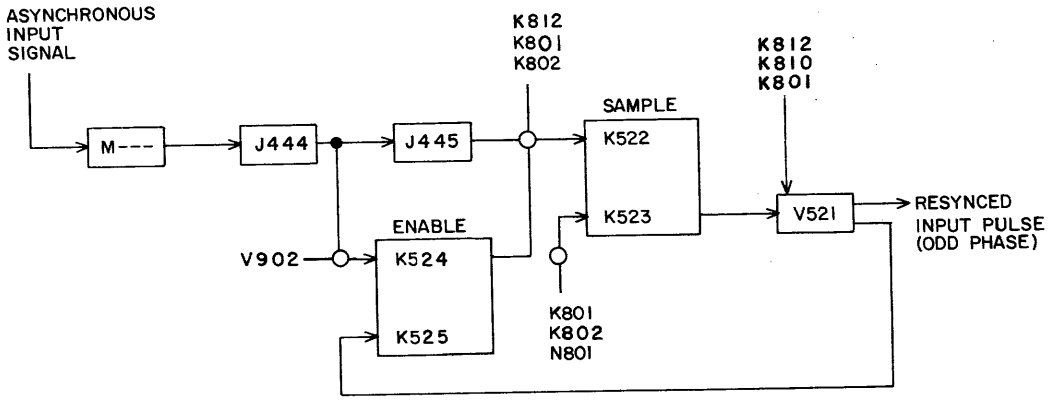


Figure 3-27. Resync Circuit

If the external signal arrives at the AND when one of the gating terms is ready to drop out, there is no assurance that the input pulse to Sample will be sufficiently long to switch that FF. For this reason, the output from Sample is not examined until 0.66 usec later, concurrent with V903 in this case, allowing enough time for the FF to settle down.

The operation of figure 3-27 may be understood by referring to the waveforms at the bottom of figure 3-26.

Time-Sharing of Resync Circuit: Asynchronous signals from outside the TeleProgrammer logic share one common resync circuit (figure 3-25, left). There are three asynchronous signals to which the TeleProgrammer must respond:

Input Disconnect
Input Ready
Output Resume

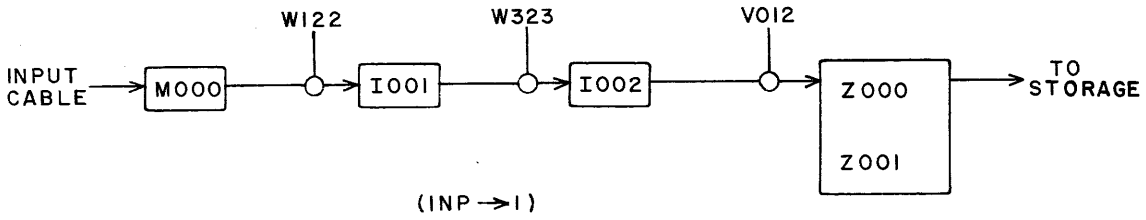
I/O Sequence Control

The I/O sequences have a general storage reference D, B, C, B', C', B', C', etc.; the first D reads the instruction, B obtains the starting address, and C obtains and compares the ending address to the starting address. The succession of B'-C' cycles obtains the data words. The I/O Sequence Control FF which is set at the end of cycle D provides an entrance into C and assures that the starting address obtained during B is used to store (or read) the first data word. During repetitive I/O operations (instruction 72 or 73), the starting address in the A register increases by 1 during B'-C' sequence until A equals terminating address, at which time the FF drops and the operation stops.

Once an input request is initiated, the main timing chain must be stopped until the word appears on the input lines; the Wait Input FF is set by either instruction 72 or 76. The input ready signal accompanying the word clears the FF. Once an output word is read out of storage, the Wait Output FF remains set until the output resume signal returns to the TeleProgrammer, indicating that the external equipment has processed the word.

INPUT DATA TRANSMISSION

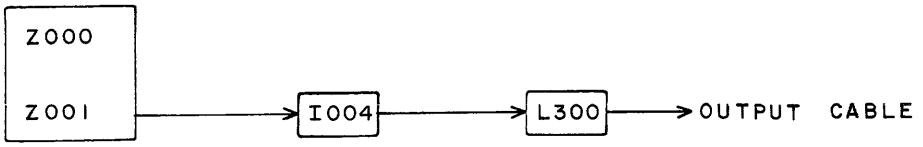
All data from external input devices reaches the Z register through an inverter rank designated I--1 and I--2. The path of input data to the least significant stage of Z is shown below. Each of the 8 stages of Z has similar logic arrangement between it and the input cable.



Input amplifier cards contain three identical circuits. Each is assigned an individual M--- symbol. These amplifiers form the input boundary elements between the input cable and the TeleProgrammer, and they convert communication line signals to TeleProgrammer signal levels.

OUTPUT DATA TRANSMISSION

Output data is transmitted from storage through the Z register to the output cable. The path of output data from the least significant stage of Z is shown below; the path is similar for each of the eight stages of Z. In order for data to appear in its correct configuration as output from the output amplifiers, it is necessary that it be taken from the clear side of the stages of Z.



Output amplifier cards contain three identical circuits. Each circuit is assigned an individual L symbol. The output amplifiers form the boundary elements between the computer and the cable groups and convert the TeleProgrammer signal levels to communication line levels.

BUFFER OPERATIONS

Buffer operations proceed as normal I/O operations with two exceptions: (1) Buffered I/O data does not tie up the TeleProgrammer; computation may continue while data is being transferred. (2) Since the buffer circuits compare the starting and terminating addresses automatically, no C' cycle is needed for subtracting A from Z. The Buffer Entrance register is loaded with the starting address by a LOAD A instruction followed by 04 (ATE); the Buffer Exit register is loaded with the terminating address (LWA + 1) by a LOAD A followed by 05 (ATX). Incrementing BER during each buffer cycle is accomplished similarly to incrementing P since BER is a 10 bit counter. In the 04, 05, 70, and 71 instructions, the next sequential memory location following the instruction word provides a jump address if the buffer is busy.

BUFFER CONTROL

A buffer cycle may be initiated at the end of any storage reference cycle by setting the Storage Interrupt FF. A buffer cycle will not interrupt A D→C cycle transfer or during an instruction involving a C→C' cycle transfer since the address in which the data was to be read from or stored during C' would be removed from S during the buffer cycle.

The buffer controls are shown in figure 3-28. During an input (70) instruction, BSI and BFR Input are set at time 23, and BFR Ready is set at time 26. These conditions initiate an input request. This frees the TeleProgrammer to continue the program until the buffer inputs are gated to receive the data word, at which time BSI is cleared. During time 25 of the current storage reference cycle A Storage Sequence Interrupt is set (I 550), followed by BFR Cycle. This interposes a buffer cycle at the end of the current SRC.

Each input ready on the buffer channel effectively clears BSI, which must be reset at time 23. Two inputs to set BSI are provided, one for the first word of input (70), the other for all other buffer word cycles. If a buffer word is received while the program is in a DD or CC' sequence SSI will not be set until time 25 of the D cycle.

If a buffer output resume or input ready clears BSI while the timing chain is stopped, Buffer Step will be set. In this manner, a buffer cycle is initiated even though the timing chain is not running.

Forced Jump

All buffer instructions provide a jump address if the buffer is busy. If a buffer operation is attempted while the buffer is busy, a C cycle is initiated which reads out the next sequential memory location as the jump address.

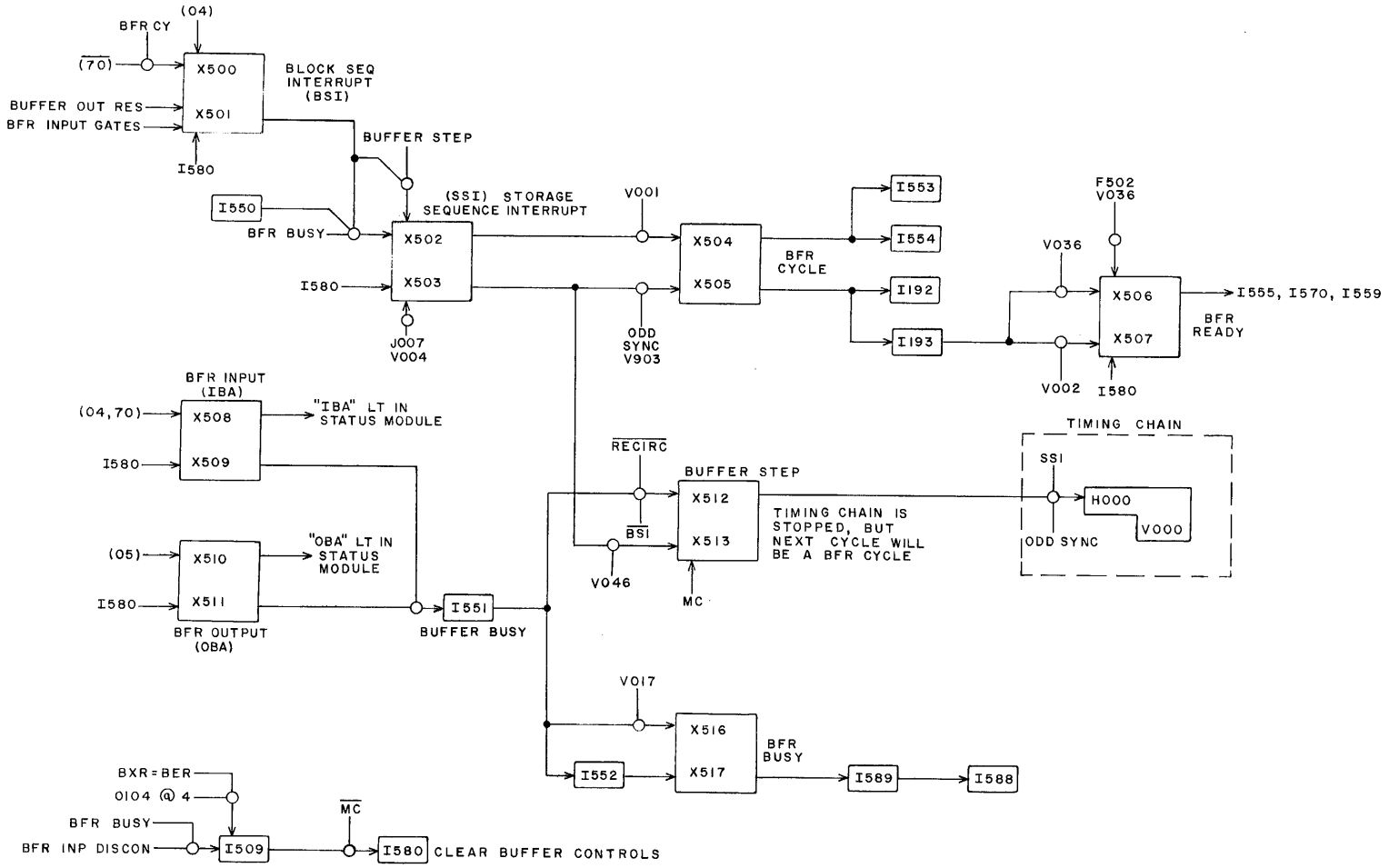


Figure 3-28. Buffer Controls

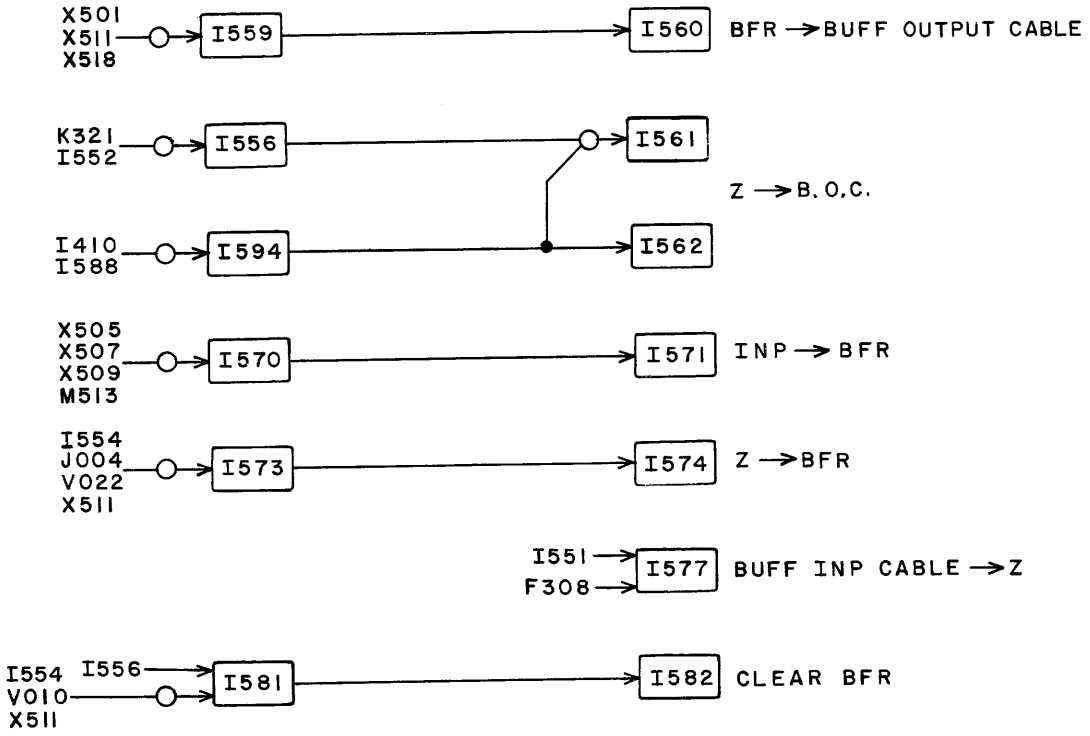


Figure 3-29. Buffer Registers Transmission Control

TRANSMISSION CONTROL

Transmission controls for the three buffer registers are shown in figure 3-29. Transmissions are detailed under the instructions during which they occur. Normal register transfer and pyramid controls are discussed elsewhere.

X04

This instruction transmits the contents of A and the lower two bits of Tag register 3 to BER. BER is "force" set directly from the A register at time 26.

X05

This instruction transmits the contents of A and the lower two bits of Tag register 3 to BXR; this is a direct transmission occurring at time 26.

X06

This instruction transmits the contents of BER to A. This is a direct transmission.

BUFFER I/O CABLES

The lower half of figure 3-29 is used in this discussion. Any equipment connected to the buffer cables may be addressed by a normal I/O instruction, providing the buffer is not busy. The use of the buffer cables for buffer operations is contingent upon the setting of the Buffer Ready FF. For the first word of buffer input, this FF is set at time 26 of the D-C cycle of the 70 instruction and at time 26 of the buffer cycle for succeeding words, but the inputs are not gated until the input ready signal is received. The outputs are gated as soon as Buffer Ready is set (time 26 of a buffer cycle for the buffer output operation).

INTERRUPT

An interrupt signal halts the TeleProgrammer main program and initiates a subroutine. At the completion of the subroutine the main program may be resumed. There are four interrupt lines: 10, 20, 30, and 40. Each transfers computer control to a distinct location in memory.

The two parts of the interrupt circuit, recognition and execution, are shown in figure 3-30.

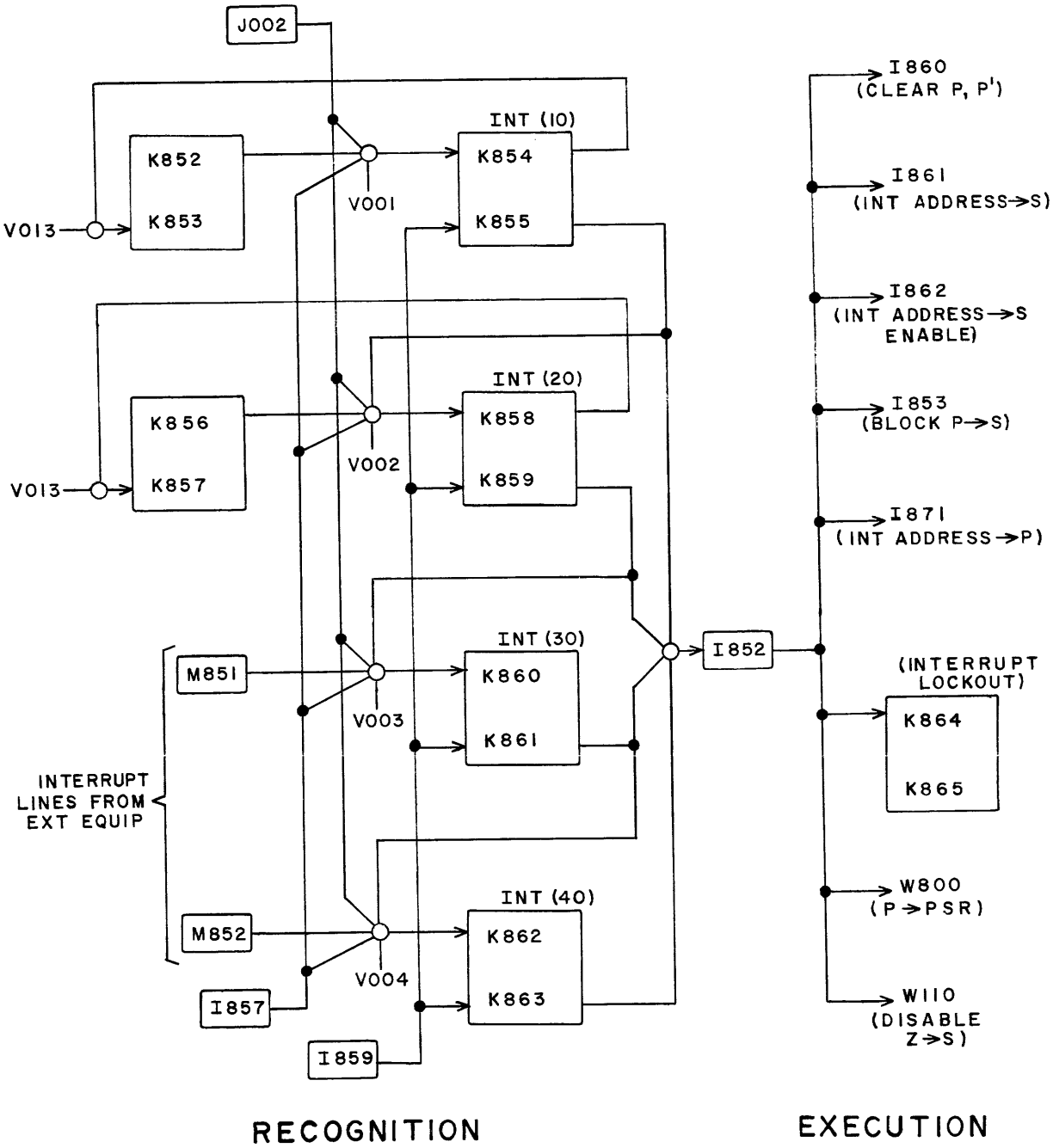


Figure 3-30. Interrupt Circuit

RECOGNITION

The interrupt is recognized at the start of the read next instruction cycle when one of the four Interrupt FFs is set. This forces a jump to the memory location associated with the interrupt level.

Interrupt 10 Recognition

Interrupt 10 is a manual interrupt activated by momentarily pressing a keyswitch. This interrupt signals sets FF K850/851, which in turn sets the Interrupt 10 FF at D_1, time 1.

Interrupt 20 Recognition

Interrupt 20 is activated each time a buffer operation is completed -- when BER = BXR, or on receipt of an input disconnect during a buffer input. FF K856/K857 retains the buffer interrupt until it is recognized at D_1, time 2.

Interrupt 30 and 40 Recognition

Interrupt FFs 30 and 40 can only be set at D_1, times 3 and 4. These FFs are activated by peripheral devices. If either of the two lines is used by more than one equipment, each equipment on that line must be interrogated during the interrupt routine to determine the signal's source. The routine then must branch to a subroutine to handle that particular interrupt, during the course of which the interrupt signal is removed from the line by the external equipment.

Priority of Recognition

Should more than one interrupt line be activated at a given time, priority is assigned to the interrupts in the following order: 10, 20, 30, and 40. This is accomplished by interconnections between adjacents FFs and by the Interrupt

Lockout FF. When an interrupt operation is in progress, this FF and the Clear Interrupt Lockout input to 1873 prevent any subsequent interrupt recognition until completion of one instruction following a CIL instruction.

EXECUTION

During the execution portion (figure 3-30) of the interrupt circuit, the TeleProgrammer stores the contents of P in PSR, and then takes its next instruction from (r) 0010, (r) 0020, (r) 0030 or (r) 0040 to enter the subroutine.

During the read next instruction cycle (D), the program address is "force" set to address 10, 20, 30 or 40. The programmer must have some subroutine starting at these locations that will perform the function required by the interrupt.

The last instruction of the interrupt subroutine will be a CIL (clear interrupt lockout). This permits the TeleProgrammer to continue the previous operation and to again check all other interrupt lines for activity in priority sequence.

INTERRUPT LOCKOUT

The interrupt lockout is established by recognition of any interrupt or by execution of any external function command. The interrupt lockout condition is removed by execution of a CIL instruction or by a master clear.

MASTER CLEAR

Whenever a console master clear is performed, all internal interrupt lines are set inactive and the interrupt lockout is removed.

GLOSSARY OF TELEPROGRAMMER TERMS

The following glossary gives the meaning of terms that are used in a relatively specialized sense in this manual.

ADDER	In general, a device used to add two quantities. Specifically, the subtractive adder in the subject computer.
ADDRESS	The number designating a storage location; also the storage location itself.
BIT	Binary digit; may be either "1" or "0".
BORROW	In a subtractive counter or accumulator, a signal indicating that in stage n, a "1" was subtracted from a "0".
BUFFER	Noun: A device in which data are stored temporarily in the course of transmission from one point to another. Verb: To store data temporarily.
BUFFERED INPUT/OUTPUT	A term indicating that the TeleProgrammer may carry on high speed computation at the same time it is exchanging data with a peripheral device. This term must be distinguished from normal I/O, during which the TeleProgrammer cannot engage in computation.
CARRY	In an additive counter or accumulator, a signal indicating that in stage n, a "1" was added to a "1".
CHANNEL	A transmission path that connects the TeleProgrammer to a given external equipment.
CLEAR	A command that removes a quantity from a register by placing every stage in the "0" state.
COMMAND	A signal that performs a unit operation, such as transmitting contents of one register to another, shifting a register, setting a FF.
COMPLEMENT	Noun: See One's Complement or Two's Complement. Verb: A command which produces the one's complement of a given quantity.
CONTENT	The quantity of word held in a register or storage location.

CORE	A small ferromagnetic toroid used as the bistable device for storing a bit in a memory plane.
COUNTER	A register with provisions for increasing or decreasing its content by 1 upon receiving the appropriate command.
END-AROUND BORROW	A borrow that is generated in the highest order of an accumulator or counter, and is sent directly to the lowest order stage.
ENTER	To manually place in a register a quantity that is not from storage. In the TeleProgrammer quantities may be entered in only the A, P, Z and Tag registers.
FUNCTION CODE	The upper 6 bits of a 8-bit instruction.
INPUT DISCONNECT	During an input instruction, a signal sent to the TeleProgrammer by the external device to indicate that the device has completed all available transmission to the TeleProgrammer.
INPUT REQUEST	A request, by the TeleProgrammer for information from an external device. Occurs during input instruction only. (See Resume.)
INSTRUCTION	The TeleProgrammer operates on a two-word instruction set. The first 8-bit word contains the Function Code and Tag register designator. The second word contains the operand or partial address.
INTERRUPT	A signal (or class thereof) which, when received and recognized by the TeleProgrammer, forces the TeleProgrammer to forestall its current operation and jump to a subroutine, the starting address of which is determined by the class of the interrupt. A subroutine may have any number of options. It may merely stop the TeleProgrammer, it may determine the nature of the interrupt in order to take corrective measures, or it may return the TeleProgrammer to another phase of the main program.
LOAD	To place a quantity from storage in the A register.
LOCKOUT	Any function (usually of machine logic) that inhibits an action which would normally occur were the lockout not imposed.

LOGICAL PRODUCT	In Boolean algebra, the AND function of several terms. The product is "1" only when all the terms are "1"; otherwise it is "0". Sometimes referred to as the result of "bit-by-bit" multiplication.
LOGICAL SUM	In Boolean algebra, the OR function of several terms. The sum is "1" when any or all of the terms are "1"; it is "0" only when all are "0".
MASTER CLEAR (MC)	A general command produced by placing the Load/Clear switch in the down (CLEAR) position. An MC clears all of the crucial registers and control FFS to prepare for a new mode of operation.
MODULUS	An integer which describes certain arithmetic characteristics of registers, especially counters and accumulators, within a digital computer. The modulus of a device is defined by r^n for an open ended device and r^n-1 for a closed (end-around) device, where r is the base of the number system used and n is the number of digit positions (stages) in the device. Generally, devices with modulus r^n use two's complement arithmetic procedures, and devices with modulus r^n-1 use one's complement procedures.
NORMAL JUMP	An instruction that jumps from one sequence of instructions to a second, and makes no preparation for returning to the first sequence.
ONE'S COMPLEMENT	With reference to a binary number, that number which results from subtracting each bit of the given number from the bit "1". A negative number is expressed by the one's complement of the corresponding positive number.
OPERAND	Usually refers to the quantity specified by the execution address. This quantity is operated upon in the execution of the instruction.
OPERATION CODE	The upper 6 bits of a 12-bit instruction which identifies the instruction. After the code is translated, it conditions the computer for execution of the specified instruction. The letter F is used to designate this code, which is expressed by two octal digits.
OVERFLOW	The condition in which the capacity of a register is exceeded

PARTIAL ADD	An addition without carries. Accomplished by toggling each bit of the augend where the corresponding bit of the addend is a "1".
PROGRAM	A precise sequence of instructions that accomplishes a TeleProgrammer routine; a plan for the solution of a problem.
READ	To place a quantity from a storage location into a register. The quantity in storage remains unchanged.
READY	The input/output control signal sent by either the TeleProgrammer or an external equipment to alert the device that is to receive a transmission. The ready signal indicates that the word or character has been transmitted.
REPLACE	In the title of an instruction, the result of the execution of the instruction is stored in the location from which the initial operand was obtained.
RESUME	The output control signal sent by an external equipment to indicate that it is prepared to receive another word or character. The resume signal is thus a request for data. (See Input Request.)
ROUTINE	The sequence of operations which the computer performs under the direction of a program.
SHIFT	To move the bits of a quantity right or left.
SIGN BIT	The bit in the highest-order stage of the register (in registers where a quantity is treated as signed by use of one's complement notation). If the bit is "1", the quantity is negative; if the bit is "0", the quantity is positive.
SIGN EXTENSION	The duplication of the sign bit in the higher-order stages of a register.
STATUS	1) The condition of an external device, as reflected in the response given to a status request interrogation by the computer. 2) The condition of the TeleProgrammer as shown by a status indicator on the Operator's Panel. May variously indicate what it is presently doing, why it stopped, or what it will do when it next starts.

TRANSMISSION, FORCED	A transmission where both set and clear inputs, only one of which will be a "1", are simultaneously gated into a FF which has not been cleared previously.
TRANSLATION	An indication of the content of a group of bit registers. A complete translation gives the exact content, while a partial translation indicates only that the content is within certain limits.
TWO'S COMPLEMENT	That number which results from subtracting each bit of a number from "0". The two's complement may be formed by complementing each bit of the given number and then adding one to the result, performing the required carries.
WORD	A unit of information which has been coded for use in the computer as a series of bits. The normal word length is 8 bits.
WRITE	To enter a quantity into a storage location.

CONTROL DATA
CORPORATION

INDUSTRIAL DATA
PROCESSING DIVISION

9549 PENN AVENUE SOUTH, MINNEAPOLIS 31, MINNESOTA