NATIONAL CLASS 310-2
ELECTRONIC DATA PROCESSOR

ENGINEERING DESCRIPTION

VOLUME 1
DESCRIPTION and OPERATION

*National**

For

The National Cash Register Company

Dayton 9, Ohio

BY THE CONTROL DATA CORPORATION

PRINTED IN U.S.A. 4-11-61

*Service Mark—Reg. U.S. Pat. Off.

Chapter One

General Description

NATIONAL CLASS 310-2

ELECTRONIC DATA PROCESSOR

ENGINEERING DESCRIPTION

VOLUME 1

DESCRIPTION and OPERATION

MAY 1, 1961

COMPANY CONFIDENTIAL

Prepared For

THE NATIONAL CASH REGISTER COMPANY

DAYTON 9, OHIO

By The

CONTROL DATA CORPORATION

ADDENDA

(for Publications 100 and 101)

Engineering Change Order 24, dated 7-6-62, incorporates the following changes in the Main Timing Chain:

B → S transfer moved to time x5
B → A transfer moved to time x6

These changes are effective for pages 3-27, 3-28, 3-30 in Volume I (100) and for most of the timing charts in Volume II (101).

ECO 24 also changes the Timing Chain Excursions Counter (top of page 3-8, Volume I) to a reflected binary type. Errors occurring on page 3-9 of that volume (figure 3-3, Main Timing Controls) may be corrected by referring to the new diagrams in Volume III (102).

GENERAL CONTENTS

Volume I	Description and Operation	
	Chapter 1	General Description
	Chapter 2	Operation
	Chapter 3	Theory
	Appendix A	Glossary
	Appendix B	Number Systems
Volume II	Maintenance	
	Appendix C	Timing Charts
	Appendix D	Building Block
	Appendix E	Card Schematics
	Appendix F	Installation
Volume III	Equipment Diagrams	
Volume IV	Equation File	

CONTENTS

Chapter 1 - General Description

General	1-1
Description of Registers	1-1
Addressing Modes	1-4
List of Instructions	1-4
Summary of Characteristics	1-8
Input-Output Operations	1-8

Chapter 2 - Operation

Operation Without Pre-Stored Program	2-2
Operation With Pre-Stored Program	2-2
Shut-Down of Equipment	2-2
Tape Loading Procedures	2-3
Photoelectric Reader	2-3
High-Speed Punch	2-4
Magnetic Tape Cabinet	2-6
Controls and Indicators	2-11
Computer Control Console	2-11
Typewriter Control Panel	2-14
Magnetic Tape Cabinet	2-15

Chapter 3 - Principles of Operation

Control Section	3-2
Registers	3-2
Master Clock	3-6
Main Timing Chain	3-7
Storage Reference Cycle	3-8
Function Translator	3-11
Operating Controls	3-13
Non-Volatile Storage	3-19
Storage Section	3-21

CONTENTS (Cont'd.)

Chapter 3 (Cont'd.)

Properties of the Magnetic Core	3-23
Storage Control	3-25
Address Selection	3-35
Electronic Theory of Memory Circuits	3-40
Arithmetic Section	3-47
Binary Arithmetic	3-47
Arithmetic Operations	3-48
Theory of the Borrow Pyramid	3-49
Pyramid Operation	3-52
Pyramid Control	3-58
Input-Output Section	3-62
Information Cables	3-62
Equipment Selection	3-65
Input-Output Control	3-66
Input Data Transmission	3-72
Output Data Transmission	3-73
High-Speed Input	3-74
Console Cabinet I/O Equipment	3-74
Photoelectric Reader	3-74
High-Speed Punch	3-81
Punch Control Unit (Chassis 10400)	3-81
Capacitive Delay Networks	3-82

APPENDIXES

Glossary of Computer Terms	A
Number of Systems	B

TABLES

1-1	List of Instructions	1-5
2-1	Function of Manual Controls and Indicators (MTU)	2-16
3-1	Decimal and Binary Equivalents	3-47
3-2	Pyramid Commands	3-58
3-3	Pin Assignments, Input/Output Cables	3-63
3-4	Input and Output Cable Lines	3-64

FIGURES

Chapter 1 - General Description

1-1	C310-2 Computer	vi
1-2	Block Diagram, C310-2 Computer	1-2

Chapter 2 - Operation

2-1	Computer Console	2-1
2-2	Photoelectric Reader	2-4
2-3	High-Speed Punch	2-5
2-4	Tape at Unload Position	2-7
2-5	File Protection Ring	2-8
2-6	Typewriter Control Panel	2-12
2-7	Magnetic Tape Switch Panel	2-13

Chapter 3 - Principles of Operation

3-1	Block Diagram, C310-2 Data Processor	3-3
3-2	Master Clock Oscillator Waveforms	3-7
3-3	Main Timing Controls	3-9
3-4	Main Control Timing	3-10
3-5	Typical Function Translation	3-12
3-6	Start-Stop Controls	3-15
3-7	Load/Clear, Enter/Sweep Controls	3-17
3-8	Non-Volatile Storage Circuits	3-20
3-9	Memory Plane	3-21
3-10	Four-Bit Magnetic Core Matrix	3-22
3-11	Typical Hysteresis Diagram	3-24
3-12	Basic Storage Reference Cycle	3-26
3-13	Storage Reference Control	3-31
3-14	Storage Sequence Control	3-33
3-15	R/W Driver Selection	3-37
3-16	Typical Diverter Selection	3-38

FIGURES

Chapter 3 (Cont'd.)

3-17	Inhibit Driver Selection	3-39
3-18	R/W Driver (51)	3-41
3-19	Diverter (52A)	3-42
3-20	Selector (53)	3-43
3-21	Current Source (54)	3-44
3-22	Inhibit Generator (58)	3-45
3-23	Sense Amplifier (57)	3-46
3-24	Borrow Pyramid Structure, Simplified	3-56
3-25	Operation of Borrow Pyramid	3-59
3-26	Pyramid Inputs (R & Q Inverters)	3-61
3-27	Resynchronizing Network	3-66
3-28	Typical Resync Circuit	3-67
3-29	I/O Controls	3-69
3-30	I/O Timing	3-75
3-31	Load Mode Controls	3-77
3-32	Normal Selections of PER and HSP	3-79
3-33	External Inputs to Z	3-81
3-34	Punch Logic	3-83
3-35	Punch Control Unit (Simplified Schematic)	3-86
3-36	Fixed Delay	3-87
3-37	Variable Delay	3-88
3-38	Capacitive Delay	3-89
3-39	Capacitive Delay Application	3-90

Figure 1-1. C310-2 Computer

GENERAL

The C310-2 Computer is a highly flexible, multi-purpose, stored program data processor and converter with high speed transistor amplifier circuits, diode logic and an extremely fast magnetic core matrix storage section. Word construction is 12 binary digits, parallel throughout, programmable to multiple precision and to alpha-numeric and binary-coded decimal.

Memory size is 4096 12-bit words with a storage cycle time of 6.4 microseconds. The standard instruction list comprises 64 instructions.

A general purpose input-output channel system is provided for attaching a variety of external devices. Input and output transmissions are either a single 5, 7, or 8-bit character, or a 12-bit word. Integral to the computer are a photoelectric reader (PER) and a Teletype Model BRPE high-speed punch (HSP). Both units use standard paper tape. Optional equipment includes:

1) A magnetic tape system, C336-4/5, with an input-output rate of 15,000 characters per second, employs up to four tape handlers.
2) A C351-4 typewriter cabinet, using a Soroban-modified IBM electric typewriter with necessary controls.
3) A variety of IBM punched card and line printing equipment with specially designed adaptor units.

The computer is contained in a cabinet resembling a standard executive desk (figure 1-1). The computer cabinet weights approximately 620 pounds and requires an air-conditioned environment. Cooling is effected by two blowers within the cabinet. The C310-2 operates from any 115 vac, single phase, 60 cps source.

DESCRIPTION OF REGISTERS

The computer contains three operational registers, A, Z, and P. These registers are referred to implicitly in the individual program steps and are indicated on the control panel.

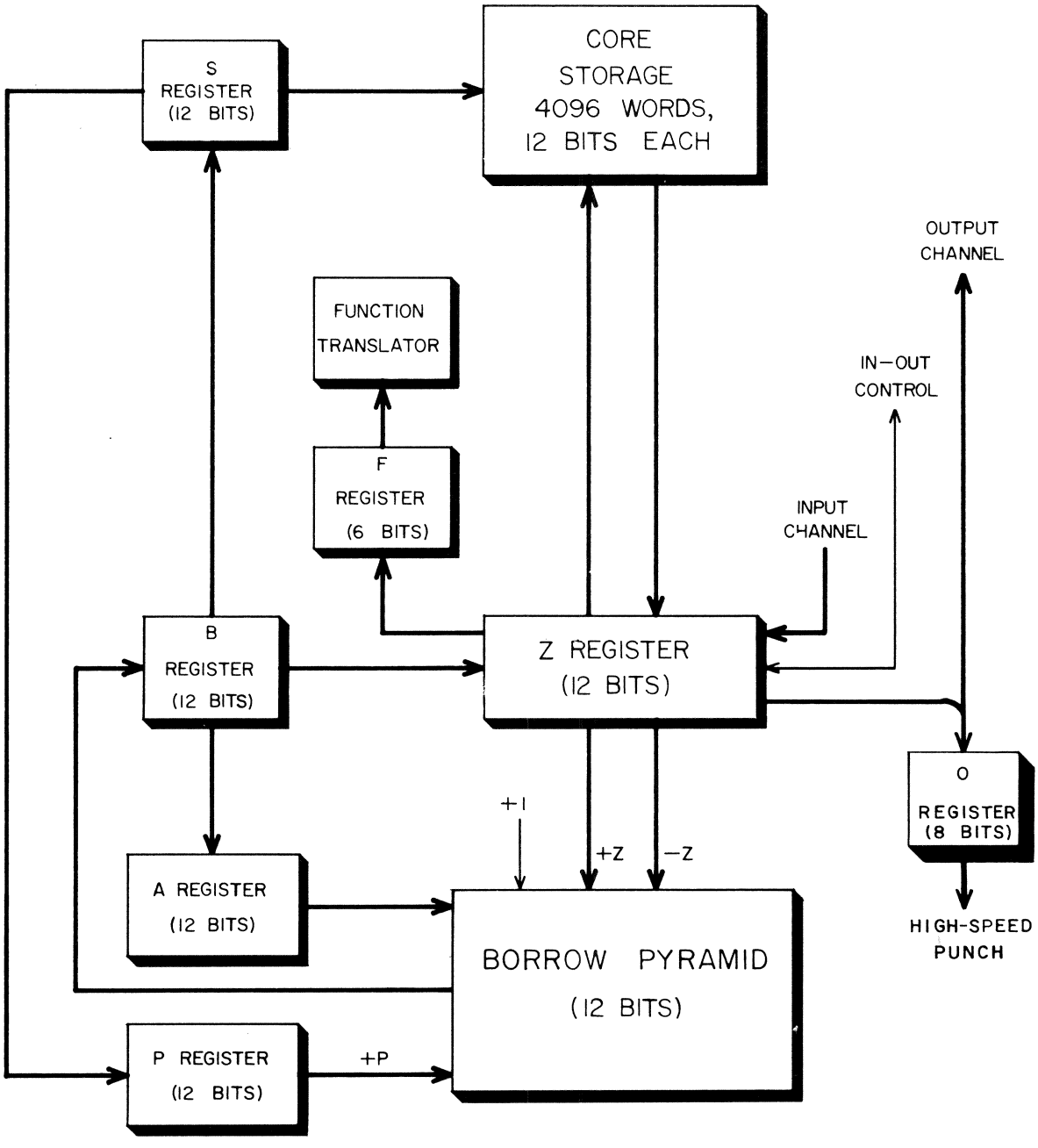


Figure 1-2. C310-2 Block Diagram

There are four transient registers, B, S, F, and O (figure 1-2) which are important only to the internal operation of the machine and are not indicated on the control panel..

A Register (12 bits)

The principal arithmetic register, A, operates as a 12-bit subtractive accumulator for most arithmetic processes. Addition and subtraction use one's complement arithmetic and are performed modulus $2^{12}-1$. The quantity zero is represented by all zeros.

Z Register (12 bits)

As a buffer register for storage, Z receives the word read out of 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 (12 bits)

P, the program address register, holds the address of the current instruction. At the beginning of each instruction, the contents of P are increased by one to provide the address. If a jump is called for the address is entered in P.

B Register (12 bits)

Results of arithmetic operations are first formed in B, an auxiliary arithmetic register, and then transmitted to the A, Z, or S registers.

S Register (12 bits)

Prior to storage reference the address is entered in S, the storage address register; the contents of S are then used to select the storage location of the reference.

F Register (6 bits)

The F register holds the upper 6 bits of an instruction word, the function code, throughout the execution of the instruction. The execution of an instruction is under the control of the quantity in F.

O Register (8 bits)

O, the punch storage register, is used on output operations involving the paper-tape punch. Data normally held in Z is transferred to O, releasing Z and enabling the computer to resume high-speed computation while the output word is recorded by the punch at a comparatively slow rate.

ADDRESSING MODES

Computer operations are 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 a 12-bit word comprising a 6-bit function code, F, and a 6-bit execution address, E. Program modifications are accomplished and operands are manipulated by direct, relative, or indirect addressing through the use of E. The functions assigned to E depend on the addressing method used as discussed below:

No Address (n)	E is used as a 6-bit operand (automatically extended to 12 bits, with the upper 6 bits zero). Arithmetic and logical operations use the 6-bit quantity in the instruction word, eliminating the need for entering many constants into storage.
Direct Address (d)	E, extended to 12 bits with the upper 6 bits zero, is used as the operand address. These instructions address the first 64 words of storage.
Indirect Address (i)	E is extended to 12 bits with the upper 6 bits zero. The 12-bit quantity read from this address (from the first 64 words of storage) is used as the address of the operand.
Relative Address	E contains a 6-bit quantity which is added to (f), or subtracted from (b), the current contents of the P register. This mode is used to address storage locations in the immediate vicinity (±63) of the address containing the current instruction.

LIST OF INSTRUCTIONS

The computer uses 64 instructions (table 1-1); 62 of which perform arithmetic or manipulative operations. The other two representing program faults may be used in any program to stop the computer.

The execution time of any computer instruction is a direct function of the number of storage references required.

Table 1-1

INSTRUCTIONS

OPERATIONS GROUP	CODE	MODE	TIME	EXPLANATION
Error Halt	00 77		6.4 μs 6.4	Stop operation. Press RUN switch, next instruction will be executed.
Fixed Shift	01	n	6.4	Shift contents of A left circular n binary positions depending on E. 0102 - 1 pos. 0110 - 3 pos. 0103 - 2 pos. 0111 - 6 pos. Multiply contents of A 0112 - by 10_{10} (12_8) 0113 - by 100_{10} (144_8)
Transmit P to A	0101	n	6.4	Transmit contents of P to A. Quantity in P defines location of current instruction.
Logical Product	02 10 11 12 13	n d i f b	6.4 12.8 19.2 12.8 12.8	Form in A the logical product of original contents of A and operand. Logical product formed for each bit of A in following manner: Contents of A 0011 Operand 0101 Logical Product 0001
Logic Sum	03 14 15 16 17	n d i f b	6.4 12.8 19.2 12.8 12.8	Form in A the logic sum of original content of A and operand. Logic sum formed for each bit of A according to following manner: Contents of A 0011 Operand 0101 Logic Sum 0110
Load	04 20 21 22 23	n d i f b	6.4 12.8 19.2 12.8 12.8	Replace contents of A with the operand.
Load Complement	05 24 25 26 27	n d i f b	6.4 12.8 19.2 12.8 12.8	Replace contents of A with the "1's" complement of operand obtained by replacing each bit with its opposite, ("1" with "0", "0" with "1").
Add	06 30 31 32 33	n d i f b	6.4 12.8 19.2 12.8 12.8	Form in A the sum, modulus $2^{12}-1$, of original contents of A and the operand.
Subtract	07 34 35 36 37	n d i f b	6.4 12.8 19.2 12.8 12.8	Form in A the difference, modulus $2^{12}-1$, of original contents of A minus the operand.
Store	40 41 42 43	d i f b	19.2 25.6 19.2 19.2	Replace operand with the contents of A, which remains unchanged.

Table 1-1

INSTRUCTIONS

OPERATIONS GROUP	CODE	MODE	TIME	EXPLANATION
Shift Replace	44 45 46 47	d i f b	19.2μs 25.6 19.2 19.2	Form in A the value of the operand shifted left end-around 1 bit position. Replace operand with result.
Replace Add	50 51 52 53	d i f b	19.2 25.6 19.2 19.2	Form in A the sum, modulus $2^{12}-1$ of original contents of A and the operand. Replace operand with result.
Replace Add One	54 55 56 57	d i f b	19.2 25.6 19.2 19.2	Form in A the sum, modulus $2^{12}-1$, of the operand and 1. Replace operand with result.
Zero Jump*	60 64	f b	6.4 6.4	If A contains all "0's", jump; if not obtain instruction from next sequential storage location.
Non-Zero Jump*	61 65	f b	6.4 6.4	If A does not contain all "0's"; jump; if not obtain instruction from next sequential storage location.
Positive Jump*	62 66	f b	6.4 6.4	If A is positive, jump; if not obtain instruction from next sequential storage location.
Negative Jump*	63 67	f b	6.4 6.4	If A is negative, jump; if not obtain instruction from next sequential storage location.
Jump Indirect	70	d	12.8	Obtain instruction address from location specified by E, 1 of first 64 storage locations.
Jump Forward Indirect	71	fi	12.8	Obtain next instruction address from location E positions following current instruction address (E+P).
Input	72	fi	12.8 + 12.8n	Read into storage from input device successive units of information; store in sequential locations from starting address to, not including, terminating address. Starting address specified as contents of storage location E positions following current instruction. Terminating address is specified as contents of storage location immediately following current instruction. Next instruction obtained from storage address 2 positions following input instruction. Input instructions terminate upon filling specified storage area or upon re-

Table 1-1

INSTRUCTIONS	OPERATIONS GROUP	CODE	MODE	TIME	EXPLANATION
	Input (cont'd)			μs	ceiving Input Disconnect; A register will contain address of storage location 1 position following last unit of input data stored.
	Output	73	fi	12.8 + 12.8n	Transmit information to output device from storage from starting address to, not including, terminating address. Starting address specified as contents of storage location E positions following current instruction. Terminating address specified as contents of storage location immediately following current instruction. Next instruction obtained from storage location 2 positions following current instruction. A register will contain terminating address which is one position greater than last word sent out.
	Output Direct	74	n	6.4	Transmit lower 6 bits (E) to output device. If more than 6 bits are used by output, the other bits will be zero.
	External Function	75	f	12.8	Transmit EXF code to external equipment. Code is specified by contents of storage location E positions following EXF instruction.
	Input to A	76	n	12.8	Read into A one frame of data from input device, clearing the previous contents of A.

* A jump causes a new sequence of instruction to be executed, starting at the jump address. The jump address in the 60-63 instructions is obtained as P+E; in the 64-67 instructions as P-E.

SUMMARY OF CHARACTERISTICS

Type of machine	Digital, stored program
Mode of operation	Parallel
Word length	12 bits
Type of instruction	Single address
Address features	Direct, indirect and relative addressing
Instruction word	12-bit word: 6-bit function code, F 6-bit execution address, E
Input-Output	Photoelectric paper tape reader: 350 characters per second 5 to 8 bits per character Teletype high-speed paper tape punch: 110 characters per second
Instruction repertoire	64 instructions for arithmetic and manipulative operations
Type of arithmetic	Binary, one's complement
Average execution time	15 microseconds

INPUT-OUTPUT OPERATIONS

Five instructions are concerned with input-output operations. Instructions 72 and 76 receive input quantities. Output data is sent by instructions 73 and 74. External function codes for directing the operation of the input-output equipment are sent by instruction 75.

Communication with external equipment is accomplished by first executing instruction 75. The operand obtained by instruction 75 is the external function code (EXF) which is sent to the equipment on the output channel. It selects the particular equipment and specifies the operating mode. The main program may then execute one of the instructions, 72, 73, 74, 76 to exchange data with the activated equipment. Only one input or output equipment is selected at a given time.

The 72 and 73 instructions initiate the following sequence of transmissions: The address designated by the instruction (E addresses forward) is placed in the A register. The address in A is 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 72 or 73 instruction. If the comparison shows

equality, the program continues with the instruction at the second sequential address.

Instruction 74 initiates a single output operation by placing the 6-bit quantity E in the output channel.

Instruction 76 initiates a single input operation which reads one frame of data from the selected input device and stores it in the A register.

The External Function (EXF) codes which select the peripheral equipment consist of two part instruction words: the unit designator (most significant 6 bits) and the function designator (least significant 6 bits).

When a status request is issued, an input instruction must also be issued to read the response into the computer. Representative 12-bit response codes are listed below.

EXTERNAL FUNCTION CODES

CONSOLE EQUIPMENT

4102 reader
4104 punch

TYPEWRITER

4210 type output
4220 keyboard input
4240 request status

MAGNETIC TAPE SYSTEM

111X write
112X backspace
113X read forward
114X request status
115X rewind unload
116X rewind load
211X write-assembly mode
213X read forward-assembly mode

ALTERNATE MAGNETIC TAPE SYSTEM

12-X
22-X assembly mode

STATUS REQUEST RESPONSES

TYPEWRITER

0010 not ready

MAGNETIC TAPE SYSTEM

0001 BCD format
0002 not ready
0004 parity error
0010 illegal BCD
0020 end-of-file mark read
0040 end of tape or load point

Chapter Two
Operation

The computer and external equipment are placed in operation by procedures which incorporate loading and unloading paper tape and magnetic tape, making manual selections, and starting the program. This chapter is organized under the following headings:

Operation without pre-stored program
Operation with pre-stored program
shut-down of equipment
Tape loading procedures
Controls and indicators

Procedures are presented for putting into service the more common external equipment including the photoelectric reader, high-speed punch, typewriter and magnetic tape units. For details on other peripheral units, see the operation sections of the pertinent instruction books.

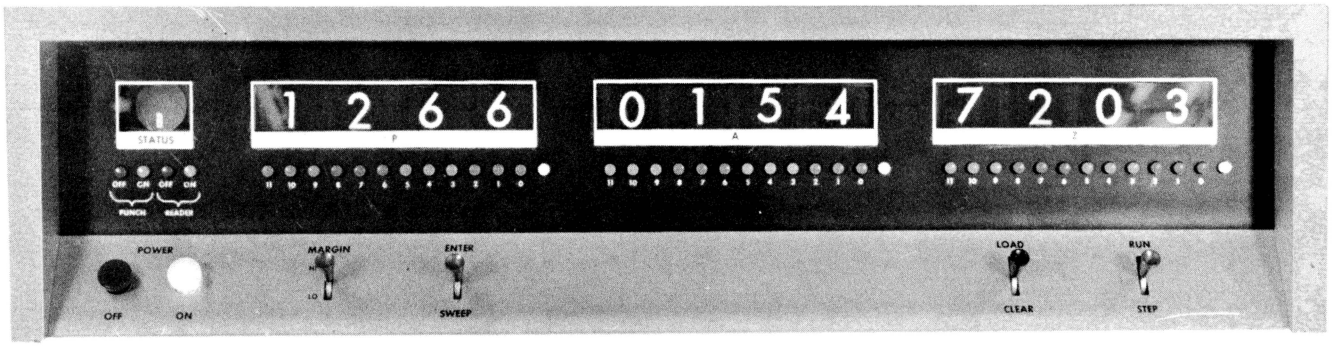


Figure 2-1. Computer Console

OPERATION WITHOUT PRE-STORED PROGRAM

When the computer is operated for the first time, or when a new program is to be used, it is first necessary to enter the program into storage by a paper tape loading routine. The procedures described below assume that the equipment has been turned on and is operating satisfactorily as determined by daily maintenance schedules.

1) Turn on the photoelectric reader from the console.
2) Insert the program tape in the reader. (Tape Loading Procedures)
3) Master Clear and insert starting address in P if tape load is not to begin at address 0000.
4) Raise Load-Clear switch to LOAD.
5) Raise Run-Step switch to RUN. Tape will now move through the reader and stop at the end of the loading routine.
6) Master clear.

The program now stored in the computer may be executed.

OPERATION WITH A PRE-STORED PROGRAM

1) Be sure that external equipment to be used by the program is properly prepared: paper in typewriter, margins properly adjusted, typewriter set to COMPUTER with power on, and so forth.
2) If magnetic tape is to be used, refer to Tape Loading Procedures for method of inserting tape in the C336-4/C336-5 tape cabinets.
3) Make certain that all registers are cleared as evidenced by the octal display on the console.
4) Set P to the location of the first program step if program is not to begin at at address 0000.
5) Place Run-Step switch to the position desired (usually RUN).

SHUT-DOWN OF EQUIPMENT

Removing power from the equipment is primarily a maintenance responsibility. The following procedures will, however, assure that the system is inoperative.

1) If photoelectric reader was used, remove paper tape from the reader and baskets; rewind the tapes.
2) Turn off reader motor (see Control and Indicators).
3) If high-speed punch was used, generate about a foot of leader by raising Load-Clear switch on console to LOAD. Tear off the tape and wind it up.
4) To unload magnetic tape, proceed as follows:

    a) Press Clear button.
    b) If computer has not rewound tape to unload position, press Rewind Unload button (tape must be stopped).
    c) Open front door.
    d) Turn file reel counterclockwise and wind all tape into file reel.
    e) Loosen file reel hub knob and remove file reel.
    f) Check if reel needs to be file protected and also if it is labeled adequately prior to storage.

5) Press Power Off button.

TAPE LOADING PROCEDURES

PHOTOELECTRIC READER

1) Release the tape release handle (figure 2-2) by turning it clockwise.
2) Move adjustable tape guide to the correct level for the paper tape to be used by pressing the handle and moving the guide so its spring catch engages the proper slot in the guide plate. For 7-level computer tape, this will be the second slot from the edge of the plate.

3) Insert the tape in the reader so that the highest level hole is toward the open side. Be sure the tape is inside of the tape guide and the leader is long enough to fit between the drive roll and the pressure roller.
4) Engage tape release handle by turning it counterclockwise.
5) Adjust tape lever switch to desired number of levels (5, 7, or 8). The upper levels not read will be transmitted as 0's.

The reader is ready to read tape in either the Load mode or PER Select.

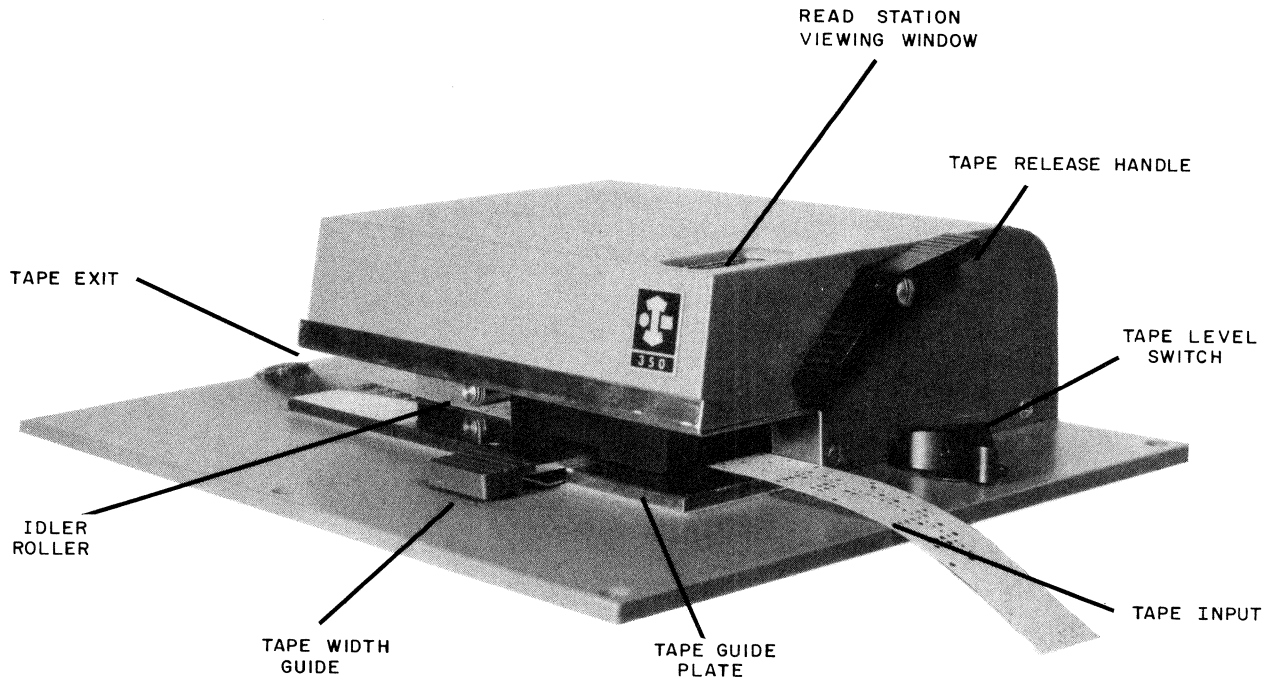


Figure 2-2. Photoelectric Reader

HIGH-SPEED PUNCH

The punch is behind the left-hand door of the computer cabinet. A thumb catch locks the punch drawer in place when the slides have been fully extended. This catch must be released to return the punch to its normal, recessed position.

To place a new roll of tape in the punch:

1) With the punch motor on, tear off the old tape and run it out of the punch block by pressing the tape-feed lever (figure 2-3).
2) Turn off punch motor at computer console to de-activate the out-of-tape buzzer.
3) Lift tape reel out of punch. Remove old roll of tape by unscrewing the X-shaped side of the tape reel.

4) Place new roll of tape on reel so tape unwinds counterclockwise and attach the X-shaped side plate.
5) Place tape reel in punch; thread tape through the loop on reel brake arm, through loop at front of punch, around the two rollers and into tape guide.
6) Slowly slide tape through the punch block. If it does not go through, tear off a piece of tape and try again. Lift tape tension lever so that the tape will start to go between the tape tension lever and the tape feed wheel.
7) Turn on punch motor.
8) Pull on end of tape and press tape feed lever. The tape should now feed automatically into the punch block. After about six inches of tape have been run out, the feed holes will be correctly punched.
9) Tear off punched tape even with the edge of the tape out guide.
10) Release the thumb catch and slide the punch back into the computer cabinet.
11) Raise the Load clear switch to LOAD. The tape should protrude from the opening in left side of cabinet.

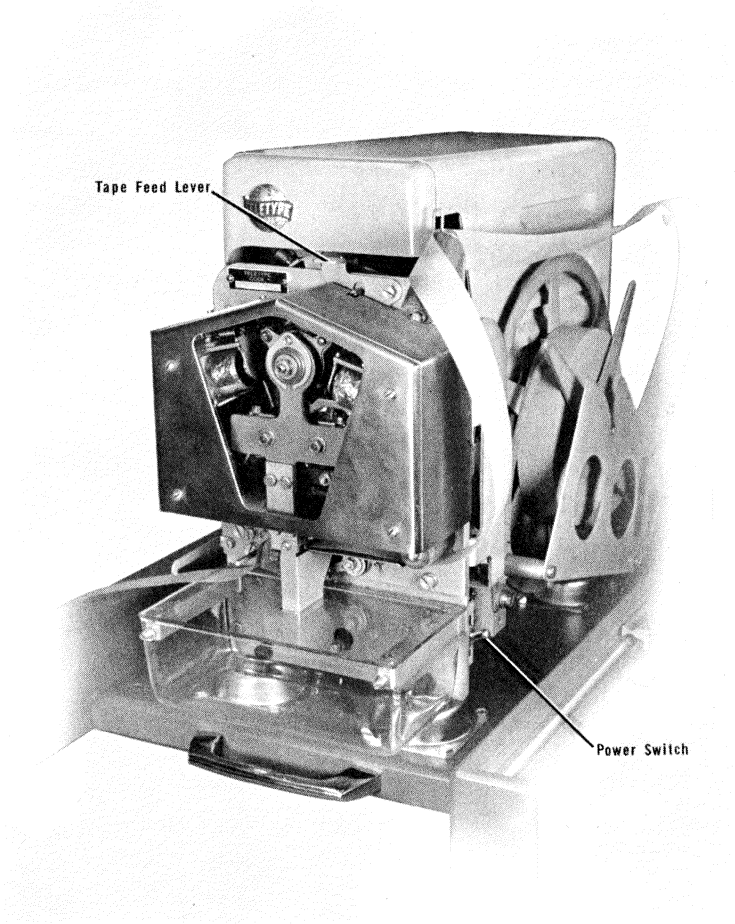


Figure 2-3. High Speed Punch

MAGNETIC TAPE CABINET

When the tape unit has been properly prepared for use, the following conditions will apply when a reel of tape is to be loaded (figure 2-4):

(1) The lower (takeup) reel is installed.

(2) The file reel tape has approximately 15 feet of metal-backed tape spliced to each end of the tape; load-point and end-of-tape reflective spot markers have been attached to the tape.

To load magnetic tape:

1) Open door to handler.
2) Check that file reel to be loaded has been file protected as necessary.
3) Mount reel on file reel hub and tighten hub knob. To insure proper reel alignment, push reel firmly against reel hub stop before tightening knob. If file protection ring has been removed from reel, see that Write Lockout lamp is on when reel is loaded; if lamp is not on, call maintenance. Check that pulling tape from reel causes it to rotate clockwise.
4) Open the tape access door.
5) Unwind approximately 6 to 8 feet of tape from the file reel.
6) Thread the tape through the sensing arms between the capstans and pinch rollers and then past the heads.
7) Wind the excess tape on the tapeup reel.

   NOTE: DO NOT STICK TAPE TO REEL.
   MAKE SURE TAPE WILL PULL OFF FREELY.
8) Close the tape access door.
9) Close the front door.
10) Press Clear button.
11) Set Unit Selection switch to position 1, 2, 3, or 4 to assign the unit a program selection number.
12) Press Binary/Coded button to select data format.
13) Press Rewind Load to move tape to load point. When Ready lamp turns on, unit is at load point and ready to be operated under computer control.

Figure 2-4. Tape at Unload Position

File Protection Ring

The back of the file reel has a slot near the hub which accepts the plastic file protection ring (figure 2-5). Writing on a tape is possible only when the reel contains a file protection ring, however, the tape may be read with or without the ring. Presence of a ring on a reel of tape is signalled by turn-off of the Write Lockout indicator immediately after the reel is loaded onto the tape unit. The ring should be removed from the file reel after writing is completed to avoid loss of valuable records through accidental rewriting.

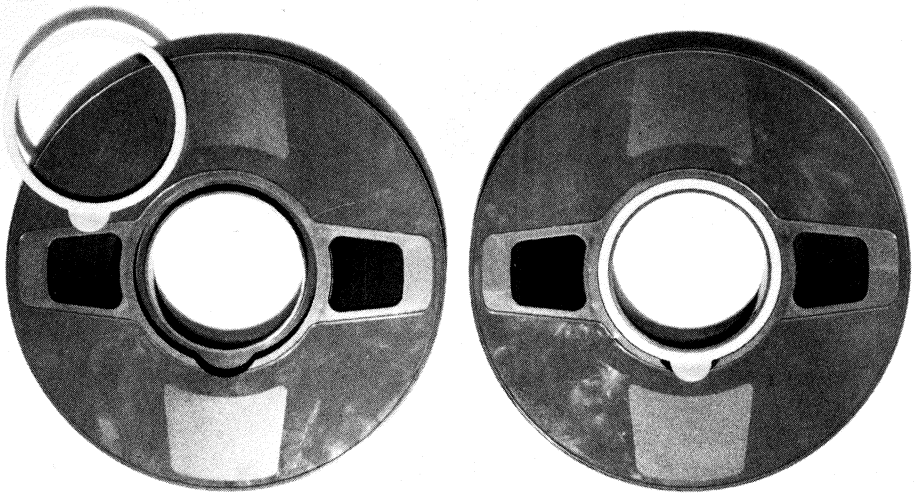


Figure 2-5. File Protection Ring

COMPUTER CONTROL CONSOLE

The control console (figure 2-1) consists of a switch panel and a display panel.

Switch Panel

Two large pushbuttons at the left of the switch panel apply and remove power from the computer cabinet. A light within the cap of the ON (blue) switch will appear when d-c voltages to the logic chassis are correct. Computer modes are controlled by four lever switches in the panel. From left to right these switches and their functions are:

Margin

This switch has three locking positions. It is primarily a maintenance tool and will not ordinarily be manipulated by the programmer-operator. When it is not in the center neutral position, the letter "M" will appear in the status window.

HI — This position makes the storage sense amplifiers less sensitive to marginal (noise) signals.

CENTER — (unmarked neutral) This position permits normal sense amplifier operation.

LO — This position increases the sensitivity of the sense amplifiers to marginal signals.

Enter-Sweep

This switch locks in both the up and down positions. The center neutral position is unmarked.

ENTER (up) — Quantities may be inserted into core storage by pressing the set buttons below the Z register display modules and then operating the Run-Step switch. Only one storage cycle is completed for each operation of the Run-Step switch. The storage location of the entered data is conditioned by the following factors.

If enter operation immediately follows a Master Clear,
storage location coincides with that shown in P.

When enter operation is NOT preceded by Master Clear,
storage location will be one greater than that shown in P at the time of entering.

SWEEP (down)

The entire contents of storage may be examined sequentially, but the data is not acted upon. Contents of sequential addresses appear in Z for every depression of the Run-Step switch to STEP. The first address to be read will be that indicated in P.

The sweep position may also be used to clear the entire contents of storage by holding down the clear button on Z while the computer is operated in the Run mode.

Load-Clear

This switch locks in the up (LOAD) position and has a spring return to neutral from the down (CLEAR) position.

LOAD (up)

This position conditions the computer to read paper tape from the photoelectric reader and store 12-bit words (two frames of tape) at successive storage addresses. The first storage address is determined by the contents of P at the time of the Load operation. In this mode, alternate (odd-numbered) frames of tape must contain a seventh-level hole. The reader starts as soon as the Run-Step switch is set at RUN, and will continue reading tape until the end of tape, or until two successive frames are read, neither of which contains a seventh-level hole. In the STEP position, two frames will be read for each depression of the Run-Step switch.

CLEAR (down)

This position is effective only when the computer is not running (Run-Step switch in neutral). The operative registers, A, P and Z are cleared, and an external clear signal is issued to peripheral equipment. Contents of storage are not affected by a Master Clear.

Run-Step

The Run-Step switch determines the manner in which instructions are to be executed. It locks in the up (RUN) position and has a spring return to neutral from the down (STEP) position.

RUN (up)

Instructions are executed at the normal high speed rate. The computer continue to run until an error stop (00, 77) occurs, or until the switch returned to neutral. In either case, the computer will complete the storage reference cycle before stopping.

STEP (down) — One storage reference cycle is completed for each depression of the switch to STEP. This permits a cycle-by-cycle execution of each instr tion. The storage cycles are performed at the high speed rate, howevei Switching from STEP to RUN will stop the computer in the same manner as though the switch were returned to neutral from the RUN position.

Display Panel

Three windows display the octal contents of the P, A, and Z registers. As shown in figure 2-1, each of these windows contains four display modules which are illuminated only when the computer is not in the run mode. Below each register window are 12 miniature push buttons which allow the operator to insert binary information into the 12-stage register. A white button to the right of each group is used to clear the contents of that register.

A fourth window, containing a single display module shows the computer status. Four miniature push buttons below the status window control power to the punch and reader. The significance of the displays within the four windows, left to right, is explained below.

Status

GREEN background	Computer running
RED background	Computer stopped

As evidenced by the symbol within the background, red may indicate:

ERR	Error
SEL	Computer is waiting for selected equipment to respond
M	Margin switch not in neutral position
OUT	Output instruction (73 or 74) prevails
IN	Input instruction (72 or 76) prevails
A, B, C, D	Storage reference cycle to be executed at next operation of Run-Step switch

P Register

Displayed numerals indicate octal address of next storage reference.

A Register

Display numerals indicate octal contents of the register.

RED background	Main timing fault

Z Register

Displayed numerals indicate octal contents of the register.

GREEN background — Displayed contents of Z is a program step

TYPEWRITER CONTROL PANEL

This panel on the typewriter cabinet to the right of the IBM typewriter contains two lever switches and two indicator lights.

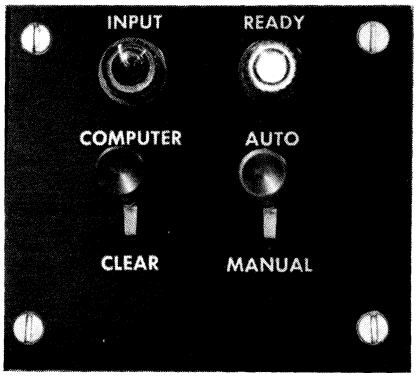


Figure 2-6. Typewriter Control Panel

Operation Mode

This switch controls the mode of the typewriter (TWR). It locks in the up (COMP) position and has a spring return to the center position when pressed to CLEAR.

COMP (up) — Puts the typewriter under control of the computer for input or output operations.

MANUAL — This unmarked center position is used when the typewriter is to be removed from computer control for typing. If a TWR input or output operation is initiated when the switch is in this position, the SEL light in the computer status window will be illuminated until the lever is moved to COMP. Should the lever be moved from COMP to MANUAL during a TWR operation, the status window will display IN or OUT and the computer will stop until the switch is returned to COMP. The computer may be re-started with the Run-Step switch.

CLEAR — This momentary position resets TWR control after power has been applied to TWR cabinet. This position will also drop a TWR EXF selection for input or output.

Input Disconnect

This switch determines the manner in which the input disconnect signal is sent to the computer. It locks in the up (AUTO) position, and has a spring return to neutral from the down (MANUAL) position. The center position (unmarked) has no effect.

AUTO — In this position an input disconnect signal is sent to the computer each time a carriage return is typed during an input operation. The carriage return code (45) precedes the disconnect signal.

MANUAL — Pressing this switch during input sends a disconnect signal to the computer immediately. It may be used at any time to terminate an input operation.

Ready Light

This light signifies that power has been applied to the typewriter control circuits and the typewriter may therefore be selected.

Input Request

This light signifies that the computer is requesting input information from the typewriter.

MAGNETIC TAPE CABINET

The switch panel for the magnetic tape cabinet is shown in figure 2-7. Table 2-1 lists the switches and indicators from left to right, and describes the function of each. For more detailed information on the tape unit operation refer to the Magnetic Tape System instruction book.

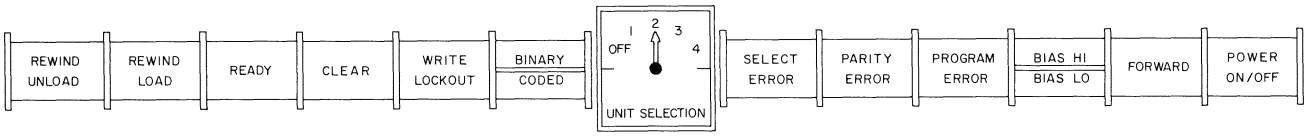


Figure 2-7. Magnetic Tape Switch Panel

TABLE 2-1. FUNCTION OF MANUAL CONTROLS & INDICATORS (MTU)

Name		Function
REWIND UNLOAD	S*	Moves tape to unload position and interlocks it to place the unit in a not ready condition. Control duplicates a programmed rewind unload instruction.
	I	Lighted by a switch or programmed rewind unload instruction; remains on until CLEAR is pressed.
REWIND LOAD	S	Moves tape to load point. When arriving at load point from unload position, ready light indicates a unit ready condition. Control duplicates a programmed rewind.
	I	Light indicates tape moving to load point.
READY	I	Light is on at all times tape is available for computer use.
CLEAR	S	Stops (immediately) tape motion and clears TCU. New manual or programmed selections are necessary to reselect tape unit and/or operation required.
WRITE LOCKOUT	S	Pressing switch prevents writing on unit; pressing switch again releases write lockout condition.
	I	Indicates write lockout condition.
BINARY/CODED	S	Selects data format.
	I	Indicates switch selection.
UNIT SELECTION	S	5-position switch: 1, 2, 3 and 4 provide unit designation. OFF disconnects unit from TCU; places unit in not ready condition.
SELECT ERROR	I	On if duplicate UNIT SELECTION switch settings exist or when switch is in OFF position.
PARITY ERROR	I	On if parity error exists on unit. Turned off by new EXF instruction (non status request) on any unit.
PROGRAM ERROR	I	On when program error exists on unit. Turned off by manually correcting error condition.
BIAS HI	I	Indicates bias level of read amplifier set to compensate for higher than normal tape signals.
BIAS LO	I	Indicates bias level of read amplifiers set to compensate for lower than normal tape signals. (Bias switch mounted at rear of cabinet of main TCU chassis-20100.)
FORWARD	S	Moves tape forward.
	I	Indicates switch action
POWER ON-OFF	S	Press to turn power on; press again to turn power off.
	I	On when power is on.

* S - Switch I - Indicator

Chapter Three
Principles of Operation

The Principles of Operation section emphasizes circuit logic rather than electrical operation. The computer is constructed from a great many circuits and stages which are electrically similar. The basic electronics involved are presented in Appendix D (Volume 2), the Building Block. The functions of the circuits are described in this chapter.

The computer may be explained conveniently by discussing it in four major catagories:

Control Section -- determines how the computer 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 computer, 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 (Volume 3) A graphic presentation of the logical relationships delineated in the equation file.

File of Equations (Volume 4) 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.

Five additional appendixes will be found useful from time to time:

Appendix A A glossary of terms with special or unusual meaning.

Appendix B An elementary discussion of number systems.

Appendix C	Instruction timing charts pinpointing the steps involved in execution of each instruction.
Appendix E	Schematic diagrams representing every card type used in the computing system.
Appendix F	Information for installing the computer; includes weights, power consumption, and so forth.

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

CONTROL SECTION

All computer 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 computer has seven registers. Three of these, A, P, and Z, are operational in that they are displayed on the console panel and their contents may be modified by manipulation of the associated bit-stage push buttons. The four transient registers B, S, O, and F, are important only to the internal operation of the computer and are not indicated on the console.

A Register (12 Bits)

The A register is the principal arithmetic register. For most arithmetic operations A, together with the borrow pyramid, operates as a 12-bit subtractive accumulator. The arithmetic processes of addition and subtraction are performed modulus $2^{12}-1$. 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 B 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.

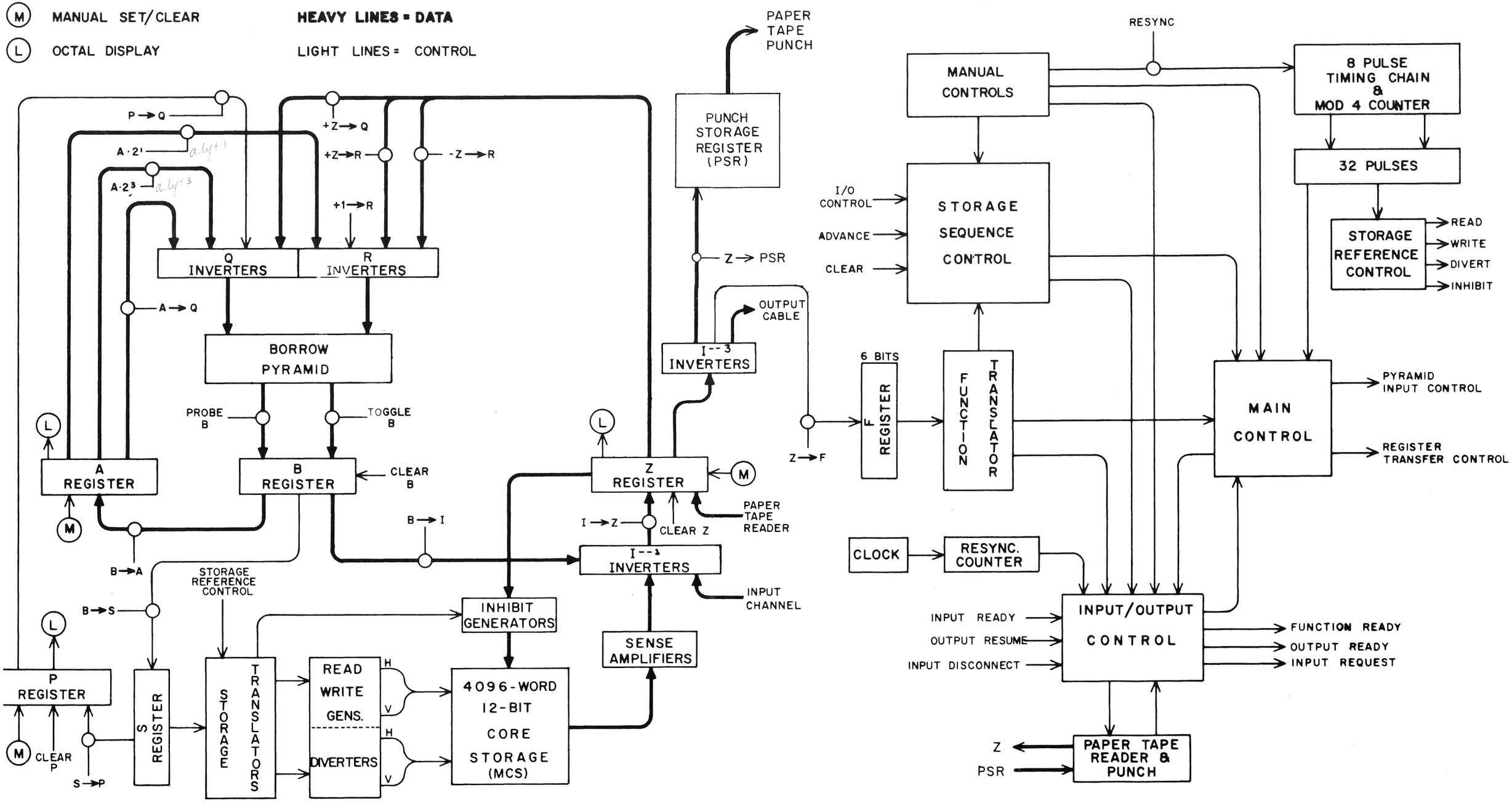


Figure 3-1. Block Diagram, C310-2 Data Processor

Outputs from A go to the borrow pyramid 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").

Z Register (12 Bits)

The Z register performs a variety of functions. It acts as a storage restoration register for 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 it is the sole means of communication with core storage.

Because the number of transmission paths through Z exceed the input and output pin capacity of any one FF card, each stage of Z has two inverters, designated by the symbol I. One inverter leads to the set input, and the other comes from the clear output of the Z-stage FF.

Controls for Z are divided into two groups -- upper (Z_U) and lower (Z_L). This is necessary because the F portion of the instruction word (upper 6 bits) is an entity apart from the E portion (lower 6 bits). This is best exemplified by a no address instruction, where E is used as an operand. In this case E is sent to the pyramid and F is sent to the F register.

P Register (12 Bits)

The P register, program address register, holds the address of the current instruction. 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, the E portion of the instruction word is added to P.

In addition to the manual set and clear inputs, P has one set input from the S register. The output from P goes to the pyramid via the Q inverter rank.

B Register (12 Bits)

The results of all operations involving the pyramid are formed in the auxiliary arithmetic register, B. The contents of B are sent under program control, to A, Z, or S. The state of each stage of B during pyramid operations is determined by one or more levels within the pyramid. Functions controlling B, labeled Clear, Toggle, and Probe, are explained in the Arithmetic Section.

S Register (12 Bits)

The S register, storage address register, 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 B. Outputs go to P, and to the storage translators for implementing the actual address selection.

F Register (6 Bits)

The Function register holds the upper 6 bits of the instruction word throughout the execution of that instruction. The inputs to F come from Z via the I-inverters. Outputs from F go to the Function Translator.

O Register (8 Bits)

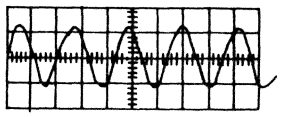
The O register, punch storage register, holds data to be sent to the high-speed punch. Each stage consists of a FF with an output controlling (through an L-card) a corresponding data-level punch puller. The FFs are set from the lower eight stages of Z (via I^{--3} inverters), and are cleared by the punch control logic.

MASTER CLOCK

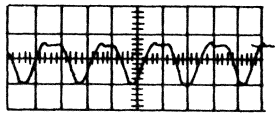
The master clock which provides the timing pulses used throughout the computer consists of nine interconnected oscillators, each of which is contained on a type 01 card. Each oscillator operates at 2.5 megacycles and produces two sine waves, even and odd, 180 degrees out of phase with one another. Each oscillator card has two logic symbols. For example, C^{--0}, represents the even phase, and C^{--1}, the odd phase. The odd phase is always the higher order symbol.

One pin on each of the nine 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 01 card. Since these outputs draw some current, each oscillator is loaded as symmetrically as possible.

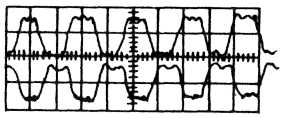
Figure 3-2 shows the waveforms of the oscillator card and the square wave produced after that wave passes through one inverter. The asymmetrical shape of the wave at the oscillator output pin is due to the bias used in clipping the sine wave peaks. Only the negative (top) portion of the wave is used in gating. Therefore the clock pulses in the computer are 0.2 microseconds in duration.



A. OSCILLATOR TEST POINT
(VERTICAL: 5v/cm;
SWEEP: 0.2 μsec/cm)



B. OSCILLATOR OUTPUT PIN
(VERTICAL: 5v/cm.;
SWEEP: 0.2 μsec/cm)



C. COMPARISON OF PHASES AT THE
OUTPUT OF SLAVE INVERTER
(VERTICAL: 2v/cm.;
SWEEP: 0.2 μsec/cm)

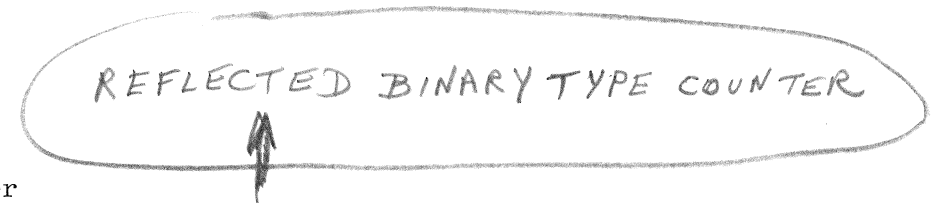
Figure 3-2. Master Clock Oscillator Waveforms

Two oscillators (C^{007} and C^{012}) drive a resynchronizing system which times asynchronous signals, such as developed by manual switches and external equipment communication lines, to the main computer. This system is discussed in the section dealing with Input-Output controls.

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 input-output operations since the external equipment is asynchronous with the computer. Any operation using these devices must be initiated by the timing chain, however.

The chain consists of eight cascaded control delays, as shown at the top of figure 3-3. For a single input (start) pulse, the chain emits a string of eight pulses, spaced 0.2 microseconds apart. A recirculation control allows the V^{007} pulse to re-enter the H^{000}-V^{000} control delay, thereby extending the number of timed pulses emitted. A string of 32 pulses (four 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 computer.



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 ~~conventional two-stage, double-rank counter~~, inputs to the first and second ranks are controlled by V^{005} and V^{000} of the timing chain (figure 3-3 lower left). Excursions counter outputs go to the primary timing control.

Primary Timing Control

The primary timing control is a series of translators which reflect the state of the excursions counter. The four passes (or quarter cycles) of each storage reference cycle are represented by J^{00-}, J^{01-}, J^{02-}, and J^{03-} (figure 3-3 lower right). The quarters are designated 1, 2, 3, 4 or early 1, early 2, etc. The early quarters begin at pulse time 5 (V^{005}) on the previous pass; the pulse from J^{010} (early 2) actually begins at time 5 during the first pass. This arrangement is shown clearly in figure 3-4, Main Control Timing.

The part these cards play in computer timing is described more completely in the section on storage control.

STORAGE REFERENCE CYCLE

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

The storage reference cycle consists of four phases: divert, read, write, and 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.

The principal function of each of the four quarters of the storage reference cycles is listed below. In general, one function is accomplished during each quarter of a cycle.

Read Address A cycle

1) Transmit E to the S Register for reference address.
2) Read 12-bit address from storage.
3) None.
4) None.

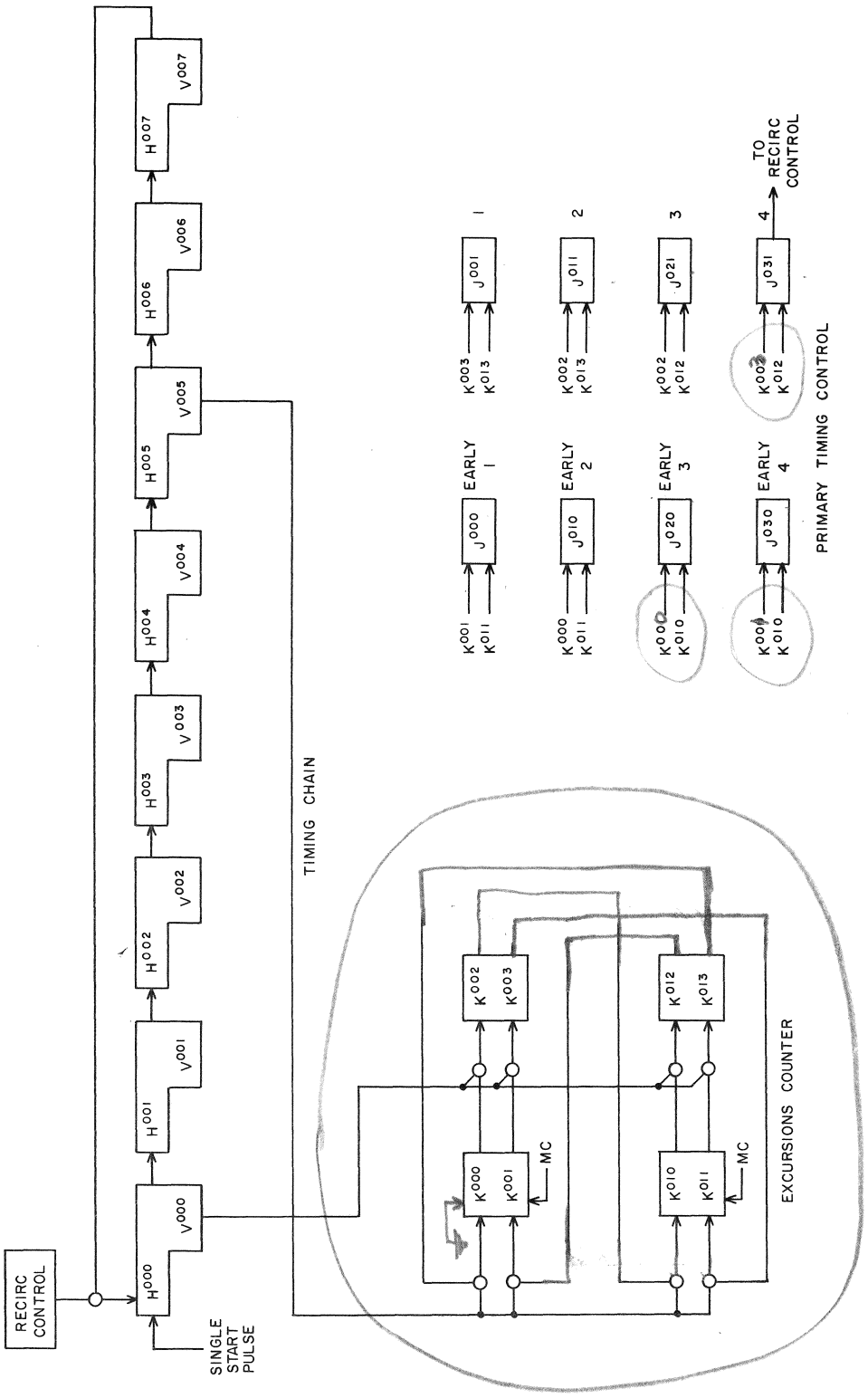


Figure 3-3. Main Timing Controls

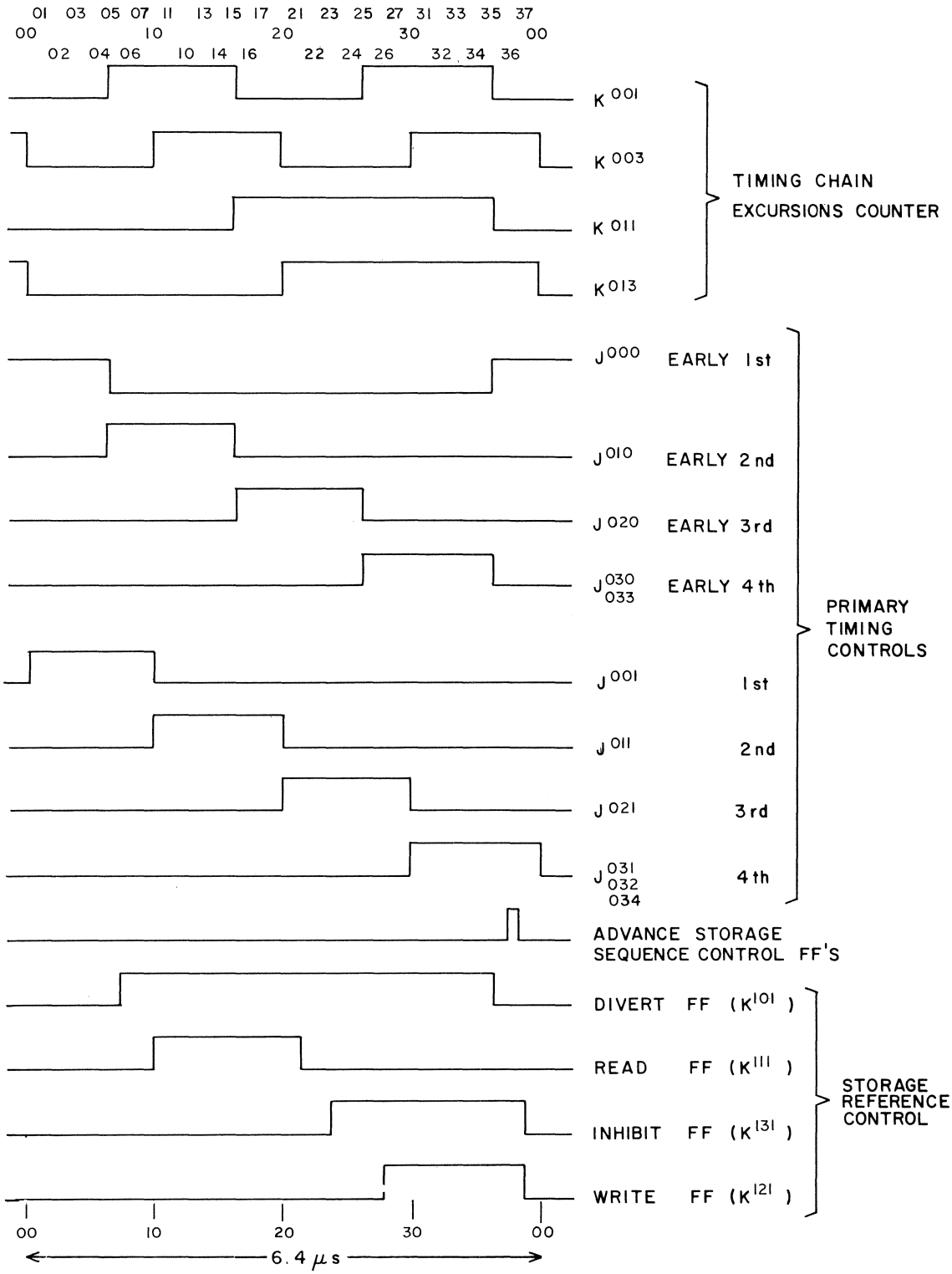


Figure 3-4. Main Control Timing

Read Operand or Address B cycle	1) Form address for reference. 2) Read operand or address from storage. 3) Perform arithmetic or logic operation. 4) Initiate external function. (instruction 75 only)
Write Operand C cycle	1) Transfer Z to A (input instruction only). 2) Transfer A to Z for recording in storage. 3) None. 4) None.
Read Instruction D cycle	1) Form address for reference. 2) Read instruction from storage. 3) Transfer F to function register. 4) Execute n (no address) instructions.

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 four-level translation is shown in figure 3-5. Other translations may involve fewer levels.

The first-level translators are made up of three groups, each of which performs a translation of two stages of F. The F^{10-} cards translate the contents of the lower two stages, the F^{11-} cards the middle two stages, and F^{12-} the last two stages. Higher-level translations involve: (1) examination of these cards, and (2) in certain cases, examination of elements in primary timing control and/or the storage sequence control. Examination of F^{213}, (figure 3-5) demonstrates the effect of the first-level translations upon higher-level operations.

Satisfaction of the AND input to F^{213} depends upon the presence of all three of the inputs. The resultant case, then, may be defined as follows (where X represents insignificant bit positions):

F^{123} - 11X XXX
F^{112} - XX1 0XX
F^{103} - XXX X11
1 1 1 011 (octal 73)

From this operation it is seen that F^{213} produces a "1" output except on a 73 instruction ($\overline{73}$) where the bar over the number expresses the Boolean NOT function.

F^{225} ($\overline{-Z \rightarrow R}$) inhibits the -Z input to the borrow pyramid on instructions which do not call for a subtract operation. (See the discussion of the pyramid in the arithmetic section.) The inputs to F^{225}, labeled 1 and 2, delineate those cases using the subtract functions as explained in the figure.

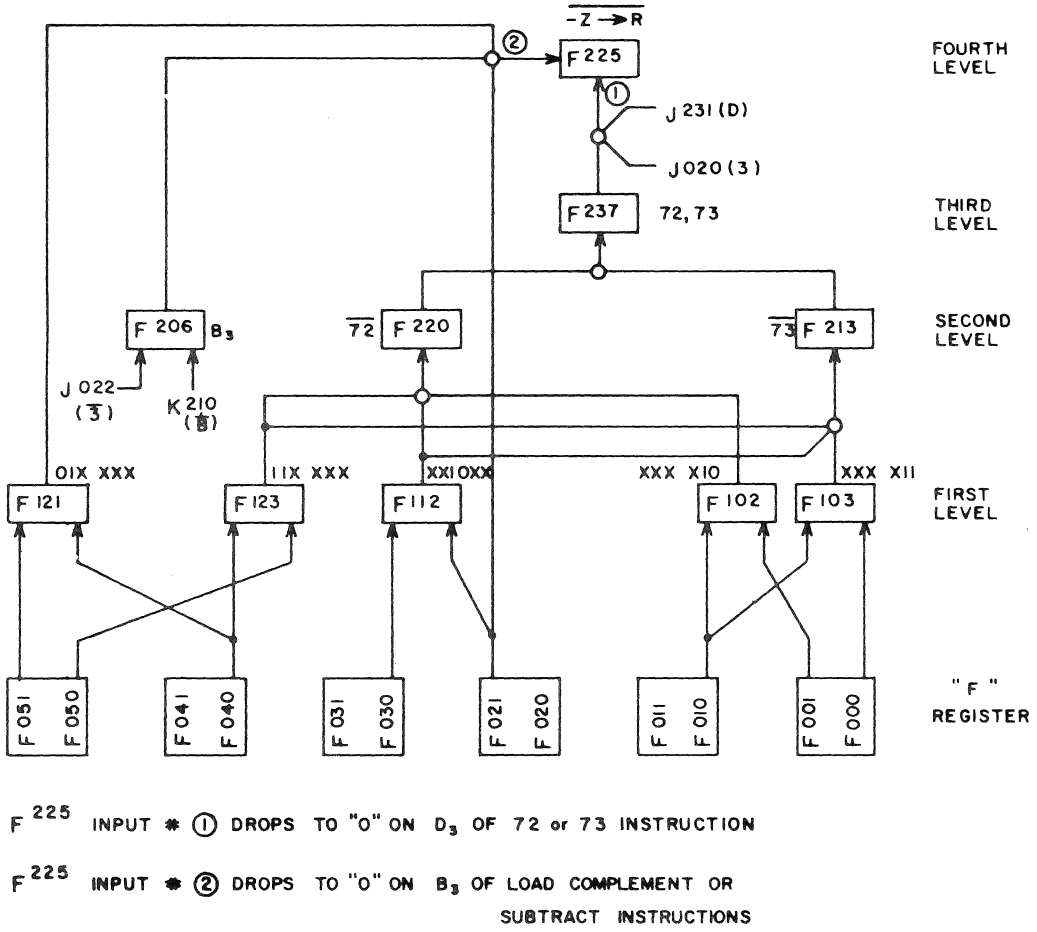


Figure 3-5. Typical Function Translation

OPERATING CONTROLS

Four lever switches for controlling the computer are provided on the switch panel of the control console.

Run-Step Switch

The Run-Step switch (figure 3-6) starts the computer 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 computer 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 sync 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. To prevent double pulses to the timing chain, the Neutral FF is cleared by the output from the third element in the chain (TC_3) which occurs 0.6 microseconds after initially starting the chain, or considerably before the succeeding even sync pulse.

Recirculation in the timing chain is maintained by recirc (J101) which is energized at the same time as the start element and 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 pulse from TC_5. This enables the left-hand AND input to the Recirc card, which is de-energized when the timing chain begins its fourth 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 four 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 computer for loading from the photoelectric 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 computer is not in the Run mode.

LOAD — When LOAD is in effect, the Not Load elements produce "0's", and the load-or-enter card is forced to a "1" because of its disabled AND input.

CLEAR — This position is effective only if the computer 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.

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 Z register (usually preceded by a Master Clear). Putting the computer in either Run or Step loads the quantity at Z into storage at the address specified by P. Then Z is cleared and P is advanced by 1 in preparation for the next manual entry into Z. 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 either in Run or Step. SWEEP may also be used to clear storage by holding down the Clear Z button while the computer operates in the Run mode.

Margin Switch

This switch controls the reference voltage for incoming signals to the sense amplifiers in core storage. It is normally used as a maintenance device, at other times it is left in the center or neutral position. Each of the three positions is locking. The circuit is described in the Storage section.

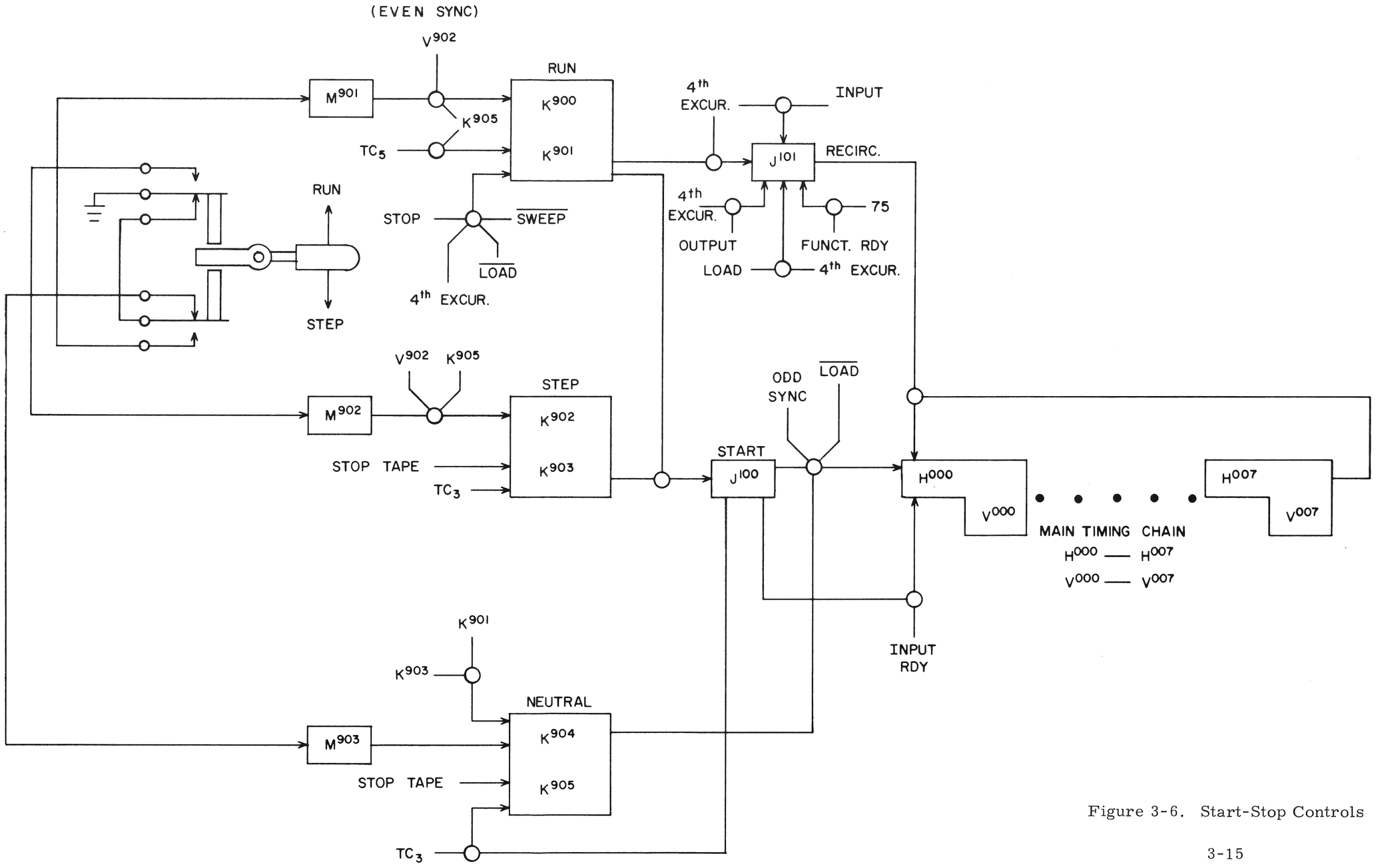


Figure 3-6. Start-Stop Controls

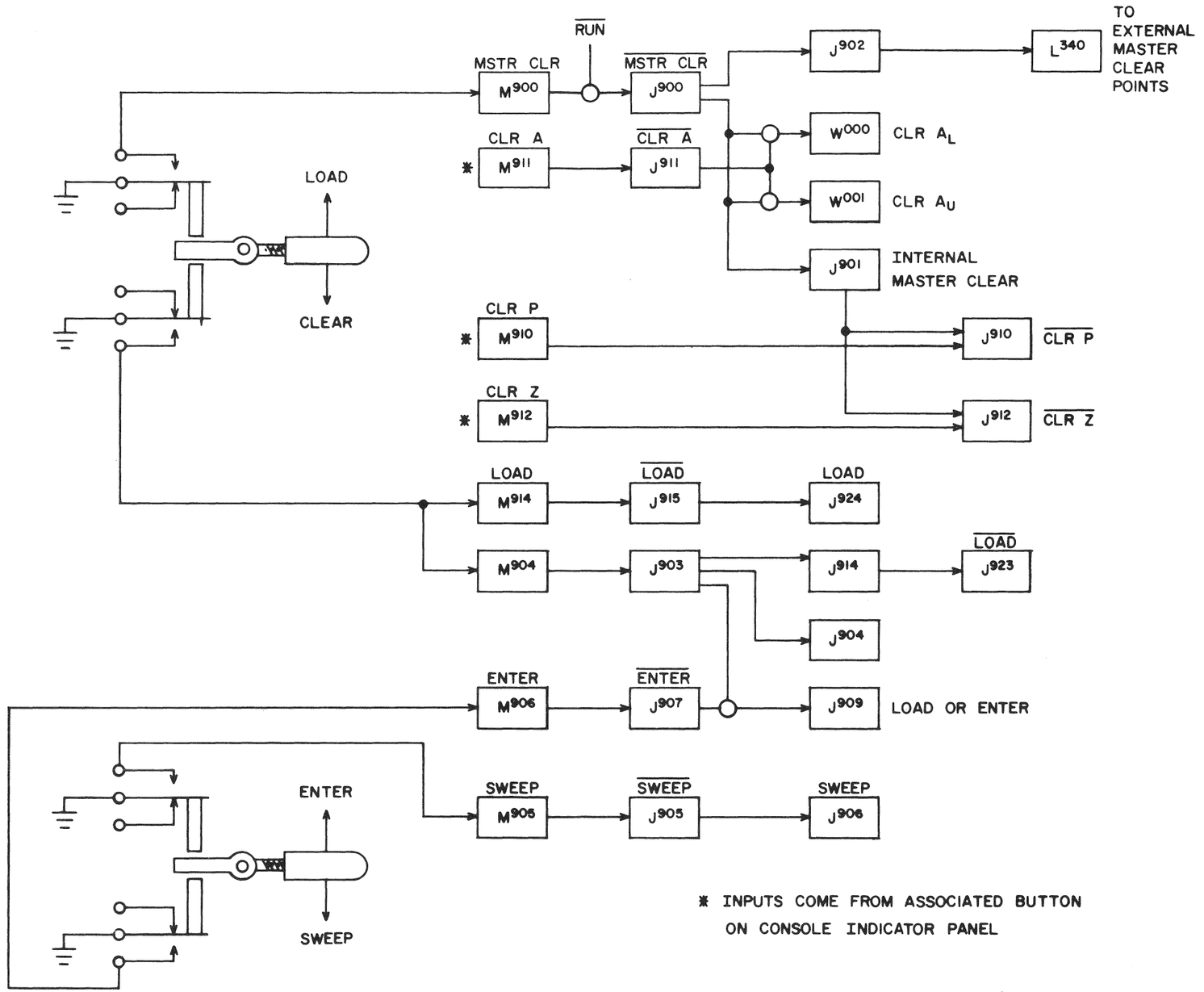


Figure 3-7. Load/Clear, Enter/Sweep Controls

HI — In this (up) position, the +20 volts applied to a voltage bus leading to pin 1 of each sense amplifier card make the amplifier less senstive to marginal signals from core storage.

NEUTRAL — This (center) position is unmarked. Since it provides no connection to the voltage bus, the reference voltage is derived from the normal sense amplifier circuits.

LO — In this (down) position, the -20 volts applied to the bus make the sense amplifiers more sensitive to marginal signals from storage.

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 due to a spurious pulse in the timing chain after power application. The logic and electrical circuits are shown in figure 3-8.

The pulsating d-c output from a bridge rectifier is connected to the input pin of a standard input amplifier. With no rectifier output, point ① is held at ground by the voltage divider consisting of two 2.2K resistors. Output from the bridge causes M^{916} to vacillate between ground and -3v during the decay of each -18v (peak) pulse. Since a 5ms delay is imposed between M^{916} and J^{214}, the negative pulses from the former will not switch the circuit, inasmuch as they occur at 8ms intervals. The resultant "1" from J^{214} holds J^{211} to a "0" except when the AND is disabled by a logical stop from the computer. No input to the Main Timing Fault (MTF) FF is possible under these circumstances.

Upon a power failure, 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. The fault light in the status panel will be visible when power is restored.

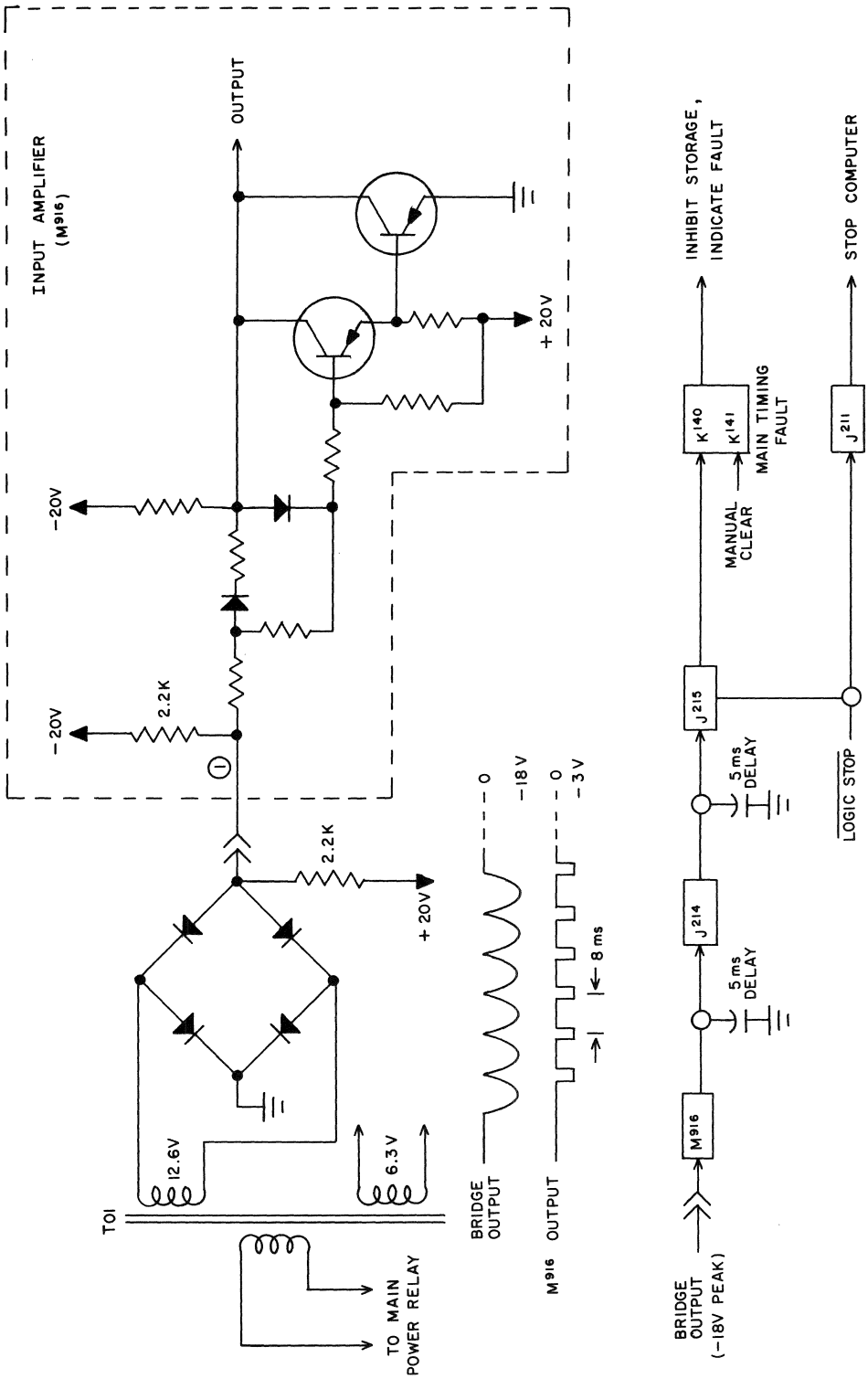


Figure 3-8. Non-Volatile Storage Circuits

A second delay at the input to J^{215} operates upon application of power by holding that input to "0" for 5ms, thereby ensuring the setting of MTF. Full d-c power is realized approximately 1ms after power is turned on, but nearly 45ms are required for the supply voltage to drop to zero. The capacitive delay is described in the Input-Output section of this chapter.

STORAGE SECTION

The storage section consists of a 4096-word, 12-bit memory employing magnetic toroids. The toroids, or cores, have an essentially square hysteresis curve. The memory cycle is 6.4 microseconds long. Information is read from memory 1.8 microseconds after the beginning of the cycle to allow time for the reference address to be formed. The read phase which takes 1.8 microseconds, transfers the information from storage address to the storage register, Z. The information in memory is destroyed, but is automatically restored by rewriting during the write phase, which is initiated 4.4 microseconds after the start of the cycle and lasts 2 microseconds.

Each bit of the 12-bit word is represented by a separate memory plane containing 4096 cores in a 64 x 64 array (figure 3-9). The 12 planes are stacked to constitute a memory plane assembly. Each core has four continuous wires running through it (figure 3-10). 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. These wires, called the horizontal and vertical drive lines, traverse each of the 12 planes. In this manner, all 12 bits of one word (the 12 cores at the corresponding coordinate location in each of the memory planes) are addressed simultaneously.

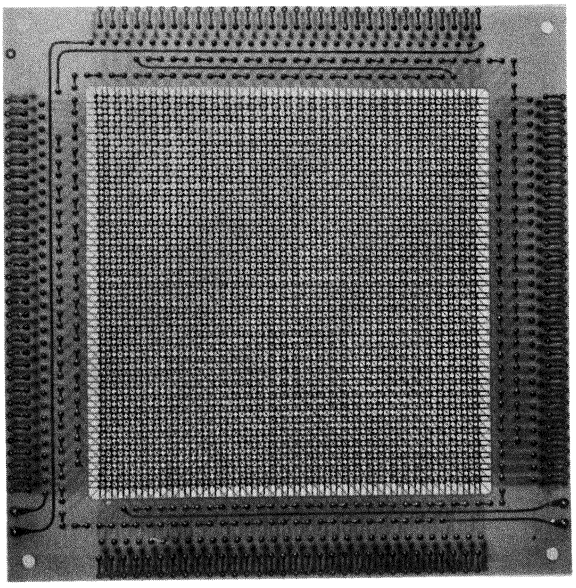


Figure 3-9. Memory Plane

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 to produce a pulse which is recognized during the read phase. The read pulse switches a core from the "1" state to the "0" state, and the write pulse switches the core from a "0" to a "1".

The third wire, the inhibit line, threads each core twice and is in parallel juxta-position with the vertical and horizontal drive lines. Since the drive lines influence all 12 cores in a selected word simultaneously, it is necessary to negate the effect of those lines in any bit position which must remain in the "0" state. The 12 inhibit lines are connected to 12 inhibit generators which are controlled by the state of a corresponding bit position of Z, the storage restoration register. If, for example, Z contains a "0" for bit 04, the appropriate inhibit generator is energized. Current through the inhibit lines is equal to half that in either drive line, but opposite to it in polarity. As a result, the effect of the 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".

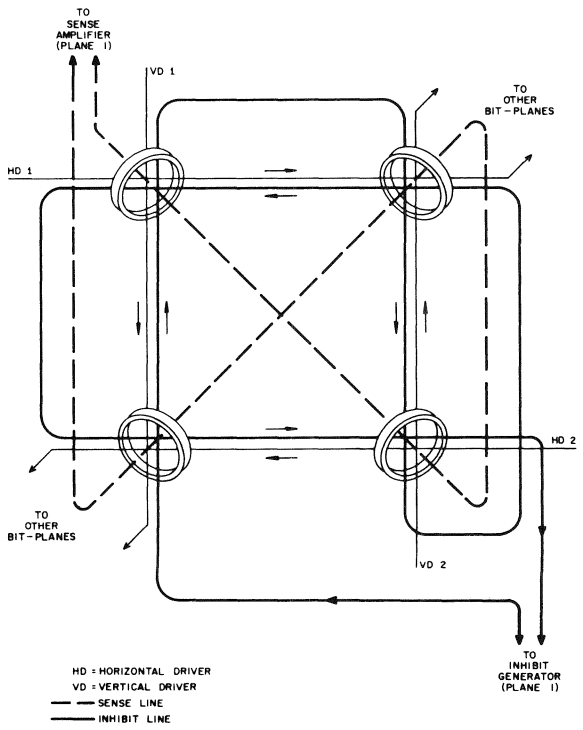


Figure 3-10. Four-bit Magnetic Core Matrix

The fourth wire, intersecting all cores of one bit-plane, 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 the Z register, where it is held for restoring memory. Writing information in memory is accomplished by blocking the signals from the sense windings to Z and enabling a transmission from the B register to Z. 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 board horizontally and vertically and terminate in tabs at either side of the edge of the phenolic memory plane 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 12 memory planes comprising the memory plane assembly, or stack, are held together by bolts passing through the four corners of each, and are separated by tubular aluminum spacers. An aluminum plate at the back and a plexiglass plate at the front protect the stack from physical damage. The stack is located in the upper right-hand section of the logic chassis, when viewing the chassis from the card side (back of machine).

PROPERTIES OF THE MAGNETIC CORE

The magnetic properties of a core are represented by its hysteresis diagram (figure 3-11), 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 drive 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.

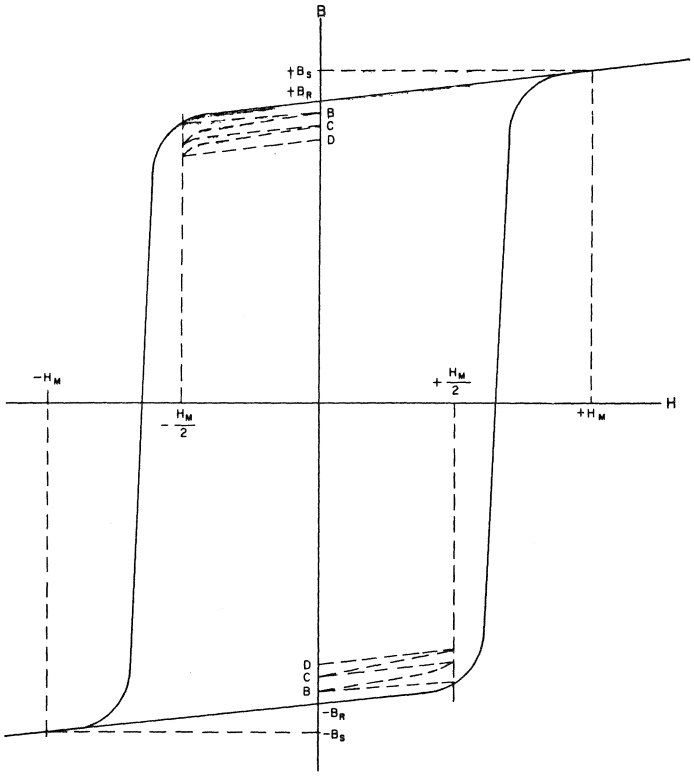
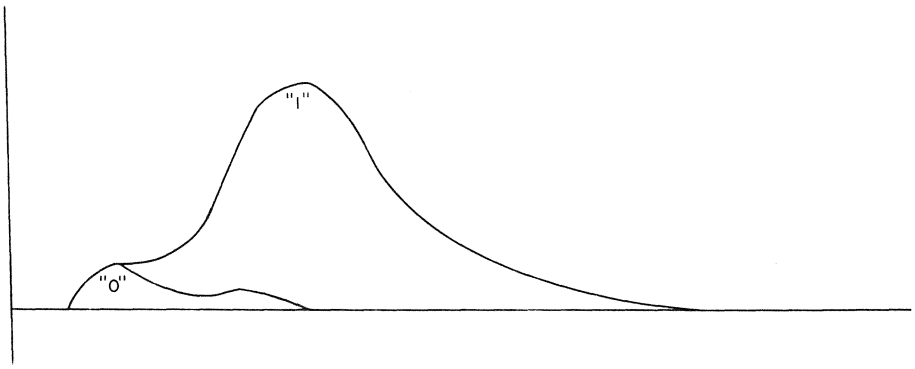


Figure 3-11. Typical Hysteresis Diagram

The basic memory cycle is composed of half-amplitude pulses, each capable of producing a field intensity of $H_m/2$. A half-amplitude pulse is insufficient to switch the core; instead the flux density returns to the residual value, or a slightly lower value, 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 somewhat and then returns to a slightly lower residual value, such as B. Since the core is now operating on a smaller loop, further half pulses reduce this remanent flux again, but 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). During the period that 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 merely 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 computer. A general discussion of the manner in which this selection is made will follow; 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 32-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 12 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 shown in figure 3-12, only operations common to all cycles appear. "EK" refers to the excursions counter. The pages following figure 3-12 present each of the four types of SR cycles, A, B, C, and D. The execution of the various functions listed opposite each pulse time depends upon the operation code held in the F register.

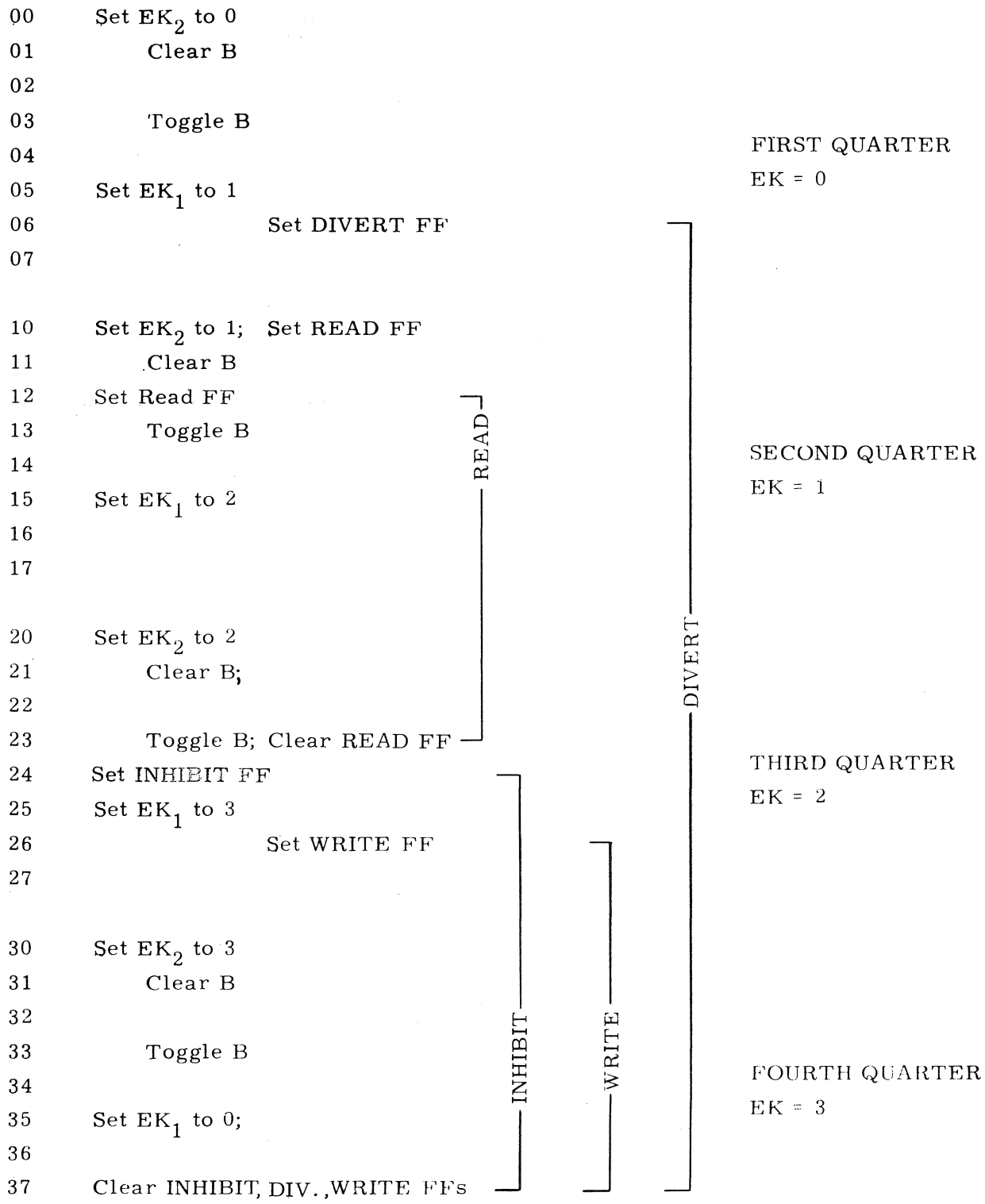


Figure 3-12. Basic Storage Reference Cycle

Read Address, Cycle A

1. Enter Z_L in S for reference
2. Read address
3. None
4. None

00 En Z_L
01
02
03
04 Probe B
05 B→S
06 ~~B→S~~
07
10 En MCS→I
11
12
13
14 Probe B
15
16 Clear Z
17 I→Z
20
21
22
23
24
25
26
27
30
31
32
33
34
35
36
37

Note: En = Enable to Pyramid

Read Operand or Address, Cycle B

1. Form reference address
2. Read operand
3. Execute
4. Initiate external function

00 En Z En + Z_L En - Z_L En P En +1 (on load or enter only)
01 Clear B
02
03
04
05 B → S
06 ~~B → S~~
07
10 En MCS → I
11
12
13
14
15
16 Clear Z
17 I → Z
Except on load or enter
20 En Z En - Z_L En Z En A En + 1 (except on load, enter, 72 or 73)
21
22
23
24 Probe B Disable Probe B
25
26 B → A
27 ~~B → A~~
30
31
32
33
34 Initiate Output
35
36 Clear Z (72, 76, Load or Enter)
37

Note: En = Enable to Pyramid

Write Operand, Cycle C

1. Transfer Z to A (76 instruction)
2. Transfer A to Z for recording in storage (except 72, 73, 76)
3. None
4. Write

00	En Z (76)	I→Z	(76 and $\overline{\text{PER}}$ or Load)
01			
02			
03			
04	Probe B		
05	B→S (72+73)		
06	~~B→S (72 + 73)~~ B→A (76)		Clear Z (72 $\overline{\text{PER}}$)
07	~~B→A (76)~~		I→Z (72 $\overline{\text{PER}}$)
10	En MCS → I (73 or 74)		En B→I (except 72, 73, 76)
11			
12		76?	
13			
14	Probe B		
15			
16	B→I effected	Clear Z	except on 72 or Load or Enter
17		I→Z	
20			
21			
22			
23			
24			
25			
26	Initiate Write		
27			
30			
31			
32			
33			
34			
35			
36	B→A (72+73)		
37	~~B→A (72 + 73)~~		

Note: En = Enable to Pyramid

Read Instruction, Cycle D

1. Form reference address
2. Read instruction
3. Set F
4. Execute

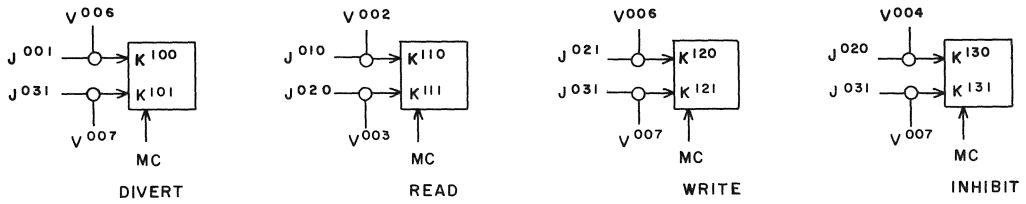
00	En P	(En +1)*	En + Z_L	En - Z_L						
01	Clear B									
02										
03	Toggle B									
04	Probe B									
05	B→S									
06	~~B→S~~									
07										
10	En MCS→I									
11	Clear P									
12	S→P									
13										
14										
15	En - X (72, 73)									
16	Clear Z									
17	I→Z									
20										
21										
22	Clear F									
23	Z→F									
24										
25										
26	B→A (72+73)									
27	~~B→A (72+73)~~									
30	Initiate Output	En Z	EN + Z_L	En - Z_L	En A	En A . 2^3	Initiate Input			
31										
32										
33	Toggle B									
34	Probe B									
35										
36	B→A									
37	~~B→A~~									

* Except after MC or Clear P

Note: En = Enable to Pyramid
A. 2^3 = Shift A three places left

Storage Reference Control

The four 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 3-13). Each set or clear input consists of an AND between a primary timing control element J, which calls out the quarter cycle involved, and a V element from the main timing chain. 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.



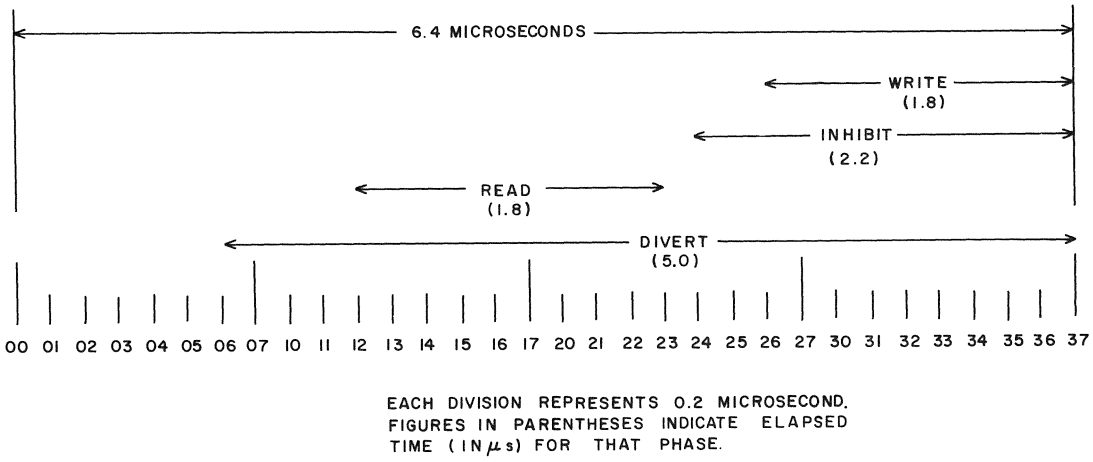


Figure 3-13. Storage Reference Control

Storage Sequence Control

If each of the 64 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 five patterns, called Storage Reference Sequences, as listed below.

D	No Address
DB	Direct or Relative Address
DAB	Indirect Address
DBC	Replace or Store Instruction, Direct or Relative Address
DABC	Replace a Store Instruction, Indirect Address

The five sequences function in the following manner:

D	Perform the indicated operation, using the 6 bits of E as an extended 12-bit operand wherein the most significant 6 bits are zeros.
DB	Perform the indicated operation using the operand stored at address E.
DAB	Perform the indicated operation, using the operand stored at the address specified by the contents of the intermediate address E.
DBC	Perform the indicated operation using the operand stored at address E, then replace the operand with the results.
DABC	Perform the indicated function, using the operand stored at the address specified by the contents of intermediate address E, then replace the operand with the results.

From these patterns it is possible to extract seven progressions.

1) D→B
2) D→A
3) A→B
4) B→C
5) B→D
6) C→D
7) D→D

An eighth progression, D→C, is necessary when storing incoming data, and must be added to complete the list.

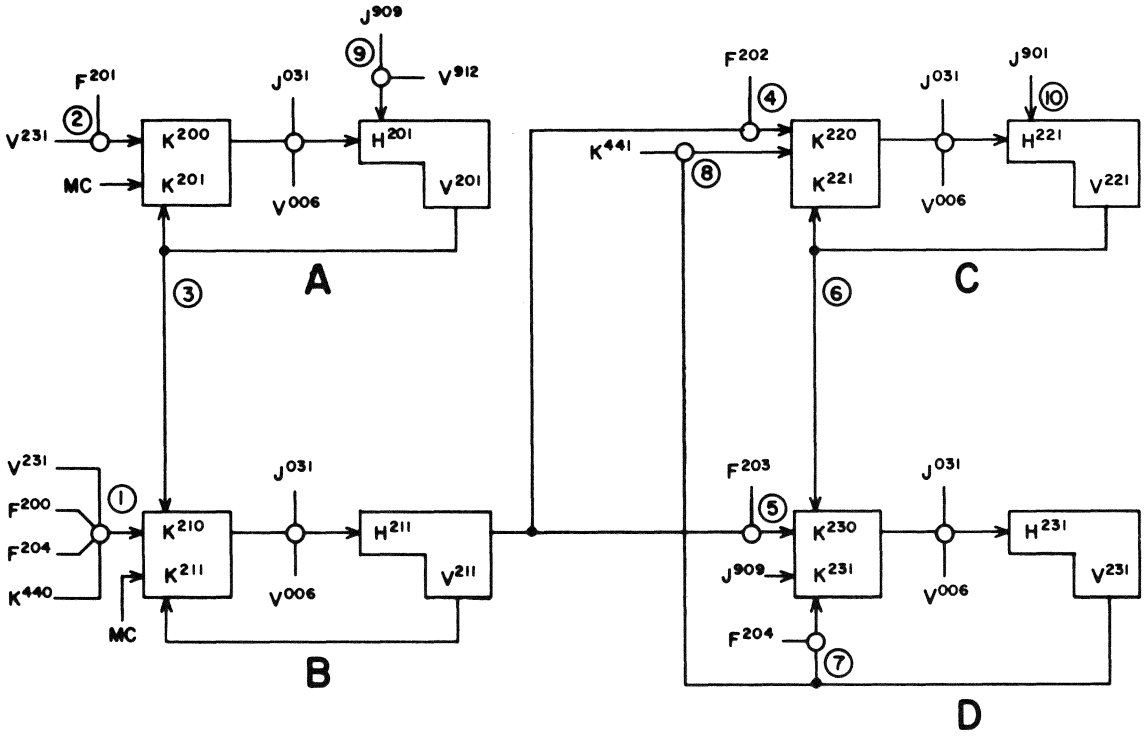


Figure 3-14. Storage Sequence Control

The Storage Sequence Control (figure 3-14) 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 $K^{210/211}$ (sequence B) controls the D→B progression.
The four AND terms are:
V^{231}, D sequence control delay
F^{200}, NOT indirect addressing mode
K^{440}, I/O Sequence Control FF indicating NOT instructions 72, 73, or 76
F^{204}, NOT D→D progression

Input (2) to $K^{200/201}$ (sequence A) controls the D→A progression.
F^{201}, direct address
V^{231}, D sequence control delay

Input (3) to K^{210} is an unconditional progression from A→B.

Input (4) to $K^{220/221}$ (sequence C) controls the B→C progression.
V^{211}, B sequence control delay
F^{202}, used on store instructions involving sequence C.

Input (5) to $K^{230/231}$ (sequence D) controls the B→D progression.
F^{203}, the inverse of F^{202}, means B→D, and is up for all instructions but those involving sequence C.

Input (6) to K^{230} is an unconditional progression from C→D.

Input (7) to K^{231} (F^{204} V^{231}) clears the D sequence FF except on instructions involving no address.

Input (8) to K^{220} controls the D→C progression. The I/O Sequence Control FF ($K^{440/441}$) is up immediately on a 76 instruction (input one word to A), or after the B sequence on 72 or 73 instructions. Thus, for 72 or 73 the storage sequence progression becomes DBC-DC-DC, etc.

Input (9) to H^{201} (sequence A) is used as a means of initiating the B sequence during enter or load modes. Notice that J^{909} also clears the D sequence FF, since no instruction is read during either of these modes.

Input (10) to H^{221} is a Master Clear which effects a transfer to $K^{230/231}$ for setting up the D sequence. This is done in order to assure that an instruction is read when the computer is put in Run or Step.

Note that sequence D is the only sequence not cleared by a master clear signal. The sequence C FF is cleared through the J^{901} input to H^{221}. Each H term has an AND input with two common terms, J^{031} and V^{006}. These terms indicate pulse time 6 during the last quarter of the associated storage cycle. At pulse time 7, then, the corresponding V^{2-1} card reads out to clear the present sequence FF, and to initiate the next storage reference cycle.

ADDRESS SELECTION

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 or diverted, to one of the eight associated drive lines by means of eight diverter cards. Thus, 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 register are divided into four octal groups. The highest-order group controls inputs to translators which select the horizontal read/write driver, the next lower group provides horizontal diverter selection, and the two remaining octal groups provide selection for the vertical read/write driver and vertical diverter.

HORIZONTAL R/W DRIVER			HORIZONTAL DIVERTER			VERTICAL R/W DRIVER			VERTICAL DIVERTER		
11	10	09	08	07	06	05	04	03	02	01	00

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 translator 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 is dependent 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, page 3-44.

The method of selecting one of the eight R/W drivers is shown in figure 3-15. 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 is solely dependent 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 the source of d-c from which the selected driver draws the read-write current. A current of 400 ma is needed to shift the state of the memory core. Each current source card provides 100 ma of this current. The half current of 200 ma needed for the H or V line is provided by a current step-up ratio of 1:2 delivered 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 3-15. Note that an even-numbered driver has its eight drive lines interlaced with those from the successively higher odd-numbered driver. Thus, G^{0-0} is interlaced with G^{0-1}, G^{0-2} with G^{0-3}, G^{0-4} with G^{0-5}, and G^{0-6} with G^{0-7}. This does not affect the selection of any chosen core, but merely the physical location of that core within the matrix. This interlacing allows connections to the drive lines to be made more conveniently.

The T^{1--} translators select the horizontal R/W drivers (G^{01-}), while the T^{0--} cards perform the vertical R/W driver selection (G^{00-}).

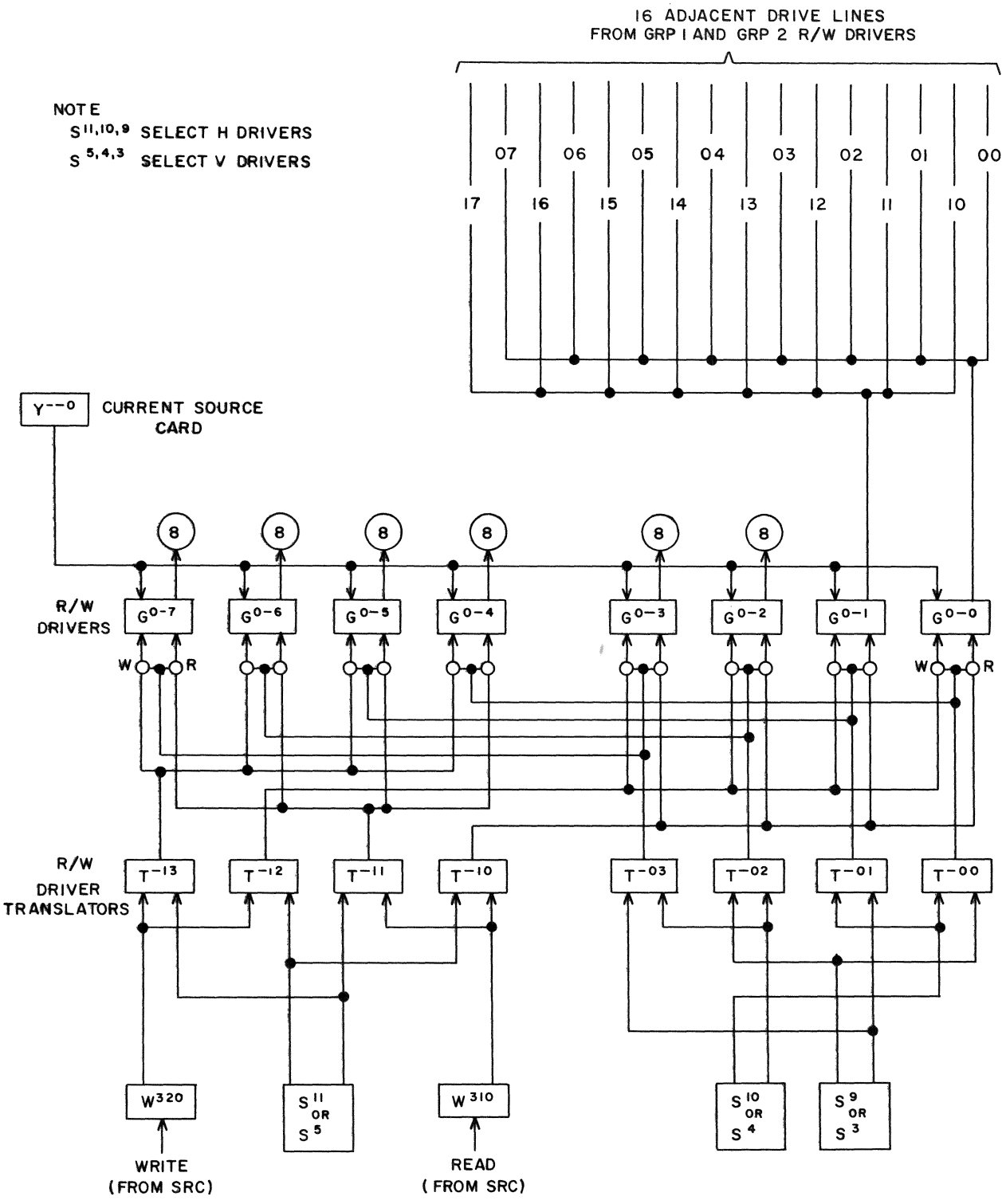


Figure 3-15. R/W Driver Selection

Diverter Selection

The diverters determine which of the eight drive lines is to carry the R/W current through the 12 bit-planes. Stages $S^{6, 7, 8}$ supply the information utilized by translators T^{300} through T^{305} to determine the horizontal drive line selected; $S^{2, 1, 0}$ are used in conjunction with T^{200} through T^{205} to select the vertical drive line. The eight horizontal diverters (D^{10-}) and the eight vertical diverters (D^{00-}) are selected by a straightforward translation of the three bits of the S register which govern that particular set. Figure 3-16 shows the diverter selection. A translation choosing diverter four (D^{-04}) is represented as an aid to understanding the octal translation.

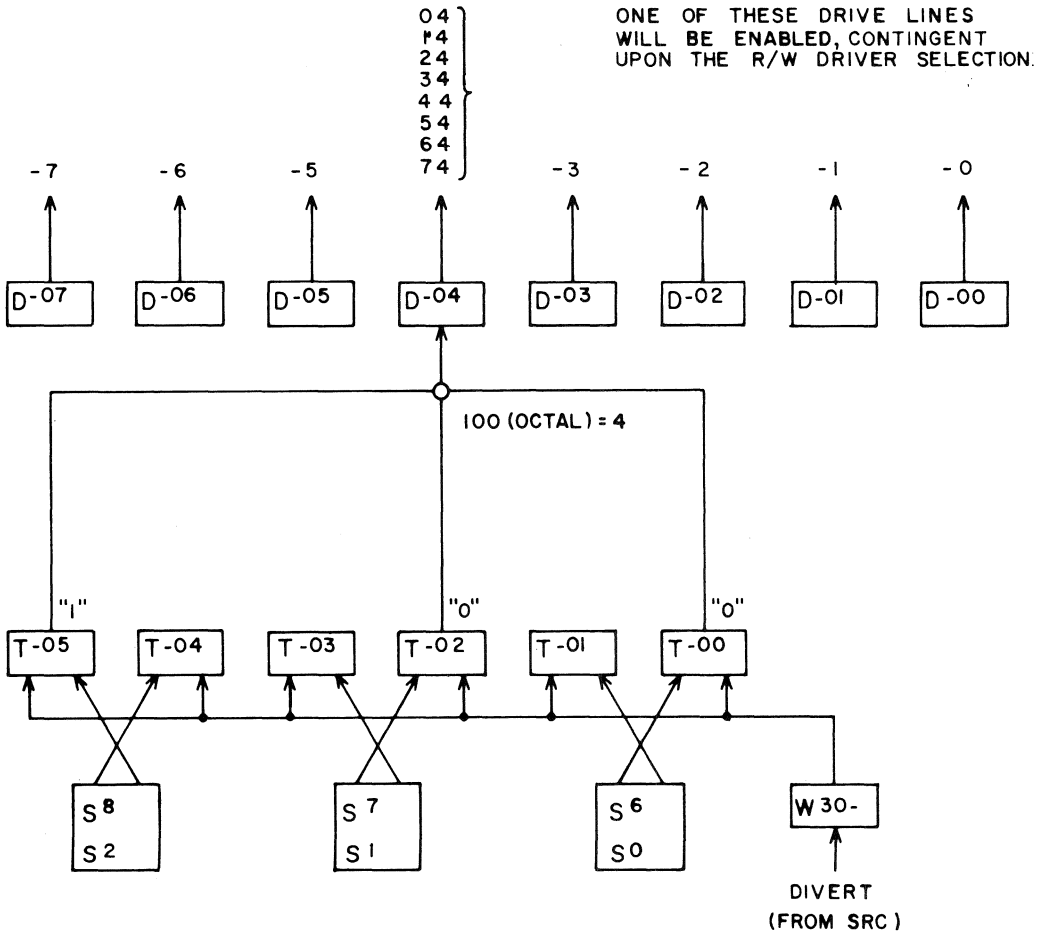


Figure 3-16. Typical Diverter Selection

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 3-17). The stage of Z controlling the inhibit translator must be a "1" if the translator is to be energized, and also, the inhibit phase from storage reference control must be in effect. A separate current source card, Y^{2--}, provides d-c current for the inhibit driver connected to each memory-plane.

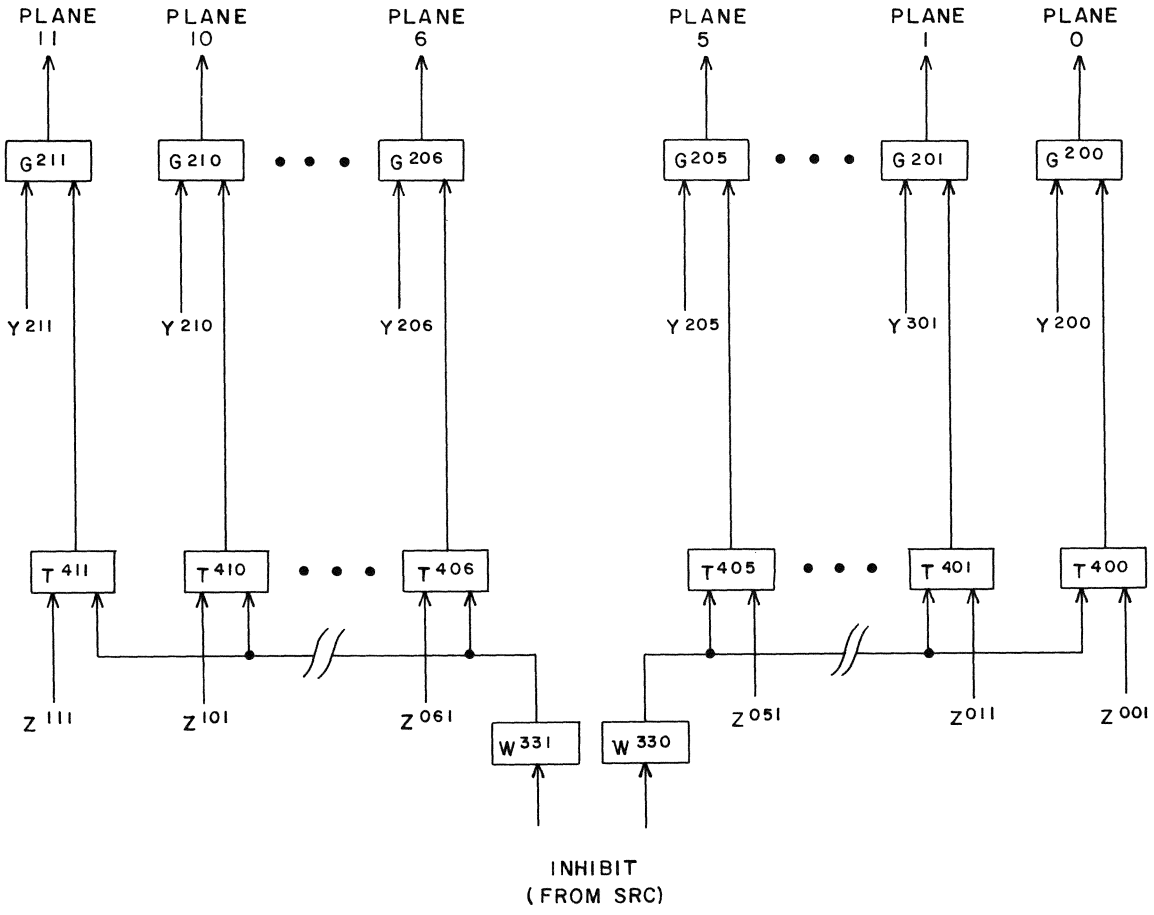


Figure 3-17. Inhibit Driver Selection

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 (card type 51)	current source (card type 54)
diverter (card type 52A)	inhibit generator (card type 55)
selector (card type 53)	sense amplifier (card type 56)

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.

Drive Generator (Type 51)

The drive generator (figure 3-18)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 an emitter-follower, and transistors Q02 and Q03, connected in parallel as amplifiers. The input signal is an AND combination of two selector outputs. A -1v input results in approximately 0v at the base of Q01. The emitter Q01 is clamped to ground by CR03, so neither Q02 or Q03 conduct. Consequently, no current flows in the primary of T01.

A -12v input signal causes Q01 to conduct; however, the conduction is held below saturation by feedback diode CR01. The negative voltage developed across R06 is applied to the bases of Q02 and Q03, causing these transistors to conduct. Current flows from the current sources through the emitters of Q02 and Q03 to the collectors, through the primary of T01, to the current sources. The current pulse from the secondary of T01, amplified by the step-down action 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 B input generates a write pulse.

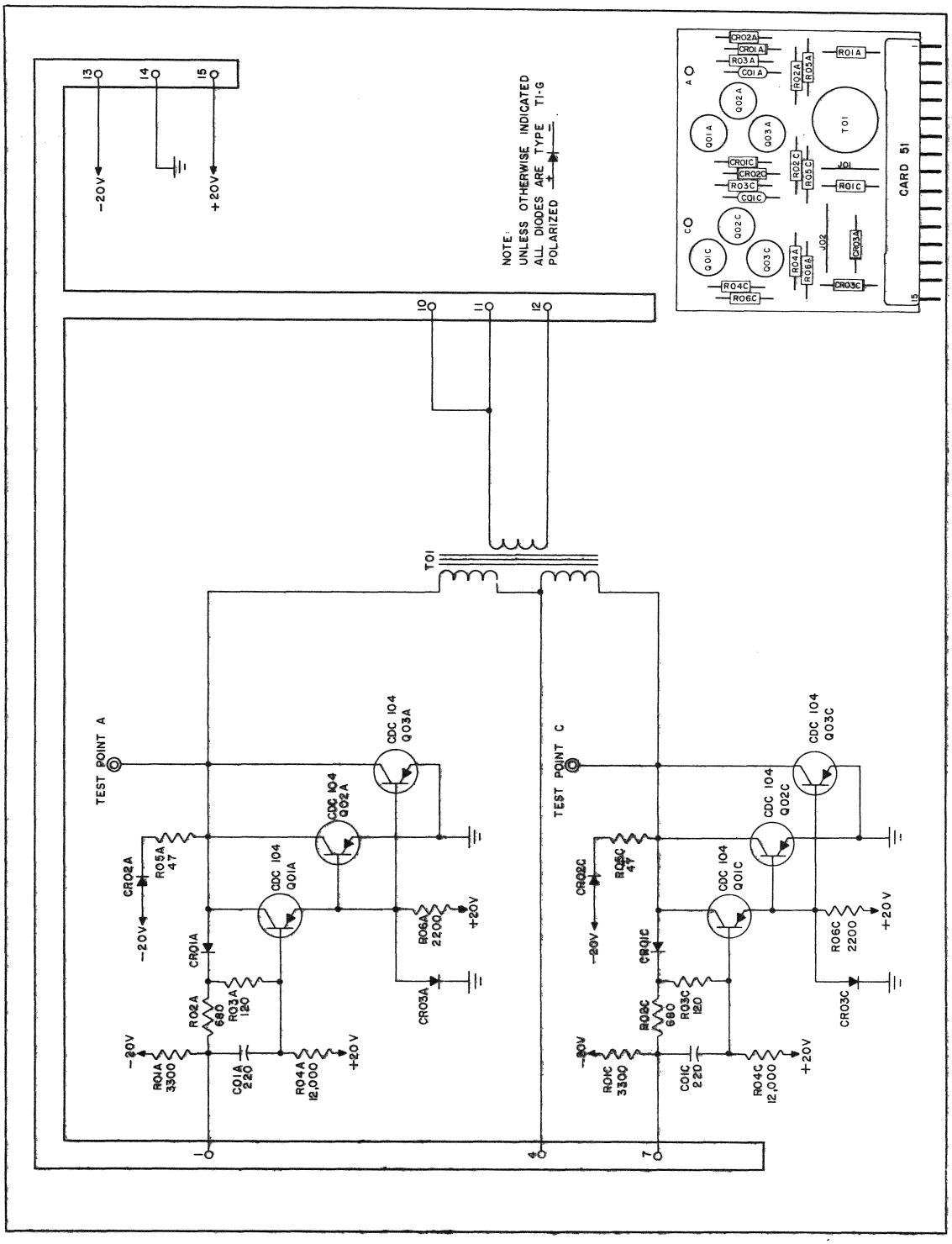


Figure 3-18. R/W Driver (51)

Diverter (Type 52A)

The diverter circuit (figure 3-19) serves as an electronic switch in series with an H or V wire of the memory plane assembly. The diverter consists of transistor Q01, connected as an emitter follower, and transistors Q02 and Q03. One or the other of these transistors 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 passes Q02. A negative pulse passes one of the pairs of diodes CR01 and CR02, CR05 and CR06, etc., and passes Q03. In either case, the current pulse is returned to the R/W driver. The bleeder networks of all the diverters are connected in parallel via terminals 11 and 12. This connection equalizes the current flow through the bleeders and thus reduces heating.

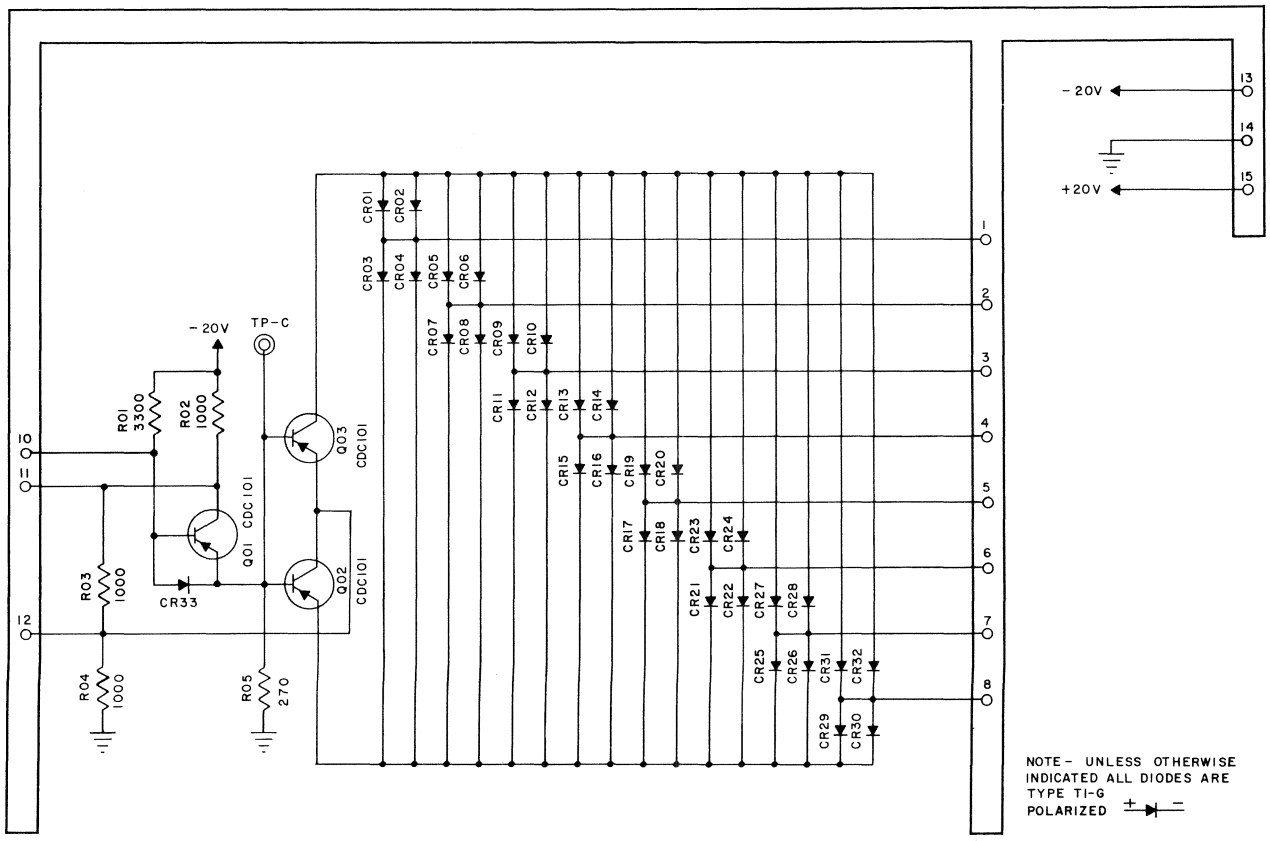


Figure 3-19. Diverter (52A)

Selector (Type 53)

The storage translators drive type 51, 52, and 55 cards, each of which presents a different impedance level input than the standard logic card. For this reason, all storage translators employ the type 51 card. Each selector card consists of two identical selector circuits. A selector circuit (figure 3-20) is similar to the standard inverter except that the resistance results in output signal levels of -1v and -12v. Each selector circuit has two input diodes (CR01 and CR02) and four output diodes (CR09, CR10, CR11 and CR12).

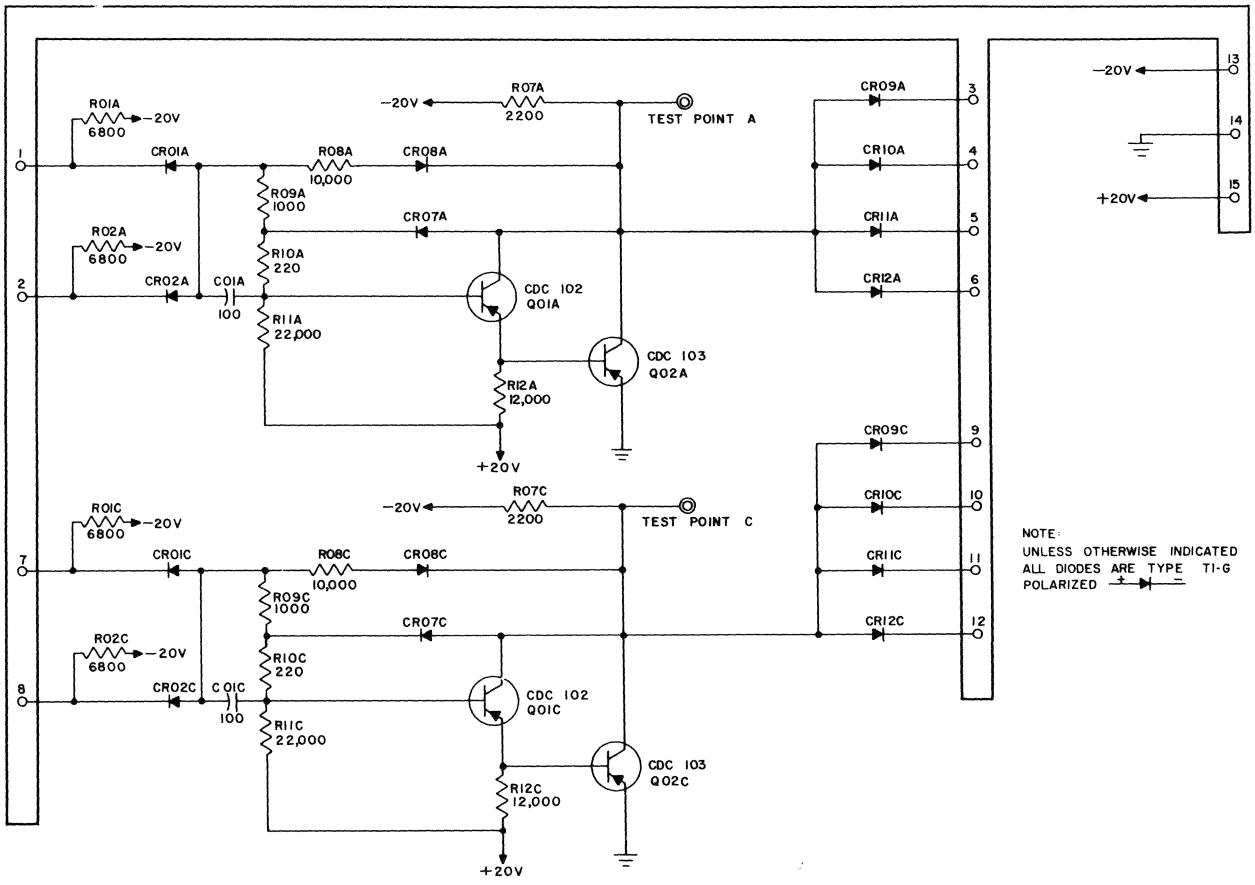


Figure 3-20. Selector (53)

Current Source (Type 54)

The current source card (figure 3-21) consists of five banks of parallel resistors. The effective resistance of four banks is 150 ohms each; that of the remaining bank is 303 ohms. One end of each bank is connected to the -20v output of the power supply. The 150-ohm banks supply current to the H, V and inhibit current generators; the 303-ohm banks usually supply current to a dummy load. *

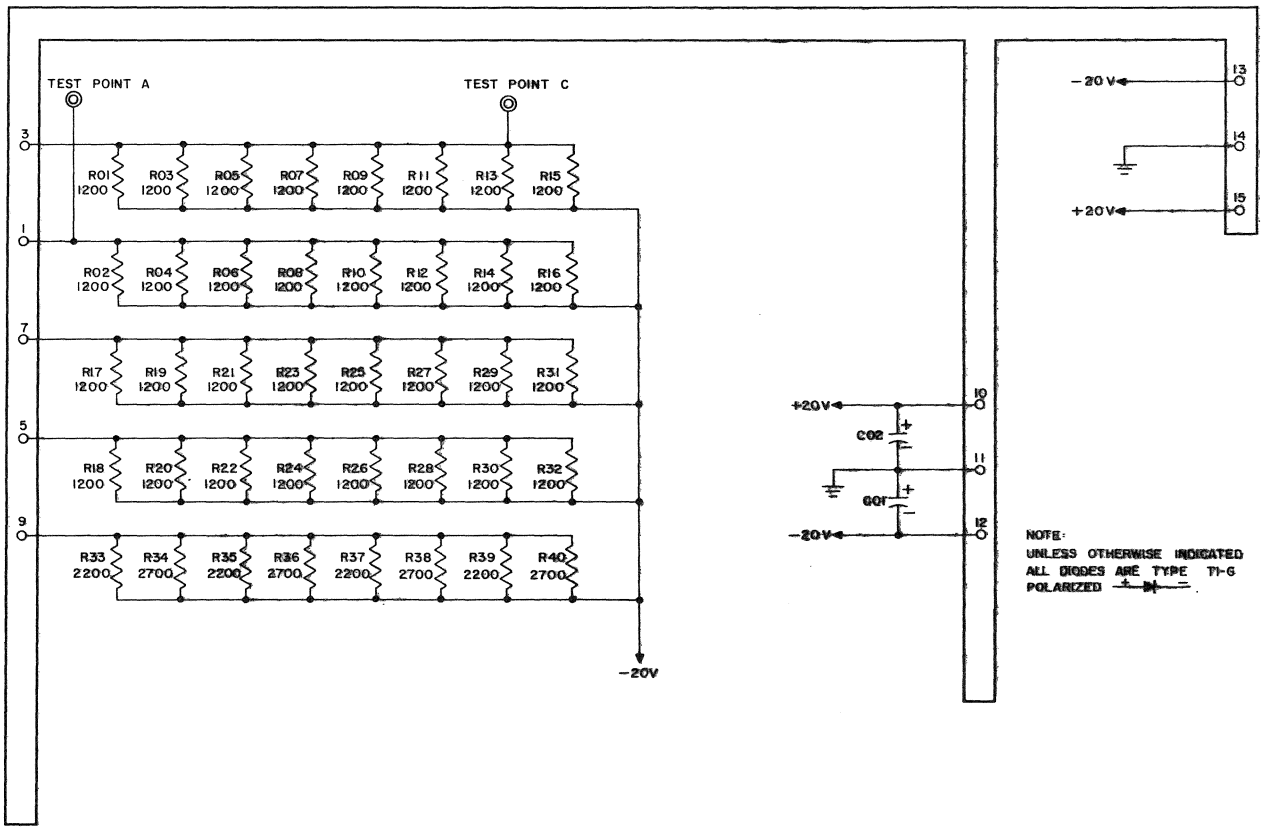


Figure 3-21. Current Source (54)

* Cards are standard types for use in various computers. The subject computer does not use a dummy load, therefore the 303-ohm banks are not connected.

Inhibit Generator (Type 58)

Each inhibit generator card (figure 3-22) has two generator circuits which are similar to the type 51 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 58 card causes Q01 to conduct and thus enable Q02 and Q03. Current from the external source connects to the generator via terminal 6 or 12 and passes through Q02 and Q03 to ground.

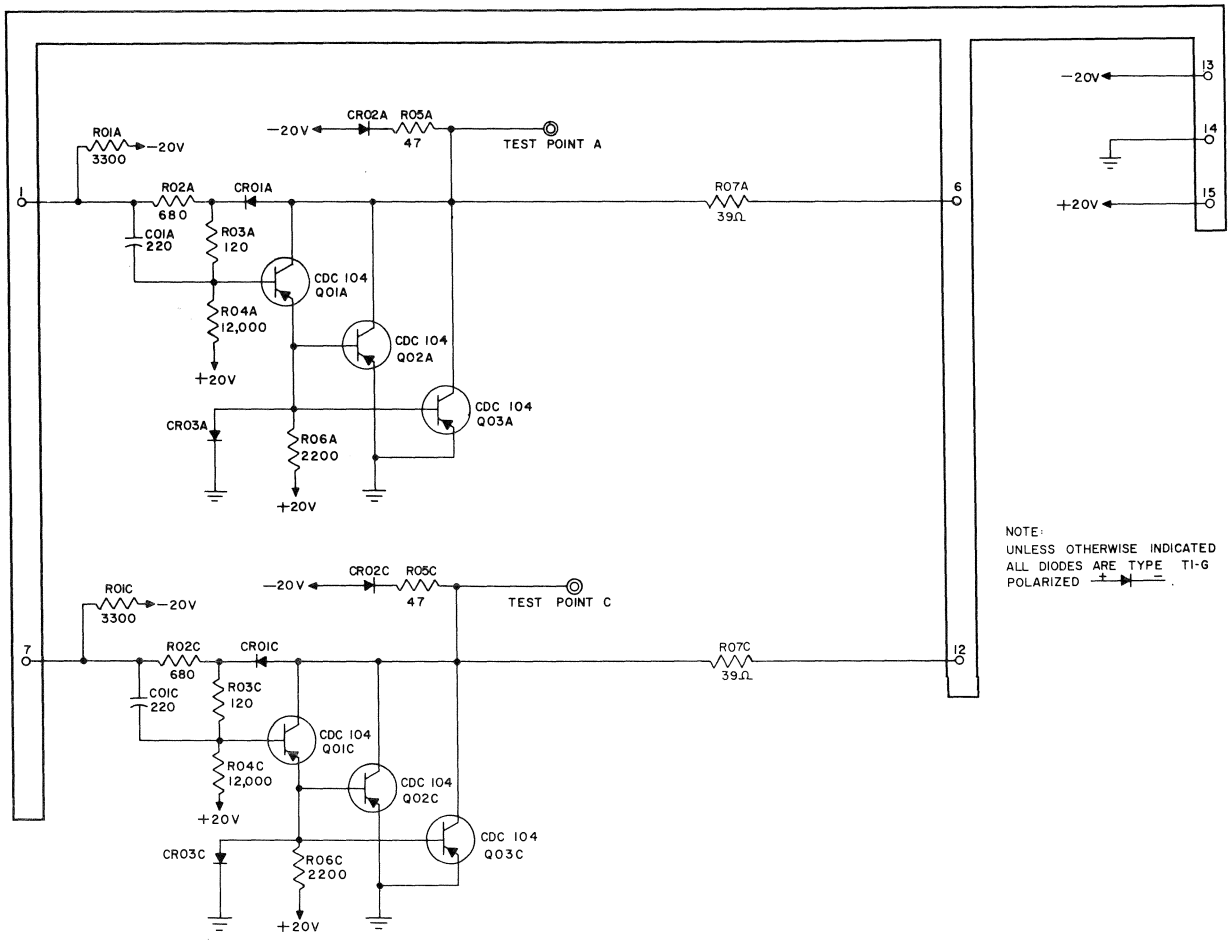


Figure 3-22. Inhibit Generator (58)

Sense Amplifier (Type 57)

The sense amplifier (figure 3-23) amplifies the signal from a memory plane (pins 2 and 4) as the result of a read pulse. Transistors Q01, Q02, Q03, and Q04 form a differential amplifier which feeds T1 through coupling capacitors C02 and C03. The secondary of T1 is connected to a bridge detector so that the signal polarity at Ⓐ is always the same. The Q05 network is a standard inverter circuit providing normal output logic levels of -3.0v and -0.5v.

Gain from the differential amplifier for the common mode component of the input signal is about 2; for the differential mode component, across R06 - C01, gain is about 100. Detector bias is determined by the Margin switch on the console. With the switch at HI, the +20v applied to pin 1 raises the reference level of the detector making it less sensitive; weak signals tend to be dropped. With the switch at LO, -20v appears at pin 1 and the detector is more sensitive to spurious pulses. No connection to pin 1 is made when the switch is in the normal, center position. Pin 6 is connected to the diverter bus (-7.0v) to provide bias for input transistors Q01, Q02. The output from Q05, as the result of a "1" signal on the sense lines, is -0.5v.

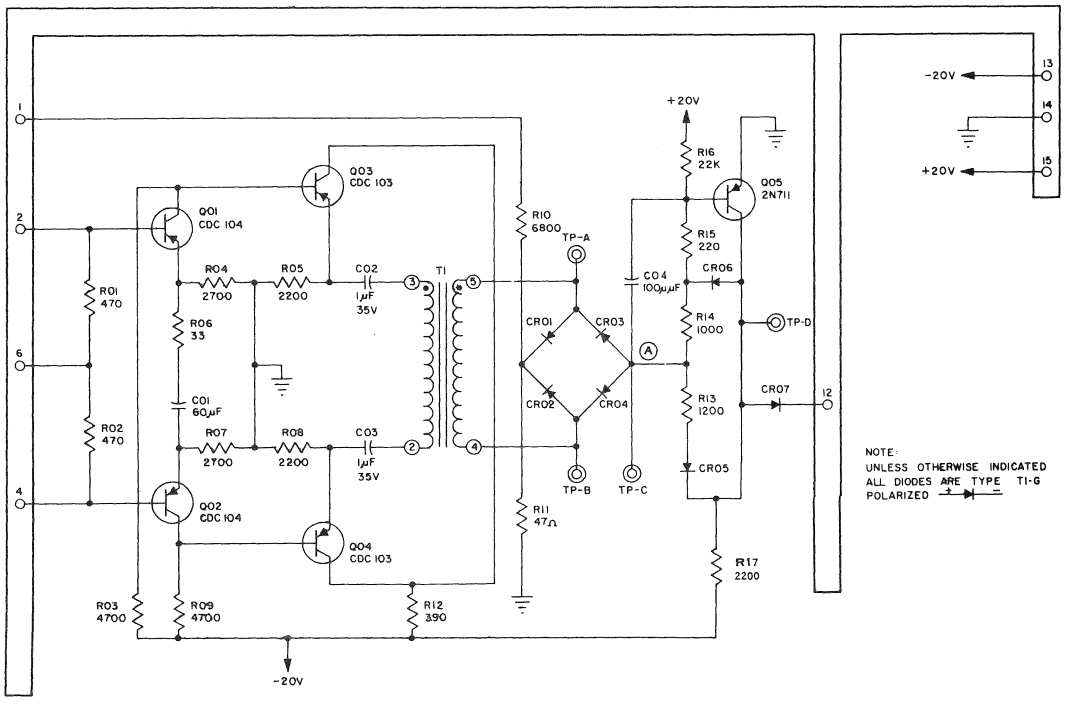


Figure 3-23. Sense Amplifier (57)

ARITHMETIC SECTION

BINARY ARITHMETIC

The binary number system is the basis for the representation and manipulation of all information within the computer. Only two digits, "0" and "1", are used in this system. This characteristic is fundamental to computer operation since it permits the assignment of a pair of conditions, in this case voltage levels, to be sufficient to encode data presented 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.

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

Binary numbers are added together according to the following rules:

0 + 0 = 0

0 + 1 = 1

1 + 0 = 1

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	
Sum	1011	(11)

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

8 (minuend)	or	8 (minuend)
-6 (subtrahend)		+4 (10's complement of subtrahend)
2 (difference)		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 =

8 + (10-6) - 10 = 2

8 + 4 - 10

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)

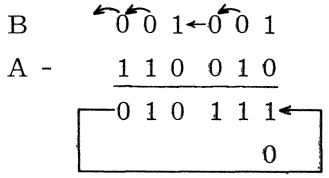
The one's complement of a binary number may also be formed by substituting "1's" for "0's" and "0's" for "1's" in the number.

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, then, 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 BORROW PYRAMID

The borrow pyramid 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 one-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 = 0 and the subtrahend = 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.

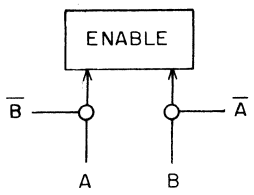
Two cases produce a remainder of 0. This means that, after the initial bit-by-bit subtraction, that stage will be unable to satisfy a borrow made upon it by some lower-order stage, and must pass the borrow on to the next higher-order position. These cases exist whenever 1 is subtracted from 1, or when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

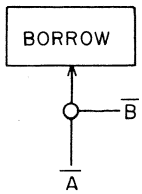
Stage-Generated Signals

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 shown diagrammatically 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.

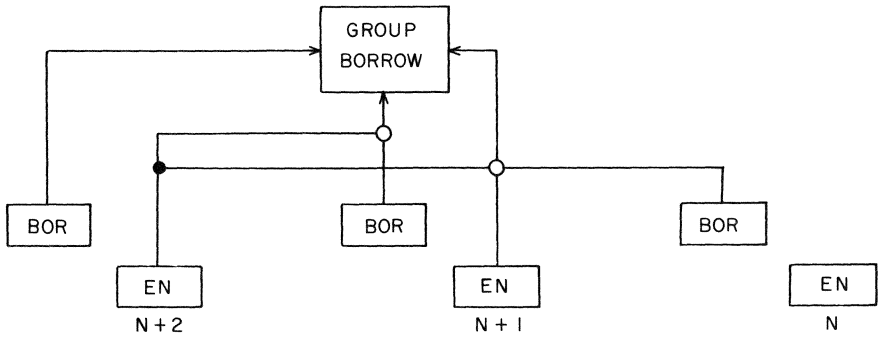




The small circle at the juncture of the symbol leaders and the arrow shaft indicates the logical AND function, which will be a "0" whenever either of the terms shown is a "0".

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 higher group. The conditions creating a group borrow are shown below.

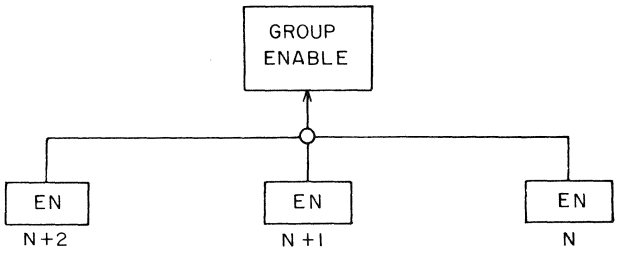


The leftmost input to the group borrow block is satisfied when the highest-order stage in that group (stage N+2) generates a borrow signal. The middle input is completed whenever the second (N+1) stage generates a borrow and the last stage has an enable signal present. The right-hand input shows that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

A group borrow is generated whenever a stage borrow is accompanied by 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 above apply regardless of the number of stages or groups involved, for a subtractive pyramid. The three-stage (octal) group is suggested because it preserves the identity of the octal group associated with the binary number system.

PYRAMID OPERATION

Three signals are used in conjunction with each pyramid operation to set the proper value for the final total into the B register FFs. The first of these signals, CLEAR, sets every stage of B to zero. 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 pyramid. 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 -- having been committed to initially clearing each stage of B -- is this: How does the final answer in the B register differ from all "0's" in that register? Stages which contain "1" are marked by an asterisk.

Example 1

E	B	B	NE	B	B	E	B	NE	NE	B	E		
1	0	0	1	0	0	0	0	1	1	0	0	2316	(4414_8)
0	0	1	1	0	0	1	0	1	1	0	1	813	(1455_8)
1	1	0	0	0	0	1	1	1	0	0	1	3129	(6071_8)
*	*					*	*	*			*		

Example 2

B	E	B	NE	B	E	E	E	E	E	E	NE		
0	0	0	1	0	0	1	0	1	0	1	1	299	(0453_8)
0	1	0	1	0	1	0	1	0	1	0	1	1365	(2525_8)
0	1	1	0	1	0	0	0	0	0	0	0	1664	(3200_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.

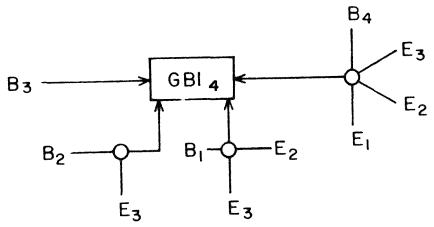
Rule 1 means that some provision must be made for complementing all stages which generate enable signals, since those stages were set to "1" by the toggle function, unless they are acted upon by a B or PB. Rule 2 means that all stages generating NE signals must be set to "1" (or complemented, since they 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.

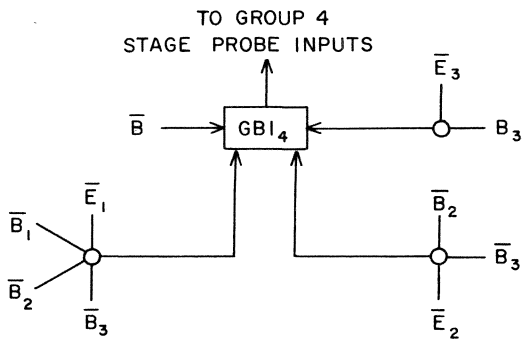
Group Borrow Input

A group borrow input card is associated with each of the four 3-stage octal groups comprising the pyramid. The philosophy of the group borrow was examined earlier, and is the same here, except that it is expanded to encompass four inputs instead of the three needed at the octal-group level, as shown below.



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 pyramid, 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 4 stage probe inputs will be "1" only when all GBI_4 inputs are down. Examination of the four situations which prompt a borrow from group 4 reveals that in every case, at least one of the terms provoking the borrow is present at each of the inputs to GBI_4:

Borrow from group 3, identified by $\overline{B}_3$, $\overline{B}$

Passed borrow from group 2, - $\overline{E}_3$, $\overline{B}_2$, $\overline{B}$

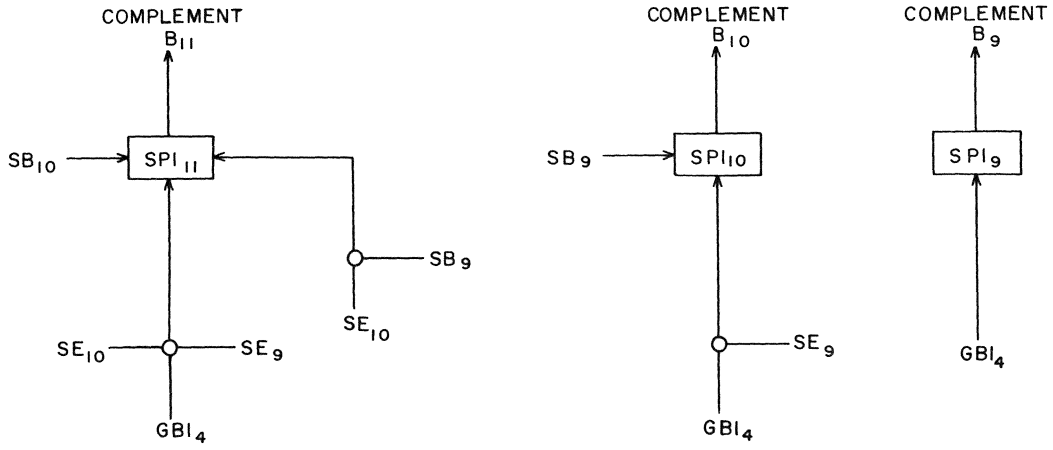
Passed borrow from group 1, - $\overline{E}_3$, $\overline{E}_2$, $\overline{B}_1$, $\overline{B}$

End-around borrow from group 4, - $\overline{E}_3$, $\overline{E}_2$, $\overline{E}_1$, $\overline{B}_4$, $\overline{B}$

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 B 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 fourth octal group (stages 9, 10, 11) of the B register.



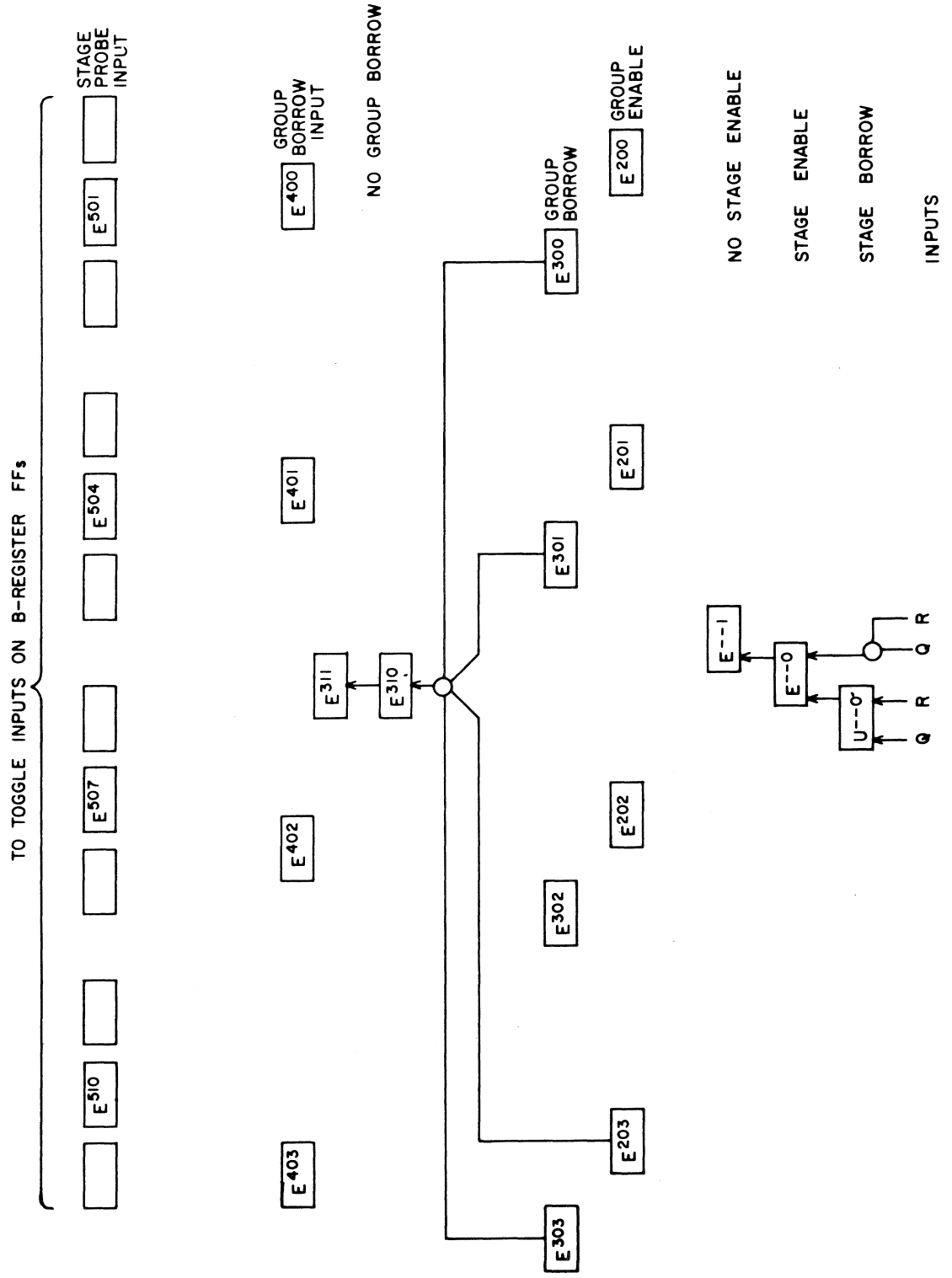


Figure 3-24. Borrow Pyramid Structure, Simplified

The center input in each case is the only one controlled by the group borrow input. Other inputs (if any) involve intra-group borrow signals. Thus, SPI_{11} will be forced to "0" by an intra-group borrow from stage 10 (SB_{10}), or by SB_9 with an accompanying enable from stage 10 (SE_{10}). Finally, SPI_{11} will be forced to "0" by a group borrow input from the fourth group, when accompanied by intervening enables from stages 9 and 10.

Final Total

Example 3 shows how the actual state of every stage of the B register is changed for each of the three signals CLEAR, TOGGLE, and PROBE. The quantities from Example 2 are used for consistency. Asterisks mark the stages which are changed by the PROBE.

Example 3

B	E	B	NE	B	E	E	E	E	E	E	NE	
0	0	0	1	0	0	1	0	1	0	1	1	
0	1	0	1	0	1	0	1	0	1	0	1	
0	0	0	0	0	0	0	0	0	0	0	0	CLEAR
0	1	0	0	0	1	1	1	1	1	1	0	TOGGLE
0	1	1	0	1	0	0	0	0	0	0	0	PROBE
		*		*	*	*	*	*	*	*		

The over-all structure of the borrow pyramid is seen in figure 3-24; figure 3-25 shows Example 2 in terms of the logic elements affected ("1" outputs) in the pyramid.

PYRAMID CONTROL

The operands upon which arithmetic functions are to be performed are introduced into the base of the pyramid through two sets of inverters carrying the designations R^{---} and Q^{---}. During each pyramid operation no more than one input to each set is enabled, and that input is selected according to whether the ensuing problem involves an arithmetic operation, transfer, or shift. These are the basic functions of which the pyramid is capable.

Pyramid Inputs

One R and one Q inverter is associated with each of the 12 pyramid stages. Each inverter has four inputs. These eight inputs may be combined in various ways to produce 13 pyramid-control commands, as shown in table 3-2.

TABLE 3-2. PYRAMID COMMANDS

TYPE	COMMAND	R	Q
Arithmetic	P + 1 → B	+1	P
	P + Z → B	$+Z_L$	P
	A + 1 → B	+1	A
	A + Z → B	+Z	A
	P - Z → B	$-Z_L$	P
	A - Z → B	-Z	A
	Fixed Multiply (A x 10 decimal)	$A \cdot 2^1$	$A \cdot 2^3$
Transfer	A → B		A
	-Z → B	-Z	
	+Z → B	+Z	
Shift	A left 1	$A \cdot 2^1$	
	A left 3 (octal shift)		$A \cdot 2^3$
	Z left 1 (shift replace)	Z	Z

Input Control

Figure 3-26 describes the four inputs to the R and Q inverters. First-level control for each of the functions resides in two W^{---} elements, one controls the lower six stages of the pyramid, the other the upper six stages. The two control elements may

act together to bring the entire contents of a register into the pyramid, or they may act individually to enter only the upper or lower half of the register. This decision is a function of the instruction word operation code (F) as expressed by the function translator.

Note in figure 3-26 that the lowest stage of R does not contain a +1 control but that the -Z input is duplicated in order to fill up the R inputs. This omission is explained by the fact that the +1→R function really deters the pyramid inputs to all but the first stage.

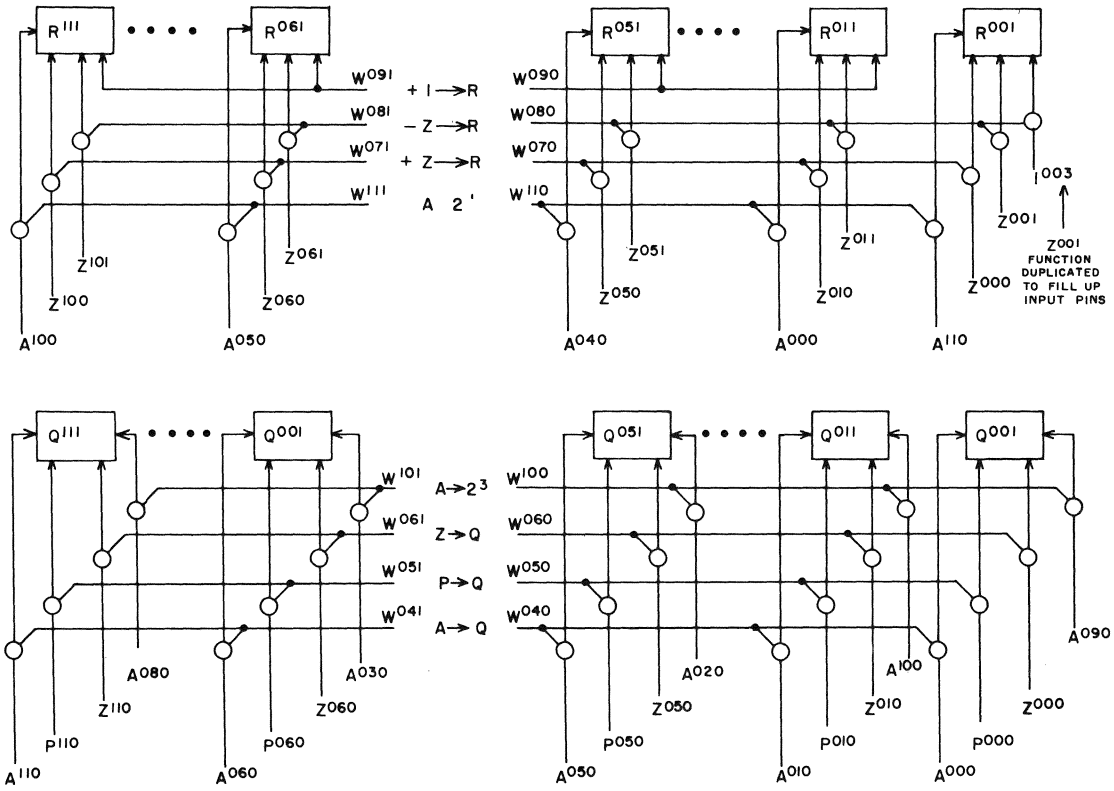


Figure 3-26. Pyramid Inputs (R & Q Inverters)

INPUT-OUTPUT SECTION

INTRODUCTION

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

Two cables, one input and one output, link the external equipment to the computer. All information from the externally located peripheral equipments must enter or leave the computer through one of these cables. The paper tape reader and punch, however, being integral to the console cabinet, do not connect to the I/O cables.

Information passes between the computer and the external equipment as a block of information at a word-by-word rate. The speed of the particular equipment in communication with the computer 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 two cables appears in table 3-3. Since the cables are identical in construction they are interchangeable with respect to physical connection to the computer 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 computer. Unique select codes sent by the computer determine which equipment attached to a cable shall communicate with the computer.

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
7	J	7
8	K	8
9	L	9
10	M	10
11	N	11
	P	
Input Ready	R	Information Ready
Input Request	S	Output Resume
Input Active	T	Function Ready
	U	Master Clear
Input Disconnect	V	Output Active
	W	
	X	
	Y	
	Z	
	a	
Ground	b	Ground

TABLE 3-4. INPUT AND OUTPUT CABLE LINES

Input Cable Lines	
Input Data and Input Status (12 lines)	Dual Purpose: 1) As data lines they hold equipment input register contents which the computer 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 computer may sample. (Computer resync circuits are oriented about the leading edge of the ready signal.)
Input Request (1 line)	Indicates to external equipment that computer has accepted input word. This line is turned off by input ready.
Input Disconnect (1 line)	Indicates to computer that input device has no more data to deliver. Computer 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 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 (EXF) codes to select or sense a condition within the equipment. Function ready alerts the equipment to sample EXF code.
Function Ready (1 line)	Accompanies EXF code; turned on by instruction 75 and causes the equipment to examine EXF 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 computer and is turned on when the computer 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 EXF code. (The computer resync circuitry orients itself about the trailing edge of the resume; when the signal drops the word is exchanged.) The computer prepares a word while the signal is up.
Master Clear (1 line)	This signal clears all external equipment. It occurs when the Load-Clear switch at the console is in the Clear (down) position.
Input Active (1 line)	Permits high-speed transfer (78KC max) of input information when external device is suitably equipped.
Output Active (1 line)	Permits high-speed transfer of output information when external device is suitably equipped.

EQUIPMENT SELECTION

The computer controls the operation of its external equipment by issuing 12-bit function codes. See code listing in chapter 1. This process, initiated by the EXF (75) instruction, selects a 12-bit code located E positions forward of the current instruction address 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 a function resume signal to the computer.

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 computer 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 12-bit status response signal on the input cable. The computer processes this response through an input instruction (73 or 76) followed by program control.

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

0004	000	000	000	100
0020	000	000	010	000
0040	000	000	100	000
	000	000	110	100
	0	0	6	4

the resultant code, 0064, must be analyzed by the programmer to determine which codes responded and what conditions exist in the unit.

---

* Unit designators in machines built before June 1961 operate in a 2-out-of-6 coding scheme (15 selections possible), and will not accommodate the full 6-bit unit selecti unless modified.

INPUT-OUTPUT CONTROL

The input-output control system (figure 3-29) 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 computer to be synchronized with those circuits. This is possible because the incoming signal is of long duration as compared to the logic pulses. Specifically, the purpose of resynchronizing is two-fold: to convert a signal which is asynchronous with respect to the computer to one that is timed by the computer clock and to resolve runt pulses.

Resynchronizing Network - The resync network (figure 3-27) consists of a two-stage, double-rank counter and four cascaded control delays. Timing of the mechanism is derived from two inverters driven directly from the master clock. The inverters present a square clock pulse to the FF inputs.

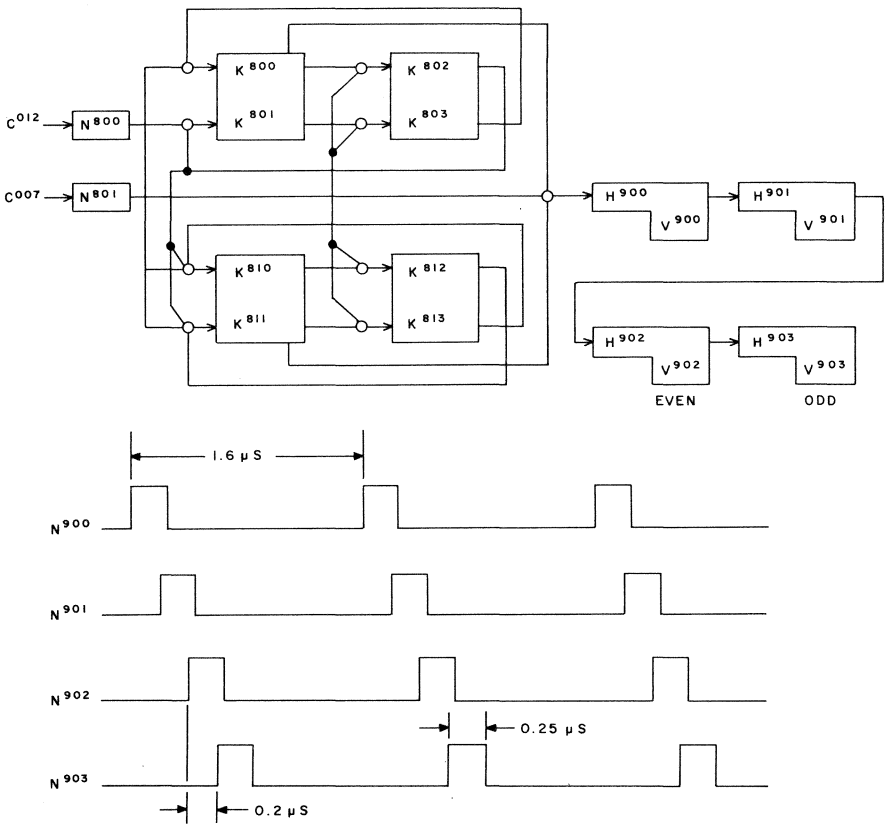


Figure 3-27. Resynchronizing Network

Concurrent with every fourth odd clock pulse, H900 is energized and V900 produces an output 0.2 usec later, followed at relative times 0.4 and 0.6 by V901 and V902 (figure 3-27). There is a lapse of 1.6 usec between successive outputs from any one control delay in the chain. The control delays are labeled "odd" and "even", depending upon the clock phase during which they produce an output; the third superscript indicates the phase time of the output.

Resynchronizing Circuit - In the typical resync circuit in figure 3-28, an input signal which normally registers -16v, must be timed with the computer logic upon going to ground (0v). A ground voltage on an input cable is a "1", a -16v signal a "0". When no external input signal is present (circuit input is -16v), and during every even sync pulse, the Sample FF is cleared while the Enable FF is repeatedly set by the following odd sync pulse. For the duration of this input condition the AND input to the control delay is not met and no pulse is generated by the circuit.

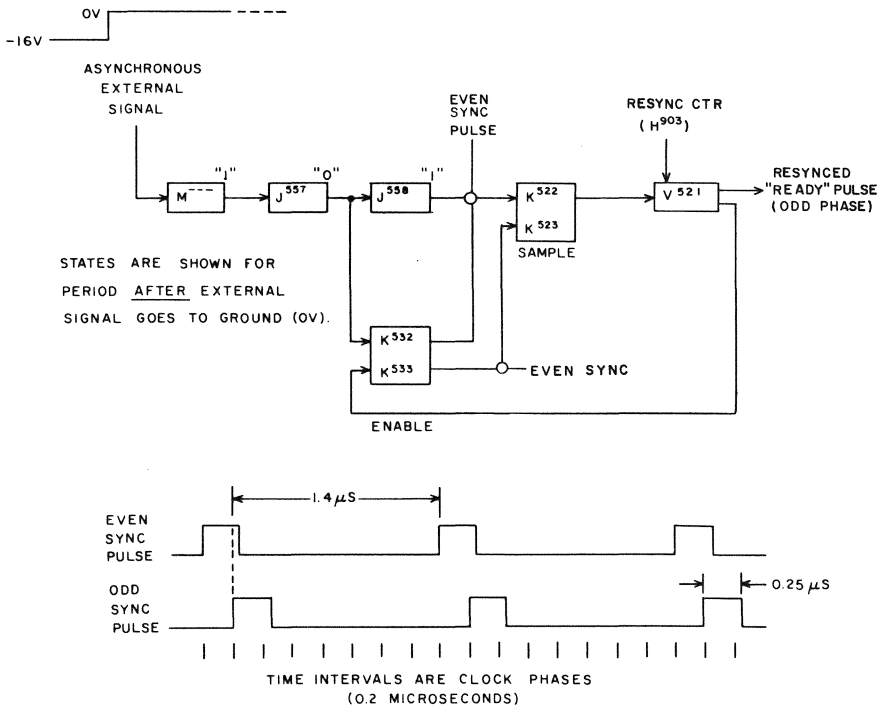


Figure 3-28. Typical Resync Circuit

When the input to the circuit changes to 0v, the Sample FF is set by the first even sync pulse. A synced output pulse which occurs during the next clock phase is sent to the other logic circuits and is also used to clear the Enable FF, disabling the SET input to the Sample FF before the next even sync pulse is generated. The result is a single synchronized pulse.

The Enable FF remains cleared until the input to the circuit returns to -16v at which time the Sample FF is cleared and the Enable FF is set. This is the normal static state of the circuit. Actually, only changes of input from -16v to 0v are sensed by the circuit. Once the circuit input has gone to 0v, it must return to -16v before anything further can be sensed by the circuit.

Runt Pulses - A runt pulse is an input to the circuit that (at approximately 0.3 usec before an even sync pulse) has a value between -16v and 0v. Such a pulse represents neither a "1" nor a "0" but rather, the voltage level taken from the input amplifier is somewhere between -3.0v and -0.5v.

For example, if the voltage level taken from the input amplifier is -1.5v, the next even sync pulse gates half-size pulses to both sides of the FF and its behavior becomes indeterminate. The FF may settle in either the clear or set state. Since this requires a much longer period than normal FF action, the clear or set condition will not be fully resolved within 0.2 usec. The maximum resolution period approaches 1.4 usec. This is the same period of time as that which elapses between the leading edges of successive even and odd sync pulses (figure 3-28). Only when the runt input is fully resolved is the control delay pulsed. The purpose of the odd and even sync pulse pair is to provide for the resolution of runt pulses within the Sample FF. The even sync pulse sets the time for sampling the asyncronous signal; the odd pulse sets the time for sampling the Sample FF.

Time-Sharing of Resync Circuit - Asynchronous signals from outside the computer logic time-share one common resync circuit as shown in figure 3-29, left. There are 6 asynchronous signals to which the computer must respond:

Input Disconnect
Input Ready-other than photoelectric reader (PER)
Output Ready - other than high-speed punch (HSP)
Function Resume (except from PER or HSP)

Input Ready (from PER)

Output Resume (from HSP)

Whenever either console-mounted input-output equipment (HSP or PER) is selected, a function resume is artificially generated from the Function Ready FF, M915 (figure 3-29, right); it is not asynchronous. The output resume line from externally-mounted devices also carries the function resume signal from those units, as shown in table 3-4.

The first four of the signals listed above are represented by the top three M-cards at the left of figure 3-29. A "1" from these cards will reach J557 only if neither HSP or PER is in use. The PER feed hole generates the Ready signal for that device. HSP Resume is discussed in connection with the punch. Since only one input-output equipment can be in use at any one time, no two asynchronous signals will ever appear simultaneously as an input to J557. The RECIRC input to that card is a lockout device which inhibits the setting of the Enable FF, thereby preventing recognition of any asynchronous signal as long as the main timing chain is operative.

Two approaches to the time-sharing network must be considered. If some device other than the PER or HSP is selected (figure 3-28), initially, the incoming lines are "0", J557 is a "1", and the Enable FF is set. As soon as any of the externally controlled signals appears, J558 goes to a "1", and the Sample FF is set on the next even sync pulse. During the next odd sync time a ready pulse is issued via V521.

For PER Input Ready (figure 3-29), $\overline{FH}$ is set by the first space between adjacent feed holes. Setting of the Enable FF parallels the action described above for Sample, and takes place when the first feed hole appears. Since, for HSP and PER, J557 and J558 will be in the states shown, the next even sync following the setting of Enable will set the Sample FF and initiate a ready to the computer. At the same time, if K405 is clear, V531 will start the main timing chain. The $\overline{FH}$ FF is cleared by the setting of Sample.

The AND input to J558 prevents the Sample FF being accidentially set by extraneous pulses which might turn on the Enable FF. Such a case could result if the operator were to insert a tape in the reader while the program was running.

The $\overline{FH}$ FF remembers that the reader has seen the blank space between two feed holes. This is important in the event that two or more spaced operations involving PER are programmed. As soon as the last frame to be read appears over the reading station and

the data has been sampled, the reader brake is applied. The tape does not stop instantaneously, however; it coasts on until an inter-frame space (no feed-hole) appears over the reading station, at which time $\overline{FH}$ is set. If the tape continues to move until the next frame is over the photocell block, $\overline{FH}$ will remember that this frame has not yet been read, and the next select PER operation will sample that frame before the tape is moved.

I/O Sequence Control

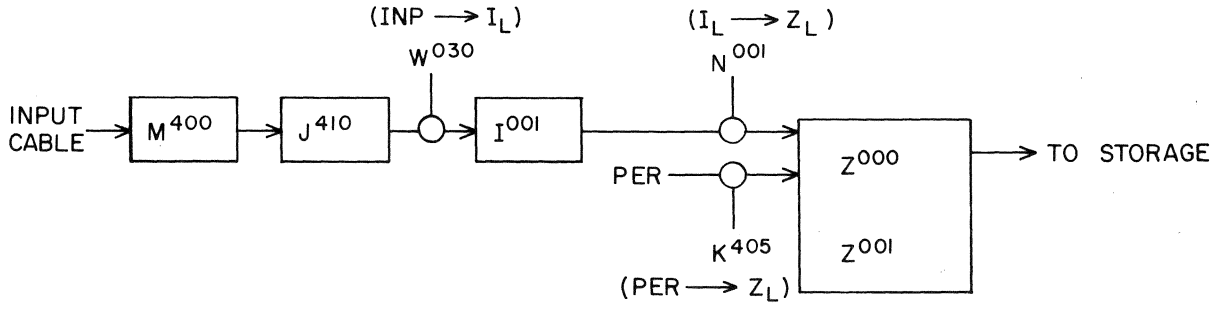
The I/O sequences have a general storage reference D, B, C, D, C, D, C, etc.; D reads the instruction, B obtains the starting address, and C stores (or reads) the data word. The succession of C-D cycles obtains the second and succeeding data words. The I/O Sequence Control FF which is set at the end of cycle B 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 C-D sequence until A = 0 (indicating that the terminating address has been reached) 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 73. 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 computer, indicating that the external equipment has processed the word.

Setting the Function Ready FF by instruction 75 produces a synthetic resume signal whenever the computer communicates with the PER or HSP as these two equipments are not connected to the input-output cables.

INPUT DATA TRANSMISSION

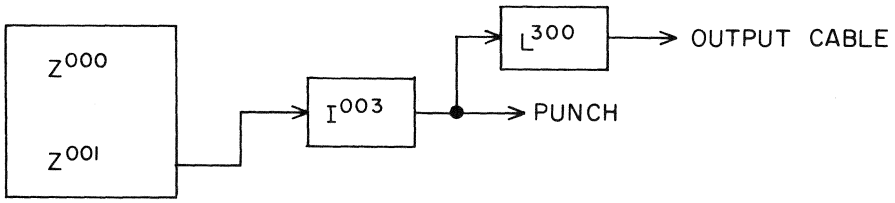
With one exception, all data from external input devices reaches the Z register through an inverter rank designated I^{--1}. Exception: the photoelectric reader data lines go directly to Z. The path of input data to the least significant stage of Z is shown below. Each of the 12 stages of Z has a similar logic arrangement between it and the input cable and the photoelectric reader.



Type 61 cards contain the three identical input amplifier circuits. Each is assigned an individual M^{---} symbol. These amplifiers form the input boundary elements between the input cable and the computer; and they convert communication line signals to computer signal levels. Each amplifier has a single input and two outputs.

OUTPUT DATA TRANSMISSION

Output data is transmitted from storage through the Z register to the punch or 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 seven lowest stages of Z. The upper five stages provide output only to the output cable. 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.



Type 62 cards containing three identical circuits are used as output amplifiers. Each circuit, assigned an individual L000 symbol, has a single input and a single output. The output amplifiers form the boundary elements between the computer and the cable groups; and convert computer signal levels to communication line levels.

HIGH-SPEED INPUT/OUTPUT (I/O Active Feature)

A complete input or output cycle requires 12.8 usec to store or deliver the word (SR cycle C) and compare the value in A with the terminating address to determine whether a new word is needed (SR cycle D). This time is normally much less than that required for the peripheral device to process a word. When the word handling rate of the external equipment approaches the 12.8 usec rate (78KC), the most efficient operation is realized if preparation and delivery of two successive words can be accomplished concurrently, if, for example, the computer can store word one while the external device is preparing word two for delivery. This is made possible by the I/O Active feature.

Figure 3-30 outlines the features of the high-speed I/O as compared to the standard I/O timing. In each case, a 78KC rate has been assumed for external equipment. Shaded areas compare elapsed times between successive storage or delivery operations for the two methods. Delays between corresponding signals at the computer and external devices are the result of the slow rise times of the input and output amplifier cards. Logic for the circuits is shown in figure 3-29.

Special circuits in the external device are needed to terminate operations if the computer fails to recognize a Ready-Resume signal. These circuits recognize the dropping of Input Request (for input) or Information Ready (for output) as the result of a normal completion of an I/O cycle.

CONSOLE CABINET I/O EQUIPMENT

The main computer cabinet houses two input-output devices; the photoelectric reader (input) is at the right of the operator's console and the high-speed punch (output) at the left.

PHOTOELECTRIC READER

The reader has two modes of operation. During load mode two successive frames of 6-bit data are loaded into one 12-bit storage address. During normal input operation the PER sends 7-bit data to the computer.

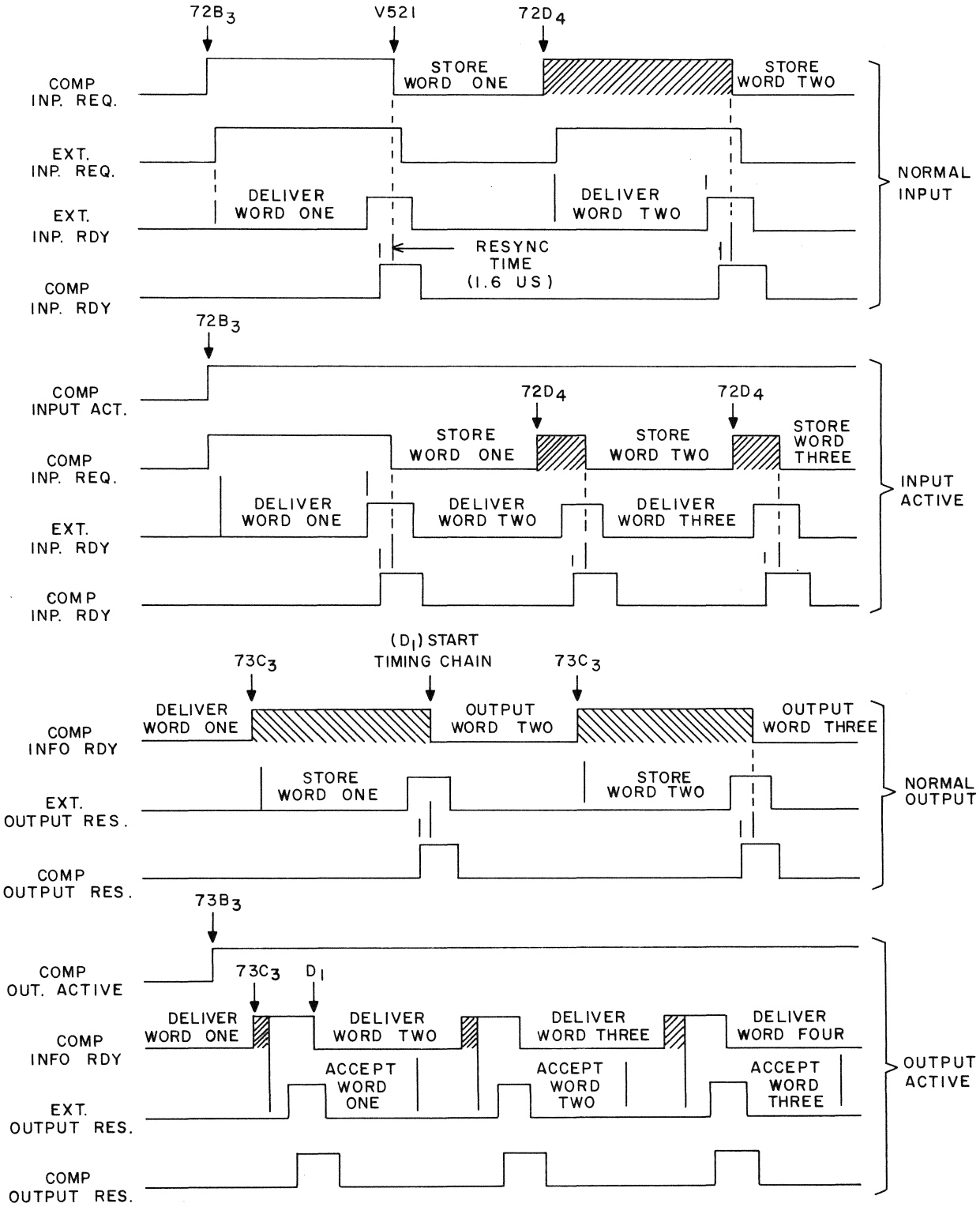


Figure 3-30. I/O Timing

Load Mode

The purpose of the load mode is to enter program data into successive storage locations within the computer. Data on the paper tape is contained in tape levels 0 through 5 (figure 3-31). A seventh level control hole appears in alternate frames for the purpose of controlling data loading into the Z register. Frames having a control hole are loaded into the upper six stages of Z; those without a control hole are loaded into the lower six stages. Two conditions are necessary:

1. The Load-Clear switch must be set to LOAD; this sets storage sequence control to the B-cycle.
2. The Run-Step switch must be set to RUN.

When these two conditions have been met, the next odd sync activates PER Input. This sets the Load Mode Control (LMC) and PER Clutch FFs, (the latter engages the reader clutch) after which tape begins to move through the reader.

The seventh level control hole of the first frame sets the Sense Seventh Level FF which causes the information to be entered in the upper six stages of Z. The trailing edge of the feed hole of this frame (V521) clears the LMC and the Sense Seventh Level FFs. Since the second frame does not contain a control hole, information held by it is entered in the lower six stages of Z. The feed hole of the second frame initiates a ready to the computer (V531) which allows the word of data to be entered in storage. When V531 clears the Clutch Control FF, a request is returned to the reader.

After at least one frame (not tape leader) has been read, the tape stops upon the appearance of two consecutive frames neither of which contains a control hole. The trailing edge of the feed hole of the first frame without a control hole operates in the usual manner. Since the load mode is still in effect the LMC FF is reset. During the third quarter of the storage reference cycle (B3) the Load Mode Stop FF is set. If no control hole appears in the next frame the Sense Seventh Level FF remains cleared and the ensuing feed hole pulse from the second frame completes the AND to stop the tape by clearing the Clutch FF and stop the computer by clearing the Run-Step FF. The process is completed by clearing the Load Mode Stop FF.

A master clear from the computer is necessary after each load mode sequence to prepare the circuit for succeeding load mode operations. This pulse clears the Load Mode Control and Load Mode Stop FFs. The same pulse sets the D-cycle FF in storage sequence control so that the computer will read the first instruction when the Run-Step key is set.

PER Select

During a normal input operation (72 or 76) involving the photoelectric reader, the Select PER FF is set by EXF code 4XX2 (figure 3-32). This FF disables the input ready gate from other external equipment and enables the PER gate for that signal. Data from successive 6 or 7-bit frames of tape are entered in the lower bits of sequential storage locations (upper bits are zeros).

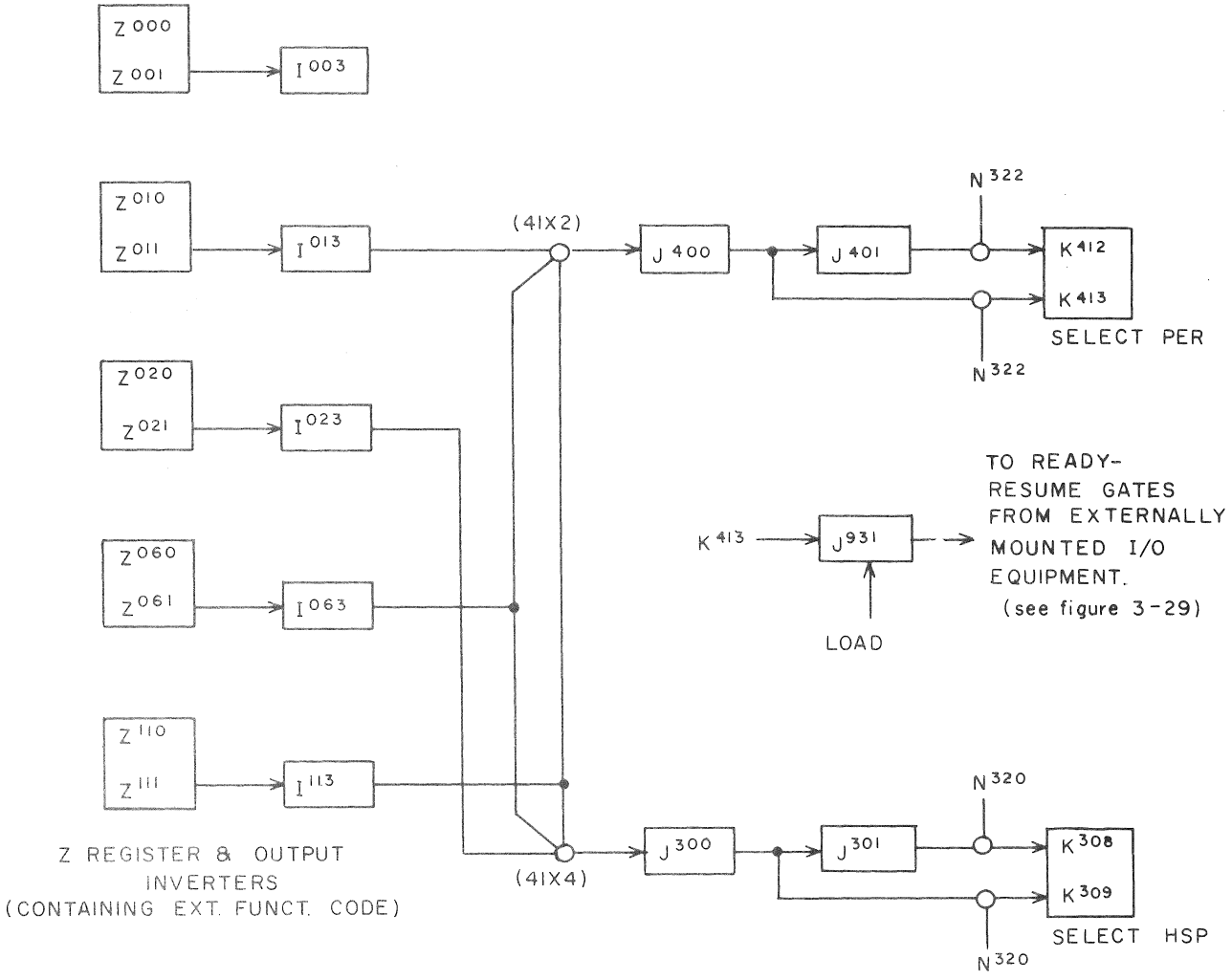


Figure 3-32. Normal Selections of PER and HSP

PER Input to Z

Data from the reader may be assembled in Z as a group of 12-bit words (load mode) or as separate 7-bit words (normal input).

Load Mode, PER to Z - First-frame data (containing seventh-level hole) must be entered in the upper half of Z (Z_U), since those bits are most significant; data from the second frame must be stored in Z lower (Z_L). The PER input to Z is shown at the extreme left of figure 3-33 (M430...M437). Input controls are at the top of the figure.

In load mode, PER→INP is "1", since LOAD is "1", and EXT→INP is forced to "0". As a result, first-frame information is fed into J410...J416. If bit one (M430) is "1", J410 is "0", and the top input to J436 (Store Seventh Level Control) is not met. The left-hand input is similarly unsatisfied, $\overline{\text{LOAD}}$ is "1", while the lower AND remains incomplete since the current frame is odd. The resultant "1" from J436 enters Z060/061 as soon as the Sense Seventh Level FF is set. Since Load Mode Control (K404/405) is also set, the PER→Z_U transfer card, J406, will allow the first six bits to be entered into Z060/061 through Z110/111. Meanwhile, the seventh-level hole has caused K404/405 to clear, enabling J405 (PER→Z_1) and the lower half of Z contains information from the second frame.

Normal Input, PER to Z - With PER selected, PER→INP is "1". Input cards, J410 through J416 and J419, will respond to holes in the tape by producing zeros. Since load mode is not in effect, K405 is a "1"; K404 is a "0". The latter input disables PER→Z_U during normal PER input instructions; the reader clutch is energized as soon as J438 goes to "1". Data from the tape is transferred to I, and then to Z. The I→Z control (used for other than a PER input) is blocked by J205 to prevent the possibility of introducing unwanted data into Z.

During normal input operations, J436 and J437 will set the seventh and eighth stages of Z if holes for these levels are present.

Normal Input, EXT to Z

If input equipment other than PER is selected, EXT→INP is "1". The 12 input lines M400 through M411 then feed the associated J4-- elements, the contents of which are transferred to I and later to Z.

HIGH-SPEED PUNCH

The computer console output device is a Teletype BRPE-11 paper tape punch (figure 3-34). The punch is selected by a 4XX4 EXF code, and the function ready signal acts through the ready control delay of the output resume resync circuit to restart the main timing chain.

An eight-stage punch storage register (PSR) is filled with output data from the Z-register whenever a punch output operation is executed. As data enters PSR from Z, the main timing chain is restarted to allow the computer to continue with high-speed computation while the punch records the output word at a slower rate.

When the computer is master cleared prior to program execution, the Punch Ready and Punch FFs are cleared to receive the first data word. Selection of HSP provides the Z → PSR to gate the output word into PSR. As the punch contacts close the Punch Ready and Punch FFs are set, the output disable is removed from the output amplifiers, and the word in PSR passes to the punch magnets. In this manner the feed hole and data holes are punched in the tape.

Setting the Punch FF clears the Punch Ready FF after a slight delay which fixes the length of time the punch magnets are energized during the feed (or code) pulse cycle. Both FFs must be cleared before new data can be accepted into PSR. Should the punch sync signal occur simultaneously with the setting of the Punch Ready FF, clearing of Punch Ready is delayed until the data has registered in PSR.

Operation of this circuit may also be started by setting the Load-Clear switch to LOAD. This allows the operator to perform a tape feed when the punch is in the normal recessed position with motor running.

PUNCH CONTROL UNIT (Chassis 10400)

The punch control unit (figure 3-35) contains the punch magnet puller modules and the -28v d-c power supply for those magnets. The -28v d-c power supply consists of a full-wave rectifier connected to the center-tapped secondary of a 115/64 vac transformer. An LC filter smooths the rectified voltage. The -28v fuse is mounted on the relay chassis.

Nine puller circuits (one feed hole and eight data levels) are contained on three type 86 cards. Operating voltages for the transistors on these cards are obtained from the computer power supply (+20, -20), and the circuits are driven by standard output amplifiers (L cards).

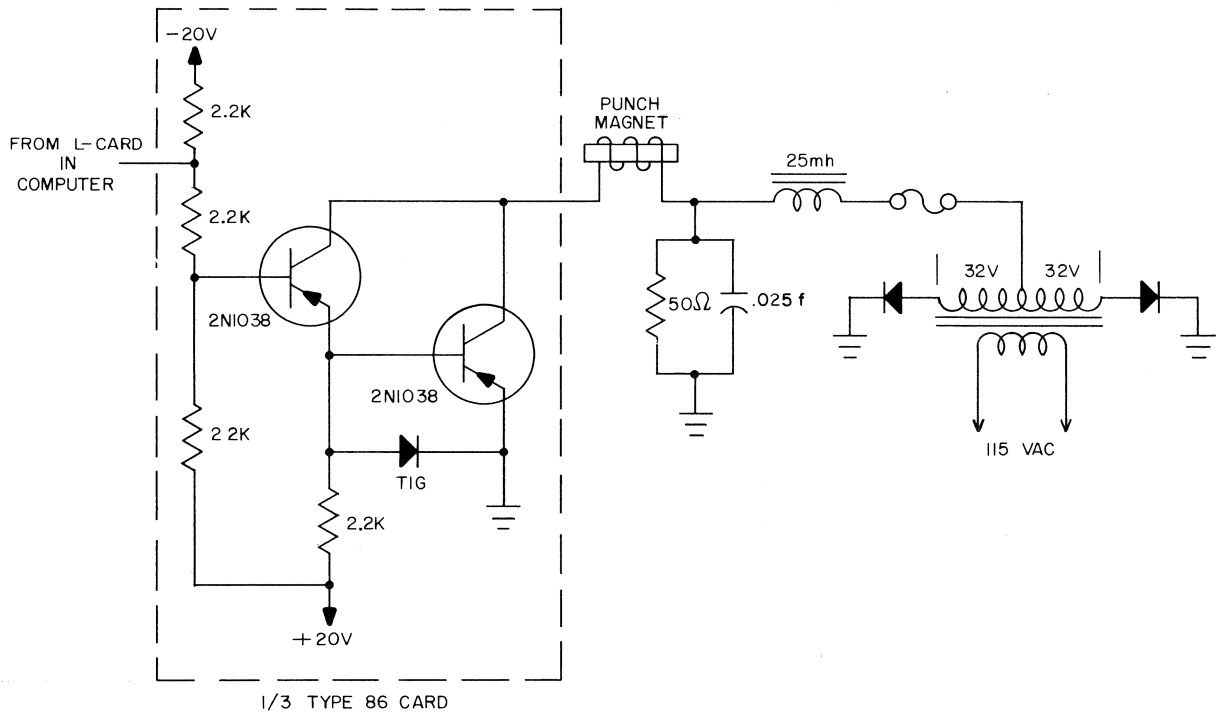


Figure 3-35. Punch Control Unit (Simplified Schematic)

CAPACITIVE DELAY NETWORKS

Capacitive delays are used in certain areas of computer logic, particularly input-output circuits, to delay the recognition of a change from the "0" to the "1" state. The delay, accomplished by connecting an integrating circuit to the junction of a pair of logic cards, may be a fixed value or variable through a limited range.

In the fixed delay, figure 3-36, the logic input to card B is delayed by a time constant which is the product of the 6800 ohm resistor on card B and the capacitor C. With the output transistor on card A grounded (logical "0" out), C is discharged to ground. When the transistor is switched off (logical "1" out), C begins charging through the 6800 ohm resistor until the threshold level of card B (-3.0v) is realized, a point determined by the time constant RC.

The actual delay time, as observed on an oscilloscope, for any fixed delay may vary considerably due to circuit constants. The delay times selected allow sufficient latitude for the circuits to operate successfully with these variations.

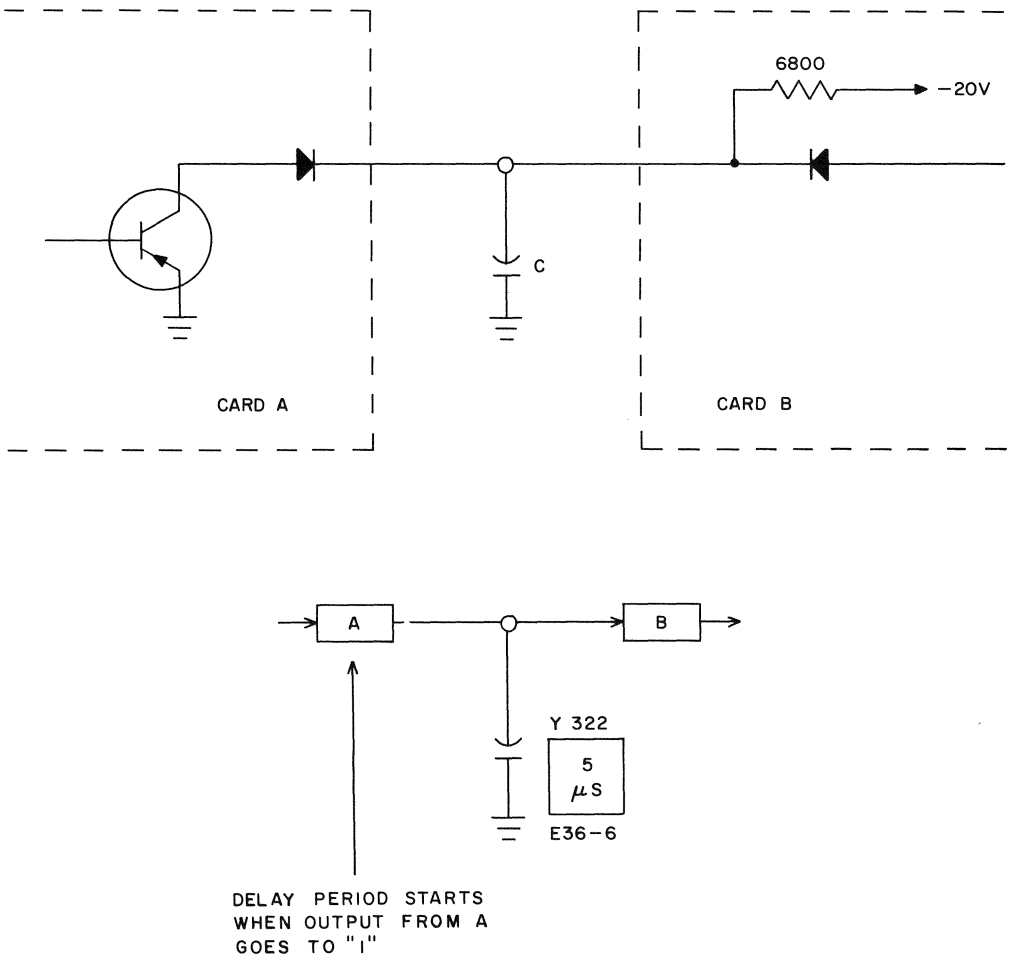


Figure 3-36. Fixed Delay

When a critical delay period is necessary a variable delay network will permit accurate adjustment. In the circuit shown in figure 3-37, a variable resistor on the 73A delay card may be adjusted to the desired time constant. The value of R in the RC factor is the effective parallel resistance of 2200 ohms and the selected setting of R.

In this circuit, the driving card is an output amplifier (L-card). This is necessary since the variable resistance on card C may draw more current than could be handled by the output transistor of a conventional logic card. An input amplifier (M-card) is required to return the logic levels to -3v and 0.5v.

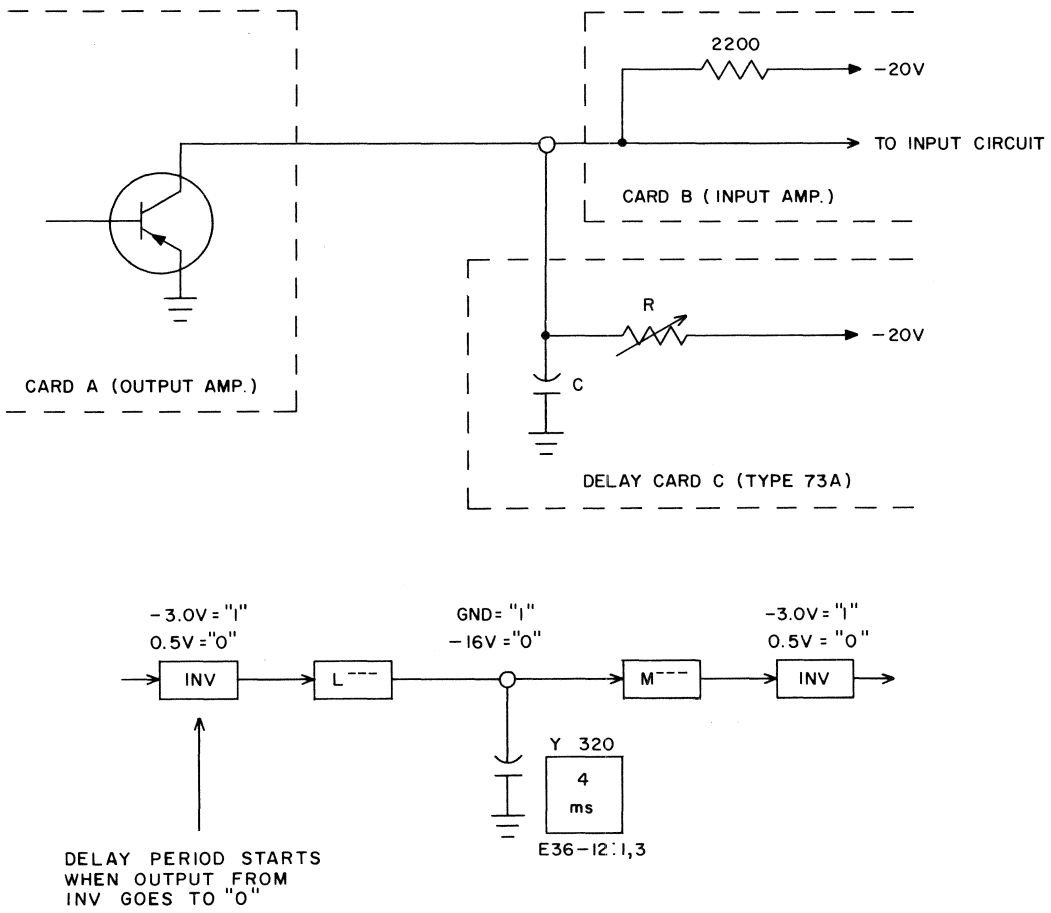


Figure 3-37. Variable Delay

When either delay is used, a Y logic symbol and a coordinate position define the location of the delay card. Numbers following the dash identify pins to which the delay components are connected. In the variable delay card in figure 3-38, the variable resistor may be connected to as many individual capacitors as necessary to produce the desired delay range; the exact delay is resolved by adjusting the resistor. It is possible for one 73A card to provide one variable delay (for example, pins 1 and 6), and up to eight fixed delays, or to provide nine fixed delays, (pin 1 not used).

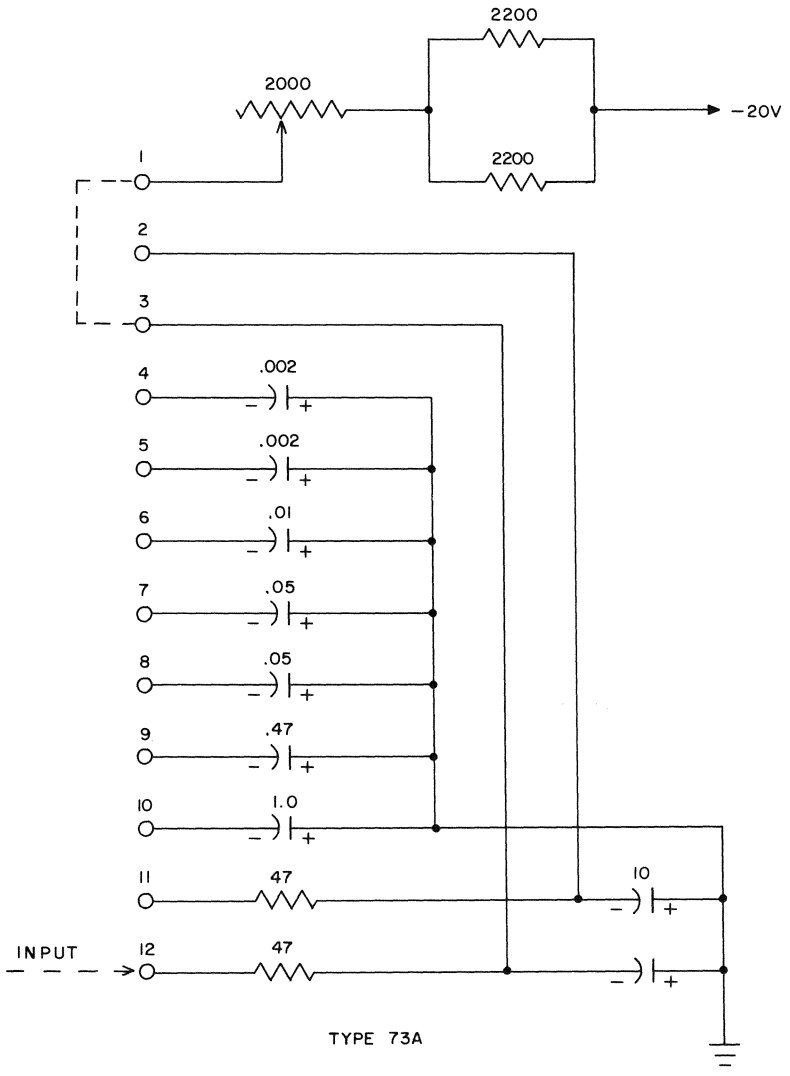


Figure 3-38. Capacitive Delay Card

The control network circuit for the high-speed punch uses capacitive delay (figure 3-39). Requisites for this circuit are a short (5us) clear pulse to PSR and an accurately adjusted 4ms delay to govern the length of the punch cycle.

With both FFs cleared, CLEAR is held to a "0" by the Punch FF, while the L-card output is at ground, thereby negating the delay network. When an input pulse sets Punch Ready, the Punch FF is also set at the next sync pulse. Clear cannot be energized, however, since Punch Ready holds it to a "0". The delay circuit begins to charge, and after 4ms a pulse from INV clears Punch Ready. The length of the resulting clear pulse is governed by the delay circuit connecting the clear sides of the two FFs.

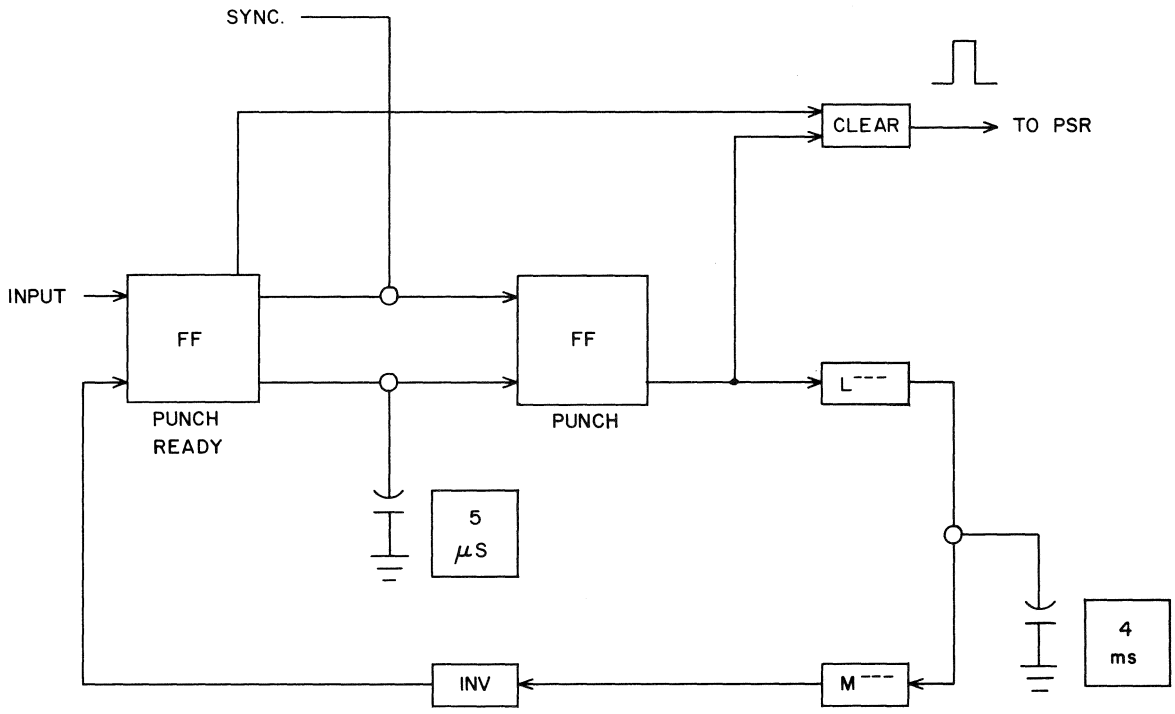


Figure 3-39. Capacitive Delay Application

APPENDIX A

GLOSSARY OF COMPUTER TERMS

The following glossary gives the meaning of terms that are used in a relatively specialized sense in this manual. In most cases, no attempt has been made to offer general definitions of the terms.

ADDER - In general, a device used to add two quantities. Specifically, the borrow pyramid in the subject computer. Also infrequently referred to as the "adder pyramid".

ADDRESS - The number designating a memory location; also used to refer to the memory location itself.

AND FUNCTION - A logical function in Boolean algebra that is satisfied (and thus has the value "1") only when all of its terms have the value "1". For any other combination of values of the terms it is not satisfied, and hence its value is "0".

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". The signal is sent to stage n + 1, which it complements.

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.

CARD - Any etched wiring board with attached components which is mounted on a chassis by means of a 15-pin connector.

CARRY - In an additive counter or accumulator, a signal indicating that in stage n, a "1" was added to a "1". The signal is sent to stage n + 1, which it complements.

CHAIN - A group of control delays connected serially and used for timing commands and signals.

CHANNEL - A transmission path (implemented by cables and connectors) that connects the computer to a given external equipment.

CHARACTER - Information handled by the computer:

1) A group of 6 bits representing bi-octal information. These bits may denote a binary quantity, a digit, letter or symbol. In Load mode, two 6-bit characters are assembled into one computer word.

2) A group of 7 bits representing an item of information. When the capacity of a device responding to a 72 (input) instruction is limited to 6 or 7 bits, those bits will be deposited in the lower portion of the selected storage address, with the remaining upper bits being zeros.

CLEAR - A command that removes a quantity from a register by placing every stage of the latter in the "0" state.

CLOCK PHASE - An output from the master clock: one of the two outputs is called the "even" phase and the other the "odd" phase. The inputs of a given control delay are gated by one clock phase and the outputs by the other.

COMMAND - A signal (generally initiated by a control sequence) 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 or word held in a register or storage location.

CONTROL DELAY - A device capable of receiving a signal pulse and "holding" it for a period determined by the basic clock frequency (0.2 usec) before allowing it to be released.

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 (if additive) or decreasing (if subtractive) its content by 1 upon receiving the appropriate command.

ENABLE - In general, to satisfy one of the conditions required for the occurrence of a command. Specifically, a signal generated in the borrow pyramid which indicates that stage (or group of stages) is not able to satisfy a borrow made upon it.

END-AROUND BORROW - A borrow that is generated in the highest-order stage of an accumulator or counter, and is sent directly to the lowest-order stage. Provision for such a borrow makes the accumulator or counter "closed".

ENTER - To manually insert into storage a quantity entered in the Z register. The operation consists of energizing the ENTER mode, inserting the desired storage address in P and the desired information in Z, and putting the computer in either RUN or STEP.

EXECUTION ADDRESS - The lower 6 bits of a 12-bit instruction. Most often used to specify the storage address of an instruction or operand. Sometimes used as the operand itself.

FLIP-FLOP (FF) - A storage device with two stable states: "1" and "0". A "1" input to the set side puts the FF in the "1" state; a "1" input to the clear side puts the FF in the "0" state. After the input is gone, the FF remains in a state indicative of the last "1" input. A stage of a register consists of a FF.

FUNCTION CODE - The upper 6 bits of a 12-bit instruction.

GATE - Verb: to satisfy one of the conditions required for the occurrence of a signal.
Noun: may denote either the AND function or the OR function, but more commonly the former.

INPUT DISCONNECT - During an input instruction, a signal sent to the computer by certain external devices which indicates that the device has completed all available transmissions to the computer.

INPUT REQUEST - A request, by the computer, for information from an external device. Occurs during input instruction only. (See Resume.)

INSTRUCTION - A 12-bit quantity consisting of a function code and an execution address. The instruction directs the computer to take a certain action.

INVERTER - A circuit which provides as an output a signal that is opposite to its input. If input is "1", output is "0". An inverter output is "1" only if all the separate OR inputs are "0".

LOAD - To take a quantity from storage and place it in a register.

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". The logical product of two quantities consists of bits, each of which is the AND function of corresponding bits of the two quantities. 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".

MASK - In the formation of the logical products of two quantities, one of them may be used as a mask for the other. The mask determines what part of the other quantity is to be considered. Wherever the mask is "0" that part of the other quantity is cleared, but wherever the mask is a "1", the other quantity is left unaltered.

MASTER CLEAR (MC) - A general command produced by placing the Load-Clear switch is the down (CLEAR) position. A 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 procedure, 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". The one's complement of a number is formed by complementing each bit individually, that is, changing a "1" to "0" and a "0" to a "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.

OR FUNCTION - A logical function in Boolean algebra that is satisfied (and thus has value "1") when any of its terms has the value "1". It is not satisfied when all terms are "0". Often called the inclusive OR function.

OVERFLOW - The condition in which the capacity of a register is exceeded.

PASSED BORROW - A borrow signal which acts upon a stage after having traversed the adjacent lower-order group by virtue of enable conditions within that group.

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 computer routine; a plan for the solution of a problem.

PYRAMID - A network of inverters that senses borrow conditions and produces borrow signals.

RANK - Registers composed of a pair of flip-flops per stage consist of two ranks, each containing one FF from the pair for each stage. Inverters arranged for parallel transmissions with one inverter per bit are also called ranks.

READ - To remove a quantity from a storage location.

READY - The input-output control signal sent by either the computer or an external equipment to alert the device that is to receive a transmission. The ready signal indicates that the word of character is available for transmission.

REPLACE - In the title of an instruction, refers to the fact that the result of the execution of the instruction is stored in the same 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.

RETURN JUMP -	A jump instruction which prepares for continuing the first sequence after the second is completed.
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. In essence performed when an instruction enters a negative number into a register.
SKIP -	To omit the execution of an instruction in a program .
SLAVE -	Usually an inverter which receives an unconditional input from another circuit and is used to provide more outputs than are available from the first.
STAGE -	The flip-flops and inverters associated with a given bit position of a register.
TOGGLE -	To complement each individual bit of a quantity as a result of an individual condition.
TRANSMISSION, CLEARED -	A transmission where both "1" and "0" are transferred into a register which has not been cleared previously.
TRANSLATION -	An indication of the content of a group 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 the given number from the bit "0". The two's complement of a number 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 12 bits.
WRITE -	To enter a quantity into a storage location.

APPENDIX B

NUMBER SYSTEMS

Any number system may be defined by two characteristics, the radix or base and the modulus. The radix or base is the number of unique symbols used in the system. The decimal system has ten unique symbols, 0 through 9. Modulus is the number of unique quantities, or magnitudes a given system can distinguish. For example, an adding machine with ten digits, or counting wheels, would have a modulus of $10^{10}-1$. The decimal system has no modulus because an infinite number of digits can be written, but the adding machine has a modulus because the highest number which can be expressed is 9999999999.

In most number systems, the relative position of a symbol determines its magnitude. In the decimal system, a 5 in the units column represents a different quantity than a 5 in the tens column. The 10 symbols are coefficients of ascending powers of the base 10. The number 984 is:

$$\begin{aligned} 9 \times 10^2 &= 9 \times 100 = 900 \\ 8 \times 10^1 &= 8 \times 10 = 80 \\ 4 \times 10^0 &= 4 \times 1 = 4 \\ & \qquad\qquad 984 \end{aligned}$$

Quantities less than 1 may be represented by using the 10 symbols as coefficients of ascending negative powers of the base 10. The number 0.593 is therefore:

$$\begin{aligned} 5 \times 10^{-1} &= 5 \times .1 = .5 \\ 9 \times 10^{-2} &= 9 \times .01 = .09 \\ 3 \times 10^{-3} &= 3 \times .001 = .003 \\ & \qquad\qquad 0.593 \end{aligned}$$

BINARY NUMBER SYSTEM

Computers operate faster and more efficiently by using the binary number system. There are only two symbols 0 and 1; the base = 2. The following table shows the position value.

2^5	2^4	2^3	2^2	2^1	2^0	
=32	=16	=8	=4	=2	=1	Binary point

The binary number 0 1 1 0 1 0 represents:

$0 \times 2^5 = 1 \times 32 = 0$
$1 \times 2^4 = 1 \times 16 = 16$
$1 \times 2^3 = 1 \times 8 = 8$
$0 \times 2^2 = 1 \times 4 = 0$
$1 \times 2^1 = 1 \times 2 = 2$
$0 \times 2^0 = 0 \times 1 = 0$ Total: 26

Fractional binary numbers may be represented by using the symbols as coefficients of ascending negative powers of the base.

Binary numbers can be converted to decimal by power addition.

$$\begin{aligned} 010111 &= 1\,(2^4) + 0\,(2^3) + 1\,(2^2) + 1\,(2^0) \\ &= 1\,(16) + 0\,(8) + 1\,(4) + 1(2) + 1\,(1) \\ &= 23_{10} \end{aligned}$$

Conversion of decimal numbers to binary:

$45_{10} \div 2$	= 22 remainder 1	record	1
$22 \div 2$	= 11 remainder 0	record	0
$11 \div 2$	= 5 remainder 1	record	1
$5 \div 2$	= 2 remainder 1	record	1
$2 \div 2$	= 1 remainder 0	record	0
$1 \div 2$	= 0 remainder 1	record	1
			101101

Thus $45_{10} = 101101_2$

Binary numbers are added according to the following rules:

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

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

	Binary	Decimal
Augend	0111	7
Addend	+0100	+4
Partial Sum	0011	
Carry	1	
Sum	1011	11

Subtraction may be performed as an addition. Decimal examples:

8 minuend	or	8 minuend
-6 subtrahend		+4 10's complement of subtrahend
2 difference		2 difference - omit carry

Subtraction performed by adding the complement:

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

Omitting the carry has the effect of reducing the result by 10.

To complement a binary number, (one's complement), subtract each bit of the number from 1.

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.

OCTAL NUMBER SYSTEM

The octal number system uses eight unique symbols, 0 through 7. With the base eight the position value is:

8^5	8^4	8^3	8^2	8^1	8^0
32,768	4,096	512	64	8	1

The octal number 513_8 represents:

$$5 \times 8^2 = 5 \times 64 = 320$$
$$1 \times 8^1 = 1 \times 8 = 8$$
$$3 \times 8^0 = 3 \times 1 = 3$$
$$331_{10}$$

Computer arithmetic is expressed in octal. Three binary digits may be expressed by one octal digit, thus reducing the total digits involved by a factor of 3. The octal system is selected as a basis for obtaining shortened expressions for numbers because of the ease with which numbers can be converted from binary to octal and from octal to binary notation.

The decimal number 38 is expressed in binary by 100110. To convert this expression to octal, separate it into groups of three digits, 100 in binary is 4 in octal, 110 in binary is 6 in octal; therefore, the binary expression 100110 is 46 in octal.

To convert an octal number to binary reverse the procedure. To convert the octal expression 513 to its binary equivalent convert each octal digit:

5	1	3
101	001	011

The binary expression for an octal digit must be filled out with "0's" to make it ree digits in length.

Octal notation is used in referring to the content of a register, the value of a contro signator, addresses in storage, and codes for instructions. The octal number systen not actually used in the operation of the computer.

Conversion of Decimal to Octal is as follows:

(1) Decimal_{10} to Octal_8

Integer example

$273_{10} \div 8 = 34$ remainder 1 record 1

$34_{10} \div 8 = 4$ remainder 2 record 2

$4_{10} \div 8 = 0$ remainder 4 record 4

421

Thus: $273_{10} = 421_8$

(2) Octal_8 to Decimal_{10}

Integer example

$$324_8 = 3(8^2) + 2(8^1) + 4(8^0)$$
$$= 3(64) + 2(8) + 4(1)$$
$$= 192 + 16 + 4$$

Thus: $324_8 = 212_{10}$

TABLE 1. COMMON PURE NOTATIONS

Decimal	Binary	Octal
00	00000	00
01	00001	01
02	00010	02
03	00011	03
04	00100	04
05	00101	05
06	00110	06
07	00111	07
08	01000	10
09	01001	11
10	01010	12
11	01011	13
12	01100	14
13	01101	15
14	01110	16
15	01111	17
16	10000	20
17	10001	21

TABLE 2. POWERS OF COMMON NUMBER SYSTEMS

$2^0 = 1$	$8^0 = 1$	$10^0 = 1$
$2^1 = 2$	$8^1 = 8$	$10^1 = 10$
$2^2 = 4$	$8^2 = 64$	$10^2 = 100$
$2^3 = 8$	$8^3 = 512$	$10^3 = 1,000$
$2^4 = 16$	$8^4 = 4,096$	$10^4 = 10,000$
$2^5 = 32$	$8^5 = 32,768$	$10^5 = 100,000$
$2^6 = 64$	$8^6 = 262,114$	$10^6 = 1,000,000$
$2^7 = 128$	$8^7 = 2,097,152$	
$2^8 = 256$	$8^8 = 16,777,216$	
$2^9 = 512$		
$2^{10} = 1,024$		

TABLE 3. RADIX OF COMMON NUMBER SYSTEMS

System	Digits	Radix
Decimal	0, 1, 2, 3, 4, 5, 6, 7, 8, 9	10
Binary	0, 1	2
Octal	0, 1, 2, 3, 4, 5, 6, 7	8

BOOLEAN ALGEBRA

In some respects Boolean algebra is similar to ordinary algebra. Because of the differences, it is simpler to regard it as unrelated to ordinary algebra. The logical equations which describe the connections of building blocks in the computer are a specialized form of Boolean algebra as applied to switching circuits.

In Boolean algebra, there are only two values or quantities to be considered, "1" and "0". In a Boolean equation the variables, or literals, are restricted to these values which are considered opposites; one is the negation of the other.

There are three operations used in such equations. The first is logical product, the AND function of two terms, indicated by a dot between the terms or by the absence of any symbol between them. This function is satisfied only when both terms are "1".

The second operation is logical sum, the OR function of two terms, indicated by a plus sign between the terms. An OR function of two terms is satisfied when one or the other, or both, of the terms are "1"; it is not satisfied only when both terms are "0". This is the inclusive OR rather than the exclusive.

The third operation is negation, the NOT function of a term, and is indicated by a bar over the term. If a term is "1", the negation of that term has the value "0", and vice versa.

The grouping of terms is indicated by parentheses. In Boolean equations for switching circuits, the circuit inputs are represented by literals which appear on the right side of the equal sign. The literal on the left side of the equal sign represents the circuit output, which is a function of the inputs on the right side of the equation.

Boolean equations are of two types: identities and transfer formulas. An identity consists of two equivalent expressions. For example, the equation $(A + B)C = AC + BC$ is an identity which states that either A or B in combination with C, is equivalent to either A in combination with C, or B in combination with C. The equation $C = A + B$ is a transfer formula; a "1" is transferred to element C if a "1" is in either element A or element B.

A Boolean expression may be reduced to its simplest equivalent by applying the basic identities of table 4. This procedure not only provides a means of understanding the circuits, but also of simplifying switching circuits by reducing the number of components necessary to perform the operations specified by a particular transfer equation. For example, the transfer equation D = AB + B + C can be reduced to its equivalent D = B + C by applying the theorems of table 1 to strike out the expression AB. Because the reduced transfer equation is simpler, a circuit built with the simplified equation contains fewer components to perform the same logical functions.

The theorems are also applied to convert the final simplified expressions into standard forms, which are more readily adaptable to a given type of switching circuit. For example, the equation D = (A + C) ($\overline{B}$) can be changed to the formula D = A$\overline{B}$ + C$\overline{B}$, which is applicable to the building block used in the computer.

By applying identities 22 and 23, an equation using the AND function can be converted into one using the OR function, and vice versa. For example, to convert Z - ABC into an equation in which the OR function is used in the right side negate both sides to obtain $\overline{Z}$ = $\overline{ABC}$. By applying identity 22 to the right side, $\overline{Z}$ = $\overline{A}$ + $\overline{B}$ + $\overline{C}$ is obtained. This procedure, called inverting the logic, has been used at times in the equations for the computer.

GLOSSARY OF BOOLEAN SYMBOLS

A B C D etc.	Switching elements. Literals denote active "1" when switching element is in the "1" state, or passive "0" input when switching element is in the "0" state.
$\overline{A}$ $\overline{B}$ $\overline{C}$ etc.	The bar denotes negation. A letter with a bar denotes an input that is active when the switching element is in the "0" state, and passive in the "1" state.
1	Active "1" input.
0	Passive "0" input.
+	Logical sum, OR function of two or more literals.
() ()	Logical product, AND function of two or more literals
=	Separates equivalent Boolean expressions or two halves of a transfer formula.

TABLE 4. BOOLEAN IDENTITIES

A. Commutative Laws

1. $A + B = B + A$
2. $BA = AB$

B. Associative Laws

3. $(A + B) + C = A + (B + C)$
4. $(AB)\,C = A\,(BC)$

C. Distributive Laws

5. $AB + AC = A(B + C)$
6. $A + BC = (A + B)(A + C)$

D. Forms Involving 1 and 0

7. $0 + 0 = 0$
8. $0 + 1 = 1$
9. $1 + 1 = 1$
10. $0 \cdot 0 = 0$
11. $0 \cdot 1 = 0$
12. $1 \cdot 1 = 1$
13. $A + 0 = A$
14. $A + 1 = 1$
15. $A \cdot 1 = A$
16. $A \cdot 0 = 0$

E. Forms With Repeated Literals

17. $A + A = A$
18. $A \cdot A = A$

F. Forms With Negation

19. $A = A$
20. $A\,\overline{A} = 0$
21. $A + \overline{A} = 1$
22. $\overline{A \cdot B \cdot C} = \overline{A} + \overline{B} + \overline{C}$ DeMorgan's Theorem
23. $\overline{A + B + C} = \overline{A} \cdot \overline{B} \cdot \overline{C}$

G. Forms Derived From Previous Identities

24. $A + AB = A$
25. $A(A + B) = A$
26. $(A + \overline{B})\,B = AB$
27. $A\,\overline{B} + B = A + B$
28. $BC = ABC + \overline{A}BC$

H. Proof of Identity 6

Conditions A B C	Left-Hand Expression A + BC	=	Right-Hand Expression (A + B) (A + C)
0 0 0	0		0
0 0 1	0		0
0 1 0	0		0
0 1 1	1		1
1 0 0	1		1
1 0 1	1		1
1 1 0	1		1
1 1 1	1		1