DEC-8/L-HR1B-D

PDP-8/L
MAINTENANCE MANUAL
VOLUME I

DIGITAL EQUIPMENT CORPORATION • MAYNARD, MASSACHUSETTS

1st Printing October 1968
2nd Printing May 1969
3rd Printing (Rev) December 1969
4th Printing July 1970

CONTENTS

CONTENTS (Cont)

CONTENTS (Cont)

ILLUSTRATIONS

ILLUSTRATIONS (Cont)

TABLES

PDP-8/L MAINTENANCE MANUAL

VOLUME 1

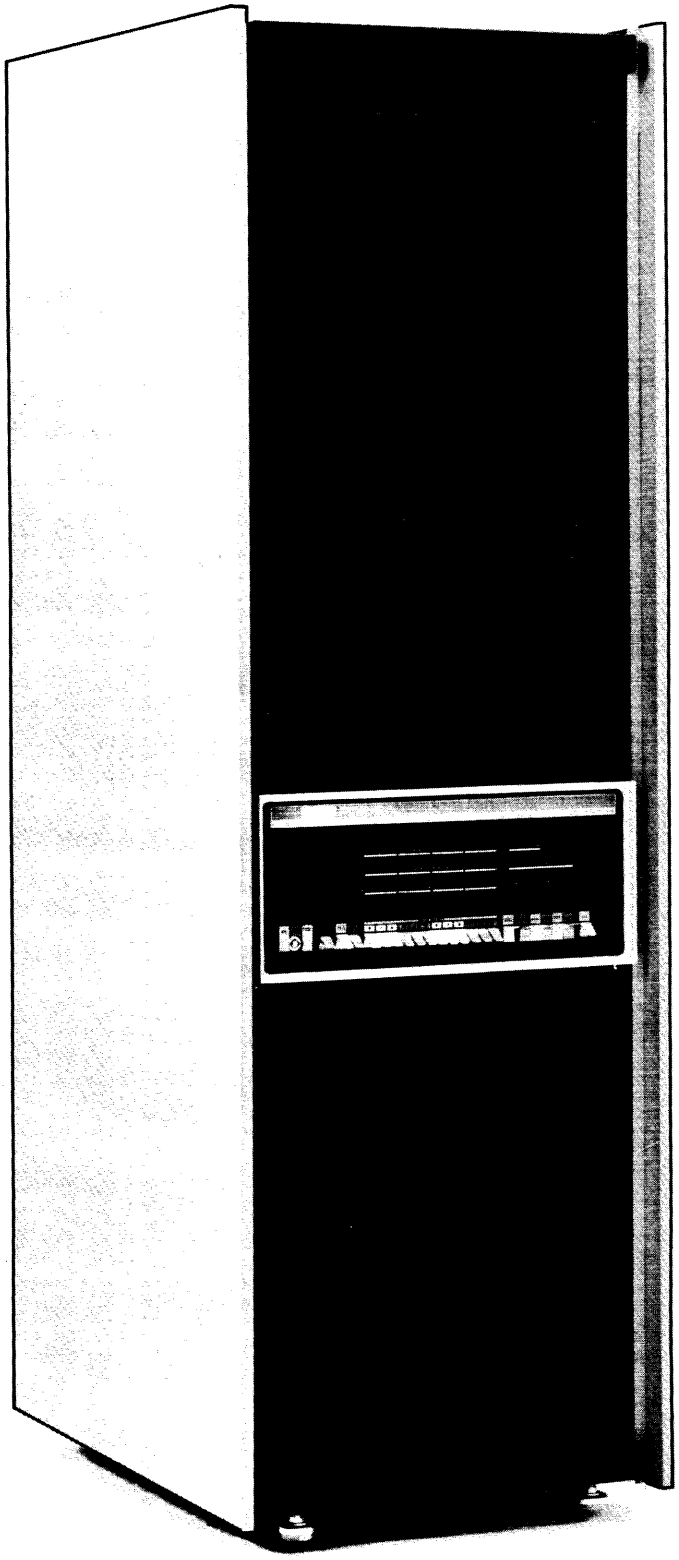


Figure 1-1 PDP-8/L

CHAPTER 1
INTRODUCTION AND DESCRIPTION

1.1 INTRODUCTION

This manual covers installation, operation, theory, and maintenance of The Programmed Data Processor - 8/L (PDP-8/L). It is the intent of this manual to provide the field service engineer or maintenance technician who is familiar with digital logic circuitry with the information he needs to install and maintain a PDP-8/L system. The bulk of the manual assumes that the reader is conversant with DIGITAL's system of logic notation. If this is not the case, the reader should refer to applicable sections in the appendix for a description.

1.2 DESCRIPTION

The Digital Equipment Corporation integrated-circuit Programmed Data Processor-8/L (PDP-8/L) (see Frontispiece) serves as a small-scale general-purpose computer that functions as an independent information-handling facility in a large computer system, or as the control element in a complex processing system.

The PDP-8/L is a one-address, 12-bit, fixed-word-length, parallel computer using two's complement arithmetic. Normal cycle time of the 4096-word (referred to as 4K) random-access, magnetic core memory is 1.6 μs. An additional 4K of memory may be available to the system simply by adding an MC 8/L memory expansion and control unit.

Standard features of the system include indirect addressing, facilities for instruction skipping and program interruption as functions of input/output device conditions, and optional high-speed information transfers into peripheral mass memory devices via a cycle stealing data break.

The PDP-8/L performs one addition in 3.2 μs (with one number in the accumulator), permitting a computation rate of 312,500 additions per second to be achieved. It performs subtraction in 6.4 μs (with the minuend in the accumulator) using two's complement addition. Multiplication takes approximately 384 μs, using a subroutine that operates on two signed 12-bit numbers to produce a 24-bit product, leaving the 12 most significant bits in the accumulator. Division of two signed 12-bit numbers takes approximately 490 μs, using a subroutine that produces a 12-bit quotient in the accumulator, and a 12-bit remainder in core memory.

Flexible, high-capacity, input/output capabilities of the computer allow it to operate a variety of peripheral equipment. In addition to the Teletype keyboard/printer and perforated-tape reader/punch equipment supplied with a basic PDP-8/L, the system can operate a number of optional devices, such as a high-speed perforated-tape reader and punch, card reader, line printer, analog-to-digital converters, cathode-ray-tube displays, magnetic-drum systems, magnetic disk-file systems, and magnetic-tape equipment. Instruments or equipment of special design can also be connected into the PDP-8/L system. The computer itself needs no modification for the addition of these peripheral devices.

The PDP-8/L is completely self-contained, and, under normal conditions, requires neither special power sources nor rigidly controlled environmental conditions. A single source of 105-130 Vac, 47-63-Hz, single-phase power permits internal power supplies to produce all required operating voltages. M-Series* modules, using TTL-type integrated-circuit packs, ensure reliable operation in ambient temperatures between +10° and +55°C.

1.3 PERTINENT DOCUMENTS

The following documents serve as source material and complement the information in this manual:

1. Logic Handbook, C-105 (1969 edition) printed by DIGITAL, which notes the function and specifications of the M-Series modules and module accessories for the PDP-8/L.

2. PDP-8/L Users Handbook.

3. Technical Manual, Automatic Send and Receive Sets (ASR) (Bulletin 273B Volumes 1 and 2). This manual covers operation and maintenance of the Teletype unit.

*M Series is a registered trademark of Digital Equipment Corporation.

4. Parts, Model 33 Page Printer Set (Bulletin 1184B) contains an illustrated parts breakdown to serve as a guide to disassembly, reassembly, and ordering replacement parts for the Teletype unit.

5. Instruction List F-816, printed by DIGITAL. This is a shirt-pocket list of all memory reference instructions, all augmented instructions, the most common IOT instructions, and the ASCII code used with many I/O devices.

6. Instruction manuals and MAINDEC programs for appropriate input/output devices are prepared by DIGITAL.

7. Digital Program Library Documents. Perforated program tapes and descriptive matter for the Program Assembler Language (PAL III), FORTRAN, FOCAL, utility subroutines, and the maintenance programs (MAINDEC) prepared by DIGITAL are available to PDP-8/L users. The list of programs currently in the library and available is provided in Appendix D.

CHAPTER 2
INSTALLATION

This chapter contains installation and interface information for the PDP-8/L and its options.

2.1 SPACE REQUIREMENTS

If DEC cabinets are not purchased, the cabinets at the installation should provide for access to all doors and panels for maintenance.

The PDP-8/L is available in either the table top or rack-mounted configuration. The rack-mounted configuration and peripherals may be purchased completely installed in DEC cabinets or unmounted for installation in a customer cabinet. The standard teletype automatic send receive (ASR) set requires a floor space of approximately 22-1/4 in. wide by 18-1/4 in. deep. The teletype signal cable requires the teletype to be placed near the computer. Figures 2-1 through 2-4 provide PDP-8/L dimensions for both tape top and rack-mounted configurations.

Figure 2-1 illustrates the table top dimensions, and Figure 2-2 illustrates the rack-mounted PDP-8/L. The rack-mounted configuration is attached to the cabinet on sliding chassis tracks. There are two sets of tracks; one set for each frame side. The track sections interlock; one section being bolted to the cabinet and the other bolted to the logic main frame.

Figure 2-3 shows the chassis track specifications. A rear view of the rack-mounted PDP-8/L and track-slide specifications is illustrated in Figure 2-4.

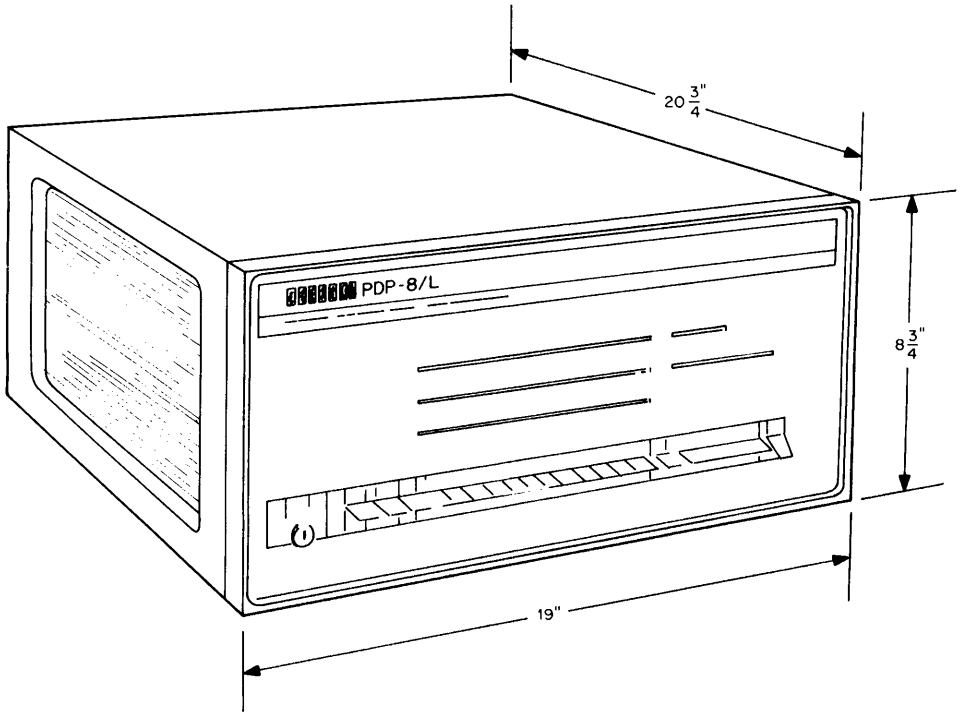


Figure 2-1 Table top PDP-8/L Dimensions

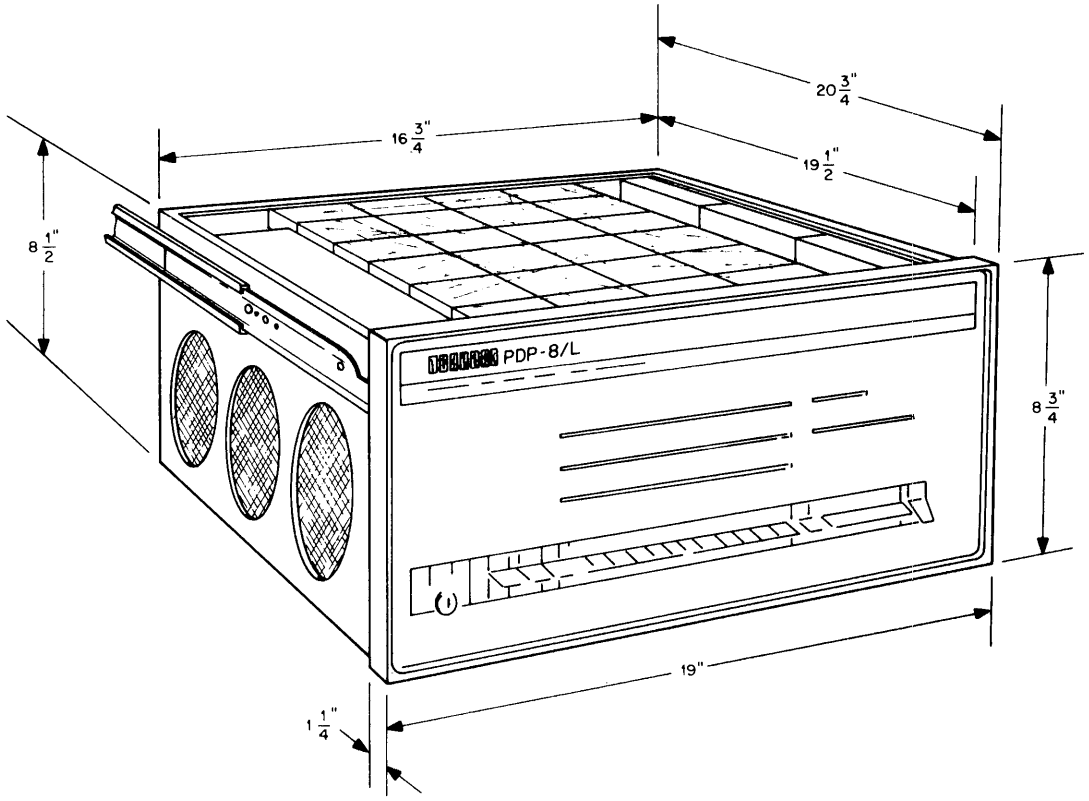


Figure 2-2 Rack-Mounted PDP-8/L

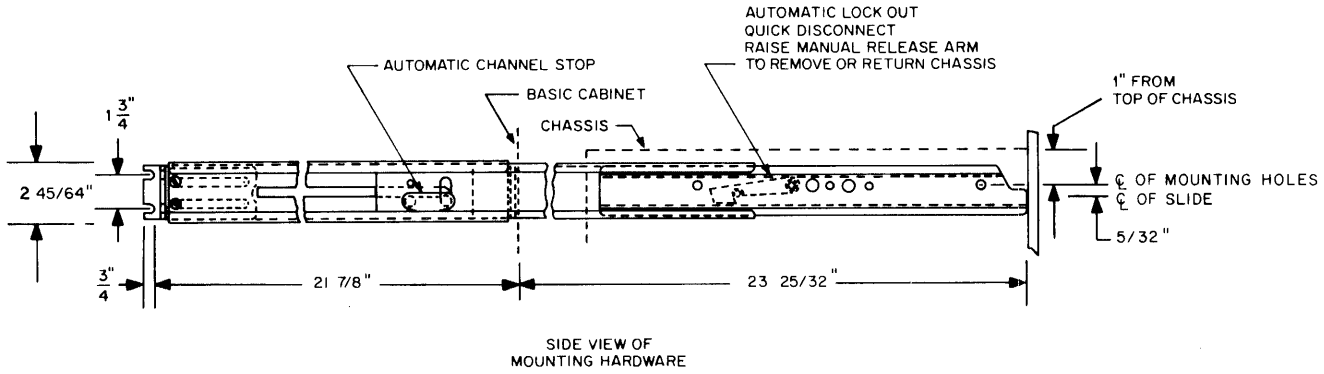


Figure 2-3 Chassis-Track Specifications

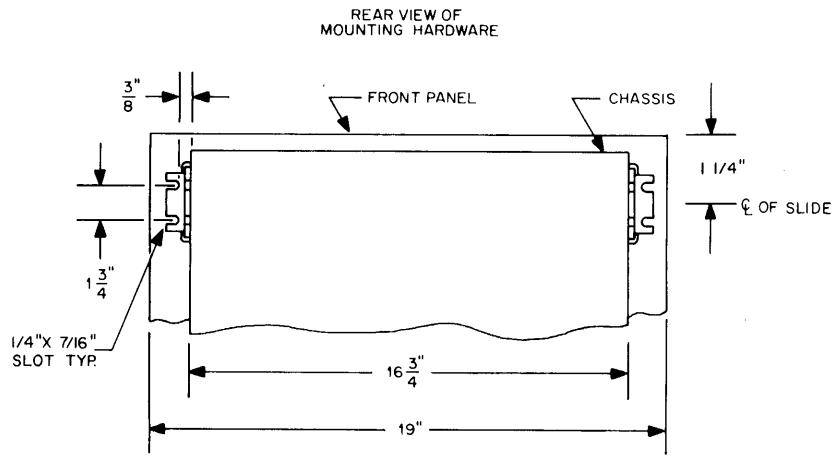


Figure 2-4 Rack-Mounted PDP-8/L Frame with Track-Slide Specifications

2.2 ENVIRONMENTAL REQUIREMENTS

The PDP-8/L is designed to operate between 50° and 130° F (0° and 55° C). The life-expectancy of the system, however, can be extended if the temperature at the installation site is maintained at between 70° and 85° F (between 21° and 30° C).

During shipping or storing of the system, the ambient temperature should be maintained within 32° to 130° F to prevent damage to the system. Although all exposed surfaces of Digital cabinets and hardware are treated to prevent corrosion, exposure of systems to extreme humidity for long periods of time should be avoided.

For external options, forced air cooling should be used if more than a few module-filled H911 mounting panels are needed. The low power consumption of M-series modules results in approximately 15W dissipation in a typical H911 mounting panel of 64 modules. If only one or two panels of logic are used, convection cooling is sufficient.

2.3 INTERNAL OPTION INSTALLATION

The installation of the internal options involves the addition of the logic modules in the proper locations. WHEN INSERTING OR REMOVING ANY MODULES IN THE LOGIC FRAME THE 8/L SYSTEM MUST BE TURNED-OFF. Refer to the module utilization drawing for the locations of each module of all internal options. If an option involves an external device, dress connecting cables through the opening at the rear of the PDP-8/L to the option.

2.4 INTERFACE

2.4.1 I/O Bus

The PDP-8/L uses a series I/O bus system to permit interface connections to be made between external devices and the 8/L without modifying the computer wiring. In a series I/O bus, the computer sends all I/O signals to the first device. This device uses the pertinent signals and sends all of the signals to the next I/O device (see Figure 2-5). This allows one set of cables to be connected to the PDP-8/L and two sets connected to each device: one receiving the I/O bus signals from the computer itself or the previous device, and one passing the I/O bus signals to the next device. The data break cables are used in the same manner.

When the equipment location does not make series bus connections feasible, or when cable length becomes excessive, additional interface connectors can be provided near the computer.

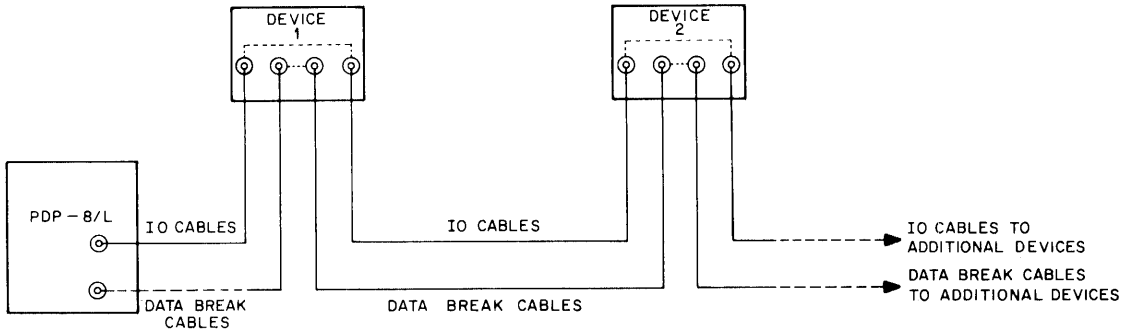


Figure 2-5 I/O Bus Configuration

2.4.2 I/O Cables

Eighteen conductor coaxial cables or flexprint cables with either M904 or M903 cable connectors, respectively, provide signal connection between the computer and optional equipment. These cables are connected by plugging the connectors into standard FLIP CHIP module receptacles. Use of coaxial cable protects systems from radiated noise and cross talk between individual lines. Coaxial cable used between the PDP-8/L and options has the following specifications:

$Z_0 = 95\,\Omega \pm 5\%$

C = 13.75 pf/ft approximately (unterminated)

L = 124 Nhy/ft approximately

R = 0.095 ohm per foot nominal

Y = 79% of velocity of light, approximately ($\approx$ 1.5 ns/ft.).

The flexprint cable is generally used because of its high flexibility. The conductors are #30 AWG flat copper.

The maximum length of I/O cabling, from the PDP-8/L to the last device is 50 ft. This can be 50 ft. of coax or a combination of coax and flexprint, in which case the flexprint cannot exceed a total length of 15 ft.

2.4.3 Cable Connectors

The M903 and M904 cable connectors are used with the cable types described above. M903 connectors are used with flexprint cable. The M904 connectors terminate the coaxial cable ends.

Both connectors are wired double-sided with 36 pin contacts. Each connector terminates two coaxial or flexprint cables. The conductors from both types of cables terminate at the connector pin locations indicated below. Each signal conductor is isolated by a ground conductor. The connector pin locations for signals are: B1, D1, E1, H1, J1, L1, M1, P1, S1, and D2, E2, H2, K2, M2, P2, S2, T2, V2. The ground pin locations on each connector are: A1, C1, F1, K1, N1, R1, T1, and C2, F2, J2, L2, N2, R2, U2.

2.5 INTERFACE CONNECTIONS

All interface connections to the PDP-8/L are made at assigned module receptacle connectors in the mounting frame. Capital letters designate horizontal rows of modules within a mounting frame from top to bottom, i.e., A is the first row, B is the second row, etc. Module receptacles are numbered from left to right as viewed from the wiring side (right to left from the module side). Terminals of a connector or module are assigned capital letters from top to bottom omitting G, I, O, and Q. Double sided connectors or modules are used with the suffix number "1" designating the one side and suffix number "2" designating the other side.

Figure 2-6 shows the signals and pin locations at each connector block. The I/O cable connectors are inserted into these locations to form the I/O bus. The arrows associated with each signal show the signal direction. The arrows pointing to the connector block

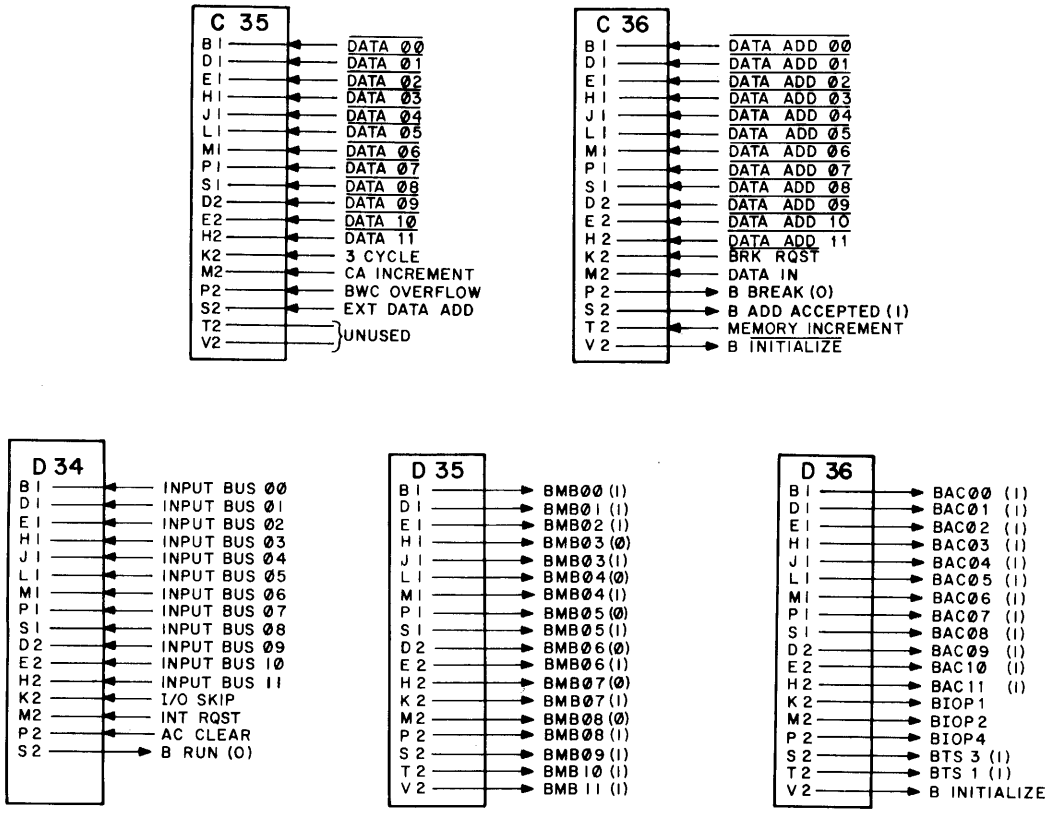


Figure 2-6 I/O Signal Connections

indicate signals coming to the computer from the I/O bus. Similarly, the arrows pointing away from the connector block indicate signals from the computer to external devices through the I/O bus.

2.6 EXTERNAL EQUIPMENT CABLE LOCATIONS

The I/O cables connected within the computer at the locations specified by Figure 2-6 are generally connected on an option mounting panel as indicated in Figure 2-7. This illustration shows the DEC convention for cable terminations.

Module slot locations 1 through 3 (looking at the wiring pin side) in the A row of an option mounting panel are reserved for program interrupt cable connections in (or out). Module slot locations 4 and 5 are reserved for Data Break cable connections in (or out).

Module slot locations 1 through 5 in the B row of the option mounting panel are wired in parallel with the top module slot locations 1 through 5. To continue the I/O cabling to the next device, the bottom slots are used.

2.7 BUS DRIVER/RECEIVER MODULES

The following paragraphs describe each bus driver and receiver module type shown on Engineering Drawing BS-8L-0-10. These circuits provide isolation and drive for signals between the processor and I/O cables.

2.7.1 M111/M906 Positive Input Circuit

Figure 2-8 shows the M111/M906 logic configuration. The M111 Inverter module is used in conjunction with the M906 Cable Terminator module which clamps the input to prevent excursions beyond +3V and ground.

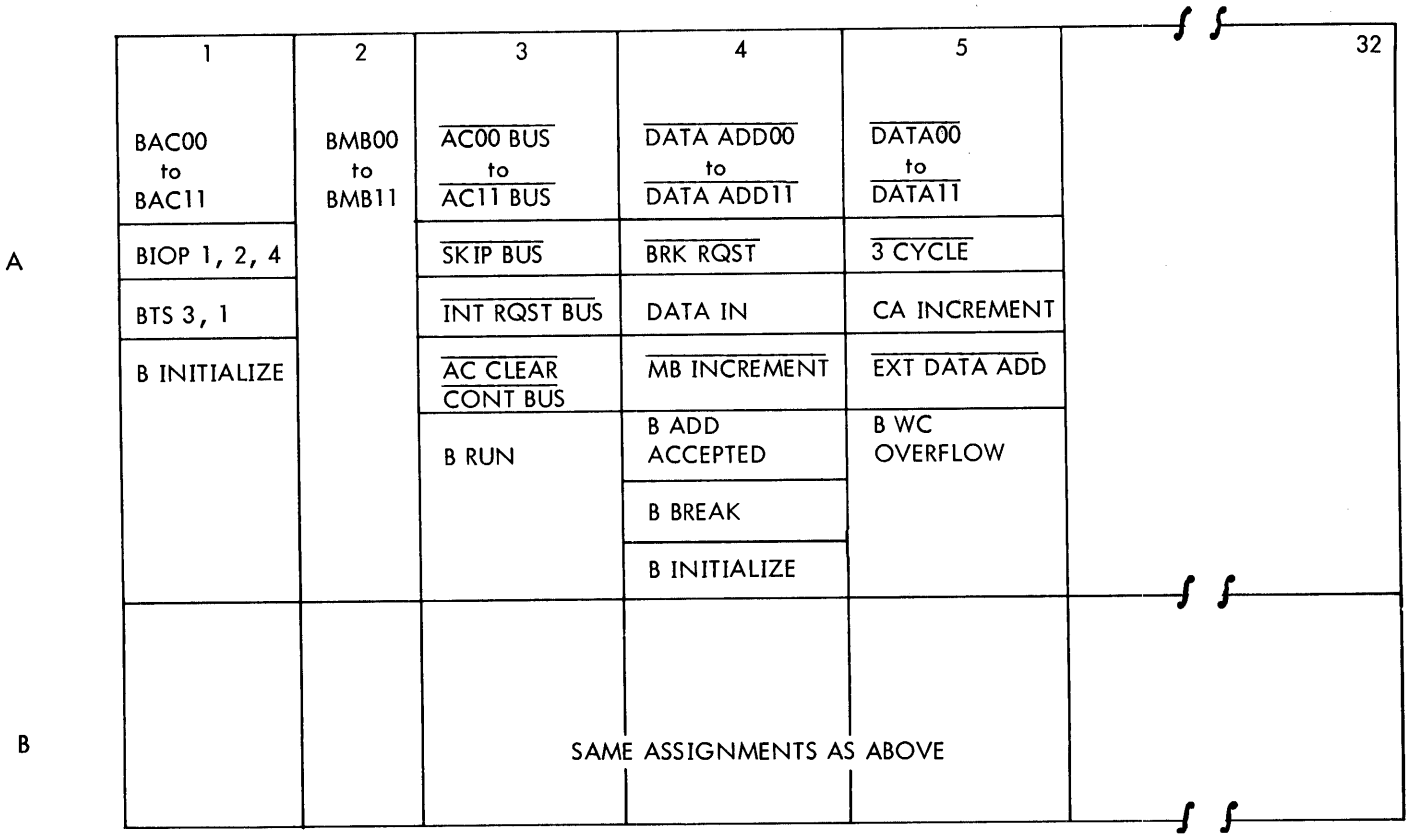


Figure 2-7 Cable Location Convention on Option Mounting Panels

The incoming DATA and DATA ADD signals for data break and other signals (Drawing BS-8L-0-10) are inverted and isolated from the I/O cables through this logic.

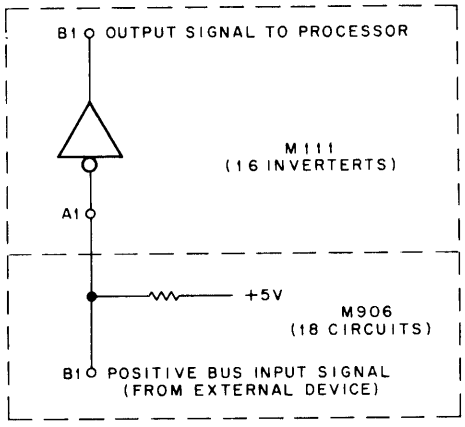


Figure 2-8 M111/M906 Logic Diagram

2.7.2 M516 Positive Bus Receiver Input Circuit

Figure 2-9 shows the M516 logic configuration. Six four-input NAND gates with overshoot and undershoot

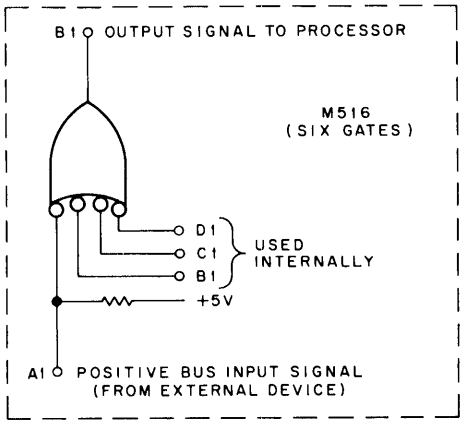


Figure 2-9 Typical M516 Positive Bus Receiver Input Circuit

clamp on one input of each gate. The I/O SKIP gate (Drawing BS-8L-0-10) is an example of M516 application. The input signals applied to pins B1, C1, and D1 are from internal options. With the I/O SKIP gate, these signals are PWR SKIP, RDR SKIP, and TT SKIP.

2.7.3 M623/M906 Positive Output Circuit

Figure 2-10 shows the M623/M906 logic configuration. The M623 Bus Driver module contains 12 negative NAND circuits. Used with the M906 Cable Terminator module, the output is clamped to prevent excursions beyond +3V and ground. The output can drive +5 mA at the high level and sink 20 mA at the low level. The AC00 through AC11 signals (Drawing BS-8L-0-10) are examples of M623 application.

2.7.4 M660 Bus Driver Output Circuit

Figure 2-11 shows the M660 logic configuration. This bus driver circuitry provides low impedance 100 ohm terminated cable driving capability, using M-Series levels, or pulses of duration greater than 100 ns. Each output can drive +5 mA at the high level and sink 20 mA at the low level, in addition to the termination current required by the G717 Termination module. The M660 module is used in the PDP-8/L for the following output signals: IOP 1, IOP 2, IOP 4, TS 3 and TS1 (Drawing BS-8L-0-10).

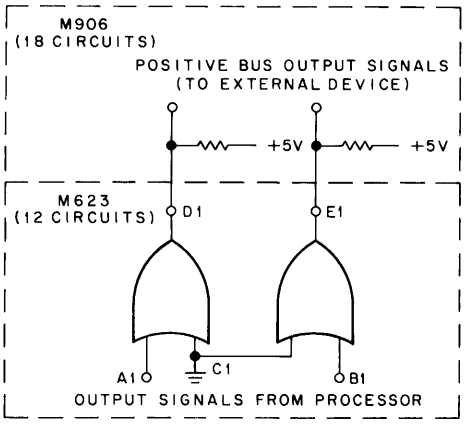


Figure 2-10
Typical M623/M906 Positive Output Circuit

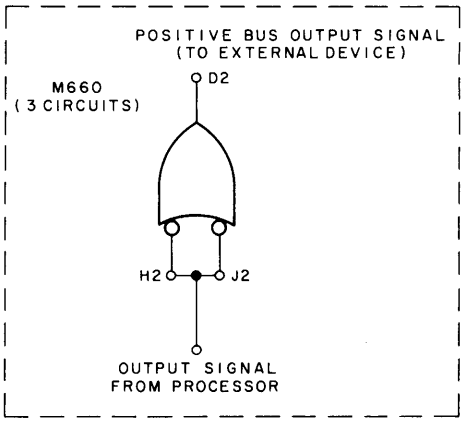


Figure 2-11
Typical M660 Bus Driver Output Circuit

CHAPTER 3
OPERATION

This chapter contains operating information for the PDP-8/L and the ASR33 Teletypewriter. Operating information for the peripheral input/output devices is contained in their respective manuals.

3.1 CONTROLS AND INDICATORS

The following paragraphs contain detailed information regarding the controls and indicators of the PDP-8/L and the ASR33 Teletypewriter.

3.1.1 Computer

Figure 3-1 shows the location of the controls and indicators of the PDP-8/L, and Table 3-1 describes their functions.

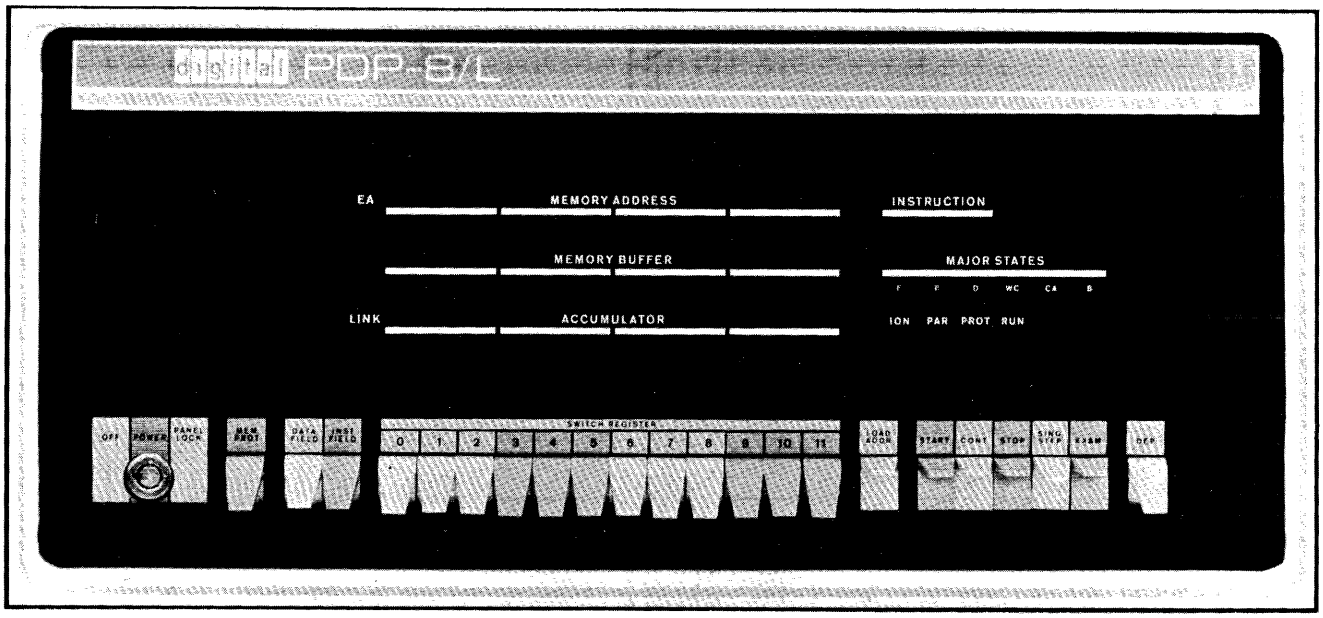


Figure 3-1 PDP-8/L Front Panel

Table 3-1
Computer Controls and Indicators

Control or Indicator	Function
OFF-POWER-PANEL LOCK switch	In OFF position, all power is removed from machine; in POWER position, the machine operates normally, in PANEL LOCK position power is applied and all manual controls except the switch register are disabled.

Table 3-1 (Cont)
Computer Controls and Indicators

Control or Indicator	Function
MEM PROT switch	When up, protects last page of memory (locations 7600_8 to 7777_8) from modification; when off, normal read/write operation prevails.
DATA FIELD switch	Determines which core memory field (if extra 4K has been added) is being addressed for data storage and retrieval.
INST FIELD switch	Determines which core memory field (if extra 4K has been added) is being addressed for instruction storage and retrieval.
EA indicator	Indicates when extended memory is being addressed.
MEMORY ADDRESS indicators	Indicates address of memory that is being operated upon.
MEMORY BUFFER indicators	Indicates content of above address.
ACCUMULATOR indicators	Indicates contents of AC.
LINK indicator	Indicates contents of L.
SWITCH register	Provides a means for manually setting a 12-bit word into the machine.
INSTRUCTION indicators	Indicates contents of Instruction Register.
MAJOR STATES indicators	Indicates which of the six major states (Fetch, Execute, Defer, Word Count, Current Address, or Break) the processor is in.
ION indicator	Indicates that interrupt system is enabled.
PAR indicator	Indicates (when parity option is installed) a parity error.
PROT indicator	Indicates that a memory protect violation has been detected. Machine will halt with this condition.
RUN indicator	Indicates RUN flip-flop is set.
LOAD ADDR switch	Transfers contents of switch register into PC and into MA.
START key	Starts the program by turning off the program interrupt circuits, clearing the AC and L, setting the Fetch state, and starting the central processor timing.
CONT key	This key sets the RUN flip-flop to continue the program in the state and instruction designated by the lighted console indicators, at the address currently specified by the PC if SING STEP key is not on.
STOP key	Causes the RUN flip-flop to be cleared at the end of the instruction in progress at the time the key is pressed.

Table 3-1 (Cont)
Computer Controls and Indicators

Control or Indicator	Function
SING STEP switch	Steps program one cycle-at-time so that operator can observe contents of register in each major state.
EXAM key	This key transfers the content of core memory at the address specified by the content of MA, into MB. The content of the MA is then incremented by one to allow examination of the contents of sequential core memory addresses by repeated operation of the EXAM key. The major state flip-flop register is cleared.
DEP key	This key transfers the content of switch register into MB and core memory at the address specified by the current content of MA. The major state flip-flop is cleared. The contents of PC and MA are then incremented by one to allow storing of information in sequential core memory addresses by repeated operation of the DEP key.

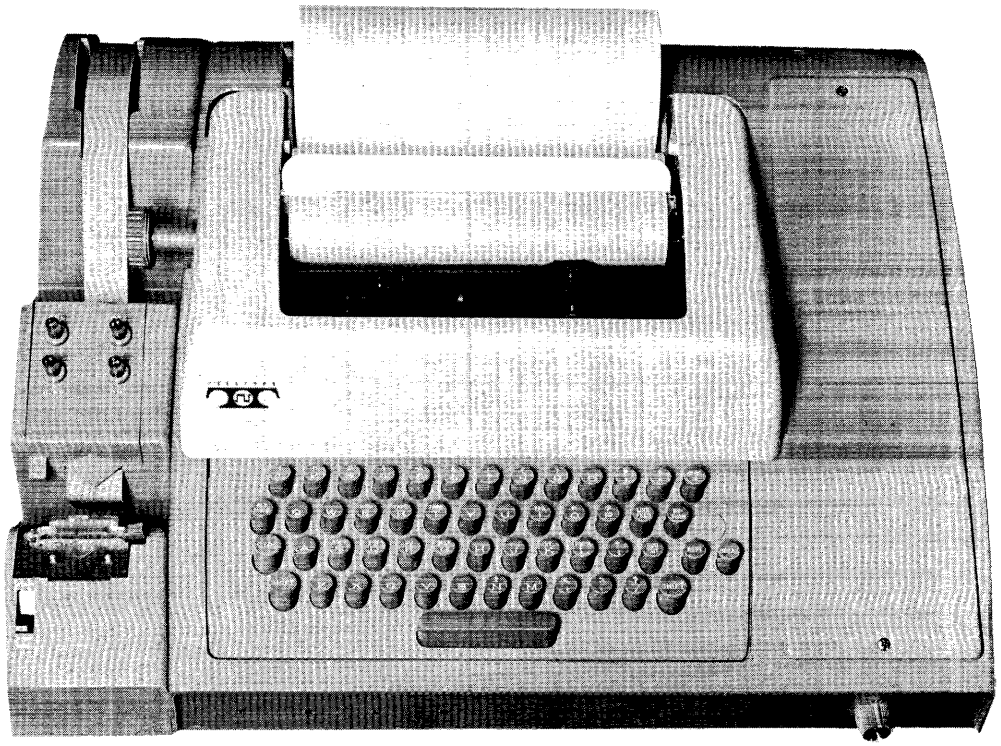


Figure 3-2 Teletype Model ASR33 Console

3.1.2 Teletype

Figure 3-2 shows the location of the ASR33 Teletype-writer controls and indicators, and Table 3-2 is a list of the ASR33 controls and indicators with an explanation of their functions.

3.2 OPERATING PROCEDURES

Many means are available for loading and unloading PDP-8/L information. The means used depend upon the form of the information, time limitations, and the peripheral equipment connected to the computer. The following procedures are basic to any use of the PDP-8/L. Although these procedures are used infrequently as the programming and use of the computer become more sophisticated, they are valuable in preparing the initial programs and learning the function of machine input and output transfers.

All of the procedures described in the following paragraphs require that the OFF-POWER-PANEL LOCK switch be set to POWER.

3.2.1 Manual Loading Procedures

Programs and data can be stored or modified manually by means of the facilities on the operator console. The chief use of the manual data storage facility is to load the Readin Mode Loader program into the computer core memory. The Readin Mode Loader (RIM)

Table 3-2
Teletype Controls and Indicators

Control or Indicator	Function
REL. pushbutton	Disengages the tape in the punch to allow tape removal or tape loading.
B. SP. pushbutton	Backspaces the tape in the punch by one space, allowing manual correction or rubout of the character just punched.
OFF and ON pushbuttons	Control the use of the tape punch with operation of the Teletype keyboard/printer.
START/STOP/FREE switch	Controls use of the tape reader with operation of the Teletype. In the lower FREE position, the reader is disengaged and can be loaded or unloaded. In the center STOP position, the reader mechanism is engaged but de-energized. In the upper START position, the reader is engaged and operated under program control.
Keyboard	Provides a means of printing on paper in use as a typewriter and punching tape when the operator presses the punch ON pushbutton. The keyboard also supplies input data to the computer when the LINE/OFF/LOCAL switch is in the LINE position.
LINE/OFF/LOCAL switch	Controls application of primary power in the Teletype and controls data connection to the processor. In the LINE position, the Teletype is energized and connected as an I/O device of the computer. In the OFF position, the Teletype is de-energized. In the LOCAL position, the Teletype is energized for off-line operation, and signal connections to the processor are broken. Only line use of the Teletype requires that the computer be energized through the POWER switch if primary power for the Teletype is supplied from a source other than the outlet at the back of the computer.

is a program used for loading into the PDP-8/L other programs that have been assembled on perforated tape in RIM format. This program and the RIM tape format are described in the PDP-8/L Users Handbook (see Small Computer Handbook, C-800, 1968 edition) and in Digital Program Library descriptions. The RIM program is also listed in Table 3-3 for rapid reference and can be used as an exercise in manual data storage. To store data manually in the PDP-8/L core memory proceed as follows:

a. Set the bit switches of the SWITCH REGISTER (SR) to correspond with the address bits of the first word to be stored. Press the LOAD ADDR key and observe that the address specified by the SR is held in the MA, MEMORY ADDRESS.

b. Set the SR to correspond with the data or instruction word to be stored at the address just set into the MA. Lift the DEP key and observe that the MB, and hence the core memory, holds the word set by the SR.

Observe that the contents of the MA have been incremented by 1 so that additional data can be stored at sequential addresses by repeated SR setting and DEP key operation.

To check the contents of an address in core memory, set the address into the MA as in step a; then press the EXAM key. The MEMORY BUFFER lights indicate the contents of the address. The contents of the MA are incremented by 1 with the operation of the EXAM key, so that the contents of consecutive addresses can be examined by repeated operation of the EXAM key after the original (or starting) address is loaded. Any address can be modified by repeating steps a and b.

3.2.2 Teletype Loading Procedures

Information can be stored or modified in the computer under program control. For example, having the RIM Loader stored in core memory allows RIM format tapes to be loaded as follows.

a. Set the Teletype LINE/OFF/LOCAL switch to the LINE POSITION.

b. Load the tape in the Teletype reader by setting the START/STOP/FREE switch to the FREE position, releasing the cover guard by means of the latch at the right, loading the tape so that the sprocket wheel teeth engage the feed holes in the tape, closing the cover guard, and setting the switch to

Table 3-3
Readin Mode Loader Program

Address	Octal Content	Tag	Mnemonic	Comments
7756,	6032	BEG,	KCC	/CLEAR AC AND FLAG
7757,	6031		KSF	/SKIP IF FLAG = 1
7760,	5357		JMP .-1	/LOOKING FOR CHARACTER
7761,	6036		KRB	/READ BUFFER
7762,	7106		CLL RTL	
7763,	7006		RTL	/CHANNEL 8 IN AC0
7764,	7510		SPA	/CHECKING FOR LEADER
7765,	5357		JMP BEG+1	/FOUND LEADER
7766,	7006		RTL	/OK, CHANNEL 7 IN LINK
7767,	6031		KSF	
7770,	5367		JMP .-1	
7771,	6034		KRS	/READ, DO NOT CLEAR
7772,	7420		SNL	/CHECKING FOR ADDRESS
7773,	3776		DCA I TEMP	/STORE CONTENTS
7774,	3376		DCA TEMP	/STORE ADDRESS
7775,	5356		JMP BEG	/NEXT WORD
7776,	0	TEMP,	0	/TEMP STORAGE

the STOP position. Load the tape in the back of the reader so that it moves toward the front as it is read. Proper positioning of the tape in the reader finds three channels being sensed to the left of the sprocket wheel and five channels being sensed to the right of the sprocket wheel.

c. Set the MEMORY PROTECT switch to down position.

d. Load the starting address of the RIM Loader program (7756_8) into the MA using the SR and the LOAD ADDR key.

e. Press the computer START key and set the 3-position Teletype reader switch to the START position. The tape is read into memory by program control.

The RIM Loader program loads the Binary Loader (BIN) program as previously described. With the BIN Loader stored in core memory, program tapes assembled in Program Assembly Language (PAL III) binary format can be stored as described in the previous procedure, except that the starting address of the BIN Loader (7777_8) is used in step d. After storing a program in this manner, the computer stops; the AC should contain all 0's if the program is stored properly. If the computer stops with a number other than 0 in the AC, a checksum error has been detected; therefore, the program has been stored incorrectly, and the storage procedure should be repeated. When the program has been stored correctly, initiate it by loading the program starting address (usually designated on the leader of the tape) into the PC and MA using the SR and LOAD ADDR key. Then press the START key.

3.2.3 Off-Line Teletype Procedure

The Teletype can operate separately from the PDP-8/L for typing, punching tape, or duplicating tapes. To use the Teletype in this manner:

a. Assure that the primary Teletype power is on.

b. Set the Teletype LINE/OFF/LOCAL switch to the LOCAL position.

c. Load the punch as follows. Raise the cover and manually feed the tape from the top of the roll into the guide at the back of the punch. Advance the tape through the punch by manually turning the friction wheel; then close the cover.

d. Energize the punch by pressing the ON push-button, and produce about 2 ft of leader. The leader-trailer can be either 200_8 or 377_8 code. To produce the 200_8 code leader, simultaneously press and hold the CTRL and SHIFT keys with the left hand; press and hold the REPT key; press and release the P key. When the required amount of leader has been punched, release the REPT key, then CTRL and SHIFT keys. To produce the 377_8 code leader, simultaneously press and hold both the REPT and RUB OUT keys until a sufficient amount of leader has been punched.

If an incorrect key is struck while punching a tape, the tape can be corrected as follows. If the error is noticed after typing and punching N characters, press the punch B. SP. (backspace) push-button N+1 times and strike the keyboard RUB OUT Key N+1 times. Then continue typing and punching with the character which was in error.

To duplicate and obtain a listing of an existing tape, load the tape to be duplicated in the paper tape reader or set the LOCAL/LINE switch to LOCAL and turn the punch on, and turn the paper tape reader on.

CHAPTER 4
THEORY

This chapter is divided into four sections and covers the theory of operation of the PDP-8/L Computer. Section I contains a discussion of the theory at a block diagram level; Section II contains a discussion in terms of general theory of operation; Sections III and IV cover detailed memory theory and detailed processor theory, respectively.

SECTION I
BLOCK DIAGRAM DISCUSSION

The following paragraphs discuss the major functional elements of the PDP-8/L as shown in the system block diagram (Figure 4-1).

4.1 REGISTERS

4.1.1 Accumulator (AC)

The 12-bit AC serves as an input/output register for programmed information transfers between core memory and peripheral equipment, and as a transfer register through which arithmetic and logic operations are performed.

4.1.2 Link (L)

This 1-bit register extends the arithmetic facilities of the accumulator and serves as the carry register for two's complement arithmetic.

4.1.3 Program Counter (PC)

This 12-bit register contains the address of the core-memory location from which the next instruction will be taken.

4.1.4 Memory Address Register (MA)

This 12-bit register contains the address in core memory that is currently selected for reading or writing. This address is decoded by the memory selection matrix to permit addressing of all 4096 words of the core memory.

4.1.5 Memory Buffer Register (MB)

All data to be written into core memory is loaded first into the 12-bit MB. Through the facilities provided by the major-register gating network, the MB accepts data from any of the major registers in the processor and, during a high-speed data-break transfer, from peripheral devices. Its only output capability, other than its direct access to core memory, is through the processor interface to optional peripheral equipment.

4.1.6 Sense Register (Sense)

All data read from core memory is strobed first into this 12-bit register. It accepts data only from the core memory and transfers data directly to the Instruction Register (IR) and, through the major register gating network, to any other register in the processor.

4.1.7 Instruction Register (IR)

This 3-bit register contains the operation code of the instruction currently being performed by the computer. The three most-significant bits of the current instruction load into the IR from Sense during a Fetch cycle. The contents of the IR are decoded to produce discrete levels for each of the eight basic instructions.

4.1.8 Switch Register (SR)

The 12-bit SR performs a dual function in that it permits the manual loading of either a discrete core-memory address into the PC or a 12-bit data or control word into core memory. The SR is loaded by 12 toggle switches located on the operator's console. Actuation of either the LOAD ADDR or DEP keys then causes the stored information to be loaded into the MA or MB, respectively.

4.2 MAJOR REGISTER GATING NETWORK

All internal data transfers occurring in the PDP-8/L are implemented through the major-register gating

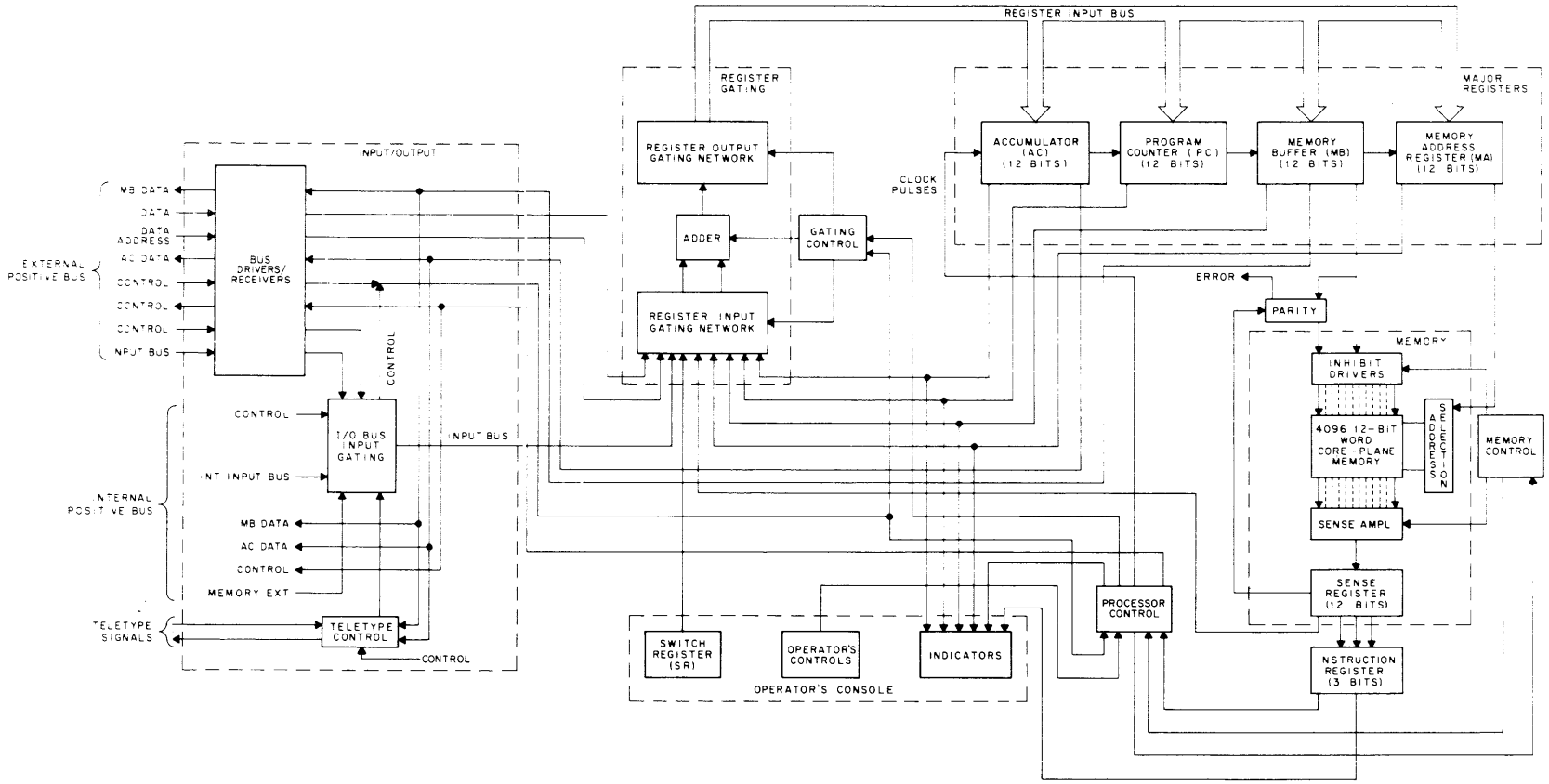


Figure 4-1 System Block Diagram

network. The network contains a separate gate structure and a common register input bus for each of the 12 bits. Transfers between registers and into and out of memory occur through the network. Data and address information received through the I/O interface also pass through this network.

4.3 TIMING AND CONTROL ELEMENTS

4.3.1 Timing Elements

The processor and memory control circuits in the standard PDP-8/L use fixed and variable delay lines in place of timing clocks. Interleaving of fixed delay sequences provides asynchronous control between the processor and the memory. The overall cycle time of approximately 1.6 μs is determined by the memory timing. For applications involving real time, the KW8I Real Time Clock option is added to the system.

4.3.2 Control Elements

Circuits in the PDP-8/L control program advance and instruction skipping. These circuits, which operate in response to conditions established in either the processor or peripheral equipment, control the flow of information between registers. In addition, they initiate program interrupt operations during which subroutines enable the servicing of peripheral equipment.

4.4. INPUT/OUTPUT

The PDP-8/L operates with two basic types of input/output (I/O) equipment. Peripherals which communicate with the processor through the external positive programmed transfer I/O bus, and those which use the external positive data break facility. In addition the high-speed perforated paper tape reader and punch peripherals operate directly from control modules located within the PDP-8/L processor assembly.

Typical of optionally available programmed transfer I/O bus peripherals are:

VC8/L	Oscilloscope Display Control
KV8/L	Storage Tube Display and Control
CR8/L	Low-Speed Card Reader
VP8/L	Incremental Plotter
PR8/L	High-Speed Tape Reader
PP8/L	High-Speed Tape Punch

The data-break peripherals, also optionally available, are represented by mass memory devices, such as magnetic tape, magnetic drum, and disk file systems.

A Teletype ASR33 Automatic Send-Receive Set is provided as standard equipment with the PDP-8/L. In addition to a manual keyboard and hard-copy printout facilities, the ASR33 contains an 8-level paper-tape punch and a paper-tape reader, all of which are interfaced with the processor through the Teletype control logic and internal positive bus.

4.5 MEMORY

The standard memory supplied with the PDP-8/L is a random-access, coincident current, magnetic-core memory with a storage capacity of 4096 12-bit words. The core planes and diode matrices that make up the core array are mounted on printed-circuit cards. These cards plug directly into the PDP-8/L logic rack receptacles with the sense and inhibit inputs attached on connector cards. An additional 4K of core memory may be optionally added externally to the PDP-8/L frame.

The major elements of the core memory are described in the following paragraphs.

4.5.1 Core Array

The ferrite-core array consists of 12 (64 x 64) core planes. This provides a total of 4096 12-bit words of data and program storage. A thirteenth core plane is optionally available to permit a parity bit for each word in memory.

4.5.2 Memory Control

Memory control circuits determine the sequence of operations of the complete read/write memory cycle, starting and stopping each function as required.

4.5.3 Address Selection

The Memory Address register (MA) contains the 12-bit address of the currently selected core-memory locat-

ion. This address is decoded through the selection switches and the diode matrix to enable passage of read/write currents through specific X and Y drive lines of the memory. The coincidence of these currents selects the specific 12-bit core-memory location desired.

4.5.4 Inhibit Drivers

The PDP-8/L memory is so configured that, unless prohibited, all bit locations of the addressed memory cell would be switched to a logical 1 during the write portion of the memory cycle. Inhibit drivers, therefore, are used to ensure that the logic 0 levels stored in the MB will be retained in the corresponding bit locations of the addressed memory cell.

4.5.5 Sense Amplifiers

During the read portion of the memory cycle, sense amplifiers detect analog signals induced in the sense windings of the core array. These signals are amplified and used to set corresponding bits of the Sense Register.

SECTION II
GENERAL THEORY

The following paragraphs discuss the major functional elements of the PDP-8/L in terms of their operational dynamics. These dynamics will be discussed in greater detail in Sections III and IV.

4.6 TIME STATES/TIME PULSES

Each computer cycle consists of four basic time divisions, T1, T2, T3, and T4, as denoted on the system flow diagrams. Each time division consists of a time state (TS) and its associated time pulse (TP). The time states each extend throughout their particular time division (TS1, TS2, TS3, TS4) and end with a time pulse (TP1, TP2, TP3, TP4).

In general, the time states generate enabling levels associated with the register outputs. Time pulses are used to strobe data into registers.

4.7 MAJOR STATES

The PDP-8/L contains six major-state flip-flops. These are: Fetch (F), Defer (D), Execute (E), Word Count (WC), Current Address (CA), and Break (B). The outputs of these flip-flops generate enabling levels used within the control elements of the processor to implement particular machine functions.

The first three major states (F, D, and E) are sufficient to perform most machine functions in the areas of logical operations, memory read/write operations, and data transfers through the I/O bus. The last three major states (WC, CA, and B) are used only for high-speed data transfers through the optional KD8L Data-Break facility.

The processor determines, near the end of each computer cycle, which major state will be needed for the activities to be performed in the next computer cycle. At the very end of the cycle (TP4) the new major-state will be entered by the setting of that particular flip-flop.

4.8 INTERNAL DATA FLOW

The simplified system block diagram shown in Figure 4-1 depicts the flow of data through the major elements of the PDP-8/L. Note that all data transfers into the four major registers (AC, PC, MB, and MA) occur through a register gating network and a common register bus. The outputs of these four registers, plus the Sense and SR, and the data input from the external interface are all connected to the input gates of the major-register gating network.

This permits incoming data to be strobed into any desired major register, or the contents of any register to be complemented, incremented, or transferred into any other major register. The complementing function is implemented by transferring the 0 output of the desired register through the gating network and back into the same register. The incrementing function is performed by transferring the 1 output of the register through the gating network while inserting a carry into the low-order bit of the word. The data is then transferred back into the desired register.

4.9 INSTRUCTIONS

Instruction words are of two types: memory reference and augmented. Memory reference instructions store or retrieve data from core memory, while augmented instructions do not. All instructions utilize bits 0 through 2 to specify the operation code. Operation codes of 0_8 through 5_8 specify memory reference instructions, and codes of 6_8 and 7_8 specify augmented instructions. Memory reference instruction execu-

tion times are multiples of the 1.6 μs memory cycle. Indirect addressing increases the execution time of a memory reference instruction by 1.6 μs. The augmented instructions, input-output transfer and operate, are performed in 4.25 and 1.6 μs, respectively. (All computer times are ±12%.)

4.9.1 Memory Reference Instructions

Since the PDP-8/L system contains a 4096-word core memory, 12 bits are required to address all locations. To simplify addressing, the core memory is divided into blocks, or pages, of 128 words (200_8 addresses). Pages are numbered 0_8 through 37_8, each field of 4096-words of core memory uses 32 pages. The seven address bits (bits 5 through 11) of a memory reference instruction can address any location in the page on which the current instruction is located by placing a 1 in bit 4 of the instruction. By placing a 0 in bit 4 of the instruction, any location in page 0 can be addressed directly from any page of core memory. All other core memory locations can be addressed indirectly by placing a 1 in bit 3 and placing a 7-bit effective address in bits 5 through 11 of the instruction to specify the location in the current page or page 0 which contains the full 12-bit absolute address of the operand.

Word format of memory reference instructions is shown in Figure 4-2 and the instructions perform as follows:

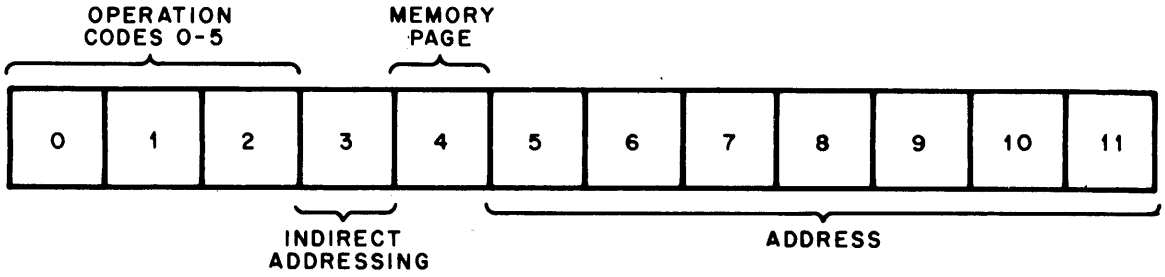


Figure 4-2
Memory Reference Instruction Bit Reference

Logical AND (AND Y)

Octal Code: 0
Indicators: IR0, FETCH, EXECUTE
Execution Time: 3.2 μs with direct addressing, 4.8 μs with indirect addressing.
Operation: The AND operation is performed between the content of memory location Y and the content of the AC. The result is left in the AC, the original content of the AC is lost, and the content of Y is restored. Corresponding bits of the AC and Y are operated upon independently. This instruction, often called extract or mask, can be considered as a bit-by-bit multiplication. Example:

Original ACj	Yj	Final ACj
0	0	0
0	1	0
1	0	0
1	1	1

Symbol: $ACj \wedge Yj = > ACj$

Two's Complement Add (TAD Y)

Octal Code: 1
Indicators: IR1, FETCH, EXECUTE
Execution Time: 3.2 μs with direct addressing, 4.8 μs with indirect addressing.
Operation: The content of memory location Y is added to the content of the AC in two's complement arithmetic. The result of this addition is held in the AC, the original content of the AC is lost, and the content of Y is restored. If there is a carry from AC0, the link is complemented. This feature is useful in multiple precision arithmetic.

Symbol: AC0 - 11 + Y0 - 11 = >AC0 - 11

Increment and Skip If Zero (ISZ Y)

Octal Code: 2
Indicators: IR2, FETCH, EXECUTE
Execution Time: 3.2 μs with direct addressing, 4.8 μs with indirect addressing.
Operation: The content of memory location Y is incremented by one in two's complement arithmetic. If the resultant content of Y equals zero, the content of the PC is incremented by one and the next instruction is skipped. If the resultant content of Y does not equal zero, the program proceeds to the next instruction. The incremented content of Y is restored to memory. The content of the AC is not affected by this instruction.

Symbol: Y + 1 = >Y
If resultant Y0 - 11 = 0, then PC + 1 = >PC

Deposit and Clear AC (DCA Y)

Octal Code: 3
Indicators: IR3, FETCH, EXECUTE
Execution Time: 3.2 μs with direct addressing, 4.8 μs with indirect addressing.
Operation: The content of the AC is deposited in core memory at address Y and the AC is cleared. The previous content of memory location Y is lost.

Symbol: AC = >Y
then 0 = >AC

Jump to Subroutine (JMS Y)

Octal Code: 4
Indicators: IR4, FETCH, EXECUTE
Execution Time: 3.2 μs with direct addressing, 4.8 μs with indirect addressing.
Operation: The content of the PC is deposited in core memory location Y and the next instruction is taken from core memory location Y + 1. The content of the AC is not affected by this instruction.

Symbol: PC + 1 = >Y
Y + 1 = >PC

Jump to Y (JMP Y)

Octal Code: 5
Indicators: IR5, FETCH
Execution Time: 1.6 μs with direct addressing, 3.2 μs with indirect addressing.
Operation: Address Y is set into the PC so that the next instruction is taken from core memory address Y. The original content of the PC is lost. The content of the AC is not affected this instruction.

Symbol: Y = >PC

4.9.2 Augmented Instructions

There are two augmented instructions which do not reference core memory. They are the input-output transfer, which has an operation code of 6, and the operate which has an operation code of 7. Bits 3 through 11 within these instructions function as an extension of the operation code and can be microprogrammed to perform several operations within one instruction. Augmented instructions are one-cycle (Fetch) instructions that initiate various operations as a function of bit microprogramming.

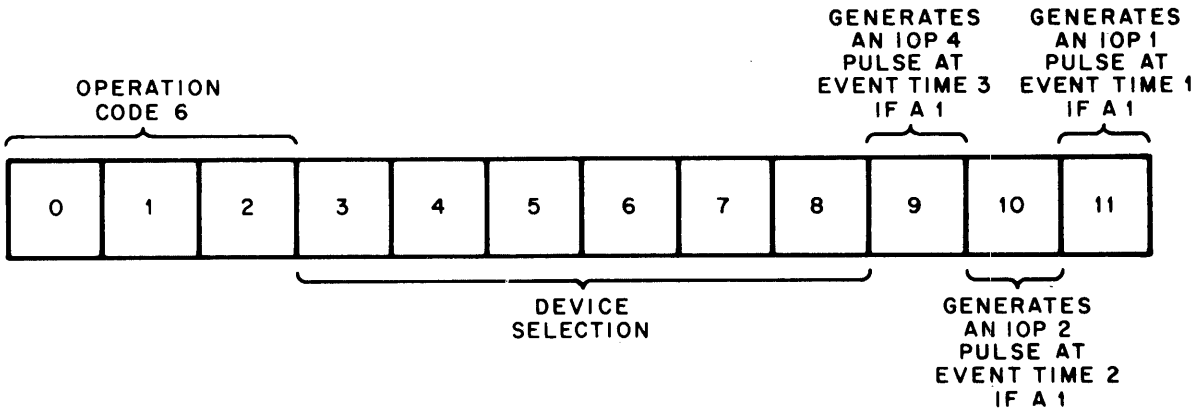


Figure 4-3
IOT Instruction Bit Assignments

4.9.2.1 Input/Output Transfer Instruction - Microinstructions of the input-output transfer (IOT) group initiate operation of peripheral equipment and effect information transfers between the processor and an I/O device. Specifically, upon recognition of the operation code 6 as an IOT instruction, the computer enters a 4.25 μs expanded computer Fetch cycle by setting a PAUSE flip-flop and enabling the IOP generator to produce IOP 1, IOP 2 and IOP 4 pulses as a function of the three least significant bits of the instruction (bits 9 through 11). These pulses occur at 1 μs intervals designated as event times 1, 2 and 3 as follows.

Instruction Bit (MB)	IOP Pulses	IOT Pulse	Event Time
11	IOP 1	IOT 1	1
10	IOP 2	IOT 2	2
9	IOP 4	IOT 4	3

The IOP pulses are gated in the device selector of the selected equipment to produce IOT pulses that enact a data transfer or initiate a control operation. Program selection of an equipment is accomplished by bits 3 through 8 of the IOT instruction. These bits form a 6-bit code that enables the device selector of a given device. The format of the IOT instruction is shown in Figure 4-3.

4.9.2.2 Operate Instruction - With operate instructions, the programmer can consider logical sequences occurring during one computer Fetch cycle. These sequences provide a logical method of forming microinstructions.

The operate instruction consists of two groups of microinstructions. Group 1 (OPR 1) is principally for clear, complement, rotate, and increment operations and is designated by the presence of a 0 in bit 3. Group 2 (OPR 2) is used principally in checking the content of the accumulator and link and continuing to, or skipping, the next instruction based on the check. A 1 in bit 3 designates an OPR 2 microinstruction.

4.9.2.2.1 Group 1 Microinstruction - The Group 1 operate microinstruction format is shown in Figure 4-4 and the microinstructions are explained in the succeeding paragraphs. Any logical combination of bits within this group can be combined into one microinstruction. For example, it is possible to assign ones to bits 5, 6, and 11; although it is not logical to assign ones to bits 8 and 9 simultaneously since they specify conflicting operations.

No Operation (NOP)

Octal Code: 7000
Sequence: None
Indicators: IR7, FETCH
Execution Time: 1.6 μs
Operation: This command causes a 1-cycle delay in the program and then the next sequential instruction is initiated. This command is used to add execution time to a program, such as to synchronize subroutine or loop timing with peripheral equipment timing.

Symbol: None

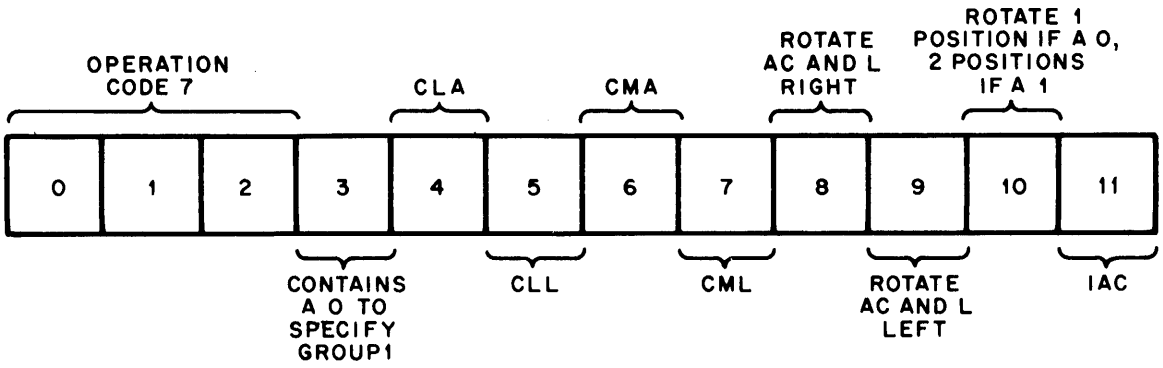


Figure 4-4
Group 1 Operate Instruction Bit Assignments

Increment Accumulator (IAC)

Octal Code: 7001
Sequence: 3
Indicators: IR7, FETCH
Execution Time: 1.6 μs
Operation: The content of the AC is incremented by one.

Symbol: AC + 1 = >AC

Rotate Accumulator Left (RAL)

Octal Code: 7004
Sequence: 4
Indicators: IR7, FETCH
Execution Time: 1.6 μs
Operation: The content of the AC is rotated one binary position to the left with the content of the link. The content of bits AC1 - 11 are shifted to the next more significant bit, the content of AC0 is shifted into the L, and the content of the L is shifted into AC11.

Symbol: ACj = > ACj - 1
AC0 = >L
L = >AC11

Rotate Two Left (RTL)

Octal Code: 7006
Sequence: 4
Indicators: IR7, FETCH
Execution Time: 1.6 μs
Operation: The content of the AC is rotated two binary positions to the left with the content of the link. This instruction is logically equal to two successive RAL operations.

Symbol: ACj = > ACj - 2
AC1 = >L
AC0 = >AC11
L = >AC10

Rotate Accumulator Right (RAR)

Octal Code: 7010
Sequence: 4
Indicators: IR7, FETCH
Execution Time: 1.6 μs
Operation: The content of the AC is rotated one binary position to the right with the content of the link. The content of bits AC0 - 10 are shifted to the next less significant bit, the content of AC11 is shifted into the L, and the content of the L is shifted into AC0.

Symbol: ACj = >ACj + 1
AC11 = >L
L = > AC0

Rotate Two Right (RTR)

Octal Code: 7012
Sequence: 4
Indicators: IR7, FETCH
Execution Time: 1.6 μs
Operation: The content of the AC is rotated one binary position to the right with the content of the link. This instruction is logically equal to two successive RAR operations.

Symbol: ACj = >ACj + 2
AC10 = L
AC11 = AC0
L = >AC1

Complement Link (CML)

Octal Code: 7020
Sequence: 2
Indicators: IR7, FETCH
Execution Time: 1.6 μs
Operation: The content of the L is complemented.

Symbol: $\overline{L}$ = >L

Complement Accumulator (CMA)

Octal Code: 7040
Sequence: 2
Indicators: IR7, FETCH
Execution Time: 1.6 μs
Operation: The content of the AC is set to the one's complement of the current content of the AC. The content of each bit of the AC is complemented individually.

Symbol: $\overline{ACj}$ = >ACj

Complement and Increment Accumulator (CIA)

Octal Code: 7041
Sequence: 2,3
Indicators: IR7, FETCH
Execution Time: 1.6 μs
Operation: The content of the AC is converted from a binary value to its equivalent two's complement number. This conversion is accomplished by combining the CMA and IAC commands, thus the content of the AC is complemented during sequence 2 and is incremented by one during sequence 3.

Symbol: $\overline{ACj}$ = >ACj,
then AC + 1 = >AC

Clear Link (CLL)

Octal Code: 7100
Sequence: 1
Indicators: IR7, FETCH
Execution Time: 1.6 μs
Operation: The content of the L is cleared to contain a 0.

Symbol: 0 = >L

Set Link (STL)

Octal Code: 7120
Sequence: 1,2
Indicators: IR7, FETCH
Execution Time: 1.6 μs
Operation: The L is set to contain a binary 1. This instruction is logically equal to combining the CLL and CML commands.

Symbol: 1 = >L

Clear Accumulator (CLA)

Octal Code: 7200
Sequence: 1
Indicators: IR7, FETCH
Execution Time: 1.6 μs
Operation: The content of each bit of the AC is cleared to contain a binary 0.

Symbol: 0 = >AC

Set Accumulator (STA)

Octal Code: 7240
Sequence: 1,2
Indicators: IR7, FETCH
Execution Time: 1.6 μs
Operation: Each bit of the AC is set to contain a binary 1. This operation is logically equal to combining the CLA and CMA commands.

Symbol: 1 = >ACj

4.9.2.2.2 Group 2 Microinstructions - The Group 2 operate microinstruction format is shown in Figure 4-5 and the primary microinstructions are explained in the following paragraphs. Any logical combination of bits within this group can be composed into one microinstruction.

If skips are combined in a single instruction, the inclusive OR of the conditions determines the skip when bit 8 is a 0; and the AND of the inverse of the conditions determines the skip when bit 8 is a 1. For example, if ones are designated in bits 6 and 7 (SZA and SNL), the next instruction is skipped if either the content of the AC = 0, or the content of L = 1. If ones are contained in bits 5, 7, and 8, the next instruction is skipped if the AC contains a positive number and the L contains a 0.

Halt (HLT)

Octal Code: 7402
Sequence: 3
Indicators: IR7, RUN off
Execution Time: 1.6 μs
Operation: Clears the RUN flip-flop at Sequence 3, so that the program stops at the conclusion of the current machine cycle. This command can be combined with others in the OPR 2 group that are executed during either sequence 1, or 2, and so are performed before the program stops.

Symbol: 0 = >RUN

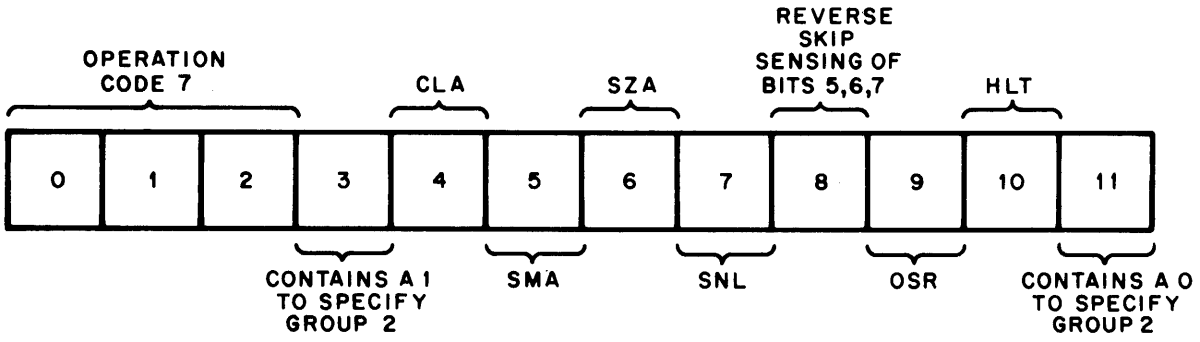


Figure 4-5
Group 2 Operate Instruction Bit Assignments

OR With Switch Register (OSR)

Octal Code: 7404
Sequence: 3
Indicators: IR7, FETCH
Execution Time: 1.6 μs
Operation: The inclusive OR operation is performed between the content of the AC and the content of the SR. The result is left in the AC, the original content of the AC is lost, and the content of the SR is unaffected by this command. When combined with the CLA command, the OSR performs a transfer of the content of the SR into the AC.
Symbol: ACj V SRj = >ACj

Skip, Unconditional (SKP)

Octal Code: 7410
Sequence: 1
Indicators: IR7, FETCH
Execution Time: 1.6 μs
Operation: The content of the PC is incremented by one so that the next sequential instruction is skipped.

Symbol: PC + 1 = >PC

Skip on Non-Zero Link (SNL)

Octal Code: 7420
Sequence: 1
Indicators: IR7, FETCH
Execution Time: 1.6 μs
Operation: The content of the L is sampled, and if it contains a 1 the content of the PC is incremented by one so that the next sequential instruction is skipped. If the L contains a 0, no operation occurs and the next sequential instruction is initiated.

Symbol: If L = 1, then PC + 1 = >PC

Skip on Zero Link (SZL)

Octal Code: 7430
Sequence: 1
Indicators: IR7, FETCH
Execution Time: 1.6 μs
Operation: The content of the L is sampled, and if it contains a 0 the content of the PC is incremented by one so that the next sequential instruction is skipped. If the L contains a 1, no operation occurs and the next sequential instruction is initiated.

Symbol: If L = 0, then PC + 1 = >PC

Skip on Zero Accumulator (SZA)

Octal Code: 7440
Sequence: 1
Indicators: IR7, FETCH
Execution Time: 1.6 μs
Operation: The content of each bit of the AC is sampled, and if any bit contains a 0 the content of the PC is incremented by one so that the next sequential instruction is skipped. If all bits of the AC contain a 0, no operation occurs and the next sequential instruction is initiated.

Symbol: If AC0 - 11 = 0, then PC + 1 = >PC

Skip on Non-Zero Accumulator (SNA)

Octal Code: 7450
Sequence: 1
Indicators: IR7, FETCH
Execution Time: 1.6 μs
Operation: The content of each bit of the AC is sampled, and if any bit contains a 1 the content of the PC is incremented by one so that the next sequential instruction is skipped. If all bits of the AC contain a 0, no operation occurs and the next sequential instruction is initiated.

Symbol: If AC0 - 11 $\neq$ 0, then PC + 1 = >PC

Skip on Minus Accumulator (SMA)

Octal Code: 7500
Sequence: 1
Indicators: IR7, FETCH
Execution Time: 1.6 μs
Operation: The content of the most significant bit of the AC is sampled, and if it contains a 1, indicating the AC contains a negative number, the content of the PC is incremented by one so that the next sequential instruction is skipped. If the AC contains a positive number no operation occurs and program control advances to the next sequential instruction.

Symbol: If AC0 = 1, then PC + 1 = >PC

Skip on Positive Accumulator (SPA)

Octal Code: 7510
Sequence: 1
Indicators: IR7, FETCH
Execution Time: 1.6 μs
Operation: The content of the most significant bit of the AC is sampled, and if it contains a 0 indicat-

ing a positive number (or zero), the content of the PC is incremented by one so that the next sequential instruction is skipped. If the AC contains a negative number, no operation occurs and the program control advances to the next sequential instruction.

Symbol: If AC0 = 0, then PC + 1 = >PC

Clear Accumulator (CLA)

Octal Code: 7600
Sequence: 2
Indicators: IR7, FETCH
Execution Time: 1.6 μs
Operation: Each bit of the AC is cleared to contain a binary 0.

Symbol: 0 = >AC

4.10 PROGRAM INTERRUPT

Some of the I/O devices used with the PDP-8/L require several instructions to complete a data transfer or perform a specified operation. Others are so slow in operation, relative to the computer, that it would consume a prohibitive amount of time to have the computer wait in a skip loop for their operation to be completed. These devices, therefore, employ the program interrupt facility.

When the program enables the program interrupt facility, the computer senses interrupt requests from peripheral devices. The interrupt may also be initiated in response to a programmed IOT instruction.

An interrupt is allowed to occur only on completion of the instruction currently in process and takes affect at the beginning of the following Fetch cycle.

A program interrupt is similar in effect to a JMS to memory address 0000. The content of PC is saved in location 0000 and the next instruction taken from location 0001$_8$. The instruction stored at this location is usually a JMP to a peripheral servicing subroutine.

After identifying the interrupting device and servicing it, the processor performs a JMP I 0000 (jump to the address specified by the content of location 0000) to return to the point at which the program was interrupted.

4.10.1 Instructions

The two instructions associated with the program interrupt synchronization element are IOT microinstructions. These instructions are:

Interrupt Turn On (ION)

Octal Code: 6001
Event Time: Not applicable
Indicators: IR6, FETCH, ION
Execution Time: 4.25 μs
Operation: This command enables the computer to respond to a program interrupt request. If the interrupt is disabled when this instruction is given, the computer executes the next instruction, then enables the interrupt. The additional instruction allows exit from the interrupt subroutine before allowing another interrupt to occur. This instruction has no effect upon the condition of the interrupt circuits if it is given when the interrupt is enabled.

Symbol: 1 = >INT. ENABLE

Interrupt Turn Off (IOF)

Octal Code: 6002
Event Time: Not applicable
Indicators: IR6, FETCH
Execution Time: 4.25 μs
Operation: This command disables the program interrupt synchronization element to prevent interruption of the current program.

Symbol: 0 = >INT.ENABLE, INT.DELAY

SECTION III
DETAILED MEMORY THEORY

The following paragraphs discuss memory theory at a detailed level.

4.11 OVERALL MEMORY THEORY

The basic PDP-8/L contains a single 4096-word, 12-bit core memory which performs all normal functions of data storage and retrieval. All necessary control elements for the memory are contained within the basic PDP-8/L.

The memory capacity of the computer can be increased, to a maximum of 8192 words with or without

parity. If the parity option is selected, the parity bit is carried as the thirteenth bit in each word. Use of the extended memory option, however, necessitates the addition of the type MC8/L Memory Extension Control. The extended memory is located external to the processor.

Figure 4-6 is a block diagram showing the interrelationship of the major elements of the PDP-8/L memory and its control elements.

4.12 MEMORY OPERATION

PDP-8/L memory operation involves five major functions: address selection, read, sense, inhibit, write. The memory control provides the timing and initiation of the read, sense, inhibit and write functions. Address selection is performed by the contents of the MA register, applied through the address selection X and Y Diode Selection Matrices.

4.12.1 Memory Control

The memory control consists of series-connected delay lines with associated logic gates and control flip-flops (Drawing BS-8L-0-13). An initiating signal, MEM START, from the central processor, cycles through the delay lines alternately setting and clearing the various control flip-flops, and returns to the processor as the MEM DONE signal. The timing for this cycle is fixed by prewired taps on the delay lines. The control flip-flops enable the read/write and inhibit currents, and control the memory cycle.

Figure 4-7 illustrates the waveshapes of the memory control signals. The transition and duration times are approximate due to the inexact delay through the pusle amplifiers (approximately 50 ns), and gates (approximately 20 ns). For explanation purposes, it is assumed that the processor POWER CLEAR signal has been generated, and the START key actuated producing MEM START.

The MEM START signal to the memory control circuitry is gated by the field selection signal $\overline{EA}$. Initially, MEM START sets the MEM ENABLE flip-flop when $\overline{EA}$ specifies basic memory field operation. The buffered MEM ENABLE output enables the contents of the sense register to the major register bus; the X- and Y-axis selection current drivers, and the inhibit drivers.

The memory timing cycle consists of two transitions through the same delay lines. After MEM START triggers the first delay, (Drawing BS-8L-0-13), MEM BEGIN is produced, setting the READ and LOCK flip-flops. The READ flip-flop enables the read/write switches to provide the read currents to the memory stack. The LOCK flip-flop prevents initiation of another memory cycle until the present cycle is completed. Another tap of the delay line provides the pulse input to the variable delay enabled by the READ flip-flop. The pulse output of the variable delay generates STROBE FIELD 0 and $\overline{STROBE}$. The variable delay is adjusted so these signals occur toward the end of the read cycle. STROBE FIELD 0 gates the sense amplifier outputs (Drawing BS-8L-0-14) into the Sense register. $\overline{STROBE}$ returns to the central processor to clear the MEM IDLE and PAUSE flip-flops (Drawing BS-8L-0-2) and continue the processor timing cycle.

Approximately 400 ns after Read is initiated, a CYC DONE pulse is produced. CYC DONE terminates the Read cycle, sets the CYCLE flip-flop, and initiates the second transition of the memory cycle through the same delay lines. The setting of the CYCLE flip-flop enables CYC DONE (delayed) to set the INHIBIT flip-flop. The inhibit output gates the data from the MB to the inhibit drivers (Drawing BS-8L-0-14). The Write portion of the memory cycle is initiated approximately 50 ns after the INHIBIT flip-flop is set. When set, the WRITE flip-flop enables the read/write switches to provide the write currents to the memory stack, and clear the LOCK flip-flop. Another delayed CYC DONE pulse clears the CYCLE flip-flop during the write portion of the memory cycle.

A second CYC DONE pulse, generated approximately 400 ns after the initiation of the inhibit operation, terminates the inhibit and write operations. In addition, this CYC DONE pulse combines with LOCK (0) to generate $\overline{MEM\ DONE}$, indicating the end of the memory timing cycle. $\overline{MEM\ DONE}$ sets the MEM IDLE flip-flop (Drawing BS-8L-0-2) in the processor control, reinitiating another processor/memory cycle.

4.12.2 Read/Write

The ferrite-core memory consists of 12 planes (13 if the parity option is selected), each containing 4096 ferrite cores arranged in a 64 x 64 core array. Each core assumes a stable magnetic state corresponding either to a binary 1 or a binary 0.

Selection and switching of the cores is provided by four windings traversing each core in the memory in a standard 3D selection technique. An X-axis read/

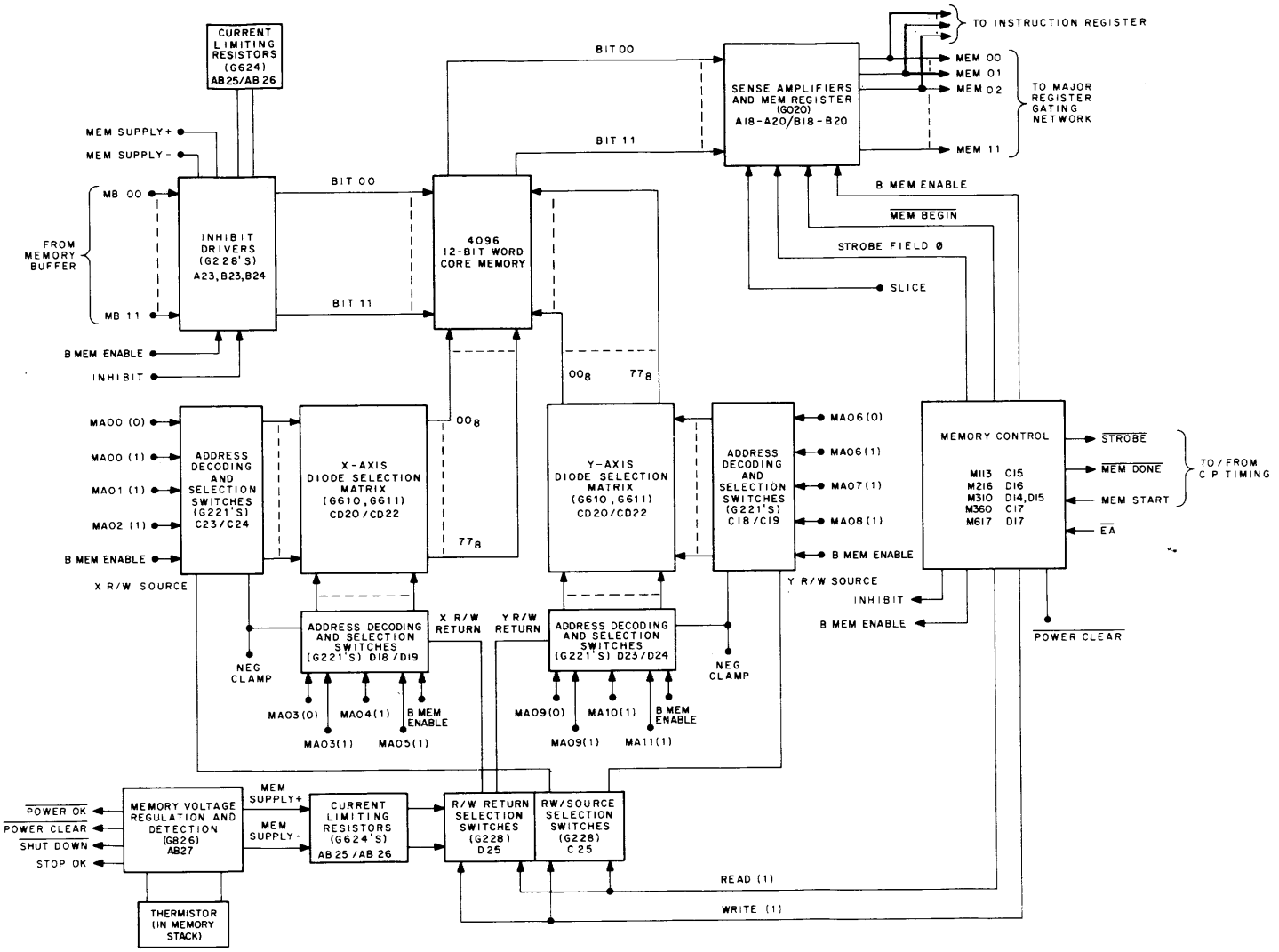
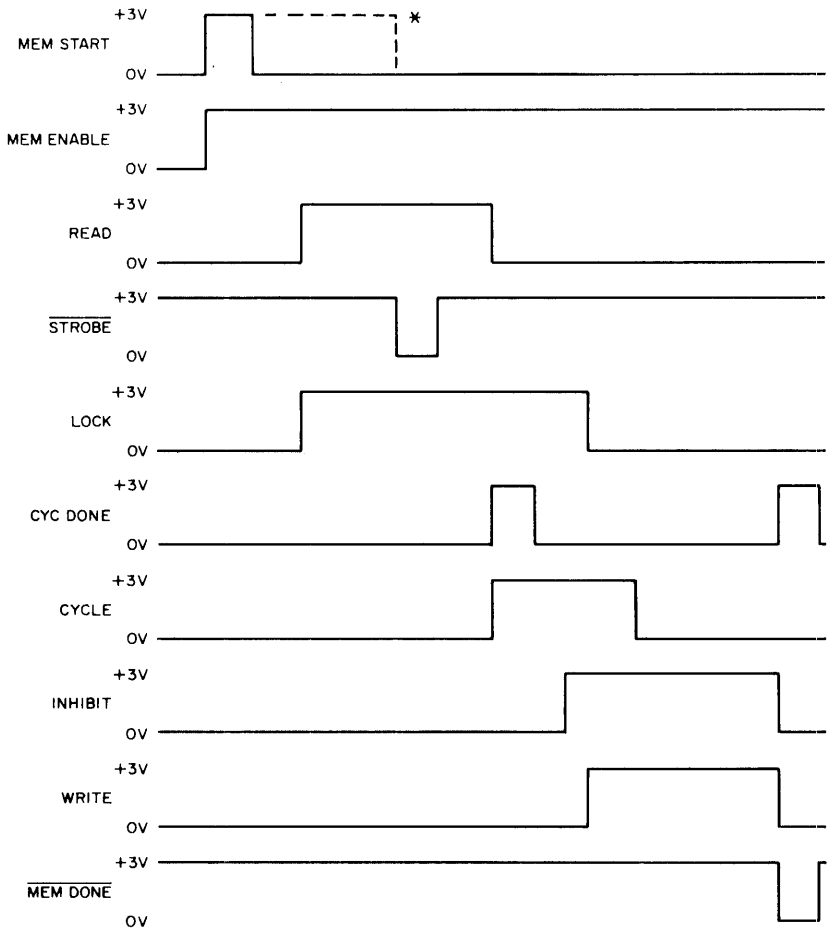


Figure 4-6 Memory System, Block Diagram



* The MEM START pulse width depends on the path of its origin i.e., approximately 150 nsec if generated by a manual function like actuating Dep key, and approximately 500 nsec if generated from a previous program instruction.

Figure 4-7 Memory Timing Diagram

write winding passes through all cores in each of 64 horizontal rows; a Y-axis read/write winding passes through all cores in each of 64 vertical rows; and a sense and an inhibit winding pass through all cores of each of 12 (or 13) planes. Through the use of selection circuits controlling the inputs of the X and Y read/write windings, any one of the 4096 12-bit word locations can be addressed for writing data into, or reading data out of memory.

The level of the read, write, and inhibit currents passing through these windings is such that no single winding produces a magnetic field strong enough to cause a core to change its magnetic state. This current level is known as the half-select value. Only the reinforcing magnetic field caused by the coincident current of both an X and a Y read/write winding can cause the core located at the point of coincidence to change state. It is this principle that allows the relatively simple winding arrangement to select one and only one memory word out of a possible 4096 in each array.

Figure 4-8 shows a simple 4 x 4 core array. The winding scheme shown on this array is identical to that used in the planes of the PDP-8/L memory.

A half-select current passing through the X2 winding from right to left (write direction) produces a magnet-

ic field that tends to change all the cores in that horizontal row from the 0 to the 1 state. The flux produced by this current is, however, insufficient to complete the state transition in any core. Simultaneously, passing a half-select current through the Y3 winding from top to bottom (write direction) tends to produce the same effect on all cores in that particular vertical row. Note, however, that both currents pass through one core, located at the intersection of the X2 and Y3 windings. This then becomes the selected core.

The X and Y windings are so configured that, when half-select value write currents are passed through each, their resultant magnetic fields add in the core at their point of intersection. Their combined (full-select) current then ensures that the selected core is left in the 1 state.

In the PDP-8/L core memory the X2 windings of all 12 planes are connected in series, as are the Y3 windings. When X2Y3 half-select write currents flow, therefore, the X2Y3 core on each plane changes to, or remains in, the 1 state. This makes each of these cores equivalent to one bit of a 12-bit storage cell.

It should be noted that passing half-select value write currents through a particular pair of X and Y windings produces the 1 state in all 12-bit positions of the selected core-memory storage cell. To store usable information, however, it is also necessary to write the 0 state during the write cycle in any or all bit locations of the selected storage cell. This is performed by the inhibit windings, and the prior occurrence of a read cycle which put all the cores in the "0" state.

Each inhibit winding, shown as a broken line on Figure 4-8, passes through all cores on a particular plane. Unlike the X and Y windings, in which the read and write currents flow in opposite directions, the half-select value current in the inhibit windings

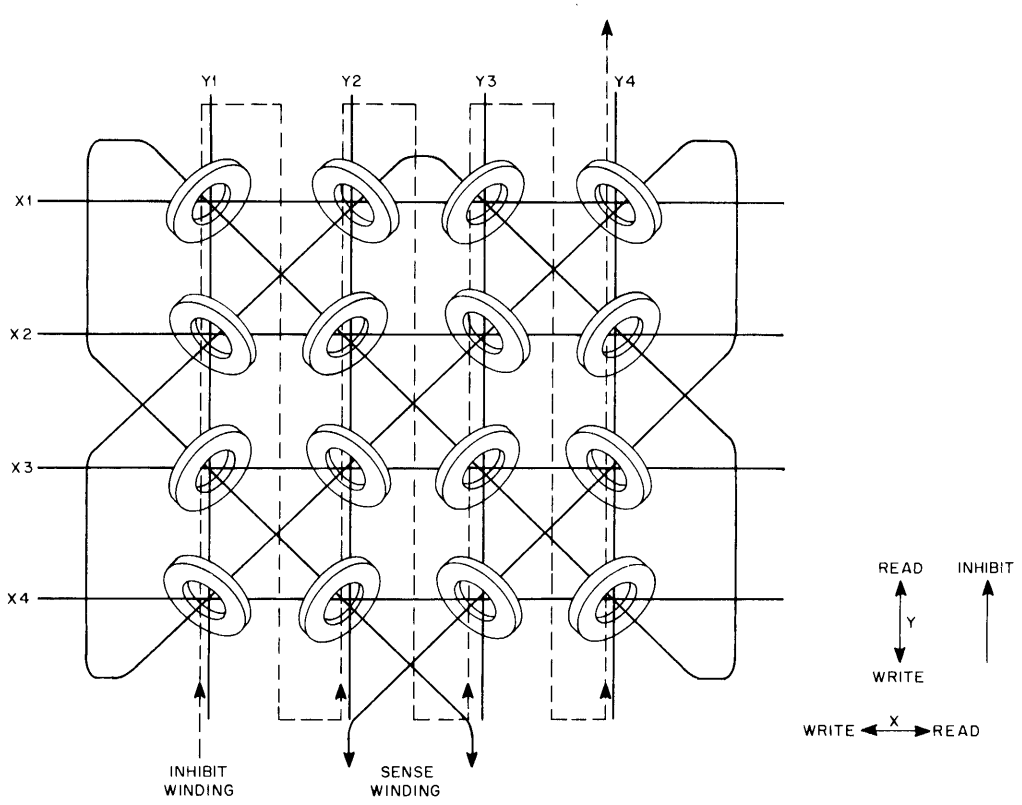


Figure 4-8 Simple Core Memory Plane

always flows in the same direction. The magnetic field generated by the inhibit current of a value and polarity which effectively cancels the field generated by the Y-axis half-select write current. This prevents the setting to the 1 state of any core through which inhibit current is flowing.

Each of the 12 inhibit windings is connected to the output of an inhibit driver circuit. Each driver, in turn, is controlled by the output of one bit of the memory buffer register, which contains the data to be stored in memory. When the write operation commences, each inhibit driver connected to an MB bit containing a 0 is activated. The resulting half-current value output of the driver then prevents the writing of a logic 1 in its assigned core plane. The MB bits containing logic "1s" disable their respective inhibit drivers. This allows the cores pertaining to these bits to have "1s" written into them. The content of the MB is therefore written intact into the selected core-memory storage cell.

To read out information contained in the 12-bit X2Y3 memory cell, half-select read currents are passed through both the X2 and Y3 windings. Since read current flows in the opposite direction of write current, all cores in the X2Y3 cell previously set to the 1 state are switched to the 0 state. Cores already in the 0 state are, of course, unaffected.

A sense amplifier circuit is provided for each of the 12 core planes in the memory. The input to each amplifier is a sense winding which passes through every core on the associated plane. If, during the read operation, the addressed core in a plane makes the 1 to 0 state transition, the flux change induces a current in the sense winding of that plane. This current develops a 20-30 mV pulse at the input to the sense amplifier. This input is amplified, shaped, and after threshold detection is used to set a SENSE flip-flop connected to the output of the sense amplifier when STROBE FIELD 0 is generated.

Addressed cores which were already in the 0 state, when saturated by the full-select read flux, will induce a limited amount of noise into their sense winding. The voltage level produced by this noise (in the order of 5 mV) will be insufficient to activate the sense amplifier associated with that plane. The SENSE flip-flop for that bit will therefore remain clear, indicating a logic 0 in that location.

Since this type of readout destroys the content of the addressed cell (by switching all cores to 0s), the data stored in the Sense Register will be transferred to the MB for restoration to its original location during the write portion of the memory cycle.

4.12.3 Address Selection

The memory selector switches decode the address specified by the MA and select the proper source and return lines for both the X- and Y-axes. These selection circuits are shown on Drawings BS-8L-0-15 (X-axis selection), and BS-8L-0-16 (Y-axis selection) in Volume II of this document. The polarity of the magnetic field applied to the cores of the addressed cell is determined by the direction of current flow through the core read/write windings. It differs for a read or write cycle. The read/write current selection circuits are shown on Drawing BS-8L-0-13 (Memory Control).

Figure 4-9 provides a combined and simplified version of these two drawings showing the selection of a Y-axis memory cell and the method of determination of read/write current direction. The drawing assumes a content of 0 in bits 6 through 11 of the MA, addressing cell 00 on Drawing BS-8L-0-16.

NOTE

The component designation numbers used in Figure 4-9 have been arbitrarily assigned to assist in this discussion and do not relate to actual designations.

With a read operation in process, the READ (1) line will be affirmed enabling gates 3 and 6. The outputs of these gates turn on both Q3 and Q6. This establishes a positive source and negative return for the windings of all Y-axis cores serviced by address-selection gates 1 and 2. This is the proper polarity and current direction for a read operation.

With MA06-11 = 0, gates 1 and 2 have been enabled, and both Q1 and Q2 will conduct. Current then flows through D1 and Q1 to D8. The read current then passes through the winding of the cores on the Y-axis of 00 attempting to switch the cores to the 0 state. There are 12 X 64 cores along this Y-axis; one for each bit and each X-axis. The read current then passes through D4, bypassing Q2 and returns to the memory supply (-) through Q6. The current does not pass through the cores on the 01 through 07 and 10 through 77 Y-axis because of back biased diodes within the matrix, and on the address selection switches (Drawing BS-8L-0-16).

In a write operation, Q4 and Q5 would be turned on by gates 4 and 5, reversing the direction of current flow. The address selection path, however, remains the same through the core winding on the Y-axis of 00. The new current direction (write) attempts to set the cores to the 1 state. As has been previously stated, each core through which write current passes will be set to the 1 state unless the inhibit driver associated with that core has been activated by the sensing of a logic 0 in that particular bit-position of the MB.

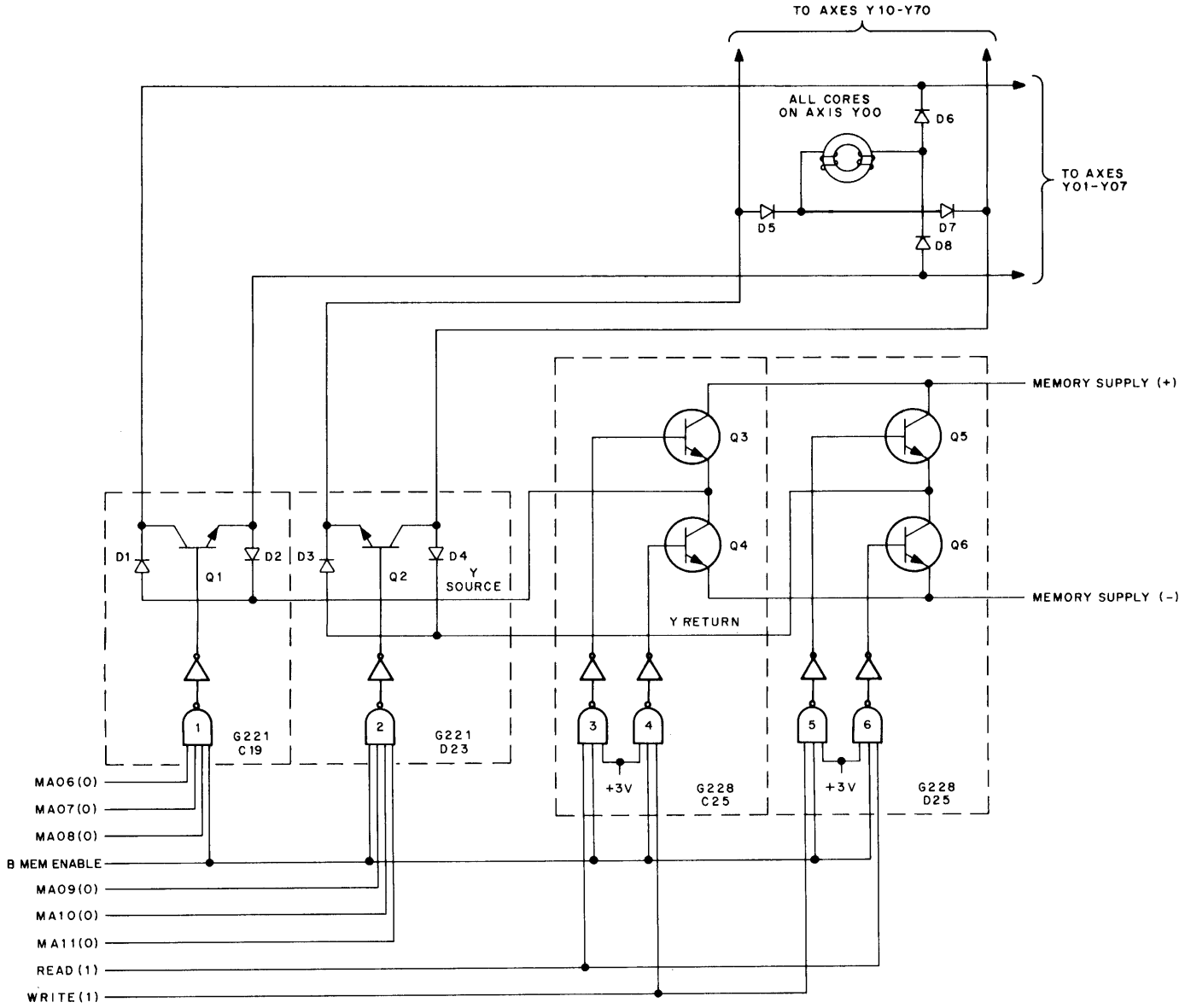


Figure 4-9 Memory Address Selector or Read/Write Current Control

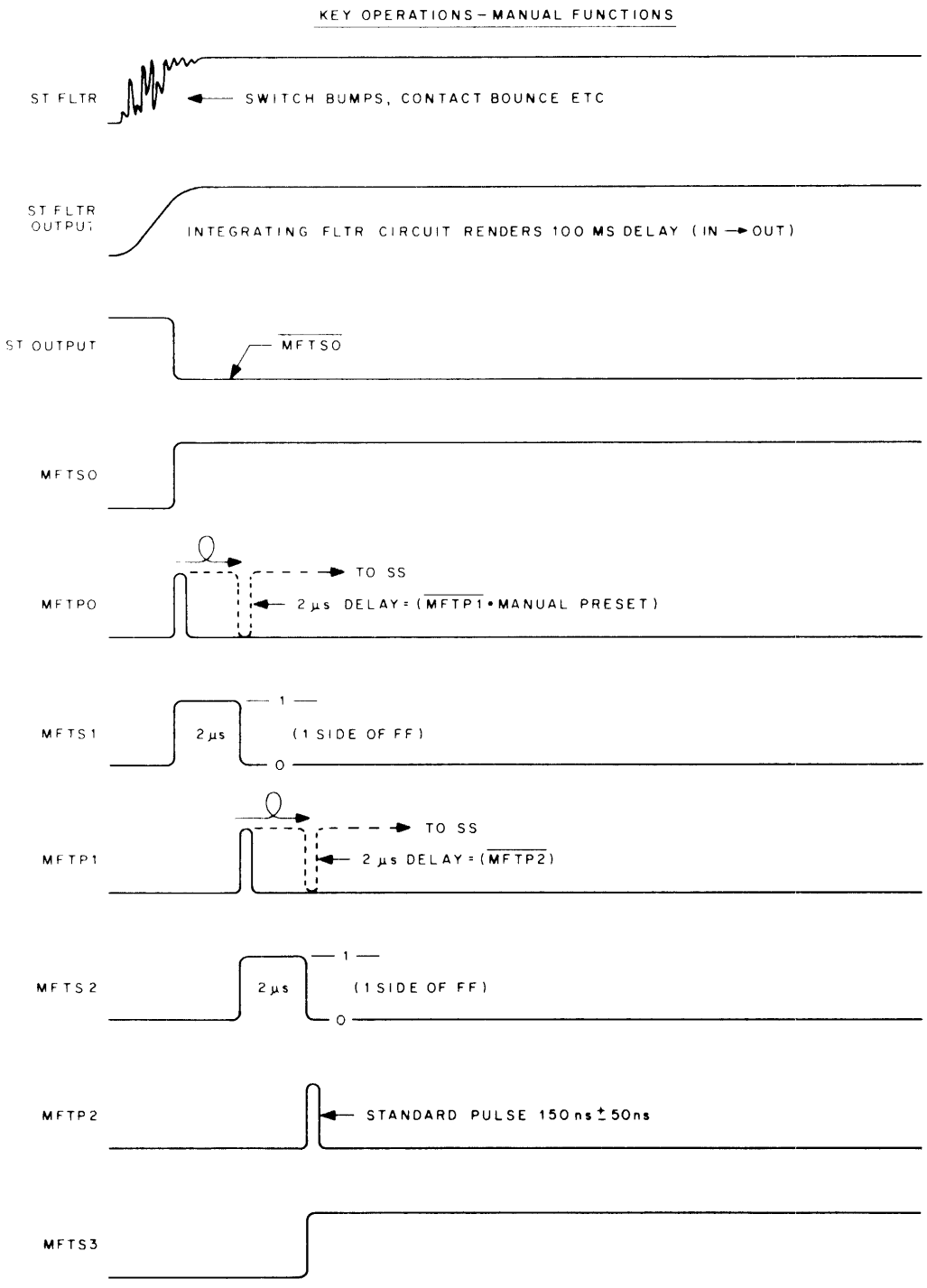


Figure 4-10 Manual Function Timing Diagram

SECTION IV
DETAILED PROCESSOR THEORY

This section describes in detail the computer timing, data flow, and the generation of the instruction set. This detailed discussion of theory enables the service technician to bridge the gap between the engineering drawings and the various logic functions.

4.13 TIMING

The following paragraphs discuss the internal timing of the processor.

4.13.1 Manual Function Timing Generator

When the computer is initially started, or when data is manually deposited, examined, or continued, or the memory addressed, the processor and/or memory cycles are entered by application of the Manual Function Time signals. These signals include the time-state levels MFTS0, MFTS1, MFTS2, and MFTS3, and timing pulses MFTP0, MFTP1 and MFTP2 (Drawing BS-8L-0-2). The levels and pulses are independent from, and not to be confused with, the processor timing-generator signals. The generator and the manual timing signals are described in the following paragraphs. Figure 4-10 shows the timing relationship between the manual timing signals.

When any of the levels $\overline{\text{KEY LA}}$, $\overline{\text{KEY ST}}$, $\overline{\text{KEY EX}}$, $\overline{\text{KEY DP}}$ or $\overline{\text{KEY CONT}}$ are activated by pressing one of the associated keys, a low-to-high level transition occurs. The transition is smoothed by an integrating filter which eliminates the noise generated by the closure of the key. To perform this function, a 100 ms delay is incorporated as part of the filter. The filter output activates the ST (Schmitt Trigger) which combines with RUN (0) to generate $\overline{\text{MFTS0}}$. The RUN (0) level controls the timing generator by preventing manual time levels and pulses from occurring if a key is pressed when the computer is running. $\overline{\text{MFTS0}}$ is inverted to produce MFTS0 which sets the MFTS1 flip-flop. MFTS0 also combines with KEY EX + DEP to enable clearing of the RUN flip-flop during T3 of the processor cycle.

Timing level $\overline{\text{MFTS0}}$ generates an $\overline{\text{MFTP0}}$ pulse. This pulse is delayed 2 μs to produce MFTP1. The MFTP1 pulse sets the MFTS2 flip-flop which clears the MFTS1 flip-flop ending that time state. MFTP1 is delayed 2 μs to generate MFTP2 which clears the MFTS2 flip-flop. The MFTS2(0) level then generates the MFTS3 timing level through a NAND gate.

The applications of the manual function timing levels and pulses are described with the key functions.

4.13.2 Manual Operations

The following keys and switches are provided on the operators' console: START, STOP, LOAD ADDR, CONT, EXAM, DEP, SING STEP, MEM PROT, DATA FIELD, INST FIELD, and the Switch Register. All are single switches with the exception of the SR which consists of a bank of 12 switches.

These switches, used singly or in combination, permit manual intervention to start the program, stop the program, load data into a selected memory location, examine the contents of a memory location, and run the program step-by-step for troubleshooting the system, or debugging new programs.

The following paragraphs contain detailed descriptions of the main control switches. The DATA and INST FIELD switches are described with extended memory operation. The MEM PROT switch function is described in Paragraph 4.15 of this chapter.

LOAD ADDR - The memory location to be addressed is toggled into the switch register, and LOAD ADDR keyed. This clears the major state register. MANUAL PRESET, which performs this operation, is generated by the MFTP0 pulse when LOAD ADDR is pressed.

No further operations occur until MFTS2 when the address toggled into the switches is loaded into the PC and MA. This occurs through the generation of an SR ENABLE signal (Drawing BS-8L-0-4) a MA LOAD signal at MFTP1 and a PC LOAD signal at MFTP2 (Drawing BS-8L-0-6).

Deposit (DEP) - If data is to be manually loaded into memory, the starting location is placed in the PC, as described with LOAD ADDR, and the data loaded, word-by-word through the SR (Switch Register) by the action of DEP.

Pressing DEP clears the major state registers during MFT0 by generating the MANUAL PRESET level. Manual time-pulse MFTP0 generates MANUAL PRESET when DEP is pressed. During MFT1 the contents (starting address) of the PC are transferred to the MA. The generation of the PC ENABLE signal (Drawing

BS-8L-0-4) by MFTS1·KEY ST+EX+DP, and the MA LOAD signal (Drawing BS-8L-0-6) by MFTP·KEY ST+EX+DP perform this transfer operation.

During MFT2, the address in MA is incremented and transferred back to PC, leaving the current address in MA and placing the next consecutive memory address in PC. This transfer is accomplished by an MA ENABLE level (at MFTS2), and a PC LOAD pulse (at MFTP2). The address is incremented by the insertion of a Carry into the adder of the least-significant bit during the transfer of the address into PC. This occurs by generating a CARRY INSERT level (Drawing BS-8L-0-5) during MFTS2.

During this transfer operation the memory cycle is started to permit loading of the SR data into the currently addressed memory location. The actuation of any manual key, except LOAD ADDR, generates a MEM START pulse at MFTP2 (Drawing BS-8L-0-2), initiating the memory cycle.

During MFT3, the actuation of Deposit generates SR ENABLE (Drawing BS-8L-0-4). This gates the data as signified by the Address switch positions, onto the major-register bus. TP2 of each cycle, the MB LOAD pulse (Drawing BS-8L-0-6), permits MB to accept the Address register data present on the major-register bus. During the write portion of the memory cycle, this data is written into the memory location specified by the contents of MA. This completes the deposit cycle.

Subsequent data is loaded in sequential memory locations by toggling the data into the SR and actuating DEP. The PC is incremented for each deposit, making further addressing unneccessary until such time as access to a non-sequential memory address is required.

Examine (EXAM) - The actuation of EXAM permits inspection of the word in the currently addressed memory location. This sequence is identical to the previously discussed deposit operation up to the start of the memory cycle.

During the read portion of the memory cycle, the contents of the address specified, when the LOAD ADDR key was pressed, are transferred from core memory to the Sense register. The memory signal STROBE FIELD 0 allows this transfer and also ends processor time-state TS1, and initiates TS2. At the end of T2, time-pulse TP2 generates an MB LOAD pulse which completes the examine operation by transferring the contents of the Sense register to the MB. Upon completion of this transfer, the processor stops. This permits the operator to examine the contents of the location addressed by observing the MB indicator lights located on the console panel.

During the write portion of the memory cycle, the content of the MB containing the word examined is restored to the original memory location. Thus, the content of the examined address remains in the MB and in the memory location. The content of the address examined may be modified through the use of the Switch Register switches and DEP. It should be noted, however, that as part of the cycle which extracted the data from memory, the PC was incremented by one to set up the address of the next instruction. The next word, therefore, would be loaded into the core-memory address next in sequence to the address of the presently displayed word. The 12-bit SR and LOAD ADDR must therefore be used to set the PC back to the address of the displayed word prior to insertion of the new word.

START - Pressing START initiates execution of a program previously loaded into core memory. When this key is pressed, MFTP0 is combined with the KEY ST level to generate the INITIALIZE and $\overline{\text{INITIALIZE}}$ signals (Drawing BS-8L-0-2) which clear the teletype circuits and initialize the machine. The MFTP0 pulse also generates MANUAL PRESET, which sets the TS1 flip-flop.

During MFT1, the contents of the PC transfer through the major register gating network to the MA, and the MFTS1(0) level clears the IOP flip-flops.

During MFT2, the AC, LINK (Drawing BS-8L-0-8) and INT ENABLE (Drawing BS-8L-0-7) flip-flops are cleared and the Fetch flip-flop is set. The memory cycle is also initiated at this time by the generation of MEM START (Drawing BS-8L-0-2).

The memory signal $\overline{\text{STROBE}}$ is activated during the memory cycle. $\overline{\text{STROBE}}$ initiates the automatic sequences of the processor by continuing the processor cycle. The RUN flip-flop is set by processor time-pulse TP3, and the program instructions are executed until either a halt command is encountered, or the computer is manually stopped.

STOP - Pressing the STOP key will halt a program at the termination of the current cycle. Operation of this key generates a KEY STOP level which clears the RUN flip-flop at the next TP3 pulse. Clearing RUN inhibits generation of the next MEM START pulse preventing another memory cycle. The transfer of any data between registers is inhibited after completion of the current cycle, preventing loss of data. The

operator can examine the contents of the registers prior to the start of the next cycle.

SING STEP - Pressing the SING STEP switch causes the program to step one cycle at a time. The RUN flip-flop is not set. Subsequent actuation of the CONT key generates a single MEM START pulse, but prevents the processor from automatic execution of the program, therefore permitting only a single processor/memory cycle to be executed.

4.13.3 Time States

Four time-state levels (TS1, TS2, TS3, and TS4) and associated time pulses (TP1, TP2, TP3, and TP4) are generated during each computer cycle. This train of levels and pulses is initiated at the start of each memory cycle and terminated upon its completion. The generation of the time state and time pulses and their relationships are discussed below and are shown in Figure 4-11.

TS1, the first time-state produced (Drawing BS-8L-0-2), is entered at the end of the previous processor cycle by TP4. The memory cycle is also initiated at this time by generating MEM START (Drawing BS-8L-0-2). The duration of TS1 and generation of time pulse TP1 depends on the memory signal $\overline{\text{STROBE}}$ which is produced during the Read portion of the memory cycle. Therefore, the duration, of TS1 depends on the memory used and the $\overline{\text{STROBE}}$ delay adjustment.

During processor time TS1, MEM START initiates the memory cycle by progressing through a delay chain (Drawing BS-8L-0-13). MEM START is delayed to generate $\overline{\text{MEM BEGIN}}$ which starts the Read function by setting the READ flip-flop. The memory signal $\overline{\text{STROBE}}$ (Drawing BS-8L-0-13) is generated by an adjustable delay toward the end of the Read portion of the memory cycle by MEM START (delayed). $\overline{\text{STROBE}}$ allows the transition from processor time state TS1 to TS2, and generates time pulse TP1 (Drawing BS-8L-0-2). $\overline{\text{STROBE}}$ also clears the MEM IDLE flip-flop disabling MEM START.

When $\overline{\text{STROBE}}$ generates TP1, the processor-timing-progression continues. TP1 is delayed by 0.15 μs to generate TP2 which clears the TS2 flip-flop and sets the TS3 flip-flop. TP1 is also delayed 0.4 μs to generate TP3. The third time-pulse (TP3) clears TS3, and if the instruction performed is not an input/output instruction (IOT), TP3 sets the TS4 flip-flop.

During processor time-state TS3 the INHIBIT and WRITE control flip-flops (Drawing BS-8L-0-13) are set by the delayed MEM START signal. At the end of the memory cycle, the INHIBIT and WRITE flip-flops are cleared and $\overline{\text{MEM DONE}}$ is generated.

$\overline{\text{MEM DONE}}$ sets the MEM IDLE flip-flop (Drawing BS-8L-0-2). The setting of this flip-flop allows the generation of MEM START (initiating another memory cycle) and TP4, if RUN is still set and PAUSE is cleared. Time pulse TP4 clears TS4 and sets TS1, initiating another processor cycle.

The paragraphs above describe the relationship between the PDP8/L processor and memory timing cycles when the previous cycle initiates the next cycle by generating TP4 and MEM START. This assumes that the PDP-8/L is running. When the computer is initially started (by pressing START, LOAD ADDR*, DEP, or CONT) the processor and memory cycles are entered in the following manner.

The processor timing cycle is initiated by generating $\overline{\text{MANUAL PRESET}}$ which sets the TS1 flip-flop, and simultaneously clears the TS2, TS3, and TS4 flip-flops. $\overline{\text{MANUAL PRESET}}$ is generated by pressing any of the keys mentioned above except for CONT. When this key is pressed, a TP4 is forced, setting the TS1 flip-flop.

By pressing any of the keys above with the exception of LOAD ADDR, MEM START is generated initiating a memory cycle.

Although normal computer operation is through the use of a continuously running stored program, proceeding in sequence from instruction-to-instruction, this process must be started manually. The SING STEP key previously mentioned may also allow computer program progression, a cycle at a time, for maintenance and program debugging purposes.

4.14 I/O TIMING

The processor timing cycle is interrupted between T3 and T4 to perform an I/O cycle, if specified by an IOT (input/output transfer) instruction. During the I/O cycle, up to three pulses (IOPs) can be generat-

* When LOAD ADDR is pressed, MEM START is inhibited, therefore, there is no memory cycle.

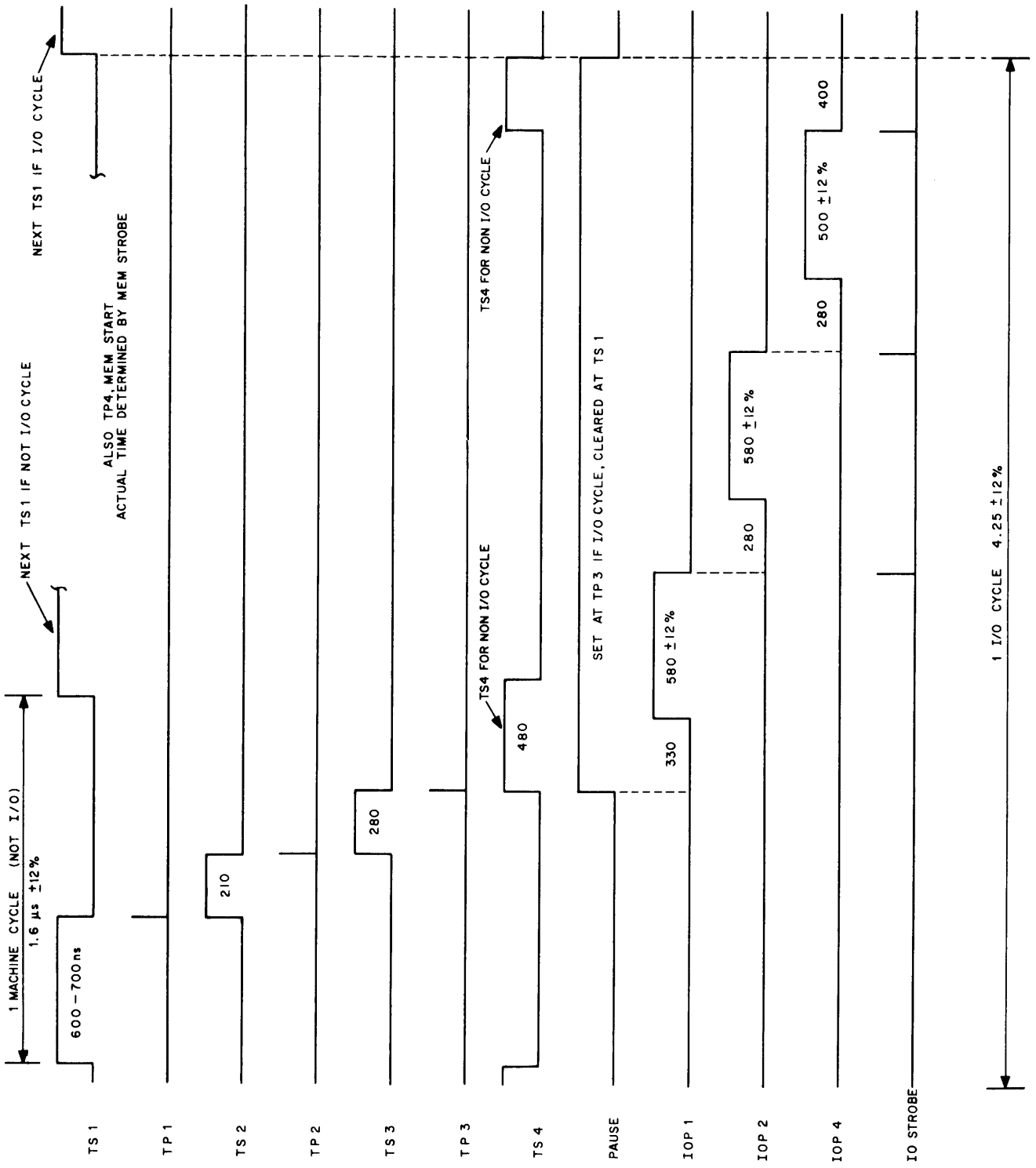


Figure 4-11 System Timing Diagram

ed, depending on the IOT instruction performed (see IOT instruction description in Paragraph 4.18.4). The IOP pulses provide control between the processor and peripheral equipment such as the ASR33 Teletype, the PR8/L High-Speed Reader, and others. The following paragraphs describe the I/O timing and the associated logic circuitry. Refer to Figure 4-12 illustrating the I/O timing, and Drawing BS-8L-0-2.

The IOP generator consists of delay lines with associated logic gates and an IOP register. $\overline{\text{I/O START}}$ is transmitted through the delay lines, re-enters the same lines as $\overline{\text{I/O RECYCLE}}$ and ends the cycle by generating $\overline{\text{I/O END}}$ which initiates processor time state TS4.

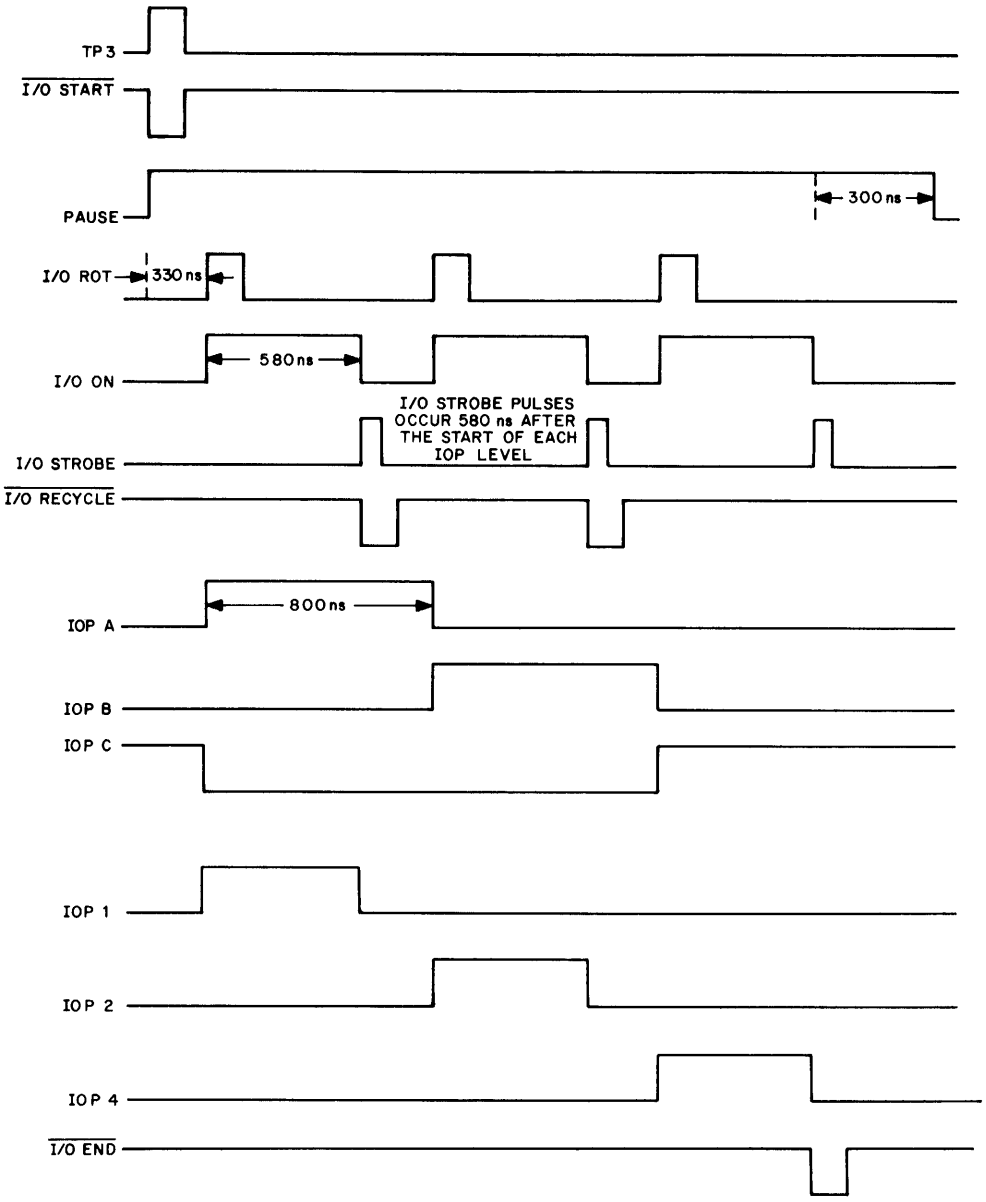


Figure 4-12 I/O Timing Diagram

When activated, I/O START sets the PAUSE flip-flop (Drawing BS-8L-0-2) to prevent initiation of another processor/memory cycle during the I/O cycle. I/O START also triggers the delayed pulse sequence. Approximately * 200 ns after I/O START, I/O ROT is produced, setting the IOPA and I/O ON flip-flops. This allows generation of an IOP1 pulse if MB11 is set. I/O STROBE is generated 600 ns later from another tap of the delay line. I/O STROBE clears I/O ON terminating IOP1 and generating I/O RECYCLE. I/O STROBE is also used in conjunction with IOP pulses to enable AC LOAD (Drawing BS-8L-0-6) for clearing and loading data from a device into the accumulator. However, the clearing and loading operations normally occur as a result of IOP2, and IOP4 (refer to Paragraph 4.18.4.) I/O RECYCLE indicates the end of the first pass through the delay lines. This signal feeds back through the same delay lines, generating another I/O ROT and I/O STROBE. On this pass IOPB is set, and IOP2 is produced if MB10 is active. When the second I/O STROBE is generated, another I/O RECYCLE is issued, and another pass through the delay lines occurs. On the third pass, IOPC is set by I/O ROT, and IOP4 is produced if MB09 is active. The setting of IOPC disables I/O RECYCLE; however, another path is enabled for I/O STROBE, and I/O END is generated. Time state TS4 is entered by I/O END, and after a 300 ns delay PAUSE is cleared. This ends the I/O cycle, and the processor cycle is continued.

4.15 MEMORY PROTECT

When the MEM PROT key is activated, access to memory addresses 7600_8 to 7777_8 in the top memory field is prohibited. Normally, the RIM and BIN loader programs are stored in this memory area. When an additional 4K of memory is added to the system, these loaders are usually stored in the optional field in the same area of memory, instead of in the basic memory field.

The protect feature ensures that no alteration of the restricted memory area occurs, either by inadvertent manual operation, or with program debugging/editing processes**. This feature also allows programming use as a "read only" section of memory, when the RIM and BIN loaders are shifted elsewhere. The following paragraphs describe the protect circuitry on Drawing BS-8L-0-3.

*I/O timing references do not include transition time through logic gates, therefore timing tolerance is ± 20%.

**The MEM PROT key must be unactuated when loading data into the restricted area of memory.

The protect function is performed by generating the PROTECT level. PROTECT is produced by assertion of either the MEM ALT 1, MEM ALT 2, or WC(0) levels combined with TS2(1) and the following signals.

KEY PROTECT (asserted by the MEM PROT key);

EMA (enabling the protect function in the basic memory; disabling it if the extended memory field is added to the system);

MA00 (1) through MA04 (1) (defining the restricted area 7600_8 through 7777_8 in memory).

The MEM ALT 1 and MEM ALT 2 levels are produced in the following manner to allow generation of PROTECT. MEM ALT 1 (Drawing BS-8L-0-4) is produced when the MB is altered by executing the DCA or JMS instructions, by incoming data with a data break request, or by pressing DEP. MEM ALT 2 (Drawing BS-8L-0-5) is generated when the MB could be altered by producing CARRY INSERT with the ISZ instruction, CA INCREMENT or MEMORY INCREMENT with 1 and 3 cycle data breaks, or the EXAM or DEP key.

WC(0) generates PROTECT at the beginning of a 3-cycle data break to prevent alteration of the MB.

When PROTECT is active, the following events occur:

a. ILLEGAL REF flip-flop is set by TP2. ILLEGAL REF (0) is logically ORed with KEY SS. This clears the RUN flip-flop at TP3.

b. MEM ENABLE 0-4/5-11 is disabled preventing a data transfer from the Sense register to the MB.

c. AC ENABLE is disabled during the DCA instruction.

d. SR ENABLE is disabled for the DEP operation.

e. CARRY INSERT is disabled (see MEM ALT 2 previously described).

f. PC LOAD is disabled during the JMS instruction.

4.16 MAJOR STATES

A total of six major states are provided to perform all computer operations. These states are Fetch (F), Defer (D), Execute (E), Word Count (WC), Current Address (CA), and Break (B). Each major state occurs in one complete 1.6 μs computer cycle. The execution of a computer instruction consumes one or more major states, depending upon the operations to be performed. The following paragraphs describe the relationship between the states, their functions and generation.

Fetch (F) - During this state an instruction is read into the Sense register and the memory buffer at the address specified by the content of the program counter. The instruction is restored in core memory and retained in the memory buffer. The operation code of the instruction is transferred to the instruction register for decoding, and the content of the program counter is incremented by one.

The Fetch state is entered by pressing START. The Manual Function Time Generator is then activated, and MFTP2 (Drawing BS-8L-0-2) is produced. This pulse, combined with Key ST, sets the FETCH flip-flop.

The Fetch state can also be entered during T4 time, by TP4 of a Fetch cycle or an Execute cycle. Entry occurs from a single-cycle Fetch cycle including all of the OPR and IOT commands and the JMP command when directly addressed. Entry into the Fetch state occurs from the Execute cycle if $\overline{\text{BRK REQ}}$ occurs. When these conditions are met, F SET (Drawing BS-8L-0-3) is active, enabling the FETCH flip-flop. F SET is produced when the following signals are inactive: $\overline{\text{D SET}}$, $\overline{\text{E SET}}$, $\overline{\text{BREAK OK}}$, and $\overline{\text{SPECIAL CYCLE}}$.

If a multiple-cycle instruction is fetched, the following major state will be either Defer or Execute. The multiple-cycle instructions include the AND, TAD, ISZ, DCA, JMS, and the indirectly-addressed JMP instruction. When the above instructions are directly addressed, the EXECUTE flip-flop is set; when the instructions are indirectly addressed the DEFER flip-flop is set. The following major state entry is executed by TP4 in both cases.

Defer (D) - When a 1 is present in bit 3 of a memory reference instruction, the Defer state is entered to obtain the full 12-bit address of the operand from the address in the current page or page 0, specified by bits 4 through 11 of the instruction. The process of address deferring is called indirect addressing because access to the operand is addressed indirectly, or deferred, to another memory location.

The Defer state can be entered only from the Fetch state when one of the multiple-cycle instructions AND, TAD, ISZ, DCA, JMS, or JMP is indirectly addressed (MB03 = 1). Under these conditions, entry is made during T4 time by TP4 in the Fetch cycle. D SET (Drawing BS-8L-0-3) enables the DEFER flip-flop. It is generated by the active levels MB03 = 1, and B FETCH (1), and the inactive level $\overline{\text{IOT+OPR}}$.

If the multiple-cycle instruction being performed is not a JMP command, entry into the Execute state is made from the Defer state. When the JMP instruction is performed with no BRK REQ, the instruction is completed and the Fetch state is entered.

Execute (E) - This state is entered for all memory reference instructions except JMP. During an AND, TAD, or ISZ instruction the content of the core memory location specified by the address portion of the instruction is read first into the Sense register and subsequently into the memory buffer, and the operation specified by bits 0 through 2 of the instruction (instruction operation code) is performed. During a DCA instruction, the content of the accumulator is transferred into the memory buffer and is stored in core memory at the address specified by the instruction. During a JMS instruction the content of the program counter is written into the next core memory address and the address specified by the instruction is transferred into the program counter to change program control.

The Execute state can be entered at the conclusion of the Fetch, Defer, Execute, or Break state if there is a PROGRAM BRK REQ. In this event the EXECUTE flip-flop is enabled by the E SET level (Drawing BS-8L-0-3). With a PROGRAM BRK REQ, E SET is generated by $\overline{\text{INT OK}}$ (Drawing BS-8L-0-7).

Entry into the Execute state can also occur from two other methods. One of these occurs at the conclusion of the Fetch state when the instruction being performed is a directly-addressed multiple-cycle command. When the signals MB03 (0) and B FETCH (1) are active and JMP is inactive, E SET is generated enabling the EXECUTE flip-flop.

The other Execute-state entry method is from the Defer state when any instruction except JMP is performed. In this event, E SET is produced by the DEFER (1) level and the JMP level inactive.

Regardless of the source of the Execute-state entry, the EXECUTE flip-flop is set in the previous major state during T4 by time-pulse TP4.

The Execute state is the last state that a multiple-cycle instruction enters. At the conclusion of this state the Fetch state is entered again except when the PDP-8/L acknowledges a device requesting any type of break request at this time.

Word Count (WC) - This state is entered when an external device supplies signals requesting a data break and that the break is a 3-cycle break. When this state occurs, a transfer word count in a core memory location designated by the device is read into the memory buffer, incremented by 1, and rewritten in the same location. If the word count overflows, indicating that the desired number of data break transfers will be enacted at the completion of the current break, the computer transmits a signal to the device. The Current Address state immediately follows the Word Count state.

The Word Count state is entered at the end of each major state excepting the Current Address or the Word Count States when there is a request for a 3-cycle data break. The request is acknowledged during T4 time.

The WORD COUNT flip-flop is enabled in the previous major state by the WC SET level (Drawing BS-8L-0-3). This level is generated by the active 3-cycle and BREAK OK levels. The WORD COUNT flip-flop is set by time-pulse TP4 in the previous major state.

At the conclusion of the Word Count state, the combination of WORD COUNT (1) and time-pulse TP4 sets the CURRENT ADDRESS flip-flop to the one state enacting Current Address entry.

Current Address (CA) - As the second cycle of a 3-cycle data break, this cycle establishes the address for the transfer that takes place in the following cycle (Break state). Normally the location following the word count is read from core memory into the memory buffer and incremented by one to establish sequential addresses for the transfers, and also is transferred to the Memory Address register to determine the address selected for the next cycle. An inhibit signal (from the data break device) can be supplied to the computer so that the word read during the cycle is not incremented. Incrementation, if it occurs, occurs on the transfer to the memory buffer from core memory. This word is rewritten into core memory at the same location. The Break state immediately follows the Current Address state.

Since the only entry path to the Current Address state is by progressing through the Word Count state with a 3-cycle data break, the CURRENT ADDRESS flip-flop is set during T4 time by TP4 when the WORD COUNT flip-flop is reset (Drawing BS-8L-0-3).

Break (B) - This state is entered to enact a data transfer between computer core memory and an external device, either as the only state of a 1-cycle data break or as the final state of a 3-cycle data break. When a break request signal arrives and the cycle select signal specifies a 1-cycle (3-cycle) break, the computer enters the Break state at the completion of the current instruction. Information transfers occur between the external device and a device-specified core memory location, through the memory buffer. When this transfer is complete, the program sequence resumes from the point of the break. The data break (one or three cycle) does not affect the contents of the accumulator, the link, or the program counter.

The Break state is entered by the active B SET level and time pulse TP4 in the final major state of the current instruction. In this event, B SET (Drawing BS-8L-0-3) is generated by the inactive BREAK OK level. Synchronization of the asychronous Break signal (from the device) is done with the BREAK SYNC flip-flop (which is set during TP1 of either a one or three-cycle break).

Entry into the Break state also occurs when there is a 3-cycle data break. This is the last cycle of this type of break and Break state entry is entered directly from the Current Address state. When this occurs, B SET, generated by CURRENT ADDRESS (0) (Drawing BS-8L-0-3), enables the BREAK flip-flop. This flip-flop is set by time-pulse TP4 in the Current Address state.

At the conclusion of the Break state; the Word Count state is entered if there is still a 3-cycle break, the Execute state is entered if there is a program break, and the Fetch state is reinstated if there is no longer a break request.

4.17 INTERNAL DATA FLOW

When the content of one of the major registers (MB, MA, AC, PC) is transferred or modified, the data flow proceeds as illustrated by Figure 4-13. For ease in understanding the data flow, the sequence of events

is described in three steps: (1) source, (2) route, and (3) destination.

4.17.1 Source

As Figure 4-13 illustrates, the 12-bit inter-register transfers are gated into the major register network by enable gates. The basic gating levels include: MA ENABLE, SR ENABLE, PC ENABLE, MEM ENABLE, and AC ENABLE. All enable gates are partially conditioned by processor or manual function time-state levels such as TS2(1), MFTS(1), TS3(1). The enable levels allow the data from one register to enter the major register gating network (Drawing BS-8L-0-9) in a parallel transfer.

4.17.2 Route

After the contents of a register(s) are enabled, the data enters the major register gating network including the adders and input to the REGister BUS lines (Drawing BS-8L-0-9, all sheets). The major register gating network consists of an upper and lower gating network and adder for each of 12 bits.

The upper level gating permits the register data to enter the adders by combining major register levels such as AC00(0), DATA00, MEM03, and other data inputs with an enable level, or levels, depending on the operation performed.

The lower level gating network includes the adder circuitry, and logic gates for shifting operations. The adder circuits permit propogation of carries, and provide a method of incrementing data as the ISZ, IAC, and MA + 1→PC functions require whether or not the operation requires a carry. The data in the upper gating levels passes through the adders to the lower level gates (Drawing BS-8L-0-9, all sheets).

When any inter-register transfer within the PDP-8/L is performed, excepting rotate, shift operations, and the AND instruction, a NO SHIFT (Drawing BS-8L-0-5) is generated. NO SHIFT allows data to pass from the adders directly through the lower level gates to the REGister BUS lines. When a shift or rotate instruction is performed, specific upper, and lower gating levels direct data to the adder through a particular lower-level gate to the bus. For example, when the RTR instruction (rotate every AC and Link bit right two places) is performed, AC ENABLE and DOUBLE RIGHT ROTATE levels are generated (Drawing BS-8L-0-5). AC ENABLE allows the AC data into the adders. DOUBLE RIGHT ROTATE directs each adder output two places to the right. For the RTR command, the content of the Link shifts to AC01, AC00 shifts two places to the right (AC02), AC01 shifts to AC03 etc. Identical data shifting into the REG BUS lines to the AC bits occurs with each bit.

When the AND instruction is performed the contents of the MB are gated in on the lower level, bypassing the adders. The NO SHIFT signal does not gate the contents of the MB; however, the AC is gated by this signal. Thus, to allow the MB data to flow to the REG BUS lines, AND ENABLE is generated, and applied in the same manner as the NO SHIFT level, for each MB bit.

4.17.3 Destination

All 12-bit inter-register data transfers enter onto the REG BUS lines 00 through 11 after passing through the major register gating network. These lines are the data input. The data on these lines is loaded into the specified register by a clocking pulse, i.e., if the AC is the destination, the AC LOAD pulse is generated. There are three other major-register load pulses. They are: MB LOAD, MA LOAD, and PC LOAD. Each load pulse occurs at the end of a processor or manual function time period by time pulses such as TP3, TP4, and MFTP1. It should be remembered that time-state levels partially condition the enable levels, and time pulses partially condition the load pulses.

As Figure 4-13 illustrates, data can enter the major register gating network from the MB (AND instruction), MA, PC, AC, SR, the INPUT BUS, DATA BUS, DATA ADDR BUS, or from the Sense (MEM) register. However, data transfers from the REGister BUS can flow only to the MB, MA, PC, or AC.

Data flow into core memory always occurs through the MB under MA addressing control. For example, when data in the AC is transferred to core memory (DCA instruction), the AC data transfers to the MB through the major-register gating network. The contents of the MB are sampled, inhibited if necessary, and written into memory. The contents of the MA determine the address location in which the Write operation occurs. When the contents of a memory location are transferred to the MB, the data flow occurs through the Sense Amplifiers into the SENSE register. The data then transfers to the IR for decoding, and through the major register gating network to the MB.

The data flow from outside the major registers to one of them, occurs through the INPUT BUS for devices such as the Teletype and through the DATA, and DATA ADDR lines for DATA BREAK devices.

4.18 OPERATING INSTRUCTIONS

The following paragraphs describe in detail, the dynamics of the computer operations.

The normal mode of PDP-8/L operation is execution of a prestored programmed instruction sequence. A program interrupt can modify programmed operation, or a data break can temporarily suspend programmed operation. A program interrupt transfers program control from the main program to a subroutine to effect an information transfer with an I/O device or peripheral equipment. A data break is an automatic operation suspending the main program for one or three cycles to permit a high-speed I/O device to exchange information with the core memory.

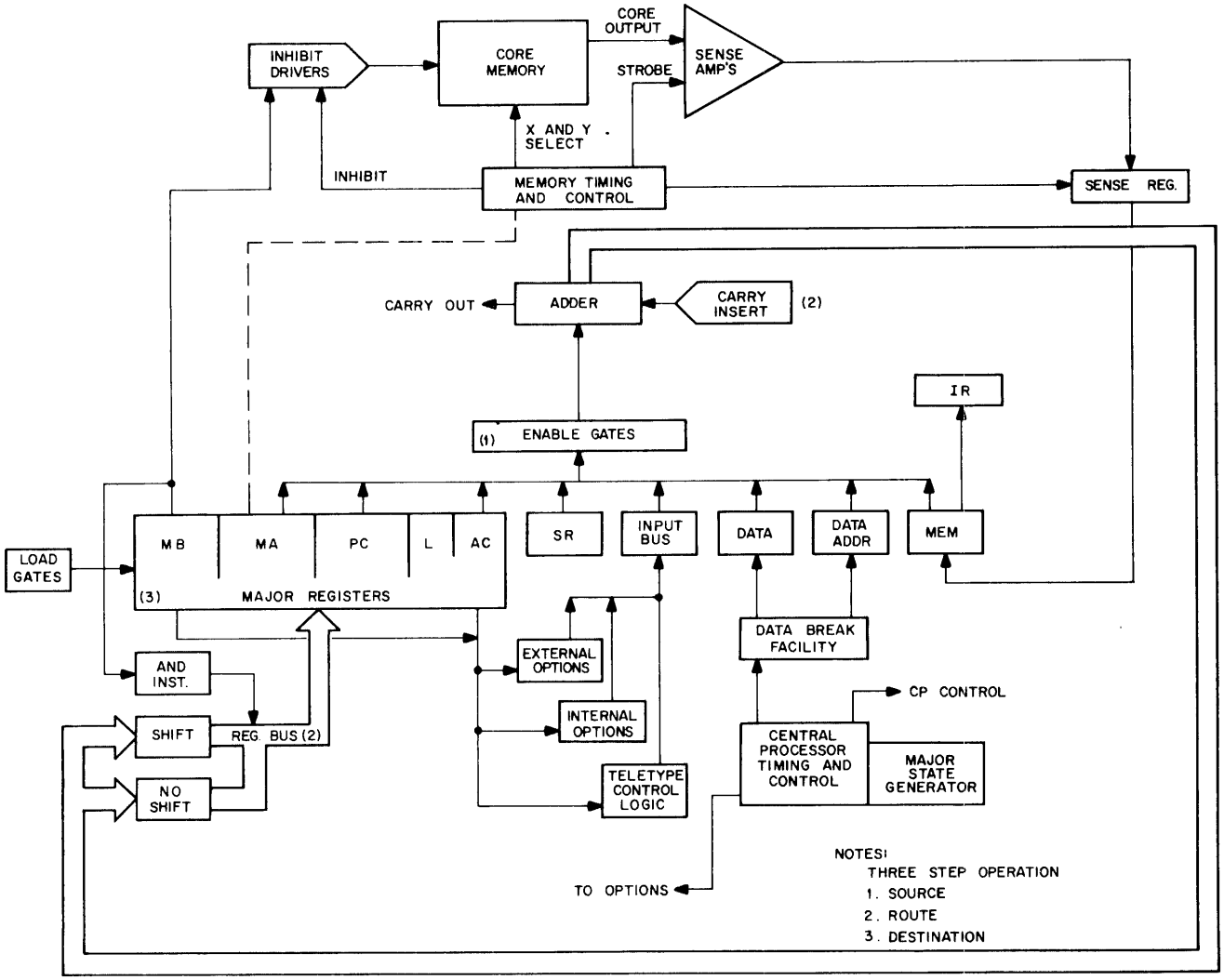


Figure 4-13 Data Flow

4.18.1 Instructions

The following explanations of the functions performed during the execution of each instruction assume that the PDP-8/L is energized and is operating normally under control of the main program. Each explanation begins at the start of the Fetch cycle, when the address of the instruction is in the MA and a memory read operation is initiated.

Instructions performed by the PDP-8/L are either memory reference instructions or augmented instructions. A memory reference instruction contains an operation code (in bits 0 through 2) and an address in core memory at which the operation is to occur (in bits 3 through 11). An augmented instruction is used when the operand is already in a register such as the AC; in this case, no memory address is required. Bits 0 through 2 of an augmented instruction contain the operation code which determines the general class of the instruction. Bits 3 through 11 of the instruction contain information which permits the required operations to occur during the two or three execution time states of a single (Fetch) cycle. Operations performed in this manner are said to be "microprogrammed," since several such operations may take place during a single instruction.

4.18.2 Memory Reference Instructions

The format of a memory reference instruction appears in Figure 4-2. With the exception of JMP, instructions which reference a memory address in page 0 or in the current page occur in two cycles: Fetch and Execute. Instructions which reference any other page require three cycles: a Fetch cycle in which the instruction word is brought out of memory and contains the effective address of the operand in the current page or page 0; a Defer cycle (refer to Direct/Indirect Addressing in this chapter) in which the absolute address of the operand is brought out of memory and enters the MA; and the Execute cycle, in which the operand is brought out of core memory and operated on.

The following explanations of memory reference instructions assume that the instruction is directly addressed; however, the JMP instruction is described with direct and indirect addressing as an example. It is also assumed that no break cycle has been initiated.

AND - The logical AND operation occurs between the contents of the addressed memory cell and the contents of the AC. The result is stored in the AC. In effect, each AC bit is compared with the corresponding memory-cell bit. Only when the AC bits corresponding to the addressed memory-cell bits are both a 1 will the particular AC bit remain a 1 at the end of the operation. The logical AND is therefore a transfer of binary 0s. The original contents of the AC are lost.

The sequence of events listed below describe the order in which the AND instruction is enacted. The events described in a through k are common to all memory-reference instructions.

a. The Fetch state is always entered with all instructions at the completion of the last instruction performed. In all cases, the FETCH flip-flop is set during T4 by time-pulse TP4.

b. With all instructions the functions that occur during times T1 and T2 are the same.

c. During TS1 of the Fetch cycle, the MA is incremented and its contents transferred to the PC by TP1. This will provide the address of the next instruction. The word in the currently addressed location is also read into the Sense register at this time.

d. During TS2 the word in the sense register is ready to transfer into the MB. The Sense bits 0 through 2 are also enabled to the IR. Time-pulse TP2 loads the Sense bits 0 through 2 into the IR and the complete word in the Sense register into the MB. For the AND instruction, the IR will be loaded with 0s because the AND operation code is 0_8. The contents of the IR (0_8) produce the AND level. This level is used by the processor for input-gating control functions for this instruction.

e. No operations occur for the AND, TAD, ISZ, DCA, JMS and indirectly addressed JMP.

f. During TS4, Sense register bits 5 through 11 are enabled to the corresponding MA bits, and during time-pulse TP4 these bits are gated into the MA. The same enabling and gating signals affect Sense register bits 0 through 4 and MA bits 0 through 4 depending on the status of MB04. This memory buffer bit determines whether the addressed cell is on the current page (MB04 = 1), or on page zero (MB04 = 0). If it is on page zero, zeroes are transferred into MA00 through MA04.

g. Also during T4 the next major state entry is determined by the status of MB03. If this bit contains a 1, indirect addressing is indicated and the DEFER flip-flop is set by TP4. If this bit contains a 0, direct addressing occurs and TP4 sets the EXECUTE flip-flop. Only one major state can be entered at any time. The state entered depends on the input-gating of each major state controlling flip-flop during TS4.

h. Towards the end of the processor cycle, the memory Write function occurs and the instruction is written back into core memory. When this is done, MEM DONE is generated, and TP4 is produced initiating another processor cycle and clearing TS4. MEM DONE also enables another memory cycle to be initiated.

i. During the strobe portion of the Execute cycle, the operand stored at the address currently held by the MA reads into the Sense register. At TP2 the operand transfers to the MB.

j. During TS3 the AND ENABLE level permits the AND-combining of the AC and MB through the major register gating network. Time-pulse TP3 sets the AC bits whose register inputs are active (high), and clears the other AC bits.

k. Toward the end of the Execute cycle the operand, which is unaltered in the AND process, is rewritten into memory during the Write portion of the memory cycle. The operand of other instructions such as the DCA, JMS, and ISZ is altered before it is rewritten.

l. If there is no break request and SKIP = 0, the contents of the PC are loaded into the MA during Execute T4 time by TP4. This time pulse also clears the IR and sets the FETCH flip-flop. This concludes the logical AND operation; the program is ready to fetch the next instruction from the location specified by the contents of the MA.

Two's Complement Add (TAD) - The contents of the addressed memory cell add to the contents of the AC in 2s complement arithmetic. The result of the addition is stored in the AC, and the operand (addend) is restored to memory. The original contents of the AC are lost.

During the Fetch cycle, the TAD instruction operates in the same manner as the AND instruction. Refer to events a through h for these operations. The operation code 1_8 is decoded by the IR to generate the TAD gating level used by the processor to implement the TAD operations. The actual two's complement add is performed in the Execute cycle in the following sequence.

a. During the memory strobe portion (T1) of the Execute cycle, the addend reads into the Sense register from the addressed memory cell.

b. During T2 the operand in the Sense register is transferred to the MB and IR for processing and decoding.

c. During T3 the MB and AC outputs are enabled and applied to the major register input gating network. Carries are generated and propogated in the adders as required. Time pulse TP3 allows generation of an AC LOAD pulse during the Execute cycle of the TAD instruction. AC LOAD transfers the sum of the AC and MB into the AC.

During the Write portion of the memory cycle the operand in the MB is rewritten into the original address cell. If there is no break request and SKIP = 0, the contents of the PC are loaded into the MA at this time by TP4. This time pulse also clears the IR and sets the FETCH flip-flop.

This concludes the TAD instruction; the program is ready to fetch the next instruction from the location specified by the contents of the MA.

Increment and Skip if Zero (ISZ) - The ISZ instruction reads the contents of the addressed memory cell into the Sense register, and transfers the contents of this register through the major register gating network with a carry insert to the MB. If the incremented contents of the MB are not 0, the program proceeds to the next instruction. If the incremented contents of the MB are equal to 0, the contents of the PC increment by 1, and the program skips the next instruction. The events that occur in performing the ISZ instruction are listed in sequence below.

a. Operations during the Fetch cycle of an ISZ instruction are similar to those during the Fetch cycle of an AND instruction. Refer to events a through h of the AND instruction. The only difference between these two instructions during the Fetch cycle is the operation code 2_8 decoded by the IR for the ISZ instruction.

b. During T1 of the Execute cycle, the word at the memory location signified by the contents of the MA is transferred into the Sense register.

c. During T2 of the Execute cycle the Sense register is transferred to the MB through the major register gating network. The incrementation occurs through the application of a carry insert level to the adder of the least significant bit during the transfer operation. If the carry insert level produces a carry out from the most significant bit, indicating an all 0 condition in the register, the SKIP flip-flop is set at TP2.

d. No event occurs during T3 of the Execute cycle. However, if the SKIP flip-flop has been set, the PC is incremented and transferred to the MA. This causes the next sequential instruction to be skipped during T4. If the SKIP flip-flop was not set, the contents of the PC are transferred to the MA without incrementation, resulting in a skipping.

e. During the memory Write portion of the memory cycle the incremented contents of the MB are written into the address cell from which they were removed.

f. Time-pulse TP4 (at the end of TS4) sets the FETCH flip-flop if there is no break request.

Deposit and Clear Accumulator (DCA) - The DCA instruction deposits the contents of the AC into the addressed memory cell and the AC clears. The original contents of the addressed cell are destroyed. The sequence of events that occur in performing the DCA instruction are listed below.

a. Operations during the Fetch cycle of a DCA instruction are similar to those occurring during the Fetch cycle of the AND instruction. Refer to events a through h of the AND instruction. The operation code for the two instructions differs. The operation code 3_8 decoded by the IR for the DCA command generates a DCA level which is used as a gate-enable signal for this instruction.

b. During T4 of the Fetch cycle the EXECUTE flip-flop is set to allow entry into this state.

c. During T2 of the Execute cycle, the DCA level combined with B EXECUTE(1) inhibits MEM ENABLE 0-4/5-11. This prevents the contents of the Sense register from transferring to the MB. Therefore, the contents of the addressed cell are lost.

The levels TS1(1), DCA, and B EXECUTE(1) combine to generate AC ENABLE during T2. This allows the contents of the AC to transfer through the major register gating bus to the MB.

At the end of T2, time-pulse TP2 generates an MB LOAD pulse that allows the contents from the AC to be loaded into the MB.

d. During T3, time-pulse TP3 generates an AC LOAD pulse; however, no enable levels are generated. This lack of enable levels places the equivalent of all 0s on the input to the major register gating network. The AC LOAD pulse therefore, loads these 0s into the AC, effectively clearing the register.

e. During the Write portion of the memory cycle the contents of the MB are written into the core location specified by the MA. During T4, if neither a break request not a skip = 1 level is present, the contents of the PC are transferred to the MA to specify the next desired core location. Time pulse TP4 sets the FETCH flip-flop allowing entry into the Fetch cycle.

Jump to Subroutine (JMS) - The JMS instruction provides an exit from the main program into a subroutine. The contents of the PC (current program count) incremented by 1, are written into the core memory address specified by the JMS instruction. That address transfers to the PC and increments by 1; this incremented address fetches the first subroutine instruction during the next instruction cycle. When the subroutine ends, the main program is re-entered by a jump indirect to the address specified by the original JMS instruction. The contents of that address are now the incremented main-program count; and transferring this count into the PC causes the main program sequence to continue.

The sequence of events in performing the JMS instruction are listed below. In addition, to further clarify the JMS operation, a sample program with this instruction is given in Table 4-1.

a. Operations during the Fetch cycle of a JMS instruction are similar to those occurring during

the Fetch cycle of the AND instruction. Refer to events a through h of the AND instruction. The operation code for the two instructions differ. The operation code 4_8 decoded by the IR for the JMS command generates a JMS level which is used as a gate-enable level for this instruction.

b. During T1 of the Execute cycle no operations occur.

c. During T2 of the Execute cycle the contents of the PC are transferred to the MB if there is no skip condition (SKIP = 0). In order to perform this operation, PC ENABLE and MB LOAD are generated. PC ENABLE allows the contents of the PC to enter the major register bus. The contents of the bus are loaded by MB LOAD into the MB.

d. During T3, the current address is incremented by one and transferred to the PC. To do this, the MA ENABLE, CARRY INSERT, and PC LOAD signals are generated. MA ENABLE allows the contents of the MA to enter the major register network bus. CARRY INSERT adds one to the contents of the bus. Finally, PC LOAD loads the PC with the incremented contents of the bus.

e. During the Write portion of the memory cycle, the contents of the MB (described in event c of this instruction) are written into memory at the location specified by the JMS instruction.

f. During T4 the FETCH flip-flop is set by TP4 if there is neither a break request nor a SKIP = 1 level.

The events above describe the JMS operation. These events are easier to understand, however, if a concrete example is given. The following events describe the sample program of Table 4-1. The program sequence assumes that the main program is in page D of memory (current page), and that the 21st instruction is JMS directly page 0 location 100_8. The following conditions are also assumed for this example: memory pages are designated 0, A, B, C, D, E; each page contains locations designated 0 through 177_8; and the subroutine is in page 0 starting at location 101_8.

a. During T4 of instruction 20_8 in the main program, the PC contains the address of the next instruction, location 21_8 in page D (current page). This address is transferred into the MA.

b. During TS1 of the Fetch cycle for instruction 21_8, the contents of cell $D21_8$ reads into the Sense register. Upon completion of the Read operation the Sense register contains JMS/0/100 (4100_8).

The JMS operation code 4_8 is in bits 00 through 02; page 0 is specified by bit 03 = 0 (denoting a direct address) and bit 4 = 0 (denoting page 0). Bits 05 through 11 specify location 100_8 (of page 0). Also during TS1 the contents of the MA increments as it transfers into the PC.

c. During T2, the contents of the Sense register (the JMS instruction) transfers into the MB. Bits 00 through 02 transfer into the IR where they are decoded to produce the JMS level.

d. No operations occur during TS3.

e. During T4 of the JMS Fetch cycle, the contents of the MB (the JMS instruction) is written back into its original core location ($D21_8$).

f. At TP4, the contents of bits 05 through 11 are transferred to the corresponding bits of the MA and, because bit MB04 = 0, bits MA00 through 04 are cleared to indicate page 0. The MA now contains $(0/100)_8$, the address specified by the JMS instruction.

g. During T1 of the JMS Execute cycle, the contents of core location ($0/100_8$), as specified by the address portion of the JMS instruction, (which is now in the MA), reads into the Sense register and is lost.

h. During T2 of the JMS Execute cycle, the contents of the PC, which is $D/22_8$, (address of the next sequential main-program instruction) transfers into the MB. The SKIP flip-flop is assumed cleared.

i. During T3, the contents of the MA, ($0/100_8$), which is the current subroutine address, is incremented by one as it is transferred to the PC.

j. During T4, the contents of the MB ($D/22_8$) are written into memory location ($0/100_8$). At TP4, the contents of the PC ($0/101_8$) transfer to the MA to select the core memory location containing the first active instruction of the subroutine. At this time, the JMS Execute cycle is terminated and the Fetch cycle of the first instruction of the subroutine entered.

k. During T1 of the Fetch cycle, the first instruction of the subroutine reads from core location ($0/101_8$) into the Sense register. The program then proceeds to execute the subroutine.

l. The last instruction of the subroutine must be a jump indirect to the location originally specified by the JMS instruction, in this case ($0/100_8$). As noted in step j above, location ($0/100_8$) contains the address in core memory of the next sequential main-program instruction ($D/22_8$). By this means the subroutine is terminated and the main program re-entered at the point at which it was interrupted.

Jump (JMP) - The JMP instruction links two program instructions that are executed consecutively when the instructions are not in sequential locations. This instruction is commonly used to link a program together when the program length extends over more

Table 4-1
Example of Register Contents During JMS Instruction

		PC Contents				MB Contents		MA Contents		
Cycle	Time	0-4 (page)	5-11 (location)			0-4	5-11	0-4	5-11	Command
Fetch or Execute	TS4	D	21	D/20	Unknown	Unknown		D	20	PC→MA 1→F
Fetch	TS1	D	21	D/21	JMS/0/100	Unknown		D	21	MA+1→PC Memory →MB
	TS2	D	22	D/21	JMS/0/100	JMS/0/100		D	21	MEM→M MEM→1B
	TS3	No Operations								
	TS4	D	22	D/21	JMS/0/100	JMS/0/100		JMS/0/100		MB→Memory MEM→MA 1→E
Execute	TS1	D	22	0/100	xxx/x/xxx			0	100	Memory→MB
	TS2	D	22	0/100	xxx/x/xxx	D	22	0	100	PC→MB
	TS3	0/101		0/100		D	22	0	100	MB→MEM MA+1→PC
	TS4	0/101		0/100		D	22	0	100	MB→MEM PC→MA 1→F
Fetch	TS1	0	102	0/101	1st subroutine instruction			0	101	MA+1→ Memory →

than one page (177_8 locations) of core memory. JMP is also extensively used in program loops such as counting and comparing in conjunction with the skip instructions.

The JMP instruction contains either the absolute core-memory address of the next operand (direct addressing) or the address of a location containing the absolute core-memory address of the next operand (indirect addressing). When the next operand is located either in the current page or page zero of memory, direct addressing is used requiring only a single fetch cycle to extract the operand and prepare for its execution. If, however, the next operand is located in any other page in memory, its 12-bit absolute address must be stored in either the current page or page zero at a location specified by bits 05-11 of the JMP instruction. This is known as indirect addressing, and requires both a Fetch cycle and a Defer cycle to extract the operand for processing.

The events that occur in performing the JMP instruction are listed in sequence below.

a. Operations occurring during T1 and T2 of the JMP Fetch cycle are identical to those events of the AND instruction in the same time periods except for the operation code 5_8 decoded by the IR for the JMP instruction. Refer to events a through d of the AND instruction.

b. Operations during T3 of the Fetch cycle depend upon whether the JMP specifies direct (MB03 = 0) or indirect (MB03 = 1) addressing. If indirect addressing is indicated no operations occur during T3. If direct addressing is indicated, the specified address (SENSE 05 through 11) is loaded into the corresponding bits of the PC. If the instruction specifies that the operand is located on page 0 (MB04 = 0), bits 00 through 04 of the PC are cleared. If, however, the instruction specifies that the operand is in the current page (MB04= 1) bits 00 through 04 of the MA are transferred to the corresponding bits of the PC.

c. The following events occur during T4 of the Fetch cycle of a direct address. JMP (if neither a break request nor a skip is specified):

The PC transfers to the MA, the JMP instruction is restored intact, to its original core memory location, and the FETCH flip-flop is set. The operand is removed from core-memory during the next machine cycle (Fetch) and implemented.

d. For an indirectly addressed JMP during T4, bits 05 through 11 of the Sense register transfer to the corresponding bits of the MA and bit 04 of the MB is examined. If MB04 = 0 (absolute address of operand on page 0) MA00 through MA04 are cleared. If MB04 = 1 (absolute address of operand on current page) bits 00 through 04 of of the MA (current page address) are circulated out of, and back into, the same MA bits. Also during T4, the JMP instruction is restored, intact, in its original core memory location. At the end of T4 the DEFER flip-flop is set allowing entry into the Defer state.

The following events relate to the JMP instruction when the Defer state is entered. This state can be entered with any of the memory reference instructions and is not restricted to JMP instruction exclusively.

e. During T1 of the Defer cycle, the absolute 12-bit address of the operand is read from the memory location specified by the JMP instruction (or any of the memory reference instructions) into the Sense register.

f. During T2, the contents of the Sense register are transferred to the MB if an Auto Index is not required. When there is an Auto Index, the contents of Sense are incremented by 1 in the major register gating network before loading into the MB. An Auto Index occurs when a memory reference instruction such as the JMP command is indirectly addressed in one of the locations 10_8 through 17_8 on page zero of memory.

g. During T3 the contents of the Sense register are transferred to the PC (intact if an Auto Index is not specified, and incremented by one if an Auto Index is specified).

h. During T4 if no break request is specified, the contents of the PC are transferred to the MA. Also during T4 the contents of the MB are written back into memory at the original location (intact if Auto Index was not performed, or incremented by 1 if Auto Index was performed). At the end of T4 the FETCH flip-flop is set allowing Fetch cycle entry for the next instruction performed. During the ensuing Fetch cycle the operand is read from memory and its operations begun.

4.18.3 Direct/Indirect Addressing

Six of the eight basic instructions in the PDP-8/L repertoire are designated as memory-reference instruc-

tions. These instructions (AND, TAD, ISZ, DCA, JMP, and JMS), as part of their function either write into or read from memory.

The first three bits (0-2) of these 12-bit instructions contain the operation code designating the specific function to be performed. The remaining nine bits (3-11) are, therefore,available to specify the memory location involved in the required operation. A complete specification of any one of the 4096 locations in the basic PDP-8/L memory, however, requires the use of 12 address bits ($2^{12} = 4096$). To minimize the number of instructions required to access memory, therefore, both direct and indirect addressing is used in the PDP-8/L.

The memory is organized into 32 pages (or blocks), each containing 128 consecutive memory locations. These pages are numbered 0 through 37_8. The specification of any of the 128 locations on a particular page requires only seven bits ($2^7 = 128$). Bits 5 through 11 of the memory-reference instructions are used for this purpose. With the operation code carried in bits 0 through 2, bits 3 and 4 remain to specify the direct/indirect addressing mode of operation.

The status of bit 3 of the instruction specifies whether direct or indirect addressing is to be performed. When bit 3 = 0 direct addressing is specified, i.e., the location specified in bits 5 through 11 contains the operand upon which the function described in bits 0 through 2 is to be performed. If bit 3 = 1 indirect addressing is specified, i.e., the location specified in bits 5 through 11 contains the absolute 12-bit address of the operand. An additional computer cycle (Defer) is therefore required to extract the 12-bit address of the operand from the specified address.

The status of bit 4 determines whether the location specified by bits 5 through 11 is on the currently addressed page of memory, or on page 0 (1 = current page, 0 = page 0). Through the use of bit 4, therefore, a memory reference instruction can address 256 locations; 128 in the current page, and 128 in page 0.

It should be noted that the full, 12-bit absolute address of the desired location must be present in the MA to permit access to that location. The 12-bit starting address of the program is entered into the MA through the switch register and the LOAD ADDR key when programmed operation commences. In normal operation, the PC is incremented during each Fetch cycle to step the address to the next sequential memory location. When a memory reference instruction is extracted from memory, only bits 5 through 11 are transferred to the MA. If bit 4 of the instruction is a 1, bits 0 through 4 of the MA are left unchanged and the next location addressed is on the current page. If, however, bit 4 is a 0, bits 0 through 4 of the MA are cleared. This addresses the bit 5 through 11 location on page 0 of the memory.

Figure 4-14 is a simplified flow chart showing the sequence of operation of both the direct and indirect addressing functions.

4.18.4 Augmented Instructions

The two classes of augmented instructions used in the PDP-8/L are: The input/output transfer (IOT) which has the operation code 6_8 and the operate instruction (OPR), which has the operation code 7_8. Augmented instructions are single-cycle (Fetch) instructions which initiate various operations as a function of bit microprogramming.

Input/Output Transfer - The IOT class of augmented instructions generate pulses (IOP pulses) that allow the PDP-8/L processor to communicate with both the internal and external devices. The IOP pulses generated by performing this instruction are used for timing, control applications, synchronization, and data transfer functions.

The format of the IOT instructions differs from that of the memory reference instructions as illustrated in Figure 4-2. Bits 00 through 02 contain the operation code (6_8), bits 03 through 08 form a code that enables device selector circuitry in a given I/O device, and bits 09 through 11 enable the generation of IOP pulses which control the data transfer, clearing, and synchronization operations.

The following events describe the operation of the IOT instructions.

a. Operations during T1 and T2 of the IOT Fetch cycle are identical to events a through d of the AND instruction, except for the operation code 6_8 decoded by the IR for IOT instructions during T2. The resulting IR generated IOT level is used in IOP generation and other processor control functions.

b. At the end of TS3 of the Fetch cycle, TP3 sets the PAUSE flip-flop and triggers the IOP generator. The processor timing sequence is interrupted between T3 and T4 to allow the I/O cycle to occur.

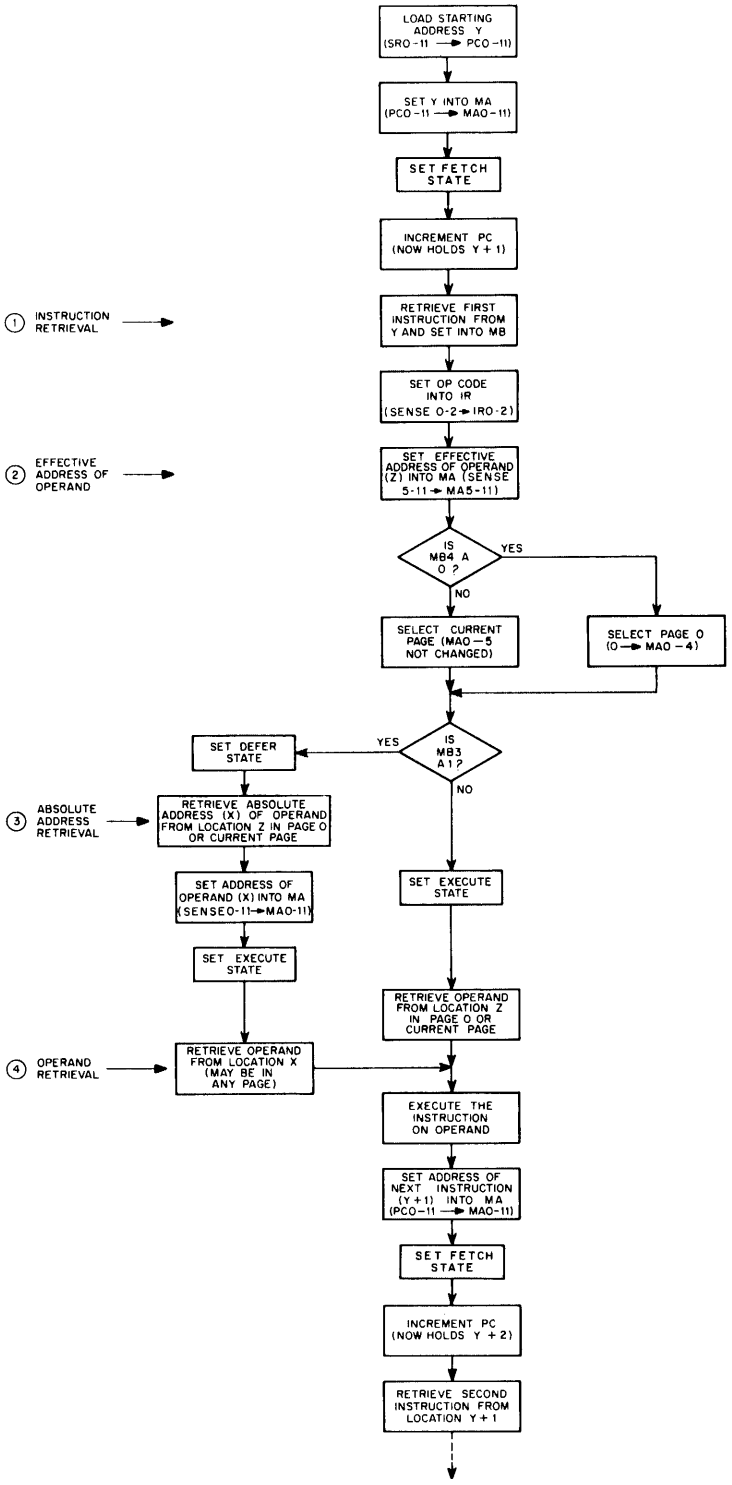


Figure 4-14 Direct and Indirect Address Selection, Simplified Flow Chart

c. During the I/O cycle specific IOP pulses are generated depending on the status of bits MB09 through MB11.

d. Any IOP pulses that are generated are gated in the device selector of the addressed I/O device to produce IOT pulses. The IOT pulses control the operation of the device, effect a transfer of information between the device and processor, or initiate action in the processor such as clearing the AC, or incrementing the PC. In addition, with each IOP pulse generated, I/O STROBE is produced and used within the processor to allow AC LOAD (Drawing BS-8L-0-6) to clock data from the I/O device into the AC.

Certain IOT instructions are normally combined to clear the AC and transfer data to the accumulator in one computer cycle. This is performed by generating AC LOAD as described above, and disabling AC ENABLE (Drawing BS-8L-0-4). AC CLEAR (Drawing BS-8L-0-4), gated by I/O STROBE, disables AC ENABLE, thus 0s are loaded into each AC bit when I/O STROBE occurs. During IOP 2 or 4, time data is loaded into the AC by I/O STROBE.

e. Two IOT instructions used with the PDP-8/L to enable/disable the program interrupt facility. These instructions ION (6001_8), and IOF (6002_8) are processor IOT instructions. When ION is performed, the program interrupt facility is enabled by setting the INT ENABLE flip-flop. IOF disables the program interrupt facility by ensuring that the INT ENABLE flip-flop is cleared.

Operate (OPR) - The OPR class of augmented instructions consists of two categories of microinstructions Group 1, and Group 2. The formats of these groups appear in Figures 4-4 and 4-5, respectively. In each case, bits 00 through 02 contain the operation code 7_8. Group 1 commands, designated by a 0 in bit 03, are called OPR 1 instructions. OPR 1 instructions perform AC and Link operations such as clearing, complementing, rotating, and incrementing. Group 2, designated by a 1 in bit 03 and a 0 in bit 11 are called OPR 2 instructions. These commands check the contents of the AC and Link, and on the basis of the results, determines whether the next sequential instruction is to be performed or skipped.

The OPR 1 operations may occur either singularly or in logical combinations. Care must be exercised, however, to ensure that contradictory or conflicting operations are not specified within the same instruction. For example, bit 08 of the OPR 1 microinstruction, when set to 1 specifies an RAR (rotate AC and L to the right one place) operation. Bit 09 similarly specifies an RAL (rotate AC and L to the left one place) operation. It is physically possible, by setting both bit 09 and 08 to the 1 state, to request an impossible, conflicting operation, i.e., rotate AC and L both right and left one place simultaneously.

Both groups of OPR microinstructions are single-cycle (Fetch) instructions. The following sequence of events describe the operations of this class of instructions. The instruction descriptions contain the primary signals necessary to perform the specified operations. Reference to the mechanization charts indicates where and how the primary signals are generated.

For ease of explanation, the signals and conditions common to most of the OPR instructions are listed below.

a. The operation code is 7_8.

b. Actual AC and L operations occur during T3 of the Fetch cycle.

c. The OPR level decoded by the IR serves as an enabling level, and also combines with MB03 to form OP1 and OP2 levels.

d. Time pulse TP3 generates the load signals such as AC LOAD, that perform the transfer of data from the major register gating network to the indicated register.

e. The major state B FETCH (1) and time-state TS3 signals are used as enabling levels.

f. Except for the rotate OPR 1 instructions, the NO SHIFT signal is generated to allow all transfers from the adder of the major register gating network onto the network bus.

g. There is only one OPR 1 NOP instruction (7000_8); however, each bit is discussed in order. Therefore, several events occur when this instruction is performed.

The OPR instructions and their operations are described below.

a. During T1 and T2 of the Fetch cycle the operations that occur are identical to events a through d of the AND instruction.

b. During T3 bit 03 of the OPR instruction is examined to determine whether the instruction

is an OPR 1 (MB03 = 0) or an OPR 2 (MB03 =1). Appropriate levels are generated in the control circuits to implement the requirements of each group. The descriptions of the instructions that follow should be carefully traced in the flow chart (Drawing BS-8L-0-1) to facilitate understanding of the system operation. The order of appearance of the instruction descriptions parallels the order shown in the flow diagram. It should be noted that, through microprogramming, operations may be combined during the same cycle. For instance; $\overline{AC} \rightarrow AC$ (CMA) and $\overline{L} \rightarrow L$ (CML) may occur in the same instruction to complement both the AC and the Link.

c. During T3 when the OPR 1 instructions are specified (MB03 = 0) the following instructions can be performed. They are:

(1) NOP (no operation). When the 7000_8 instruction is performed, MB04 = 0, MB06 = 0, and no other operand exists. The contents of the AC are circulated through the major register and adder network, and returned unaltered to the AC (AC → AC). This transfer occurs when AC ENABLE, AC LOAD, and NO SHIFT are generated. The events described in 5 and 9 of the OPR 1 instructions also occur simultaneously with the AC circulation.

(2) CMA (complement the AC). When the 7040_8 instruction is performed, MB04 = 0, MB06 = 1, and no other operand exists. The 0 side of all AC bits ($\overline{AC}$) is transferred through the major register gating network to the AC ($\overline{AC} \rightarrow AC$). This transfer occurs when the AC ENABLE, NO SHIFT and $\overline{AC}$ LOAD signals are generated.

(3) CLA (clear the AC). When the 7200_8 instruction is performed MB04 = 1, MB06 = 0, and no other operand exists, logic 0 levels are transferred to all AC bits. This transfer occurs by producing AC LOAD and NO SHIFT levels, with no enable levels existing. This operation clears (sets all bits to 0) the AC register (0 → AC).

(4) CLA/CMA (clear and complement the AC). When the 7240_8 instruction is performed MB04 = 1, MB06 = 1, and no other operand exists. The AC is cleared then complemented ($AC + \overline{AC} \rightarrow AC$, set AC = -1). This effectively sets all of the AC bits to the 1 state. This operation occurs when the AC ENABLE, $\overline{AC}$ ENABLE, NO SHIFT and AC LOAD signals are generated.

(5) NOP (no operation). When the 7000_8 instruction is performed, MB05 = 0, MB07 = 0, and no other operand exists (refer to instruction description 1). The $\overline{L} \rightarrow L$ circulation occurs and the contents of the Link are circulated through the major register gating network and returned unaltered to the Link. This transfer occurs when the L ENABLE, AC LOAD, and NO SHIFT signals are generated.

(6) CML (complement the Link). When the 7020_8 instruction is performed, MB05 = 0, MB07 = 1, and no other operand exists. The 0 side of the Link bit ($\overline{L}$) transfers through the major register gating network into the Link ($\overline{L} \rightarrow L$). This transfer occurs when L ENABLE, AC LOAD, and NO SHIFT levels are generated.

(7) CLL (clear the Link). When the 7100_8 instruction is peformed, MB05 = 1, with no other operand exists, and a logic 0 level is transferred to the Link. The operation is performed by generating AC LOAD and NO SHIFT signals with no enable levels existing.

(8) STL (set the Link). When the 7120_8 instruction is performed, MB05 = 1, MB07 = 1 and no other operand exists, the combined operations of the CML and CLL instructions described in 6 and 7 above occur ($\overline{L} + L \rightarrow L$). The Link is set to the 1 state by clearing it, then complementing it.

(9) NOP (no operation). When the 7000_8 instruction is performed, MB08 = 0, MB09 = 0, and no other operand exists. In addition to the events described for this instruction in 1 and 5, the NO SHIFT level is generated. This level controls transfers of all data from the adder onto the major register bus. This signal is generated for all OPR 1 instructions, and is gated by MB08 = 0, and MB09 = 0. It should be noted that the NO SHIFT signal is also generated for all instructions except for the EAE shift group, but there is a different path activated.

(10) RAR (rotate the AC and Link right one place). When the 7010_8 instruction is performed, MB08 = 1, and no other operand exists. The contents of the AC and L are shifted one bit to the right. The Link status is transferred into AC00, while AC11 status transfers into the Link. The RAR operation occurs when the L ENABLE, AC ENABLE, RIGHT SHIFT and AC LOAD signals are generated.

(11) RAL (rotate AC and the Link left one place). When the 7004_8 instruction is performed, MB09 = 1, and no other operand exists. The contents of the AC and Link are shifted one bit to the left. The content of AC00 is transferred to the Link, and the content of the Link bit is transferred to AC11. The RAL operation is performed when LEFT SHIFT, L ENABLE, AC ENABLE, and AC LOAD signals are generated.

(12) RTR and RTL (double rotate, right or left). When the 7012_8 instruction (RTR) or the 7006_8 instruction (RTL) is performed, MB10 = 1, and the operand for either the RAR or RAL instruction exists. However, to avoid confliciting operations, only one of these operands should exist at one time. The RTR command rotates the contents of the AC and Link two places to the right. Similarly, the RTL command rotates the contents of the AC and Link two places to the left.

Both the RTR and RTL instructions occur when L ENABLE, AC ENABLE, DOUBLE RIGHT ROTATE or DOUBLE LEFT ROTATE, and AC LOAD are generated.

(13) IAC (increment the AC). When the 7001_8 instruction is performed, MB11 = 1 and no other operand exists. The AC is incremented by 1, by transferring the contents of the AC to the major register gating network, adding a logic 1 through the network adder, and transferring the incremented contents into the AC. The AC ENABLE, AC LOAD, and CARRY INSERT signals are generated to perform this instruction.

The IAC command can be combined with either the CLA or CMA commands to perform the functions of both during one cycle. When the CLA/IAC instruction or the CIA (IAC/CMA) instruction is performed, the operations of both separate instructions are performed simultaneously, during T3.

When MB03 = 1, and MB11 = 0 the OPR 2 group of microinstructions is specified. These microinstructions may be performed singly or in useful logical combinations. The available commands include: CLA, HLT, OSR, and seven skip instructions dependent upon the status of the AC and/or Link.

The operations performed during T1 and T2 of the Fetch cycle of the OPR 2 class of instructions are identical to those previously described for those time states in the AND instruction. The following instruction descriptions therefore, describe the OPR 2 instructions during T3 of the Fetch cycle.

(1) SZA (skip on zero AC). When the 7440_8 instruction is performed MB06 = 1 and no other operand exists. The contents of the AC register are checked and if the AC = 0, the next sequential program instruction is skipped. When the AC = 0, the SKIP flip-flop is set to the one state by TP3, B FETCH (1), OPR, and a skip-enable level generated by AND-combination of all AC 0-side outputs.

(2) SMA (skip on minus AC). When the 7500_8 instruction is performed MB05 = 1 and no other operand exists. The content of AC00 is sensed to determine its status. If AC00 = 1, a minus AC is specified, the SKIP flip-flop is set, and the next sequential program instruction is skipped. This flip-flop is set by a load pulse generated by the TP3, B FETCH (1) and OPR signals. The skip-enable level is produced by MB05 (1) combined with AC00 (1).

(3) SNL (skip on non-zero Link). When the 7420_8 instruction is performed, MB07 = 1 and no other operand exists. The Link bit is sensed to determine whether it is in the 1 state, and if it is, the next sequential program instruction is skipped. The SKIP flip-flop is enabled by LINK (1), and MB07 (1), and set by the combination of TP3, B FETCH (1), and OPR signals.

(4) SKP (skip unconditonally). When the 7410_8 instruction is performed, MB08 = 1, and no other operand exists. The next sequential instruction is skipped regardless of the contents of the AC and Link. The SKIP flip-flop is enabled by MB08(1), MB11(0), and OP2; it is set by the combination of TP3, B FETCH (1), and OPR.

When one of the SZA, SMA, or SNL operands is combined with MB08 (1), reverse sense skipping occurs, i.e., SZA becomes SNA (skip on non-zero AC, 7450_8), SMA becomes SPA (skip on plus AC, 7510_8), and SNL becomes SZL (skip on zero Link, 7430_8).

(5) HLT (halt operation). When the 7402_8 is performed, MB10 = 1 and no other operand exists. This operation clears the RUN flip-flop,

inhibiting the generation of a MEM START level which prevents the start of another machine cycle. The computer stops after T4 time.

(6) OSR (inclusive OR between the SR and AC). When the 7404_8 instruction is performed, MB09 = 1, and no other operand exists. The result of the inclusive OR remains in the AC. The OSR operation occurs when the AC ENABLE, AC LOAD, SR ENABLE, and NO SHIFT signals are generated.

(7) CLA (clear the AC). When the 7600_8 instruction is performed MB09 = 1 and no other operand exists. The AC is loaded to all 0s (0 → AC). This instruction is identical to the OPR 1 CLA instruction with the exception of the OP 2 and MB04 levels. The CLA instruction (OPR 2) exists in order to combine with the OPR 2 microinstructions. As a result the AC can be cleared after it is sensed by one of the skip instructions, or combination between the OSR and CLA may be made resulting in the LAS (load the AC with the contents of the SR).

(8) NOP (no operation). When the 7400_8 instruction is performed, MB03 = 1, and no other operand exists. The same events described for the NOP conditions of the OPR1 instruction descriptions 1, 5, and 9 occur.

The OPR2 instructions listed above may be logically combined to perform more than one operation in a single Fetch cycle. Examples of two of the combined microinstructions are listed below; however, many other useful combinations exist.

(1) SZA CLA (7640_8). When this instruction is performed, the content of the AC is sensed. If each AC bit is a binary 0, the next instruction is skipped and the AC is cleared.

(2) SNA SZL (7470_8). When this instruction is performed the next sequential instruction is skipped if both the AC = 0 and the Link = 0.

CHAPTER 5
MAINTENANCE

This chapter contains information pertinent to preventive maintenance, corrective maintenance, and troubleshooting techniques for the PDP-8/L.

5.1 EQUIPMENT

Table 5-1 lists the equipment and relevant specifications needed for maintenance of the basic PDP-8/L. Also included is the actual equipment used by Digital Equipment Corporation field service personnel.

Table 5-1
Maintenance Equipment

Equipment	Specifications	Model or Type
Multimeter	10K ohms/volt-20K ohms/volt	Triplett Model 310
Oscilloscope	dc to 50 mc with calibrated deflection factors from 5 mV to 10V/div. Maximum horizontal sweep rate of 0.1 μs/div. Delaying sweep is desirable and dual trace is a necessity.	Tektronix Type 453
Probes	X10 with response characteristics matched to oscilloscope	Tektronix Type P6010
Clip-on current probe	2 mA/mV or 10 mA/mV	Tektronix Type P6019 with passive terminator
Recessed Probe Tip		Tektronix
Unwrapping tool		Gardner-Denver 505-244-475
Wire-Wrap tool		Gardner-Denver A-20557-29
30 gauge bit for wrap tool		Gardner-Denver 504221
Sleeve for 30 gauge bit		Gardner-Denver 500350
Spray paint		Krylon 1501 Glossy white

Table 5-1
Maintenance Equipment (cont.)

Equipment	Specifications	Model or Type
Spray paint		DEC black
Module Extender (2)		DEC No. W982
Jumper Wires		Assorted lengths affixed with 30 gauge termi-point connectors

5.2 PROGRAMS

Table 5-2 lists the Maintenance Programs supplied by DEC for ascertaining proper operation of the PDP-8/L.

Table 5-2
Maintenance Programs*

Program Name	DEC Number	Use
Instruction Test 1	MAINDEC 8I-D01B	Tests AND, TAD, and Operate Instructions only
Instruction Test 2	MAINDEC 8I-D02B	Extensive test of Auto Index, Indirect Address, and the DCA Instruction
Instruction Test 2B	MAINDEC 08-D02A	Tests 2s add and rotate logic
Random JMP Test	MAINDEC 08-D04B	Extensive test of JMP instruction
Random JMP-JMS Test	MAINDEC 08-D05B	Extensive test of JMS instruction
Random ISZ Test	MAINDEC 08-D07B	Extensive test of ISZ instruction
Memory Checkerboard	MAINDEC 08-D1J0	Tests memory circuits susceptibility to noise
Memory Address Test	MAINDEC 08-D1B0	Tests address selection logic
Memory Power on/off Test	MAINDEC 08-D1AB	Test ability to retain memory information during loss of power

* Programs are subject to change

5.3 PREVENTIVE MAINTENANCE

A systematic preventive maintenance program can be a useful deterrent against system failures. Proper application of such a program is an aid to both serviceman and user, since detection and prevention of probable failures can reduce maintenance and downtime to a minimum.

Scheduling of computer usage should always include time set aside for maintenance purposes. Careful diagnostic testing can make evident problems which may only occur intermittently during on-line operation.

We suggest weekly program checks scheduled on the following criteria:

1000-hours - electrical
500-hours - mechanical

or at least every three months.

5.3.1 Weekly Checks

Time should be scheduled each week to operate the MAINDEC programs. Run each program listed in Table 5-2 for at least five minutes. Take any corrective action necessary at this time and record the results in the log book. External cleanliness of the system should also be maintained on a weekly basis.

Many hours of computer downtime can be avoided by rigid adherence to a schedule based on the condition of the air filter. A dirty filter can cause machine failure through overheating which has a number of bad effects. The frequency of this practice depends upon system environment and usage. After several weeks, the frequency of cleansing for the particular environment will be determined. The procedure for filter cleansing is described under Preventive Maintenance Tasks.

5.3.2 Preventive Maintenance Tasks

The following tasks should be performed on at least a three-month's schedule:

a. Clean the exterior and interior of the equipment cabinet, using a vacuum cleaner and/or clean cloths moistened in nonflammable solvent.

b. Clean the air filter. Use a vacuum cleaner to remove accumulated dirt and dust.

c. Lubricate slide mechanisms and casters, with a light machine oil. Wipe off excess oil.

d. Visually inspect equipment for general condition. Repaint any scratched areas with DEC black paint or Krylon glossy white No. 1501.

e. Inspect all wiring and cables for cuts, breaks, fraying, wear, deterioration, kinks, strains, and mechanical security. Tape, solder, or replace any defective wiring or cable covering.

f. Inspect the following for mechanical security: key switches, control knobs, lamps, connectors, transformers, fans, capacitors, etc. Tighten or replace as required.

g. Inspect all module mounting panels to ensure that each module is securely seated in its connector. Remove and clean any module which may have collected excess dirt or dust.

h. Inspect power supply components for leaky capacitors, overheated resistors, etc. Replace any defective components.

i. Check the output voltage of the 718 power supply as specified in Table 5-3. Use a multimeter to make these measurements without disconnecting the load. With the exception of Memory Supply + the outputs of the supply are not adjustable; therefore, if any output voltage is not within tolerance, the supply is considered defective and corrective maintenance should be performed.

j. Run all MAINDEC programs to verify proper equipment operation. Each program should be allowed to run for at least five minutes.

k. Perform all preventive maintenance operations for each peripheral device included in the system.

l. Enter preventive maintenance results in the log book.

5.4 CORRECTIVE MAINTENANCE

The PDP-8/L is constructed of reliable TTL M-series modules. Use of these circuits with faithful performance of the preventive maintenance tasks ensure relatively little equipment downtime due to failure.

Should a malfunction occur, maintenance personnel should analyze the condition and correct it as indicated in the following procedures. Neither special test equipment nor special tools are required for corrective maintenance other than a broad-bandwidth oscilloscope, a Tektronix Type P5019 current probe, and a multimeter. The best corrective maintenance tool is a thorough understanding of the physical and electrical characteristics of the equipment. Persons responsible for maintenance should be thoroughly familiar with the system concept, the logic drawings, the operation of specific module circuits, and location of mechanical and electrical components.

It is virtually impossible to outline any specific procedures for locating faults within complex digital systems, such as the PDP-8/L. However, diagnosis and remedial action for a fault condition can be undertaken logically, and systematically in the following phases:

a. Preliminary Investigation
b. System Troubleshooting
c. Logic Troubleshooting
d. Circuit Troubleshooting
e. Repairs and Replacement
f. Validation Test
g. Recording

5.4.1 Preliminary Investigation

Before commencing troubleshooting procedures, explore every possible source of information. Think over the problem. Gather all available information from those users who have encountered the same problem and check the system log book for any previous references to the problem.

Do not attempt to troubleshoot by use of complex system programs alone. Run the MAINDEC programs and select the shortest, simplest program available which exhibits the error conditions. MAINDEC programs are carefully written to include program loops for assistance in system and logic troubleshooting.

5.4.2 System Troubleshooting

Once the problem is understood and the proper program is selected, the logical section of the system at fault should be determined. Obviously, the program which has been selected gives a reasonable idea of what section of the system is failing. However, faults in equipment which transmits or receives information, or improper connection of the system,

Table 5-3
Power Supply Specifications

Voltage	Current	Use	Pins (G785) (AB28)
+7V (rms) ± 20%	1.75A	Panel Lights	AN2, AP2,AR2,AS2
+5V, ±3%	7A	Logic	AA2,AF2,AH2,AJ2,AK2 AL2,AM2,BA2
-15V (-14V to -19V)	1.5A	Keys and Switches, Sense Amplifier Main Supply	AB2
-15V (Zener Regulated)	N/A	Sense Amp Slice Reader Clock	AV2 BB2
-30V (-28V to -38V)	1.75A	Memory Supply -, Teletype	BS2,BT2,BU2,BV2
-6V (varies)	1.7A	Memory Supply +	BM2,BN2,BP2,BR2

frequently give indications similar to those caused by computer malfunctions.

Disconnect any peripheral devices which are not necessary to operate the failing program.

Now, reduce the program to its simplest scope loop and duplicate this loop in a dissimilar portion of memory to verify, for instance, that an operation failure is not dependent upon memory location. This process can aid in distinguishing memory failures from processor failures. Use of the technique described above often pinpoints the problem to a few modules.

5.4.3 Logic Troubleshooting

Before attempting to troubleshoot the logic, make sure that proper and calibrated test equipment is available. Always calibrate the vertical preamp and probes of an oscilloscope before using. Make sure the oscilloscope has a good ac ground and keep the dc ground of the probe as short as possible.

Use the oscilloscope to trace signal flow through the suspected logic element. Oscilloscope sweep can be synchronized by control pulses or by level transitions which are available on individual module terminals at the wiring side of the logic. Care should be exercised when probing the logic, to prevent shorting between pins. Shorting of signal pins to power supply pins can result in damaged components. Within modules, unused gate inputs are held at +3 V. This voltage is introduced from pin U1 or V1 of modules M113, M117, or M617. The number in parentheses beside each +3V input represents the wiring run number for that +3V line. Each line can handle a maximum of 15 loads.

5.4.4 Circuit Troubleshooting

Engineering schematic diagrams of each module are supplied with each PDP-8/L system and should be referred to for detailed circuit information. Copies of engineering schematic diagrams are contained in Volume II.

Visually inspect the module on both the component side and the printed-wiring side to check for overheated or broken components or etch. If this inspection fails to reveal the cause of trouble or to confirm a fault condition observed, use the multimeter to measure resistances.

CAUTION

Do not use the lowest or highest resistance ranges of the multimeter when checking semiconductor devices. The X10 range is suggested. Failure to heed this warning may result in damage to components.

Measure the forward and reverse resistances of diodes. Diodes should measure approximately 20Ω forward and more than 1000Ω reverse. If readings in each direction are the same and no parallel paths exist, replace the diode.

Measure the emitter-collector, collector-base, and emitter-base resistances of transistors in both directions. Short circuits between collector and emitter or an open circuit in the base-emitter path cause most failures. A good transistor indicates an open circuit in both directions between collector and emitter. Normally 50 to 100Ω exist between the emitter and the base, or between the collector and the base in the forward direction, and an open circuit condition exists in the reverse direction. To determine forward and reverse directions, consider a transistor as two diodes connected back to back. In this analogy, PNP transistors would have both cathodes connected together to form the base, and both the emitter and collector would assume the function of an anode. In NPN transistors the base would be a common-anode connection; and both the emitter and collector, the cathode.

Multimeter polarity must be checked before measuring resistance, since many meters apply a positive voltage to the common lead when in the resistance mode. Since IC's contain complex circuits with only the input, output, and power terminals available, static multimeter testing is limited to continuity checks for shorts between terminals. IC checking is best done under dynamic conditions using a module extender to make terminals readily accessible. Using PDP-8/L engineering drawings and M-series module schematics, you may locate an IC on a circuit board as follows.

a. Hold the module with the handle in your left hand; component side facing you.

b. IC's are numbered starting at the contact end of the board; upper right hand corner.

c. The numbers increase toward the handle.

d. When a row is complete, the next IC is located in the next row at the contact end of the board. (See Figure 5-1).

e. The pins on each IC are located as Figure 5-2 illustrates.

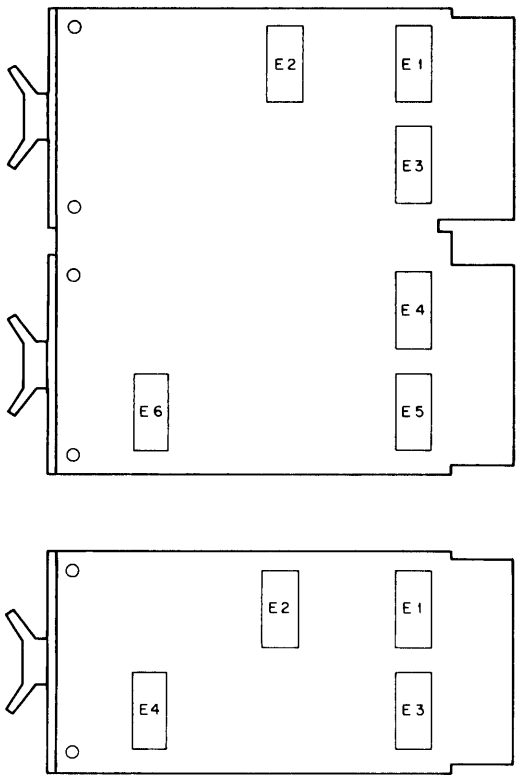


Figure 5-1 IC Location

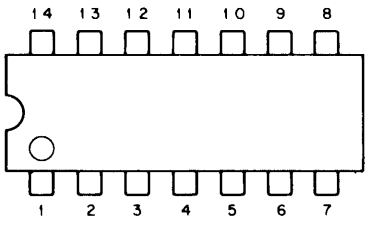


Figure 5-2 IC Pin Location

5.4.5 Repairs And Replacement

When soldering semiconductor devices (transistor, diodes, rectifiers or integrated circuits) which may be damaged by heat, physical shock, or excessive electrical current, take the following special precautions:

a. Use a heat sink, such as a pair of pliers, to grip the lead between the joint and device being soldered.

b. Use a 6V iron with an isolation transformer. Use the smallest iron adequate for the work. Use of an iron without an isolation transformer may result in excessive voltages presented at the iron tip.

c. Perform the soldering operation in the shortest possible time to prevent damage to the component and delamination of the module etched wiring.

d. IC's may be easily removed by using a solder puller to remove all excessive solder from contacts. Then, by straightening the leads, lift the IC from its terminal points. If it is not desirable to save the defective IC for test purposes; then the terminals may be cut at the IC body and each terminal removed from the board individually.

CAUTION

Never attempt to remove solder from terminal points by heating and rapping module against another surface. This practice can result in module or component damage. Always remove solder by the use of a solder-sucking tool.

When removing any part of the equipment for repair and replacement, make sure that all leads or wires which are unsoldered, or otherwise disconnected, are legibly tagged or marked for identification with their respective terminals. Replace defective component only with parts of equal or better quality, and equal tolerance.

In all soldering and unsoldering operations in the repair and replacement of parts, avoid placing excessive solder or flux on adjacent parts or service lines. When repair has been completed, remove all excess flux by washing junctions with a solvent such as trichlorethylene. Be very careful not to expose painted or plastic surfaces to this solvent.

5.4.6 Validation Tests

Always return repaired modules to the location from which they were taken. If a defective module is

replaced by a new one, while repairs are being made, tag the defective module noting the location it was taken from and the nature of the failure. When repairs are completed, return the repaired module to its original location and ascertain that the repairs have resolved the problem.

To confirm that repairs have been completed, run all tests which originally exhibited the problem. If modules have been moved during the troubleshooting period, return all modules to their original postions before running the validation tests.

Anytime that a module is replaced by one from spares to correct a problem, always return the module to its original location to confirm its defectiveness before initiating repair procedures.

5.4.7 Recording

A log book is supplied with each PDP-8/L system. Corrective maintenance is not complete until all activities are recorded in the log book. Record all data indicating the symptoms given by the fault, the method of fault detection, the component at fault, and any comments which would be helpful in maintaining the equipment in the future. The log should be maintained on a daily basis, recording all operator usage and preventive maintenance results.

5.5 ADJUSTMENTS

Note: The measurements and adjustments in the following paragraphs are analog in nature, and are not of the on-off, high-low nature. These areas represent tuned sections of the computer that must be properly aligned by qualified personnel.

Adjustments of the PDP-8/L should never be undertaken until it has been confirmed that a failure is due to circuit aging or misalignment rather than a component failure. Replacement of certain components or excessive environmental handling may preclude any other corrective action.

5.5.1 Power-Up Threshold Adjustment

This adjustment is preset at the factory and should only be attempted in the field by skilled personnel.

The G826 Negative Regulator Control module contains a difference amplifier that compares the +5V supply voltage with an adjustable reference threshold voltage. The threshold reference is properly set when the voltage at test point A27U2 changes from a low voltage (-6V) to a higher voltage (-2V) as the +5V supply voltage passes through 4.75V after turn-on. The threshold voltage is adjusted by the G826 variable resistor R2, 1000Ω Minnelco, and performs several functions. When the +5V supply voltage is less than this threshold, the regulated memory supply voltage is held off. A comparison amplifier on the G785 module, continually monitors the raw power supply input to the +5 regulator and anticipates any voltage decrease. When a change occurs, the LINE LOW level is asserted. LINE LOW controls various logic signals to the central processor by initiating POWER OK and a power failure action SHUT DOWN; a STOP OK level from the processor allows a shut down of the memory supply voltage when the +5V supply voltage fails. As the threshold is reached during turn-on, POWER CLEAR produces a pulse to ground and then returns to a high level. Also, with these conditions, POWER OK is a low logic level, and SHUT DOWN is a high logic level. The inverse of these levels should exist before the threshold is reached. The STOP OK level originates from the central processor and its undriven input should remain high having no effect during alignment.

5.5.2 Memory Alignment Procedure

To adjust or check the memory currents, the PDP-8/L should be allowed to warm up for approximately 1 hr before measurements are made. In addition, the measurements should be performed at an ambient temperature of 25°C.

The G826 negative regulator control module at location AB27 and the negative regulator portion of the 718 power supply control the memory voltage regulation, and therefore control the memory currents.

The voltage difference between MEMORY SUPPLY+ and MEMORY SUPPLY - provides the read/write, and inhibit currents through the memory stack.

The negative regulator control (G826) serves to vary the 718 regulator for temperature variations by a thermistor and difference amplifier tracking circuit which compares the regulated memory voltage to an adjustable reference. A Trimpot (R28, 2000Ω Bourns) located on the G826 module varies this reference voltage, and therefore, varies the memory currents.

Adjustment of the trimpot varies the MEMORY SUPPLY+ voltage between approximately -1V and

-12V; the subsequent variation of the regulated memory supply voltage should be between -18V and -29V. Normally, the regulated memory voltage should be set to approximately -22.5V, measured with a multimeter across memory supply+ and memory supply-.

If the voltage can be adjusted but not to the above value, the thermistor across the regulator control (pins B27R2 and B27S2) should be investigated. The thermistor is located within the memory stack and outputs on the inhibit connector card (pins B22S2 and B22T2); its resistance should be 330Ω±10% at 25°C. Shunting resistance in the regulator control would reduce this value if a measurement is made with the modules connected.

If no voltage adjustment is possible, either the Negative Regulator (G785) portion of the 718 power supply or the Regulator Control (G826) can be at fault.

In alignment of the memory, the actual outputs of the various memory control flip-flops should be checked against those of Figure 5-3. An approximate initial STROBE adjustment can be such that its leading edge occurs 500 ns after the leading edge of MEM START. The width of the strobe signal should be less than or equal to 80 ns and have an approximate adjustment range for its leading edge from MEM START from 350 ns to 650 ns. A clockwise rotation of the adjustment on the variable delay line (M360 at C17) increases this delay. The final adjustment of strobe must be made in relation to the analog data signal from core memory.

The R/W selector switches are examined by inspecting the current wave forms on the current loops at the source and return signals. The current waveturns are similar to those of Figure 5-4 which represent the equal amplitude read/write currents measured on the memory stack input. When running a multiple-selection program such as MEMORY CHECKERBOARD, the wave forms differ in the amplitudes of the read/write currents due to the contribution of the base currents to the emitter currents of the address decoding and selection switches.

The additional current (approximately 30 mA) appears on the return current through the R/W selection switches to MEM SUPPLY-. Depending upon the loop examined, it is sometimes additional read current and sometimes additional write current. This difference in current should not be confused with amplitude variations within the read/write current due to bad address decoding selection switches; these differences are also apparent at the memory stack inputs and vary with different addresses. The current loops, provided for use with a Tektronix Type P6019 current probe, are as follows:

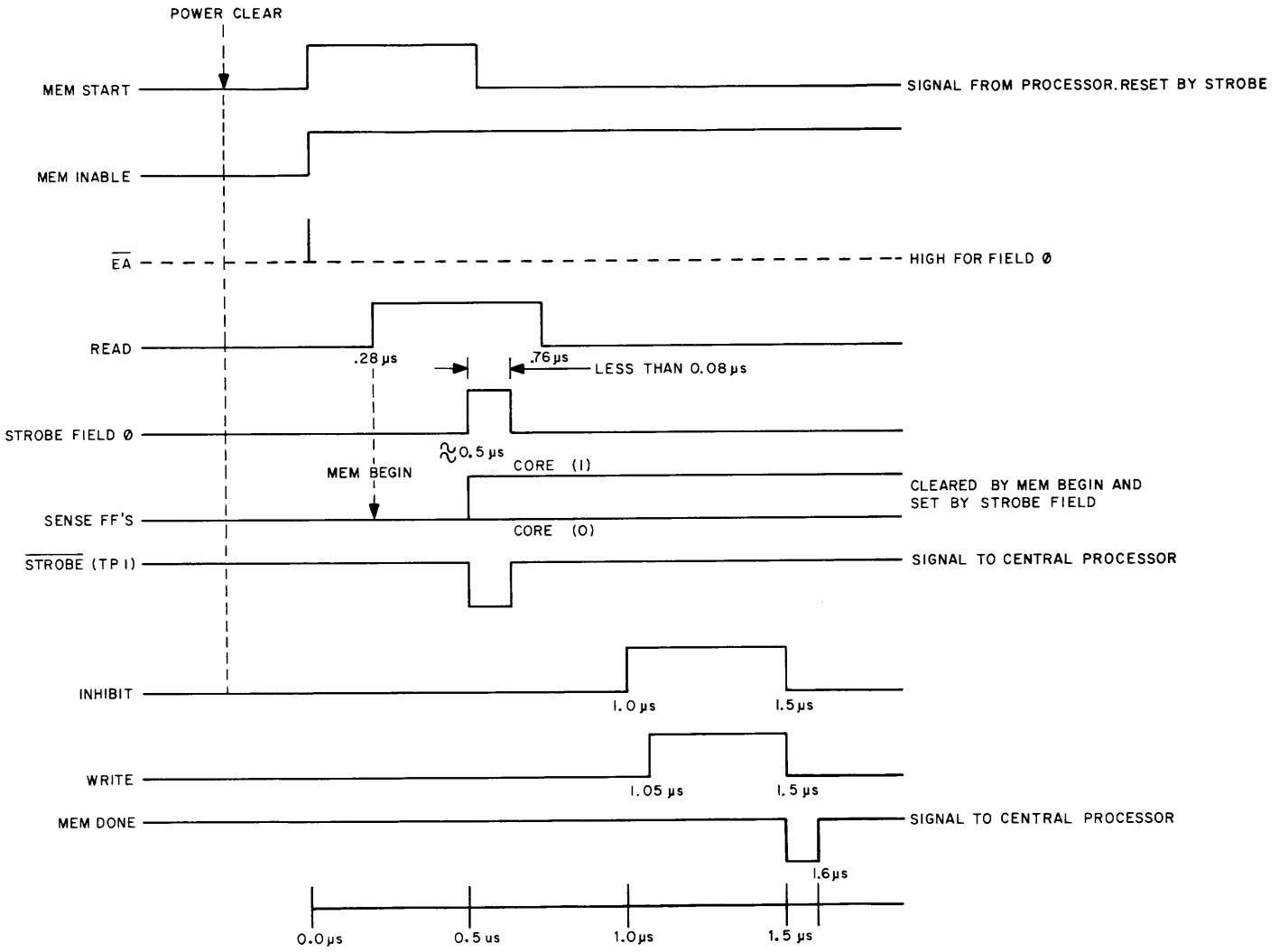
X Source	C23T2 to C25K1
Y Source	C18T2 to C2551
X Return	D19T2 to D25K1
Y Return	D23T2 to D2551

Examination of the current at these loops has the advantage that the currents for all memory addresses pass through these common points. The waveshape should be inspected on each of the four loops to check that no leakage to ground exists between the voltages, MEMORY SUPPLY+ and MEMORY SUPPLY-.

The regulated memory voltage was previously adjusted in a static condition for -22.5V; there should be no change although the memory is now cycling. The resultant stack current should be approximately 320 mA (this value varies with stack vendors, and is best determined by adequate core output with minimum noise). The lower read/write current amplitudes on the current amplitudes on the current loops equal the current amplitudes in the stack windings; the current measurements can conveniently be made there. Correlation between the voltage and current should exist. Variations in current amplitude at successive addresses should be less than 20 mA. Failure of the R/W selection switches is indicated by the lack of proper read/write currents of any address. If all the R/W selection switches appear to have failed, inspection should be made at the input logic signals, READ (1) and WRITE (1), and the input supply voltages. To analyze the read/write currents for individual addresses, the following program may be used:

```
0000    Beg, LAS        7604
0001         DCA Temp   3004
0002         TAD I Temp 1404
0003         JMP Beg    5000
0004    Temp, 0         0000
```

The read/write currents for individual addresses may be examined by setting the desired address into the switch register. Of course, since the currents for all memory addresses pass through the common test point, you will also be examining the currents for those locations which the test program occupies. You must first ascertain that the currents are proper for those addresses within the program; if not, relocate



NOTE: TRANSITION TIMES ARE MEASURED FROM THE POSITIVE TRANSITION OF MEM START AND ARE APPROXIMATE.

Figure 5-3 Memory Control Waveforms

the program to some other area of memory. By selecting individual addresses via the switch register, waveform deformities may be traced to defective associated R/W switches.

If the improper waveform remains fixed to a specific address and replacement of the associated R/W switch does not correct the problem; then, the logic signal inputs from the memory address (MA) register should be inspected. If those signals are correct and vary according to the selected address, attention should be turned to the memory stack with its attendant diode selection matrices.

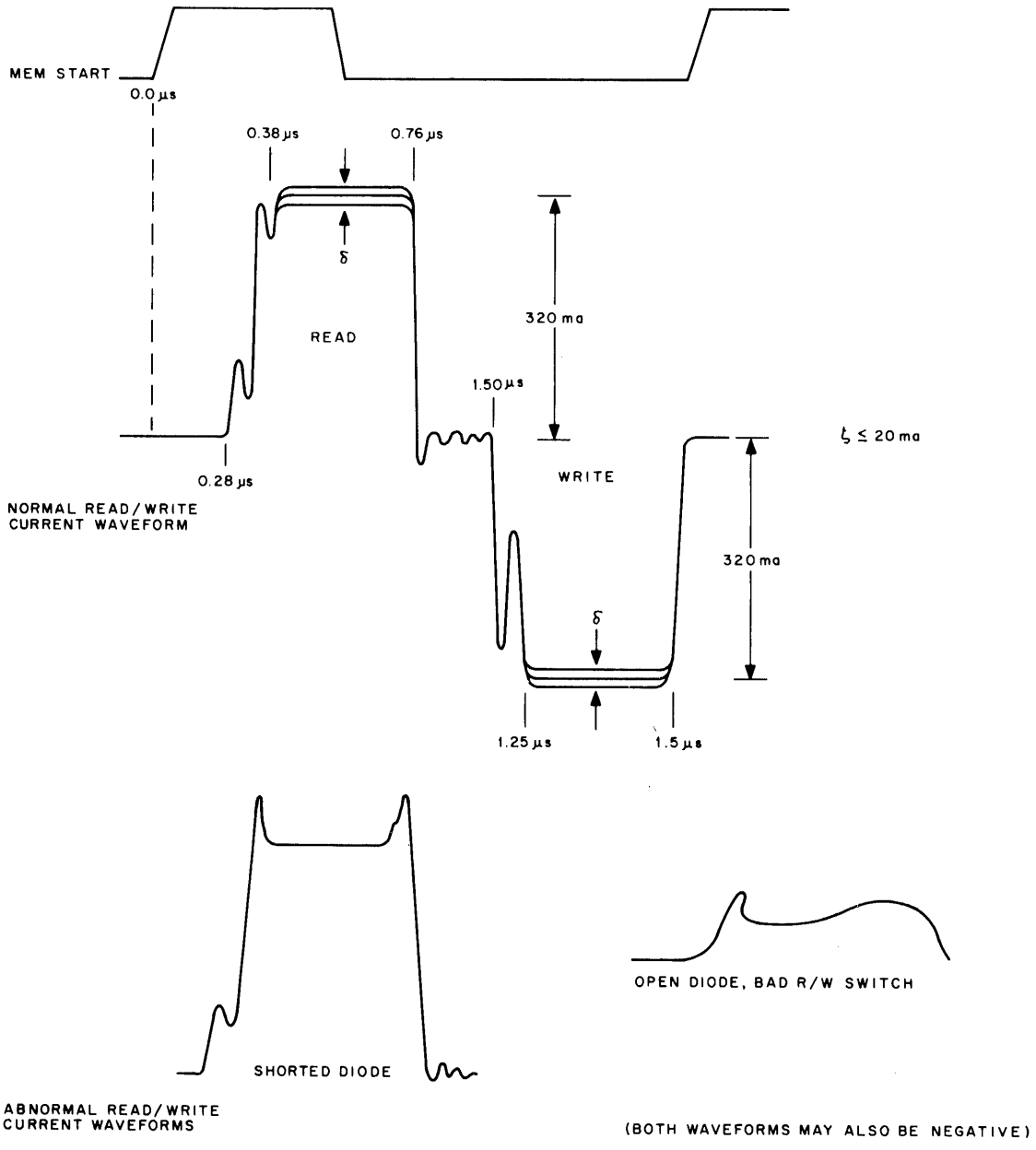


Figure 5-4 Representative Read/Write Current Waveforms

The diode selection matrices (G611 and G610) sandwich the memory planes of ferrite cores between them. Also connected to this unit are two W025 cable connectors, one for the inhibit inputs, the other for the sense outputs.

The X-axis Diode Selection Matrix (Drawing D-BS-8L-0-15) has half its diodes on the G610 board and half on the G611 board; the same is true of the Y-axis Diode Selection Matrix (Drawing D-BS-8L-0-16). The inductor symbol connecting the centers of the two sets of diodes represents the stack winding traversing the 12 core planes. The windings are identified on the diode matrix selection boards as X, 0-64 and $\overline{X}$, 0-64 for the X-axis windings and Y, 0-64 and $\overline{Y}$, 0-64 for the Y-axis windings. Suspected opens and shorts in the windings, detected during dynamic tests, should be verified by measurements across these points with an ohmeter. The resistance of the read/write winding is 3.5Ω ±10%. The forward and reverse resistance of the diodes in the matrix should also be checked when address selection failures are attributed to stack failures.

CAUTION

The memory stack is expensive and fragile; it is easily damaged and must be handled with care.

Twelve inhibit drivers are associated with each memory stack. To inspect the inhibit currents you will need two module extender boards (W982) at AB22. These extender boards provide current loops for each of the 12 inhibit lines. Inhibit current should be inspected and compared with the waveform in Figure 5-5. Inhibit current amplitude is approximately 290 mA as noted; more important, however, is the ratio of read/write and inhibit currents. This ratio is 0.85 of the read/write current and should exist regardless of the read/write current amplitude. Failure of a specific inhibit driver can be determined by movement of the suspected driver to another location. Movement of the failure indicated that the module should be repaired or replaced. No movement of the failure indicates that either the input signals or output load is causing the failure. The logic inputs, B INHIBIT and B MEM ENABLE, from the memory control; the connection through the current limiting resistor; and the memory buffer signals should be checked. If the stack is suspected, the specific winding should be measured for a resistance of 14Ω ±10%.

Twelve 6020 sense amplifiers are associated with each memory stack; each amplifier transforms the analog pulse output of ferrite core to a usable logic level. The sense windings for each bit enter a differential amplifier with a threshold voltage established as a function of the fixed SLICE voltage. The test points (pins E1 and K2) after the amplifiers allow observation of the waveforms for comparison with those of Figure 5-6. The preliminary setting of STROBE should now be modified as a function of the data output from core memory.

The leading edge of $\overline{\text{STROBE}}$ is set approximately at the center or just past the midpoint of the amplifier "one" output. This adjustment should be late enough to sense all "one" data with normal variations in delay, and yet centered in the "zero" data output. Each sense amplifier test point should be examined and the final adjustment of the strobe signal should be made using the most sensitive sense amplifier as a criteria.

The lack of proper waveforms at all addresses indicates that the sense amplifier or the core winding is in error. The sense amplifier may be checked by exchanging a known good amplifier with the suspected one. The absence of an input signal indicates the stack sense winding should be checked or the sense connector (W025 at AB21) for a resistance of 21Ω ±10%.

Testing and adjustment of the memory section of the PDP-8/L is now completed by running the memory address test and memory checkerboard test (worst pattern). Final adjustment of the read/write currents may be necessary to increase or decrease the amplitude of the core input and corresponding noise.

5.6 ASR33 TELEPRINTER AND CONTROL MAINTENANCE

This section contains information pertinent to the maintenance of the ASR33 and its associated control logic.

5.6.1 Equipment

Table 5-4 lists the special tools needed for maintenance of the ASR33 Teleprinter. All of these items can be obtained from Digital Equipment Corporation or from the Teletype Corporation.

5.6.2 Programs

Table 5-5 lists the maintenance programs supplied by DEC for aid in maintaining the ASR33 and associated control logic.

5.6.3 Preventive Maintenance

Teletype preventive maintenance is scheduled on the same frequency as discussed in Paragraph 5.4.

CAUTION

Do not use alcohol, mineral spirits, or other solvents to clean plastic parts with protective decorative finishes. Normally, a soft, dry cloth should be used to remove dust, oil, grease, or otherwise clean parts or subassemblies.

To clean plastic surfaces, we recommend using any of several household cleaner-waxer liquids such as

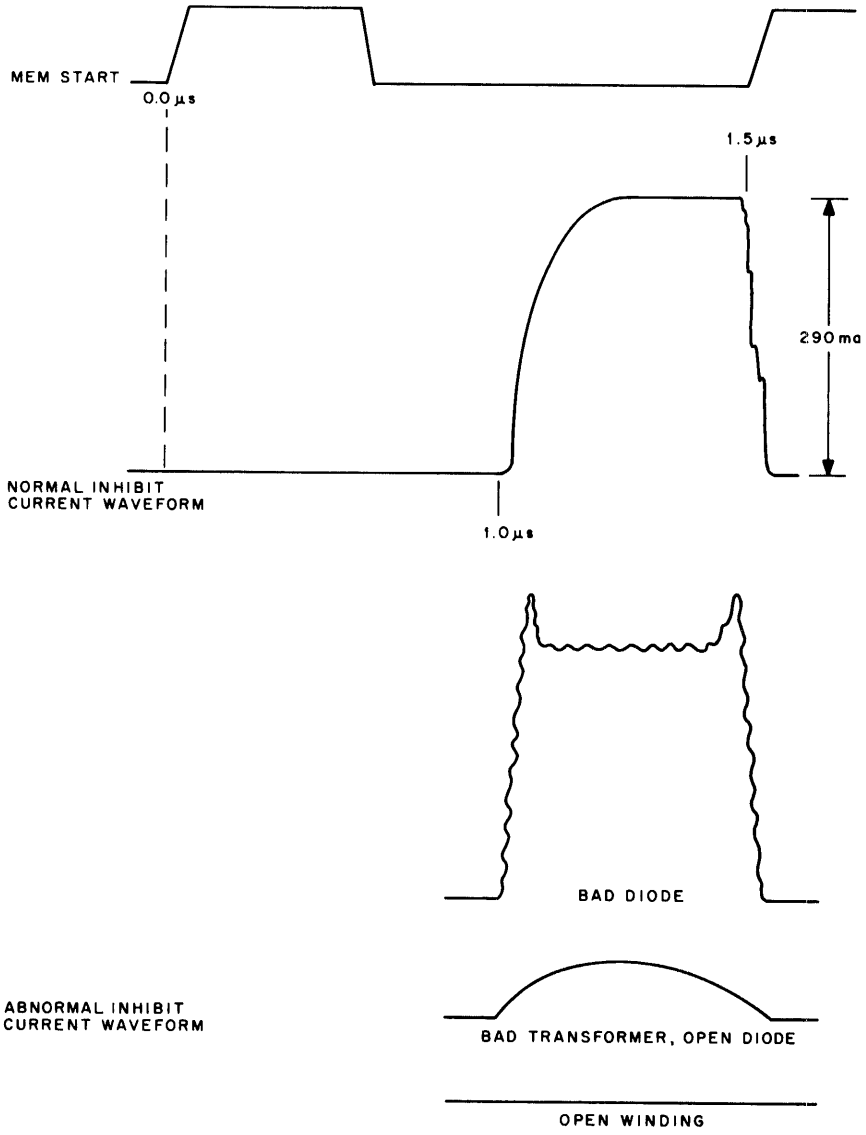


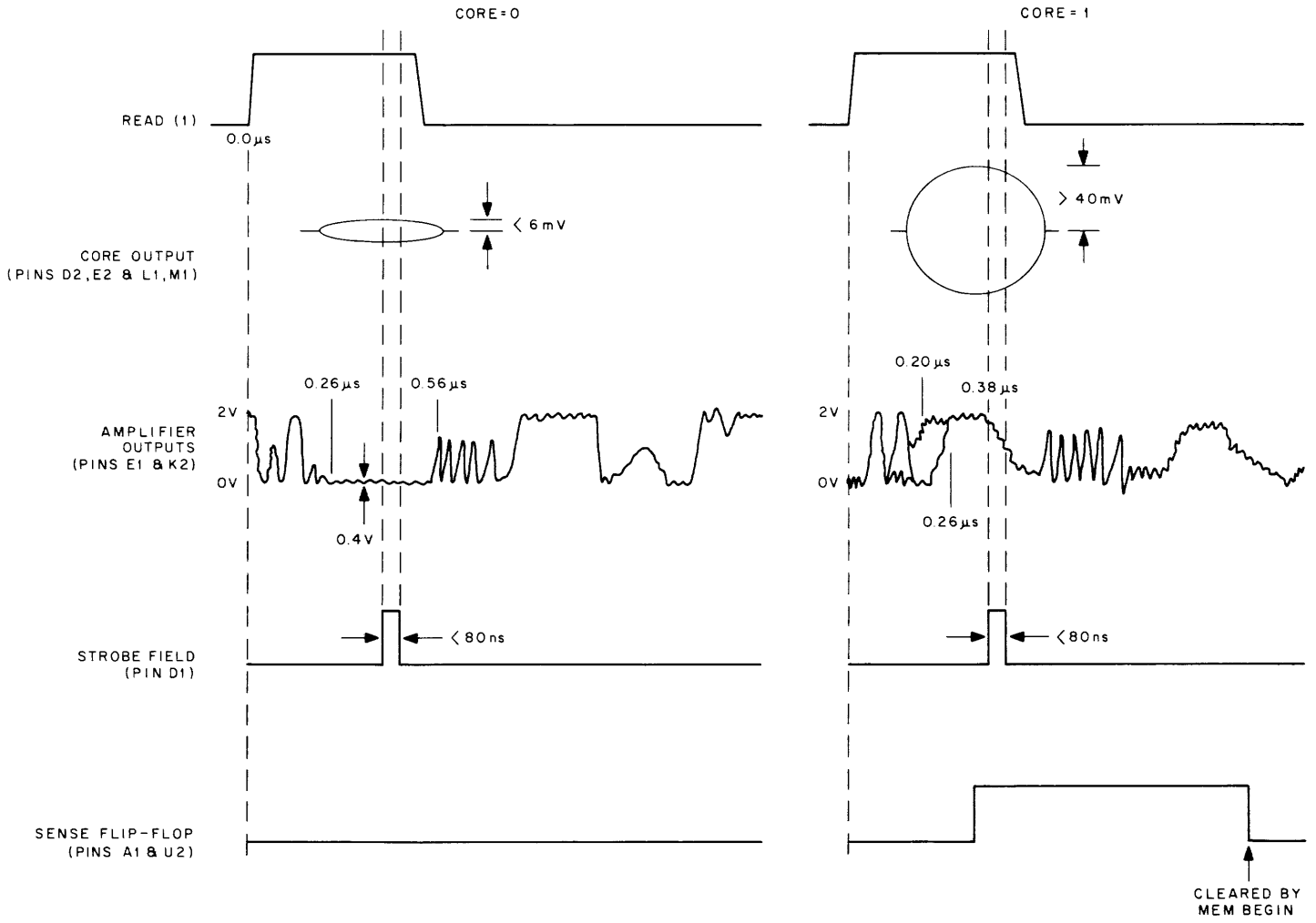
Figure 5-5 Representative Inhibit Current Waveforms

Table 5-4
Teletype Maintenance Tools

Item	Part No.	Item	Part No.
8 oz. scale	110443	Bending Tool	180993
32 oz. scale	110444	Extractor	182697
64 oz. scale	82711	Tweezer	151392
Set of gauges	117781	Spring hook (push)	142555
Offset screwdriver	94644	Spring hook (pull)	142554
Offset screwdriver	94645	Screw holder	151384
Handwheel	161430		
Handwheel adaptor	181465		
Contact adjustment tool	172060		
Gauge	180587		
Gauge	180588		
Gauge	183103		

Table 5-5
Teleprinter Maintenance Programs

Program Name	DEC No.	Use
Reader Test	MAINDEC-8I-D2PB	Function test and exerciser for ASR33/35 teletype paper tape reader
Punch Test	MAINDEC-8I-D2QB	Function test and exerciser for ASR33/35 teletype paper tape punch
Keyboard Test	MAINDEC-8I-D2RB	Function test and exerciser for ASR33/35 teletype keyboard
Combination Test	MAINDEC-8I-D2TB	Exerciser program used to test ASR33/35 printer and punch simultaneously



NOTE: CORE OUTPUTS ARE OBSERVED DIFFERENTIALLY AT THE PINS NOTED.

Figure 5-6 Representative Sense Amplifier Waveforms

"Jubilee" or "Jato." To clean the printer platen, we recommend using a lacquer thinner.

During overhaul, subassemblies and metal parts can be cleaned in a bath of trichlorethylene. Proper lubrication should be performed often.

Weekly Tasks

a. Inspect platen and paper guides. Wipe clean, using a soft, dry cloth.

b. Clean external areas of paper tape punch and reader, using a soft brush or cloth.

c. Remove and empty paper tape punch chad box.

d. Run teleprinter combination test (MAINDEC-8I-D2TB) for approximately 15 min.

Preventive Maintenance Tasks

a. Inspect platen and paper guides. Clean platen, using a lacquer thinner to remove shiny surfaces.

b. Clean ribbon guides and replace ribbon, if necessary.

c. Remove cover and check for vibration effects; loose nuts, screws, retaining clips, etc.

d. Remove distributor rotor and clean disk surface, using cotton swab moistened in "Freon" or "Trichlorethylene."

e. Check selector magnet coil for signs of overheating.

f. Clean between selector magnet pole piece and armature with bond paper to remove any lubricant or dirt.

g. Clean and lubricate Teletype as instructed in Teletype Bulletin 273B. Follow instructions literally so as not to over lubricate.

h. The following adjustments should be checked. Pages indicated are in Bulletin 273B, Volume II.

Trip Shaft	574-122-700 Page 13
Trip Lever	574-122-700 Page 14
Brush Holder (Distributor)	574-122-700 Page 15
Clutches	574-122-700 Pages 16-24
Code Bar Reset	574-122-700 Pages 30-34
Print Suppression	574-122-700 Page 35
Blocking Levers	574-122-700 Page 37
Print Suppression	574-122-700 Page 43
Carriage Drive Bail	574-122-700 Page 44
Print Trip Lever	574-122-700 Pages 61-62
Dashpot	574-122-700 Page 78
Final Printing Alignment	574-122-700 Page 85
Line Feed	574-122-700 Pages 89-95
Keyboard Trip Lever	574-122-700 Page 141
Reader Trip Lever	574-124-700 Pages 6-9
Detent Lever	574-124-700 Page 10
Sensing Pin	574-124-700 Page 15
Tape Lid Latch Handle	574-124-700 Page 18
Feed Pawl	574-125-700 Page 11
Registration	574-125-700 Page 12

i. Run each of the Teletype MAINDEC Programs for at least two passes each.

j. Check that tape holes are being punched cleanly.

5.6.4 Corrective Maintenance

Figure 5-7 is a simplified drawing of the control circuits for the ASR33 Teleprinter. Details of the cable connector are included to show how a teleprinter is modified to operate with the PDP-8/L. During off-line operation, the keyboard distributor effectively drives the printer selector magnet. This means that any character received from the keyboard or paper tape reader is automatically reproduced on the printer and paper tape punch. During on-line operation, this continuity is broken and a teletype receiver (M706) is used to accept the input from the reader or keyboard while a teletype transmitter (M707) is used to drive the printer and paper tape punch. The clock (M452) develops a TTI clock (880 Hz) and a TTO clock (220 Hz).

These clocks are used to shift the bits through the transmitter and receiver buffers. Adjustment is made by viewing the TTO clock output with the oscilloscope probe on C33K2 and adjusting the trimpot for a 4.5 to 4.6 ms repetition rate. Most teletype problems can be traced to one of four areas as follows.

a. ASR33 keyboard or reader
b. ASR33 printer or punch
c. M706 receiver
d. M707 transmitter

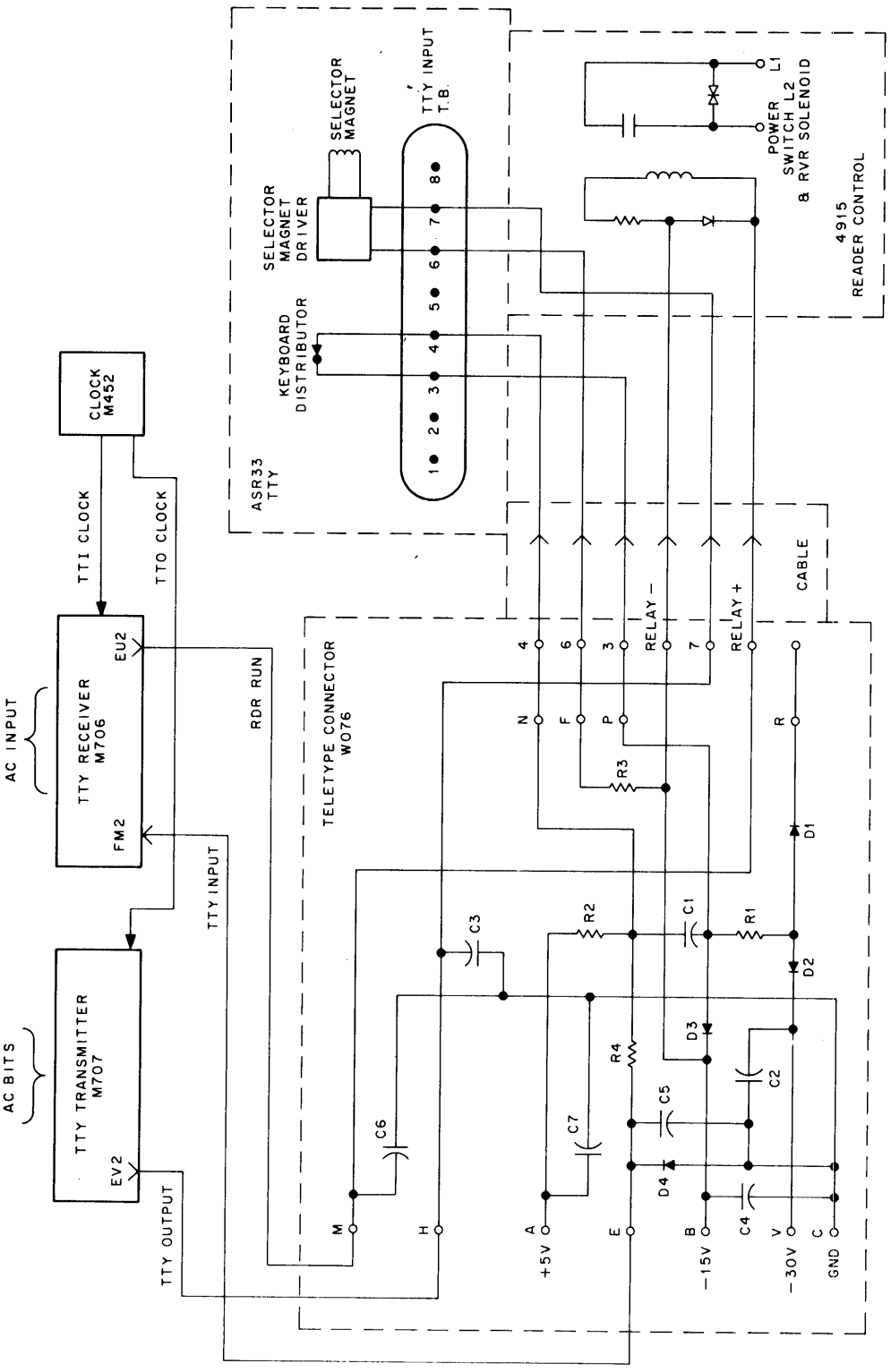


Figure 5-7 Teletype Connections