Digital Equipment Corporation
Maynard, Massachusetts

MASTER

digital

PDP-12 maintenance manual volume I

DEC-12-HR1B-D

PDP-12
maintenance manual
volume I

1st Edition May 1971
2nd Printing (Rev) March 1972
3rd Printing November 1972

CONTENTS

CONTENTS (Cont)

CONTENTS (Cont)

CONTENTS (Cont)

CONTENTS

ILLUSTRATIONS

ILLUSTRATIONS (Cont)

ILLUSTRATIONS (Cont)

TABLES

FOREWORD

The *PDP-12 Maintenance Manual,* published in four separate volumes, is a guide for Field Service Engineers or other personnel involved with the care and maintenance of the PDP-12 Computer. The Maintenance Manual is organized as follows:

VOLUME I PRINCIPLES OF OPERATION

This volume contains a description of PDP-12 logic. An overall view of the system is presented in seven chapters entitled Central Processor, Memory, I/O Bus, Teletype, LINC Devices, LINCtape Control System, and Prewired Options. The text describes logical relationships among the various elements of the PDP-12.

VOLUME II INSTALLATION AND MAINTENANCE

The first chapters of this volume describe the unpacking, installation, and preliminary check-out procedures for the PDP-12. The remainder of the volume comprises procedures used in the day-to-day maintenance, adjustment, and repair of the computer.

VOLUME III SYSTEM DRAWINGS

Volume III consists primarily of flow charts and block schematics that describe the PDP-12. The block schematics, lists, and flow charts in Volume III are reduced (11 in. x 17 in.) versions of engineering drawings.

VOLUME IV MODULE SCHEMATICS

The circuit schematics in Volume IV describe all the module types used in the PDP-12 and its options, including both the regular production DEC modules and those designed especially for the PDP-12.

CHAPTER 1
CENTRAL PROCESSOR

1.1 INTRODUCTION

The PDP-12 (Programmed Data Processor-12) is a versatile digital computer that makes use of two distinct operating modes within its single Central Processor (CP); each operating mode has its own complete instruction set. With this feature, the PDP-12 is both a laboratory-oriented machine with built-in facilities for I/O, auxiliary storage, and control and sensing of external equipment; and a general-purpose computer with flexible I/O capability, to which numerous peripheral devices may be easily attached. The logic is fully parallel, using a basic word length of 12 bits. The processor cycle time is 1.6 μs ±20 percent; most instructions require from one to three cycles for execution.

NOTE

In the following discussions of the CP, *System Drawings Volume III* of this manual is referenced often; the reader should refer to the system drawings as they are referenced. The system drawing numbers are arranged in alphabetical order by a three-letter designation. The complete system drawing numbers for the logic are D-BS-EP12-0-CIN through D-BS-EP12-0-SLA. The system drawing numbers always contain D-BS-EP12-0; thus, only the last three letters are used to reference system drawings in the text.

1.2 PDP-12 FUNCTIONAL ELEMENTS OVERVIEW

The functional elements of the PDP-12 are grouped in three categories (see Figure 1-1): Central Processor, Memory, and Input/Output. Principal data and control paths are shown in the figure. Related functions are grouped within boxes, accompanied by the three-letter reference designation to the appropriate system drawing number in Volume III. For example: PDP-12 operations are initiated from the console switches, the circuit elements of which are shown in block schematic CSI. Inputs from the switches can be sent directly to the CP shift gates (dwgs. *PRA* through *PRF*) or to the console timing logic (dwg. *CST*). This reference system relates every element in this figure to one or more of the block schematics.

1.2.1 CP Functional Elements

Functional elements of the CP are grouped as follows:

a. Console Switches and Indicators
b. Console and CP Timing Logic
c. Instruction Decoding and Register Control
d. Active Registers and Processor Register Bus
e. I/O Control and Buffers and Teletype ® Circuits

® Teletype is the registered trademark of Teletype Corporation, Skokie, Illinois.

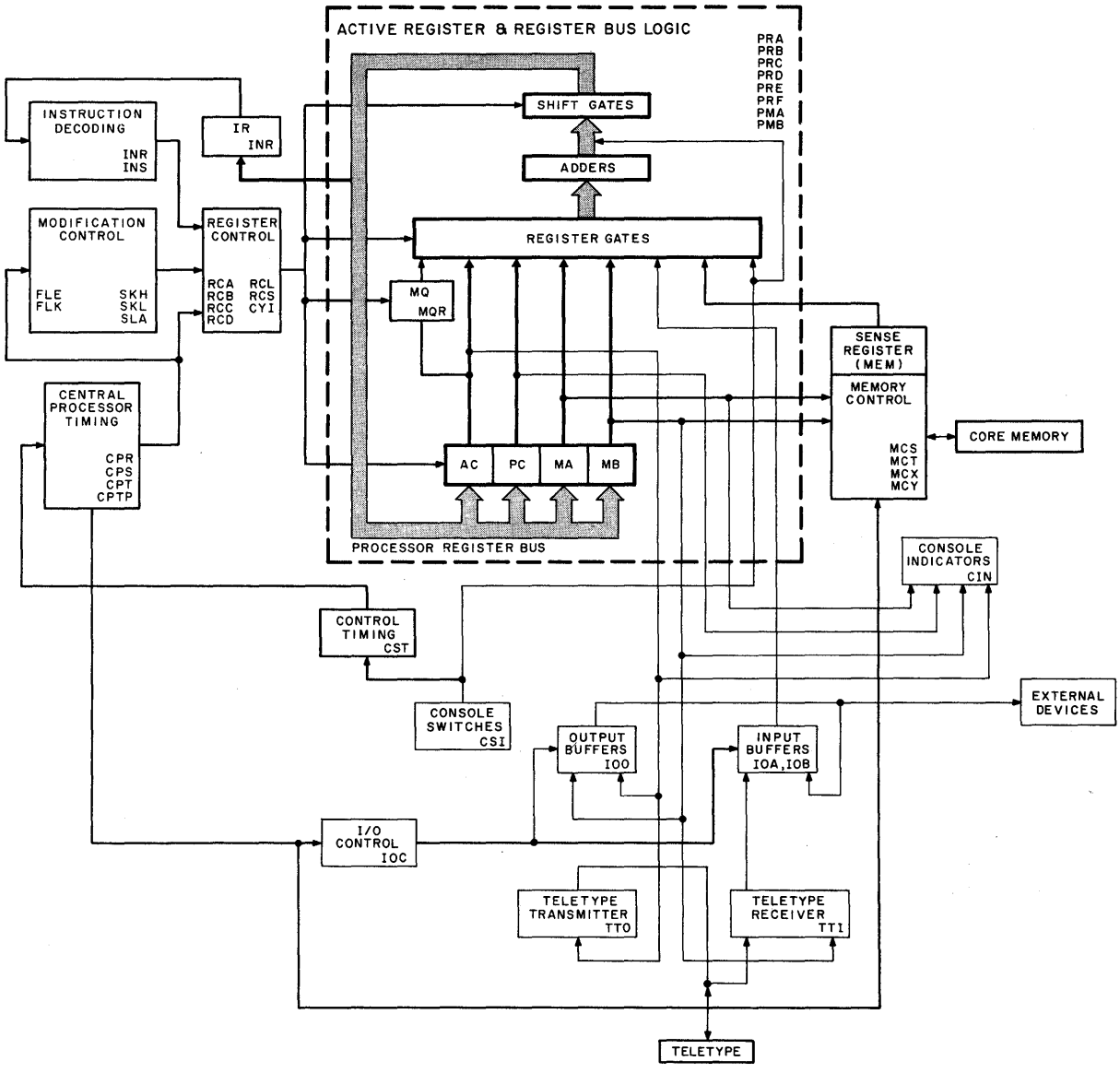


Figure 1-1 Central Processor Overview

Console Switches and Indicators – These switches on the console front panel

a. Provide data inputs to the processsor register bus logic, via the register gates and the shift gates

b. Control inputs that start the machine by initiating the console manual timing chain, which in turn starts the CP timing cycle.

The console indicators usually represent the contents of all active registers and the states of most of the control flip-flops in the PDP-12. These indicators can provide valuable assistance in troubleshooting.

Console and CP Timing Logic – Console and CP timing logic control the sequence of CP operations. Signals from the console switches initiate a sequence of manual-function time states and pulses. The final console time pulse starts CP timing, which continues until the processor is stopped. CP timing controls the start of the memory Read-Write cycle and synchronizes the transmission of data between the I/O buffers and the CP.

Instruction Decoding and Register Control – The contents of the instruction register (IR) are used to establish signal levels and set flip-flops that ultimately (through the various elements of the register control circuits) determine the paths of data into, through, and out of the active registers and the processor register bus logic.

Active Registers and Processor Register Bus – These circuits are the vital components of the PDP-12 Central Processor. The active registers (AC, PC, MA, IR, and MB) determine the locations in which the program and data are stored in memory and store the data. Data can be sent from these registers to various other elements of the system (e.g., the I/O buffers or memory control), but data can enter the active registers only through the processor register bus. This facility is illustrated in Figure 1-1. Information paths go from the active registers into the register gates, then to the adders, and finally to the shift gates. From the shift gates, data enters any of the active registers via the processor register bus.

I/O Control and Buffers; Teletype Logic – Information leaving or entering the I/O Bus must pass through the I/O buffers. The I/O Control synchronizes the transmission of data between the buffers and the processor register bus, under the control of CP timing.

Data that pass between the Teletype and the CP have a separate path and control: the Teletype transmitter and receiver. Incoming data pass through the Teletype receiver to the internal I/O Bus; outgoing data move from the AC, through the Teletype transmitter, directly to the device.

1.2.2 Core Memory and Memory Control

The transmission of data between core memory and the CP is controlled by the memory Read-Write cycle. This cycle is initiated by CP timing; information is strobed from memory into a buffer (dwg. *MEM*) located in the memory control and passes to the register bus logic. Later in the Read-Write cycle, the contents of the CP memory buffer condition the inhibit drivers of the memory control to determine the pattern that is written back into the cores.

1.2.3 Input/Output

I/O devices are divided into two main categories: those that are connected to the CP *through* the I/O Bus and those that communicate *directly* with the CP. The latter group includes the LINCtape, display, A/D converters, and relays. Each of these devices has its own control and its own distinct data paths.

Two types of devices are connected to the I/O Bus:

a. The pre-wired bus options, including real-time clocks, incremental plotter, extended arithmetic element (EAE), power fail restart, and additional Teletypes of Dataphones.®

®Dataphone is a registered trademark of the Bell Systems.

b. Other devices that are not pre-wired options, which can be added using the BA12 Peripheral Expander; these devices include the line printer, high-speed paper-tape reader and punch, card reader, etc.

1.3 DATA FLOW

1.3.1 Internal Flow

The principal paths of data flow are shown in Figure 1-2. At the center of this flow is the cyclic path from the active registers (*SOURCE*) through the enable gates, adders, and shifters onto the processor register bus (*ROUTE*), and back to the active registers (*DESTINATION*). Logical and arithmetic operations, shifts, and internal data transfers are all performed as the data are circulated.

An instruction is fetched from core memory and placed in the IR. The contents of the IR are interpreted, using binary-to-octal decoders (dwg. *INS*) and other logic (dwg. *SLA, SKL, FLE*), to provide signals that determine the events that are to occur as data move through the processor logic. The paths from the active registers or external sources to the enable gates are established by the register control logic (dwg. *RCA* through *RCD*), according to the signals that result from the decoded instruction. The enable gates (dwgs. *PMA, PMB, PRA* through *PRF*), which are multiple-input NAND/NOR gates, combine the enabled inputs and send the outputs to the adders, where they are combined with the Carry Insert input (dwg. *CYI*) to provide both a Sum Output and a Carry Output to the next higher-order adder. A simplified schematic of the adder (see Figure 1-3) illustrates this process. The sum output is routed to the Shift gates (dwgs. *PRA* through *PRF*), where, depending on signals from the shift control logic (dwgs. *RCS, FLE*), the output is passed through directly or shifted right or left. The Register Load Control (dwg. *RCL*) enables inputs to the specified active registers that receive the shifter outputs. *The key to understanding the internal flow is:* information can enter an active register (AC, PC, MA, MB, and IR) only through the register bus. Outputs from these registers may go anywhere: I/O buffers, Teletype, memory, MQ, etc. The memory field registers and buffers, however, are treated differently.

NOTE

The MQ is not part of the register bus logic; information enters it only from the AC.

1.3.2 Major Registers

The major registers are listed and described in Table 1-1.

Table 1-1
Major Registers

Major Register	Description
Accumulator (AC) 12 Bits	This register contains data being operated upon. Its contents can be shifted or rotated right or left; incremented, cleared, or complemented; stored in memory or added to the contents of a memory register; and logically or arithmetically compared with the contents of any memory register. The AC holds the sum after an addition and part of the product after a multiplication. The AC is also involved in the transfer of data to and from various other registers outside the CP.
Link (L) 1 Bit	The Link is an extension of the AC. When a Carry occurs out of AC00 during a 2's complement addition, the Link is complemented. It may be set or cleared independently of the AC under 8-Mode control and may or may not be included in shifting and rotating operations performed on the contents of the AC.
Program Counter (PC) 12 Bits	This register contains the address of the next instruction to be executed within the memory field selected by the Instruction

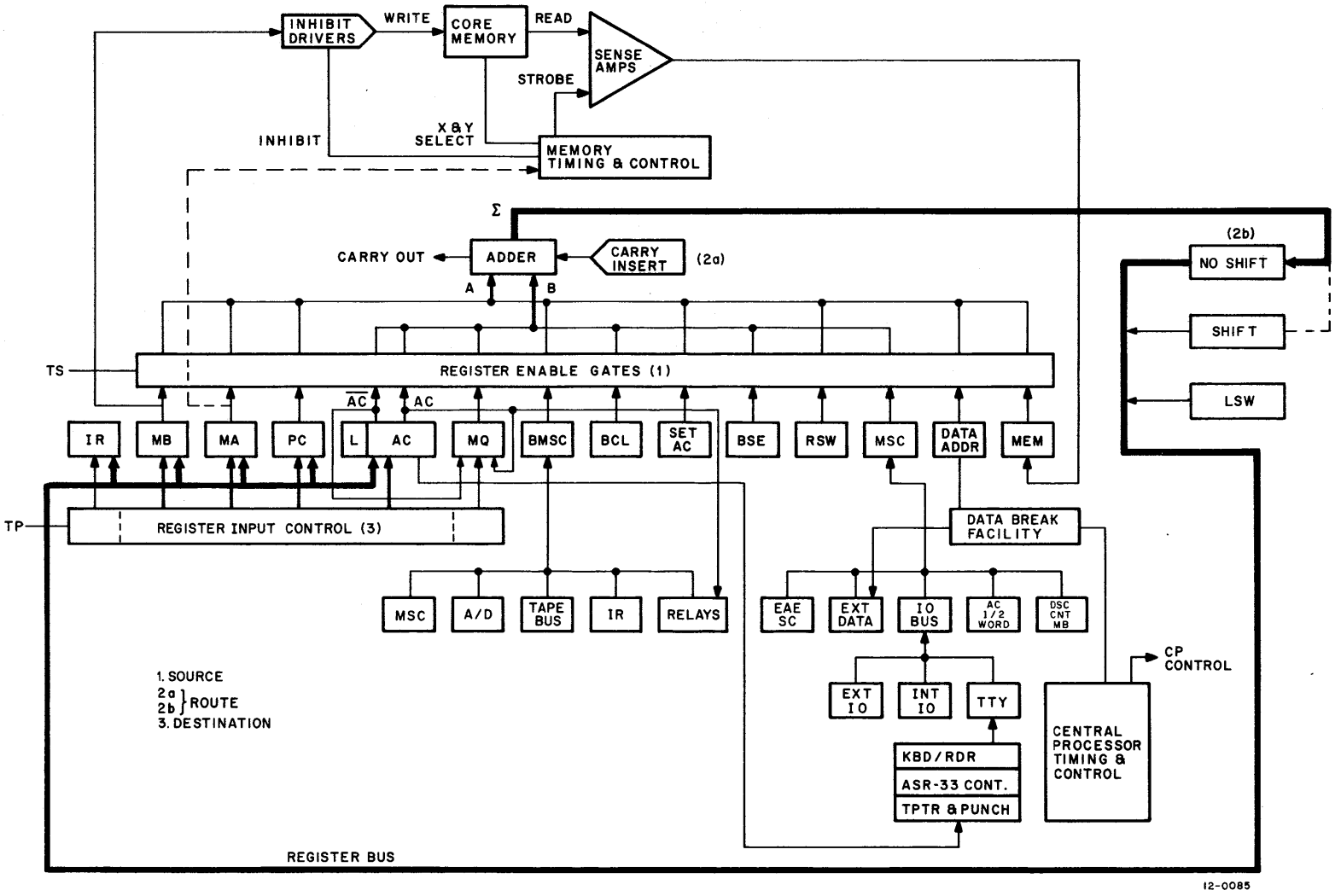


Figure 1-2 PDP-12 Data Flow

Table 1-1 (Cont)
Major Registers

Major Register	Description
Program Counter (PC) 12 Bits (Cont)	Field Register. In 8 Mode, the PC is used as a 12-bit counter; in LINC Mode, it is used as a 10-bit counter.
Instruction Register (IR) 12 Bits	This register contains the complete binary code of the instruction being executed.
Memory Address Register (MA) 12 Bits	This register contains the address for memory references. Whenever a core memory location is being accessed, both for reading and for writing, the MA contains the address of that location.
Memory Buffer (MB) 12 Bits	Information passing between memory and other registers in the PDP-12 must go through the memory buffer register, whether the transfer involves the CP, an external I/O device, or another memory register. The instruction register (IR) is the only register that can be loaded directly from memory.
Mode Status Register 1 Bit	This register indicates the current operating mode (LINC or 8) of the CP.
Instruction Field Register (IF) 5 Bits	This register selects the memory field containing the executable program. In the LINC Mode, it is used to designate one of up to thirty-two 1024-word segments. In the 8 Mode, the three high-order bits of the IF are used to designate one field (of up to eight 4096-word fields).
Data Field Register (DF) 5 Bits	This register selects the memory field containing data to be indirectly accessed by the memory reference instructions of a program. The fields are specified in each mode in the same way that the IF specifies the instruction field.

1.3.3 Adder

The arithmetic unit of the CP is the adder. The CP contains 12 adders (see prints *PRA* through *PRF*). A simplified schematic of the adder is shown in Figure 1-3.

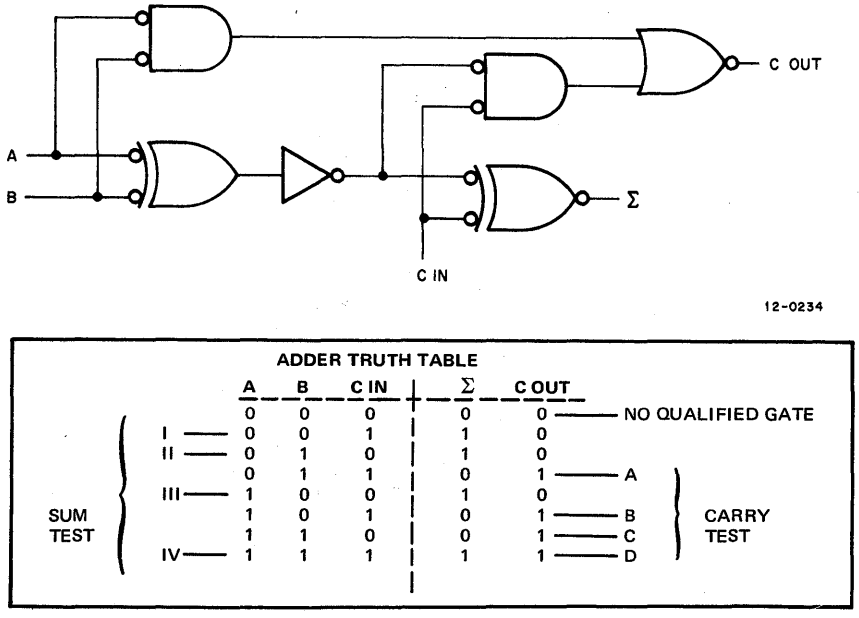


Figure 1-3 CP Adder, Simplified Logic Diagram

As shown in the truth table, when both A and B are logical 0s (+3V in this case), a logical 0 is the output at Σ.

If either input A or B is a logical 1 (0V in this case), a logical one is the output at Σ. If both inputs A and B contain a logical one, the output is 0 and a Carry Out is generated. If Input A or B is true, and Carry Insert is true, then Σ = 0 and C Out = 1. Input A and B and C results in Σ = 1 and Carry Out = 1.

1.4 CENTRAL PROCESSOR TIMING DESCRIPTION

CP timing determines the order of events in the PDP-12. CP timing comprises a series of signal levels designated *time states*, each of which is terminated by a *time pulse*. During a time state, operating conditions are established, the register gate inputs are enabled, and data are operated on in the adders and shifters. The time pulse causes data at the shifter outputs to be loaded into the specified active registers. Time pulses also synchronize CP operations with memory operations, as well as with I/O control.

The duration of the CP timing cycle is 1.6 μs ± 20 percent. The cycle is divided into five time states ranging from 250 ns to 520 ns in duration. The time pulses, which are approximately 100 ns long, simultaneously terminate one time state and initiate the next. Time states are triggered by leading edges of time pulses; thus, the duration of the pulse does not affect the length of the cycle.

The relation of time pulses to time states is shown in Figure 1-4. Information is processed through the register bus during every time state; data can circulate through this logic, into and out of the active registers, up to five times in a single cycle. The end of one cycle signals the start of the next.

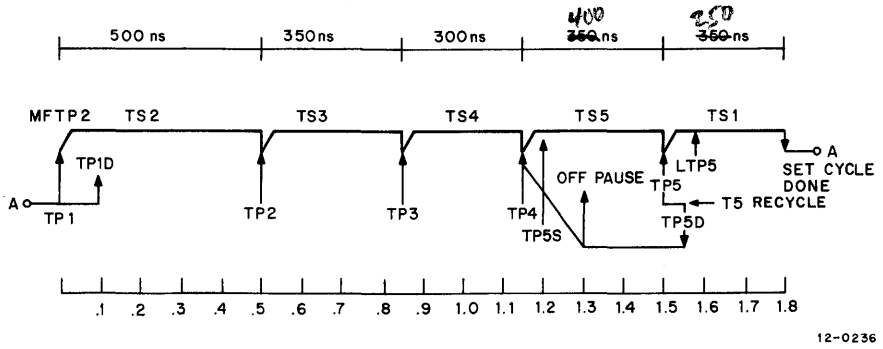


Figure 1-4 CP Timing Cycle: Time Pulses and Time States

The first CP timing cycle is initiated by the conditions established at the end of the Manual Function Timing Chain. The cycle begins with Time Pulse 1 (TP1); the first time state is TS2. The sequence proceeds from TS2 to TP5, then to TS1, which is the beginning of another cycle. For certain LINC instructions and for the PDP-8 EAE operations, a special pulse, TP5D, causes TS5 to recycle until the operation is complete.

Both the console and CP timing chains are propagated by a series of pulses taken from delay line taps. The time states are established by cross-coupled NOR gates used as RS flip-flops. The sequence and duration of pulses and levels in both chains are shown in Figure 1-5.

Table 1-2 shows the function of each time state and time pulse used by the CP.

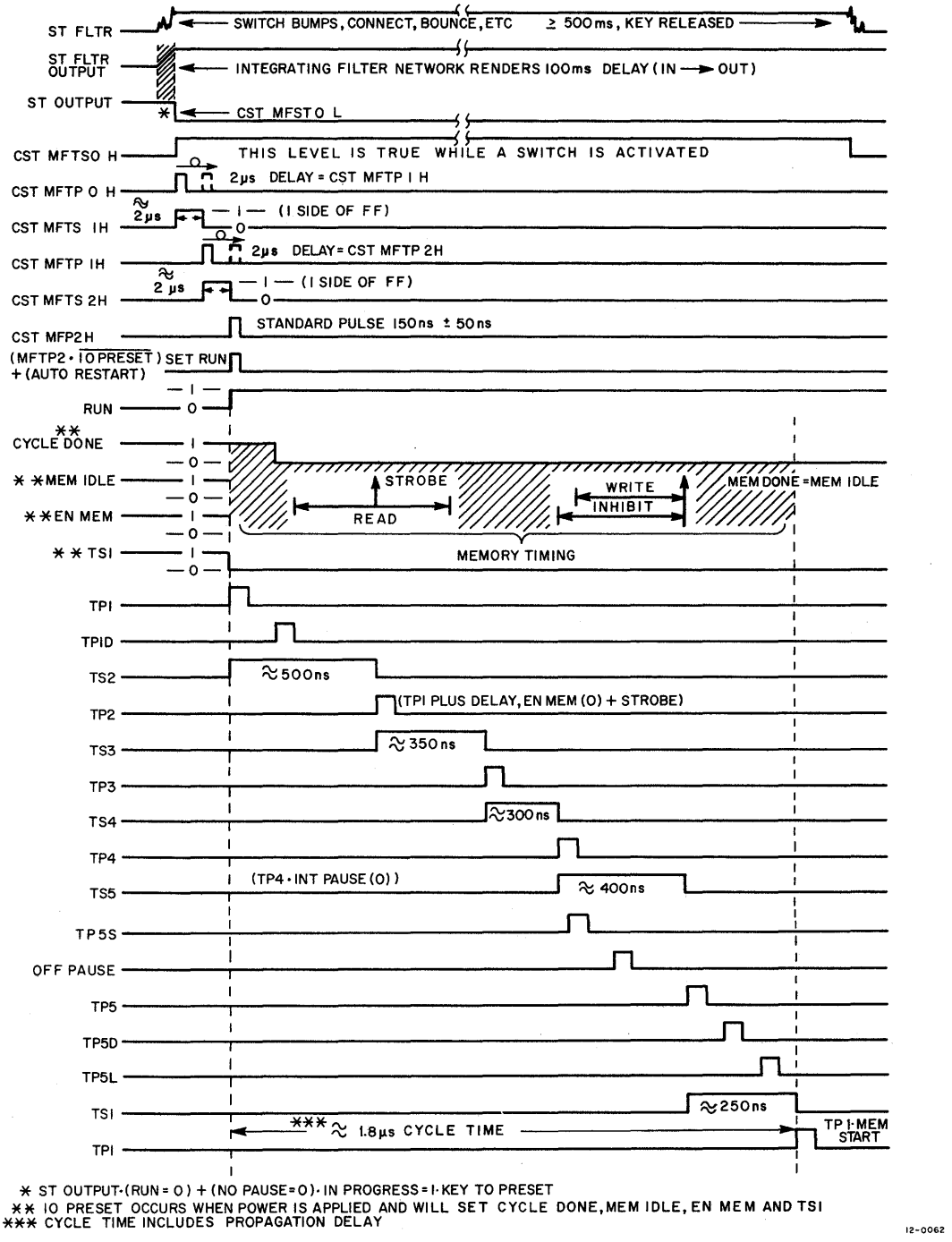


Figure 1-5 CP Timing and Run

Table 1-2
Time States and Time Pulses

Time State/Pulse	Function
TS1	Readies CP for next cycle and restart timing or start timing
TP1	Finds starting location of program or instruction; generates START MEM if EN MEM is set
TS2	Accesses memory for information
TP2	Completes memory access and/or finishes finding address
TS3	Clears previous commands and/or determines if indirect or direct addressing is used
TP3	Places memory information into buffer to be acted upon
TS4	Decodes command and/or enables register for functions to be performed
TP4	Performs operations
TS5	Cleans up and finishes instruction or enters extra cycles if instruction not complete
TP5S	Checks if any outside peripheral is ready to send or receive information
TP5D	Starts the TP5 sequence of pulses when T5 RECYCLE is set
OFF PAUSE	Checks for T5 RECYCLE
TP5	Does another cycle or cleans house to start new instruction and performs operations
~~ETP5~~ TP5L	Sets up memory for outside peripheral if ready to send or receive information
T5 RECYCLE	Recycles in TS5 to complete long commands

1.5 MAJOR STATES

Most PDP-12 instructions require more than one cycle to accomplish decoding, access to memory for data, and execution of the specified operations. In addition, special operations are required when a Program Interrupt or a Data Break is requested. To accommodate these different kinds of events, nine different timing-cycle operational sequences have been established, designated *Major States.* These Major States and associated operations are tabulated in Table 1-3, which is a condensation of the information presented on the Major State Flow Diagrams (PDP-12-0-10 through 23). A block diagram of the Major States and principal distinguishing features is shown in Figure 1-6. It can be seen from Table 1-3 and Figure 1-6 that certain Events are associated with specific time pulses, regardless of the Major States in which they occur. For example, the MA is loaded only at TP1. The MB is normally loaded at TP3. The memory Read-Write cycle is started at TP1 (except when using KEY DO) whenever memory access is required, and the MEMORY STROBE triggers TP2 and reads the contents of the addressed register into MEM.

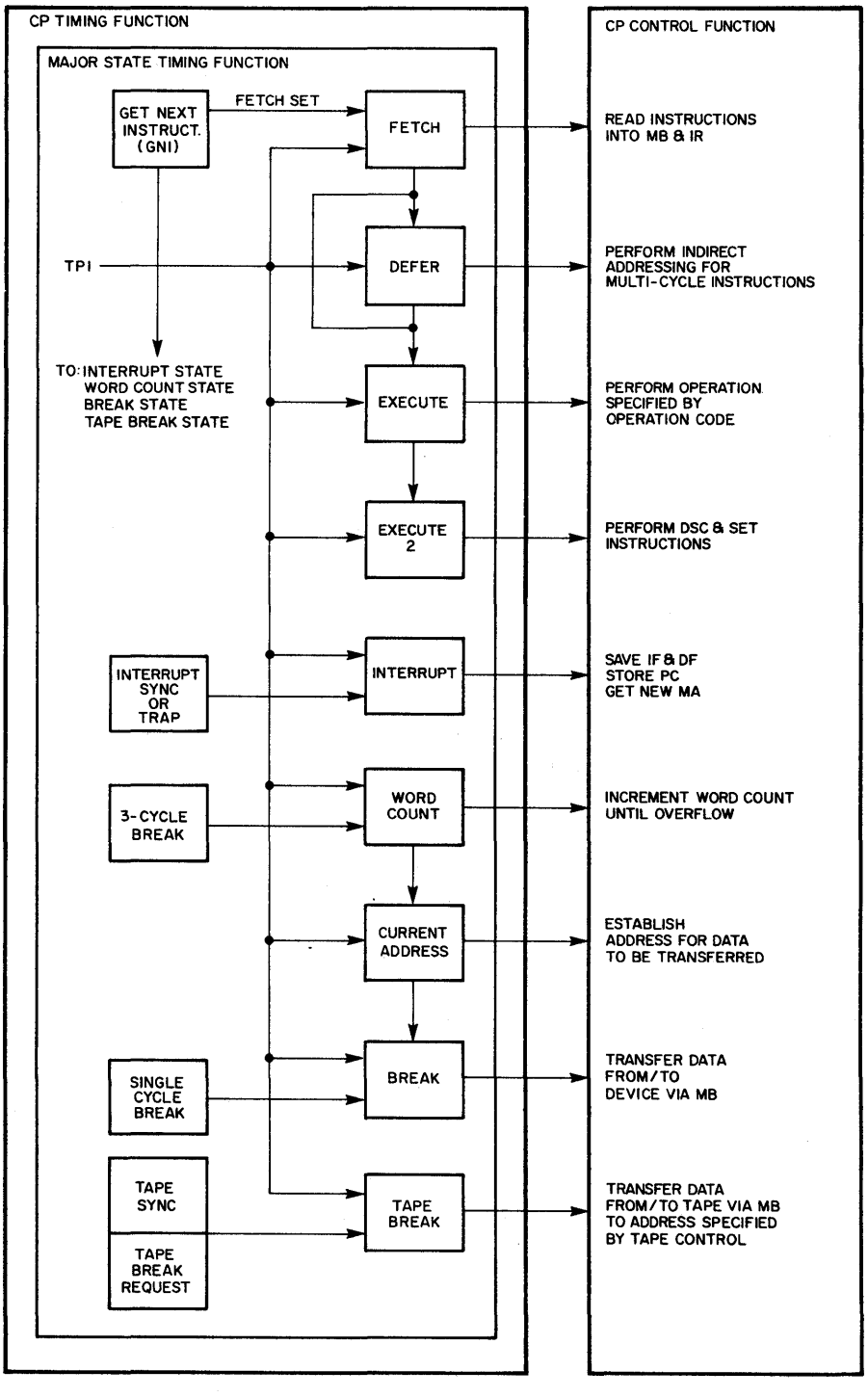


Figure 1-6 CP, Major State Block Diagram

Table 1-3
Summary of Operations

		Fetch	Defer	Execute	Exec 2	Interrupt	Word Count	Current Address	Break	Tape Break
	Load MA:	LINC 8 from PC & IF	LINC 8 from IF & PC from MB or IR	LINC 8 from PC, MB from MB or IR (XSK)	LINC Only from IR & IF (DSC & SET)	Save Fields, Clear IF & DF*	from DATA ADD	Increment MA	from MB (3-cycle) or DATA ADD	from TAPE ADD
	TS1	Set INT, WC, BREAK T BRK on break requests				LINC Set MA, Clear 8 INT EN, Clear MA, TRAP INT EN				
TP1	Start Mem.					*at TP5D				
TP1D	Clear Cycle Done									
	Load PC: TS2	from MA Clear IR	from MA	from MA or RSW	(DSC) Set Vertical size in AC	Clear IR	Clear IR	Clear IR	Clear IR	Clear IR
TP2	Strobe Mem.									
	Load MB:	from MEM or LSW	from MEM from MEM or RSW	from MEM, AC, or PC and RSW	(DSC) from MEM, incremented	from PC & SKIP	from MEM	from MEM	from MEM or DATA IN	from MEM or TAPE BUF
	TS3	Load IR from MEM or LSW. Clear SKIP & H	autoindex	Set Flow Carry → L MEM → AC Half-word Load MB Set PAUSE or INT PAUSE	Carry → SKIP		CARRY → WC OVERFLOW			Set INT PAUSE
TP3										
	Load PC:	(JMP) MB & MA → PC	(JMP) MB → PC	MB → AC AC ½-word shifts						
	TS4	(SAM, Rotates) Set T5 RE- CYCLE JMP Set Field Clear INT INHIBIT	(OPR) Set SKIP, Clear AC, L Comp. AC, L Rotate AC, L (IOT) set IOT PAUSE	Set FLO, IOT PAUSE Restart Display Start Tape (MUL) AC → MQ Set RECYCLE SYNC MB0 → H	MB or AC → AC MA → PC	MA → PC				
TP4				OFF Set T5 RECYCLE PAUSE (MUL)						
	TS5	MSC class op- erations MB → IB, DF Set SKIP, INT INHIBIT (JMP) 0 → PC (SAM) Recycle till ready; then AD → AC (Rotate) AC & MQ shifts	Rotate AC, L RSW → AC (IOT) IOP event times 1, 2, 4 Clear IOT PAUSE	Set SKIP CLR MA AC ½-word shifts (DSC) Set Intensity & Vert Buffers (DIS) Set display control buffers (JMP) IR & IF → PC (IOT) IOP events	Complement AC To FETCH	(DSC) Load co-ordinate buffers To DSC control flow			LTP5: Set Ext. Address	Wait for word Last Word: Last Checksum → AC Clear INT PAUSE
TP5		To FETCH, DEFER, EXECUTE, or INTERRUPT	To FETCH or To EXECUTE EXECUTE	(MUL) Recycle till done [TP5D] To FETCH, EXEC 2, or TAPE BREAK	(DSC done, SET) To FETCH	To FETCH	To CA	To BREAK	To FETCH or T BRK	To FETCH

The names of the Major States describe the nature of the events which occur during the cycle. The Major States are as follows:

FETCH – A new instruction is obtained from the core memory specified by the contents of the MA, and loaded into the IR, where it is decoded. (The instruction is also loaded into the MB at the same time.) If no further memory reference is required, the specified operations are carried out during T4 and T5. LINC ROTATE class EAE, MUL, and SAM instructions (except when in Fast-Sample mode) recycle in T5 until the required operations are completed.

DEFER – Whenever an indirect memory reference is required, the DEFER state is entered immediately after FETCH. Data are read from the addressed memory locations into the MB, where the data becomes available as an operand for an address in the next cycle (which is EXECUTE for all instructions except 8-Mode JMP). In the latter case, the new address is placed in the PC, and the next cycle is FETCH.

EXECUTE – The MA is set up, and data from memory are accessed. Operations are carried out during time states 3, 4, and 5. For EAE instructions MUL, SAM, and ROTATE, the processor recycles in T5 until the operation is complete.

EXEC 2 – This state is used in LINC Mode only, to provide the additional memory reference required by SET and DSC instruction. For DSC, the display buffers are loaded during T5, and the display control sequence is initiated.

INTERRUPT – If the Program Interrupt is enabled and an Interrupt Request occurs, the INTERRUPT state is entered at TP1. The memory field registers are saved; the MA is set to the proper Interrupt Address; and the PC is loaded from the MA. Refer to Chapter 3 for a detailed discussion of the INTERRUPT state.

WORD COUNT – This is the first cycle of a three-cycle Data Break. The contents of the Word Count register (number of words being transferred) are incremented. Refer to Chapter 3 for detailed description.

CURRENT ADDRESS – This is the second cycle of a three-cycle Data Break. The contents of the Current Address register (address of the data being transferred) are incremented. Refer to Chapter 3 for further explanation.

BREAK – This is the last cycle of a three-cycle Data Break, and it is the only cycle of a single-cycle Break. In either case, the data are transferred between memory and the I/O Bus during the BREAK state.

TAPE BREAK – This is a Data Break cycle used only by the LINCtape processor for transferring data between memory and the LINCtape buffer. Refer to Chapter 6.

1.6 CONSOLE INPUTS AND TIMING

The PDP-12 console switches provide data and control inputs for starting and operating the CP. Figure 1-7 shows the principal information paths from the console to the CP. Control inputs (STARTs, EXAMs, FILLs, DO, CONT, I/O PRESET) start the console timing chain to perform the operations required. The stop switches cause the internal operations to halt under various conditions. The SENSE switches are inputs to the skip logic (SKL) for program branching. The LEFT and RIGHT SWITCHES are data inputs, sensed by the program or used in the course of a DO operation. The INST FIELD switches and high-order LEFT SWITCHES set the memory field registers. The MODE switch determines the initial operating mode (LINC or 8) to be established when I/O PRESET is pressed.

Whenever a Console Control switch is actuated (dwg. *CST*), the console timing chain is started. This chain consists of three time states, each terminated by a time pulse. The last time pulse may cause a CP timing cycle to begin.

1.6.1 Console Timing (dwg. *CST*)

The manual function timing chain is initiated whenever a control switch (START, FILL, EXAM, CONT, DO, or I/O PRESET) is pressed. The entire circuit is found on a single double-width M700 Module.

The switch output is smoothed by an integrating filter, for input to a Schmitt trigger. The trigger output, a negative-going pulse, starts the timing chain, provided that the RUN flip-flop (dwg. *CPR*) is clear and that a LINCtape operation is not in progress. The trigger output generates MFTS0 L, which remains low until the switch is released.

At the same time that MFTS0 L is established, MFTP0 H occurs and inaugurates a series of delays, each about 2 μs in length. MFTP0 H sets the MFTS1 flip-flop; 2 μs later, and MFTP1 H sets MFTP2 to terminate the previous time state by clearing MFTS1. Finally, MFTP2 H clears MFTS2. For all control switches except I/O PRESET, MFTP2 H initiates the CP timing cycle; thus, CST MFTP2 L direct-sets the RUN flip-flop (dwg. *CPR*). The pulse also generates CPTP SET CYCLE DONE L, which sets the CYCLE DONE flip-flop (dwg. *CPT*).

1.6.2 Console Switch Functions

The manual function timing chain is initiated by these switches: I/O PRESET, FILL, EXAM, FILL STEP, STEP EXAM, START 20, START 400, START LS, DO and CONT. Brief descriptions of their operations are given here; the START LS Switch is described in detail in Section 1.8.

Except for CONT, all the control switches cause the SKIP, H, and Major State flip-flops to be cleared at MFTP0. Except for I/O PRESET, all these switches inaugurate at least one cycle of CP timing.

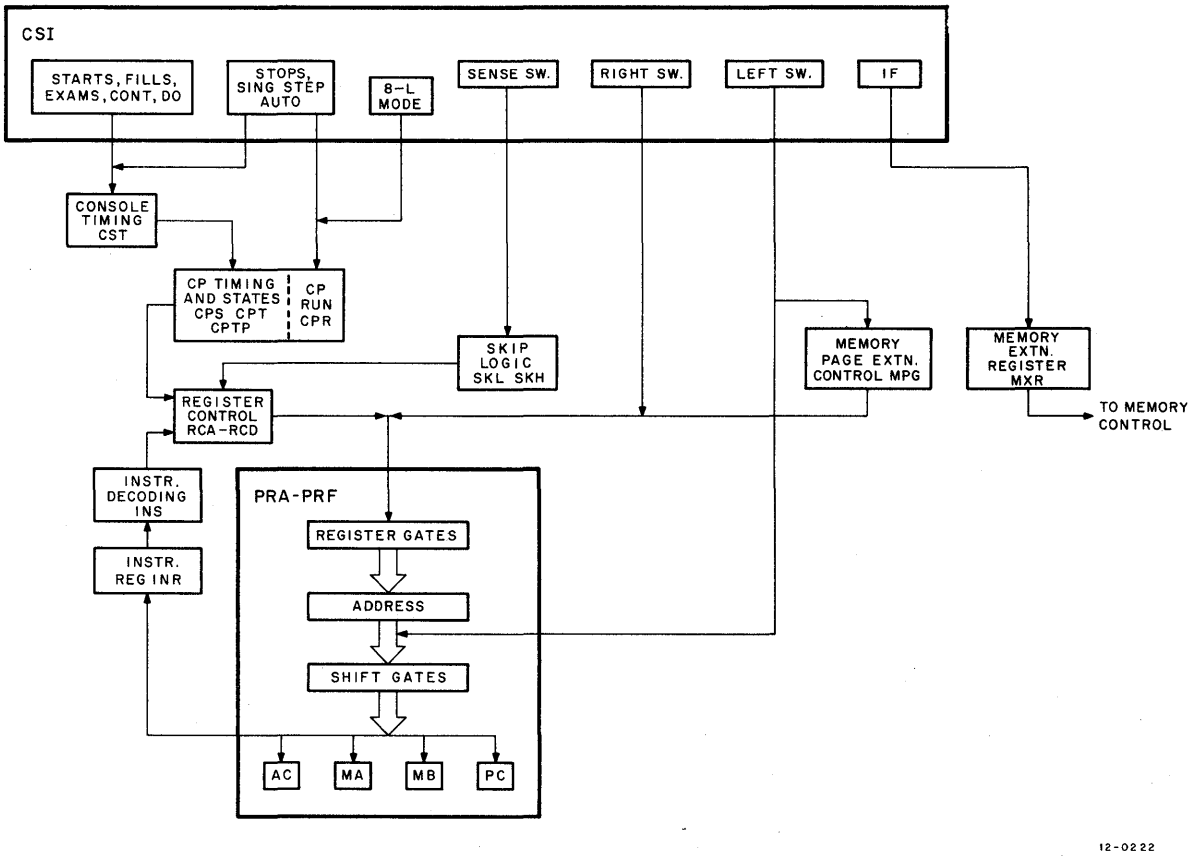


Figure 1-7 Console Switch Inputs to the PC

1.6.3 I/O PRESET

The CP is set to initial conditions by KEY I/O PRESET or by a signal called MCT POWER CLR L, which generates I/O PRESET when the machine is turned on. The switch signal, CSI KEY I/O PRESET L, is inverted and gated with MFTP1 H to provide CST I/O PRESET, in both H and L versions. These two signals clear almost every active register and flip-flop in the CP. CST I/O PRESET L also generates IOC I/O PRESET, which is used to clear most of the flip-flops in the various I/O device controls.

At MFTP2, the CP operating mode is established according to the setting of the console MODE switch. KEY I/O PRESET H is gated with MFTP2 to produce MPG MEM EXTN PRESET L (dwg. *MPG*), which is inverted (dwg. *CPR*) to yield CPR SET MODE H. This, in turn, is gated with the output of MODE switch (KEY LINC) to establish the mode. The mode-controlling flip-flop is shown as a pair of cross-coupled NOR gates (CPR). LINC memory fields are established at MFTP2; Memory Page Extension Control flip-flops IB03, IF03, SF06, DF03, and DF04 are direct-set.

I/O PRESET does not clear the IR. The three high-order bits, IR bits 0 through 2, are directly set to 1s by IOR I/O PRESET (dwg. *INR*) to ensure no conflict between the setting of the MODE switch and a mode-change instruction (LINC) that might have been left in the IR.

NOTE

The I/O PRESET operations are also executed when computer power is turned on; the processor is set for 8 MODE, regardless of the position of the MODE switch.

In the following descriptions of the flow diagrams, T_N is often referenced. T_N is comprised of TS_N and TP_N.

1.6.4 FILL and EXAM

The FILL switch is used to deposit instructions or data from the memory, using the LEFT SWITCHES (LSW) on the PDP-12 Control Panel to specify the address used. The EXAM switch is used to examine the contents of the core location specified by the LEFT SWITCHES. The logic for the FILL and EXAM functions is identical during manual timing. At MFTP2, the bits of the LSW are loaded into the MA. This action is accomplished by enabling LSW during MFTS2 (dwg. *RCC*) and loading MA at MFTP2 (dwg. *RCL*). The RUN flip-flop is set (dwg. *CPR*), and memory timing is started (dwg. *CST*). The Get Next Instruction (GNI) signal (dwg. *CPS*) is disabled because no subsequent Major State is required. The outputs of the LEFT SWITCHES (dwg. *RCC*) are enabled during TS1 and are loaded into the MA (dwg. *RCL*) at TP1. During T2, the instruction register (INR) is cleared by CPTP TP2 H · CPS DEFER (0) H · CPS EXECUTE (0) H · CPS EXECUTE 2 (0) H, as shown on the INR print. The only difference between a FILL function and an EXAM function occurs at T3. The contents of RSW are loaded into MB for a FILL function, and MEM is loaded into MB for an EXAM function. No action occurs during T4, and during T5 the RUN flip-flop (CPR) is cleared to stop timing. The AUTO flip-flop (dwg. *CST*) is also reset if it had been set by pressing the AUTO switch.

1.6.5 FILL STEP

When the FILL STEP key is depressed, the contents of RSW are loaded into memory at the location specified by LSW. At MFTP0 the SKIP, H, and Major State flip-flops are cleared.

During MFTP1 the FILL STEP flip-flop is direct-set (dwg. *CST*), and the AUTO flip-flop is set if the AUTO switch is depressed simultaneously with the FILL STEP switch. At MFTP2, MA is loaded into itself (dwg. *RCB*), because the MA is always enabled during MFTP2, which occurs at the same time as TP1. The RUN flip-flop is direct-set by CST MFTP2 (dwg. *CPR*), and the EN MEM flip-flop is set (dwg. *CST*). The Get Next Instruction

gate is inhibited (dwg. *CPS*), thereby preventing the FETCH flip-flop from being set to initiate another FETCH cycle. At T2 the IR is cleared (dwg. *INR*); at T3 the RIGHT SWITCHES (RSW) are enabled (RCD), and their contents are loaded into the MB. During T5, the RUN flip-flop is cleared by the FILL STEP input on the data side of the flip-flop, and the AUTO flip-flop is cleared (dwg. *CST*) if the AUTO switch had previously been depressed.

After the FILL STEP operation is completed and the FILL STEP switch is released, a STEP EXAM operation is performed. The STEP EXAM flip-flop is set at TP5, as shown on the *CST* print. Signal CST STEP EXAM (1) L · CST MFTS0 H produces CSI (KEY STEP EX. MFTS0) L. This signal is ANDed with CPT TS1 (1) H to produce a Carry Insert at T1 to increment the MA. The remainder of the function is treated as in the STEP EXAM function.

1.6.6 STEP EXAM

When the STEP EXAM switch is depressed, the contents of memory are displayed by the memory buffer display on the front panel. The address of the data to be displayed is specified by the memory address (MA). At MFTP1, the STEP EXAM flip-flop is set by CSI KEY STEP EXAM IN H and CST SET KEYS H, as shown on the *CST* print. The AUTO flip-flop is also set if the AUTO switch is on. At MFTS2, Carry Insert (dwg. *CYI*) is enabled by KEY STEP EXAM and CPT TS1 (dwg. *CYI*) and loaded during TP1 to increment the MA. The RUN and START MEM flip-flops are set, and GNI is disabled. During T2 the IR is cleared. At TS3, MEM is enabled (dwg. *RCB*) and loaded into the MB at TP3. During T5 the RUN, STEP EXAM, and AUTO flip-flops are cleared.

1.6.7 START 20

This switch causes the processor to start at location 20 of the currently selected instruction field. At MFTS1, PMB BMSC 07 H is enabled by MFTS1 (1) H and CSI KEY ST 20 H, as shown on the PMB print. At MFTP1, bit 07 is loaded into the Program Counter (PC). At MFTP2, the PC, now containing the number 20, is loaded into the MA. The FETCH and RUN flip-flops are also set at this time to start the program at location 20.

1.6.8 START 400

The switch has the same effect as START 20, except that PMA TMSC 03 H is enabled to start the program at 400.

1.6.9 START LS

This switch causes the processor to start the program at the address specified by the LEFT SWITCHES. At MFTS1, the bits of LSW are enabled by CSI KEY LSW H · CST MFTS1 (1) H to provide RCC EN LSW H, which is routed (dwgs. *PRA* through *PRF*) and ANDed with CSI LSW 00 through 11 H. LSW are loaded into the PC during MFTP1, as shown on the *RCL* print. Flip-flops IB03-04 and IF03-04 are also direct-set at this time by CSI LSW 00-01 · RCC ST LSW EN H, as shown on the *MPG* print. When extended memory is used, CSI IF 00 through 02 flip-flops are direct-set IB00 through 02, as shown on the memory extension register (*MXR*) print. The RUN flip-flop is direct-set as shown on the *CPR* print, by CSI MFTP2 L · -CSI KEY I/O PRESET L to start memory and timing.

1.6.10 DO

This switch causes the processor to perform one instruction. In the LINC Mode, the processor performs the instruction defined by the LEFT SWITCHES. RIGHT SWITCHES are also used if the instruction is a two-word format instruction. In the 8 Mode, the processor performs the instruction defined by the LEFT SWITCHES only.

At MFTP1, the AUTO flip-flop (dwg. *CST*) is clocked and set if the AUTO Switch on the front panel is depressed. The DO flip-flop is direct-set by CST MFTP1 and CSI KEY DO. The ENABLE MEM flip-flop (dwg. *CST*) is direct-cleared by CST SET DO L, which is routed from the 0 side of the DO flip-flop. At MFTP2 or TP1, the FETCH flip-flop (dwg. *CPS*) is set. The RUN flip-flop (dwg. *CPS*) is also direct-set by CST MFTP2 ·-CSI KEY I/O PRESET H.

At T1, PC is enabled and loaded into MA. At T2, MA is loaded into the PC, and the IR is cleared. During TS3 LSW are enabled by RCC EN LSW L (as shown on the *RCC* print) and routed to the logic shown on the *PRA* through *PRF* prints, where IR and MB are loaded simultaneously at TP3. At T5, the RUN and AUTO flip-flops are cleared and memory timing is stopped, if the AUTO switch is not depressed or a multicycle instruction was not executed.

1.7 CENTRAL PROCESSOR TIMING – LOGIC DESCRIPTION (dwgs. *CPTP, CPT, CPR, CPS*)

The CP timing cycle can begin only when the three flip-flops CPR RUN, CPT CYCLE DONE, and CPT MEM IDLE are all in the 1 state. RUN and CYCLE DONE are set by MFTP2; MEM IDLE is set when a MEM DONE pulse occurs at the end of a memory Read-Write cycle.

The cycle starts with the TP1 pulse, which is provided in both positive- and negative-going forms. At the same time, the memory Read-Write cycle is initiated by generating the CPTP START MEMORY H pulse (except when a DO switch function is depressed). The pulse CPTP TP1 H sets off the first delay line in the time pulse sequence (dwg. *CPTP*). This also establishes CPT TS2 H by setting the flip-flop TS2 to the 1 state (dwg. *CPT*).

NOTE
Levels TS1 through TS4 are conditioned by means of RS-type flip-flops, shown on dwg. *CPT* as cross-coupled NAND gates.

The first pulse, tapped off the delay line at 100 ns, is CPTP TP1D H, which clears the CPT CYCLE DONE flip-flop. The next pulse, at 350 ns, is gated with CST EN MEM (0) H to provide CPTP TP2 whenever a DO function is being executed and memory has not been enabled. In normal operation, however, CPTP TP2 is triggered by the MCT STROBE L pulse from memory control. TP2 L clears TS2 and sets TS3; TP2 H starts the next delay (dwg. *CPTP*). TP3 and TP4 are tapped off this line at 250 ns and 550 ns, respectively. Each of these clears the previous time state and sets the next one. TS5 is established by setting a D-type flip-flop, CPT TS5 (dwg. *CPT*) · CPTP TP4 H, gated with INT PAUSE (0) H, sets off the next delay line in the series, produces CPTP TP5S (generated immediately), and clocks the delay line.

The first pulse from the delay is CPTP OFF PAUSE H, tapped at 100 ns; it controls the action of T5 RECYCLE (see Paragraph 1.7.1). Time pulse CPTP TP5 H is tapped at 300 ns. The next pulse is CPTP TP5D H, tapped at 350 ns.

The last pulse, LTP5, is tapped at 400 ns. This pulse clocks the enabling flip-flops for extended memory IF, DF, BF, and SF registers (MC12-0-MXF). Finally, if T5 RECYCLE is clear, CPTP TP5 H sets CPT TS1, starting the delay line (dwg. *CPTP*), which, tapped at 200 ns, sets the CYCLE DONE flip-flop (dwg. *CPTP*) and establishes one of the conditions for TP1.

1.7.1 T5 RECYCLE

The LINC instructions ROR, ROL, SCR, SAM, and MUL, and the PDP-8 EAE instructions all require additional time at the end of a cycle to complete their operations. To provide more time, time states TS5 and TP5 are repeated until the operation is finished.

When CPTP TP4 H sets CPT TS5, it is also gated with one of the recycle condition signals at NOR GATE M117 location L22 to set the RECYCLE SYNC flip-flop (dwg. *CPT*). The output of this flip-flop enables the data input of the T5 RECYCLE flip-flop, which is set at the occurrence of the OFF PAUSE H pulse. The 1 output of T5 RECYCLE causes TS5 to remain set when CPTP TP5 H occurs. At the same time, the signal CPT SET TS1 L is inhibited, thus preventing TS1 from being established. The timing chain to produce T5 is restarted by CPTP TP5D H, gated with CPT T5 RECYCLE (1) H. As soon as the recycle condition is no longer true, the RECYCLE SYNC flip-flop is cleared, disabling the T5 RECYCLE flip-flop (at the next CPTP OFF PAUSE H pulse) and allowing the normal sequence of events to proceed.

1.7.2 CP Timing and the Memory Read-Write Cycle

In normal operation, the memory cycle is initiated once during each CP timing cycle. At TP1, the CPTP START MEMORY H pulse is produced, which starts the memory timing chain (dwg. *MCT*; refer to Chapter 2 for a detailed description). The Read-Write cycle proceeds independently, but the CP cycle is dependent upon the memory cycle at two points:

a. TP2 is initiated by the MCT STROBE L pulse; the CP cycle then proceeds to completion.

b. TP1 is not asserted until the memory cycle is finished, and MEM IDLE (1) H is true.

In practice, the memory cycle is somewhat faster than the CP cycle; consequently, there is no delay at either of the two points.

1.8 THE CENTRAL PROCESSOR IN ACTION

The following discussion is conducted on a *gate-chasing* level and is keyed directly to the logic schematics in Volume III (also, refer to Tables 1-4 and 1-5). The activities of the CP are detailed sequentially, from the pressing the START LS through the completion of a LINC Iβ class instruction.

Two important signals, not directly in this sequential flow but essential to the understanding of processor action, are CPS GNI and CPT TS1.

1.8.1 Time State 1: CPT TS1 H

Whenever the processor is not running (RUN flip-flop is clear), the flip-flop TS1 (dwg. *CPT*) is in the 1 state, because the processor does not stop until a cycle is complete. The final pulse CPTP TP5 H, sets CST TS1 (1) H. CPT I/O PRESET L also sets TS1 to 1.

1.8.2 Get Next Instruction: CPS GNI (dwg. *CPS M115 K04*)

The circuitry that produces the CPS GNI is a very common example of the PDP-12 inhibit logic. Basically, the signal is true *unless* one or more of several disabling conditions is true. Refer to the inputs on the gate that produces CPS GNI L (dwg. *CPS M115 K04*). This gate is a NAND gate; thus, the signal is true only when all the inputs are high. The names for two of these inputs are preceded by negation signs (-). In each case, this notation means that the input is high when the named signal is *not* true. To find the source of the signal, remove the negation and change H to L. Example: The source of -CPS DEFER SET H is CPS DEFER SET L. Examination of the AND-NOR gate expansion M160 (dwg. *CPS*) reveals that a similar condition holds for some inputs. Consequently, unless specifically inhibited, CPS GNI L is always true.

Table 1-4
Functional Logic Description, START LS

Time State/ Major State	Logical Sequence	Logic Component Location and Drawing Reference	Description
	1	NOR GATE M115, K13 (*CSI*)	START LS produces the signal CSI KEY STARTS H, as do any of the start keys.
	2	INVERTER M111, K13 (*CSI*)	CSI KEY STARTS H is inverted to CSI KEY STARTS L.
	3	NOR GATE M117, L22 (*CST*)	The signal CSI KEY STARTS L qualifies the gate and is routed through a filter and Schmitt trigger to eliminate switch bounce.
	4	NAND GATE M700 (*CST*)	From the Schmitt trigger, a 100 ns pulse is sent to the NOR gate and is ANDed with the signal at pin KP2. The gate qualifies because the RUN flip-flop is not set at this time.
MFTS0	5	NAND GATE M700 (*CST*)	The output MFTS0 L is inverted to MFTS0 H.
MFTP0	6	NAND GATE M617, J07 (*CPS*)	Approximately 50 ns after MFTS0 occurs, the signal MFTP0 and -CSI KEY CONT H produce CPS CLR STATES to clear all Major State flip-flops.
	7	AND/NOR GATE M160, J27 (*CPS*)	-CSI KEY CONT H and CST MFTP0 produce the signal CKH CLR SKIPS to direct-clear the H and the SKIP flip-flops.
MFTS1	8	NAND GATE M113, K30 (*RCC*)	CSI KEY ST LSW and CST MFTS1 produce RCC ST LSW EN L.
	9	NOR GATE M617, H23 (*RCC*)	RCC ST LSW EN L is inverted to RCC ST LSW EN H.
	10	AND/NOR GATE M160, K36 (*MPG*)	RCC START LSW EN H ANDed with CSI LSW 00 H (if LSW 00 is depressed) produces MPG SET IB-IF 03 L to direct-set IB 03 and IF 03 flip-flops.
	11	NAND GATE M113, K36 (*MPG*)	IB 04 and IF 04 flip-flops are direct-set by RCC ST LSW EN H and CSI LSW 01 H, if LSW 01 is depressed.
	12	NAND GATE M113, K38 & K40	Extended memory flip-flops IF 0 through 2 are set by RCC ST LSW EN H and CSI IF 0 H through CSI IF 02 H.

Table 1-4 (Cont)
Functional Logic Description, START LS

Time State/ Major State	Logical Sequence	Logic Component Location and Drawing Reference	Description
MPTP1	13	(*CST*)	MPTP1 occurs approximately 2 μs after MFTS1 flip-flop is set.
	14	NAND GATE M113, H21 (*RCL*)	CST MFTP1 H and CSI KEY STARTS L produce the signal RCL START PC L.
	15	NOR GATE M617, J21 (*RCL*)	The PC LOAD signal is produced by RCL START PC L. The PC is loaded with the contents of LSW.
MFTS2	16	(*CST*)	CST MFTP1 H sets CST MFTS2 and clears CST MFTS1.
	17	NAND GATE M117, K05 (*CPS*)	This signal CPS GNI H produces the signal CPS FETCH SET L when the following conditions are true: *a.* −CPS INTERRUPT SET (no interrupt has occurred) *b.* −CPS TAPE SET (no tape break request has occurred) *c.* −CPS BREAK SET (no three-cycle data break request has occurred).
	18	NAND GATE M113, H24 (*RCC*)	CPS FETCH SET L produces the signal RCC PC FOR MA.
	19	NAND GATE M115, H25 (*RCC*)	RCC PC FOR MA and CST MFTS1 (0) H and CPT TSL (1) H produces the Enable signal RCC EN PC 2-11 H.
MFTP2 or TS1	20	(*CST*)	Approximately 2 μs after CST MFTP1 H has occurred, CST MFTP2 occurs. CST MFTP2 clears the flip-flop CST MFTS2.
	21	NAND GATE M113, L26 (*RCL*)	CST MFTP2 and -CSI KEY I/O PRESET produce the Load signal RCL LOAD MA H.
	22	NOR GATE M112, L10 (*CPR*)	Signals CST MFTP2 L and -CSI KEY I/O PRESET produce the signal CPR SET RUN L to direct-set the RUN flip-flop and start CP timing.

Table 1-5
Functional Logic Description, ADA I 12

Time State/ Major State	Logical Sequence	Logic Component, Location and Drawing Reference	Description
			It is assumed that the instruction is in memory and will be fetched.
TS1	1	FLIP-FLOP M216, J12 (*CPT*)	The signal CPT SET CYCLE DONE L direct-sets the CYCLE DONE flip-flop.
	2	NAND GATE M617, L05 (*CPTP*)	Because all inputs on the gate are true, the gate qualifies and generates CPTP TP1 L.
FETCH TP1	3	NAND GATE M617, L05 (*CPTP*)	CPTP TP1 L produces CPTP TP 1 H.
	4	FLIP-FLOP M216, K06 (*CPS*)	When the signal CPS FETCH SET L is true, the flip-flop is set by the pulse CPTP TP1 H. All other Major State flip-flops are cleared.
	5	NAND GATE M112, K06 (*CPTP*)	The signal CPTP START MEMORY L is generated here.
	6	NOR GATE M617, J21 (*RCL*)	CPTP TP1 L produces the signal RCL LOAD MA H. The MA contains the address where the instruction is to be fetched.
	7	DELAY LINE M310, H06 (*CPTP*)	CPTP TP1 H triggers the delay line and produces the signal CPTP TP1 D 100 ns later.
	8	NOR GATE M115, L08 (*CYI*)	CPS FETCH (1) L produces the signal CYI PC INCREMENT.
	9	NAND GATE M617, J06 (*CPT*)	The pulse CPTP TP1 direct-clears the CPT TS1 flip-flop and direct-sets the CPT TS2 flip-flop.
	10	NOR GATE M617, J07 (*CPS*)	CPS FETCH (1) H produces the signal CPS FETCH B (1) H.
TS2	11	AND/NOR GATE M160, J26 (*RCB*)	CYI PC INCREMENT and CPT TS2 (1) H qualify the Enable gates to produce RCB EN MA 0-4 H and RCB EN MA 5-11 H.
	12	AND/NOR GATE M160, L11 (*CYI*)	The signal CYI CARRY INSERT L is generated by signals CYI PC INCREMENT H and CPT TS2 (1) H and CST EN MEM (1) H.

Table 1-5 (Cont)
Functional Logic Description, ADA I 12

Time State/ Major State	Logical Sequence	Logic Component, Location and Drawing Reference	Description
	13	(*PRF*)	The signal CYI CARRY INSERT L is routed to adder bit 11 to increment the PC. If PRF PARTIAL SUM L already contains a logical 1, the output out of the adder is zero and a Carry Out to the next higher order adder occurs.
TP2	14	NOR GATE M617, J07 (*CPTP*)	Approximately 500 ns to 520 ns after the CPT START MEMORY H pulse the MCT STROBE L pulse occurs, producing the signal CPTP TP2 H. The MCT STROBE L signal also clears the MEM IDLE flip-flop.
	15	NAND GATE M113, H21 (*RCL*)	CYI PC INCREMENT H and CPTP TP2 H produce the signal RCL LOAD PC H, which is used to load the PC register.
	16	(*CPT*)	CPTP TP2 H · CPS DEFER (0) H · CPS EXECUTE (0) H · CPS EXC 2 (0) H produce the signal INR CLEAR IR L to direct-clear the IR.
TS3	17	NAND GATE M617, H38 (*INR*)	CPTP TP2 L clears the CPT TS2 flip-flop and sets the CPT TS3 flip-flop.
	18	NAND GATE M117, H30 (*RCB*)	RCB EN MEM H is produced by RCB GO MEM TS3 L. (This signal is high because the inhibit logic that produced it is not qualified.) · CPS INTER (0) H · CPT TS3 (1) H · CST EN MEM (1) H.
TP3	19	INVERTER M111, J09 (*CPTP*)	Approximately 350 ns after CPTP TP2 H CPTP TP3 L is generated.
	20	NAND GATE M113, H21 (*RCL*)	CPTP TP3 H and CPS EXEC (0) H produce RCL LOAD MB H. The contents of MEM are loaded into the MB.
	21	NAND GATE M113, H36 (*INR*)	CPTP TP3 and CPS FETCH B (1) H produce the signals INR LOAD IR 0-7 and INR LOAD IR 8-11. The contents of MEM are loaded into the IR. ADA I 12 (1132_8) can now be decoded.
	22	AND/NOR GATE M160, 27 (*SKH*)	CPTP TP3 H · CPS FETCH (1) L produces the signal SKH CLEAR SKIPS L, which direct-clears the H and the SKIP flip-flop.

Table 1-5 (Cont)
Functional Logic Description, ADA I 12

Time State/ Major State	Logical Sequence	Logic Component, Location and Drawing Reference	Description
TS4	23	NAND GATE M113, J33 (*INS*)	CPR L MODE H · INR IR00 (0) H · INR IR01 (0) H produce the signal INS SGRP L, which is inverted to INS SGRP H.
		DECODER M161, J35 (*INS*)	INS SGRP H, in conjunction with IR bits 2 through 6, decodes INS ADA H.
	24	NAND GATE M115, H31 (*INS*)	INS SGRP H ·INR IR 02 (1) H produce the signal INS INDEX CLASS L, which is inverted to INS INDEX CLASS L H.
	25	DECODER M161, J39 (*INS*)	INR IR 08 (1) H · INR IR 10 (1) H produce INS N EQ 12 H and INS N EQ 12 L. The ADA instruction is decoded into usable parts (INS ADA, I = 1, and N = 12).
	26	NAND GATE M113, J33	INS INDEX CLASS L H · INS (I EQ 1 · B EQ 0) L produces INS LINC IND I RECT H when INS (I EQ1 · BEQ 0) L is not true.
	27	NAND GATE M160, K08 (*CPS*)	INS LINC INDIRECT H · CPS FETCH B (1) H produce the signal CPS DEFER SET L.
	28	(*CPTP*)	CPTP TP4 L is generated approximately 300 ns after CPTP 3 L. CPTP TP4 L clears flip-flop CPT TS4 and direct-sets flip-flop CPT TP5.
TP5	29	(*CPTP*)	CPTP OFF PAUSE occurs approximately 250 ns after CPTP TP4 L and CPTP TP5 H is generated approximately 250 ns after CPTP OFF PAUSE H.
	30	FLIP-FLOP M216, L07 (*CPT*)	The MEM IDLE flip-flop is set by CPT MEM DONE H.
	31	FLIP-FLOP M216, L17 (*CPR*)	The RUN flip-flop is clocked by CPTP TP5 H and is left-set because the inhibit logic that produces the signal CPR EN RUN H is not qualified.
	32	FLIP-FLOP M216, J03 (*CPT*)	The TS5 flip-flop is clocked by CPTP TP5 H. The data input of the flip-flop is connected to the T5 RECYCLE flip-flop. Because the T5 RECYCLE flip-flop is not set, the TS5 flip-flop is reset.

Table 1-5 (Cont)
Functional Logic Description, ADA I 12

Time State/ Major State	Logical Sequence	Logic Component, Location and Drawing Reference	Description
TS1	33	NAND GATE M115, J05 (*CPT*)	CPT IOT PAUSE (0) H · CPTP TP5 H · CPT T5 RECYCLE (0) H produces the signal CPT SET TS1 L which sets the TS1 flip-flop.
	34	NAND GATE M115, K29 (*RCA*)	The signal INS (L INDEX ·$\beta \neq 0$) L (β = 12) and CPS FETCH (1) H and CPT TS1 (1) H produce the Enable signal RCA ENABLE IR 8-11 H.
DEFER TP1	35	(*CPTP*)	With flip-flops CPR RUN · CPT MEM IDLE · CPT CYCLE DONE set, CPTP TP1 L pulse is produced.
	36	FLIP-FLOP M216, K06 (*CPS*)	The flip-flop is clocked by CPTP TP1 H. The data side of the redefined DEFER flip-flop is low because the inhibit logic was qualified by CPS FETCH B (1) H · INS LINC INDIR H.
	37	NAND GATE M112, J10 (*CPTP*)	CPTP TP1 L and CST EN MEM produce the signal CPTP START MEMORY H to start another memory cycle.
TP2	38	NOR GATE M617, J07 (*CPTP*)	Approximately 500 ns to 520 ns after CPTP TP1 H, the signal MCT STROBE is generated at the memory and routed to the NOR gate to produce CPTP TP2 H, which is routed into a delay line.
	39	FLIP-FLOP M216, L07 (*CPT*)	The MEM IDLE flip-flop is direct-cleared by MCT STROBE L.
TS3	40	(*CPT*)	CPTP TP2 L clears flip-flop CPT TS2 and sets flip-flop CPT TS3.
	41	NAND GATE M113, K30 (*CYI*)	CPS DEFER B (1) H and INS INDEX CLASS L H · -INS NEQ 00 H produce the signal CYI LINC INDEXING H.
	42	NAND GATE M113, K30 (*CPI*)	CYI LINC INDEXING H and INR IR 07 (1) H produce the signal CYI T3 INDEX H.
	43	AND NOR GATE M113, K30 (*CYI*)	The signals CYI T3 INDEX H and CPT TS3 (1) H and -PMA HALF WORD L produce the signal CYI CARRY INSERT L.

Table 1-5 (Cont)
Functional Logic Description, ADA I 12

Time State/ Major State	Logical Sequence	Logic Component, Location and Drawing Reference	Description
	44	NAND GATE M117, H30 (*RCB*)	CST EN MEM (1) H · CPT TS3 (1) H · CPS INTER (0) H · RCB GO MEM TS3 L (The last signal is high because the inhibit logic did not qualify it.) produce the signal RCB E MEM 0-5 H and RCB EM MEM 6-11 H. The address in the β register 12 can now be incremented.
CPTP TP3	45	NAND GATE M113, H21 (*RCL*)	CPS EXC 2 (0) H · CPTP TP3 H produce the signal RCL LOAD MB H. The contents of MEM, along with Carry Insert, are loaded into the MB.
TS4	46	(*CPT*)	CPTP TP3 clears flip-flop CPT TS3 and sets flip-flop CPT TS4.
	47	NAND GATE M117, H30 (*SKH*)	If MB bit 00 = 1, then signals CPTP TP4 H · CPS DEFER B (1) H · CPR L MODE H PRA MB 00 (1) H direct-set the H flip-flop.
TS5	48	FLIP-FLOP M216, J03 (*CPT*)	CPTP TP4 L direct-sets the TS5 flip-flop.
TP5	49	(*CPTP*)	CPTP TP5 is generated 400 ns after CPTP TP4 L.
	50	FLIP-FLOP M216, L07 (*CPR*)	CPTP TP5 H clocks the RUN flip-flop. The data input of the flip-flop is true because the inhibit logic is not qualified.
TS1	51	(*CPT*)	CPTP TP5 H direct-clears the TS5 flip-flop and sets the TS1 flip-flop.
	52	FLIP-FLOP M216, J12 (*CPT*)	The signal CPT SET CYCLE DONE L direct-sets the CYCLE DONE flip-flop.
	53	NAND GATE M117, L13 (*RCB*)	CPS DEFER SET H · CPTS (1) H · CPS DEFER (0) H produces RCB EN MB 5-11.
	54	AND/NOR GATE M160, J26 (*RCB*)	INS INDEX CLASS L H · CPS DEFER β (1) H · CPT TS1 (1) H produces RCB EN MB 2-4 H.
EXECUTE TP1	55	NAND GATE M617, L05 (*CPTP*)	The RUN, MEM IDLE and CYCLE DONE flip-flops are set; thus, CPTP TP1 H is produced, and the EXECUTE cycle is entered.

Table 1-5 (Cont)
Functional Logic Description, ADA I 12

Time State/ Major State	Logical Sequence	Logic Component, Location and Drawing Reference	Description
	56	NAND GATE M117, K05 (*CPS*)	The inhibit logic is no longer qualified, and the EXECUTE cycle is entered.
	57	NOR GATE M617, J21 (*RCL*)	CPTP TP1 H produces RCL LOAD MA H. The new address is placed in the MA.
TS2	58	(*CPT*)	CPTP TP1 clears CPT TS1 flip-flop and sets CPT TS2 flip-flop.
	59	(*CPTP*)	Approximately 520 ns after CPT TS1, MCT STROBE from memory produces CPTP TP2 H and clears the MEM IDLE flip-flop.
TS3	60	NAND GATE M117, H30 (*RCB*)	RCB GO MEM TS3 L · CPS INTER (0) H · CPT TS3 (1) H · CST EN MEM (1) H produce RCB EN MEM 0-5 H and RCB EN MEM 6-11 H.
TP3	61	(*CPTP*)	Approximately 350 ns after CPTP TP2 H, CPTP TP3 occurs.
	62	NAND GATE M113, H21 (*RCL*)	CPS EX 2 (0) H · CPTP TP3 H produce the signal RCL LOAD MB H. The MB has been loaded with the contents of MEM.
	63	FLIP-FLOP M216, L07 (*CPT*)	CPT MEM DONE L sets the MEM IDLE flip-flop.
	64	NAND GATE M113, J25 (*RCB*)	INS ADA L · CPS EXECUTE B (1) H produces RCB AC ADDS L.
	65	AND/NOR GATE M160, J26 (*RCB*)	RCB AC ADDS H · CPT TS4 (1) H · CPS EXECUTE B (1) H produce RCB EN MBL. RCB EN MB L enables MB 0 through 1.
	66	AND/NOR GATE M160, J20 (*RCA*)	RCB AC ADDS · CPT TS4 (1) H produce the signals RCA EN AC 0-5 H and RCA EN AC 6-11 H. The summation of AC and MB are enabled and are loaded into the register bus.
TP4	67	(*CPTP*)	Approximately 300 ns after CPTP TP3, CPTP TP4 is true.

Table 1-5 (Cont)
Functional Logic Description, ADA I 12

Time State/ Major State	Logical Sequence	Logic Component, Location and Drawing Reference	Description
	68	NAND GATE M113, H21 (*RCL*)	RCB AC ADDS L · CPTP TP4 H produce the signal RCL LOAD AC. The summation of MB and AC is loaded into the AC.
	69	FLIP-FLOP M216, J12 (*FLE*)	If overflow out of adder bit 00 occurs, the FLOW flip-flop is set.

1.9 8-MODE INSTRUCTIONS

When computer power is first applied, the PDP-12 is placed in 8 Mode by MCT PWR CLEAR L. The PDP-12 Computer can also be switched to 8 Mode from LINC Mode by setting the MODE switch to 8 Mode and pressing I/O PRESET.

Direct and indirect addressing schemes are used in the 8 Mode. A block diagram of the addressing scheme used in the 8 Mode is illustrated in Figure 1-8.

The following paragraphs describe 8-Mode FETCH, DEFER, and EXECUTE cycles. A brief description of control signals for all 8-Mode instructions is also included.

1.9.1 FETCH Cycle

All 8-Mode instructions are treated identically during T2 and T3 of the FETCH cycle. The only exception occurs when the DO flip-flop is set by the DO key: the PC is not advanced, and the contents of LSW are loaded into MB and IR. When the DO key is cleared during T2, ENABLE MA (dwg. *RCB*) and a CARRY INSERT (dwg. *CYI*) for bit 11 are generated during TS2. The PC is advanced in this fashion during TP2. The IR (dwg. *INR*) is also cleared during TP2.

During TP3, the SKIP and H flip-flops (dwg. *SKH*) are cleared. An ENABLE MEM signal is generated (dwg. *RCB*) during TS3, and MEM is simultaneously loaded into MB and IR during TP3. An instruction is decoded during T4 into a memory reference instruction (MRI), operate instruction (OPR), or Input/Output transfer instruction (IOT). During T4, the instruction register (dwg. *INR*) is gated to the instruction decoders (dwg. *INS*). which are qualified by the MODE flip-flop. Bit 03 is examined during this time to determine the addressing scheme and the next Major State. If bit 03 is set, the DEFER flip-flop (CPS) is set, thereby initiating a DEFER cycle. If bit 03 is not set, the EXECUTE flip-flop is set, thereby initiating an EXECUTE cycle for all instructions except OPR and JMP.

1.9.2 DEFER Cycle

During T1 of the DEFER cycle, the selection of page 0 or current page is determined by MB bit 04. If MB bit 04 = 0, MB bits 5-11 are enabled (dwg. *RCB*) during TS1 and loaded into MA bits 5-11 (dwg. *RCL*) at TP1. MB bits 0-4 were not enabled during this transfer; consequently, 0 is automatically placed in MA bits 0 through 4, which selects an address on page 0. If MB bit 04 = 1, an indirect address on the current page is selected and MB bits

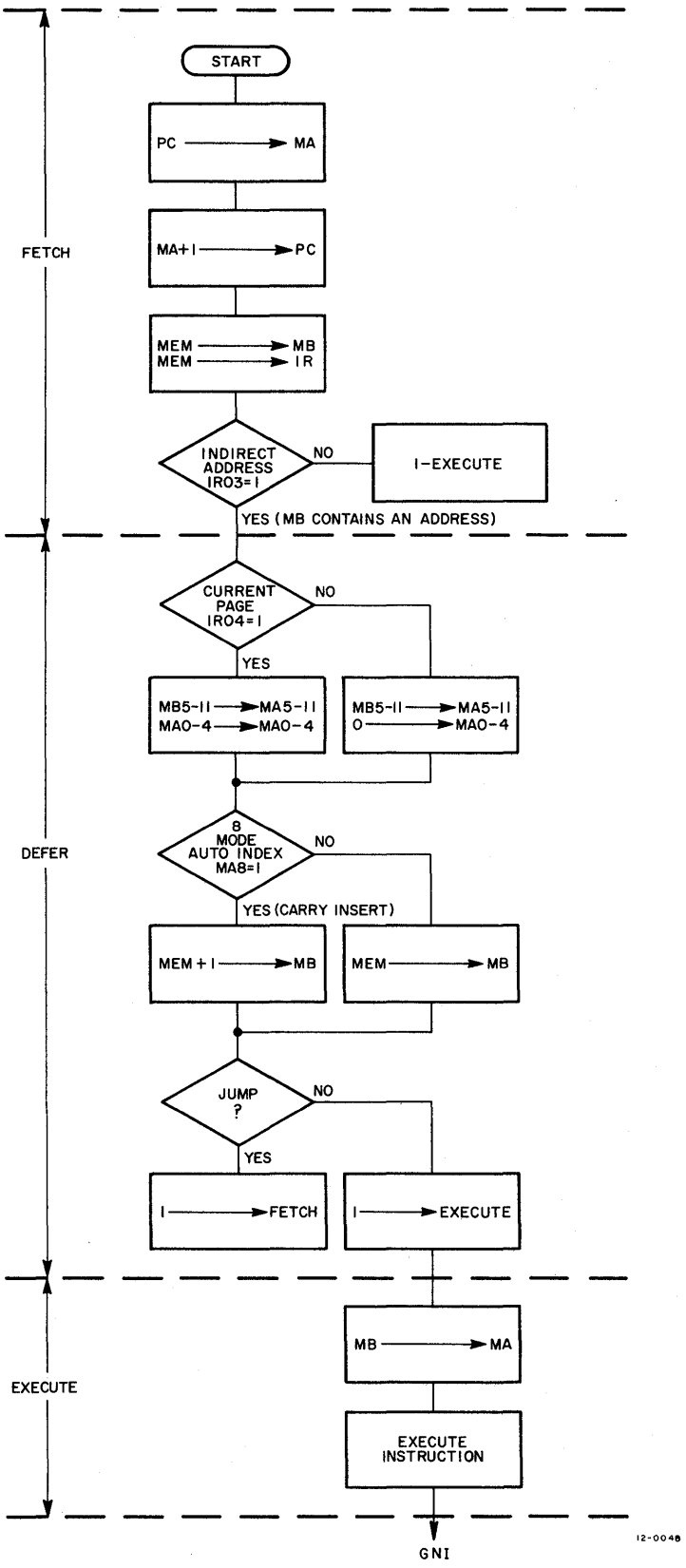


Figure 1-8 8-Mode Address Modification

0-11 are loaded into the MA. During T2 STROBE, memory is read. At TS3, the decoder is checked to determine if the locations in absolute address 0010 through 0017 have been selected for auto-indexing. If auto-indexing is selected, a Carry Insert (dwg. *CYI*) is generated and added during the MEM to MB transfer. If JMP was decoded at TP4, the MB is loaded into the PC, and the computer proceeds to another FETCH cycle. For any other instruction, the EXECUTIVE flip-flop is set, and the EXECUTE Major State is entered.

1.9.3 EXECUTE Cycle

The EXECUTE Major State is entered from either the FETCH or DEFER cycle, depending on the addressing scheme used. If the EXECUTE cycle is entered from the FETCH cycle, IR bit 04 is examined to determine if the current page or page 0 is addressed (similar to operation in the DEFER cycle). When EXECUTE is entered from the DEFER cycle, the MB is loaded into the MA, because it contains the effective address of the operand. During T2, the address specified by the MA is read into the MB as it is for all compute cycles, except the EXC 2 Major State and a SET instruction.

During the remaining time states, the MRI instructions are executed. They are as follows:

a. AND and TAD

b. ISZ

c. DCA

d. JMS

AND (0000) The AND instruction causes a bit-by-bit Boolean AND operation between the contents of the AC and the data word specified by the instruction or MB. During T3, the contents of MEM are loaded into the MB.

During T4 the following functions are enabled and loaded into the AC simultaneously, thus placing $\overline{AC}$ and MB into the AC.

a. $\overline{AC}$, the complement of AC (dwg. *RCC*)

b. BCL, a logical AND of MB (dwg. *RCC*)

The result in the AC is then complemented during T5 by enabling $\overline{AC}$ and loading AC.

The truth table for the logical AND operation is shown below.

MB	AC	Result
1	1	1
1	0	0
0	1	0
0	0	0

TAD (1000) Two's complement add. The logical flow of AND and TAD is identical through T3 of the EXECUTE cycle. During T4, the AC and the MB are enabled and loaded into the AC and LINK to find the sum of AC and MB. The GNI is set to initiate a new FETCH cycle.

ISZ (2000) Increment and Skip if zero. MEM (dwg. *RCB*) and CARRY INSERT (dwg. *CYI*) to bit 11 are enabled during TS3 and loaded into MB at TP3. PRA CARRY 00 (dwg. *SKH*) is tested for Overflow; if Overflow occurs, the SKIP flip-flop is set. At T1 of the next FETCH cycle, the PC is enabled and if the SKIP flip-flop is set, a CARRY INSERT (dwg. *CYI*) is generated to increment the MA.

DCA (3000) Deposit and clear the AC. During T3 the AC is loaded into the MB. The control signals are AC ENABLE (dwg. *RCA*) and MB LOAD (dwg. *RCL*). At T4 the AC is cleared by not providing AC enable and loading AC.

JMS (4000) Jump to subroutine. The instruction stores the pointer address (PC) in the first location of subroutine at T3, enables PC (dwg. *RCC*), and loads the MB (dwg. *RCL*). Control of the program is transferred to the second location of the subroutine at T4 by incrementing the contents of the MA with a CARRY INSERT and loading the PC.

1.9.4 8-Mode Operate Instructions

Operate instructions require only one cycle for completion and can be microprogrammed. The FETCH cycle for T1, T2, and T3 has been described previously. Operate instructions are divided into two classes. An Operate instruction is decoded (dwg. *INS*) by ANDing IR bits 00, 01, and 02, and the output from the MODE flip-flop (dwg. *CPR*). IR bit 03 is examined (dwg. *SLA*), to determine if Operate 1 or Operate 2 class instructions are to be decoded.

Operate 1 Class Instructions – Operate 1 instructions are decoded when IR bit 03 = 0. Operate 1 instructions are primarily used to manipulate the LINK and the AC. MB bits 04 and 06 are used to clear or complement the AC as shown below:

MB04	MB06	Action	Instruction
0	0	AC → AC	NOP
0	1	$\overline{AC}$ → AC	CMA
1	0	0 → AC	CLA
1	1	$\overline{AC}$ + AC → AC	CLA, CMA

CMA (7040) CompleMent the AC is decoded by signals SLA OPR1 H, CPT TS4 (1) H, and PRD MB06 (1) H (dwg. *RCC*), thus enabling AC. The AC is automatically loaded by an Operate 1 instruction during every TP4 (dwg. *RCL*).

CLA (7200) CLear the AC. No Enable signal is provided during TS4, and the AC is loaded during TP4. The effect is that of loading all zeros into the AC, thus clearing the AC.

CLA, CMA (7240) CLear and CoMplement the AC. The result of clearing and complementing the AC is that all AC bits contain binary 1s. This is done simply by ORing AC and $\overline{AC}$ (dwg. *RCL*) and loading the AC.

MB bits 05 and 07 are used to clear or complement the LINK as shown below.

MB05	MB07	Action	Instruction
0	0	L → L	NOP
0	1	L → L	CML
1	0	0 → L	CLL
1	1	L + L → L	CLL, CML

CML (7020) CoMplement the LINK is decoded by SLA MB07 (1) H, and ANDed with SLA (OPR1 · TS4) H (dwg. *FLK*). If the LINK = 0, the AND/OR combination routed to the data side of the LINK flip-flop is disqualified, setting the LINK to 1. If the LINK = 1, -FLK NOT ADDER LINK H is qualified, setting the LINK to 0 when the flip-flop is clocked by RCL AC LOAD H at TP4.

CLL (7100) CLear the LINK is accomplished by inhibiting NOT LINK ENABLE, which qualifies the AND/OR gate routed to the data input of the LINK when RCL AC LOAD H clocks the LINK flip-flop (dwg. *FLK*).

CLL, CML (7120) CLear and CoMplement the LINK results in setting the LINK flip-flop to 1. This is accomplished by inhibiting the AND/OR input of the LINK (dwg. *FLK*).

IAC (7001)	Increment the AC is decoded by SLA OPR1 H · PRF MB 11 (1) H (dwg. *CYI*). This action generates a CARRY INSERT at TP4 to adder bit 11 (dwg. *PRF*). MB08 and MB09 are used to rotate the AC right or left; MB10 determines whether to rotate once or twice.
RAR (7010)	RotAte Right is decoded by PRE MB08 (1) H · SLA OPR SHIFT EN H (RCS) to produce RCS EN SHIFT RIGHT H which is routed to the Load gates (dwgs. *PRA* through *PRF*).
RAL (7004)	RotAte Left is decoded by PRE MB09 (1) H · SLA OPR SHIFT EN H (RCS) to produce RCS EN SHIFT LEFT, which is routed to the Load gates (dwgs. *PRA* through *PRF*).
RTR (7012) and RTL (7006)	Rotate Twice Right and Rotate Twice Left is decoded by SLA OPR1 H · CPT TS5 (1) H PRF MB10 (1) H. This gate (dwg. *SLA*) accommodates another Rotate operation during T5 for a total of two Rotate operations; one during T4 and another during T5.

Operate 2 Class Instructions – Operate 2 class instructions are decoded the same way Operate 1 class instructions are decoded (*i.e.* by ANDing IR bits 00 through 02). IR bit 03 = 1.

SMA (7500)	Skip on a Minus AC (dwg. *SKH*) is executed by testing AC bit 00. This signal ANDed with MB bit 05 sets the SKIP flip-flop at T4 when AC 00 = 01 and MB08 = 0.
SPA (7550)	Skip on a Positive AC is executed by the same gate as SMA, except the signal is ANDed with MB08 1 (H), the reverse sensing bit.
SZA (7440)	Skip on a Zero AC is executed by testing AC bits 00 through 11 for 0s. When MB bit 06 is set and the AC is equal to zero, SKH AC EQ 0 is enabled to set the SKIP flip-flop at T4.
SNA (7450)	Skip on a Non-zero AC. In this case, the same gate that was used for SZA is used. SK H AC EQ0 is not qualified, thereby qualifying the AND/OR combination SLA OPR2 · PRE MB 08 (1) H, which sets the SKIP flip-flop at T4.
SNL (7420)	Skip on a Non-zero LINK is executed by FLK LINK (1) H and PRD MB 07; if the gate is qualified, the SKIP flip-flop is set at T4.
SZL (7430)	Skip on a Zero LINK is decoded by the same gate as SNL, except that it is ANDed with the reverse sensing bit MB 08 (1) H.
OSR (7404)	Inclusive OR of Right Switch Register with AC. OSR is decoded by PRE MB09 (1) L and RCA (OPR2 · TS5) H to enable RSW (RCD). During TS5, the AC is ORed with RSW by enabling SET AC (RCA).
HLT (7402)	The computer stops at the conclusion of the current machine cycle. HLT is decoded by SLA OPR · PRF MB 10 (1) H · CPS FETCH. This signal is routed to the AND/OR combination controlling the data to the RUN flip-flop. CPR HALT H is ANDed with TSS UF (0) H. TSS UF (0) H is routed to +3V if the TSS 12 option is not included. The RUN flip-flop is cleared at TP5 to stop processor timing.

1.10 INTERRUPT

INTERRUPT is checked at T1 of every FETCH cycle. When the program interrupt facility is enabled by the ION instruction (6001), and the Interrupt Request signal is true, the INTERRUPT Major State is entered. The ION instruction is decoded (dwg. *IOC*) by INS IOT · MB bits 03 through 08 · CPTP TP4 H to provide IOC PROC IOT L, which is ANDed with PRF MB 10 (1) H to set the INTERRUPT ENABLE flip-flop. The INTERRUPT ENABLE

flip-flop sets the INTERRUPT DELAY flip-flop. IOC INTERRUPT DELAY (0) H is ANDed with other signals to set the INTERRUPT SYNC flip-flop (dwg. *CPS*). CPS INTERRUPT SYNC (1) is ANDed with -CPS BREAK SET H, -CPS TAPE SET H, CPS GNI H to provide CPS INTERRUPT SET L to set the INTERRUPT flip-flop.

In the INTERRUPT cycle during TS1, the DF and IF are transferred to the save field register by CPS INTERRUPT SET H · CPT TS1 (1) H, which provide MEA LOAD SF L. MEA LOAD SF L or · MPG LINC SET L to provide MPG LOAD SF (dwg. *MPG*), which clocks the SAVE FIELD flip-flops and loads the IF and DF into the SF. The IF and IB flip-flops are direct-cleared at this point by MPG-0-IF-IB03-04 L, and the DF flip-flops are cleared by MPG 0-DF03-04 L (dwg. *MPG*). This action is necessary because all interrupts are trapped to the IF 0.

If the INTERRUPT occurs in the PDP Mode, the MA is cleared (location 0000) by inhibiting the enable gates and loading the MA with all 0s. The INTERRUPT ENABLE flip-flop is also direct-cleared (dwg. *IOC*) (see Paragraph 1.21 for LINC Mode interrupts).

1.11 LINC MODE INSTRUCTIONS

LINC-Mode instructions are grouped in the following manner according to their addressing schemes and interrelationships. A block diagram for LINC-Mode addressing is shown in Figure 1-9.

a. Direct Address

b. β Class Instructions

c. α Class Instructions

d. MSC Instructions

e. Rotate and Shift Instructions

f. Skip Instructions

g. Operate Instructions

h. LINCtape Instructions

i. Trap Instructions and Interrupt

1.11.1 LINC FETCH

LINC instructions are identical during T1 through T3 of the FETCH cycle. If the DO flip-flop is not set at T1, the PC is enabled and loaded into MA2 through 11. If the SKIP flip-flop is set, the MA is incremented by CARRY INSERT. The current field (IF03 and 04) is also loaded into MA00 and 01. At T2, the Program Counter is incremented by CYI PC INCREMENT and FETCH (dwg. *CYI*). The PC is loaded by CYI PC INCREMENT and CPTP TP2 (dwg. *RCL*). The IR is cleared by CPTP TP2 (dwg. *INR*). During TS3, MEM is enabled and loaded into the MB by RCB GO MEM TS3H · CPS ENTER (0) H · CPT TS3 (1) and CST EN MEM (1) H (dwg. *RCB*). The MB is loaded by EXC 2 (0) H and TP3 H. The IR is loaded every TP3 during a FETCH cycle.

If the DO flip-flop is set, at T1, the MA is not incremented by the SKIP flip-flop, because the EN MEM flip-flop is direct-cleared by CST SET DO L. A CARRY INSERT for a SKIP is only possible when CPS FETCH SET H · CST EN MEM (1) H are true (dwg. *CYI*). At T2, the PC is also not incremented because the EN MEM flip-flop is not set. During T2, the IR is cleared, as is the case in all FETCH cycles during T2. At T3, LSW are enabled by RCC EN LSW. The switch inputs are routed to enable gates (dwgs. *PRA* through *PRF*) and loaded into MB and IR simultaneously.

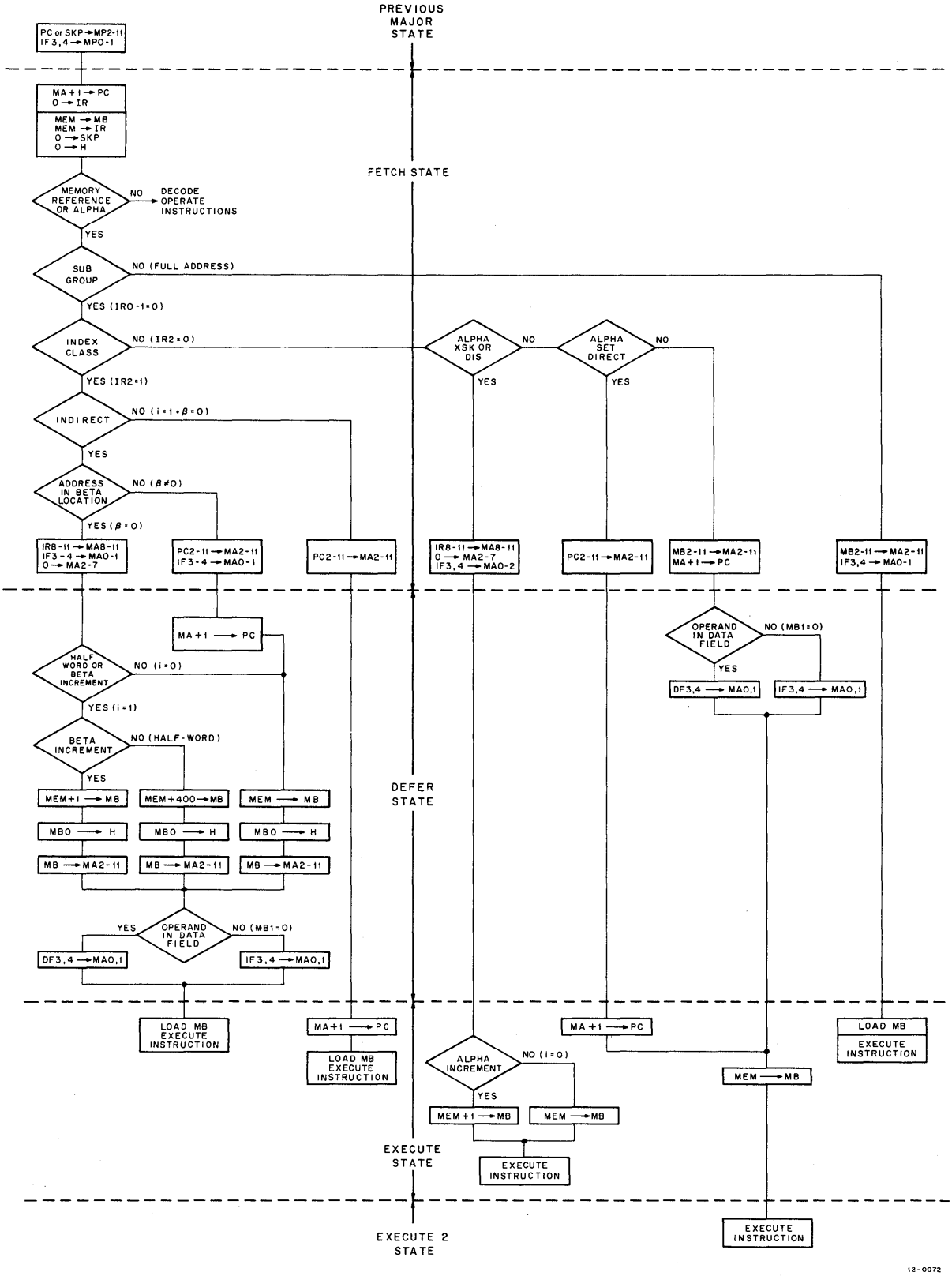


Figure 1-9 LINC Mode Address Modification

1.11.2 Full Address Instructions

MB Bits 00 and 01 indicate the operation to be performed, while bits 02 through 11 contain the address of the operand. The Full Address instructions are limited to the current instruction field (IF) addresses 0000 through 1777. Full Address instructions are decoded from IR bits 00 and 01 into 2000 (ADD), 4000 (STC), and 6000 (JMP) (dwg. *INS*).

The EXECUTE Major State for ADD and STC is identical for both instructions during T1. MB02 through 11 are enabled (RCB) and loaded into MA. MA is always loaded during TP1. IF 3 and 4 are enabled (dwg. *RCB*) and loaded into bits 00 and 01 of the MA to stay in the current field. Memory is strobed during T2.

ADD Instruction – For the ADD instruction, MEM is enabled and loaded into MB during T3 of the EXECUTE cycle. During T4, MEM and AC are enabled and loaded into the AC to add the contents of MB and AC, leaving the sum in the AC.

STC Instruction – For the STC instruction, AC is enabled and loaded into MB during T3 of the EXECUTE cycle, thus storing the contents of the AC in the MB; the AC is cleared during T4.

JMP (6000) Instruction – The LINC JMP instruction is used to change program sequence, and also, in conjunction with the LIF instruction, to change the instruction fields. The DJR instruction is also used in conjunction with the JMP instruction. The DJR instruction clears the SAVE PC flip-flop, which disables PC in TS3 of the EXECUTE Major State and enables MEM. This procedure ensures that location 0000 will not be changed when the next (*and only the next*) LINC-Mode JMP is used. This operation is useful because an Interrupt may occur within a LINC-Mode subroutine that uses location 0000 of the current IF to retain the subroutine return; therefore, it must not be destroyed. The JMP is further decoded in TS4 of the FETCH cycle to provide X = 0 and X ≠ 0 (dwg. *INS*). When X = 0, and, if the SAVE PC flip-flop is set, the next instruction is fetched from address 0000 of the current memory field. When X ≠ 0, IB 00 through 04 are transferred to IF 00 through 04, by MPG LOAD IF03 – 04 of TP4 of the FETCH cycle, the IF register is loaded with the contents of the IB register.

During the EXECUTE Major State at T1, MA2 through 11 is cleared and IF 03 and 04 are loaded into MA00 and 01. Memory is strobed in T2, and the status of the SAVE PC flip-flop determines whether PC or MEM is enabled at T3. At TP3, with the SAVE PC flip-flop set, PC 02 through 11 is loaded into MB 02 through 11, and MB bits 00 and 01 are set with RCC SET PC0-1. At T4, the SAVE PC flip-flop is set. This accomplishes the storing of the contents of the PC + 6000 in location 0000 of the current memory field.

The operand address of the JMP instruction is placed in the PC at T5 enabling IR 2 through 11, which enables IF 03–04 to PC00–01 and loads the PC.

1.11.3 Index Class Instructions

LINC instructions, with IR bits 0 and 1 on a zero and IR bit 2 on a one, are known as Index Class instructions. There are four methods of addressing, called Iβ, with the Index Class instructions. The four methods are as follows, where I = IR07, $1 \leqslant \beta \leqslant 17$.

1. I = 0 β = 0, operand address is in next location.
2. I = 0 $\beta \neq 0$, operand address is in β register.
3. I = 1 β = 0, operand is in next location.
4. I = 1 $\beta \neq 0$, operand address -1 in β register. β register is incremented by one during instruction execution.

All four methods have identical functions in the FETCH cycle of the instruction. For 1, 2, and 4 the DEFER cycle is entered after the FETCH cycle. The signal CPS DEFER SET L is true because INS LINC INDIR H is true; therefore, the DEFER flip-flop is set at TP1. For I = 0, β = 0 the signal SLA2 WORD FORMAT H is true, thus qualifying the M115 NAND gate with output pin J1 which yields RCC EN PC 2-11 H, thus transferring the PC to the MA at TP1. The contents of MEM are loaded into the MB at TP3 and then transferred to the MA at TP1 of the EXECUTE cycle, because RCB TS1 EN MB L is true and RCB EN MB 2 through 4 H and RCB EN MB 5 through 11 H are generated.

For I = 0, $\beta \neq 0$ and I = 1, $\beta \neq 0$ bits 8 through 11 of the IR are transferred to MA08 through 11 and MA02 through 07 are cleared. This is accomplished with the signal INS (L INDEX * B EQ 0) L which gives RCA ENABLE IR 08 through 11 H. At TP3 MEM is transferred to the MB and if I = 1 CYI CARRY INSERT L is generated and the MB is indexed by one. The MB is transferred to the MA at TP1 of the EXECUTE cycle, thus giving the address of the operand.

For I = 1, β = 0 the signal SPS DEFER SET L is not true and the EXECUTE cycle is entered after the FETCH cycle. PC 02 through 11 is transferred to MA 02 through 11 at TP1 of the EXECUTE cycle.

The functions for Index Class instructions, except MUL, are executed at T3 through T5 of the EXECUTE cycle and are listed below:

Instruction	Octal Code	Description
LDA	1000	LoaD the Accumulator. At TP3, MEM is loaded into MB, and MB is loaded into AC at TP4, thus loading the contents of memory into the AC.
STA	1040	STore the Accumulator. At TS3, AC is enabled by RCB (STC · DCA · STA) (dwg. *RCA*) and is loaded into MB at TP3 for storage in memory.
ADA	1100	ADd to the Accumulator. At TP3, MEM is loaded into MB, and at TS4 MB is enabled on one side of the adder and AC on the other; consequently, the contents of memory are added to the AC by loading AC at TP4.
ADM	1140	ADd to Memory. At TS3, MEM and AC are enabled as shown (dwgs. *RCA* and *RCB*) in the same manner as in the ADA instruction; however, both MB and AC are loaded at TP3 to place the summation of the memory and the AC into the memory and AC. The FLOW flip-flop is set (dwg. *FLE*) if an arithmetic overflow occurs.
LAM	1200	Link Add to Memory. For the LAM instruction the LINC, MEM, and AC are enabled at TS3 and are loaded into MB and AC at TP3. The LAM instruction is similar to the ADM instruction, except the LINK is enabled to AC11 via CARRY INSERT (CYI).
		NOTE **For the BCL, BSE, BCO, and SAE instruction, MEM is enabled at TS3 and loaded into MB during TP3.**
BCL	1540	The BCL instruction is a logical AND of MB and AC. Bit Clear is decoded by INS BCL · SLA TS4 to provide an enable gate to the adder, using the signal RCC EN BCL H. The AC is enabled by RCC BSL EN AC L (dwg. *RCA*). The AC is loaded at TP4.
BSE	1600	The BSE instruction is an inclusive OR between the AC and the MB. Bit Set is decoded by INS BSE · SLA TS4 to provide an enable gate to the adder, using the signal RCC EN BSE. The AC is loaded at TP4 (dwg. *RCL*).

Instruction	Octal Code	Description
BCO	1640	The BCO instruction is an exclusive OR of MB and AC. BCO is decoded by SLA TS4 H · INS BCO to provide RCC EN BCL L and RCC EN BSE L. The AC is loaded at TP4.
SAE	1440	Skip if the Accumulator is Equal to the operand. The SAE instruction is decoded by SKL AC EQ MB H · INS SAE H to provide SKL LINC EX SKIP H to both the data and the clock input of the SKIP flip-flops.
SRO	1500	Rotate and Skip. The memory is loaded into the MB and rotated one place to the right to bring MB bit 11 into MB bit 00 at T3. SRO is decoded by INS SRO H · SLA EX TS3 H to provide RCS EN RIGHT SHIFT H, which is routed to the inputs on the adders. At TP5, the SKIP flip-flop is clocked by SKL LINC EXC SKIP H if INS SRO and PRA MB00 (0) are high.
LDH	1300	LoaD AC with Half-word. At TS3, MEM is enabled and loaded into MB to retain the complete word in memory. The AC is also loaded by INS LDH L · CPS EXECUTE B (1) H · CPTP TP3 H (dwg. *RCL*). At TS4, if the H flip-flop is 0, RCD EN AC N- 6 R H is enabled, which is ANDed with PRA AC 01 (1) H to provide PMB TMSC 07. Bit 02 is routed to TMSC bit 8, and the process continues. The AC is loaded at TP4, thus transferring the left-half of the AC to right-half of the AC, leaving the left half cleared. When H = 1, INS LDH H · SLA EX TS4 H · SKH H (1) H provide RCA EN AC 6-11 (dwg. *RCA*), which retains the right-half of the AC and clears the left-half with the Load AC pulse.
STH	1340	STore Half-word from AC. When the H flip-flop is 0, the right side of the AC is enabled by INS STH H · SKH H = (1) H (dwg. *RCA*) and is loaded into the left side of the MB by TMA TMSC 00 through 05 (dwg. *PMA*). RCB EN MEM 6-11 H is also enabled to retain the right- half of the word.
ST		When the H flip-flop is 1, AC 6 through 11 is enabled by INS STH H · SKH (1) H and is loaded into MB. RCB EN MEM 0 through 5 is also enabled to retain the left-half of the word in memory.
SHD	1400	Skip if Half-word Differs. At TS3, MEM is enabled and loaded into MB. When H = 0, at TP4 the right and left side of the AC are swapped by routing signals 6 through 11 to 0 through 5 of Processor Miscellaneous A (dwg. *PMA*) and routine 0 through 5 to 6 through 11 of Processor Miscellaneous B (dwg. *PMB*). At TP5, if INS SHD H · SKH (0) H · -SKL ACL EQ MBL H are true, the signal SKL LINC EX SKIP H is generated (dwg. *SKL*), setting the SKIP Flip-flop (dwg. *SKH*). The right and left half are swapped again at TP5 to provide the original word. When H = 1, if INS SHD · SKH (1) H · -SKL ACR EQ MBR are true, the signal SKL LINC EX SKIP is also generated, setting the SKIP flip-flop at TP5.
DSC	1740	DiSplay Character; at TS3, MEM is enabled and loaded into MB. If the display is busy, the T5 timing chain is inhibited by -CPT EN INT PAUSE H (dwg. *CPTP*), which is set low by DSC BUSY (dwg. *CPT*). DSC BUSY · CPTP TP3 also direct-sets the INTERNAL PAUSE flip-flop. When DSC BUSY goes low, -CPT EN INT PAUSE is high and initiates the T5 timing chain, thus clearing the INTERNAL PAUSE flip-flop at the generation of

Instruction	Octal Code	Description
DSC (Cont)		the next OFF PAUSE pulse. When the display is not busy, MB is loaded into the intensification register (dwg. *DSI*) by DSC LD IN H, (dwg. *DSC*). The AC is loaded into the vertical register (dwg. *DSY*) by DSC LD VERT H. The VERTICAL flip-flop VA7 and VA8 (dwg. *DSC*) and VA9 and VA10 (dwg. *DSY*) are cleared at T5 by DSC INITIALIZE L to provide automatic spacing. The HORIZONTAL flip-flop HA9 is direct-set, and HA10 is direct-cleared by DSC INITIALIZE L to move four points over from the starting point of the display. The EXEC 2 Major State is now entered by CPS EXECUTE B (1) H · SLA (DSC · SET) H (dwg. *CSP*). At TP1 of the EXEC 2 Major State, the MA is set to a one by CARRY INSERT, which is enabled by CPS EXECUTE (1) H · INS DSC H · CPT TS1 (1) H, to obtain the next address at address 0001 of the current field for the horizontal axis. At TP2, the AC is incremented by 30_8 (bits 07 and 08 are set to 1). 1). This is done by signals PRD AC 07 (1) L · DSC SIZE (1) L, which provide PMB TMSC 07 H, and signals SLA DSC CNT MB H · PMA SIZE (1) H, which provide PMB TMSC 08 H. The two resulting signals are routed to the enable gates of the adders. For half-size characters, 14_8 (bits 08 and 09) is added to the AC by enabling PMB BMSC 08 and PMB BMS 09 H. For full-size characters, at T3 the summation of MEM and 10_8 (bit 08 = 1) is enabled by RCA SET AC H and provides PMB BMSC 08 H. The summation is loaded by -INS SET H · CPTP TP3 H (dwg. *RCL*). For half-size characters, PMB BMSC 09 H is enabled by RCA DSC SET AC H · DSC SIZE 0 H. At T5, the VERTICAL flip-flops V8 through 11 (dwg. *DSY*) are zeroed by DSC LD VTL and DSC CLR V8-11 L, which are enabled by CPT TS5 B (1) H · SLA (DSC · EXECUTE 2) H (dwg. *DSC*). The DSC ACTIVE flip-flop is also set by DSC LD HORZ H. The CP timing can now continue with another timing cycle while the timing chain (dwg. *DSC*) is started to complete the DSC instruction. The DSC instruction and the associated circuitry are described in further detail in the Display Control section of Chapter 5.
MUL	1240	The contents of the AC (multiplicand) are MULtiplied by the contents of the register Y (multiplier). The product is left in the AC and the MQ; the sign of the product appears in the LINK and AC_0. At T3, MEM is enabled (dwg. *RCB*) and loaded into the MB by CPTP TP3 · -INS SET, causing the multiplier to be loaded into the MB. The down-counter (IR bits 08 through 11) are zeroed by CPTP TP3 · SLA (MUL · EXECUTE) H (dwg. *INR*). At T4, the multiplicand is checked for the sign by checking AC bit 00. When AC bit 00 = 0, RCS · EN AC TO MQ H loads AC into the MQ, and the sign bit is placed in the LINK for future reference by signal INS MUL GO · RCS EN AC TO MQ H. When AC bit 00 = 1, RCS EN – AC TO MQ H loads AC into the MQ, and the LINK is set to 1. The RECYCLE SYNC flip-flop (dwg. *CPT*) data input is set high by SLA (MUL · EXECUTE TS4) L, and CPTP 5 H clocks the flip-flop. The RECYCLE SYNC flip-flop is set, and high signal is sent to the data input of the T5 RECYCLE flip-flop, which is clocked 50 ns later by CPTP OFF PAUSE. The T5 RECYCLE is initiated. For the duration of the MUL instruction, T5 RECYCLE remains set, resulting in repeated generation of the T5 timing chain (dwg. *CPTP*).

Instruction	Octal Code	Description
MUL (Cont)		MB is now loaded into the AC to check the sign. In the down counter, IR bits 08 through 11 are set to 1 for a value of 17_8.
		The IR is set to 17 by signals SLA MUL GO H · -INS N EQ 01 H, which produce INR COUNT ENAB H (dwg. *INR*). INR COUNT ENAB H is used to enable the four IR bits. INR IR 11 (0) H sets bit 11, INR IR 10 H sets bit 10, INR IR 09 · INR CARRY 10 L sets bit 09, and INS N EQ 00 sets bit 08. This N counter is decremented at the occurrence of the TP5D pulse.
		When N = 17, the sign of the multiplier is checked. If the AC bit 00 = 0, the AC is enabled and loaded into the MB and AC simultaneously; the LINK remains unchanged. If AC bit 00 = 1, RCC T5 COMP AC L is used to load AC into MB and AC simultaneously, and the LINK is complemented. When N = 16, the AC is cleared, because it contained the multiplier from the previous operation and it must be 0000 initially when the multiplication begins in the next cycle. The down counter is again decremented.
		Multiplication begins when N = 15; the AC is always enabled by SLA MUL ADD L · CPT TS5. The MB is also enabled when MQ bit 11 = 0 by SLA MUL ADD H · MQR MQ (1) H. RCS EN MQ SH RIGHT H is also activated, from N = 15 until N = 3, and the AC is loaded at every TP5. The result is that the AC is loaded into the AC and shifted right one place whenever MQ bit 11 = 0, and the summation of MB and AC is loaded into AC and shifted right one place whenever MQ bit 11 = 1. The partial product of AC bit 11 is moved into AC bit 00, and the MQ is rotated right one place. This operation is repeated ten times from N = 15 to N = 3.
		When N = 2, the AC is left unchanged by loading AC into the AC for a fractional multiplication indicated by H = 1, or MQ is loaded into the AC and shifted right one more place for an integer multiplication indicated by H = 0. When N = 1, the AC is left unchanged by loading AC into AC if the LINK = 0, or the AC is complemented if the LINK = 1. The RECYCLE SYNC flip-flop is cleared and, at the next OFF PAUSE pulse, T5 RECYCLE is cleared initiating another Major State.

1.12 *a* CLASS INSTRUCTIONS

There are three *a* class instructions; SET, DIS, XSK; each instruction uses registers 0000 through 0017 in a unique way. The following paragraphs contain a brief description of the instructions and a detailed description of the flow diagrams associated with each instruction.

$a = 0 \leqslant a \leqslant 17$

1.12.1 SET Instruction 0040 + I + *a*

Set the contents of memory register *a* equal to the contents of memory register Y.

ADDRESSING FOR SET:

I	*a*	Y
0	$0 \leqslant \beta \leqslant 17$	Y (P+1)
1	$0 \leqslant \beta \leqslant 17$	P+1

For I = 0, the DEFER state is entered after the FETCH state. CPS FETCH (1) H * INR IR 07 (0) H * INS SET H are true, making CPS DEFER SET L true. The operand address is loaded into the MB at TP3. The MB

is transferred to the MA at TP1 of the EXECUTE cycle, and the operand is loaded into the MB at TP3. When I = 1, the MA is loaded with the contents of the PC at TP1 of the EXECUTE cycle and the operand is loaded into the MB at TP3.

The EXEC 2 state is always entered for the SET instruction. CPS EXC 2 SET L is true at TP1, thus enabling the data input to the flip-flop. At TP1, bits 8 through 11 of the IR are transferred to MA 8 through 11, and MA bits 2 through 7 are cleared. IF 3 and 4 are loaded into MA 0-1, retaining the same memory bank. The RCL LOAD MB L pulse is inhibited for this cycle; therefore, the contents of the MB are written into memory at TP3.

1.12.2 XSK Instruction (Index and Skip) 0200 + I + *a*

At TP1 of the EXECUTE cycle, the *a* register IR08-11 is loaded into the MA 8 through 11, MA 2 through 7 are cleared, and the current field is loaded into MA 00 and 01. The *a* register is strobed at TP2, and at TP3 MEM is loaded into MB for I = 0; MEM is incremented by CARRY INSERT for I = 1 and loaded into the MB. At TP5, if the Skip condition is met, PRA CARRY 02 H is generated and is ANDed with INS XSK H (dwg. *SKL*). This signal provides SKL LINC EX SKIP, which sets the SKIP flip-flop.

1.12.3 DIS (Display) 0140 + I + *a*

A point is intensified on the display with the horizontal position specified by bits 3 through 11 of the *a* register, and the vertical position specified by bits 3 through 11 of the AC. The EXECUTE cycle is always entered after the FETCH cycle for this instruction. MB 00 is loaded into the channel flip-flop at TP5 to select one of two channels. If the display has not completed the previous display instruction, the signal CPT EN INT PAUSE L is true and the CP timing will pause in time state 5 (CPT TS5 (1)) until DSC BUSY becomes false. This allows TP5 to occur and the necessary load pulses are generated to perform the display. The M711 Scope Control Module generates all the clock and load pulses to complete the instruction.

DSC INITIALIZE L is the first pulse generated, yielding the DSC LD VERT H, DSC LD HORZ H, and DSC SET IN 11 L pulses. The DSC LD HORZ H pulse starts the 25 μs setup delay and sets the ACTIVE flip-flop. When the delay has timed out an intensify pulse (DS CINTEN) is generated, and the ACTIVE flip-flop is cleared, thus signifying the completion of the display.

The following instructions are not *a* class instructions; however, they are used in conjunction with memory and, consequently, will be discussed here.

Instruction	Octal Code	Description
LIF	060600+N	$0 \leqslant N \leqslant 37$. This instruction sets the instruction field to the value "N". It is utilized in conjunction with the LINC JMP instruction – the LIF instruction loads the IB register with the value "N", and the next LJMP instruction transfers the IB to the IF register. The LIF signal is ANDed with CPTP TP4 to give MPG LINC SET L, which generates the Load IB pulse. The LINC SET pulse also generates a MPG LOAD SFL pulse, which transfers the contents of the IF to the save field register.
LDF	0640+N	$0 \leqslant N \leqslant 37$. This instruction loads the data field register with the value N. This signal LDF is gated with CPS FETCH (1) and CPTP5 to generate MPG LINC SET DF L. This pulse clocks DF03 L4 and also generates MXF LOAD DF0-2 H.

1.12.4 SAM 0100+N ($0 \leqslant N \leqslant 37$)

The SAM instruction samples one of 32 channels, and converts the analog voltage on that channel to a ten-bit binary number that is contained in the AC at the completion of the instruction. This is a single-cycle instruction; however, to allow sufficient time for the conversion, the timing chain is recycled in TS5 when the fast sample option is not selected.

The SAM level is decoded at TS4 and is ANDed with YAD A-D PAUSE (1) and YADC FAST SAM (0). When these three signals are true, YADC A-D RECYCLE L is generated and the CPT RECYCLE SYNC flip-flop is set at TP4. Bits 07-11 of the IR are loaded into YADC IR 07-11 at TP4 to select the channel. YADC LOAD A-D IR also initiates a 6 μs delay to allow the analog voltage to settle and, at the completion of the delay, the conversion is started with the YADC START pulse. The YADC DONE pulse is generated when the conversion is complete and the YADC A-D PAUSE flip-flop is cleared. This disables the YADC A/D RECYCLE level; the RECYCLE SYNC and T5 RECYCLE flip-flops are also cleared with the next TP5 pulse. The AC is loaded with the contents of the A/D register at every TP5; thus, the final converted number is transferred with the last LOAD AC pulse.

1.13 MISCELLANEOUS INSTRUCTIONS

MSC Instructions – This LINC-Mode instruction set contains single-cycle Operate instructions similar to the 8 Mode. The instructions are classified MSC on LINC FETCH 1A (dwg. D-FD-PDP-12-0-12) and are labeled N = 0 through N = 17. The MSC signal is true when IR0-6 are on a zero. The operations of the MSC instructions are as follows:

a. N = 0 performs a HLT (0000). The RUN flip-flop is cleared (dwg. *CPR*) by INS MSC H · INS NEQ 00 H · CPS FETCH (1) H and TP5.

b. N = 1 is used to decode an AXO (0021) or XOA (0001) instruction, depending on the state of the I-bit. This instruction is discussed in the LINCtape processor chapter.

c. N = 2 (0002) is used to perform the PDP instruction. The PDP instruction changes the MODE flip-flop to 8 Mode (dwg. *CPR*). The instruction is executed by CPTP TP1 H · INS MSC H · INS NEQ 02 H.

d. N = 3 is used to perform the TAC or the TMA instruction. This instruction is discussed in Paragraph 6.6.1 (LINCtape Control).

e. N = 4 and I = 0 are used in conjunction with AC bits 2 through 7 to decode an ESF (0004) instruction.

(1) ESF and AC bit 2 (1) set the TRAP flip-flop (dwg. *CPS*), which allows the use of LINC-8 programs or other undefined instructions (refer to Paragraph 1.19).

(2) ESF and AC bit 3 (1) set the TAPE TRAP flip-flop (dwg. *RCD*), which allows the user to inhibit all tape instructions and interrupt the main program when the EN TRAP flip-flop is set.

(3) ESF and AC bit 4 (1) sets the DSC SIZE flip-flop (DSC) for half-size characters to be displayed on the VR12.

(4) ESF and AC bit 5 (1) sets the YADC FAST SAMPLE flip-flop (YADC) for the Fast Sample mode of analog inputs.

(5) ESF and AC bit 6 (1) sets the ENABLE TELETYPE INTERRUPT flip-flop (dwg. *RCA*) to disable the program interrupt when the Teletype flag is set, even if the interrupt facility is enabled.

(6) ESF and AC bit 7 (1) generates an I/O preset to clear all device flags.

f. N = 4 and I = 1 are used to decode and execute the SFA (0024) instruction. N = 4 and I = 1 are decoded by INS MSC H · INR IR 07 (1) H · INS NEQ 04 H to produce RCA EN LD MSC 4. This instruction places the contents of the special function flip-flops into the AC. AC bits 0 through 1 and 7 through 11 are cleared. This instruction is used to check the status of the special functions register.

g. N = 5 is decoded into a QAC (0005) instruction. The contents of MQ 0 through 10 are placed in AC 1 through 11, and AC 0 is cleared. The instruction is executed by SLA (MSC · TS5) · INS EQ 5 H to provide RCC EN MQ H. AC is then loaded by INS N EQ5 L (dwg. *RCL*). RCC EN MQ and CPR L Mode provide RCS EN SHIFT RIGHT H.

h. N = 6 (0006) is decoded into a DJR instruction which direct-clears the SAVE PC flip-flop (dwg. *RCD*). The DJR instruction has been discussed in conjunction with the JMP instruction.

i. N = 11 is decoded into a CLR (0011) instruction. The AC is cleared by disabling the register bus and loading AC, using the signal INS EQ 11 L (dwg. *RCL*). The LINK is cleared by signal INS MSC H · INS N EQ 11 H (dwg. *FLK*), and the MQ is cleared by loading MQ (dwg. *RCL*).

j. N = 14 is used for an ATR (0014), AC to Relays instruction. The contents of AC 6 through 11 are transferred to the relay buffer. The instruction is decoded by INS MSC H · INS NEQ 14 H · CPTP5 H (dwg. *IOR*) to provide the signal IOR LOAD RELAY L that clocks the RELAY BUFFER flip-flops.

k. N = 15 is decoded into an RTA (0015), Relays to AC instruction. The contents of the relay buffer are transferred into AC 6 through 11. The instruction is decoded by signal SLA (MSC · TS5) L and INS NEQ 15 L to provide the signal RCD EN RELAYS H (dwg. *RCD*). The 1 side of the RELAY flip-flops is routed through processor miscellaneous B gates, and the AC is loaded by the signal N EQ 15 L and CPTP TP5 (dwg. *RCL*).

l. N = 17 is decoded into COM (0017) complement the Accumulator instruction. The instruction is decoded by SLA (MSC · TS5) H · INS NEQ 17 H to provide the signal RCC T5 COMP AC L. The AC is loaded by the signal INS NEQ 17 and CPTP TP5 (dwg. *RCA*).

1.14 SHIFT AND ROTATE INSTRUCTIONS

The Shift and Rotate instructions in the LINC Mode are ROR, SCR, and ROL. The operations for all these instructions are identical for TS1 through TS3 in the LINC FETCH cycle. The contents of the AC and MQ are shifted or scaled "N" places. ($0 \leqslant N \leqslant 17$).

At T4 of FETCH 1B, N is examined for 0. If N = 0, no operation is performed, and the CP proceeds to fetch the next instruction.

When $N \neq 0$ and either ROR, ROL, or SCR is decoded, signal SLA L ROTATES is produced (dwg. *SLA*). SLA L ROTATES produces the signal INR COUNT ENAB H which is ANDed with TP4, to direct-set the RECYCLE-SYNC flip-flop (dwg. *CPT*). CPTP OFF PAUSE is generated (dwg. *CPTP*), 50 ns after TP4, which sets the T5 RECYCLE flip-flop.

The following list is a description of LINC Mode Shift and Rotate instructions:

ROR (030 n) — The contents of the AC and the MQ are Rotated to the Right *N* places. At TS5, AC is enabled by SLA ROTATES (dwg. *RCA*). SHIFT RIGHT is enabled by SLA (ROR + SCR) and CPT TS5 B (1) H. RCL LOAD AC H and RCL LOAD MQ H are generated at CPTP5. The Link is included in the shift when IR bit 07 = 1 (dwg. *FLK*). ADDER bit 11 is rotated back to LINK (dwg. *FLE*).

SCR (034 n) — The contents of the AC and the MQ are Scaled Right *N* places. The sign bit (contents of ACC0) is not changed. AC is enabled and rotated into the MQ in the same manner as accomplished in the ROR instruction. If I = 0 the Link is not changed. If I = 1 AC11 is shifted into the Link.

ROL (024 n) — The contents of the AC are ROtated to the Left *N* places. If I = 1, the Link is shifted into the AC11. The AC is enabled (RCB) by SLA ROTATES and loaded at CPTP TP5. RCS EN SHIFT LEFT H is enabled by INS ROL and CPTP TP5 H (dwg. *RCS*). FLE LOW END SHIFT IN is enabled by the ROL instruction IR 07 (0) and AC 00 (0).

1.15 SKIP INSTRUCTIONS

The Skip instructions are one-cycle instructions. When the Skip condition is true and IR 07 = 0, the SKH SKIP flip-flop is set and the next sequential instruction is skipped. When IR 07 = 1 the inverse occurs: the SKIP flip-flop is set when the Skip condition is not true.

The Skip instructions in the LINC Mode are SXL, SNS, SKP, APO, AZE, LZE, QLZ, and FLO.

SXL (040 n) — Skip On eXternal Level, N. The external levels for SKL X SENSE L associated with the SXL instructions are shown on the SKL print. The AND/OR combination M141 Module at location K37 (dwg. *SKH*), decodes the XL levels.

The following Skip conditions are decoded by INS SKP and NEQ 0 through 05 and N EQ 11, 12, 14, 15 and 16 (dwg. *SKH*). Each of these gates when qualified produces the signal SKL I SENSE L, which sets the SKIP flip-flop.

SNS (044 n) — SeNse Switch inputs CSI SNS 00 through 05 are ANDed with INS N EQ 00 L through 05 to provide SKL I SENSE L.

SKP (0456) — SKP unconditionally. When this instruction is executed, the SKL I SENSE level is always true.

APO (0451) — Skip if Accumulator is POsitive by testing the contents of AC bit 0. This instruction is decoded by INS N EQ 11 · PRA AC 00 (0) H (dwg. *SKL*).

AZE (0450) — Skip if Accumulator is 0000 or 7777 (-0). SKH AC EQ 0 L and -CYI CARRY OK L are ORed together and the output signal is ANDed with INS N EQ 10 to provide SKL I SENSE L.

LZE (0452) — Skip if the Link is 0. The LINK bit is ANDed with N = 12 to generate SKL I SENSE L.

FLO (0454) — Skip if the Flow flip-flop is set. This instruction is executed by FLE FLOW (1) H · INS NEQ 14 (dwg. *SKL*).

QLZ (0455) — Skip if MQ 11 = 0. MQR MQ 11 (0) H is ANDed with N = 15 to generate SKL I SENSE L.

1.16 OPERATE INSTRUCTIONS

1.16.1 IOB

IOB is decoded with INS IOT L by INS LOPR H · INS N EQ 00 H · CPS EXECUTE (1) H. The IOB instruction permits one 8-Mode IOT instruction to be executed after the IOB instruction is given. The octal code for the IOT instruction is located in the next sequential memory location.

The INS IOT level is generated by ANDing the signals INS L OPR H, INS N EQ 00, and CPS EXECUTE (1). The CPT IOT PAUSE flip-flop is set by CPTP TP4 and IOT, and the I/O timing chain is initiated. The IOT timing is shown in Figure 1-10.

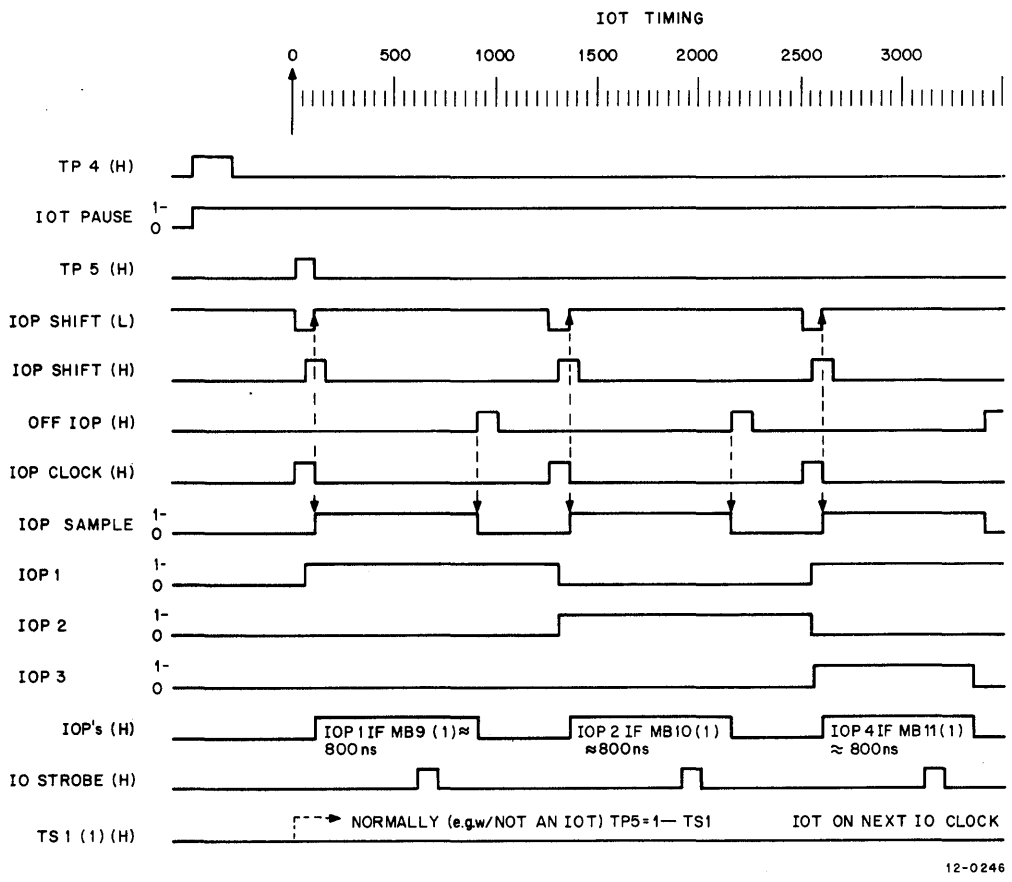


Figure 1-10 IOT Timing

The logic for the IOP generator consists of a re-entrant delay chain which generates three time states. These time states are gated with MB bits 11, 10, and 9 to generate IOP 1, IOP 2, and IOP 4 respectively. CPTP5 and CPT IOT PAUSE (1) generate IOC IOP SHIFT. The leading edge of the shift pulse sets IOP FF1, and the trailing edge sets IO SAMPLE. Approximately 500 ns later, I/O STROBE is generated if IOB 11 = 1. This pulse is used to generate the load pulses to load data from the I/O Bus. IOC OFF IOP is generated approximately another 200 ns later and clears IOC IOP SAMPLE. IOC IO CLOCK is generated another 100 ns later, this pulse restarts the timing generator if IOT PAUSE = 1. The next SHIFT PULSE sets IOC IOP FF2 and clears IOP FF1. The timing sequence continues for two more timing cycles. The signals IOC OFF IOP H and IOC IOP 3 (0) clear CPT IOT PAUSE, terminate the IOT instruction, and allow the CP to enter the FETCH Major State.

1.17 RSW INSTRUCTION

The RSW instruction (0516) places the contents of the RIGHT SWITCHES into the AC. IR bits 0 through 5 are decoded to yield INS LOPR H. INS LOPR H is gated with CPT TS5 B (1) H to produce SLA (LOPR · TS5) H, which is ANDed with INS NEQ 16 to enable RIGHT SWITCHES 0 through 11. The AC is loaded at TP5.

1.18 LSW INSTRUCTION

The LSW instruction (0517) places the contents of the LEFT SWITCHES into the AC. SLA (LOPR · TS5) H is gated with INS N EQ 17 H to enable LEFT SWITCHES 0 through 11. The AC is loaded at TP5.

1.19 LINCtape INSTRUCTIONS

LINCtape instructions are two-word format instructions. These instructions include RDE, RDC, RCG, WRI, WRC, WCG, CHK, and MTB. Other LINCtape-oriented instructions include Skips (STD, TWC, IBZ, STB, and LMR) and Operate instructions (AXO, XOA, TAC, and TMA). For a more detailed discussion of LINCtape instructions, refer to Chapter 6.

1.20 TRAP INSTRUCTIONS AND INTERRUPT

When an INTERRUPT occurs and the CP is in the LINC Mode (if the EN TRAP flip-flop is not set), 0040 is loaded into the MA by enabling PMB TMSC 06 H. PMB SET INT ADD L is true, giving PMB TMSC 06 H. When the ENABLE TRAP flip-flop is set, PMB TMSC 06 H and PMA BMSC 05 H are both enabled and loaded into the MA, thereby loading location 0140 into the MA. PMA BMSC 05 H is enabled by CPS TRAP L and PMB SET INT ADD L.

The instructions that can be trapped and their codes are listed below:

Code	Instruction
0501-0515	Operate 01-15
0521-0535	Operate 01-15 (I = 1)
0700-0737	LINCtape operations when TRAP and EN TAPE TRAP flip-flops are set
0740-0777	EXECUTE
0540-0577	Undefined
1700-1737	Undefined

Of these instructions, two are unique: 1700 and 1720. If these instructions are trapped to 140_8, the return jump in location 140_8 is incremented twice instead of once, because these instructions are I-β class instructions.

1.21 ENGINEERING DRAWING DESCRIPTIONS

The name of each block schematic has a unique three-letter code (i.e., CIN RCA). The name of all the signals generated on that particular print are prefixed with this code.

1.21.1 Console Indicators (D-BS-EP12-0-CIN)

The connectors shown on this drawing are single-width modules with the A side at the top of the drawing and the B side at the bottom. The signals shown on this drawing are connected through cables to the console indicator panel.

1.21.2 Central Processor Run (D-BS-EP12-CPR)

The MODE and RUN flip-flops and their associated logic are contained on this drawing. The MODE R/S flip-flop comprises two M617 NOR gates and is set to 8 Mode when power is turned on. The MODE flip-flop may be changed by a manual switch or under computer control.

The RUN flip-flop must always be set to generate computer timing. The RUN flip-flop is direct-set with either of the following key functions (FILL, EXAM, FILL STEP, and STEP EXAM) START 20, 400, and LSW.

1.21.3 Central Processor States (D-BS-EP12-0-CPS)

The logic for the nine Major States is contained in this drawing. Only one state can be entered at TP1. One of the major control signals is Get Next Instruction (GNI). GNI is controlled by inhibit logic and must be true to

enter a FETCH cycle. A FETCH cycle can be inhibited, however, by a Tape Break, Data Break, or an INTERRUPT request. The Tape Break has priority over all BREAK and INTERRUPT States. The Data Break has the second highest level of priority and is controlled by the BREAK SYNC flip-flop. The INTERRUPT has the third highest priority level and is controlled by the INTERRUPT SYNC flip-flop.

1.21.4 Central Processor Time States (D-BS-EP12-0-CPT)

Flip-flops TS1 through TS5 and logic for T5 RECYCLE are contained on this drawing, as well as the INTERNAL PAUSE and IOT PAUSE flip-flops. Two more flip-flops that are necessary for the generation of time states CYCLE DONE and MEM IDLE are shown on this drawing.

The TS1 through TS4 flip-flops are cross-coupled NOR gates. TS5 is direct-set by TP4 and can be controlled by the T5 RECYCLE flip-flop and TP5.

The T5 RECYCLE flip-flop is set only when the RECYCLE SYNC flip-flop is true. The CYCLE DONE and MEM IDLE flip-flop have to be set to start another timing cycle. The CYCLE DONE flip-flop is set by MFTP2 and CPTP SET CYCLE DONE. The MEM IDLE flip-flop is set by I/O PRESET, and MCT MEM DONE.

1.21.5 Central Processor Time Pulses (D-BS-EP12-0-CPTP)

The generation of all time pulses is depicted in this drawing. The timing chain is started off by the occurrence of TP1. To generate TP1, the RUN, CYCLE DONE, and MEM IDLE flip-flops have to be set. TP2 is the result of a STROBE signal from memory when EN MEM is set or is generated approximately 350 ns after TP1 when EN MEM is cleared. The time pulses are generated from TP3 to TP5, independent of the memory or the time states.

1.21.6 Console Switch Inputs (D-BS-EP12-0-CSI)

The logic for all console switch inputs is found on this drawing. Switches that have similar functions are ORed together before they are routed to an M111 Inverter. The Key Start switches are ORed to provide a single output CSI KEY STARTS. A similar procedure is followed for KEY FILL, FILL STEP, EXAM, and STEP EXAM to generate CSI (FILLS + EXAMS).

1.21.7 Console Starts (D-BS-EP12-0-CST)

Manual function states are generated in M700 Module. Manual time pulses are generated whenever any one of 10 keys is depressed on the console control panel. Each switch is routed to a filter and a Schmitt trigger to generate MFTS0. Manual timing proceeds automatically through the delay lines. The input at KP2 must be true to initiate the manual function states, and timing which controls the start of the memory Read/Write cycle.

The DO flip-flop is used to perform one instruction at a time. It also clears the EN MEM flip-flop. The FILL STEP flip-flop is direct-set when the FILL STEP key is depressed; the FILL STEP flip-flop is cleared at TP5D, at which time the STEP EXAM flip-flop is set, because the Fill Step function is followed by a Step Exam function. The AUTO flip-flop allows the STEP EXAMS, FILL STEPS and DO functions to be repeated at a time rate determined by the delay of the manual control knob AUTO RESTART DELAY on the A/D panel.

1.21.8 CARRY INSERT (D-BS-EP12-CYI)

The signal CARRY INSERT is connected to adder bit 11 and is used to increment the major registers. All the logic to generate CYI CARRY INSERT L is shown on this drawing.

The four registers that can be incremented are MA, PC, MB, and AC.

1.21.9 Flow and End Shift (D-BS-EP12-FLE)

The logic for the FLOW flip-flop, High-End Shift and Low-End Shift, is contained on this drawing. The FLOW flip-flop is used to indicate Overflow from AC bit 00 when doing an arithmetic operation in LINC Mode. The signal FLE HIGH END SHIFT IN H is connected to adder bit 0 and is used for ROTATE RIGHT instructions. The signal FLE LOW END SHIFT IN H is connected to adder bit 11 and is used when doing ROTATE LEFT instructions.

1.21.10 LINK Logic (D-BS-EP12-0-FLK)

The logic for the LINK flip-flop is contained on this page. The LINK flip-flop is an extension of the AC. When a CARRY OUT occurs out of AC bit 00 during a 2s complement addition, the LINK is complemented. It can be set or cleared independently of the AC under 8-Mode control, and may be included in shifting and rotating operations performed on the contents of the AC. The LINK adder is comprised of a M111 Inverter at location K09 and the NOR gate M160 at location J13.

1.21.11 I/O and EXT MEM Cables (D-BS-EP12-0-ICB)

The cable connections N14 through N18 shown on this drawing are used to interface external I/O devices to the computer. Cable connections N20 through N24 are used in conjunction with the MM 8 I -A or -B Extended Memory. Cable connection N13 is used to sense external levels.

1.21.12 Instruction Register (D-BS-EP12-0-INR)

The instruction register and its associated logic is contained on this drawing. The IR is direct-cleared when the NAND gate M617 at location H38 is qualified. Flip-flops IR00 through IR2 are direct-set by I/O PRESET to ensure that the first random instruction in the IR is not CHANGE MODE when power is turned on. The IR is connected to the register bus and is normally loaded during FETCH. IR bits 08 through 11 (N bits) are used as a down counter for MUL instructions and LINC ROTATES. Instruction register bit 07 is called the I bit for LINC Mode addressing and LINC instructions.

1.21.13 Instructions (D-BS-EP12-0-INS)

This drawing shows the major decoding of instructions and associated logic. Decoding is accomplished by binary to decimal decoders. By grounding pin U1, only binary-to-octal is decoded. Seven M161 Decoders are shown on this drawing. Only one decoder is used to decode 8-Mode instructions.

1.21.14 I/O Input Part A (D-BS-EP12-IOA)

Inverters for signals from the External I/O Bus are shown on this drawing. The Internal/External I/O Bus is shown on the right side of the drawing and comprises M516 Modules in locations M17 and M18.

1.21.15 I/O Input Part B (D-BS-EP12-0-IOB)

State signals from external peripheral devices are shown here. These signals request a function from the CP or modify the operation of the program. The three signals EXT DA0-2 select the extended memory fields for Data Break devices. The 12 XL signals XL0-XL13 are external skip levels for the LINC SXL instruction.

1.21.16 I/O Control and Timing (D-BS-EP12-IOC)

The I/O timing chain is shown on the bottom of the drawing. IOP control pulses are generated by the logic on this drawing and routed to peripherals. Interrupt control is shown on the top right side of drawing. The flip-flops INTERRUPT ENABLE and INTERRUPT DELAY allow the processor to finish the current instruction before recognizing the interrupt. The logic for the three-cycle Data Break control is shown on the top left corner of the drawing. Word Count Overflow indicates that data transfer is complete. Address Accepted indicates that the computer has acknowledged the Break Request.

1.21.17 I/O Buffers (D-BS-EP12-0-IOO)

All the output signals are buffered before going onto the I/O cables. These signals are used for both data and synchronization pulses for the I/O devices.

1.21.18 Relay Buffer (D-BS-EP12-0-IOR)

The logic to control the relays is shown on this drawing. Relay register bits 00 through 05 are loaded with AC bits 06 through 11 when the ATR instruction is executed. The ATR instruction is decoded by a NAND gate M115, to generate the IOR LOAD RELAYS pulse.

1.21.19 Interprocessor Cables (D-BS-EP12-0-IPC)

The cables shown on this drawing connect the memory and the Central Processor.

1.21.20 MEM EXTN AC Inputs (D-BS-EP12-0-MEA)

The logic on this drawing shows the extended memory I/O Bus. The instruction RDF is decoded by the NAND gate M117 at location L24, and RMF is decoded by NAND gate M617 at location L05.

1.21.21 MEM PAGE EXTN Control (D-BS-EP12-0-MPG)

The control logic for LINC fields is shown on this drawing. Under normal operating conditions the IB and IF flip-flops are set identically. When a LIF instruction is decoded, the IB flip-flops are set to contain the new field requested and the Save Field (SF) flip-flops are loaded with the contents of the IF and DF flip-flops. The fields are not changed, however, until the next JMP Y instruction ($Y \neq 0000$) is executed. When the JMP ($Y \neq 0000$) is executed, the new field is clocked into the IF flip-flops from the IB flip-flops.

1.21.22 MUL Quotient (D-BS-EP12-0-MQR)

The MQ register and associated logic is shown on this drawing. The MQ is loaded only from the AC and is used as an extension of the AC when doing ROTATE instructions and for the KE12 Option. The MQ also contains the least significant bits of the partial product when performing a MUL instruction.

1.21.23 Processor Miscellaneous A and B (D-BS-EPR-0-PMA, PMB)

Eight extra enable gates to the adder are provided on these two drawings for bits 00 through 12. The enable gates accommodate I/O Bus, A/D, Display, Tape, Relay, and any miscellaneous signals for transfer to a main register.

1.21.24 Processor Register Bits 0 through 11 (D-BS-EP12-0-PRA through PRF)

The logic on these drawings shows six M221 Main Register Modules and additional gating. Each module handles two bits of each of the four registers (AC, MB, PC, and MA). There are two inputs to the adder; each is connected to an AND/NOR combination enable gate. Each adder is connected to the next higher order adder by CARRY OUT, and to the lower order adder by CARRY IN. Adder outputs combine with the shifting logic to enable or disable major registers through the register bus. Comparator logic is also provided between AC and MB, which is routed to the Skip logic.

On the *PRA* drawing, the AND/NOR gate M160 at location H13, and the NAND gate M627 location J14 inhibit the adders from indexing beyond bit 2 in LINC Mode. PRA CARRY IN 01 H is connected to the Carry input of bit 01.

1.21.25 Register Control A (D-BS-EP12-0-RCA)

The enable logic for the AC is contained on this drawing. The AC can be enabled in separate halves for half-word instructions. The enable logic for the IR is also contained on this drawing. Depending on the instruction being executed, IR02-07 or IR08-11 is enabled.

1.21.26 Register Control B (D-BS-EP12-RCB)

This drawing contains enable logic for MA, MB, MEM, IF, and DF. The enable levels for the MA, MB, and MEM are divided into sections to facilitate the various instructions. Sometimes only specified bits are required for a transfer between registers.

1.21.27 Register Control C (D-BS-EP12-0-RCC)

The enable logic for the PC, MQ, and the AC COMPLEMENT are shown on this drawing. Other Miscellaneous functions are also enabled on this drawing. The complement of the AC is usually enabled at TS5, except for an 8-Mode Complement AC instruction. The entire PC is normally enabled in 8 Mode. In LINC Mode, PC02 through 11 are enabled because PC 00 and 01 contain the current field. The logic to enable the MQ is found on the right side of the drawing. The enable gating for a BIT CLEAR and BIT SET is also shown.

The signal RCC EN I/O BUS H is shown on this drawing, and the signal routed to the T MSC logic.

1.21.28 Register Control D (D-BS-EP12-0-RCD)

Enable logic is shown for the following signals.

a. RCD EN A D H – this signal is routed to B MSC to provide the inputs for the A/D register.

b. EN AC N-6 L and EN AC N-6R – these signals are used for half-word operations.

c. EN RSW – this signal is routed to the enable gates (dwgs. *PRA* through *PRF*). It enables the 12 right console switches onto the register bus.

d. RCD EN DATA ADD – this signal is routed to the inputs to main registers. It enables the external address lines during a break request.

e. RCD EN TAPE BUS – this signal enables the tape bus and is routed to MBSC.

The Save PC flip-flop and TAPE TRAP flip-flops are also shown on this drawing. Both flip-flops are controlled by Miscellaneous instructions, as explained in Paragraph 1.13.

1.21.29 Processor Register Load Control (D-BS-EP12-0-RCL)

The Load Control gates for major register gates (MB, MA, MQ, AC and PC) are shown on this drawing. As shown in the logic on the left, the MB is always loaded at TP3 except for a SET instruction in the EXECUTE 2 Major State. The MA is loaded when a manual key is depressed that generates MFTP2 H, or by CPTP TP1 L. The MQ is loaded for MUL, CLR, DSC · EXECUTE, EAE, and ROTATE instructions. The control logic to load the AC is shown on the top right of the drawing; the control logic to LOAD PC is shown on the bottom right of the drawing.

Example of the Load Operation:

> To transfer the contents of the MB register to the AC, the MB is enabled on the main register bus RCS EN NO SHIFT and a Load AC pulse is generated.

1.21.30 Register Shift and MQ Inputs (D-BS-EP12-RCS)

The shift logic to control the shifting operations of the register bus is shown on this drawing: ENABLE NO SHIFT, ENABLE SHIFT RIGHT, and ENABLE LEFT SHIFT. Only one of the shift control signals can be true at one time. The logic to shift the contents of the MQ register right or left is also contained on this drawing. The logic to enable AC to MQ during a MUL instruction is shown on this drawing.

1.21.31 SKIP FF and H BIT (D-BS-EP12-0-SKH)

The control logic for the SKIP flip-flop is shown on this drawing. All Skip instructions that are executed set the SKIP flip-flop when the Skip condition is true. The output of the SKIP flip-flop is routed to CARRY INSERT (dwg. *CYI*) to increment the MA in the next FETCH cycle when the SKIP flip-flop is set. The logic to decode all 8-Mode Operate Skips is shown on this drawing. Sense Line inputs for LINC Skip instructions are ANDed with IR bit 07 to produce SKH L FETCH SKIP L. Both the H and SKIP flip-flops are direct-cleared by SKH CLR SKIPS L. The H flip-flop is used for Half-word instructions, and is set whenever MB bit 00 is 1 during the DEFER cycle.

1.21.32 EP12 SKIPS (D-BS-EP12-0-SKL)

The logic contained on this drawing is used for LINC Skip instructions. The signal SKL I SENSE L is qualified by one of 14 possible internal Skip conditions, the signal SKL X SENSE L is produced by one of 12 possible external Skip conditions and, four internal Skip conditions. The signal SKL LINC EX SKIP L is used for SHD, SRO, SAE, and XSK instructions.

1.21.33 Special Level 1 (D-BS-EP12-0-SLA)

This drawing contains logic for special level signals. The INDEX CLASS and 2 WORD FORMAT signals are decoded on the top left side of the drawing. Control signals for arithmetic, rotate, and multiplication operation, are established at the bottom left side of the drawing. At the center of the drawing, instructions that require similar logic are combined to produce one single instruction control signal. At the right side of the drawing, control signals for a specific time state are produced. The 8-Mode Operate 1 and Operate 2 instructions are also decoded.

CHAPTER 2

MEMORY

2.1 INTRODUCTION

The PDP-12 uses a 4096-word, mass storage, 12-bit random access core memory. Memory expansion can be accomplished in increments of 4096 words to a maximum of 32,768 words. However, any increase in size of the memory from the basic configuration necessitates the addition of the Type MC12 Memory Extension Control with the first 4K of extended memory. The first 4K of extended memory is plugged directly into the processor mainframe. The balance of the extended memory (MM8/I options) must be located external to the computer.

2.2 PHYSICAL DESCRIPTION

The elements that constitute the PDP-12 core memory system and their functional relationships are illustrated in Figure 2-1. For simplicity the illustration depicts a 5-bit system, although the PDP-12 computer is a 12-bit system.

As shown in Figure 2-1, memory is addressed through the memory address register (MA). Address selection is achieved through the X and Y selection matrix. One sense register element (MEM), sense amplifier, and inhibit driver is required for each bit.

Memory stack (DEC Part No. 30-05256-1) is a triple-width module (see Figure 2-2). It is plugged into the processor mainframe locations C4, D4, and C6, D6. The cables are connected to two W025 Modules that are plugged into location A5, B5 for inhibit and A6, B6 for sense.

The ferrite core memory consists of 12 planes, each containing 4096_{10} ferrite cores arranged in a 64 x 64-core array. Each core assumes a stable magnetic state corresponding to either a binary 1 or a binary 0. Figure 2-3 shows a simple 4 x 4 core array.

Each core is threaded by four windings: X, Y, sense, and inhibit. There are 64 X and 64 Y selection lines; one X line is threaded in series through each corresponding horizontal row in each array, as is a Y selection line in vertical columns. Each individual selection line (X or Y), therefore, threads 768 (64 x 12) individual cores in the 4096_{10} word stack. One *sense line* is wound through every core of each array. There are, therefore, 12 sense lines. One *inhibit line* also threads through each core array for a total of twelve inhibit windings.

2.3 READ/WRITE

Each X, Y, and inhibit line, when excited, carries a current of insufficient magnitude to cause a change in the magnetic state of a core; this current is designated the *half-select* value. X and Y half-select currents, simultaneously flowing through a common core in the same direction, are cumulative and produce sufficient flux in the core to cause a change of state, provided the core is not already in that state. This current, now designated *full-select* (flowing in the Write direction, through a core initially in the 0 state), causes the core to change to the 1 state. If current flow is in the Read direction (and the core is in the 1 state), a change to 0 state occurs.

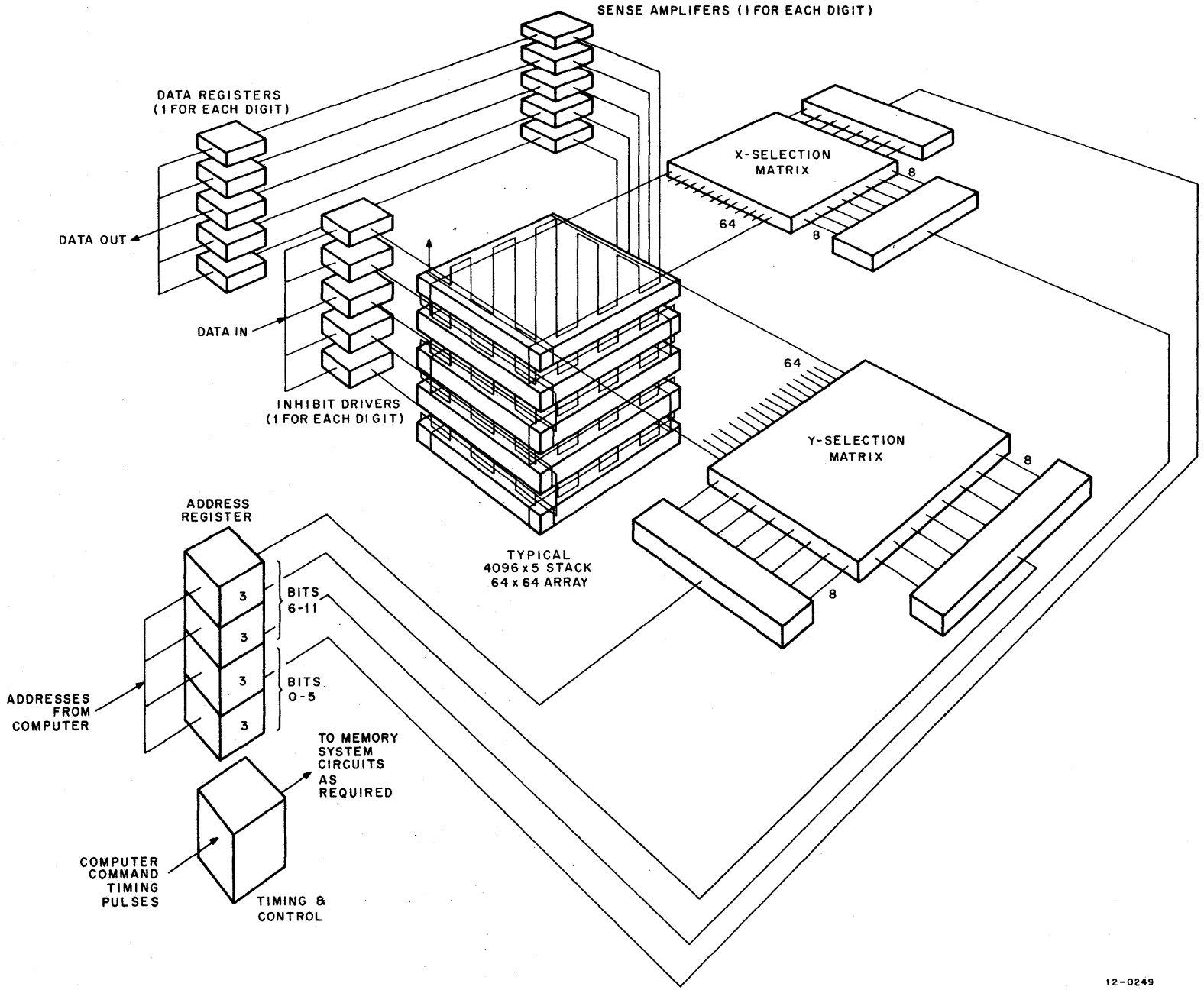


Figure 2-1 Elements of PDP-12 Core Memory

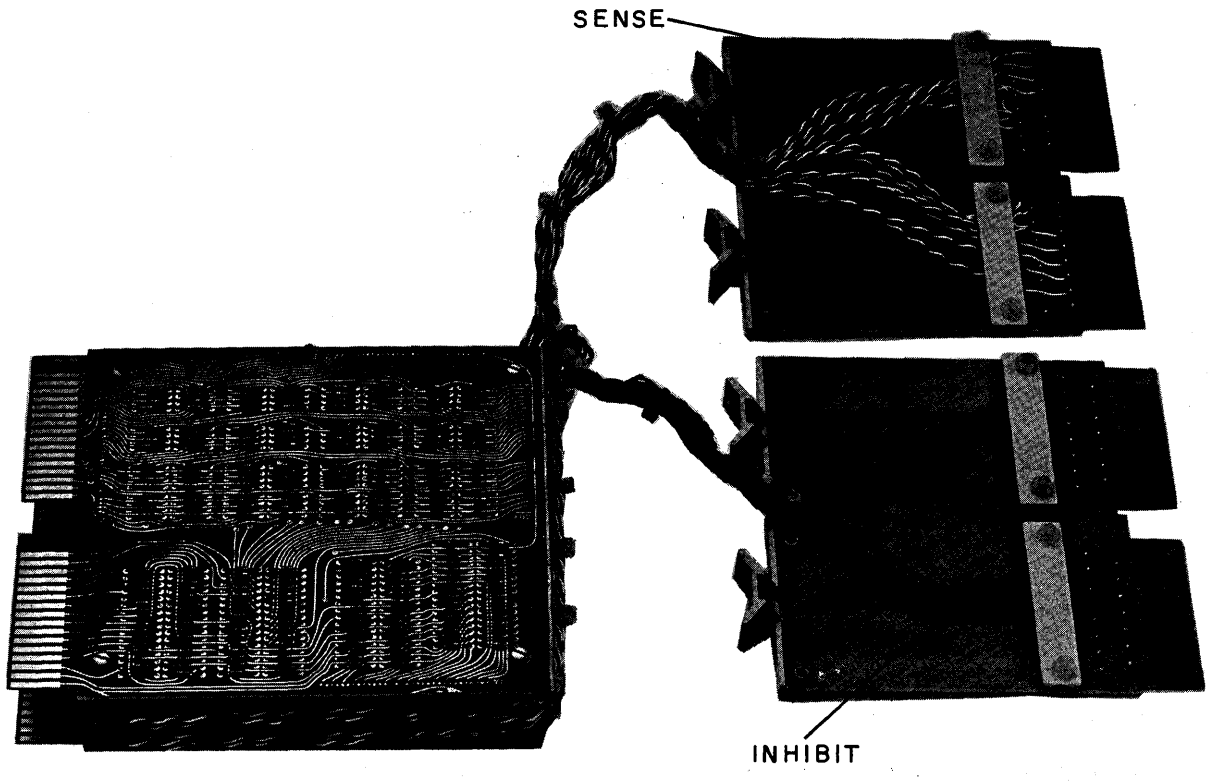


Figure 2-2 PDP-12 Core Memory Stack

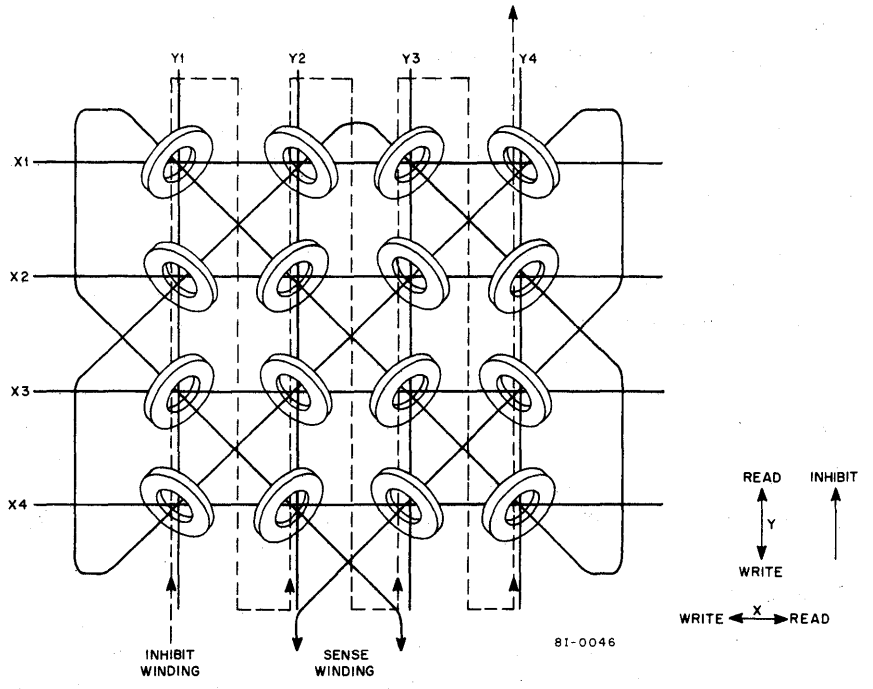


Figure 2-3 Simple Core Memory Plane

To write a 0 into a memory location, half-select current is caused to flow in the inhibit winding associated with that bit. This current flows in the opposite direction to Y Write current. The selected core, therefore, is subjected only to X half-select current, and no change of state occurs; the Y and inhibit half-select currents cancel one another. Effectively, a 0 is written into the location (cores are always assumed to be initially in the 0 state prior to a Write cycle). Full-select current in the Read direction through a core in the 0 state causes no change of state.

When reading, the change of state of a selected core from a 1 to a 0 induces a voltage of approximately 45 mV in the associated sense winding. This voltage is amplified and sets the associated flip-flop in the sense register. The memory bit location is now 0, because this is a *Read-Destroy* memory system. A selected bit location containing a 0 does not induce a voltage of sufficient amplitude to set a sense register flip-flop, because no change of state occurs in the core.

When one core of the 64 x 64 core array is subjected to full-select current, the remaining 63 cores threaded by each line are half-selected. Half-selecting a core causes a small output to be sensed by the sense winding (approximately 5 mV). The polarity of these half-select outputs is dependent upon the initial states of the cores. As a worst case example: If all 126 cores subjected to half-read current are in their 0 state, the voltage sensed is of the same polarity and much greater in amplitude (126 x 5 mV) than the individual output from the selected core. This problem is overcome by winding the sense lines in such a way that the effects of the voltages sensed from the half-selected cores do not add.

The sense amplifiers provide amplification of the analog output of the addressed memory ferrite cores. During the READ portion of the memory cycle, sense amplifiers detect the analog signals induced in the sense windings. These signals are amplified (the 1s only) and strobed into their respective sense register flip-flops, setting only those corresponding bits of the register to 1 (3V level).

A sense amplifier circuit is provided for each of the 12 core planes in memory. If, during a Read operation, the addressed core in a plane makes a 1 to 0 state transition, the flux change induces a current in the sense winding of that plane. This current develops a 40 mV to 50 mV voltage pulse at the input to the sense amplifier. This input is amplified, shaped, and, after threshold detection, used to set a SENSE flip-flop connected to the output of the corresponding sense amplifier when a STROBE signal is received from memory control. Addressed cores that were already in the 0 state, when saturated by the full-select Read flux, induce a limited amount of noise into their sense winding. The voltage level produced by this noise (in the order of 5 mV) is sufficient to activate the sense amplifier associated with that plane. The SENSE flip-flop for that bit, therefore, remains unaffected, indicating a logic 0 in that location.

Because this type of readout destroys the content of the addressed cell (by switching all cores to 0s), the data stored in the sense register are transferred to the MB for restoration to their original location during the WRITE portion of the memory cycle; however, with some software instructions, the output of the sense register is not gated to the MB, and there is no core restoration.

2.3.1 Sense Register (MEM)

All data that is read from core memory through the sense amplifiers are strobed first into this 12-bit register. It accepts data only from core memory (1s only) and transfers data directly through the major register gating network.

2.4 MEMORY ADDRESSING

Selection of one 12-bit word from the 4096_{10} that may be addressed in one field is accomplished through X and Y selection switch matrices. Bits 00 through 05 of the memory address (MA) word are used to select one of 64

X lines that thread through each of the 12 planes. MA Bits 06 through 11 select one of 64 Y lines in a similar manner. Memory addressing is shown in Figure 2-4.

The memory selector switches decode the address specified by the CP memory address (MA), and select the proper source and return lines for both X- and Y-axes.

Even though page addressing is used in 8 Mode, the MA always contains the absolute addresses for the selected memory field. A relationship between absolute address and page addressing in the 8 Mode is shown in Figure 2-4.

The polarity of the magnetic field applied to the addressed cores is determined by the direction of current flow through the core Read/Write windings.

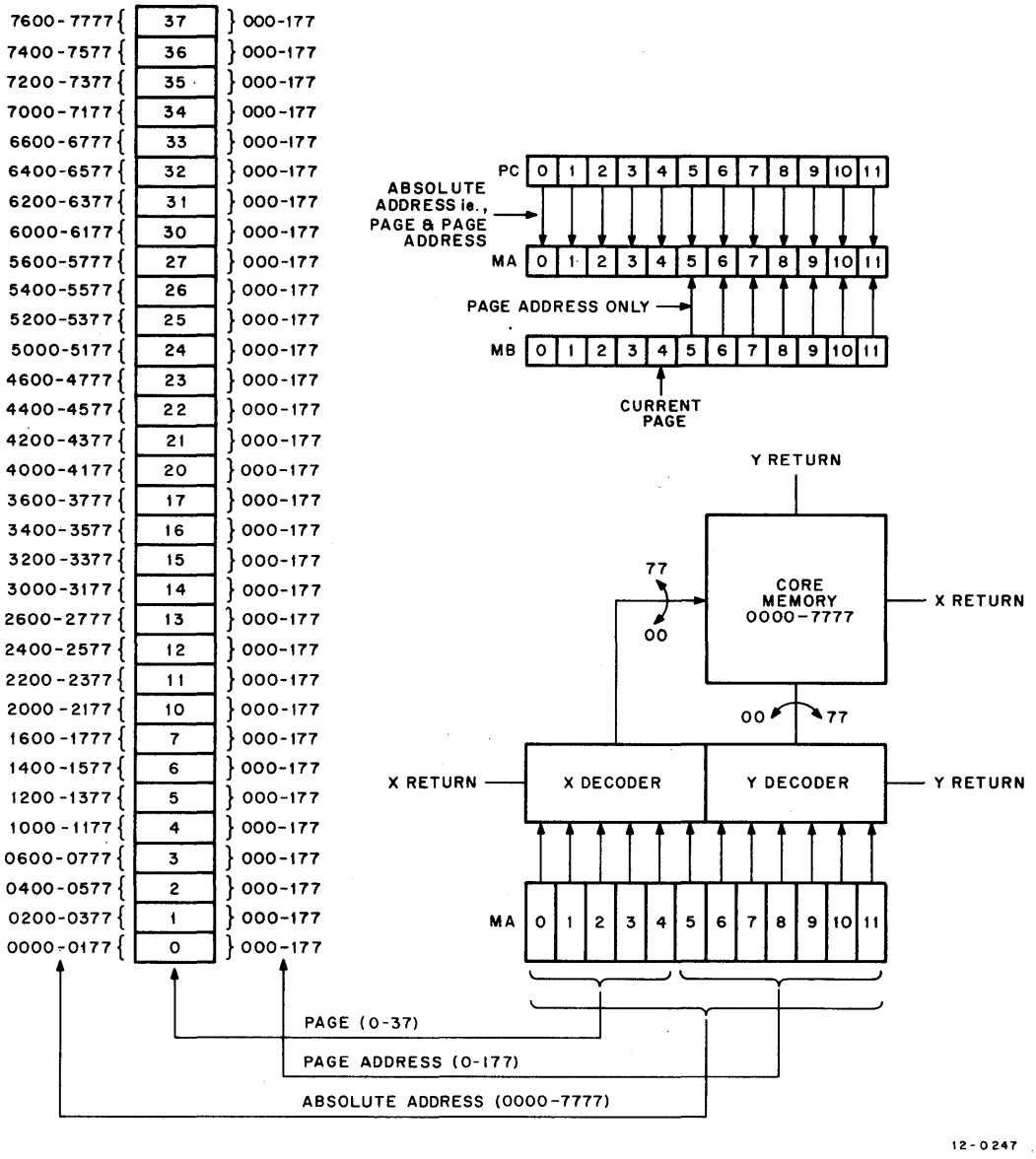


Figure 2-4 Page Addressing in 8 Mode

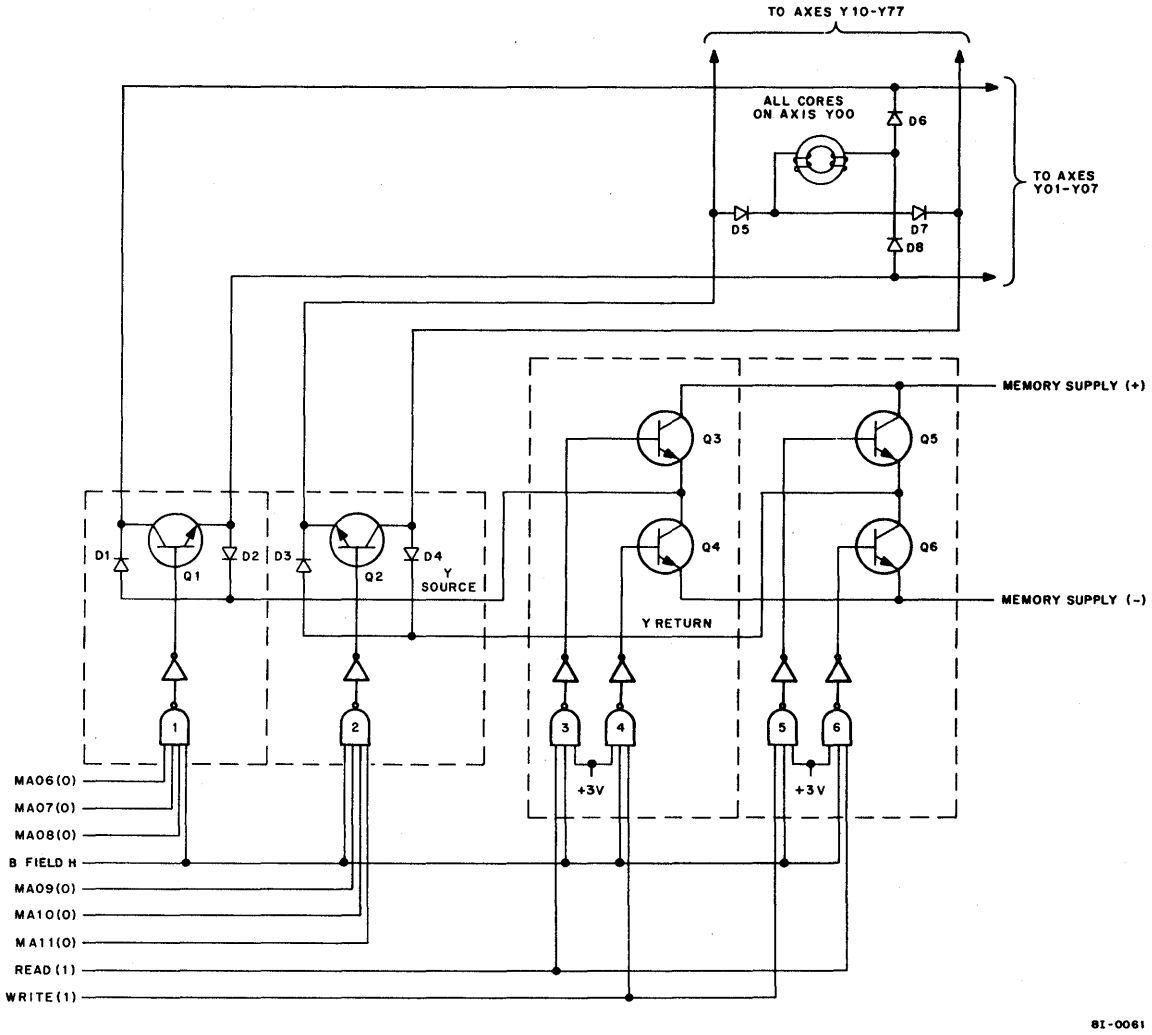


Figure 2-5 Simplified Memory Address Selector and Read/Write Current Control

Figure 2-5 provides a simplified version of the selection of a Y-axis memory cell and the method of determining the Read/Write current direction. It is assumed in the drawing that bits 6 through 11 contain 0, selecting address 00.

NOTE

The component designation numbers used in Figure 2-5 have been arbitrarily assigned in this discussion, and do not relate to actual designations.

With a Read operation in process, the Read (1) line enables 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 MA 06 through 11 = 0, gates 1 and 2 have been enabled, both Q1 and Q2 conduct, and current flows through D1 and Q1 to D8. The Read current then passes through D8 and the winding of cores on the Y-axis of 00, and switches the cores to the 0 state if the X-axis is also selected. There are 12 x 64 cores along this Y-axis, one for each bit and each X-axis. The Read current then passes through D7, Q2, and returns to the Memory Supply (−) through Q6. The current does not pass through the cores on the 01–07 and 10–77 Y-axis, because of back-biased diodes within the matrix and on their address selection switch.

In a Write operation, Q4 and Q5 are turned on by gates 4 and 5, reversing the direction of current flow. The address selection path now becomes D4, Q2, D5; the cores of the Y00 axis, D6, Q1 and D2. This new current direction attempts to set the cores to the 1 state. As previously stated, each core through which Write current passes is set to the 1 state, unless the inhibit driver associated with that core has been activated by the sensing of logic 0 in that particular bit position of the Central Processor MB.

Figure 2-6 illustrates the timing relationship between the CP and memory Read/Write FETCH cycle. The cycle time shown is 1.8 μs, which is due to the characteristics of the core memory. The STROBE pulse is adjusted to match the core output of the memory. The relationship of the times given are approximate, with the exception of MEM START/TP1 and STROBE/TP2. TP1 generates MEM START and STROBE generates TP2.

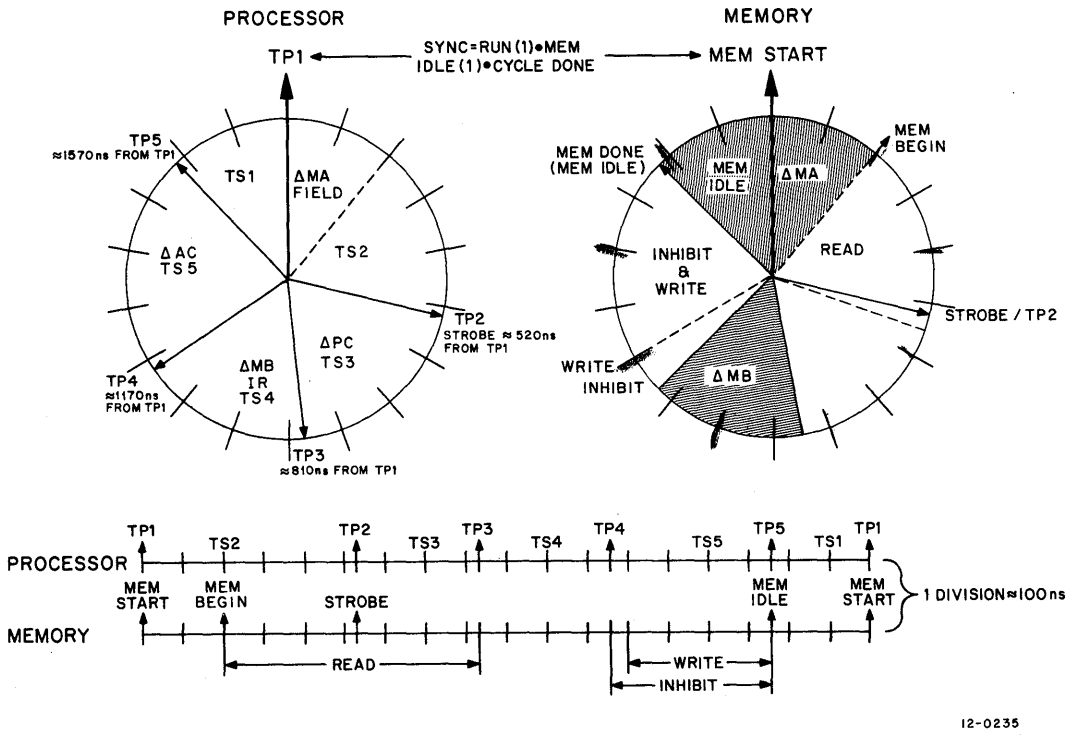


Figure 2-6 CP and Memory Timing Relationships

The MA and FIELD are loaded at TP1 and MEM START is generated. Approximately 200 ns later MEM BEGIN is generated, which sets the READ flip-flop and the memory Read cycle is initiated. STROBE is generated approximately 300 ns later, and the contents of memory are loaded into the MEM register. TP2 is generated by STROBE and the contents of the PC are incremented by one. TP3 is generated approximately 320 ns after TP2, and the contents of the MEM register is transferred to the MB and IR. The READ flip-flop is cleared and, approximately 200 ns later, the INHIBIT flip-flop is set. Approximately 50 ns later, the WRITE flip-flop is set and the memory Write cycle is initiated. The contents of the IR are now decoded to define the instruction. The contents of the MB are written back into memory, MEM FINISH clears the WRITE flip-flop and generates MEM DONE. The MEM IDLE flip-flop is set by MEM DONE, thus indicating the end of the memory cycle.

2.5 MC12 MEMORY EXTENSION CONTROL

Additional core memory can be added to the standard PDP-12 in 4096-word increments. The addition of seven 4K fields, plus the standard memory, yields the maximum storage capacity of 32,768 words. Figure 2-7 illustrates the extended memory organization. As shown in this block diagram, the PDP-12 mainframe contains the standard memory (field 0). The MC12 Option provides the first memory extension and the memory selection control for all eight memory extensions. Local memory timing for both field 0 and field 1 is provided by the basic memory. This control and associated Read/Write and addressing techniques do not differ from those of the basic PDP-12.

2.5.1 Instruction Field Register (IF)

This three-bit register determines the memory field to be used for storage and retrieval of program instructions. The IF register may be loaded from either the IF switch register (IFSR) or the Instruction Buffer (IB) register. All program-executed transfers to the IF enter from the IB (D-BS-MC12-0-MXR). The contents of the IFSR, however, are loaded directly into the IF, and the IB, under manual control of the KEY START LSW. When a JMP or JMS instruction is executed, the contents of IB are transferred to the IF. When an Interrupt occurs, the contents of the IF are transferred to the Save Field register (SF). At the end of the Interrupt subroutine, the contents of the SF are restored to the IF through the IB by execution of the RMF (Restore Memory Field) instruction. When the CIF (Change Instruction Field) instruction is executed, the contents of the MB06 through MB08 are transferred to the IB, and then the contents of IB are transferred to the IF at the execution of a JMP or JMS instruction. During the time between the CIF instruction and the JMP or JMS, Program Interrupts are inhibited.

2.5.2 Data Field Register (DF)

This three-bit register determines the field to be used for data storage and retrieval. The DF can be loaded from either MB06 through MB08, or the Save Field register.

The CDF (Change Data Field) instruction can be used to alter the content of the DF to permit selection of a different field of memory. Bits 06, 07, and 08 of the CDF instruction contain the desired field address. During a Program Interrupt operation, the contents of the DF are automatically stored in the Save Field register and restored to the DF upon completion of the Interrupt subroutine by the RMF instruction.

2.5.3 Instruction Buffer Register (IB)

This three-bit register provides input buffering for data transfers made into the IF under program control. Manual transfers from IFSR are made directly into the IF and, simultaneously, into the IB. The IB is loaded into IF at the execution time of every JMP or JMS instruction. Transfers into IB are made under program control during the execution of CIF (Change Instruction Field) and RMF (Restore Memory Field) instructions.

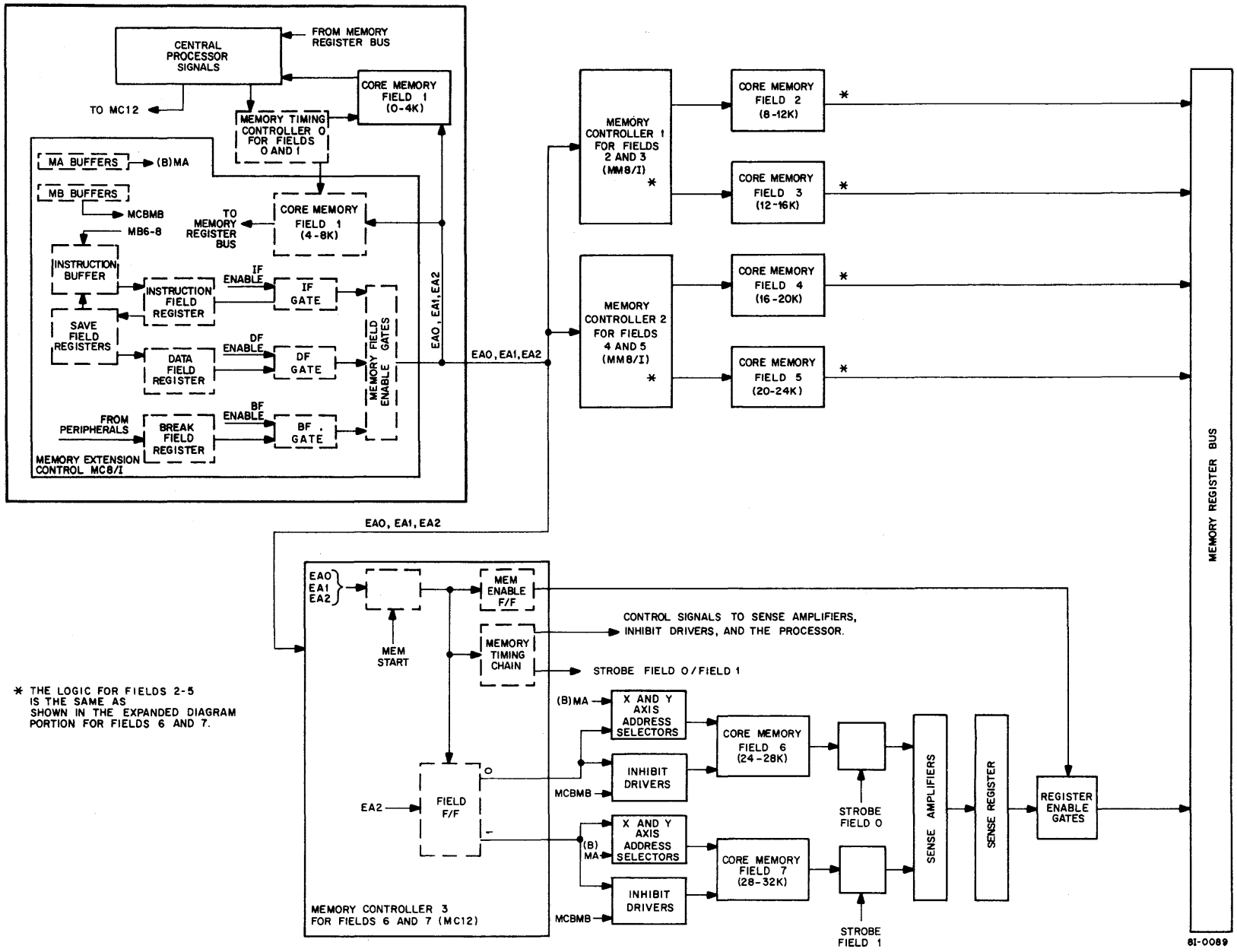


Figure 2-7 Extended Memory Block Diagram

2.5.4 Save Field Register (SF)

This six-bit register provides temporary storage during a Program Interrupt for the contents of both IF and DF. When the Interrupt state is entered, the contents of the IF and DF are loaded into the SF register. At the conclusion of the Interrupt subroutine, an RMF instruction loads the contents of SF0-2 into IB for subsequent transfer into IF and the contents of SF3-5 into DF. The last instruction in the subroutine (JMP 10) completes the transfer from IB to IF.

2.5.5 Break Field Register (BF)

The three-bit register determines the field to be used for storage and retrieval for data transfers from I/O devices using the Data Break facility. The BF is loaded from the EXT DATA ADD 0, 1, and 2 lines by the load BF pulse, which is generated at the end of every cycle. Each device puts its own field address on the EXT DATA ADD 0, 1, and 2 when it requests a BREAK.

2.5.6 Field Selection

Figure 2-7 illustrates the memory extension control gating that allows access to all memory fields. The gate output EA0, EA1 and EA2, taken collectively, form a code that selects one of the memory fields. The EA0 and EA1 levels enable one of the four MM8I Memory Controllers (each of which controls two memories). The EA0 and EA1 configurations are: 00 (for fields 0 and 1), 01 (for fields 2 and 3), 10 (for fields 4 and 5) and 11 (for fields 6 and 7). The third enable level (EA2) specifies which of the two memory fields within the group indicated by EA0 and EA1 is selected. Table 2-1 clarifies select coding, and the distinction between the enable levels.

Table 2-1
Field Select Codes

EA0	EA1	Memory Controller Selected (EA0 and EA1)	EA2	Field Selected
0	0	0	0	0
0	0		1	1
0	1	1	0	2
0	1		1	3
1	0	2	0	4
1	0		1	5
1	1	3	0	6
1	1		1	7

The memory field enable gates may be activated from one of the following three sources: the IF register, the DF register, or the BF register. The IF register is gated to the EA0, EA1, and EA2 levels at all times except during a BREAK cycle or an indirectly referenced data handling instruction.

When START, EXAM, or KEY FILL is actuated, the IF register is gated to EA0, EA1, and EA2 to generate the memory field select code. The DF register is gated to the extended addressing bits when any indirectly-addressed, memory-referenced instruction other than JMP or JMS is executed. The BF decoder allows the BF to generate the memory field select code during a BREAK cycle.

2.5.7 Interrupt Inhibit

The interrupt inhibit logic disables the data input to the INT SYNC flip-flop in the processor from the time a CIF instruction is decoded, until a JMP or JMS command finishes the CIF command. This prevents honoring an Interrupt before the field is changed. Interrupt synchronization is restored after the IF is loaded from the IB, allowing further Program Interrupts to occur.

2.6 MEMORY BLOCK SCHEMATICS

2.6.1 Interprocessor Cables (D-BS-EM12-0-IPCM)

All cable connections shown on D-BS-EM12-0-IPCM are routed to the cable connections shown on D-BS-EP12-0-IPC. These cables connect the memory to the CP.

2.6.2 MCS Sense Amplifiers and Inhibit Drivers (D-BS-EM12-MCS)

Twelve inhibit drivers are associated with each memory stack. Inhibit current waveforms should be inspected and compared with the waveforms in Figure 2-8. Inhibit current amplitude is approximately 290 mA. Each inhibit driver is enabled by MC7 B INHIBIT H and MCT B FIELD 0 H, and the corresponding MC bit on a zero.

Twelve sense amplifiers (G020 or G021) are associated with each memory stack; each amplifier transforms the analog pulse output of the ferrite core to a digital logic level. The G020 Module is used in computers that have a basic 4K memory; when an additional 4K memory is added, this module is replaced with a G021, which provides an additional amplifier input. The sensor 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 amplifier allow observation of the waveforms for comparison with those shown in Figure 2-9.

2.6.3 MCT Memory Control (D-BS-EM12-0-MCT)

The logic shown on this drawing consists of the memory voltage regulators (G805, G826), two voltage drivers (G228), two strobe delays (M310), memory control flip-flops, and associated gating and buffers.

The voltage regulators monitor the various voltages used in the system. When a voltage does not meet the correct operating potential, the MCT POWER OK level goes high and generates the MCT PWR STOP L signal, which halts the computer. The G826 Regulator also generates MCT POWER CLEAR when the system power is turned on.

The two G228 Drivers contain the switches for the memory Read/Write currents. The two M360 Strobe Delays generate the MCT STROBE pulses for memory fields 0 and 1. The MCT FIELD flip-flop selects the strobe for the appropriate memory field. The three memory timing delays (310) generate the necessary timing pulses to control the READ, WRITE, and INHIBIT flip-flops. MCT MEM DONE is generated at the end of the timing sequence.

The control flip-flops have the following functions:

MCT MEM ENABLE – The output of the flip-flop is buffered to produce MCT B MEM ENABLE H. This level enables the SENSE flip-flops onto the MEM lines for transfer to the CP registers. The flip-flop is set by CPTP START MEMORY to select the first 8K of memory fields 0 and 1.

MCT Field – When set to the 0 state, the first 4K (field 0) is selected. When set to the 1 state, the second 4K (field) is selected. MXFGA2 H controls the data input and the flip-flop is clocked with CCPT START MEMORY H, which is gated with -EA0 and EA1.

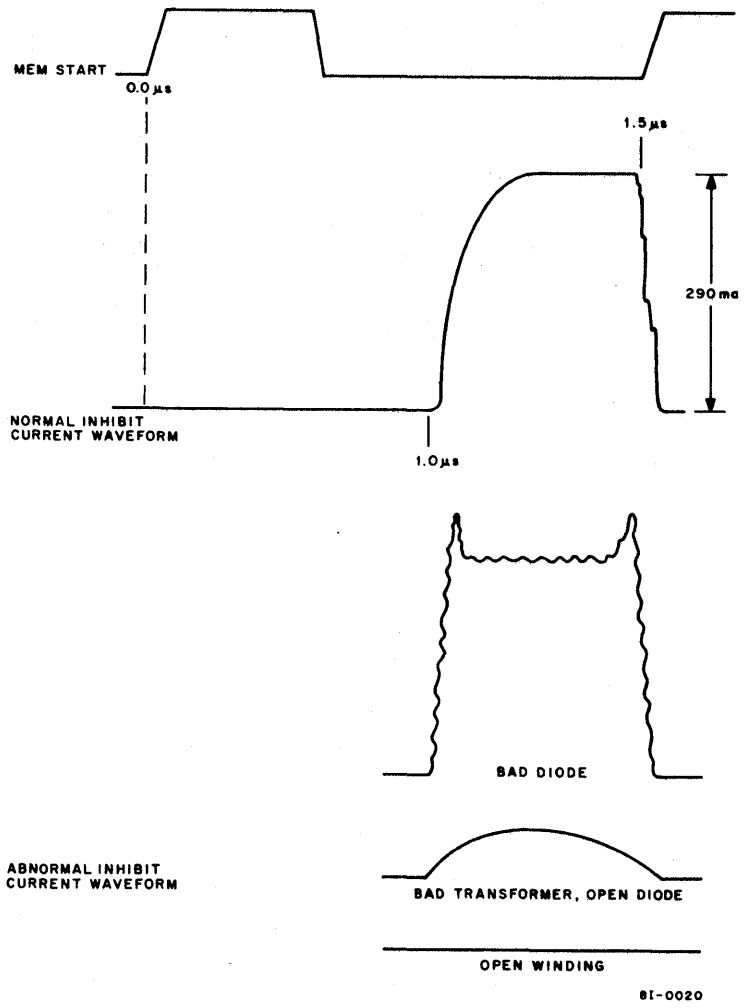


Figure 2-8 Representative Inhibit Current Waveforms

MCT READ – Set when the READ portion of the memory cycle is being performed. It enables the appropriate gates to control the Read current.

MCT INHIBIT – Set when the WRITE portion of the memory cycle is being performed. It enables the appropriate gates to control the Write current.

2.6.4 MCX *X*-Axis Selection (D-BS-EM12-0-MCX)

The *X*-Axis selection of the memory stack is achieved by decoding the MA00 through MA05 bits. MA00 through MA02 are decoded into a binary count of 0 through 7, and MA03 through MA05 are also decoded into a binary count and routed to the X selectors. These selectors are connected to a diode selection matrix and select one core in the 8 x 8 matrix of 64 cores. 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 selection boards as X $0\text{-}77_8$. Suspected opens and shorts in the windings, detected during dynamic tests, should be verified by

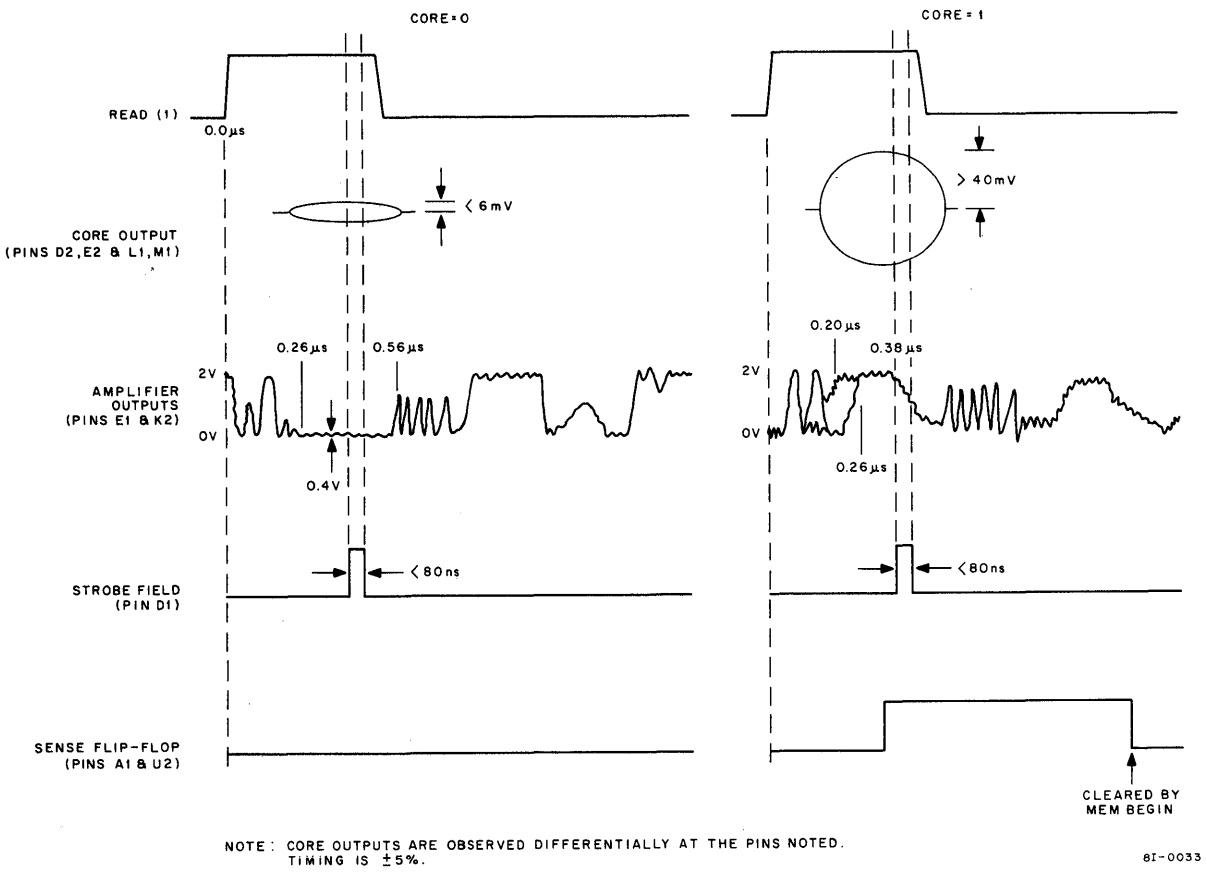


Figure 2-9 Representative Sense Amplifier Waveforms

measurements across the points with an ohmmeter. The resistance of the Read/Write winding is 3.5Ω ±10 percent. The forward and reverse resistance of the diodes in the matrix should be checked when address selection failures are attributed to stack failures.

2.6.5 MCY *Y*-Axis Selection (D-BS-EM12-0-MCY)

The *Y*-Axis selection of the memory stack is achieved by decoding the MA06 through MA11 bits. MA06 through MA08 are decoded into a binary count of 0 through 17, and MA07 through MA11 are also decoded into a binary count 0 through 7 and routed to the Y selectors. These selectors are connected to the diode selection matrix and select one of 64 planes. The windings are identified on the diode selection boards as Y $0\text{-}77_8$.

2.6.6 MEM EXTN Buffer (D-BS-MC12-0-MXB)

The signals required to control the Extended Memory Option (MM8I) are buffered for extra driving capabilities. Two M617 Modules in locations M22 and M23 buffer the MA bits. The two M617 Modules in locations M24 and M25 buffer the MB bits. A Non-Existent Memory Detect Module is also included to generate a strobe pulse and allow the computer timing chain to continue if a non-existent memory is addressed.

2.6.7 MEM EXTN Field (D-BS-MC12-0-MXF)

The gate outputs EA0, EA1, and EA2, select up to 8-4K memory fields. The EA0 and EA1 levels enable one of the four memory controllers (each of which controls two memories). The EA0 and EA1 configurations are: 00 (for fields 0 and 1); 01 (for fields 2 and 3); and 10 (for fields 4 and 5). The third enable level (EA2) specifies which of the two memory fields within the group indicated by EA0 and EA1 is selected. Register enable flip-flops are also provided for the Instruction Field, Data Field, and Break Field.

2.6.8 MXI Inhibit Drivers (D-BS-MC12-0-MXI)

The function of the inhibit drivers is identical to those described in Paragraph 2.6.2. These drivers control the second 4K memory field.

2.6.9 MEM EXTN Register (D-BS-MC12-0-MXR)

The memory extension registers increase the addressing capabilities up to 32K of core memory. There are three, 3-bit registers: DF (Data Field), IF (Instruction Field) and BF (Break Field). The 3-bit IB (Interrupt Buffer) register acts as an intermediate holding buffer for the IF, and is loaded with the LIF, CIF, and RMF instructions. The contents of the IB are transferred to the IF with the LINC, JMP and 8, JMP instructions. The DF register is loaded with the LDF, CDF, and the RMF instructions. The BF register is loaded with the three extended address bits EXT0 through 2, which are controlled by the peripheral devices. The register is clocked with the MXF LOAD BF pulse, which is generated at the end of every CP timing cycle.

The 6-bit SF (Save Field) register retains the contents of the IF and DF registers in the event of a program interrupt. The SF register outputs are gated onto the data inputs of the IF and DF register flip-flops, which are restored with the RMF instructions.

2.6.10 MXX *X*-Axis Selection (D-BS-MC12-0-MXX)

X-Axis selection is identical to the functions described in Paragraph 2.6.4. These selectors are utilized for the second 4K memory field.

2.6.11 MXY *Y*-Axis Selection (D-BS-MC12-0-MXY)

Y-Axis selection is identical to the function described in Paragraph 2.6.5. These selectors are also utilized for the second 4K memory field.

CHAPTER 3

I/O BUS

3.1 INTRODUCTION

The Input/Output (I/O) Bus section of the PDP-12 System contains the bus drivers, bus receivers, and control logic required for communication between the system and peripheral equipment containing digital interface logic compatible with the I/O Bus.

The I/O Bus may be operated in two major modes: for program-controlled or device-controlled transfers. Program-controlled transfers require that the program contain subroutines (or at least a JUMP to an I/O routine) that directly control data transfer. Data Break transfers are controlled by the peripheral device on a cycle-stealing basis. The transfers are made between instructions of the program being performed.

Chapter 3 is divided into three sections:

a. Block Diagram Descriptions
b. Flow Diagram Descriptions
c. Block Schematic Descriptions

3.2 BLOCK DIAGRAM DESCRIPTIONS

The following discussions describe the relationships of the major functional blocks of the I/O Bus. Certain functional elements are used for both program-controlled and Data Break transfers. Both types of peripherals may be connected to the I/O Bus at the same time.

3.2.1 Program-Controlled Transfers

There are two classes of program-controlled transfers as follows:

a. One that uses a conditional skip instruction to test the state of the selected peripheral. If the flag is set, the program skips to a data transfer subroutine, then continues. However, in many cases, it is necessary to service the peripheral within a certain period of time, or the data from the peripheral is lost. In these cases, the conditional skip instruction is followed immediately by a JMP .-1 instruction, and the program is halted in a time-wasting loop until the selected peripheral sets its Transmit or Receive Flag.

b. A more efficient transfer uses a Program Interrupt feature. The program currently being processed continues normally until any peripheral on the bus sets a flag; if the program has previously enabled the Interrupt facility, an Interrupt-conditioned JMP is then performed to a subroutine that interrogates each peripheral to determine which flag has been set. When this has been determined, the appropriate data transfer subroutine is performed, and the program resumes with the instruction following the last program instruction executed before the Interrupt JMP.

Direct Program Transfer – Figure 3-1 is a block diagram of the PDP-12 logic used for direct program I/O Bus data transfers. The Interrupt facility is also shown in this drawing, but is used only for Interrupt program transfers.

The most significant octal digit of an Input/Output Transfer (IOT) instruction is 6_8. When this digit is decoded (dwg. *INS*), the IOP generator is enabled (dwg. *IOC*). The least significant octal digit of the IOT instruction determines which of the three IOP pulses are gated out to the peripheral via the IOP buffers. These pulses time data transfer events within the peripheral.

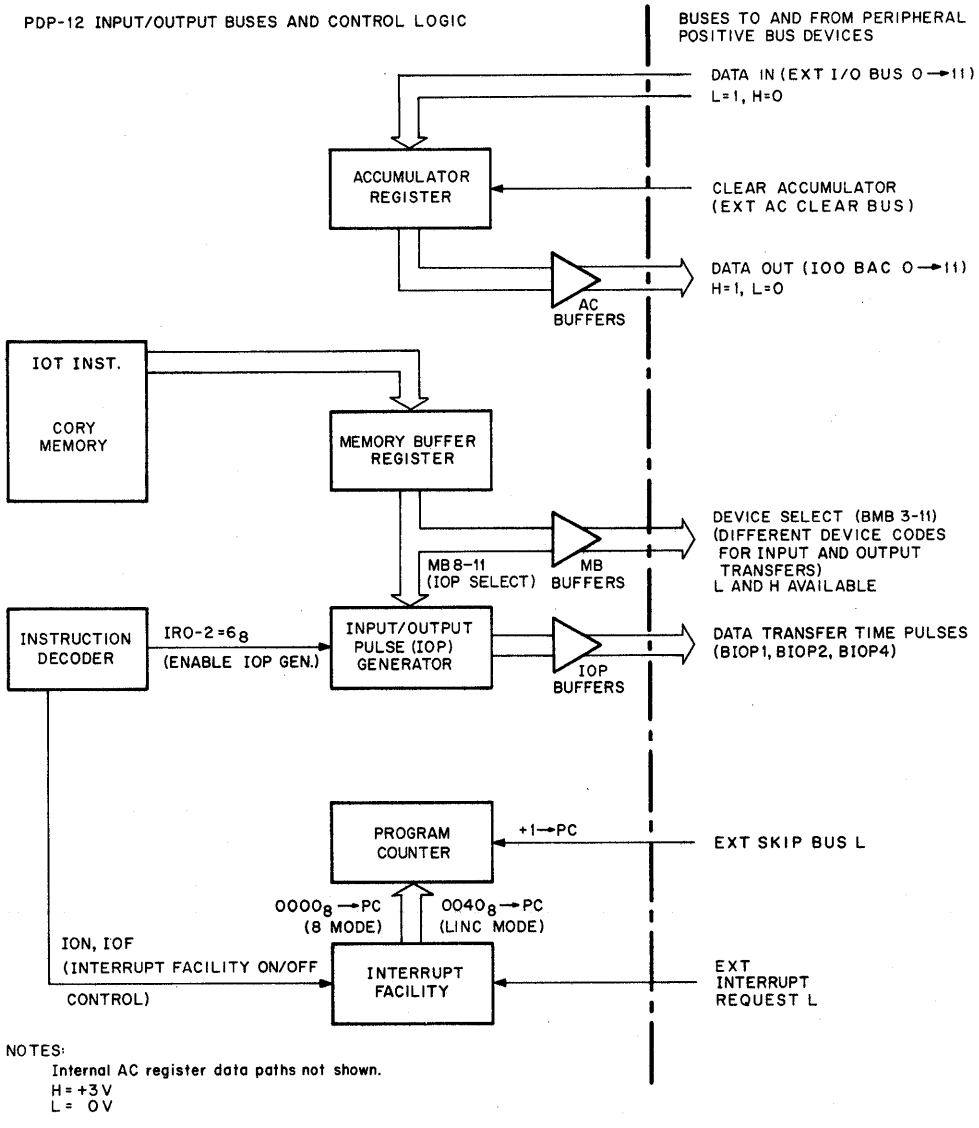


Figure 3-1 Program-Controlled I/O Bus Block Diagram

The peripheral device is selected by the middle octal digits NN ($6NNX_8$) of the IOT instruction. Each peripheral contains a device select decoder which gates the buffered IOPs (dwg. *IOC*) from the bus into that peripheral when it is addressed.

Data transfer usually takes place between the AC and the peripheral data register during the last IOP (IOP 4). IOP 1, the first IOP, is usually enabled by IOT Skip instructions to gate the status of the selected peripheral flag onto the skip bus. The second IOP pulse (IOP 2) is normally used to clear the flag.

NOTE

In some peripherals, the least significant three bits of the IOT instruction are used within the peripheral in combination with the BIOP pulses to produce "sub-device" control pulses.

Program Interrupt Transfer – Figure 3-1 illustrates the Program Interrupt facility and its relationship to the program counter. The Interrupt facility is turned on and off by program instructions: ION enables the facility, IOF disables the facility.

When the Interrupt facility is enabled, any peripheral that sets its flag activates the Interrupt Bus. If the current program is being executed in 8 Mode, the Interrupt facility loads the Program Counter (PC) with absolute address 0001_8. If the current program is being executed in LINC Mode, the PC is loaded with absolute address 0041_8. The previous contents of the PC are stored in 0000_8 and 0040_8, respectively.

A subroutine starting at address 0001_8 or 0041_8 now proceeds to check each of the peripherals on the bus until the device with the set flag is encountered. This device flag causes a program skip so that the next instruction executed is a JMP to the appropriate service routine for that peripheral. If the addressed peripheral does not have its flag set, the skip to that service routine is not performed, and the subroutine continues to interrogate peripherals.

With the exception of the Interrupt Bus and the Interrupt facility, the hardware involved with Program Interrupts is the same as that used in direct program transfers.

3.2.2 Data Break Transfers

There are two classes of peripheral-controlled Data Break transfers: single-cycle or three-cycle. Although simpler and faster, the single-cycle Data Break requires more digital hardware within the peripheral to monitor word count and control the current address of the data being transferred to or from the PDP-12.

The three-cycle Data Break, though slightly slower than a single-cycle Data Break, greatly reduces the digital hardware required in the peripheral. The three-cycle Data Break uses two core memory registers to monitor WORD COUNT and CURRENT ADDRESS, and therefore requires three core memory cycles instead of one for each transfer.

NOTE

Only one Data Break peripheral can be connected directly to the PDP-12. A Data Multiplexer must be used if more than one of these peripherals is to be used in the system. A DM04 Multiplexer must be used if only positive bus peripherals are involved. If negative bus peripherals are used, a DM01 Multiplexer and a DW08A I/O and Data Break Converter must be located between the peripherals and the PDP-12 Data Break facility. Both negative and positive bus peripherals may be used in the same system if a DM04, DW08A, and DM01 (if more than one negative bus peripheral) are used.

Single-Cycle Data Break – A block diagram of both single- and three-cycle Data Breaks is presented in Figure 3-2. Single-cycle data and control paths are shown as solid lines.

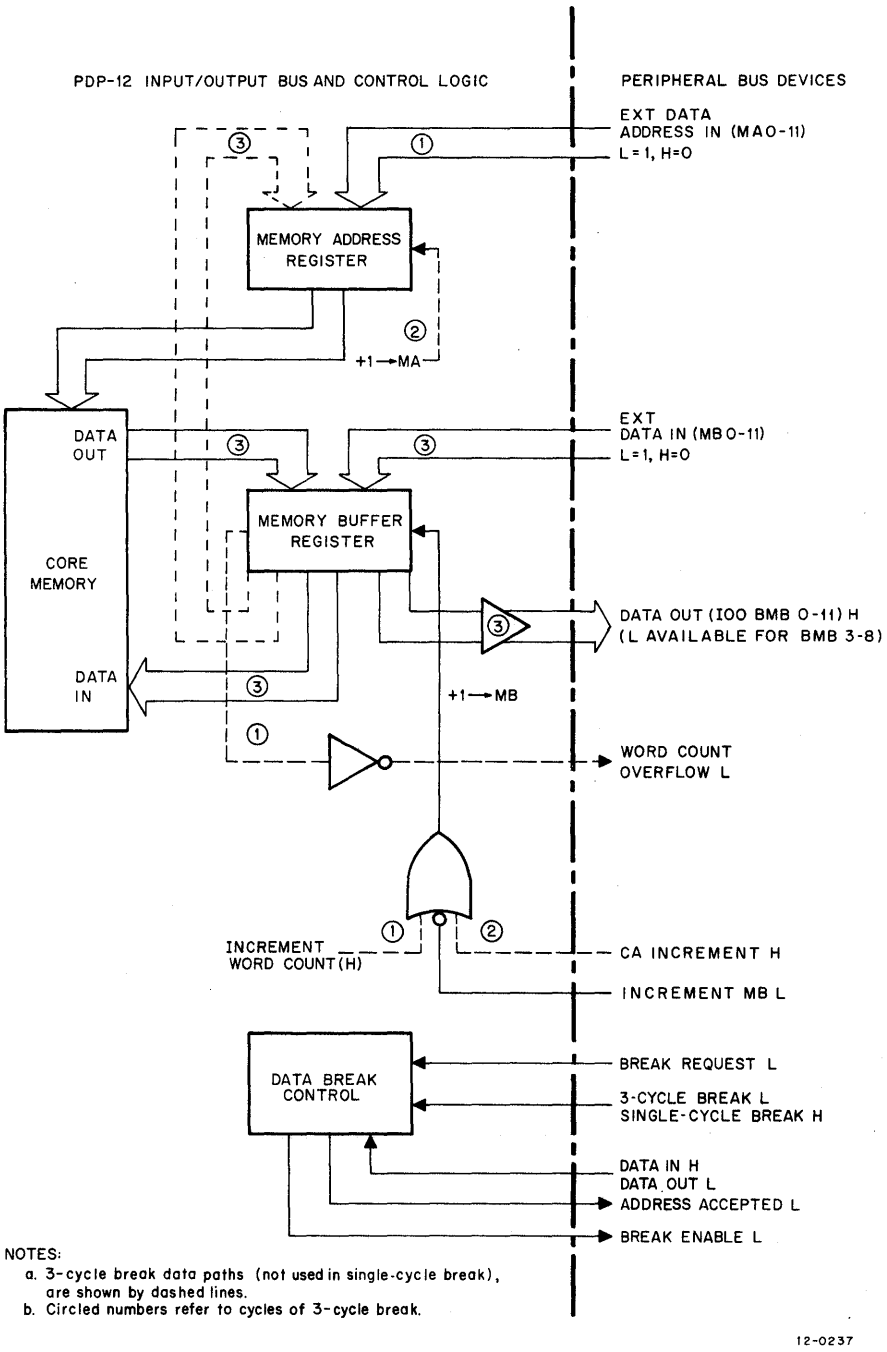


Figure 3-2 Data Break I/O Bus Block Diagram

The device-requesting service sets its Break Request flip-flop. The Break Request is honored at the completion of the instruction in progress when the Break Request occurred. At this time, a Break Enable signal from Data Break Control to the peripheral is initiated and remains active until the end of the BREAK cycle. The peripheral supplies the core memory address to the MA register. Data is transferred via the memory buffer register, the direction of transfer being controlled by the peripheral via the Data In/Data Out Bus and the Data Break Control.

The External Data Address output from the peripheral is used as the input to the MA register. Thus, there is a discrete core memory address for each value read by the peripheral.

A special use of the Data Break facility is Memory Increment. In this mode of use, when addressed, the contents of a peripheral-specified core location is incremented by one. The peripheral must generate the signal EXT INCREMENT MB L, but no data output to the MB is required.

Three-Cycle Data Break – Data and control paths exclusively a part of the three-cycle Data Break are shown in dashed lines on Figure 3-2. The single-cycle Data Break forms the third cycle of the three-cycle Data Break, with the exception that the data address is taken from a core memory Current Address register instead of directly from the peripheral.

The three-cycle Data Break is enabled by the signal EXT 3 CYCLE L (dwg. *IOB*). When the request is honored by the processor at the end of the current instruction, the address of the Word Count register is accepted from the peripheral via the MA register.

The contents of the Word Count register are incremented during the first cycle of the three-cycle Data Break. This register specifies the length of the data table for a given peripheral, and is initially loaded by the program with the complement of the WORD COUNT. As the data transfer progresses, this count is incremented until overflow occurs in the Memory Buffer register, when the final data word is transferred. The Word Count Overflow is provided as a bus signal IOC WC OVERFLOW (0) H to the peripheral, which then can be used as an INTERRUPT condition to inform the program that the block of data has been completely transferred to or from the peripheral.

The second cycle of the three-cycle Data Break is used to obtain the CURRENT ADDRESS within the data table in core memory. The Current Address register is the next sequential address following the Word Count register address, the Memory Address register being incremented automatically at the end of the first cycle. The CURRENT ADDRESS is incremented in the memory buffer, and the contents of the Memory Buffer register are used during the final cycle of the Data Break to address in the data table of core memory. The signal IOB CA INCREMENT H (dwg. *IOB*), allows peripheral control over incrementing the Current Address register.

During the final cycle of the three-cycle Data Break, the data word is transferred between the peripheral and core memory via the Memory Buffer register. This third cycle is identical to the single-cycle Data Break, except that the core memory data address is specified by the contents of the Memory Buffer register instead of by a hardware flip-flop register within the peripheral.

3.3 FLOW DIAGRAM DESCRIPTIONS

The following discussions describe the sequence of events shown on flow diagrams for Interrupt and Data Break Transfers in Volume III of this manual.

3.3.1 Direct Program Transfer (IOT)

The flow diagram for a program-controlled Input/Output Transfer (IOT) is shown as part of Volume III drawing D-FD-PDP-12-0-21 at coordinates A-C, 2.

NOTE

In this and subsequent discussions, alphanumeric coordinates refer to the border around all engineering drawings, and are meant as an aid to locating the portion of the drawing under discussion.

At T4 (time state 4) of any 8 Mode FETCH cycle, the IOT PAUSE flip-flop (dwg. *CPT*) is set if an IOT op code ($6XXX_8$) is decoded. With the PAUSE flip-flop set, T5 continues until all three IOP pulses have occurred (whether or not all three are gated to the peripheral by the least significant digit of the instruction). The termination of IOC IOP 4 H, IOC OFF IOP H resets the IOT PAUSE flip-flop to terminate T5, and the program continues with the memory start pulse of the FETCH cycle of the following instruction.

3.3.2 Program Interrupt Transfer (INTERRUPT)

The flow diagram for a jump to an Interrupt routine is shown as part of Volume III drawing D-FD-PDP-12-0-20 (B-D, 1-4).

Assuming that the Interrupt flag (dwg. *CPS*) has been raised by a peripheral during the previous instruction EXECUTE cycle (prior to T5), the data field and instruction field are transferred to the Save Field register by the start memory pulse (T1) of the first cycle of the INTERRUPT Major State.

If the current instruction is being performed in 8 Mode, the MA register is loaded with 0000_8, where the address in the PC just prior to the Interrupt is stored. (An indirect Jump 0 may be done at the end of the device-servicing routine, to return to original program.) Location 0040_8 is used for the same purpose in LINC Mode. Locations 0001_8 and 0041_8 are used to begin the service routine, as the MA + 1→ PC during T3 of the Interrupt cycle. If the TRAP Flag is set (a software Interrupt feature), the two corresponding locations are 0140_8 and 0141_8. The Interrupt Enable flip-flop is cleared at T1 of an Interrupt cycle.

The instruction register is cleared during T2. At T3 the contents of the PC, plus a skip, are transferred to the memory buffer to be stored as the return address upon completion of the Interrupt routine. The INTERRUPT cycle is completed at T1 by incrementing the new contents of the MA register by one, and transferring the next sequential address of the Interrupt routine to the Program Counter.

3.3.3 Data Break Transfers (BREAK)

The flow diagram for both single- and three-cycle Data Breaks is shown on Volume III drawing D-FD-PDP-12-0-23.

Single-Cycle Data Break – The single-cycle Data Break cycle is entered at coordinates D,3 of the flow diagram. At T1 of this cycle, the address from the peripheral hardware address register is gated into the Memory Address registers to the Break Field registers. If extended memory is used, three additional memory address bits are also gated. The instruction register is cleared at T2.

At T3, one of three events occurs. If the peripheral is transferring data into core memory, data from the peripheral is loaded into the Memory Buffer register. If the peripheral is accepting data from core memory, data at the selected address is loaded into the Memory Buffer register and transferred via the BMB lines (one ICB). If the

peripheral is operating as a Memory Increment device, no data is transferred, but the contents of the selected address (corresponding to the sample value of the parameter being measured by the peripheral) are incremented by one.

The BREAK cycle is completed at the end of T3, and control returns to the program in progress at the time the BREAK occurred with T1 of the FETCH cycle.

Three-Cycle Data Break – The three-cycle Data Break is entered at coordinates D,7 of the flow diagram, which is T1 of the WORD COUNT cycle. The address of the Word Count register in core memory must still be provided; because this is a fixed address, the peripheral may be hardwired to provide it, thus eliminating a hardware register in the peripheral. If extended memory is used, three additional Memory Address bits are also gated in. The instruction register is cleared at T2.

At T3 of the WORD COUNT cycle, the core memory Word Count register is incremented by one. The Word Count register was previously loaded with the negative value of the number of words in the core memory data table for the peripheral. Because the Word Count register is incremented each time a transfer occurs with this peripheral, the complete table transfer is completed when the word count is zero. This condition is indicated by an overflow out of the sign bit of the Memory Buffer register (IOO WC OVERFLOW). The CURRENT ADDRESS (dwg. *CPS*) flip-flop is now set, and the break continues immediately with the CURRENT ADDRESS cycle of the three-cycle Data Break.

During T1 of the CURRENT ADDRESS cycle, the second cycle of a three-cycle Data Break, the MA register is incremented by one to obtain from core memory the contents of the Current Address register. The instruction register is cleared at T2. Normally at T3, the contents of the Current Address register are incremented by one and loaded into the MB. This leaves the CURRENT ADDRESS of the next core memory data table word in the Memory Buffer register. Under special applications, the increment of the Current Address register may be inhibited and the register contents are returned to core memory unchanged. Because the contents of the Current Address register are incremented before they are used as the address of the word in the data table, the starting address of the data table must be specified by initializing software as (address-1) in order that the first (and subsequent) transfers occur at the correct core memory address. The BREAK flip-flop is now set, and the three-cycle Data Break continues into the BREAK cycle.

The BREAK cycle for the three-cycle Data Break is identical to the single-cycle Data Break, with the following important differences. First, the core memory address at which the data transfer occurs is transferred directly from the Memory Buffer register (still holding the incremented CURRENT ADDRESS) to the Memory Address register. Second, it is not possible to use a three-cycle Data Break to build a histogram, because the histogram requires the peripheral to supply the memory address from its data output. In a three-cycle Data Break, 12 bits of the core memory data address are supplied from the Current Address register.

3.4 BLOCK SCHEMATIC DESCRIPTIONS

The discussions under this heading briefly describe the logic presented on the five I/O block schematics in Volume III of this manual. Refer to Chapter 5 of the *PDP-12 System Reference Manual* for timing diagrams of I/O Bus operations.

3.4.1 I/O and External Memory Cables (D-BS-EP12-0-ICB)

All external I/O Bus connections to the PDP-12 are made via module sockets at mainframe coordinates N14 through N18. The socket at N14 provides connections for the IOO BAC (to peripheral) Bus; the IOT Buffered In/Out Pulses (BIOPs 1, 2, 4); Buffered Time States 2 and 5, available to peripherals for synchronizing events; and the Initialize Bus, available to all IOT peripherals.

The Buffered Memory Bus output is provided to the peripherals via mainframe module socket N15. The logic true level output is high for all twelve bits; in addition, inverted form logic is provided for bits 3 through 8 in order to facilitate setting up device code decoders in addressable (IOT) peripherals. This socket provides the data output to Data Break peripherals.

AC inputs (EXT I/O Bus) from IOT peripherals are accepted at mainframe module socket N16. This socket also accepts I/O Bus signals for Interrupt Request, Skip, and Clear Accumulator. A Buffered Output Bus signal from the PDP-12 RUN flip-flop (dwg. *CPR*) is also provided at this socket.

Core memory addresses are accepted from Data Break peripherals via the mainframe module socket at N17. This socket also provides the connections for the following Data Break I/O Bus signals: Break Request, Increment Memory Buffer, and Data In/Data Out Control. Data Break Output Bus signals provided at this socket are the Buffered Break Pulse (active for the duration of the BREAK cycle), the Address Accepted Pulse (used by three-cycle peripherals), and one Initialize Pulse for use by Data Break peripherals.

Data inputs from Data Break peripherals are accepted via mainframe module socket N18. This socket also contains connections for Data Break I/O Bus inputs: three-cycle/single-cycle control, and Current Address Increment (for use by three-cycle peripherals). If extended core memory is used, the additional three address bits from the peripheral are also accepted at this socket. The Word Count Overflow output to three-cycle Data Break peripherals is also routed via this socket.

The remainder of the sockets shown on this drawing are used for Extended Memory and External Level signals (may be used as flags from non-digital external equipments).

3.4.2 I/O Inputs Part A (D-BS-EP12-0-IOA)

The bus receivers for Data Break peripheral data input and address buses, and the data input gates for IOT peripherals (B-D, 1-2) are shown on this block schematic.

3.4.3 I/O Inputs Part B (D-BS-EP12-0-IOB)

This block schematic contains the IOT input bus gates: Interrupt Request, Skip, and Clear Accumulator. The Data Break request bus inputs: Break Request, Three-Cycle/Single-Cycle, Increment Memory Buffer, Data In/Data Out, and Increment Current Address are shown at coordinates C-3, 4. The Data Break extended core memory address bus receivers are shown on this drawing. The pullup resistors shown at B-D, 2 are part of the Program Interrupt bus.

3.4.4 I/O Control and Timing (D-BS-EP12-0-IOC)

Control elements for IOT and Data Break transfers are shown on this drawing.

IOT transfers are initiated when the IOT PAUSE (dwg. *CPT*) flip-flop is set. This signal is introduced to the I/O control logic to reset the third bit of a three-bit shift register (C, 6-8). The IOT PAUSE flip-flop also initiates the IOP delay line pulse generator logic (B, 1-6) and sets the IOP SAMPLE flip-flop (C, 4).

The IOP SAMPLE flip-flop enables the three IOP output gates at location B, 4-8. The Enable is ANDed in these gates with the three flip-flops of the shift register, and with the least significant three bits of the IOT instruction. The IOT bits determine which of the IOP pulses are to be gated out to the peripheral; the shift register determines the timing of the selected pulse or pulses. The shift register and pulse generator delay line determines IOP pulse timing.

The shift register and delay line operate as follows. On the trailing edge of Central Processor TP5 (Time Pulse 5), following setting of the IOT PAUSE flip-flop, the delay line input is initiated and the third flip-flop of the shift

register is reset, thus completing the conditions required to set the first flip-flop of the register which controls IOP 1. The SAMPLE flip-flop is also set when the delay line propagation begins. Therefore, if the least significant bit of the IOT instruction is set, BIOP 1 is immediately available on the bus.

The I/O STROBE pulse occurs 750 ns after CPTP TP 5 and CPT I/O PAUSE. Part of the delay is caused by the delay line which is trapped at 500 ns, and the other part by propagation delay through gates and pulse amplifiers. If the least significant octal digit of the IOT instruction is other than 0 (location B, 1-3), the STROBE pulse is used for gating within the processor.

At 650 ns into the delay line, the IOP SAMPLE flip-flop is reset by the IOC OFF IOP H at location C, 1. This terminates the IOP currently in progress. After 1200 ns, the propagated pulse is returned to the input of the delay line via the IOC I/O CLOCK H at location B, 2. This causes an IOC IOT SHIFT H pulse, and the process is repeated with IOC IOP 2 flip-flop set.

The IOT PAUSE flip-flop is reset at the completion of IOC IOP 4 H (generated by the IOC IOP 3 flip-flop) by ANDing the IOC IOP 3 flip-flop with the IOC Off IOP H pulse.

The IOC I/O PRESET L signal used to produce the IOO BA INITIALIZE H signal is developed in logic shown on this drawing. The WORD COUNT OVERFLOW and ADDRESS ACCEPTED flip-flops used with Data Break peripherals are shown at locations D, 6 and D, 5 respectively.

The Interrupt facility logic is shown at location D, 4-2.

3.4.5 I/O Output Buffers (D-BS-EP12-0-IOO)

The output buffers (bus drivers) for standard IOT and Data Break peripherals are shown on this drawing. At locations B-D, 7-8, the Data Output Buffers for IOT peripherals are shown. The data output buffers for Data Break peripherals are shown at locations B-D, 4-5 and C-D, 1-2. Bits 3 through 8 of this bus are also used to address IOT peripherals; therefore, both true and inverted forms of these bits are provided to facilitate setting up the device select decoders within the peripherals.

The IOP buffers for IOT peripherals are shown at B-C, 1-2. CP time state buffers for TS2 and TS5 are also provided and are shown at B, 1-2. The signal IOO BA INITIALIZE H and IOO BB INITIALIZE H are generated on logic shown on this drawing at B, 1-2.

The bus signal buffers for CP RUN, BREAK, ADDRESS ACCEPTED, and WORD COUNT OVERFLOW are shown at A-B, 4-8.

CHAPTER 4
TELETYPE

4.1 INTRODUCTION

The Teletype® Model 33 ASR is used on all standard configurations of the PDP-12 Computer and is connected internally to the I/O Bus (dwg. *IOA*). The Teletype is accessed for input or output by programs in either the LINC or PDP (dwg. *CPR*) operating mode. The Teletype is equipped with a paper-tape reader and punch; the reader and keyboard use the same input path and instructions, while the printer and punch use the same output path and instructions. The maximum transfer rate in either direction is 10 characters per second. The Teletype has both *full-duplex* and *half-duplex* capability. In full-duplex operation, data may be transmitted in both directions simultaneously, without interference. In half-duplex operation, data may be transmitted in only one direction at a time.

Although other Teleprinter and tape devices are compatible with the PDP-12, only the Model 33 ASR is considered in this discussion.

4.2 TELETYPE CONTROL

The ASR Teletype and the Teletype Control functions (see Figure 4-1) are logically divided into two sections: the receiver (dwg. *TTI*) or Teletype input (TTI) which consists of the keyboard, tape reader, and associated Teletype control circuitry for computer input; and the transmitter (dwg. *TTO*) or Teletype output (TTO), which consists of the Teleprinter, tape reader, and Teletype control circuitry for computer hardcopy output (printed page or perforated tape).

For output, the Teletype Control converts an 8-bit ASCII character code into an 11-unit serial code, and transmits it to the Teleprinter/punch. For input, the control assembles the serial code into an 8-bit character and places it on the I/O Bus for transmission to the AC.

A free-running 880-Hz clock, along with respective clock dividers (one in the receiver and one in the transmitter circuitry), provides the means of synchronizing the computer to the 110-Hz rate of the Model 33 ASR Teletype. During the receiving or transmitting of the 11-unit Teletype word, the Teletype Control senses the last two units (stop pulses) to maintain a positive synchronous lock.

4.3 RECEIVER (TTI)

The Teletype Control performs a Read operation in which an asynchronous serial data word from the Teletype keyboard or tape reader is clocked into the receiver TTI shift register, where it is held as an 8-bit word for parallel transfer to the AC. The readiness of the device for a transfer is signaled by a status signal (KEYBOARD FLAG (1) L) which results in an Interrupt Request to the processor. At the program-determined time, the assembled

® Teletype is a registered trademark of the Teletype Corporation.

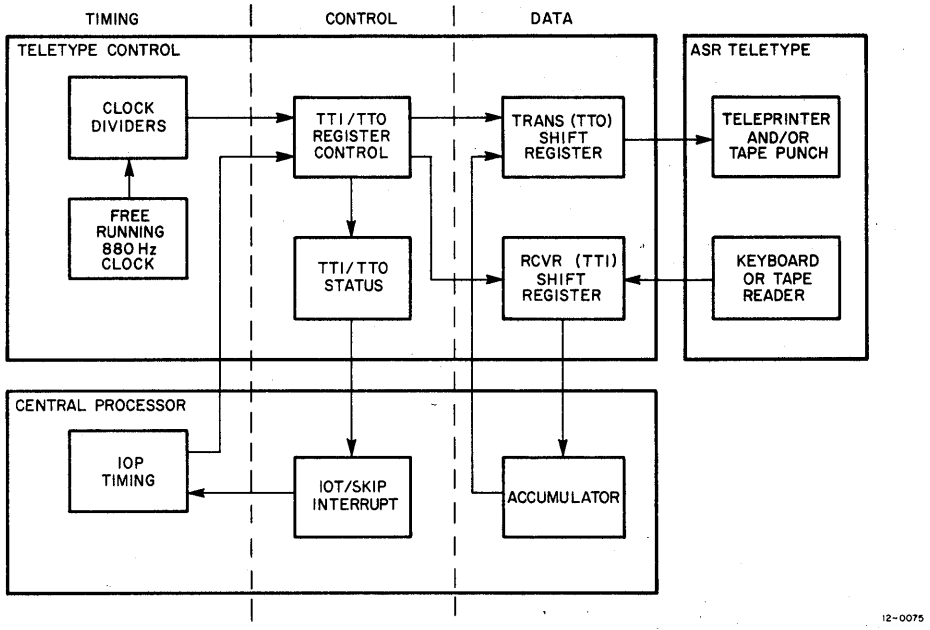


Figure 4-1 Teletype Control Block Diagram

8-bit word is strobed in parallel into the accumulator via the I/O Bus. The receiver (dwg. *TTI*) is then cleared and ready to repeat the transfer data.

4.4 TRANSMITTER (TTO)

The Teletype Control TTO performs a print and/or punch operation in which a parallel 8-bit word from the AC is loaded into the TTO shift register, from which it is clocked as an asynchronous serial 11-unit word, to the ASR-33 Teletype unit. At the Teletype unit, the word is decoded and printed and/or punched on tape. Transfer control between the AC and the Teletype is accomplished by a programmed Teletype instruction. When the Teleprinter Flag is set, the parallel data transfer from the AC to the TTO shift register is programmed, and the 110-Hz clocking of the word to the Teletype decoder begins. The TTO register control and shift register are then cleared to receive the next word. A detailed logic description is provided for the Read cycle, the Print/Punch cycle, and the program instruction set for Teletype operation.

4.5 DETAILED LOGIC DESCRIPTION

The following paragraphs describe the Read (dwg. *TTI*) and Print/Punch (dwg. *TTO*) cycles, and the program instructions for Teletype operation.

4.5.1 Read Cycle

Figure 4-2 shows the sequence of events that occur when a serial word from the Teletype unit is assembled in the receiver. The receiver circuitry is shown on engineering drawing D-BS-EP12-0-TTI.

The clock (dwg. *TTO*) is free-running and produces TTO TTI CLOCK H pulses at 880 Hz for the receiver, and, by use of two flip-flop frequency dividers, 110 Hz TTO SHIFT pulses for the transmitter.

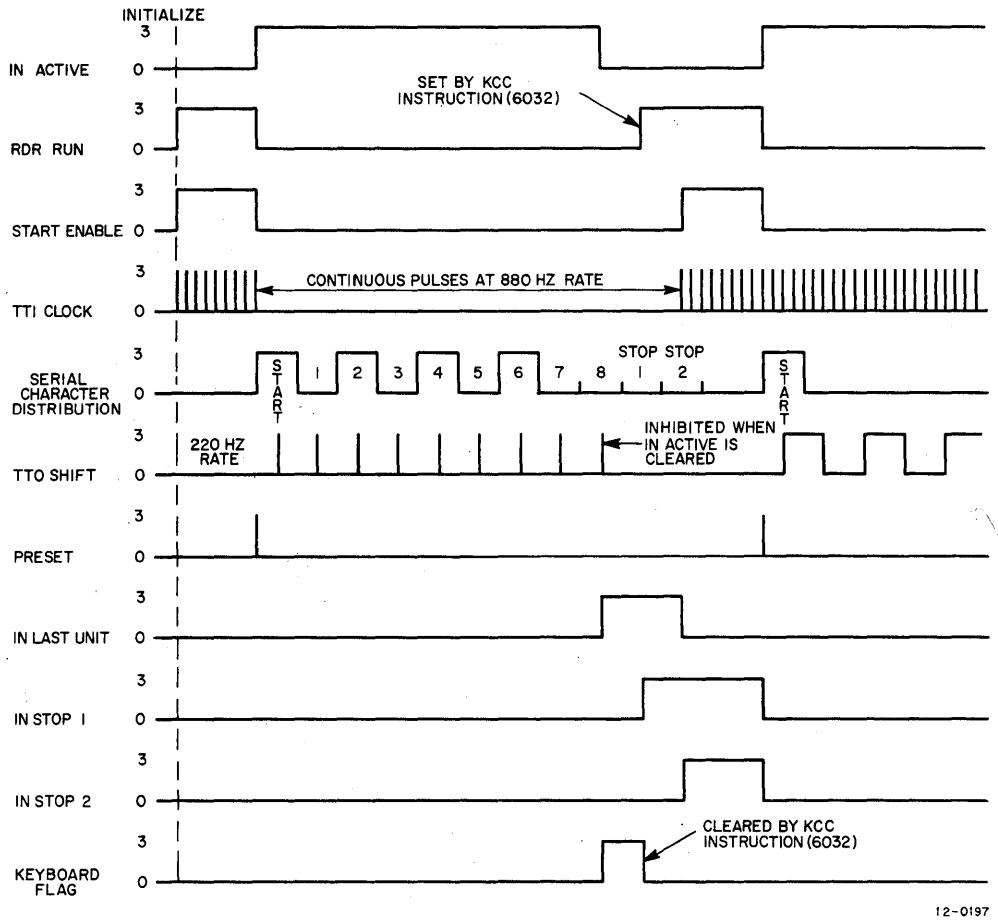


Figure 4-2 Teletype Read Cycle Timing Diagram

When the computer is turned on or the CSI START (START LSW, START 20, or START 400) key is depressed, the processor IOC I/O PRESET H level is generated. IOC I/O PRESET H clears the TTI IN ACTIVE (1) L flip-flop, the TTI SPIKE DETECTOR (1) H flip-flop, and generates TTI KCC L to clear the TTI Keyboard Flag and set the TTI READER RUN flip-flop. Clearing the TTI IN ACTIVE (1) L flip-flop generates the TTI START ENABLE L signal. This signal, combined with the start bit (produced in the Teletype distributer by program control) or the output of the Schmitt trigger, allows a TTO TTI CLOCK H pulse to generate TTI REC PRESET L. The TTI REC PRESET L pulse allows the serial data from the Teletype to synchronize with the receiver logic in the following manner:

> TTI REC PRESET L sets the TTI IN ACTIVE flip-flop and the TTI register flip-flops to a binary 1, and clears the TTI READER RUN flip-flop. With TTI READER RUN cleared, a relay in the ASR tape reader is energized and releases the tape-feed latch; tape motion is thus stopped only between sensing of the end of character and the beginning of the next character. In addition, the TTI SPIKE DETECTOR flip-flop is set to sample the line after the start bit is received. If the start bit is not present at this time, the TTI IN ACTIVE flip-flop clears and awaits another TTI Preset signal. This eliminates noise on the start bit in the first 1/2 unit. If the start bit is still present, e.g., no false start due to noise, TTI shift pulses are generated and the start bit is loaded (a binary 0) into TTI 0. The shift pulses are synchronized to occur during the middle of each serial

bit-time. As Figure 4-2 illustrates, each succeeding TTI SHIFT H pulse loads the character bits into the receiver serial shift register. When the start bit is shifted into TTI 7, the next TTI SHIFT H pulse produced sets the TTI IN LAST UNIT and TTI KEYBOARD FLAG (1) L flip-flops. Setting of the TTI IN LAST UNIT clears the TTI IN ACTIVE flip-flop, disabling further TTI shift pulses. When cleared, TTI IN ACTIVE allows the TTI CLOCK SCALE 2 pulses to set the TTI IN STOP 1 and TTI IN STOP 2 flip-flops, which count out the stop time. TTI IN STOP 2 when set, clears the TTI IN LAST UNIT flip-flop, thus allowing Start Enable to produce TTI REC PRESET when the start bit of the next character appears at the output of the Schmitt trigger.

When the KEYBOARD FLAG (1) L flip-flop is set, TTO TT INT L (dwg. *TTO*) is generated, TTO TT INT L generates IOB INT RQST H in the CP. If the Program Interrupt facility is enabled, IOB INT RQST H indicates to the PDP-12 that a device is requesting service. The program then enters a Search subroutine to determine which device issued the Interrupt. This is accomplished by executing a series of flag-checking Skip instructions. When the Keyboard Flag-sensing instruction KSF (6031) is performed, and the flag is raised, TTI SKIP (dwg. *TTI*) is generated, producing IOB I/O SKIP H in the CP. IOB I/O SKIP H forces the processor program counter to increment by one; thus the next instruction is skipped. A service routine for the Teletype receiver is entered when the skip occurs. In this routine, the TTI Keyboard Flag is cleared, TTI READER RUN is set to release a new serial word to the receiver, and the previously assembled word is strobed in parallel to the processor.

4.5.2 Print/Punch Cycle

Figure 4-3 illustrates the sequence of events that occur when a parallel word from the AC is loaded into the transmitter logic, and disassembled into serial word format for use by the Teletype unit. The transmitter logic is shown on engineering drawing D-BS-EP12-0-TTO.

When the computer is turned on, or when the START (START 20, START 400, or START LSW) key is depressed, the processor IOC I/O PRESET L level is generated. This level clears the TTO ENABLE, TTO OUT ACTIVE, TTO TPR FLG (1) L (Teleprinter Flag), and TTO register flip-flops. Execution of the program instruction TLS (6046) generates TTO SELECT H in the transmitter logic, and IOP2 and IOP4 in the processor. IOP4 and TTO SELECT H combine in the transmitter logic to load the parallel word (AC bits AC04 through AC11) into the TTO register, and set the TTO ENABLE flip-flop. IOC IOP2 H and TTO SELECT H are combined to clear the Teleprinter Flag.

TTO shift pulses are produced by dividing TTO CLOCK L pulses with TTO FREQ DIV. The frequency at the TTO CLOCK L output is 220 Hz, and is provided by the M452 CLOCK Module at N08 (dwg. *TTO*). The output of TTO FREQ DIV flip-flop is 110 Hz. Synchronization is provided by using both of these clock pulses as shown in Figure 4-3, and described below.

When the first TTO CLOCK pulse following the setting of TTO ENABLE occurs, the following events take place: the OUT ACTIVE flip-flop is set, the start bit is placed on the line to the Print Selector Magnet driver (TTO line), and TTO OUT STOP is set (discussed later). Alternate TTO CLOCK pulses produce a TTO SHIFT pulse, which shifts the data word bit-by-bit into the LINE flip-flop. As data is shifted onto the PSM line, binary 0s are shifted into the TTO register through the TTO ENABLE flip-flop. When the parallel-to-serial conversion is completed, an 8-input NAND gate senses all 1s in the TTO register, and its output TTO=0 enables the Teleprinter Flag to be set simultaneously with the next (and last) TTO SHIFT pulse. When TTO OUT ACTIVE is cleared, TTI SHIFT is disabled, and the two 1-unit stop bit times are counted out by the TTO OUT STOP register. The first TTO CLOCK pulse occurring after TTO ACTIVE was cleared, clears the TTO OUT STOP 1 flip-flop. The TTO OUT STOP 1.5 and the TTO OUT STOP 2 flip-flops are cleared by the next two consecutive pulses. If another TLS instruction has been issued (Print/Punch Another Character), the operation commences with the occurrence of the first

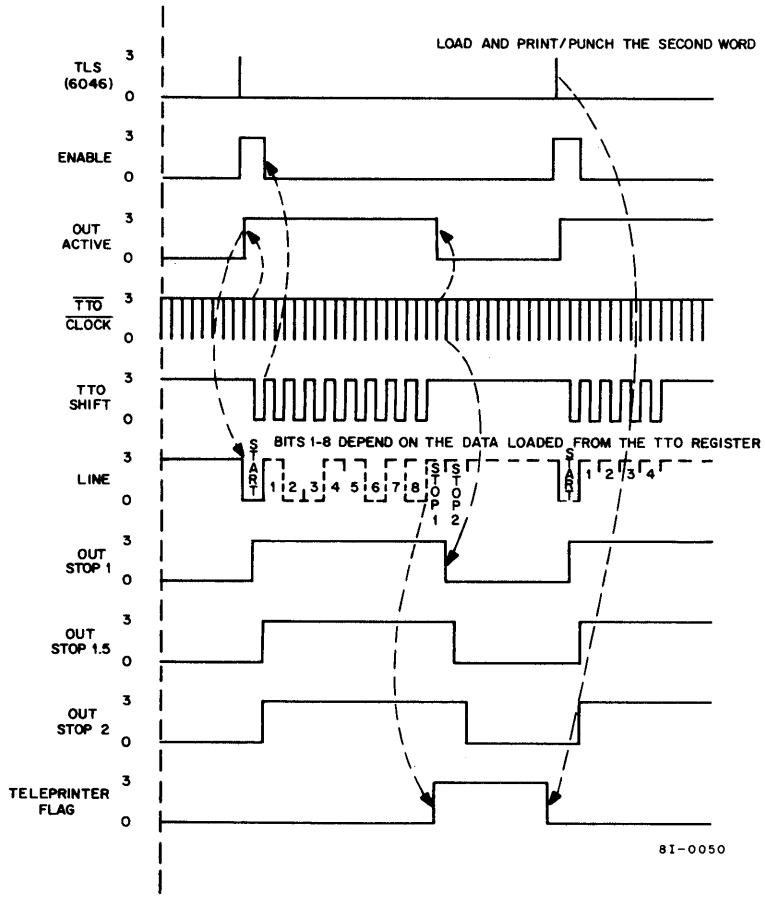


Figure 4-3 Print/Punch Cycle Timing Diagram

TTO CLOCK pulse after TTO OUT STOP 2 is cleared. For the print/punch operation with the ASR-33, a 2-unit OUT stop time period is required because of the inherent electro-mechanical delay time in the Teletype unit.

When the Teleprinter Flag is set, TTO TT INT L (dwg. *TTO*) is generated. TTO TT INT produces IOB INT RQST H in the processor. If the Program Interrupt facility is enabled, IOB INT RQST H indicates that a device is requesting service. The program then enters a Search subroutine to determine which device issued the interrupt. This is accomplished by executing a series of flag-check Skip instructions. When the Teleprinter flag-sensing instruction TSF (6041) is performed and the flag is raised, TTO TT SKIP is generated, producing I/O SKIP in the processor. I/O SKIP forces the program counter to increment by one; thus, the next instruction is skipped. A service routine for the Teletype transmitter is entered when the skip occurs. For the transmitter service, a new character is transferred to the transmitter to be printed or punched by the Teletype unit. Table 4-1 contains descriptions of the Teletype keyboard/reader and the Teleprinter/punch instructions.

Table 4-1
Teletype Instruction Description

Mnemonic	Octal Code	Description
KSF	6031	Generates IOC IOP1 to sense the status of the TTO Keyboard Flag. When the flag is set, the next sequential program instruction is skipped. This indicates that the assembled word is ready for transfer to the AC.
KCC	6032	Generates IOC IOP2 to clear the Keyboard Flag and set TTI READER RUN flip-flop. In addition, TTO TT AC CLEAR is generated to clear the AC.
KRS	6034	Generates IOC IOP4 to transfer the assembled word in parallel to the AC through the major register bus. The Keyboard Flag flip-flop is not cleared.
KRB	6036	Generates IOC IOP2 and IOC IOP4 to perform the functions of the KCC and KRS commands during a single computer cycle. The Keyboard Flag flip-flop is cleared.
TSF	6041	Generates IOC IOP1 to sense the status of the Teleprinter Flag. When the flag is set, the next sequential program instruction is skipped. This indicates that the print/punch operation has completed.
TCF	6042	Generates IOC IOP2 to clear the Teleprinter Flag.
TPC	6044	Generates IOC IOP4 to load the parallel word into the transmitter register, initiating the print/punch operation.
TLS	6046	Generates both IOC IOP2 and IOC IOP4 to combine the functions of the TCF and TPC instructions in one computer cycle.

CHAPTER 5
LINC DEVICES

5.1 INTRODUCTION

The I/O control devices in this chapter are control modules that interface with the PDP-12A LINC System. All other peripherals are considered optional I/O devices, and are omitted from this discussion.

The LINC System I/O devices are as follows:

Tape Control	Relay Control
A/D Control	Display Control
Sense Line Control	Speaker Control

5.2 TAPE CONTROL

The TU55 and TU56 Transports are controlled by a fully-buffered tape processor; once initiated by the LINC program, tape operations are carried out independently of the CP. Tape control is described in detail in Chapter 6 of this manual.

5.3 A/D CONTROL

Sixteen analog input channels are provided with the PDP-12A System. The first eight channels are wired to potentiometers, all of which provide a variable reference voltage of -5V to +5V. The potentiometers can be varied by control knobs located on the data terminal panel. The second eight channels can be plugged into the data terminal panel through standard phone jacks, for which the inputs should be ±1V.

An additional 16 channels (20 through 27 and 30 through 37) can be added as an option to the PDP-12A. This option (AM12 and AG12) is described in Chapter 7 of this manual.

Figure 5-1 illustrates the relationship of the CP to the A/D Converter. A SAM instruction decoded in the CP is routed to the A/D Control, where the A/D Converter is started to convert an analog input signal. The control, timing, addressing, and data relationship is illustrated in Figure 5-2. There are two types of inputs to the A/D Converter. Channels 0 through 7 are potentiometer inputs that can be manually adjusted by the operator. Channels 10 through 17 are external analog inputs that are applied to differential preamplifiers. The preamplifiers provide impedance matching, common-mode rejection, overload protection, and isolation to the computer and external reference source. Both input types described above are routed to FET-switched multiplexers. The SAM (Sample) instruction selects one channel at a time for A/D conversion. The selected channel is applied to the SAMPLE & HOLD circuit through a buffer amplifier. The SAMPLE & HOLD circuit (D-BS-AD12-0-YAD) establishes the level of the analog voltage signal and applies it to the A/D Converter, where a 10-bit A/D conversion takes place. When the conversion is complete, the contents of the conversion buffer are strobed in parallel to the AC. The numerical representation of the signal is placed in AC bits 3 through 11; the sign is in AC bits 0 through 2. The total time from the selection of a multiplexer channel until the actual word is placed in the AC is approximately 19 μs. The PAUSE flip-flop is set at this time, and the processor waits until the instruction is completed. The program for continuous sampling of channel 1 is as follows:

SAM 1	19.2 μs
STA 1	4.8 μs
XSK 1	3.2 μs
JMP.-3	3.2 μs
TOTAL	30.4 μs

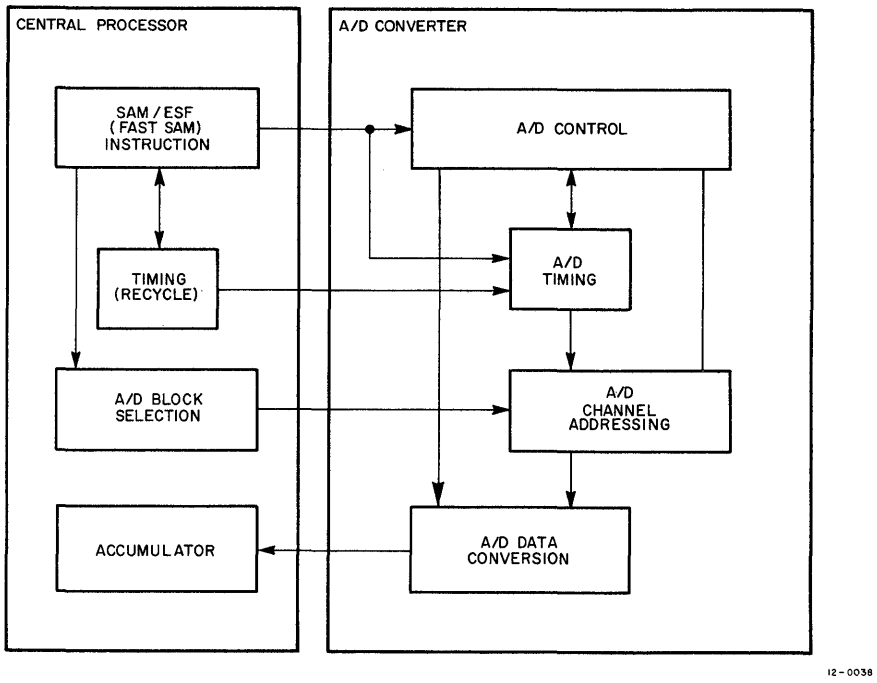


Figure 5-1 A/D Control General Block Diagram

It can readily be seen that the sampling frequency would be $1(30.4 \times 10^{-6})$, or approximately 33 kHz. The highest sampling frequency for five channels would be approximately 6 kHz. As the number of samples is increased, the sampling rate is decreased, restricting input bandwidth possibilities. The ESF instruction before a SAM instruction solves this problem (see FAST SAMPLE). A single LINC-Mode SAM instruction is used for A/D control during a LINC FETCH state.

A FAST SAMPLE (special function) instruction, which is a SAM instruction initialized by an ESF instruction, also institutes an A/D conversion readout to the Central Processor AC. With a FAST SAM, the order of events is reversed: the current contents of the converted buffer are transferred to the AC, then the specified channel (as addressed in the octal code of the current SAM instruction) is selected and a new conversion commences. The results of this new conversion can then be read by a subsequent FAST SAM instruction. A time advantage is realized because less than 1.6 μs is required to execute a readout of the conversion buffer to the Central Processor AC. With a FAST SAM, the requirement for the CP to pause for 19 μs is eliminated provided that a previous SAM instruction has been given and 19 μs has elapsed between the previous and current SAM instruction.

When a FAST SAMPLE instruction is decoded and the YADC A D BUSY flip-flop is not set (indicating that a previous conversion is still in progress), the AC is loaded with the data from a previous instruction by signal YADC LOAD IR H (as shown on the *YADC* print) and a new conversion is started by YADC START H pulse.

The conversion in progress can be read only when the conversion is done, about 19 μs later, when another FAST SAM instruction is given.

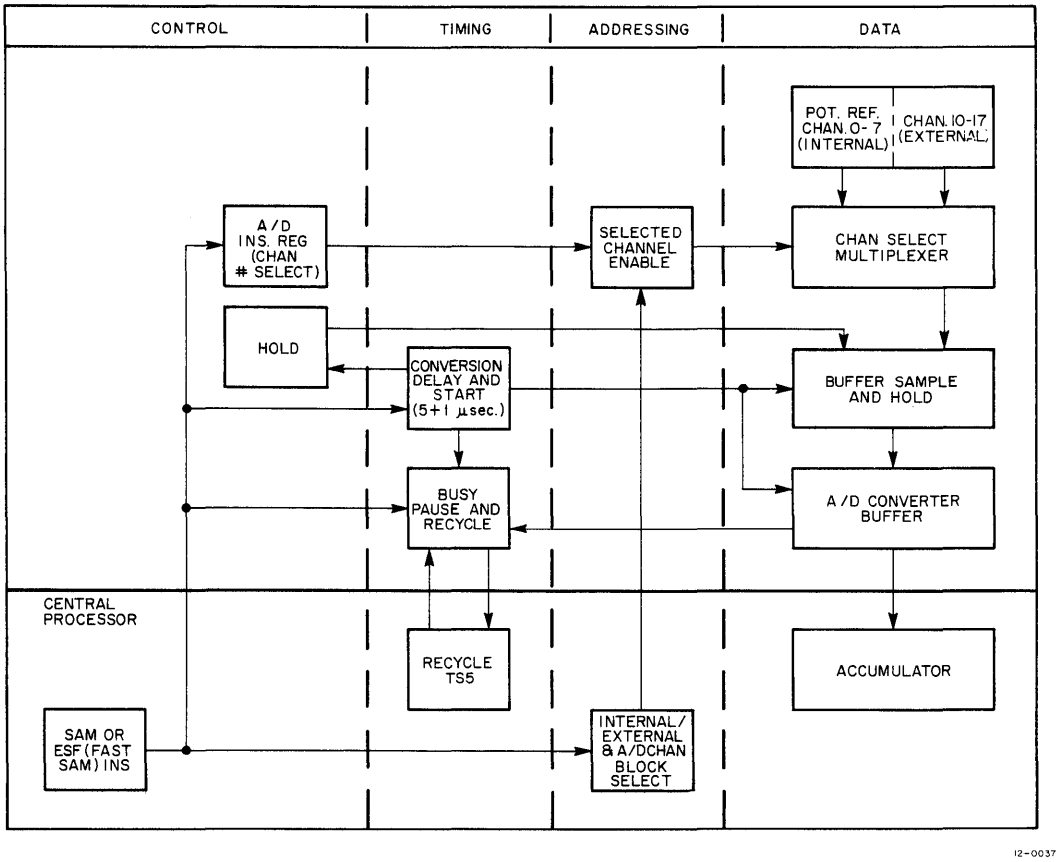


Figure 5-2 A/D Control Block Diagram

5.4 CP FLOW DIAGRAM DESCRIPTION

SAM Instruction

At T4 of the FETCH cycle, the RECYCLE SYNC flip-flop (dwg. *CPT*), is direct-set by signals YADC A D RECYCLE L and CPTP TP4 H. YADC RECYCLE L was previously enabled by signals YADC A D PAUSE (1) H, YADC FAST SAM (0) H, and INS SAM H. YADC A D PAUSE was left set (1) from the previous SAM instruction or I/O PRESET. Simultaneously, YADC LOAD A D IR H is enabled to load YADC IR 07 through 11 with the channel to be sampled. This number is decoded by the A131 Multiplexers (dwg. *YAD, YADA*, and *YADB*). The A/D Control enters a 5 μs settling delay to permit the chosen amplifiers to reach a stable state. YADC A D BUSY was set at the beginning of this transition. The trailing edge of the 5 μs delay sets up a 1 μs delay and sets YADC HOLD. YADC HOLD (1) L is used to begin a HOLD cycle on the A404 Sample & Hold circuit (dwg. *YAD*) and the 1 μs delay gives the circuit time to stabilize. The negative-going edge of the 1 μs delay pulse produces a YADC START H pulse, which initiates an A/D conversion (dwg. *YAD*).

During the initial 5 μs time delay, CPTP OFF PAUSE H sets CPT T5 RECYCLE. Following this pulse, the occurrence of CPTP TP5 will clock a 1 into CPT TS5 for as long as CPT T5 RECYCLE is a 1. Hence, the CP is held in Time State 5 during the SAM instruction.

At each occurrence of a TP5 during the SAM instruction, the AC is loaded with the current contents of the A/D Converter. Signals YAD AD03 through 11 are inverted by inverters producing signals YAD AD03 through 11. These signals, along with the inverted YAD 2 are fed into the adder via PMA BMSC 00 through 11 (dwgs. *PRA, PRB, PRC, PRD, PRE,* and *PRF*). RCL LD AH H is generated every TP5 to accomplish this.

The following table indicates the effect of inverting the last 9 bits of the A/D Converter to achieve a 1's complement result in the AC:

Table 5-1
Inversion Effects

A214 Preamplifier Input Voltage	A404 S & H Output Voltage	Resulting 10 bits in A811	Result in AC
+1000	0.000	0000	0777
+.985	0.075	0001	0776
.	.	.	.
.	.	.	.
.	.	.	.
+.005	+4.975	0777	0000
-.005	+5.025	1000	7777
.	.	.	.
.	.	.	.
.	.	.	.
-.985	+9.925	1776	7001
-1.000	+10.000	1777	7000

Note that YAD 02 is duplicated on BMSC 00 through 02 (dwg. *PMA*).

The signal YAD DONE H is generated by the A/D Converter after the result is complete. This signal clears YADC A D BUSY, YADC HOLD, and YADC A D PAUSE. YADC A D RECYCLE is now disabled. The next occurrence of CPTP TP5 clears CPT RECYCLE SYNC. The next CPTP OFF PAUSE clears CPT RECYCLE, allowing the following occurrence of TP5 to clear CPT TS 5 and set YADC A D PAUSE. Hence, timing is advanced to a new memory cycle (TS1), with YADC PAUSE set for the next SAM instruction.

Execution of a FAST SAM instruction permits the programmer to interrogate the contents of the A/D Converter (left from the previous conversion), strobe this data into the AC, and start a new conversion. This avoids the problems incurred with the regular SAM instruction (i.e., lower sample rate, possibility of missing a Request or an Interrupt during SAM execution, etc.).

As the FAST SAM instruction requires only 1.6 μs for execution, other programming can be accomplished before returning to initiate another FAST SAM (not less than 18.2 μs). The A/D Control is put into a FAST SAM mode by performing instruction ESF (0004) with AC05 = 1. This sets YADC FAST SAM. The 5 μs and 1 μs time delay string are generated in an analogous manner to the regular SAM instruction, but at CPTP TP5D. This allows the previous contents of the A/D Converter to be strobed into the AC at TP5 beforehand (as explained above).

Note that unless A D BUSY = 1 (dwg. *PADC*), YADC A D RECYCLE L is not generated, and the processor continues to the FETCH state after TP5. If a conversion was in progress, however, when FAST SAM was given, the CP will recycle in TS5 until the conversion is complete, read the final result into the AC at TP5, and start a new conversion at TP5D.

5.5 SENSE LINE CONTROL

As illustrated in Figure 5-3, the PDP-12 contains a total of 16 sense line inputs. These inputs provide the user with a facility to directly test external conditions such as relay closures, in addition to the normal array of peripheral I/O devices. Of the 16 external sense line inputs, three have been defined:

SXL	1	15	(KST)	Key Struck
SXL	1	16	(STD)	Tape Instruction Done
SXL	1	17	(TWC)	Tape Word Complete

The remaining sense lines require an input of +3V, and are reserved for the user to implement in his particular system. The succeeding program sequence is altered (skipped) under the following conditions:

a. I = 0 and the condition is met

b. I = 1 and the condition is not met

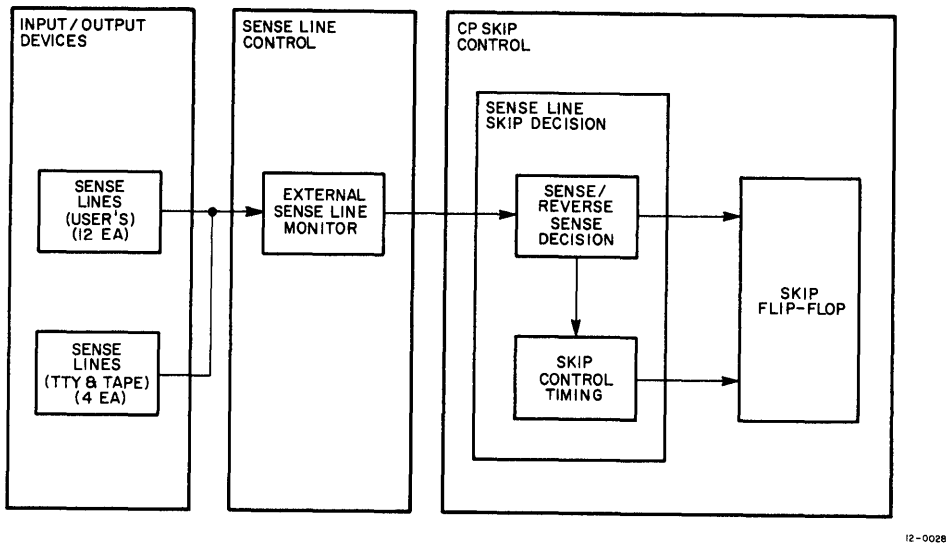


Figure 5-3 Sense Line Control Clock Program

As shown in Figure 5-4, the user's external sense line inputs are routed through the External I/O Input Bus (D-BS-EP12-0-IOB). Each of the 16 sense lines is applied to an AND gate (D-BS-EP12-0-SKL). The sense line inputs are ANDed with the program-requested skip condition indicated by N EQ 00 through 17. When a sense line level is at +3V and the associated N EQ input is also high, SKL X SENSE L is produced (dwg. *SKL*). SKL X SENSE L is ANDed with INS SXL L (see *SKH* print) and the SKIP flip-flop (D-BS-EP12-0-SKP) is set if I = 0. When SKL X SENSE is high, the SKIP flip-flop is set if I = 1.

5.6 RELAY CONTROL

The relay buffer is a six-bit register connected to six relays that are mounted on the data terminal panel. The relays can be used for controlling experiments or external equipment not otherwise directly interfaced with the PDP-12A Computer. The contact rating of the relays is 2A, at 28 Vdc non-resistive load. One A at 110 Vac resistive load is also acceptable. Contact closure time is approximately 20 ms. The condition of the relay (closed or open) is displayed on the console front panel. The six programmable double-pole, double-throw relays are activated by LINC-Mode instructions, and can be examined by the AC at any time.

As shown in Figure 5-5, the relay buffer register flip-flops are collectively clocked by the same signal (IOR LOAD RELAYS H). The IOR LOAD RELAYS L signal is enabled by INS MSC H · INS N EQ 14 H · CPTP TP5 H. A pulse (IOR LOAD RELAYS H) to the relay buffer enables the flip-flops to be set to the same states as the corresponding AC register flip-flops (bits 6 through 11 only).

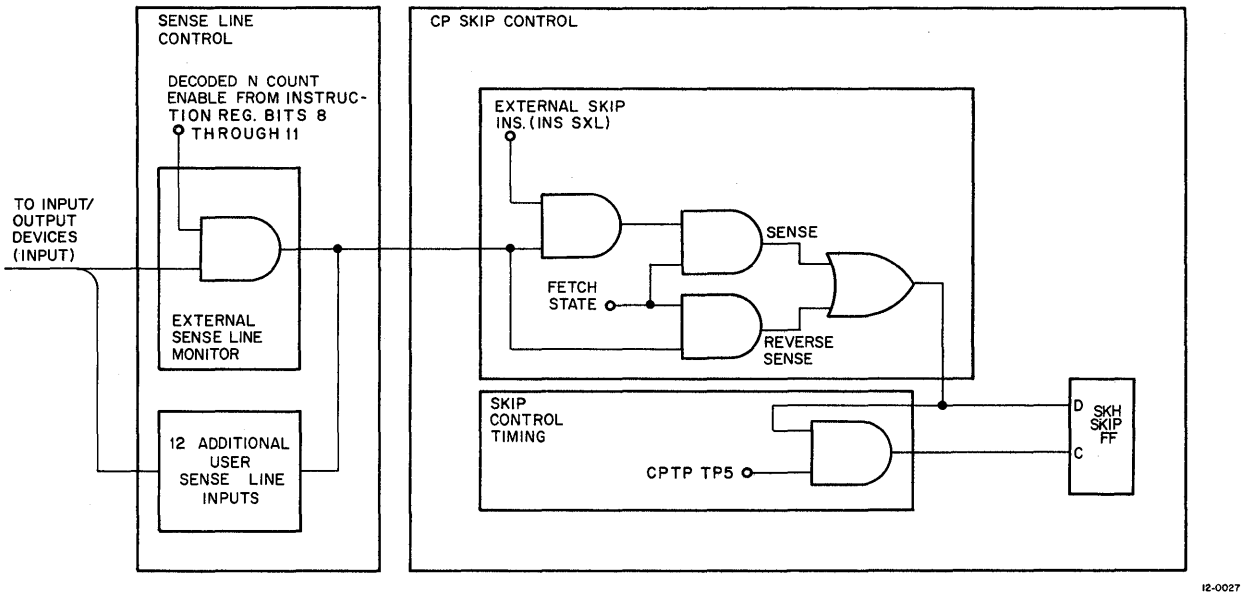


Figure 5-4 Sense Line Control Functional Block Diagram

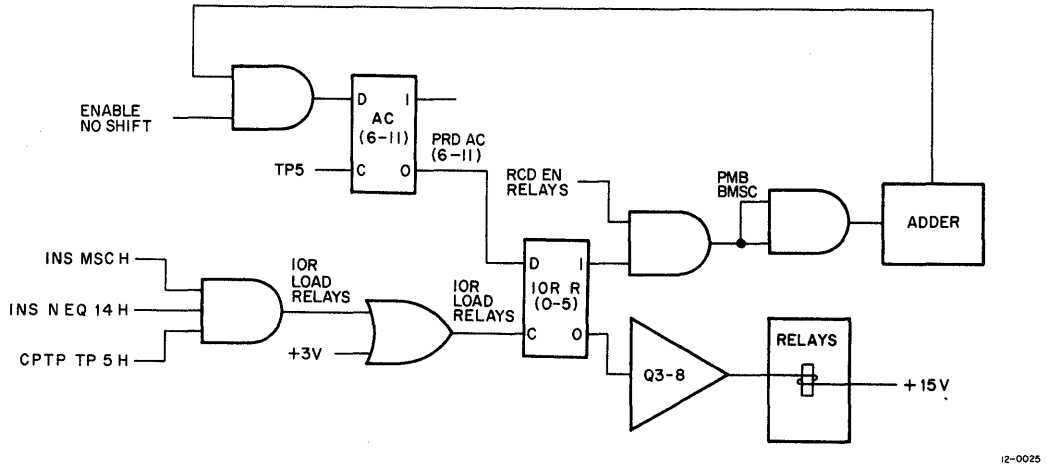


Figure 5-5 Relay Control Functional Block Diagram

When the AC flip-flop is set to 1, the associated relay buffer flip-flop is set, causing the relay to energize. The relay buffer flip-flops that are set ground the base junction of a corresponding relay driver/amplifier transistor (Q3 through Q8) located on the data terminal panel. The grounded base in turn causes the relay driver-amplifier to saturate, thereby energizing the relay armature.

Two separate programmed instructions are necessary to effect a transfer between the AC register and the relays: a LOAD AC instruction and an ATR (AC to relays) instruction. If the programmer wishes to initiate a readback from the relay register to the AC register, a RTA (relays to AC) instruction transfers the contents of the relay buffer register as follows: The output of the relay buffer to the AC is routed to the 1 side of the respective relay buffer flip-flops. The output is ANDed with the enable relay (RCD EN RELAYS H) signal, and applies through PMB BMSC 6 through 11 to the enable gates (dwgs. *PRA* through *PRD*).

5.7 DISPLAY CONTROL

The basic PDP-12 includes a rack-mounted 7 x 9 in. CRT display for presenting data in graphic, symbolic, and alphanumeric form. An interactive CRT display of data stored in memory or on tape is provided by use of the LAP6-DIAL program. Sample analog inputs or stored data in memory can also be presented in graphic form on either of the two channels on the VR12. Waveforms can be displayed simultaneously by turning the VR12 Channel Select switch to Channels 1 and 2. The Display instruction (DIS) brightens a spot in a 512 x 512 array at specified coordinates, *x* (vertical) and *y* (horizontal). The Display Character instruction (DSC) brightens spots in a subarray of 2 x 6 points, according to the pattern held in memory. Alphanumeric and other characters are generated on a point-by-point basis at high speed.

An optional display can be driven in parallel with the display scope for additional display capability. If desired, two different displays can be presented on each scope.

The potentiometer control knobs (numbered 0 through 7) mounted on the data terminal panel, while not directly related to the display system, can be used by the operator to set or vary display parameters used by the program (for example, to determine the size or position of the display). These potentiometers are directly connected to input channels 0 through 7 of the multiplexed A/D conversion system.

The *x* and *y* deflection voltages are derived from fully buffered 9-bit (0 to 6V) D/A Converters. These analog output voltages are available on the 24-contact blue ribbon connector of the data terminal panel, allowing the user to connect an auxiliary scope (VR12, Tektronix 503, or similar unit) for remote display of the same information sent to the PDP-12 display.

The pin assignments are:

1	Channel Select	9	Shield Ground
2	GND	10	Y HQ Ground
3	Shield Ground	11	Y Deflection
4	Intensified Pulse	12	Shield Ground
5	GND	13-18	Not Used
6	Shield Ground	19	503 Intensify
7	X HQ Ground	20	GND
8	X Deflection	21	Shield Ground
		22-24	Not Used

The output drive capability of the D/A Converters is -0.3V to -6V, which is capable of driving a load resistance of 1KΩ connected to ground. This allows up to 200 ft of cable for a remote display scope. These analog outputs can also be used for *x-y* plotters, auxiliary scope display, or any voltage-controlled device.

Figure 5-6 shows the relationship of the LINC Scope Control (VC12) to the CP and to the VR12 Point Plot Display. The VC12 Control allows the CP to continue with the main program without consuming any more CP time than the time necessary to decode the DIS or DSC instruction.

Only when a new DSC instruction is given while the previous instruction is still being executed by the Display Control, does the CP have to wait. Only the VC12 is covered in this manual; consult the *VR12 Point Plot Display Maintenance Manual*, DEC-CR-H6AA-D for information pertaining to the operation and maintenance of the VR12.

The Display Control discussion is divided into four functional parts (see Figure 5-7):

a. **Timing** – discusses the clock and delay circuits that control the order and sequence of events.

b. **Character Generation** – discusses the addressing (*x*-axis and *y*-axis coordinates) and shifting of intensification bits (display characters or spot).

c. **Addressing** – discusses the location on the Display Scope Face at which the Display Character or the intensified spot occurs.

d. **Data Handling** – discusses the relationship between the bit word and the character or spot displayed.

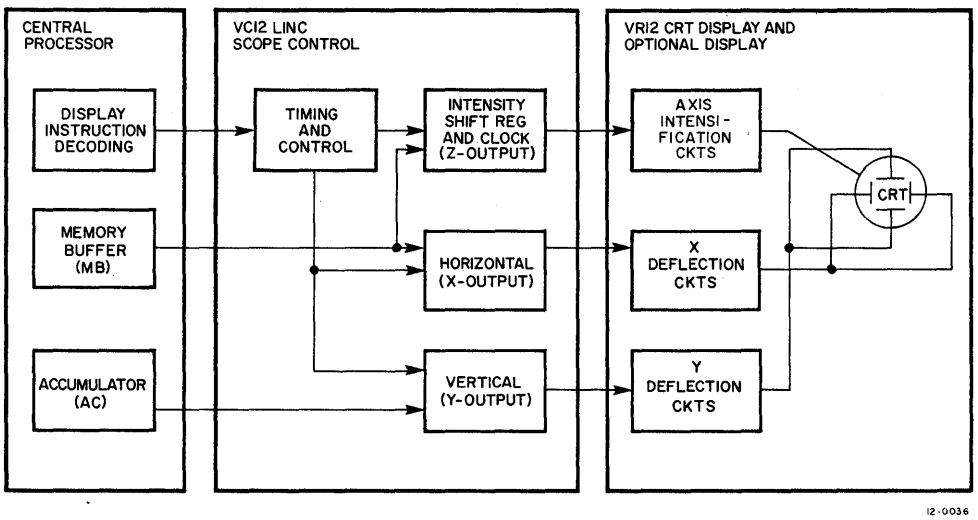


Figure 5-6 Display Control General Block Diagram

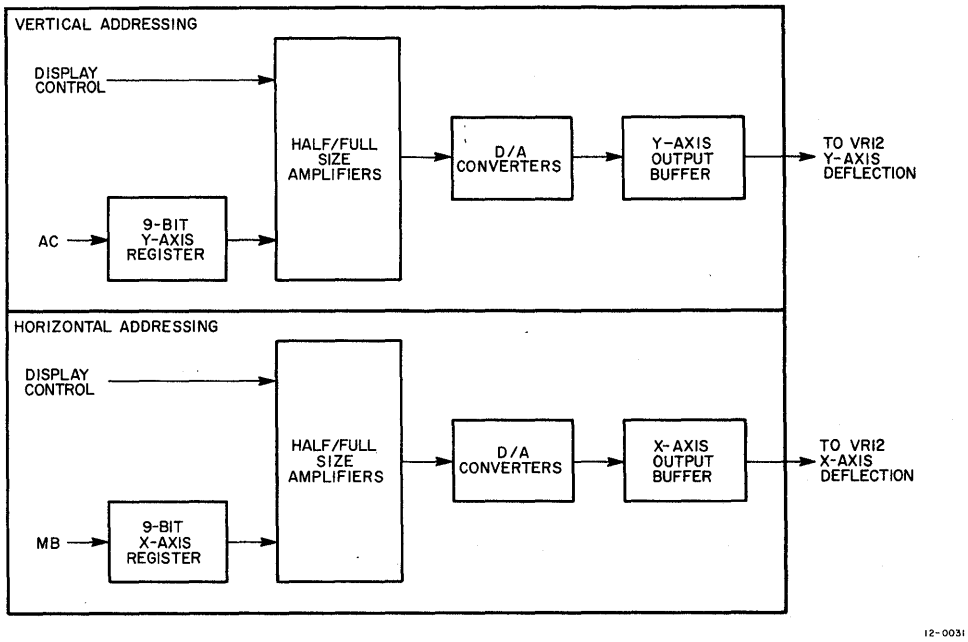


Figure 5-7 Display Control Block Diagram

5.7.1 Timing

In the previous paragraphs, the Display Control circuitry was described as *fully buffered*. Fully buffered in this context means that the Display Control itself initiates the timing or sequence of events. The CP can then advance to the next FETCH state and continue on with the next programmed instruction. The actual time consumed by the CP is 4.8 to 6.4 μs. A single DIS, however, requires 3.2 μs for execution, and not less than 25 μs for completion. Thus the CP would be idle for an unacceptable period of time if there was not a separate timing function in the Display Control circuitry itself. There are four distinct time intervals required for the display of an alphanumeric character or single point. The Display Control and D/A Converters are initialized for these time periods during the LINC EXECUTE (dwg. PDP-12-0-17 and -18), and EXECUTE 2 (dwg. PDP-12-0-20) cycles.

Display Instruction

The DIS instruction enters the EXECUTE cycle with the instruction appearing in the memory buffer (MB) and instruction register (IR) of the CP. At TP1 of the EXECUTE cycle, the MA is loaded with the address of the α-register containing horizontal coordinate information by RCL LOAD MA H. The MB is loaded with the contents of the α-register when the I-bit = 0 (bit 7), or the contents incremented by 1 if I=1 by RCL LOAD MA H at TP3. This specifies the horizontal coordinate.

The INTERNAL PAUSE flip-flop will hold the processor in TS5 if the display was previously BUSY. This is accomplished by CPT INT PAUSE being set at TP3 by DSC BUSY H and DSC or DIS during an EXECUTE cycle.

The absence of CPT INT PAUSE (0) H will now inhibit further generation of CPTP TP5. Hence, the computer is paused in Time State 5. When the previous display is complete, -CPT EN INT PAUSE H will enable generation of CPTP OFF PAUSE H and CPTP TP5 to clear CPT INT PAUSE and advance timing to the next memory cycle. When the program is in TS5 of the EXECUTE cycle, display registers are set up to intensify the point. DSX CHAN is loaded from MB00 to choose which of two VR12 (14) channels are to be utilized. DSX H3 through 11 are loaded with the horizontal coordinates stored in the MB, DSX HA9 and 10 are set to 1 and 0 respectively; hence, all horizontal information will be displayed on the CRT by two dots.

At this time the vertical coordinate is loaded into the DSY V3 through 11 register and V7 through 10 (dwg. *DSC DSY*) are cleared. DSC SET IN 11 sets DSI IN11 to enable a single point to be intensified later. DSC ACTIVE is now set, and the processor continues on to the next FETCH cycle while the Scope Control commences its own timing (dwg. VC12-0-4).

DSC Instruction

The DSC instruction enters the EXECUTE cycle with the instruction appearing in the MB and the IR. When the processor reaches Time State 3 (TS3), the MB contains a character pattern obtained from β-class addressing (dwg. PDP-12-0-16).

As seen on PDP-12-0-17, the processor will go into an Internal Pause mode if the Scope Control is busy. This action is described above.

The same initializing of registers occurs during TS5, with the exception that the MB is loaded into the intensity register rather than the channel and horizontal registers, and DSC ACTIVE is not set. The EXECUTE 2 β-register 1 of the current instruction field (which contains the horizontal coordinate) is addressed. T2 loads the AC9 (which contains the vertical coordinate) with a 30_8 and 14_8 offset, which replaces the last few bits of the given coordinate. This ensures proper vertical spacing between displayed characters, and causes characters to appear on the same horizontal plane. T3 puts the sum of the contents of β-register 1 and an offset into the MB. This is done for the same reason as above, and is described in more detail in Paragraph 5.7.2. The horizontal and vertical offsets are loaded into the AC and MB (dwgs. *PMA* through *PMF*) via the BMSC and TMC gates (dwg. *PMB*). They are enabled with signals RCA DSC SET AC H and SLA DSC CNT MB H. The status of DSC SIZE, which is set with bit 4 of the AC and the ESF instruction, determines whether a full-size (AC04 =1), or half-size offset is to be instituted.

Time State 5 loads the horizontal register (dwg. *DSX*) with the MB (horizontal coordinate) and clears V8 through 11 (dwg. *DSY*) for a half-sized, and V7 through 11 for a full-sized character. These bits will be incremented and decremented by VA9 and 10 during the display interval. The DSC ACTIVE flip-flop is set and the next instruction is fetched.

Display T1

The Initial Point Set-up Delay indicated on drawing VC12-0-4 occurred simultaneously with T5 of the previous EXECUTE (DIS) or EXECUTE 2 (DSC) cycle. It will last 25 μs, as determined by DSC INITIAL DELAY L.

Display T2

Display T2 is a 1.5 μs period in which a point-to-point movement is allowed when doing a DSC. When the PRR Switch (located on M711) is in the SLOW position, the period is increased to 7 μs. However, if bit 11 of the intensity register = 0 (no point to be intensified), this period is reduced to 0.5 μs. Start timing is shown on drawing DSC. The START input to the clock is caused by DSC INITIAL DELAY H and DSC ACTIVE (1) H. SHORT CLOCK 2 speeds up this period when necessary.

Display T3

Display T3 is the intensification interval. The CLK1 OUTPUT becomes high and an intensification pulse occurs at DSC INTEN H if DSI IN 11 (1) H is true. This output will be positive, going negative, if the M711 polarity switch is - (negative). The intensity pulse width is determined by the setting of the WIDTH switch on the M711 Module, 0.5 μs if MIN., 10 μs if MAX.

Display T4

This time interval is a non-adjustable 8 μs period where the *x, y,* and intensification registers are modified before intensifying the next point. CLK 2 (dwg. *DSC*) generates DSC SHIFT H, which shifts the intensification register right one place. Hence, a new bit appears in DSI IN 11. DSX HA9 is complemented by the shift pulse (moving the X coordinate 4 dot places right or left). If DSC SIZE is off (half-size characters), HA10 will be complemented by the shift pulse as well (via DSC HA10 H). As HA9 and HA10 are entered as opposites, the net effect of this operation is to move the dot by only 2 horizontal positions. When HA9 = 0, the point is on the left side of the character in the same horizontal plane as the previous point. Hence, no changes to the vertical (Y) register are necessary (note the error on VC12-0-4, Rev. C and previous issues). When HA9=(1), we are again intensifying a right point and the vertical register must be incremented by 2 (half-size) or 4 (full-size). This action is seen on drawings DSC and DSY. Control returns to Display T2 until all points have been displayed. When the intensification register is completely cleared (all points have been shifted out), the point or character is complete and DSC ACTIVE is cleared, terminating the display. Note that a DIS instruction is a 1-dot case of the DSC instruction in this control logic.

5.7.2 Character Generation

Characters are generated by intensifying points within a two (horizontal) by six (vertical) grid on the scope display face, according to the bit configuration of a selected 12-bit operand. When the operand is obtained, each bit in the intensification shift register controls one of 12 specific locations within the intensification grid (see Figure 5-8).

If bit 11 of the intensity register is 0, the corresponding grid point is not intensified; conversely, if bit 11 is 1, the associated point is intensified. By programming two consecutive character display instructions (DSC), a second two by six grid is displayed to the right of the first, thus producing a four by six overall grid. Through proper selection of operands, the combined grids can now display letters of the alphabet, numbers, and symbols meaningful to users of the PDP-12A (see Figure 5-9).

5.7.3 Addressing

Providing a visual display, whether video or graphic, requires a means of accurately controlling where (on the surface of the displaying media) the display is to take place. In the PDP-12A, this is accomplished by the Display Addressing function.

The PDP-12A Display Control uses the rectangular coordinate method of locating the position on a CRT screen where a particular display will occur. For this reason, the PDP-12A Video Display (VR12), which contains a total

usable display area of 58.5 sq. in. (6.5 x 9 in.), is divided into a grid of 512_{10} x 512_{10} points. The horizontal distance between points is 0.0176 in.; the vertical distance is 0.0127 in. Therefore, the location of any point on the screen can be expressed in terms of the vertical and horizontal point locations. (See Figure 5-8.)

POINT 6 (BIT 6)	POINT 12 (BIT 0)
POINT 5 (BIT 7)	POINT 11 (BIT 1)
POINT 4 (BIT 8)	POINT 10 (BIT 2)
POINT 3 (BIT 9)	POINT 9 (BIT 3)
POINT 2 (BIT 10)	POINT 8 (BIT 4)
POINT 1 (BIT 11)	POINT 7 (BIT 5)

12-0035

Figure 5-8 Bit Positions of DSC Instruction

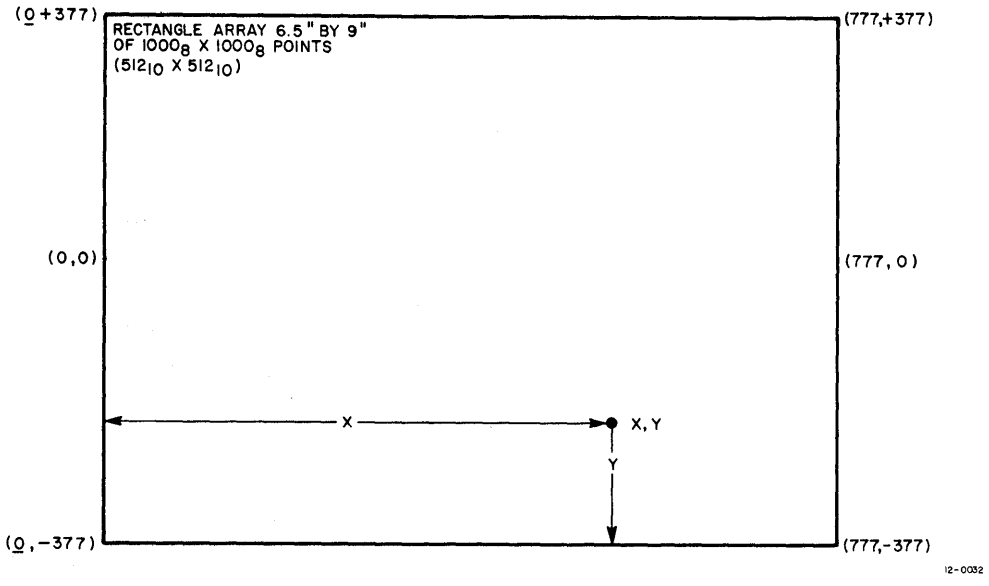


Figure 5-9 Display Point Location

In addressing any single point or group of points (characters), the horizontal (x-axis) and the vertical (y-axis) coordinates must be provided. This is accomplished in the Display Control circuitry by two 9-bit words held in the vertical and horizontal registers, respectively.

Before display occurs, the display location is determined by the addressing function of the Display Control circuitry (Figure 5-10). The circuitry is comprised of two 9-bit registers that supply the horizontal and vertical coordinates. Both of the registers have individual bit (9 each) D/A Converters. These converters establish the proper voltage levels (fully-buffered) that position, on the display CRT or other display surface, the display character or an intensified spot.

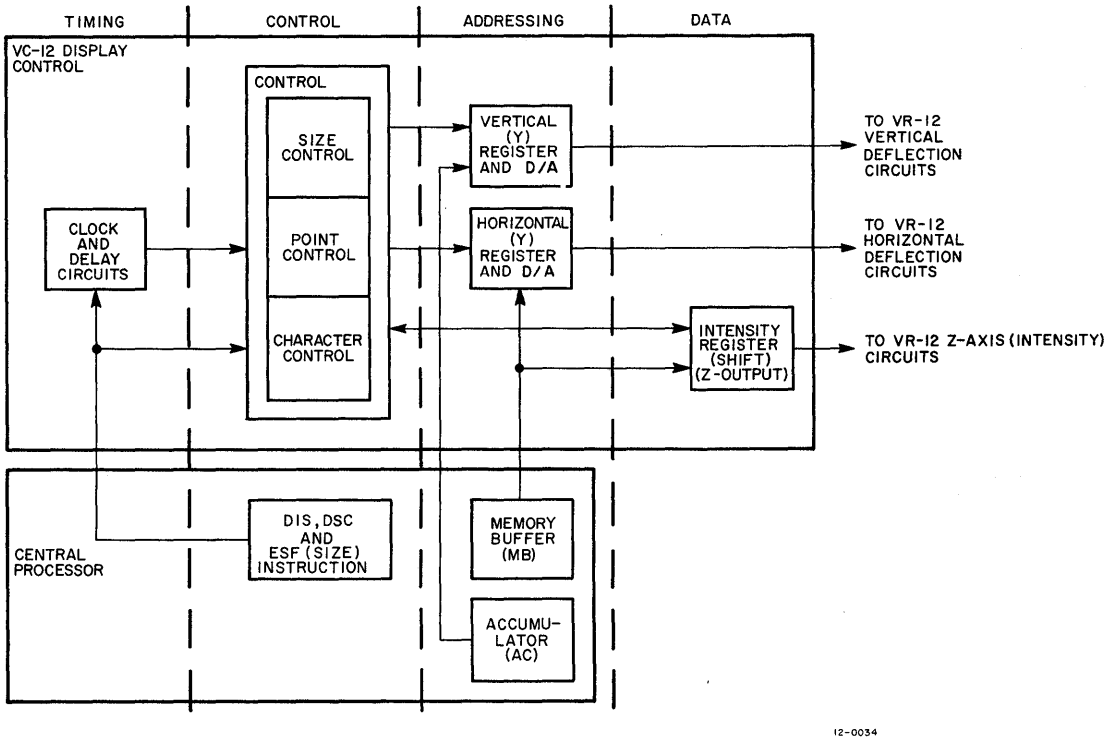


Figure 5-10 Display Addressing Block Diagram

In addition to providing the display address, the vertical and horizontal registers and associated D/A Converters are incremented by the Display Control circuitry so that the intensified points comprising the character matrix can be uniformly separated and displayed in the correct format. This separation results in display characters that are large enough to read, and also provides for two different sizes of displays. The approximate heights of the characters are 0.3-in. and .15-in., respectively.

In addition to point separation (resulting in the two display character sizes), the addressing function allows the control circuitry to increment the two registers in both the vertical axis (y) and horizontal axis (x), so that adequate spacing results between characters and between lines of characters (see Figure 5-11). This automatic spacing is accomplished by adding 10_8 points (full-size) and 4 points (half-size) to the horizontal register and 30_8 points (full-size) and 14_8 points (half-size) to the vertical register; thus the progression of characters and intensified points appear right-reading, with proper line separation.

The DIS instruction belongs to the α instructions. The DSC belongs to the β class instructions. To display a single point, DIS α is used. Display of a single point on the scope with location x = 525 and y = 240 would use the following program:

```
*20
CLR                 /CLEAR AC, LINK, MQ
SET I 5             /PLACE HORIZONTAL
525                 /LOCATION IN 5
LDA I               /PLACE VERTICAL
240                 /LOCATION IN AC
DIS 5               /DISPLAY POINT AT COORDINATES SPECIFIED BY
                    /SET AND LDA INSTRUCTIONS
JMP.-1              /DISPLAY POINT AGAIN
```

When the DSC instruction is used, the vertical coordinate is held in the AC. The horizontal coordinate is held in register 0001.

When a DSC instruction is executed, the following events occur:

a. The display intensification pattern is transferred from *y* to the Display Control intensification buffer.

b. The contents of bits 3 through 6 of the AC are placed in the Display Control *y* buffer: bits 7 through 11 of the AC are set to 30_8 for a full-size character.

c. The contents of register 0001 (the *x*-coordinate) are incremented by 10_8 and transferred to the Display Control *x*-buffer.

Symbols may be displayed using the LEFT and RIGHT SWITCHES with the following program. The LEFT SWITCHES generate the pattern for the left half of the symbol matrice shown in Figure 5-11, and the RIGHT SWITCHES generate the right half.

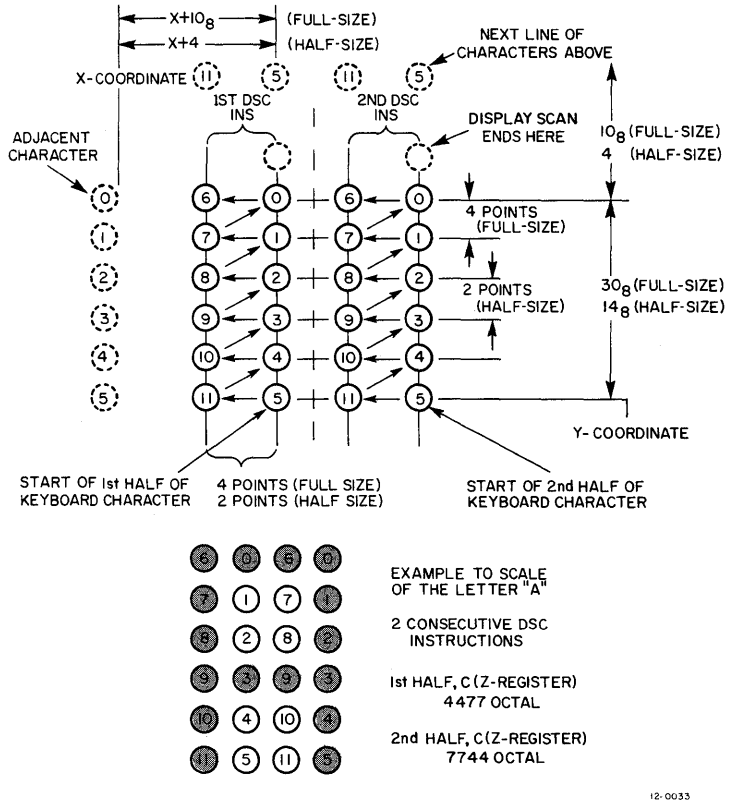


Figure 5-11 Display, Letter A

```
4020/BEGIN
BEGIN,      SET I 3          /POINTER FOR WORD
            0067             /POINTER LOCATION -1
            LSW              /GET LEFT SWITCHES
            STC 70           /STORE IN FIRST ADDRESS
            RSW              /GET RIGHT SWITCHES
            STC 71           /STORE IN SECOND ADDRESS
            DSC I 3          /DISPLAY FIRST HALF ADDRESS + 1 AS
                             /SPECIFIED BY β
            DSC I 3          /DISPLAY SECOND HALF ADDRESS + 1 AS
                             /SPECIFIED BY β
            CLR              /CLR
            STC 1            /RESTORE HORIZONTAL
            JMP BEGIN        /JUMP RESTART
```

Memory location 1 contains the h-bit plus the horizontal coordinate; the AC contains the vertical coordinate.

The following table provides a list of pattern words for character display.

Table 5-2
Pattern Words for Character Display

4477	/A	4577	/E	7741	/I
7744		4145		0041	
5177	/B	4477	/F	4142	/J
2651		4044		4076	
4136	/C	4136	/G	1077	/K
2241		2645		4324	
4177	/D	1077	/H	177	/L
3641		7710		301	
3077	/M	4543	/Z	0523	
7730		6151		500	/APOSTROPHE
3077	/N	4177	/	6	
7706		0000		4163	/
4177	/O	2040		0	
7741		0410		0	/
4477	/P	0	/	6341	
3044		7741		2050	/*
4276	/Q	2000	/	50	
376		2077		404	/+
4477	/R	3410	/BACK ARROW	437	
3146		1010		605	/,
5121	/S	0	/SPACE	0	
4651		0		404	/-
4040	/T	7500	/!	404	
4077		0		1	/.
177	/U	6006	/"	0	
7701		60		601	//
176	/V	3614	/NUMBER SIGN	4030	
7402		1436		4136	/0
677		7721	/DOLLAR SIGN	3641	
7701		4677		2101	/1
1463	/X	1446	/%	177	
6314		6130		4523	/2
770	/Y	5166	/&	2151	

Table 5-2 (Cont)
Pattern Words for Character Display

7007					
4122	/3	5126	/8	1212	/=
2651		2651		1212	
2414	/4	5120	/9	4200	/
477		3651		1024	
5172	/5	4200	/:	2055	/?
651		0		4020	
1506	/6	2601	/;		
4225		0			
4443	/7	2410	/		
6050		0042			

5.7.4 Data Handling

The Data Handling function of the PDP-12A Display Control is concerned with the relationship between the operand (the data character word or display point) and the character or spot displayed. The Data Handling function is comprised of the intensification shift (*z*-axis) register, the clock, gating, and control signals. Control of the Data Handling function is through timing and display. See Figure 5-12.

The *z*-axis register holds the operand, whether it is a full 12-bit character word or simply a point. The action of the *z*-axis register is explained first for a DSC instruction and secondly for a DIS instruction. The purpose of the clock and gating is to sense the state of bit 11 of the *z*-axis, to generate intensification pulses concurrent with a logical 1 state, and to withhold an intensification pulse if the content of bit 11 is in the 0 state. A START pulse is received from display control and timing that enables the display clock, (dwg. *DSC*). The intensification pulse is then routed through a Channel Select switch (one or two or both) to the VR12 Display and/or a remote display.

During the execution of DSC instruction, the *z*-axis intensity register holds the data character word for display. This is accomplished by systematically shifting right and examining the contents of bit 11 of the *z*-axis register. If the current content of the bit location is a logical 1, an intensification pulse is generated for .5 μs; if the current content of bit 11 is a logical 0, no pulse is generated. Next, the register is shifted one place to the right, the content of bit 10 is located in bit 11, bit 11 is sensed again, and the process is repeated until the entire *z*-axis register contains all 0s. This entire operation takes 15 to 51 μs, depending on the number of logical 1s initially deposited in the *z*-axis register.

NOTE

The instruction setup requires either 4.8 or 6.4 μs. The Display Control then allows 25 μs for point coordinate settling, 1 μs for each unintensified point, and 3 μs for each intensified point. When no points remain to intensify, the display process is completed, leaving the vertical and horizontal coordinates of the D/A at the last intensified point of the DSC instruction.

5.8 PDP-12 SPEAKER

The PDP-12 Speaker is mounted on the rear of the A/D (0 through 7) panel; the volume control is mounted on the left front side of this panel. The speaker is included in all system configurations and, although not strictly a LINC device, is included in this section.

The speaker frequency is dependent on the frequency of change of AC00. Each time AC00 changes, an output is generated unconditionally to the speaker.

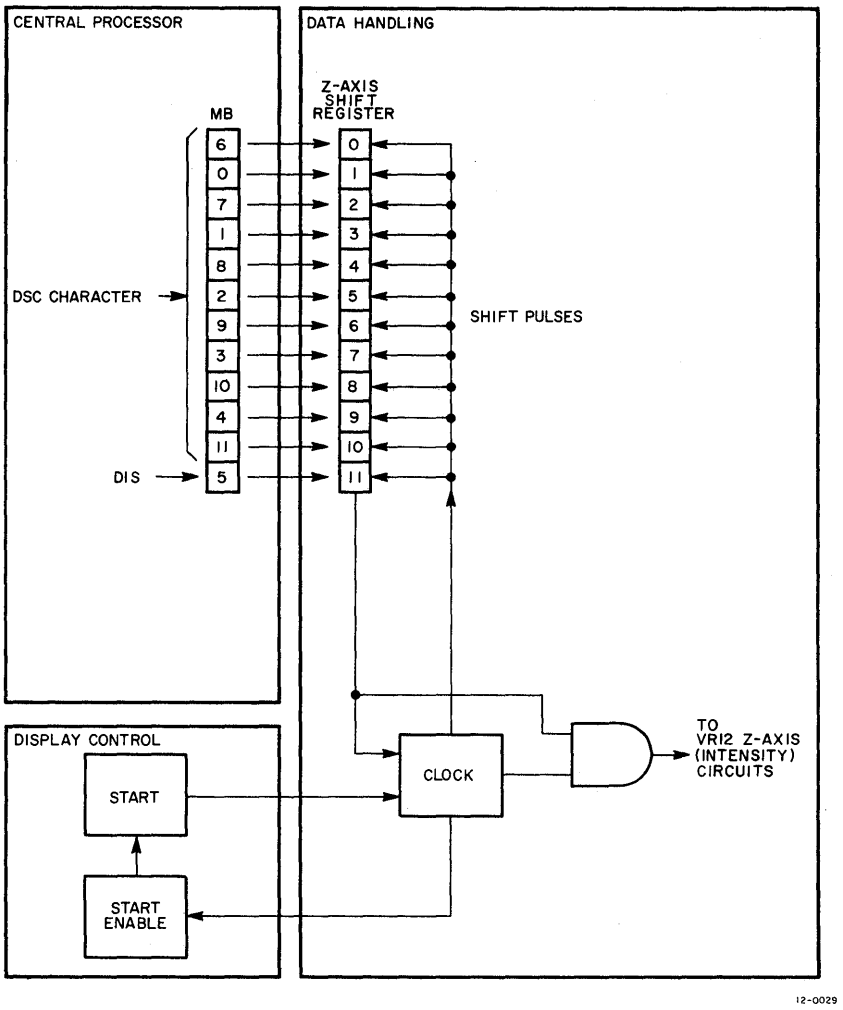


Figure 5-12 Data Handling Block Diagram

A block diagram of the speaker and its control is illustrated in Figure 5-13. The speaker capability can be illustrated by the following program (START 20, 8 Mode).

```
0020    7001    INCREMENT AC
        5020    JUMP START
```

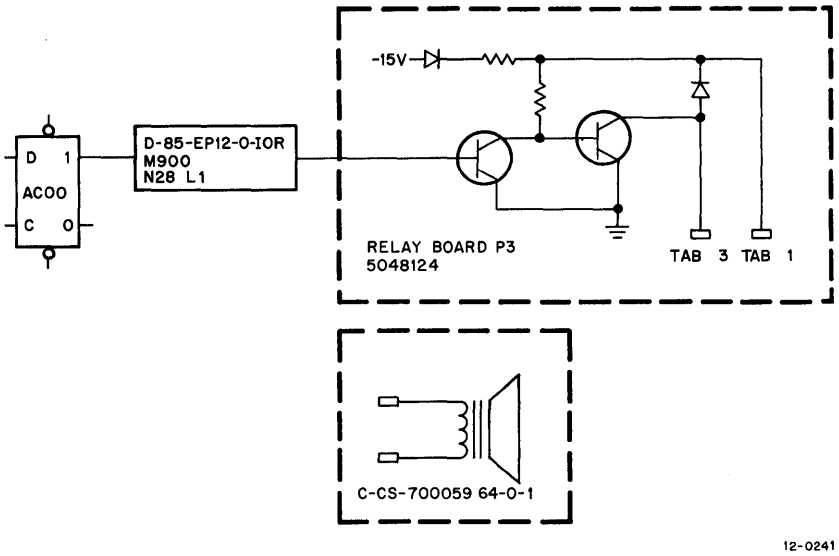


Figure 5-13 Speaker and Control

CHAPTER 6
LINCTAPE CONTROL SYSTEM

6.1 INTRODUCTION

This chapter, together with the referenced documents, provides information on the principles of operation of the Type TC12 LINCtape Control and associated tape transports Types TU55 or TU56. This chapter assumes a prior knowledge and understanding of the PDP-12 Central Processor (CP) and the TU55/TU56 Tape Transports.

6.2 GENERAL DESCRIPTION

The LINCtape System (Figure 6-1) is supplied with the basic PDP-12A and -12B Computer Systems, and serves as an auxiliary magnetic tape data storage facility. The LINCtape System stores information at fixed positions on magnetic tape similar to a magnetic disk or drum storage device, rather than at unknown or variable positions as in conventional magnetic-tape systems. This storage facility provides for replacement of blocks of data on tape in a random fashion, without disturbing previously recorded information. Specifically, during the writing of information on tape, the system reads format (mark) and timing information from the tape, and uses this information to determine the exact position at which to record the written information. The same mark and timing information is used to locate data to be read back from the tape.

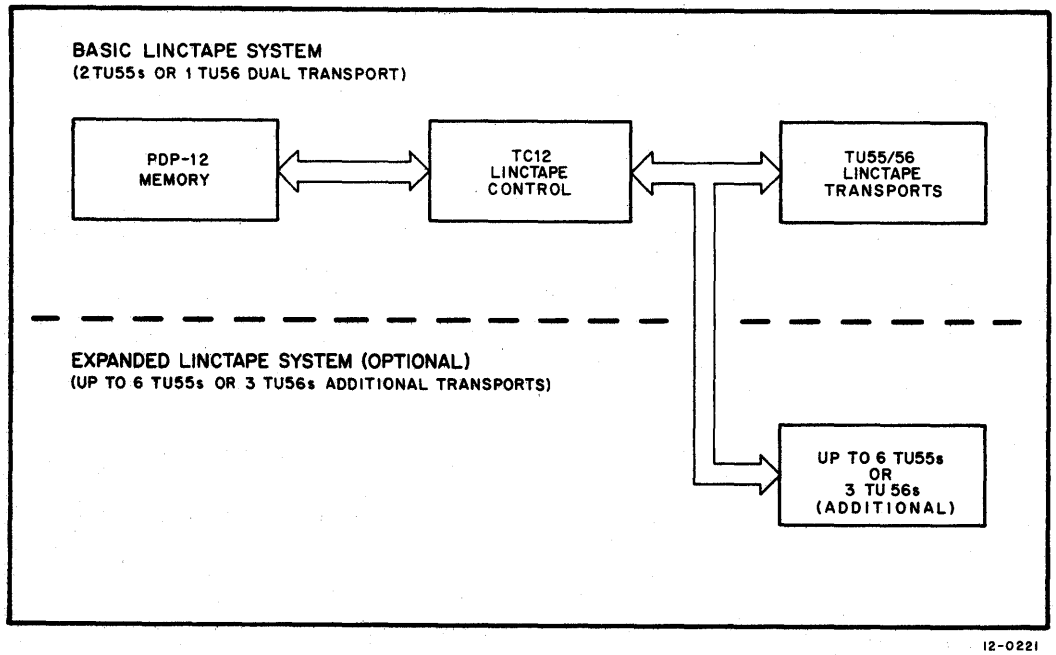


Figure 6-1 LINCtape System Configuration

The PDP-12 LINCtape Control is an independent, fully-buffered tape processor that operates directly into core memory on a cycle-stealing basis, with the CP using the Tape Break (Single-Cycle Data Break) facility. A maximum LINCtape configuration comprises eight TU55 Transports or four TU56 Dual Transports (see Figures 6-1, 6-2, and 6-3) and the TC12 LINCtape Control. Each standard format tape will carry 131,000 words, thereby adding up to an additional one million words of accessible storage to the computer. All LINCtape instructions are directly executable by the tape processor.

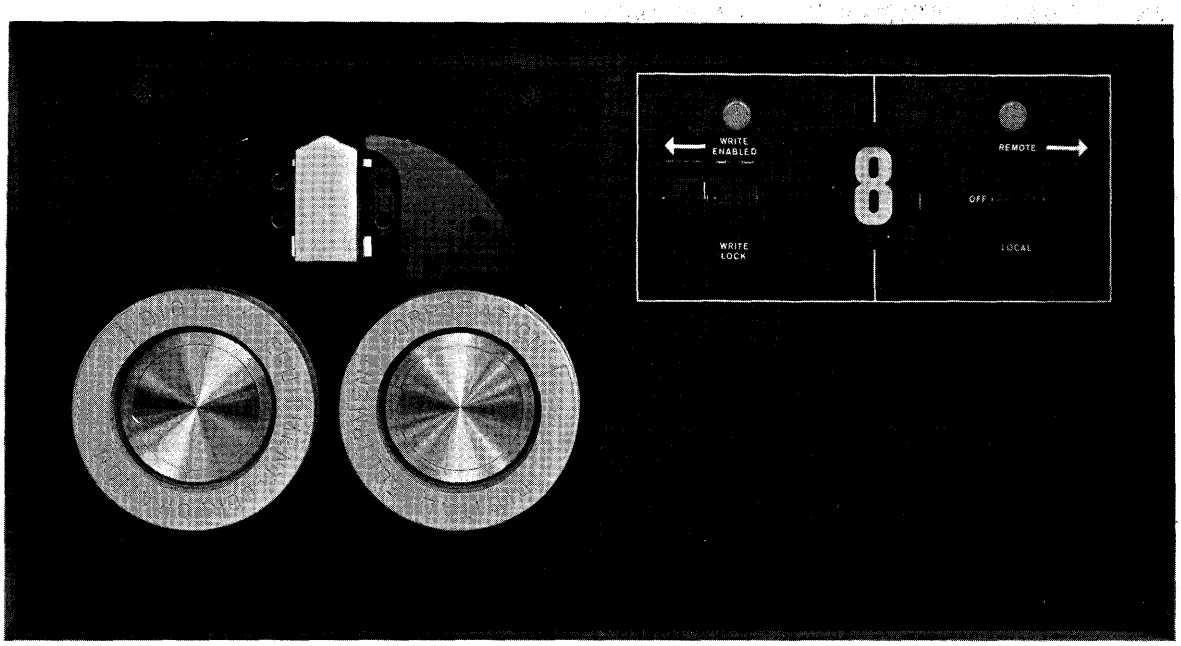


Figure 6-2 TU55 LINCtape Transport

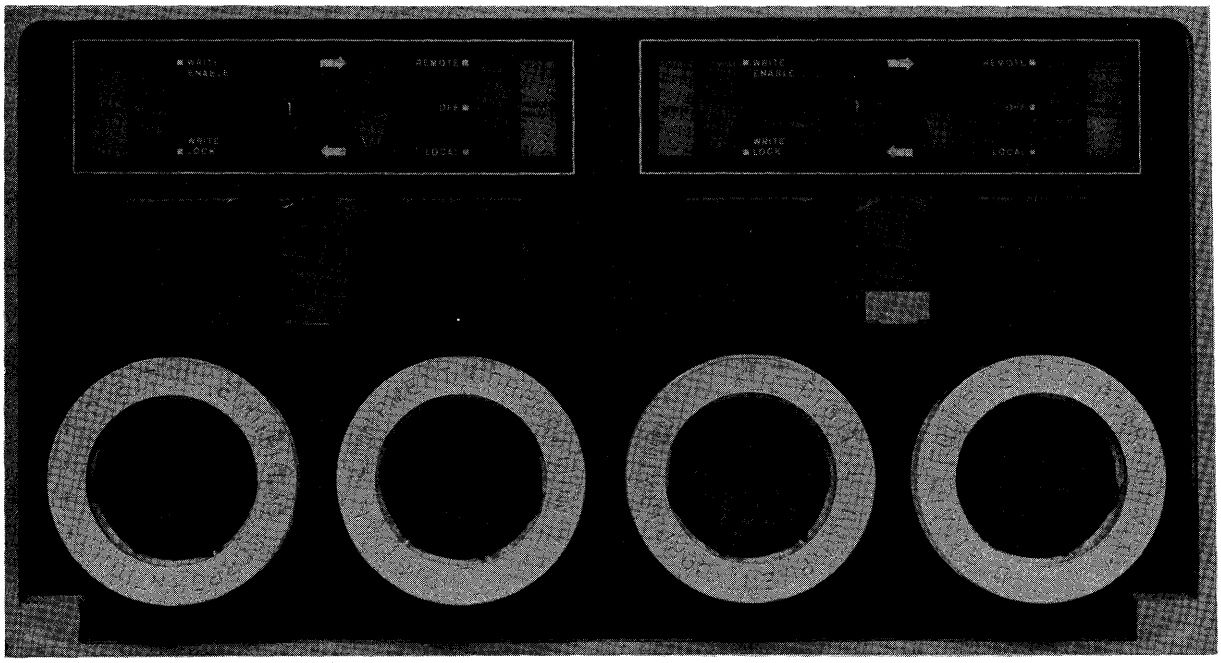


Figure 6-3 TU56 Tape Transport

Normally, the CP waits until a tape operation is complete before continuing. If programmed in the No Pause Mode, the CP can continue with the program as soon as the LINCtape instruction has been interpreted and the operation initiated.

6.3 REFERENCED DOCUMENTS

The Digital Equipment Corporation documents listed in Table 6-1 contain material which supplements the information contained in this chapter. These documents are supplied with each PDP-12 Computer, or may be obtained from the nearest DEC field office, or from the main office:

Digital Equipment Corporation
146 Main Street
Maynard, Massachusetts

Table 6-1
DEC Documents

Doc. No.	Title	Description
DEC-12-SRZA-D	*PDP-12 System Reference Manual*	Programming and operating information for the PDP-12, including brief instructions on the TC12 LINCtape Control. Also a partial LINCtape library system and list of the LINCtape utility routines.
DEC-12-HR2A-D	*PDP-12 Maintenance Manual Vol. II*	Information on installation and maintenance, including the LINCtape Control. Also trouble-shooting and repair and replacement.
DEC-12-HR3A-D	*PDP-12 Maintenance Manual Vol. III*	A complete collection of the PDP-12 Engineering Drawings.
DEC-12-HR4A-D	*PDP-12 Maintenance Manual Vol. IV*	A complete collection of all module schematics.
H-TU55	*Type TU55 Tape Transport Maintenance Manual*	Transport drive logic and internal operations, plus preventive and corrective maintenance instructions.
H-TU56	*Type TU56 Tape Transport Maintenance Manual*	Same as the TU55.

6.4 SPECIFICATIONS

A summary of the characteristics of the LINCtape Control and associated equipment is listed in Table 6-2.

Table 6-2
Summary of Equipment Specifications for the LINCtape Control

Tape Characteristics and Density

a. Tape Density: Approximately 420 bits/in.

b. Each word is assembled in approximately 120 μs

c. Tape speed is approximately 80 in. per second

d. Data block transfer rate is approximately 30 ms

NOTE

Transfer rates vary by 20% as the effective reel diameter changes.

Table 6-2 (Cont)
Summary of Equipment Specifications for the LINCtape Control

Addressing		
a. Mark and timing tracks allow searching for a particular block.		
Time		
a. Start time is 150 ms ± 15 ms; stop time is 150 ms ± 15 ms.		
b. Turn Around time is 200 ms ± 50 ms.		
Input Signals to Transport from Control		
Commands		
	FORWARD REVERSE	Normally complementary levels
	GO STOP	Normally complementary levels
	ALL HALT	(Stop transport)
Unit Select		Select unit 0 through 7
Information		Analog Write signals to the tape heads
Output Signal from Transport to Control		
Control	WRITE ENABLE	(Ground level assertion)
Information		Analog Read Signals from the tape heads
Environmental Conditions		
Thermal Dissipation		2150 BTU/HR
Operating Temperature		50° – 95°F ambient
Humidity		10% – 90% relative humidity

NOTE

The magnetic tape manufacturer recommends 40% – 60% relative humidity and 60° – 80°F, as an acceptable operating environment for LINCtape.

6.5 LINCTAPE FORMAT

6.5.1 Track Arrangement

LINCtape uses a 5-channel format. To reduce bit dropout and minimize the effect of skew, each channel is redundantly recorded on two nonadjacent tracks of the 10-track LINCtape (see Figure 6-4). The five LINCtape channels include: a timing channel (simultaneously recorded on tracks 1 and 10); a mark channel (tracks 2 and 9); data channel 1 (tracks 3 and 6); data channel 2 (tracks 4 and 7); and data channel 3 (tracks 5 and 8).

Information is transferred between the tape and the LINCtape Control through a 10-track Read/Write head. Series connection of corresponding track heads within each channel and the use of Manchester phase-recording techniques, rather than amplitude-sensing, virtually eliminate dropouts.

6.5.2 Word Assembly and Disassembly

When reading, the tape Read/Write Buffer (RWB) register assembles 12-bit computer length words from four consecutive 3-bit lines read from the data channel tracks (DC1-DC3) of tape (see Figure 6-5). At the same time the mark track is decoded to specify whether the current 12-bit word represents a data word, a block number, or

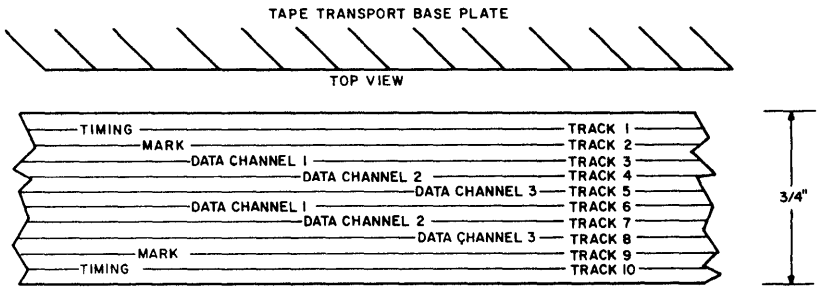


Figure 6-4 Track Allocation Showing Redundantly Paired Tracks

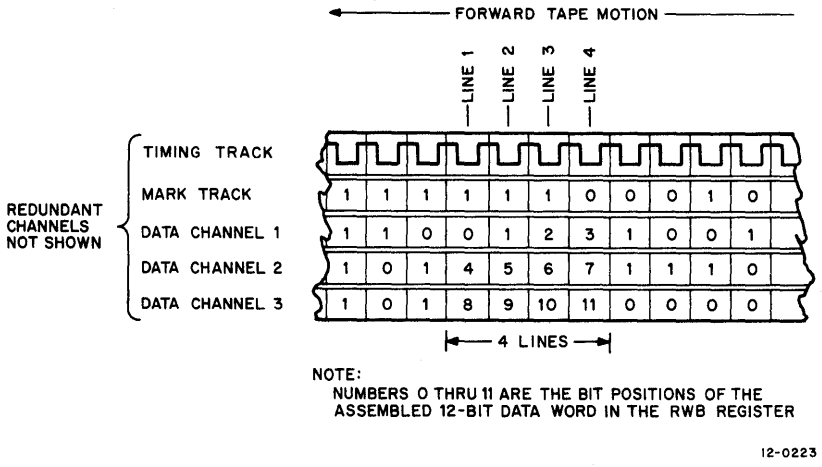


Figure 6-5 Structure of 12-Bit Data Word (4 Tape Lines)

tape control information. The LINCtape Control uses timing signals from the timing track to synchronize the reading of the mark track, and reading from, or writing on, the data channels. During data writing, the LINCtape Control disassembles 12-bit words and distributes the bits in such a manner that they are recorded on four successive data channel lines (see Figure 6-5). A checksum* is recorded after the data (at the end of the block) and is used during subsequent checking operations to verify the data.

*Checksum – See Paragraph 6.6.4.

6.5.3 Overall LINCtape Format

Figure 6-6 shows the overall format for a standard reel of LINCtape. Each reel of standard LINC-formatted tape contains the following primary areas:

a. Leader of blank tape

b. Front end zone

c. Data area

d. Interblock zones

e. End zone

f. Trailer of blank tape

The tape leader and trailer are nothing more than approximately five ft of blank tape that is provided to protect the data storage areas of tape while installing, threading, or removing tapes on the transport hubs. The end zones, both front and back, provide additional protection to the data areas, and, in standard computer-programmed tape operations, the detection of an end zone either reverses the tape motion, or brings both reels to a complete stop without pulling the tape off the trailing reel. The front interblock zone and the negatively numbered front blocks provide block separation and buffering for Tape Acceleration, Turnaround, and Search operations.

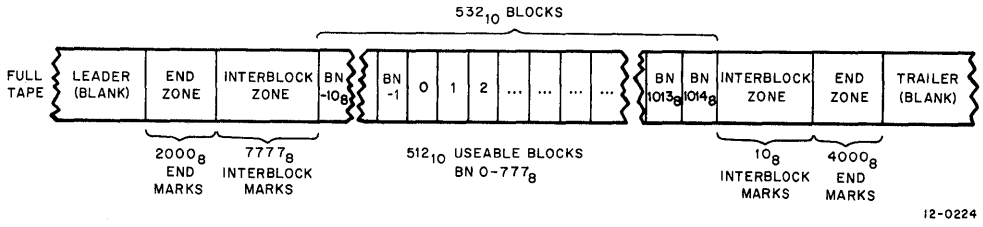


Figure 6-6 Standard LINCtape Format

The beginning of a block of tape data is indicated by the forward block mark. When the mark decoder senses a block code on the mark track, the Read/Write Buffer register contains the number of that particular block. Blocks -10_8 through -1 and blocks 1000_8 through 1014_8 on a standard LINCtape provide a turnaround area and are not available to the programmer for data assignment, although the format is identical to that of block 000 through 777_8. These additional blocks allow smooth searching and turnaround when accessing the blocks at the beginning and the end of the tape. Following the last data block is another end zone, and, as mentioned above, this segment provides protection to the data areas and prevents running beyond the formatted area of the tape.

LINCtapes are subdivided permanently into blocks by the initial marking process (MARK 12), or formatting which records a fixed pattern on the tape. This pattern includes fixed block addresses that permit addressing the information stored on tape by means of a block number. Information is transferred and checked in units of complete blocks that are specified by their block addresses. Each block includes a checksum for verifying the integrity of information transfers with the tape.

Standard LINCtape contains 1000_8 (512_{10}) addressable blocks numbered consecutively from 000 through 777_8. Each block on a standard LINCtape contains 400_8 (256_{10}) data words corresponding to the contents of one-quarter of a LINC Mode core memory field (1024_{10}-word field of memory in LINC Mode). See Figure 6-7. The first core memory address involved in the data transfer is the first address within the specified memory quarter (see Figure 6-8). Quarter 0 begins with address 0, quarter 1 with address 400_8, and so on, up to quarter 7, which begins with address 3400_8.

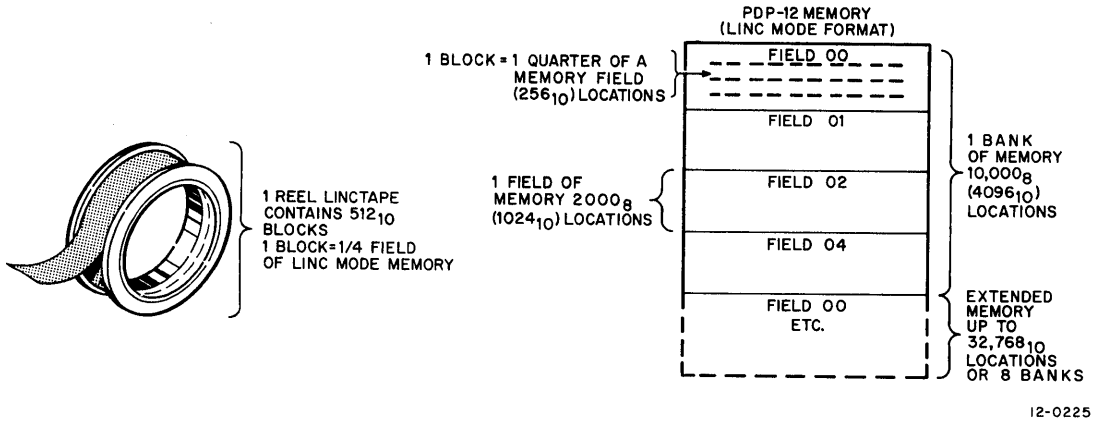


Figure 6-7 LINCtape Reel to Memory Comparison

6.5.4 Non-Standard LINCtape

The length of blocks as well as the format of the block addresses can be varied by using a non-standard marking program. Once a given tape is marked, however, the format is fixed for that tape unless it is completely erased and re-marked.

6.5.5 Detailed LINCtape Block Format

Figure 6-8 illustrates a detailed single LINCtape block and a portion of the adjacent areas. Each area of the block, along with its associated mark track code, is shown. Starting at the left is the forward block number area

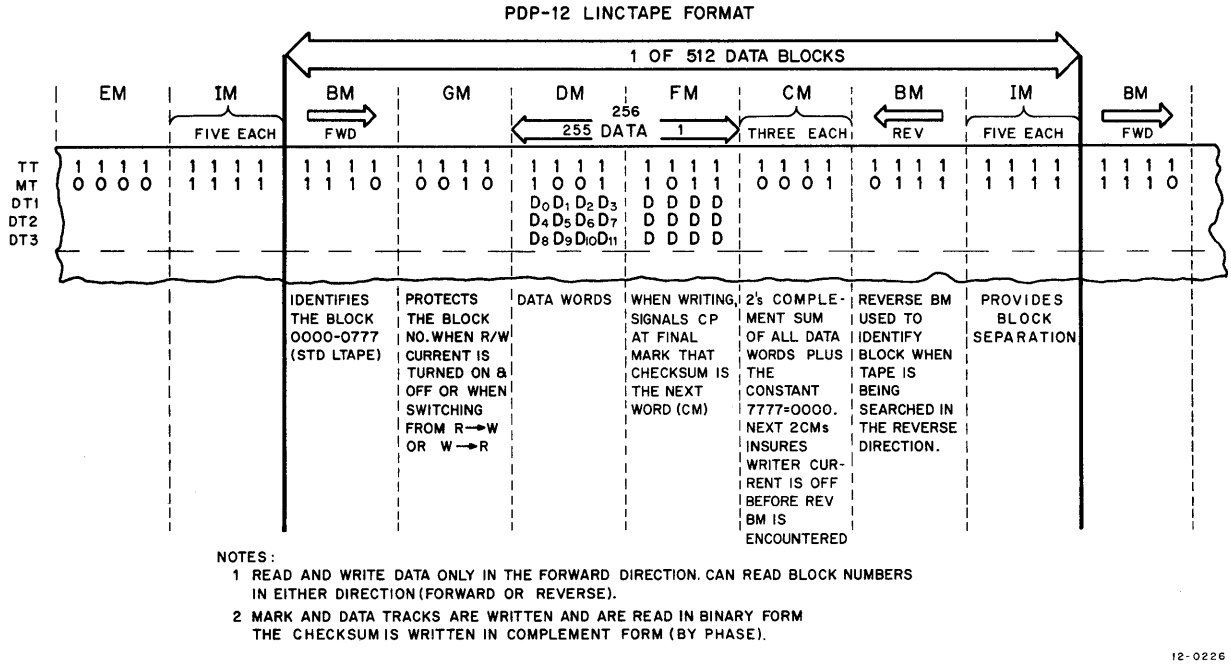


Figure 6-8 Standard LINCtape Detailed Format

BM (*B*lock *M*ark), which is identified by its unique mark track code. Recorded on the three data channels (disregarding the redundant tracks) is the block number, a 12-bit word that identifies this particular block. Note that every word on the tape occupies four lines or columns.

NOTE

The recording technique and arrangement of the Read/Write Buffer register is explained in Paragraph 6.6.3.

Continuing from left to right, the next area is the guard word, GM (*G*uard *M*ark). The guard word protects the adjacent block number area from transients when the Read/Write current is turned on and off, and allows time for the tape processor to switch Read, Write, or Search Modes.

Following the guard word is the data word abbreviated DM and FM (*D*ata *M*ark and *F*inal *M*ark, respectively). The data word is the information recorded on tape from core memory. The final data word (FM) is specially identified to signal the mark track decoder that this is the end of the current data block.

The final data word is followed by three check words, CM (*C*heck *M*ark). The first checksum word contains the checksum (recorded complement of the sum of data words). The CM also guarantees that the data writers will be turned off before the reverse block number area is encountered by the tape heads. The next two Check Mark areas, which contain the mark track identification for checksum words, provide an additional buffer area to protect the reverse block number (RBM) which follows the third checksum word.

The reverse block number, RBM (*R*everse *B*lock *M*ark) is used to identify the block when the tape is being searched in the reverse direction.

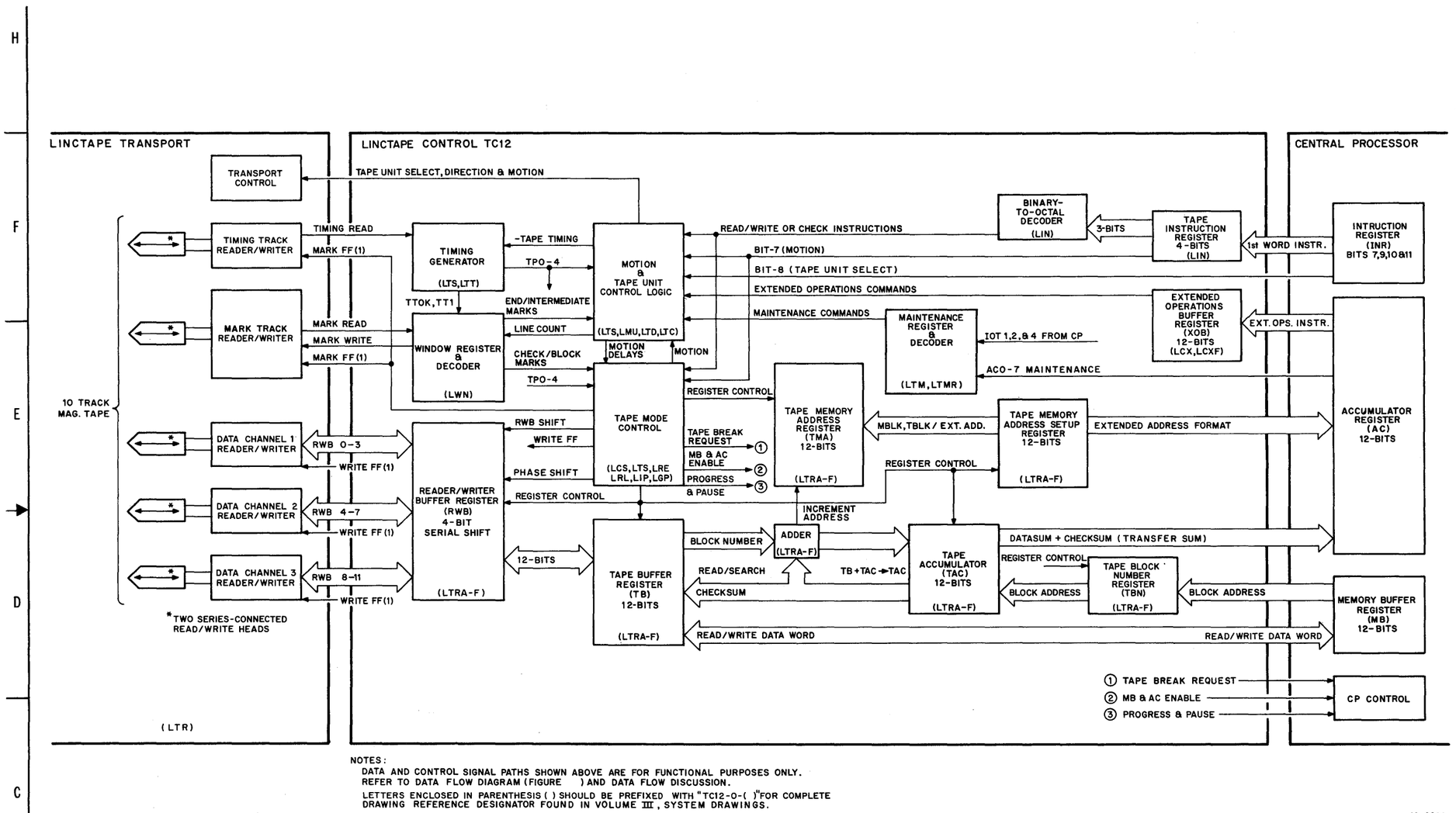
The next five words comprise the interblock zone IM (*I*nterblock *M*ark). When an IM has been decoded by the mark track decoder, and, if the IBZ instruction has been programmed, the tape processor signals the CP and the main program can be interrupted to allow the CP to skip to a tape service routine.

6.5.6 LINCtape Standard Format Summary

Table 6-3 presents a summary of the standard LINCtape format; listed are the various zones of LINCtape, the number of words in each zone, and the function of each zone.

Table 6-3
LINCtape Format Summary

Mark	Number of Words	Function
EM	1024	Front End Zone
IM	4096	First Intermediate Zone
BM)	1	Forward Block Zone Number
GM	1	Guard Word
DM	255	First 255 Data Words
FM } per block	1	Final Data Word
CM	3	Check Marks
(R) BM	1	Backward Block Number
IM)	5	Second Intermediate Zone
EM	2048	Back End Zone



NOTES:
DATA AND CONTROL SIGNAL PATHS SHOWN ABOVE ARE FOR FUNCTIONAL PURPOSES ONLY. REFER TO DATA FLOW DIAGRAM (FIGURE) AND DATA FLOW DISCUSSION.
LETTERS ENCLOSED IN PARENTHESIS ( ) SHOULD BE PREFIXED WITH "TC12-0-( )" FOR COMPLETE DRAWING REFERENCE DESIGNATOR FOUND IN VOLUME III, SYSTEM DRAWINGS.

12-0227

Figure 6-9 LINCtape Overall Block Diagram

Table 6-4 provides a summary of LINCtape channel (tape) assignments.

Table 6-4
Mark Track and Channel Assignment

Mark Track Codes		
EM	0000	End Mark
IM	1111	Interblock Mark
BM	1110	Forward Block Mark
(R) BM	0111	Reverse Block Mark
GM	0010	Guard Mark
DM	1001	Data Mark
FM	1011	Final Mark
CM	0001	Check Mark
Channel Assignment		
TT	1111	Timing Track
MT	----	Mark Track (see code above)
DT1	DDDD	Data Track 1
DT2	DDDD	Data Track 2
DT3	DDDD	Data Track 3
----	----	----------
DT6	DDDD	Data Track 6
D = Data bits (4 lines)		

6.6 BLOCK DIAGRAM DISCUSSION

Figure 6-9 shows an overall functional block diagram of the LINCtape Control. Data and control signal paths have been simplified for the purpose of this discussion. A source-route-destination discussion of data flow is presented in Paragraph 6.6.4.

Timing pulses for tape operations are generated from signals provided by the timing tracks; these signals are read by the timing track tape head (actually 2 heads in series). The Read/Write outputs are used to generate tape pulses TT0 through TT4, which synchronize and control all LINCtape operations once the CP has initialized the tape processor, and proper tape speed is achieved.

Mark track data is used to locate specific areas on the tape. These marks identify block zones, intermediate zones, check zones, and beginning- and end-of-tape areas. Mark track decoding is accomplished by the Window register and mark track decoder. Outputs of the decoder go to the motion and tape unit control and the tape mode (Major State) control logic. The motion and tape unit control logic selects a particular tape transport and determines the tape direction for the selected tape transport. The tape instruction, a two-word instruction from the CP, is deposited in the tape instruction register (first word) and the TBN register (second word). A conventional binary-to-octal decoder determines the specific tape operation to be performed, i.e., a single block read (RDE) or perhaps a group instruction: Write and Check Group (WCG).

NOTE
Extended Tape operations and Extended Addressing Tape operations are discussed in Paragraph 6.6.1.

Once the CP has supplied the two-word tape instruction, the tape processor assumes complete control of the tape operation and generates all necessary command, control, and timing signals to complete the tape operation.

Data bits are read from and written onto three data channels of tape (disregarding the redundant tracks). The data are assembled (during a Read) and disassembled (during a Write) in the Read/Write Buffer register (RWB). One line of tape is read by the data channel tape heads to produce three corresponding bits in the RWB register (see Figure 6-10). The data in the RWB are then shifted left one place, and three more bits are read from the tape. After four Read-And-Shift operations, the RWB register contains a complete 12-bit word, read from four lines of tape. Data is also written on tape three bits at a time from the RWB register in a similar (though reversed) manner as in the Read operation. The RWB register sends tape information to the tape buffer (TB) during a Read operation and receives the 12-bit data word from the TB during a Write instruction.

6.6.1 Extended Operations

The Extended Operations facility allows transmission of data between tape and any program-defined area of memory. Extended Operations are controlled by the contents of the Extended Operations Buffer (XOB), which in turn must be loaded from the CP accumulator prior to giving the first word of the tape instruction to the tape processor. The miscellaneous instruction AXO loads the XOB from the Central Processor AC.

The specific operations that can be performed from the Extended Operations facility are:

a. Extended Memory Addressing
b. Mark Condition
c. Enable Tape Interrupt
d. Maintenance Mode
e. Enable Extended Address Mode
f. Do Not Pause During Execution
g. Hold Unit Motion
h. Extended Units

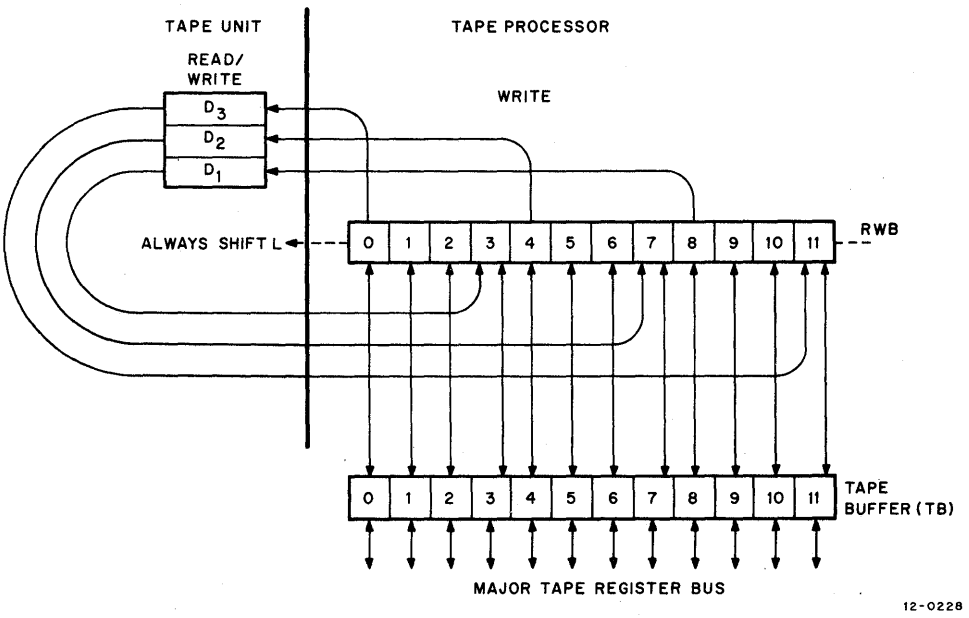


Figure 6-10 Read/Write Buffer Interface Diagram

Unlike the other data or control registers, the XOB is not located on the register bus modules, but on various modules throughout the tape processor. The following paragraphs discuss the functions of the XOB register with the corresponding AC bits (see Figure 6-11).

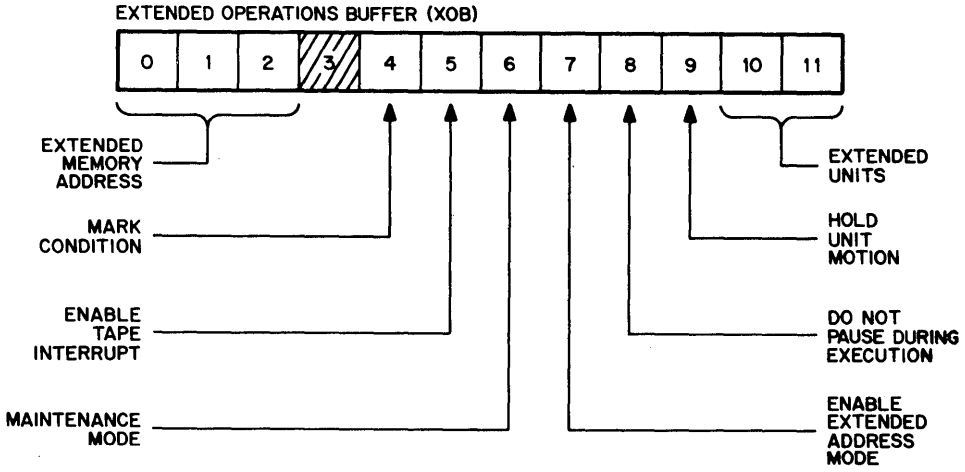


Figure 6-11 Extended Operations Buffer Bit Assignments

Extended Memory Addressing (XOB 0-2) – The three most significant bits, 0 through 12, are the Extended Memory Address bits. See the TC12-0-LCXF Block Schematic discussion. Central Processor AC bits 0 through 2 are loaded into tape field flip-flops 0 through 2, thus designating the 4K memory field to be addressed.

NOTE
Bit XOB 3 is not used.

Mark Condition (XOB 4) – The MARK flip-flop (shown in drawing TC12-0-LCX) is used in conjunction with the MARK switch on the operator's console to allow the MARK 12 program to format virgin or previously marked tape. The MARK flip-flop can be set only when MARK 12 program is running and the MARK switch is depressed by the operator, thereby minimizing the possibility of accidentally destroying a tape format by enabling the MARK flip-flop.

Enable Tape interrupt (XOB 5) – When this flip-flop is set (TC12-0-LCX), a Program Interrupt will occur whenever the Interrupt is enabled and the Tape Done flag is set.

Maintenance Mode (XOB 6) – When this flip-flop (TC12-0-LTM) is set, all timing signals and data are prevented from entering the tape control registers from the tape unit Read/Write circuits. Instead, IOT instructions are used as input to the tape control to simulate the functions of the tape head and processor. In addition to the IOT functions shown on drawing TC12-0-LTMR, Appendix F of the *System Reference Manual* contains a listing of all the tape maintenance functions.

Enable Extended Address Mode (XOB 7) – When this flip-flop is set (TC12-0-LCX), the limitation of block-to-memory quarter transfers no longer applies. The transfer is executed as follows:

1. The contents of the TMA Setup register are placed in the TMA.
2. The second word of the tape instruction is taken as an 11-bit block number and placed in the TBN.
3. The transfer is effected between tape and designated area of the memory field specified by bits 0 through 2 of the XOB discussed above.

Do Not Pause During Execution (XOB 8) – When this flip-flop is enabled (TC12-0-LCX), the CP continues with the main program after the tape instruction is initiated.

Hold Unit Motion (XOB 9) – This flip-flop (TC12-0-LCX) when set, keeps the tape unit in motion after the completion of the instruction, even though the unit is deselected (logically).

NOTE

With the Hold Unit Motion and the Do Not Pause During Execution flip-flops set, it is impossible to do two back-to-back tape instructions and enable both the 0 and 1 units for simultaneous motion.

Extended Units (XOB 10-11) – The preunit flip-flops (TC12-0-LMU) select one of up to six additional transports (units 2 through 7) which may be part of the tape system. Refer to Page 3-47, Paragraph 3.6.7 of the *System Reference Manual.*

6.6.2 Extended Addressing

Extended Address format is used for tape operations with nonstandard formatted tape. The standard LINCtape format, as described in Paragraph 6.5.3, is not used. To address more than 1000_8 tape blocks, the second tape instruction word requires more than the nine bits allotted in standard LINCtape format. In Extended Addressing, therefore, the second word is allotted 11 bits for addressing the desired data blocks. Group instructions (transfers of more than one data block per tape operation) cannot be performed in this mode. In the Extended Addressing mode of operation, the number of data words in a block can vary (400_8 words in a standard LINCtape block).

Prior to issuing the tape instruction, the first Memory Address in the data transfer is loaded from the AC and placed into the TMA Setup register, using the instruction, TMA. The second word of the tape instruction is taken as an 11-bit block number (bits 1–11), and placed in the TBN. The transfer is effected between tape and the designated address of the 4096-word memory field which is specified by bits 0 through 2 of the Extended Operation Buffer (XOB). The transfer is thus independent of the LINC Memory Field Assignments. The CP may pause or not pause during an Extended Addressing operation, depending on the state of bit 8 of the XOB.

NOTE

A standard MARK program is available for 129 words per block, and up to 2000_8 blocks per reel of tape.

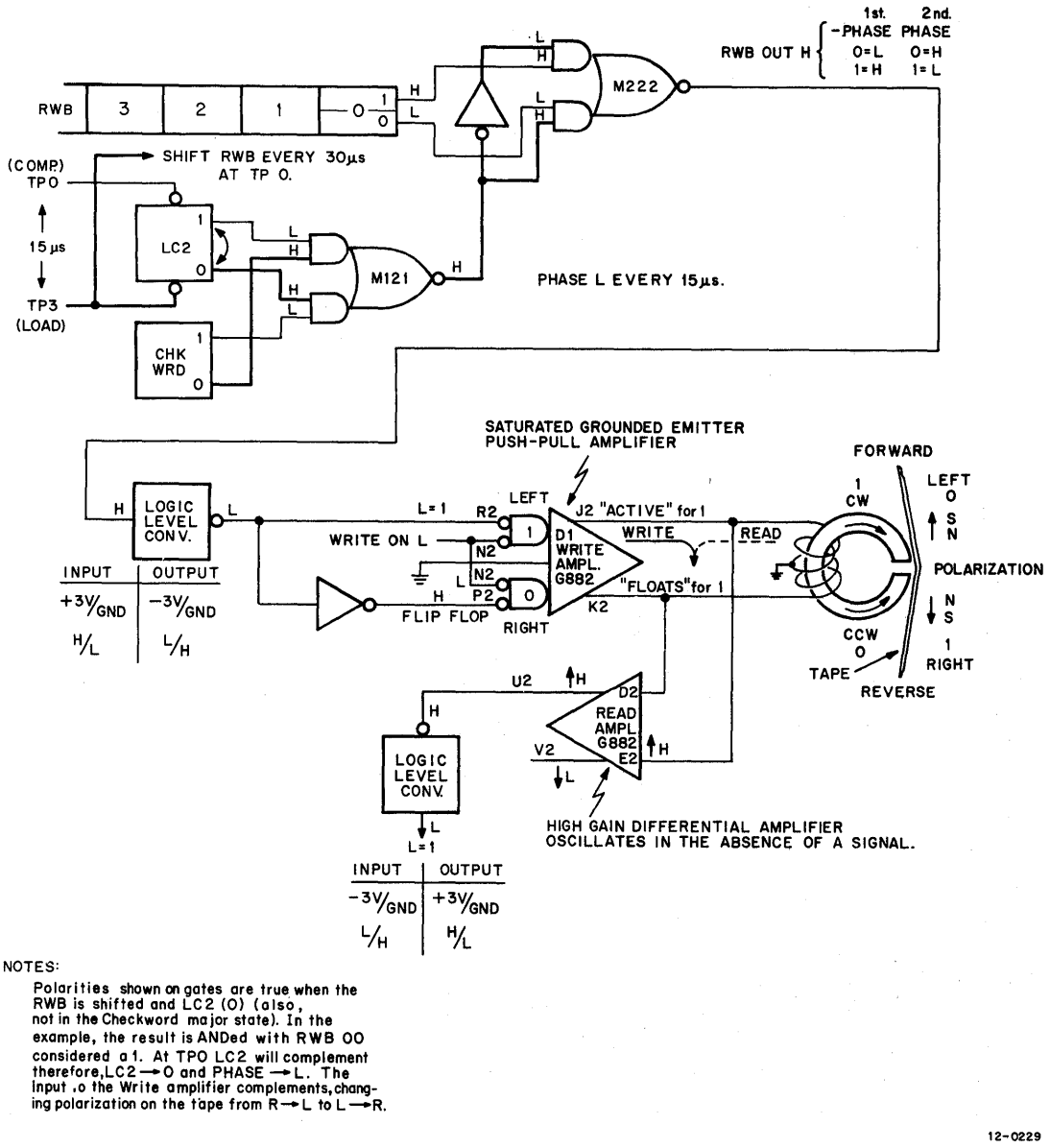
The address for an Extended Address operation is provided before the tape instruction is given. A miscellaneous class instruction is given, loading the CP accumulator, which in turn is loaded into the TMA Setup register with the 11-bit address. The tape instruction is then given and the operation is performed.

As in all Extended Memory operations, whether with tape or some other mass storage device, data transfers will not cross 4096_{10} memory bank boundaries; address 7777_8 is followed by address 0000.

6.6.3 Basic Read/Write Discussion

The functional diagram of the Read/Write logic for the LINCtape Control is shown in Figure 6-12. Each channel of the Read/Write circuitry contains a logic level converter and input gates, a Write amplifier governed by the flip-flop outputs, and a Read amplifier. Read inputs are paralleled with the Write amplifier outputs across the head, allowing the Read amplifier to respond to signals from both the tape head and Write amplifier.

The Read amplifier is a high-gain differential amplifier augmented by a transient positive feedback. When a signal of either polarity is sensed by the head, the Read-amplifier outputs switch immediately and are asserted, thus preventing *head cross-talk* (simultaneously writing the data channels, while reading the timing and mark tracks). The Read amplifier outputs U2 and V2 are standard DEC logic levels of -3V and ground (negative logic).



NOTES:

Polarities shown on gates are true when the RWB is shifted and LC2 (0) (also, not in the Checkword major state). In the example, the result is ANDed with RWB 00 considered a 1. At TP0 LC2 will complement therefore, LC2 → 0 and PHASE → L. The input to the Write amplifier complements, changing polarization on the tape from R → L to L → R.

12-0229

Figure 6-12 Read/Write Simplified Logic

When input E2 is more positive than D2, the output V2 is asserted at ground and U2 is negative; when D2 is more positive, the output levels are reversed. Due to the positive feedback, the Read amplifier oscillates in the absence of changing input signals. The Read amplifier output waveforms are therefore rectangular whenever the differential input signal is indeterminate (see Figure 6-13).

The Write amplifier is a saturated grounded-emitter push-pull amplifier, with its outputs resistive-coupled through pins J2 and K2. If enable level (pin R2) is asserted negative, the Write amplifier is governed entirely by the state of the flip-flop. When the flip-flop is 1, K2 floats while J2 is returned through the resistance and saturated output collector to -13V. When the flip-flop is 0, J2 floats while K2 is negative. In the two tracks corresponding to each channel on tape, information is recorded in a manner that makes Read signals from the two head inductors reinforce on playback.

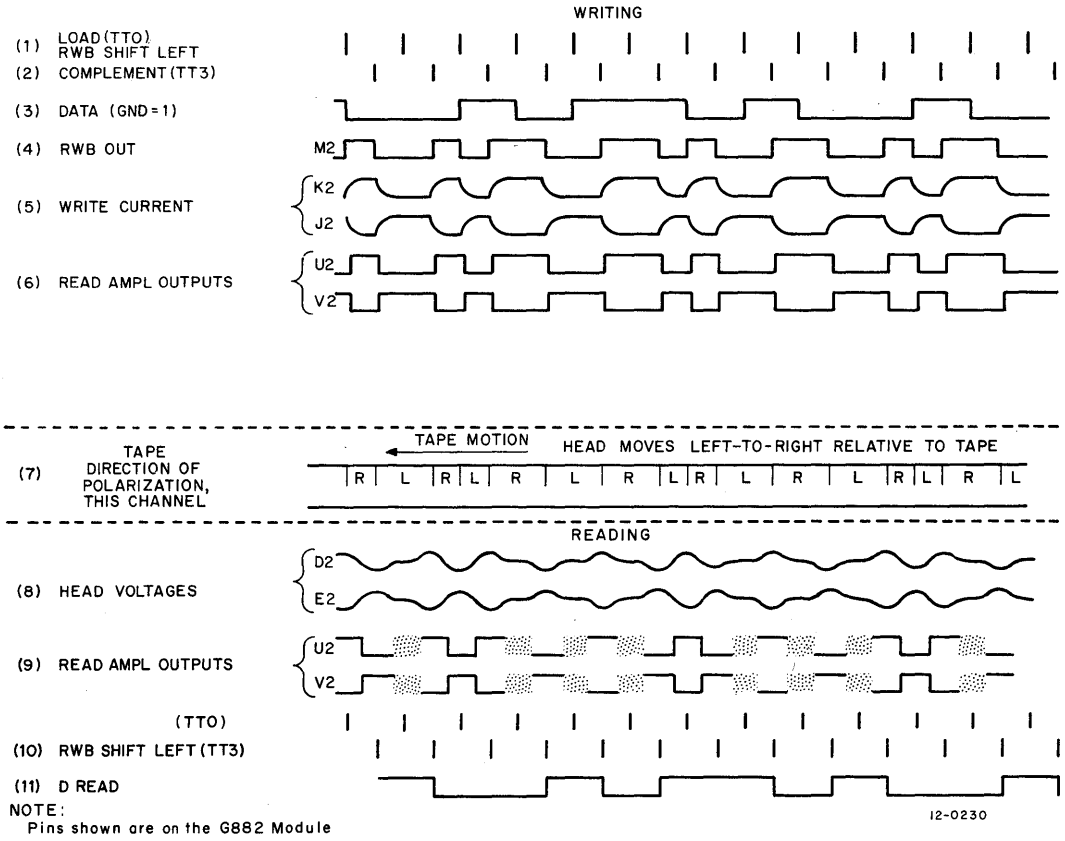


Figure 6-13 Read/Write Logic and Waveforms

The two inductors can be considered as a single-head inductor, the winding of which is center-tapped to ground, and which reads and writes in a single track.

When the Write flip-flop contains 0, current flows from ground through the head inductor into K2, and the polarization of the head core is oriented clockwise. The tape polarization, as the tape moves across the head, is oriented down (left-to-right), regardless of the direction of tape motion. Similarly, when the flip-flop contains a 1, tape polarization is oriented down regardless of the direction of tape motion. When reading, the current induced in the head by a change in polarization flows opposite to the current required to cause the same change; consequently, the current induced by a left-to-right (L-R) tape-polarization change is a current flowing out of the head toward pin E2. The head is a source; therefore, when a terminal is a current source, it is positive. Thus a L-R tape-polarization change causes the Read amplifier input E2 to be positive; consequently, V2 is ground and U2 is negative. In like manner, the right-to-left (R-L) polarization change induces a positive signal at D2 and results in V2 being asserted negative and U2 at ground.

The Manchester phase-recording technique used in the LINCtape Control requires two pulses to write each bit on tape: a Load pulse which is a TB → RWB or a LRL SHIFT RWB and occurs at TT0, and a Complement pulse which is LTS PHASE H and occurs at TT3. Therefore, for the purpose of this discussion, the two pulses will be referred to simply as the Load pulse and the Complement pulse respectively.

The Load pulse loads the Write flip-flop with the value of the bit to be written (Read/Write Buffer register bits 0, 4, and 8, which are written simultaneously). Depending on the state of the Write flip-flop (see Figure 6-12), the Load pulse may or may not cause a magnetic polarization change to take place on the tape. The Complement

pulse however, causes a tape magnetic polarization change. It is this loading (RWB shift) and subsequent complementing (PHASE), occurring alternately at 15-μs intervals, that cause full magnetic polarization changes on the tape.

When reading, the logical value of a recorded bit is detected by sensing the tape head inductor output as the polarization change passes over the tape head inductor. The output level is clocked into RWB-register bits 3, 7, and 11 with the RWB shift pulse. If the logical value was a 1, a right-to-left (R-L) tape magnetic polarization took place. If the logical value was a 0, a left-to-right (L-R) polarization took place.

As shown in Figure 6-13, the Load and Complement pulses alternate. This relationship is shown in lines 1 and 2 and occur at approximately 15-μs intervals. Line 3 shows a string of consecutive bits to be written on tape. In line 4, the Write flip-flop receives each bit at a Load pulse and assumes the opposite state on the Complement pulse.

In line 7, the direction of tape magnetic polarization is labeled as R and L for right and left, respectively. The R-L and L-R transitions are detected by the Read amplifier as negative and positive half-sinusoids at pin E2 (opposite polarity at pin D2). If the tape is read in the same direction as written, the tape position corresponding to the time that the Write flip-flop was complemented will show a R-L change as a 1; a L-R change as a 0. The tape head voltages at Read amplifier inputs pins E2 and D2 are shown in line 8; the Read amplifier outputs are shown in line 9. During reading operations, the Load (RWB shift) pulses in line 10 (TP3) coincide with those in line 2 which complemented the Write flip-flop when writing. The R-L polarization change, representing a 1, results in a ground level at U at the time of the Load (shift) pulse. Consequently, as shown in line 11, a 1 is shifted into the RWB as the first bit read.

If the tape is read opposite to the direction in which it was written (when for example, reading reverse block numbers) the magnetic polarizations reach the head gap in reverse order; that is, the head senses a L-R change where a 1 was written, etc. The contents of the mark channel are selected to the advantage of this condition. Data written in one direction and read in the opposite direction will be complemented (see Figure 6-14).

6.6.4 Tape Processor Register Description

The following paragraphs discuss the major registers of the TC12 LINCtape Control as shown on the PDP-12 LINCtape Flow Diagram (see Figure 6-15).

Tape Accumulator (TAC) – The 12-bit TAC serves as the arithmetic register in the tape processor. When reading the datasum* is computed in the TAC and added to the checksum** read from tape to determine if the data transfer was accurate. When writing, the datasum is also computed in the TAC and, when the final Data Mark signal is detected, the resulting datasum is written on tape in complement form (checksum). The contents of the TAC (transfersum***) can be read into the CP accumulator, using the LINC-Mode TAC instruction. AC contents equal to minus zero indicate an accurate transfer. During SEARCH operations, the computations to determine the desired block number are performed in the TAC.

Tape Buffer (TB) – The Tape Buffer is an intermediate register primarily used to hold all the tape data from the RWB register. The TB receives the 12-bit tape word from the RWB register during a Read operation, and transfers the tape word to the RWB register during a Write operation. During the Read operation, when a tape word

*Datasum – 2's complement of all the data words in a block.

**Checksum – complement of the datasum, written on tape immediately following the Final Mark.

***Transfersum – contents of TAC after comparing a new datasum with the checksum previously written on the tape.

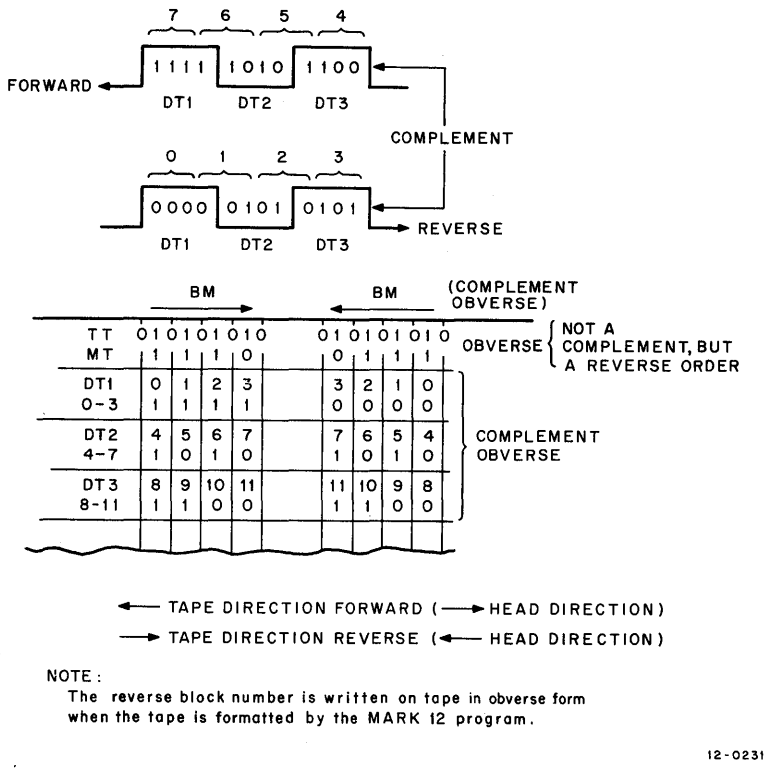


Figure 6-14 Reading A Block Mark

is to be transferred to the CP, a 12-bit parallel transfer is made from the TB to the CP memory buffer. When writing, the direction is reversed; information from the CP memory buffer enters the TB, and is transferred to the RWB register for disassembly onto the tape.

Read/Write Buffer (RWB) – The Read/Write Buffer register is a three-section shift register which corresponds to the three lines of data on tape (see Figure 6-5). During a Read operation, the RWB is loaded in four discrete operations three bits at a time. After the RWB has been loaded and shifted the fourth time, the fully assembled 12-bit tape word is parallel-transferred to the TB. During a Write operation the RWB is loaded in a single 12-bit parallel transfer from the tape buffer, where it is sequentially disassembled in four operations, three bits at a time, while being recorded on tape.

Tape Block Number (TBN) – The Tape Block Number 12-bit control register is loaded with the number of the tape block to be accessed in a data transfer. As the tape is searched, the block number read from tape is compared with that in the TBN. During group operations, the TBN contains the number of the first block to be accessed.

Tape Memory Address (TMA) – The Tape Memory Address 12-bit control register holds the Memory Address accessed during a data transfer. In Extended Address Mode, the TMA is loaded from the TMA Setup register at the beginning of a tape instruction. In standard mode, the TMA is loaded with IF 3 and 4 or DF 3 and 4, and GP0 and GP1, depending on the quarter of memory defined by MB bits 0, 1, and 2. The TMA is incremented by 1 for each data word transferred.

PDP-12 LINCTAPE FLOW DIAGRAM

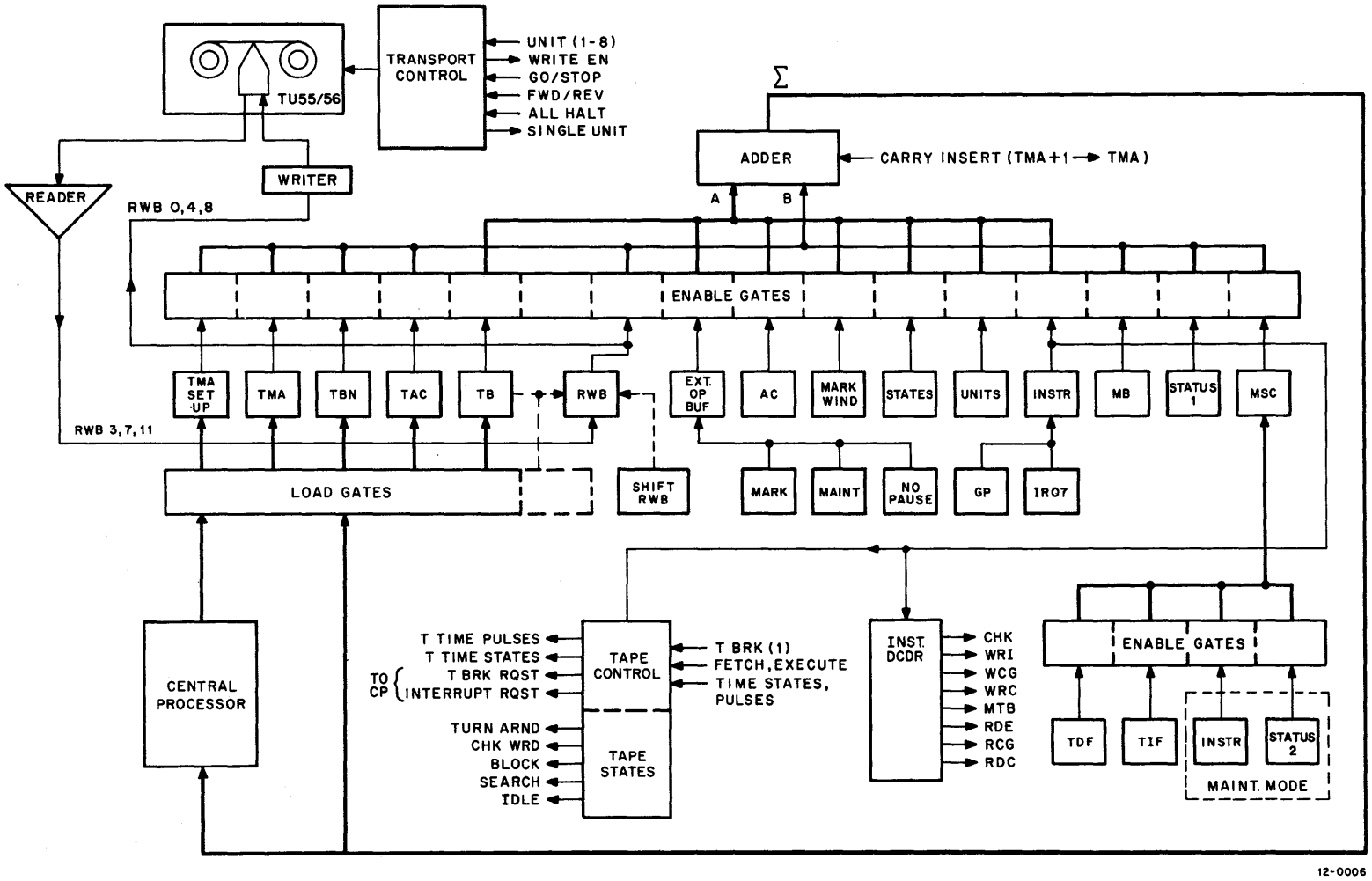


Figure 6-15 PDP-12 LINCtape Flow Diagram

TMA Setup Register – This 12-bit register defines the first address in memory to be accessed in a data transfer when operating in Extended Address Mode. The TMA Setup register is loaded from the CP accumulator with the MSC I 3 instruction.

Extended Operations Buffer (XOB) – This 12-bit control register selects the various extended tape operations. These operations include extended memory addressing, tape interrupt, the no-pause condition, hold motion, and extended tape units (tape units 2 through 7). The XOB is loaded from the CP accumulator. Refer to Paragraph 6.6.1, Extended Operations.

LINCtape Instruction Register and I Bit – This 3-bit register is loaded with the three least significant bits of the CP instruction register (IR), and decoded to determine the particular tape operation to be performed. The I-bit flip-flop is loaded from bit 7 of the CP instruction register. This flip-flop determines whether or not the selected tape unit will stay in motion after completion of the tape instruction.

Tape Bus Gating Network – All data transfers occurring in the LINCtape Control to and from the CP, with the exception of the RWB and the selected tape unit Read/Write heads, are implemented through the major-register gating network (see Figure 6-15). The network contains a separate gate structure and common register input bus for all 12-bit parallel transfers. Transfers between registers within the LINCtape Control, as well as transfers into and out of core memory, occur via the tape bus that is the output of the adder logic.

6.7 SYSTEM DRAWING DISCUSSION

The following paragraph is divided into seven areas; each keyed to one of the specific LINCtape Flow or Timing diagrams (D-FD-TC12-0-10 through D-FD-TC12-0-16) which comprise the Major State flow and timing of the LINCtape Control logic. The discussions and referenced drawings present an overview of the tape processor operations. Detailed discussions concerning the specific logic are presented in Paragraph 6.8 (Block Schematic Discussions).

6.7.1 Tape Processor Major State Flow (TC12-0-10)

A brief description of the five Major States of the TC12 Tape Control illustrated in TC12-0-10 is presented below:

IDLE: As the name implies, no tape operation is currently under progress. However, the tape processor control logic is ready to receive a tape instruction from the CP. Although the tape processor logic is inactive, a tape transport can be moving although it is not logically selected. If, during execution of the WRC, WRG, RDG, and RDC instructions, a bad checksum results, the IDLE state will be entered after each block transfer, and then the SEARCH state, to repeat the instruction.

SEARCH: The SEARCH state is entered only from the IDLE state. The tape processor is in this state when searching for a desired block. When the desired block is found and the selected tape transport is moving in the forward direction, the BLOCK state is entered.

BLOCK: The BLOCK state can be entered only from the SEARCH state with the TAC EQ 7777 * FWD * BM * TP2 signal. The tape processor remains in this state while performing all data transfers with the CP.

CHECK WORD: The CHECK WORD state is entered from the BLOCK state with the BLOCK (1) * TP2 signal. When all of the data constituting a block is transferred, the tape processor enters the CHECK WORD state, indicating that the checksum is accessed on tape. The CHECK WORD flip-flop controls the phase of the checksum written on tape. See drawing TC12-0-LTS and its description, Paragraph 6.8.15.

TURN AROUND: The TURN AROUND state is entered from the CHECK WORD state with I = 0 * CM * CHECK WORD (1) * TP1. Its main function is to stop the tape transport at the completion of a tape instruction when the I bit is not set.

6.7.2 Timing Diagram Discussions

The timing diagrams flow horizontally from left to right with the timing references shown vertically.

LINCtape Instruction Setup Timing (TC12-0-11) – Shown on the LINCtape Instruction Setup Timing drawing are:

a. The interrelationship of timing between the tape processor and the CP.

b. The two major timing setup pulses of a tape instruction execution; MTP SETUP and MTP SETUP 2.

c. Time of events and order of events beginning with receipt of a tape instruction through tape transport unit selection, direction, and motion control.

The execution of any tape operation by the LINCtape Control requires the sequence of events (some of which are conditional, depending upon the specific instruction), as depicted on the Instruction Setup drawing.

The Instruction Setup Timing drawing shows the interrelationship between time states of the CP (the lower half of the drawing) and the tape processor (the upper half). Note that no attempt was made to scale the time reference going from left to right across the drawing. The events depicted on this timing diagram occur during the initializing of the tape processor by the CP when a tape instruction is issued.

MTP SETUP (magnetic tape setup), conditions the tape processor to receive the first word of the two-word tape instruction (TC12-0-LIP). This process occurs at TS5 of the CP FETCH cycle. Several tape processor actions take place:

1. The PRE U2 flip-flop is loaded with bit 8 of the Instruction Register (TC12-0-LMU).
2. The Tape Instruction Register (TINR) is loaded (TC12-0-LIN) and the PROGRESS flip-flop is set (TC12-0-LIP).
3. The tape processor is forced to the IDLE state and the Window register is initialized (TC12-0-LWN). If the HOLD MOTION flip-flop is set, the tape UNIT EN flip-flop (TC12-0-LMU) is cleared.

During TS4 of the CP EXECUTE cycle, the following tape processor events occur: (MTP SETUP 2);

1. The Central Processor MB, which contains the starting block address, is enabled (bits 3 through 11 (TC12-0-LRE).
2. If an Extended Addressing Format Tape operation is to be performed, MB bits 0 through 2 are also enabled.

AT TP4 of the EXECUTE cycle of the CP, the MTP SETUP 2 pulse is generated and the following occurs:

1. The IN PROGRESS flip-flop (TC12-0-LIP) is set, enabling the tape processor to begin the tape instruction.
2. Central Processor MB bits 3 through 11 (0 through 11 if the EX ADD FORMAT flip-flop, TC12-0-LCX is set) are loaded into the TBN, and MB bits 0 through 2 are loaded into the GROUP Register (TC12-0-LGP).

3. Instruction Field 3 and 4 and Data Field 3 and 4 of the CP are loaded Tape Instruction Field 3 and 4 (TC12-0-LCXF).

After MTP SETUP 2, the tape processor is completely in control of the tape operation. If the NO PAUSE flip-flop (TC12-0-LCX) is not set, the CP pauses in TS5 and awaits a Tape Break Request as shown on drawing TC12-0-LIP.

A series of Motion Delay pulses are generated by the tape processor, as shown on drawing TC12-0-LTD:

1. MTN DLY 1 clears the MOTION flip-flop if a new tape transport unit is to be selected.
2. MTN DLY 2 deselects all tape transports and clears the UNIT EN flip-flop (TC12-0-LMU).
3. MTN DLY 3 loads the UNIT (tape transport select flip-flops and clears the DIRECTION flip-flop (Reverse), and selects the new tape unit (conditional).
4. MTN DLY 4 sets the UNIT EN and the MOTION flip-flops, if the selected tape is in an operational condition.

LINCtape Search Timing (TC12-0-12) – Shown on the LINCtape Search Timing drawing are:

1. The sequence and timing of tape processor actions during a SEARCH for a particular block address.
2. Address Setup calculations for the current block transfer, Normal LINCtape format non-group, Normal LINCtape format group, and Extended Format.

Before any tape transfer can be performed, the desired block (address) of data must be located.

Following the events described in the Instruction Timing Setup, the tape processor enters the SEARCH state. As shown on the SEARCH Timing drawing, the mark track is being scanned; when a Block Mark or Reverse Block Mark (depending upon transport direction) is sensed and decoded, the following events occur:

1. TS0 enables the RWB register, which contains the block number.
2. The TB is loaded with the contents of the RWB at TP0.
3. At TP4, the TMA is loaded with the first Memory Address to or from which the data is being transferred. In Extended Address mode, the TMA is loaded with the contents of the TMA Setup register.
4. At TS1 enable the TAC and TB, and load the TAC, performing the addition of both at TP1. The TAC contains the desired block number and the TB contains the block number just read from the tape which is in complement form. The TAC should contain 7777_8 when the desired block is located on the tape. When the desired block is located and the transport is moving in the forward direction, the BLOCK state is entered. If the block number read is not the desired block, the tape transport either continues in the same direction, or the direction is reversed, depending on the status of bit 0 of the TAC.

   For the MTB instruction the PROGRESS flip-flops (TC12-0-LIP) are cleared, terminating the instruction, and the contents of the TAC are transferred to the Central Processor AC if NO PAUSE is cleared (CP IDLE) (TC12-0-LCXF).

The format shown as "ADDRESS SETUP GATING" is as follows:

(TINR11 = 0) and (QN0 = 0) then (TIF0-2 → TF0-2) and (TIF3-4 → TMA0-1) and (QN1-2 → TMA2-3) and (0 → TMA4-11).

a. TINR11 = 0, indicates a nongroup instruction
TINR11 = 1, indicates a group instruction

b. QN0 = 0, indicates the operation involves the instruction field
QN0 = 1, indicates the operation involves the data field

(QN = quarter number is stored in the GP register)

If performing a group instruction, the second term of the above sample would be:

TAC9 = 0, indicating the operation involves the instruction field, or
TAC9 = 1, indicating the operation involves the data field

c. TIF0-2 → TF0-2, or TDF0-2 → TF0-2 identifies the particular 4K memory field involved.

d. TIF3-4 → TMA0-1, or
TDF3-4 → TMA0-1, identifies the field or memory bank.

e. QN1-2 → TMA2-3, identifies which quarter number of which 1/4 of the LINC memory field.

In Extended Format, the TMA Setup → TMA.

The upper two Address Setup Gating expressions constitute the nongroup instructions, consisting of RDE, RDC, WRI, and WRC.

The second two Address Setup Gating expressions constitute the group instructions consisting of RCG, WCG, and non-data transferring instructions: MTB and CHK.

The Extended Format instructions consist of AXO, XOA, TMA, and TAC.

Block Mode Reading (TC12-0-13) —The Block Mode Reading timing diagram shows the functions performed when executing Read instructions when in the BLOCK and CHECK WORD states.

The events depicted on this drawing occur when the tape processor is performing the following Read instructions and the desired block is located:

RDE	Read Tape
RDC	Read and Check
RCG	Read and Check Group

Observing the drawing from left to right, the Guard Mark (GM) is detected by the mark track decoder (TC12-0-LWN) and serves as a buffer area. The RWB is shifted left 1 place at TP3, and data from the three data channels are shifted into bits 3, 7, and 11 of the RWB.

When a Data Mark (DM) is decoded during a Read operation, the following functions occur:

1. The RWB is enabled at TS4 and is loaded into the TB at TP4.
2. At TS0, the TB and TAC are enabled; the data word in the TB and the partial datasum is in the TAC.
3. At TP0, the TAC is loaded (TAC * TB → TAC), and the TAPE BRK REQ flip-flop (TC12-0-LIP) is set, requesting a TAPE BREAK cycle from the CP.

If the CP is paused, it leaves the PAUSE state and enters the TAPE BREAK state. If the CP is not paused, the TAPE BREAK state is entered at the end of the current instruction. Timing is now referenced to the CP (CP timing).

1. At TP1, the TMA is transferred to the MA.
2. At TP2, the TMA is incremented and the TAPE BRK REQ flip-flop is cleared.
3. At TS3, the TB is enabled on the tape bus and loaded into the MB at TP3 for deposit in memory. The CP pauses again at TS5 and waits for another Tape Break Request. This is repeated for all Data Marks (DM).

After the first Tape Break, the CP pauses in the TAPE BREAK state for the completion of the block transfer.

When a Final Mark (FM) is encountered, indicating the last data word in the block, the identical operations are repeated.

After the FM, the Check Mark (CM) is encountered. This signifies that the checksum has been read and is in the RWB register.

1. The checksum is loaded into the TB at TP4 and added to the TAC at TP0. If the NO PAUSE flip-flop is equal to zero, the TAC is loaded into the AC at TP1.
2. The CHECK WORD state is entered at TP2. For a RDE instruction, if the (transfer check) TAC equals 7777_8, the PROGRESS and IN PROGRESS flip-flops are cleared at TP4. If the TAC is not equal to 7777_8, the PROGRESS and IN PROGRESS flip-flops are cleared at TP2 of the next CM, thus terminating the instruction. For the RDC instruction, if the TAC is not equal to 7777_8, the complete tape operation is repeated until the TAC does equal 7777_8. Then the PROGRESS and IN PROGRESS flip-flops are cleared, terminating the tape operation.

Block Mode Write (TC12-0-14) – Shown on the Block Mode Write timing diagram are the operations of the tape processor executing the Tape Write instruction:

WRI	Write Tape
WRC	Write and Check
WCG	Write and Check Group

NOTE
It is assumed that the correct block has been located on the tape (see SEARCH timing, TC12-0-12).

When the desired tape block has been located and the transport is moving in the forward direction, the BLOCK state is entered and the following tape processor events occur:

1. The WRITE SYNC flip-flop (TC12-0-LCS) is set with the next TP1 pulse after entering the BLOCK state, and the WRITE flip-flop is set with the following TP3 pulse.
2. The line counters (LC00 and LC01) are both set with BM * TP4.
3. After two counts (LC00 (0) LC01 (1)), the TAPE BRK REQ flip-flop (TC12-0-LIP) is set.
4. When the CP enters the TAPE BREAK state, data is read from the address specified by the TMA and is loaded from the MB into the TB register.
5. The TB is added to the TAC at TP0 to generate the datasum. The TB is also transferred to the RWB register at TP0.
6. At TP3, the outputs of the RWB register bits 0, 4, and 8 are complemented, thus writing the first line on tape. To complete the writing of a 12-bit word on tape (4-lines), the RWB is shifted left 3 times, and the output of RWB bits 0, 4, and 8 are complemented after each shift.

This sequence of events continues until the FM is decoded from the mark window (TC12-0-LWN) and the following events occur:

1. The datasum (which has been computed in the TAC) is loaded into the TB, transferred to the RWB, and written on tape in complement form.
2. At TP2 * FM, the CHECK WORD state is entered, controlling the PHASE level (TC12-0-LTS) which controls the writing of the checksum.

When the CM is decoded from the mark track, the WRITE SYNC and the WRITE flip-flops are cleared and the Write amplifiers are disabled.

For a WRI instruction, the WRITE CYCLE flip-flop and the PROGRESS flip-flop are cleared, indicating the completion of the instruction. For a WRC instruction, the WRITE CYCLE flip-flop is cleared and the check portion of the instruction is performed (see Block Mode checking). For the WCG instruction, if the GP EQ GPCNT flip-flop (TC12-0-LGP) is on a zero, the group count register is incremented and another block is written. This process is continued until the GP EQ GPCNT flip-flop is set, (1). Then the check portion of the instruction is performed.

Block Mode Checking (TC12-0-15) – Shown on the Block Mode Checking timing diagram are the operations of the tape processor when executing a CHK (Check one Tape Block), and the check portions of the WRC and WCG instructions.

The functions of the tape processor when executing the checks are similar to the Read instructions (TC12-0-13) except for the absence of the data transfers to the CP.

No tape processor operations occur during a GM. When a DM is encountered, the following events occur:

1. The RWB is enabled at TS4 and is loaded into the TB at TP4.
2. At TS0 the TB and TAC are enabled.
3. The TAC is loaded at TP0 (performs the addition of the TB and TAC).

The above action is repeated with each DM and the FM; thus, the datasum is now computed in the TAC.

When the CM is encountered, indicating that the checksum has been read, the following events take place:

1. The three functions in 1, 2, and 3 above are repeated.
2. If NO PAUSE flip-flop is cleared:
   a. The TAC is enabled on the tape bus at TS1.
   b. The tape bus is loaded into the Central Processor AC (TAC → AC).
3. The CHECK WORD state is entered at TP2.

For a CHK instruction, the PROGRESS flip-flops are cleared, terminating the instruction.

If a WRC instruction is being performed, and the contents of the TAC is not 7777_8 (indicating a bad check), the WRITE CYCLE flip-flop is set, the SEARCH state is entered, and the current block is rewritten and checked again. This sequence continues until a good transfer check (TAC = 7777_8) is obtained.

If a WCG instruction is being performed, the same functions occur as for the WRC. Each block in the group is checked and, if a bad transfer check is obtained, the WRITE CYCLE flip-flop is set, and the current block and the remaining blocks in the group are rewritten and checked until a good transfer check is obtained for the total number of blocks in the group.

Mark Timing (TC12-0-16) – Shown on the Mark Timing diagram are the functions of the tape processor when executing the MARK 12 tape formatting program. Accomplished when formatting virgin tapes are the following:

1. Recording the Timing tracks (tape tracks 1 and 10)
2. Recording the Mark track codes (tracks 2 and 9)
3. Numbering the blocks both forward and reverse (tracks 3 through 8)

The first three waveforms: mark clock, time counter 01, and time counter 00 (TC12-0-LTS) establish the necessary timing intervals for the timing tracks. The mark clock is a M401 Variable Clock, an RC-coupled multivibrator which produces timing pulses at a repetition rate of 7.5 μs (±5%) when enabled from the Mark flip-flop (TC12-0-LCX). Fine adjustment can be made from an internal potentiometer (see tape adjustments). The time counters constitute a simple frequency divider which will provide the 15 μs timing pulse separation necessary for the timing track.

LTS TIME WRITE (TC12-0-LTS) – is a 15-μs square wave that is recorded on the timing tracks to generate TP0 and TP3 when read from the tape.

PHASE (TC12-0-LTS) – provides a signal to introduce the magnetic polarization when writing in the mark and data channels.

Mark Window Register (TC12-0-LWN) – controls the format written on the mark track. It is loaded from AC bits 8 through 11, shifted left, and recorded serially to give the necessary control marks which define the areas on tape.

6.8 BLOCK SCHEMATIC DISCUSSION

The remaining paragraphs contain detailed discussions of the LINCtape Control logic. Each discussion of the referenced block schematic drawing (Volume III, D-BS-TC12-0-LCS through D-BS-TC12-0-LWN) also calls out the signals and logic levels originating or interfacing with other modules, for purposes of clarity. It is recommended that the reader read Paragraph 6.7, this manual, before utilizing the 6.8 discussions, to ensure a complete overview of the operations of the LINCtape Control.

6.8.1 Tape Control States and Instruction (TC12-0-LCS)

The five Major Tape State flip-flops: IDLE, SEARCH, BLOCK, CHK WRD (Check Word) and TURN ARND (Around) and their respective enabling logic are shown. In addition, the WRITE AND WRITE SYNC flip-flops and associated control logic are shown and discussed.

IDLE – There are five conditions in which the IDLE flip-flop is set and hence, the IDLE major state entered:

1. TAPE PRESET (Power Clear, ESF and AC_7 (1), IOT 6152 and AC_0 (1))
2. MTP SETUP (Start a tape instruction)
3. LTT CLOSE WINDOW (Tape timing is not correct)
4. CHECK WRD (1) * IN PROGRESS (1) * TP1 when the tape instruction is not completed in the CHECK WORD state; (WRC, WRG, RDG)
5. TURN ARND (1) * TP2 (End of tape instruction).

SEARCH – The SEARCH flip-flop is set when the tape processor is in the SEARCH state (indicating that it is searching for the desired block number on tape). It is set with the first TP1 pulse, and cleared when the desired block is located on the tape (with the signal TAC = 7777 * FWD * SEARCH * TP2). For the MTB instruction, it is cleared when the first Block Mark (BM) is encountered on the tape (MTB * BM * TP2). The SEARCH flip-flop, along with the BM signal (TC12-0-LWN), is used to generate the signals necessary to perform the initialization of the line counters and the setup of the TMA register.

BLOCK – The BLOCK flip-flop is set with the same signal that clears SEARCH. The BLOCK Major State is entered only from the SEARCH Major State. It is set during all active data transfers with the CP and is one of the qualifying levels to enable data on the tape bus and initiate a Tape Break request.

CHECK WORD – The CHECK WORD flip-flop is set when the tape processor has completed the transfer of a block of data. When a Read instruction has been performed, the CHECK WORD flip-flop is set, indicating that the checksum has been accessed and is available in the TAC. For a Write instruction, it complements the phase of the checksum and allows it to be written on tape in complement form.

TURN AROUND – When the TURN ARND flip-flop is set, the MOTION flip-flop is cleared when the next BM is encountered on the tape at the completion of a tape instruction. If the I-bit is not set, the TURN AROUND state is entered, thus allowing the tape transport to stop. The TURN ARND flip-flop is set with the signal, (CM * CHK WRD * TP1), and is cleared with (TP1 * BM * TURN ARND (1)).

The WRITE SYNC flip-flop synchronizes the WRITE flip-flop with the tape timing and mark tracks, thus protecting the control information (timing and mark tracks, block numbers, etc.) on tape, and assures that

data is written in the correct data areas of each block. The WRITE SYNC flip-flop is set with the first TP1 pulse after the tape processor enters the BLOCK state. The WRITE flip-flop is set with the next TP3 pulse. Both the WRITE SYNC and the WRITE flip-flops are cleared with LCS CLR WRITE, which is generated with CM and TS1 (1).

The WRITE CYCLE flip-flop – (lower left), is one of the qualifying levels for the WRITE flip-flop. During the check portions of the WRC and WCG instructions, it holds the WRITE flip-flop cleared. It also controls the COUNT GPCNT pulse (TC12-0-LGP) for a WCG instruction. The WRITE CYCLE flip-flop is set at the start of every tape instruction. In the event of a bad checksum for a WRC WCG instruction, it is set in order that the affected block can be rewritten. The flip-flop is cleared with CM * CHECK WORD * TP1.

The LCS TAPE OK signal is true when the requested tape transport has the correct manual switch settings:

Unit Selector on line (0 through 7)
REMOTE/OFF/LOCAL switch to REMOTE
WRITE/LOCK, ENABLED for Write instructions

6.8.2 Tape Extended Operations (TC12-0-LCX)

The Tape Instruction Field flip-flops 3 and 4 and the Tape Data Field flip-flops 3 and 4 determine which 1K field of a particular 4K memory bank is accessed for data transfers to tape. They are loaded with IF 3 and 4, and DK 3 and 4 at the beginning of a tape instruction (MTP SETUP 2).

The five flip-flops shown on the right select a particular Extended Tape operation. The MARK, TAPE INT EN, EX ADD FORMAT, NO PAUSE and HOLD MOTION flip-flops are selected with the AXO instruction and AC bits 4 through 9 respectively (AC06 (1) selects Maintenance Mode).

MARK – The MARK flip-flop is used during all tape-formatting operations.

TAPE INT EN – Allows the tape control to interrupt the CP at the completion of a tape instruction.

EX ADD FORMAT – Is used for nonstandard format tapes. When this format is selected, the eleven least significant bits (bits 1 through 11) of the second word of a tape instruction designate the tape block number. The first Memory Address for the data transfer is loaded from the AC into the TMA Setup register prior to the tape instruction. This can be any random 12-bit address. Group instructions cannot be performed when EX ADD FORMAT is selected.

NO PAUSE – When the NO PAUSE flip-flop is set, the CP does not pause for the duration of the data transfer. The CP continues with the program, and the tape control will request a tape break when it is ready to transfer a word of data.

HOLD MOTION – The HOLD MOTION flip-flop allows the operator to keep a tape transport in motion even though that particular unit is not logically selected by the tape control.

NOTE
With the HOLD MOTION and NO PAUSE flip-flops set, it is impossible to do two back-to-back tape instructions and enable both the 0 and 1 units for simultaneous motion.

6.8.3 Tape Extended Fields (TC12-0-LCXF)

Tape Fields 00 through 01 (LCXF TF0 through TF02) extend the capability of the TMA register to allow the addressing of up to 32K of core memory.

When the Extended Address format is selected, the fields are loaded with AC_{0-2} by the AXO instruction. When the Extended Address format is *not* selected, the fields are loaded with TIF 0 through 02 or TDF 0 through 02 by the BM * SEARCH * TP1 pulse. Tape Instruction Field 0 through 02 (TIF 0 through 02) and Tape Data Field 0 through 02 (TDF 0 through 02) are loaded with IF 0 through 02 and DF 0 through 02 respectively, at the initiation of a tape instruction. This allows full buffering of the tape processor; the CP does not have to provide the field assignment after the tape instruction is initiated.

6.8.4 Tape Group Counter (TC12-0-LGP)

The three group count flip-flops (GPCNT 0 through 2) and associated gating logic comprise the counter for the RDG and WCG instructions. The LPG COUNT GPCNT pulse increments by one (+1) at the end of each block transfer. The three group flip-flops (GP 0 through 2) are loaded with MB 0 through 2 from the second word of the tape instruction and indicate the number of additional blocks to be transferred after the first block. The GP CNT flip-flops are compared with the GP flip-flops after each block transfer. When the group (GP) and the group count (GPCNT) flip-flops are equal, the GP EQ GPC flip-flop is set, indicating that the requested number of blocks have been transferred.

6.8.5 Tape Instructions (TC12-0-LIN)

Two major operations are performed by the logic shown on the LIN drawing. The contents of the Tape Accumulator (TAC) are examined for the correct checksum (7777_8), and tape instruction decoding. Decoding of the tape operation to be performed is accomplished by the three tape instruction register flip-flops TINR 0 through 2 shown on the right side of the drawing. Decoded are bits 9 through 11 of the first word of the two-word tape instruction. The contents of the CP Instruction Register (INR 9 through 11) are loaded into the tape instruction register by the initiation of the tape instruction, and decoded by the binary-to-octal decoder shown in the upper right of the drawing. The I flip-flop, when set, allows the selected tape transport to be left in motion after completion of the current instruction. When the I flip-flop is a zero, the tape control will enter the TURN AROUND state, thus stopping the transport.

The binary-to-octal decoder is the same as those used by the CP for instruction decoding (M161). Outputs 0 through 3, true when bit 9 is a zero, are the Read and MTB instructions. Outputs 4 through 7, true when bit 9 is a one, are the Write and Check instructions.

The AND gate network shown on the left of the drawing decodes the contents of the TAC register. The gate is qualified when the datasum for the Data Transfer instruction is correct, and also when the desired tape block has been located during a SEARCH.

6.8.6 Interprocessor Signals (TC12-0-LIP)

The primary tape control interprocessor signals are generated by the logic shown on this drawing. The tape processor is fully buffered and independent of the CP; therefore, certain signals must be provided to the CP from the tape control so that both processors may operate asynchronously or interleaved, depending upon the program.

TAPE BRK REQ – When the TAPE BRK REQ flip-flop is set, the tape processor is indicating to the CP that it is ready to effect another transfer of a word of data. The CP then enters the TAPE BREAK state and the memory address register (MA) is loaded with the tape memory address (TMA) at TP1. If the signal LIP TAPE OUT (Read) is true, the contents of the tape buffer (TB) are enabled on the tape bus, loaded into the MB, and subsequently written into memory. When the signal LIP TAPE OUT is not true (Write), the contents of memory designated by the MA are loaded into the MB, then loaded into the tape buffer (TB), and subsequently written on tape. This operation is similar to all one-cycle Data Break I/O devices.

CHK SUM LOAD AC – The CHK SUM LOAD AC signal is supplied to the CP register load control logic, where it is used to generate an AC load signal to load the checksum into the AC.

TAPE PAUSE – When the TAPE PAUSE signal is true, the CP will pause until the complete data transfer has been accomplished. When the TAPE PAUSE signal is not true, the CP continues to execute the main program after initiating the tape instruction.

TAPE INTERRUPT – This signal is connected to the common interrupt bus of the CP. At the completion of a tape instruction the TAPE DONE flip-flop is set and, if the TAPE INTERRUPT is enabled, the CP is interrupted, signaling the CP that the tape instruction is complete.

IN PROGRESS and PROGRESS – These two flip-flops indicate the status of the tape processor. When they are set, there is some tape operation in progress. The PROGRESS flip-flop is set with MTP SETUP (TS5 of the first word of the tape instruction) and generates the TAPE PAUSE signal if NO PAUSE operation is selected (see Tape Extended Operation TC12-0-LCX). The IN PROGRESS flip-flop is set with MTP SETUP 2 (TS5 of the second word of the tape instruction) and is instrumental in controlling the tape processor states. Both flip-flops are cleared at the completion of all tape instructions except the MTB instruction. In this case, the IN PROGRESS flip-flop is cleared when the first BM is encountered on tape. The PROGRESS flip-flop is cleared at this time only if the ACIP delay (TC12-0-LTD) has timed out. This prevents the issuance of another tape instruction while the tape transport is turning around, thus preventing undesirable tape snapping.

TAPE WORD – This flip-flop is used in conjunction with the MARK 12 tape formatting program. It is applied to the skip logic and is checked with the SXL 17 instruction.

6.8.7 Tape Unit and Motion (TC12-0-LMU)

The logic shown in this drawing performs tape transport selection, direction and motion control, and enabling levels. Tape unit selection is controlled by the unit flip-flops U 0 through 2 and the preunit flip-flops PREU 0 through 2 which select up to eight TU55 or four dual tape transports (units). The tape unit flip-flops are decoded by the transport control logic (shown on TC12-0-LTC). PRE U2 and U2 select units 0 and 1. The PRE U2 flip-flop is loaded with bit 08 of the CP instruction register, with the LIP MTP SETUP pulse.

PRE U0 and PRE U1 are loaded with AC bits 10 and 11, respectively, by the AXO instruction. PRE U0 through 2 are jam-transferred into U0 through 2 with the LTD MTN DLY 3 signal. The preunit flip-flops are compared with the unit flip-flops to determine if a different tape unit is being selected. When a change in tape units is made, LMU NEW UNIT is true (high) and the motion flip-flop is cleared; it could have been set from the previous tape instruction. The UNIT EN flip-flop enables the unit selection signals to the tape transport, thereby eliminating transients on the selection when changing tape units.

The DIRECTION (an OR gate) flip-flop controls the direction of tape travel on the selected tape unit. When DIRECTION is cleared (0), the tape moves in the Reverse direction (when facing the transport; left-to-right). When the flip-flop is set (1), the direction is Forward (right-to-left).

At the beginning of each tape instruction, reverse direction is selected if the motion flip-flop is cleared.

6.8.8 Tape Register Enable Control (TC12-0-LRE)

The enable signals generated by the logic shown on this drawing are used to condition the tape processor major registers, the CP accumulator (AC) and memory buffer (MB).

The EN AC level is generated by a TMA (MSC I 3) instruction; AC → TMA Setup. It is also generated with a tape maintenance (IOT 6154) instruction. The EN MB, which is generated by the tape processor during the second word of a tape instruction, enables the block number to be loaded from the MB into the TBN register. It is also enabled during the TAPE BREAK cycle of all Write instructions. The MB is loaded into the tape buffer (TB).

The logic shown on the lower left side of the drawing generates the levels necessary to perform the correct addressing of memory. These signals are enabled at TTS 4 • BM • SEARCH. When TINR (Tape Instruction Register)

bit 11 is a 1, to determine a group instruction, specific address gates are enabled. If bit 11 is a 0, the instruction is a non-group instruction and a different addressing sequence is utilized.

The logic shown on the right side of the drawing generates the tape processor major register enable signals. To accomplish a data transfer from a particular register, this register is enabled onto the tape bus, and the appropriate load pulses are generated. The EN GPCNT signal for the Group Count register (for a multiple block transfer) is also generated the same time as the EN TBN (TBN + 1 → TBN during multiple block transfer).

6.8.9 Tape Register Load Control (TC12-0-LRL)

The Load pulses shown on this drawing strobe the data from the tape bus into the specified registers. When the register is clocked, the data enabled on the tape bus is jam-transferred into that particular register.

The SHIFT RWB pulse shifts the RWB register left in three four-bit segments. Data from the three data channels (on tape) are read into bits 3, 7, and 11 of the RWB register when reading, and, when writing, data is shifted out on RWB bits 0, 4, and 8.

6.8.10 Transport Control (TC12-0-LTC)

The control logic shown on this drawing provides the interface between the tape processor and the TU55/56 Tape Transports. The right side shows the unit select gating network and the logic converters. Both the TU55 and TU56* require negative logic levels of -3 and 0V.

Shown on the left side of the drawing are the transport control logic level converters and the transport unit selector decoder.

The B UNIT SEL level is true when only one tape transport is selected. The WRITE EN LEVEL is true when the WRITE switch is (operator-selected) enabled on the selected tape unit. Both of these signals generate the TAPE OK level shown on the TC12-0-LCS drawing.

6.8.11 Tape Delays (TC12-0-LTD)

The tape delays are divided into two basic groups: the motion delays and the tape transport fail delays.

The motion delays MTN DLY 1 through 4 provide the signals that select the tape unit and control the selected tape unit motion (forward or reverse). MTN DLY 1 is used to clear the MOTION flip-flop (TC12-0-LMU) if a different tape unit has been selected since the previous tape instruction. MTN DLY 2 deselects all tape units by clearing the LMU UNIT EN flip-flop (TC12-0-LMU). MTN DLY 3 zeros the DIRECTION flip-flop (TC12-0-LMU) (reverse) and selects the new tape unit. MTN DLY 4 sets the UNIT EN and MOTION flip-flops (TC12-0-LMU) if the TAPE OK signal is true (TC12-0-LCS).

Six status signals are generated by the delay generators shown in the upper half of the LTD drawing: the TTOK (Tape Timing OK), XTLK (Crosstalk), ACIP (Acceleration In Progress), TAPE FAIL DELAY, NO TAPE and TAPE FAIL.

TTOK – The TTOK (Tape Timing OK) delay assures that the tape is moving fast enough over the tape head (within 50 percent of maximum) so that the information read from tape is meaningful. When the outputs are not true, tape timing pulses are inhibited and the tape processor enters the IDLE state. The M307 Integrating-One-Shot is triggered by the signal from the timing track (LTT READ 0) at an approximately 30 μs rate, and has a delay period of 48 μs. The delay is also triggered by LTT SIMULATE TP0, which results from an IOT 6151 Maintenance instruction.

*optional positive

XTLK DELAY – The cross-talk delay inhibits erroneous tape timing pulses (TP0 and TP3) due to noise on the signal from the timing track. When the outputs are true, tape timing pulses are inhibited. The M307 Integrating-One-Shot is triggered by TP0 and TP3 at approximately a 15 μs rate, and has a delay period of 9 μs .

ACIP – The Acceleration In Progress signal is used to inhibit tape timing pulses while the tape transport is in the process of accelerating up to speed and decelerating for a tape turnaround, thereby allowing the tape to reach operational speed before allowing the timing generator to decode the timing track. The M307 Integrating-One-Shot is triggered by a change in direction or when the MOTION flip-flop goes to the 1 state, and has a delay period of 180 μs.

TAPE FAIL DELAY – This 300 ms delay generates a NO TAPE signal, which will then qualify the TAPE FAIL signal to control the MOTION flip-flop (TC12-0-LMU).

TAPE FAIL – The TAPE FAIL DELAY generates a TAPE FAIL pulse every 300 ms, when tape timing pulses fail to occur after the initiation of a tape instruction. The M307 Integrating-One-Shot is triggered by MTP SETUP and by the NO TAPE pulse, which is generated when the delay times out. If the absence of the TP0 pulse was due to the TAPE OK signal, the MOTION flip-flop remains cleared until this signal becomes true.

6.8.12 Tape Maintenance Signals (TC12-0-LTM)

The 4-bit maintenance instruction register shown on this drawing is decoded to generate the enable signals for the various tape maintenance operations performed with IOT 6154. IOTs 6151 and 6152 are gated with the AC bits to generate the pulses to load the various tape registers.

These IOTs are used for maintenance purposes only.

6.8.13 Tape Reader-Writers (TC12-0-LTR)

Shown on the drawing are the Read/Write circuits associated with the five tape channels. These circuits consist of G882 Modules, which are high gain differential amplifiers connected directly to the Read/Write heads through W032 Connector Cable. The reader-half of the module receives outputs from the Read/Write head to produce complementary outputs at terminals V2 and U2. At terminals J2 and K2, the writer-half of the G882 Module produces complementary outputs that go into the Read/Write head for writing on tape. The W603 converts the G882 negative level (0V, -3V) to positive (0V +3V) levels. The W512 converts the M222 positive logic (0V, +3V) to negative logic (0V -3V) levels. There are five identical tape channel tracks as follows: one timing, one mark and three data tracks.

Basic timing signals for tape operation are provided by the timing track through the tape Read/Write head shown on the left of the drawing. The timing signal read from this channel is essentially a sinusoid with a period of approximately 30 μs; each time the sinusoid crosses the zero reference (15 μs), the reader output changes. The output from the reader is connected to a +3V and 0V line by W603 and goes through buffers to generate LTR T READ, which in turn generates TP0 and TP3, the two basic timing pulses for tape processing.

Unique control marks, defining specific areas of tape, are recorded on the mark track. This data is read by the mark track reader and sent to the window shift register (see Tape Mark Window), for decoding. During the marking of tape (using the MARK 12 program, for instance), data is entered into the mark channel writer from the window register flip-flop 0 at TP0. The mark writer flip-flop is then complemented at TP3. Therefore, the magnetic flux on the tape is changed from 0 to 1, or from 1 to 0 in synchronization with TT3 (corresponding to the time that the three data writers are complemented). This process insures that the tape head signal, when read back, will be maximum at TP3, which means that the Read amplifier will be fully saturated with either a 0 or a 1 at TP3 when data is read.

Three data tracks are used for data storage. Data is recorded on tape serially, in groups of four 3-bit lines which correspond to the one 12-bit data word. The Read/Write buffer disassembles data to be written on tape or receives (assembles) data from the three data tracks during reading. Writing of data from the RWB is accomplished by data writer flip-flops D1-D3 WRITE, respectively, by the TP0 pulse. Approximately 15 μs later, the writer flip-flops are complemented by time pulse TP3, which writes the data on tape. This flux change saturates the reader on readback when data is strobed in at TP3. TP3 initiates the RWB shift-left pulse, which loads the reader outputs into bits 3, 7, and 11 of the RWB register and shifts the other RWB bits left one place.

6.8.14 LINCtape Register Bus (TC12-0-LTRA-F)

The six 12-bit registers shown on this drawing store information during all LINCtape operations. The 12-bit inter-register transfers are gated into the major register network by enabling gates (TC12-0-LRE). All the enabling gates are conditioned by the tape processor or maintenance enable levels such as the EN TAC, EN TB, etc. These enable levels allow the data from a register to enter the major register gating network in a parallel transfer. When the contents of a register(s) are enabled, the data enter the major register gating network through the adders and onto the tape bus lines. When any inter-register transfer within the tape processor is performed (excepting the serial parallel shifting transfers of the RWB register), all 12-bit inter-register data enter the tape bus lines 00 through 11 through the major register gating and adder networks. These lines are the data input (terminal D) of all the register flip-flops. The data on these bus lines are loaded into the specified register by a clocking action; i.e., if the TAC is the destination, the TAC LOAD (TC12-0-LRL) pulse is generated. There are five major register Load pulses. They are: LOAD TMA SETUP, LOAD TBN, LOAD TB, LOAD TMA and LOAD TAC. Only the RWB register does not require the use of the adder. The adder has a carry input which allows a 12-bit addition with ripple carry.

Special note should be taken of the Read/Write Buffer register insofar as it functions as a shift register. During a Tape Read operation, the data is loaded into the RWB bits 3, 7, and 11 from the tape readers by the SHIFT RWB pulse. After the four separate Read and Shift Left operations, the RWB register contains the complete 12-bit word. Then an EN RWB and a Load TB pulse loads the word into the tape buffer (TB) for transfer to the CP on the tape bus.

During a Tape Write operation, the TB is loaded (in parallel) into the RWB register. When the word is transferred to tape, it is loaded from RWB bit locations 0, 4, and 8 to the tape heads (D3, D2, and D1 respectively).

6.8.15 Tape States (TC12-0-LTS)

Shown on the Tape States drawing are the five tape time states generator flip-flops (TTS00 through 04), the line counter flip-flops (LC00 through 02), a mark clock, and time counters TC00 and TC01.

The five tape time states (TTS00 through 04) are controlled by corresponding tape time pulses TP0 through TP4 (TC12-0-LTT). The time state flip-flops are used to synchronize the loading of the various tape registers. Data are enabled on the tape bus with a time state signal, and the selected register is loaded with the time pulse. As in the CP, time states enable and time pulses load the registers. A time pulse terminates a time state and sets the next state; i.e., TP0 terminates TTS0 and sets TTS1.

The line counter comprises LC00 through 01, and LC 02 into a 3-bit up counter that synchronizes the 12-bit computer word with the tape word (4 three-bit lines). LC00 and LC01 provide the four count, and LC02 provides the clock pulse. When a Write operation is executed, the TAPE BRK REQ is set (TC12-0-LIP) at a count of one, indicating that the tape processor is ready to receive another word from the computer. LC02 controls the PHASE level that writes data on tape (TC12-0-LTRA-F).

The TIMING OK signal verifies that the selected tape transport is operating correctly to accomplish data transfers. All the tape timing pulses are inhibited when this signal is not true.

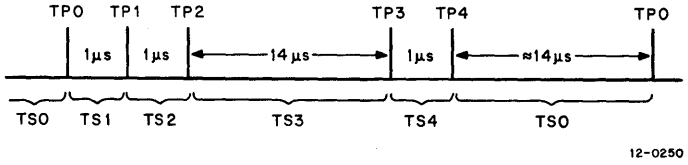
The TIME WRITE and MARK CLOCK signal are used to generate a timing track when formatting a virgin tape.

The MARK CLOCK clocks TC01 every 7.5 μs. The output of TC01 is a 15 μs square wave which clocks TC00.

6.8.16 Tape Time Pulses (TC12-0-LTT)

The timing track signal from the tape is used to establish all tape processor synchronization. TP0 and TP3 generate the pulses to establish the tape time states (TC12-0-LTS).

The logic circuitry shown on this drawing gates the timing signals from the Read/Write tape heads (TC12-0-LTR), shapes and amplifies them for generating TP0 and TP3. These pulses occur at an interval of approximately 15 μs. TP0 generates TP1 one μs later, and TP1 generates TP2 one μs later. TP3 generates TP4 one μs later. These pulses generate the Load pulses required to transfer data to the various registers.



The CLOSE WINDOW pulse initializes the tape processor. It is generated by TAPE PRESET (TC12-0-LIP), (TC12-0-LIP), MTP SETUP at the beginning of each tape instruction and when the tape timing does not meet specifications.

6.8.17 Tape Mark Window (TC12-0-LWN)

Shown in the drawing is the mark track logic which consists of a four-bit window register, a window shade flip-flop, and all the necessary gating for controlling and decoding the window register.

When a virgin tape is to be formatted, AC bits 8, 9, 10, and 11 are loaded into the window register and shifted left, out to the mark track reader/writer and written on the mark channel of the tape. During all other tape operations, this information is read back by serially shifting the mark track data with the LOAD WINDOW pulse, into the window register (WIND 03). The four window register bits and the SHADE are decoded to generate the various signals necessary to identify the information read/written in the three data channels (D1, D2, and D3). These signals enable the appropriate logic levels which control the data transfers.

CHAPTER 7

PREWIRED OPTIONS

7.1 INTRODUCTION

Thirteen prewired options are available for the PDP-12 Computer System; the mainframe of the system is prewired to accept these options. Table 7-1 lists the options in alphabetical order and their applicability to the various system configurations that are defined in Chapter 1 of the *PDP-12 System Reference Manual.*

Table 7-1
Prewired Options

Type	Name	System Application
1. AG12	Preamplifier	PDP-12A
2. AM12	Expanded Multiplexer	PDP-12A
3. DP12-A	Teletype Dataphone	PDP-12A, B, C
4. DP12-B	Teletype Dataphone (EIA Level)	PDP-12A, B, C
5. KE12	Extended Arithmetic Element	PDP-12A, B, C
6. KP12	Power Failure Restart	PDP-12A, B, C
7. KT12	Time-Sharing Options	PDP-12A, B, C
8. KW12-A	Real-Time Clock	PDP-12A, B, C
9. KW12-B	Simple Clock	PDP-12A, B, C
10. KW12-C	Simple Clock	PDP-12A, B, C
11. TC12-F	8 Tape Control	PDP-12A, B
12. XY12	Incremental Plotter	PDP-12A, B, C

7.2 AG12 AND AM12 OPTIONS

The analog-to-digital converter (AD12) includes 16 channels of input through a FET-switched multiplexer that drives a SAMPLE & HOLD circuit. The SAMPLE & HOLD output is converted by the 10-bit A/D converter that is controlled by the LINC-Mode SAM instruction. Eight of these sixteen channel (Channels 0 through 7) inputs are wired directly to potentiometers that are used by numerous software programs as parameter inputs. The remaining eight channels (Channels 10 through 17) are wired to differential preamplifiers that provide an input range of ± 1V and an input impedance of 70 kΩ 300 pF in parallel. The common frequency of the preamplifiers passes signals up to 60 kHz at 30B down. The preamplifiers provide 10,000 percent overload protection with 1 μs recovery time. As explained in Chapter 5 on the A/D Control, two modes of sampling operation are used, in which the user is given the option of pausing until A/D conversion is complete, or continuing with the programming.

The AD12 can be expanded by another 16 channels for a total of 32 channels by ordering the AM12 option.

7.2.1 AG12 Preamplifier

If high input impedance for sensitive signals is required, the AG12 option provides an additional 16 differential preamplifiers. The preamplifiers consist of two A214 Modules that are plugged into locations E29 and F29. An extra Analog Extension Panel Assembly (DEC Part No. 7006046) with two blue ribbon connectors (Amphenol 26-4401-24P) are also provided. The Amphenol connector is mated with a connector (DEC Part No. 12-03578) to provide user interface. Two 6783 Modules are used to route signals from locations F30 and F31 to the Amphenol connectors. The Analog Extension Panel Assembly is shown on Volume III drawing D-AD-7006040-0-0.

The specifications for AG12 are as follows:

Input Voltage Range:	±1V
Input Resistance:	70 kΩ, ±2%, 300 pF in parallel
Common Mode Rejection:	±3.5V from system fault line ground
Input Protection:	±67V from fault line indefinitely
Overvoltage Recovery Time:	8 μs
Frequency Response:	0- to 30-kHz flat 60 kHz – 3 dB down

7.2.2 AM12 Expanded Multiplexer

The AM12 option consists of two additional A131 Multiplex Modules that are plugged into locations D31 and D30 respectively. User interface is provided through two W021 Modules, which are plugged into location F30 (CHA 20 through 27) and F31 (CHA 30 through 37) on the computer wire panel.

The specifications for the AM12 are as follows:

Input Voltage Range:	±1V
Input Resistance:	110 MΩ, 300 pF to selected mutiplex
Input Protection:	120V RMS for 5s ±8V indefinitely
Common Mode Rejection:	None

7.2.3 Logic Description

As shown in Figure 7-1 and the *TADC* print, IR bits 07 through 11 are used to select a particular multiplexer. IR bits 08 and 09 are decoded to provide N EQ 00, N EQ 10, N EQ 20, and N EQ 30. These signals are used to partially enable the select logic for one group of 8 (Channels 0 through 7, 10 through 17, 20 through 27, or 30 through 37) FET switches. IR bits 09, 10, and 11 are decoded into 00 through 07 to provide the signal necessary to enable a particular FET switch. The control instruction for this logic is SAM, which has been described in Chapter 1.

7.3 DP12-A TELETYPE AND DP12-B ASYNCHRONOUS MODEM

The DP12-A Teletype and DP12-B are prewired options that permit interfacing a second Teletype or Modem, respectively, to the PDP-12 Computer. Either the PDP12-A or the PDP12-B may be installed at one time, but both cannot be installed at the same time.

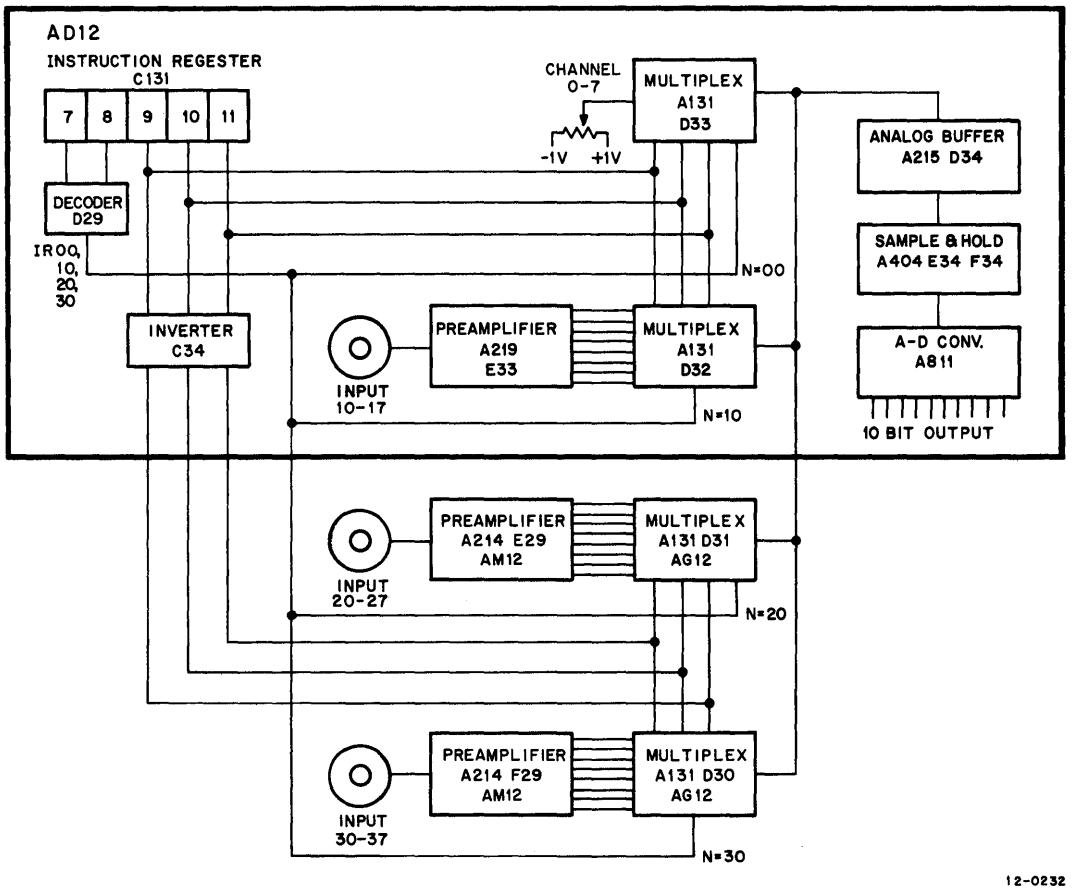


Figure 7-1 Multiplexer Selection

The DP12-A Teletype Interface is designed to interface 110-baud (10 characters per second) asynchronous devices utilizing US ASCII bit code. Teletype input (DTTI) to the printer or paper-tape punch, as well as Teletype output (DTTO) from the keyboard or paper-tape reader, is transferred by the DP12-A under program control. The 33 ASR Console Teletype is recommended for this application.

The DP12-B Asynchronous Modem Interface is a crystal-controlled interface capable of accepting data at any user-specified rate in the 110 to 100,000 baud range. The unit is designed for US ASCII bit code. Operation is full-duplex, and signals are EIA-compatible. Suitable devices utilizing the DP12-B as an interface are Bell Datapone Model No. 103, Datapoint 3300 Display, or any other EIA-compatible (see EIA-standard RS232-B) device meeting the requirements mentioned above. The DP12-B can also be used for interprocessor communications.

7.3.1 Component Description

The module requirements for DP12-A and DP12-B options are shown in Table 7-2.

The M850 Module is connected at one end of the 25-ft cable (BC01-A-25), that is supplied with the DP12-B. The other end is terminated in a 6 in. DB-25P 25-pin connector, wired in accordance with EIA Standard RS232-B. The W076 Module is mounted on the cable to a DEC-supplied Teletype. Either a Teletype-compatible W076 Teletype Connector or M850 EIA Lever Converter can be utilized as a connector on the DP12-A and DP12-B.

Table 7-2
Module Requirement for DP12-A and DP12-B

Module	Description	Module Slot	DP12 A	DP12 B
M706	Teletype Receiver	M09, N09	x	x
M707	Teletype Transmitter	M10, N10	x	x
G700YA	Dataphone Disable	N03	x	x
M216	Six Flip-Flops	N12		x
M405	Crystal Clock	N11		x
W076	Teletype Connector	N03	see below	
G718	Timing Jumper	N11	x	
M850	EIA Level Converter	N03	see below	

To prevent false computer interrupts when neither module is in place, a G700YA Dataphone Disable Module is supplied. This module, when inserted in connector slot N03, grounds the signal DTTI DATA L, disabling the Interrupt from the DTTI KEYBOARD FLAG.

When ordering a DP12-B option, the intended baud rate must be specified. The crystal clock (M405) is selected to be 128 times the baud rate at baud rates less than 10,000 baud. For baud rates between 10,000 and 100,000 baud, the oscillator frequency is 16 times the baud rate. In this higher rate range, it is also required to remove the wire connecting N12L2 to N11D2 on the mainframe and add a jumper wire from N12L2 to N12M1.

Examples:

a. For a computer display terminal operating at 2400 baud:
Crystal clock frequency = 2400 baud x 128 = 307.2 kc.

b. For a modem operating at 10,000 baud:
Crystal clock frequency = 10,000 x 16 = 16 kc.

7.3.2 Programming

Programming of the DP12 is identical to that of the system Teletype, but uses I/O codes 40 and 41. Instruction mnemonics for the DP12 System are not recognized by the program assembler (LAP6-DIAL V1 through V5), and must be defined by the programmer (see Table 7-3). The instructions use IOC IOP1 for performing skips, IOC IOP2 for clearing flags and the AC, and IOC IOP4 to perform data transfers. The LINC instruction ESF (AC bit 6 = 1) can be used to disable computer Interrupts when the Keyboard Flag is raised.

Table 7-3
DP12 Instruction Mnemonics

Instruction	Octal Code	Mnemonic	Description
Skip on Keyboard Flag	6401	DKSF	The contents of the PC are incremented by 1, skipping the next instruction when the Keyboard Flag is high.
Clear Keyboard Flag and AC	6402	DKCC	Clears the AC and Keyboard Flag, assembles a new character in the DTTI buffer, and resets the Keyboard Flag high when ready to transfer a new word.

(continued on next page)

Table 7-3 (Cont)
DP12 Instruction Mnemonics

Instruction	Octal Code	Mnemonic	Description
Read Keyboard Buffer Static	6404	DKRS	The contents of DTTI buffer are ORed with bits 4 through 11 of the AC. Neither the AC nor the Keyboard Flag is cleared.
Read Keyboard Buffer Dynamic	6406	DKRB	Microinstruction of DKCC and DKRS. AC and Keyboard Flag cleared, contents of DTTI transfer into bits 4 through 11 of the AC. A new character is loaded into the DTTI register, resetting Keyboard Flag high.
Skip on Teleprinter Flag	6411	DTSF	The contents of the PC are incremented by 1 (skipping the next instruction) when the Teleprinter Flag is high.
Clear Teleprinter Flag	6412	DTCF	Set the Teleprinter Flag low.
Load Teleprinter and Print	6414	DTPC	The contents of bits 4 through 11 of the AC are loaded into the DTTO buffer, then shifted out serially to transmit a character. The Teleprinter Flag is high.
Load Teleprinter Sequence	6416	DTLS	Microinstruction of DTCF and DTPC. The Teleprinter Flag is cleared and the contents of bits 4 through 11 of the AC are transferred to the DTTO buffer. When the character has been shifted out, the Teleprinter Flag is reset to a high. A new character can be loaded into the DTTO buffer now.

7.3.3 Cable Connections – DP12-B

The 25-pin connector mounted on one end of the BC01-A Cable meets EIA standards. (A logical 1 is defined as being greater than +3V, and a logical 0 less than -3V.) The connector wiring is as shown in Table 7-4.

7.3.4 Logic Description

Logic used for DP12 is similar to that used on the PDP-12 Teletype. Refer to Chapter 4 for details.

7.4 KE12 EXTENDED ARITHMETIC ELEMENT

The Extended Arithmetic Element (EAE) option for the PDP-12 enables the CP to perform arithmetic operations at higher speeds. These higher speeds are made possible by incorporating the EAE components with the existing CP logic in such a way that they operate asynchronously. The EAE components consist of EAE timing and control logic, a 12-bit Multiplier Quotient Register (MQ), and a 5-bit Step Counter (SC). The EAE instructions are discussed in Section 7.4.4.

7.4.1 Timing and Control Logic

The KE12 logic circuits operate in conjunction with the accumulator (AC), MQ link (L), and memory buffer (MB), to perform parallel arithmetic operations of high speed on positive binary numbers. Figure 7-2 is a simplified block diagram of the KE12 EAE option.

Table 7-4
DP12-B Cable Connector Wiring

Pin Number	Signal Name
1	Protective Ground*
2	Transmitted Signal**
3	Received Data**
7	Signal Ground*
20	Data Terminal Ready†
* Pins 1 and 7 are tied together ** With respect to the computer † Held at +5V	
Cable Connection – Slot N03	
A2	+5V
B2	-15V
C2	Ground
D2	
E2	DTTI DATA IN L
F2	
H2	DTTO MAGNET DRIVER H
J2	
K2	
L2	
M2	DTTI READER RUN (0) H
N2	
P2	
R2	
S2	
T2	

Most of the transfer of information between the KE12 and the PDP-12 CP occurs during the EAE instruction FETCH cycle. All arithmetic operations, with the exception of NMI (normalize), require a FETCH cycle for referencing the next memory location. During the FETCH cycle, one of the operands for a multiply (MUY) or a divide (DVI) is obtained, or the number of shifts to be performed during the long-shift feature is obtained.

The T5 RECYCLE is set by the EAET ON to provide EAET5 RECYCLE when programmed for arithmetic operations, which require at least 7.8 μs for completion. Setting the CPT T5 RECYCLE prevents the CP from advancing in its cycle until the arithmetic operation is completed. At the finish of the arithmetic operation, the EAE logic clears the EAET ON flip-flop to 0, which causes the CPT RECYCLE SYNC flip-flop to clear. Then, 50 ns after the CPT RECYCLE SYNC flip-flop clears, CPT T5 RECYCLE clears and the next TP5 pulse sets TS1, allowing the CP to continue with the program. MB bits for EAE are decoded in the N-register of the CP and are routed to the EAE logic.

7.4.2 Multiplier Quotient Register (MQ)

The MQ is a 12-bit register that acts as an extension of the AC during EAE operations. The MQ contains the multiplier at the beginning of a multiplication, and the least significant half of the product at the conclusion. The MQ contains the least significant half of the dividend at the start of a division and the quotient at the end. The MQ contains the least significant part of a number during a shift or a normalize operation.

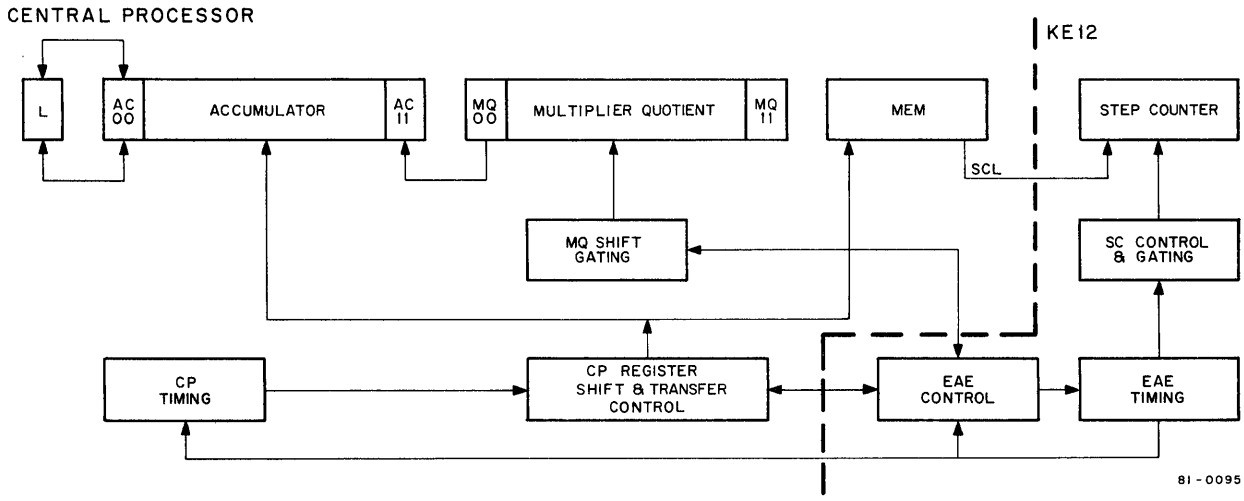


Figure 7-2 Simplified Block Diagram, KE12 EAE

7.4.3 Step Counter Register (SC)

The EAE SC is a 5-bit register loaded with the complement of the contents of MB bits 7 through 11 for the ASR, LSR, SCL, and SHL instruction, and is set to contain all 0s for DVI, MUY, and NMI instructions. It is used to record the number of shifts performed, and stops the shifting process after the correct number of shifts.

7.4.4 EAE Instructions

The instructions for the EAE are a class of GROUP 3 Microinstructions containing binary 1s in MB bits 00 through 03 (hence, code 7nnn) and bit 11, indicating the EAE group. The setting of these bits distinguishes the EAE instruction from all other instructions; all EAE except NMI are two-word instructions. These instructions can be subdivided further into two basic groups: the load group and the EAE operation group (instructions used when EAE timing is started and the CP is in the T5 RECYCLE state). The EAE microinstructions are augmented microprogrammable instructions. They can, therefore, be combined to perform non-conflicting logical operations.

7.4.5 Load Group

Clear the Accumulator (CLA) – The CLA instruction (7601) clears the AC during logical sequence 1; therefore, this instruction can be microprogrammed with other EAE instructions that load the AC during logical sequence 2, such as SCA or MQA.

Clear the Accumulator and Multiplier Quotient (CAM) – The CAM instruction (7621) clears the AC during logical sequence 1, as in CLA; during logical sequence 2 the MQ is cleared by enabling MB bit 07.

0 => AC, 0 => MQ

Multiplier Quotient Load into Accumulator (MQA) – The MQA instruction (7501) loads the contents of the MQ into the AC. This command is given to load the 12 least significant bits of the product into the AC after a multiplication, or to load the quotient into the AC after a division. The AC should be cleared prior to issuing this command. The CLA instruction can be combined with the MQA to clear the AC and then load the MQ into the AC. If the AC is not cleared prior to the MQA command, the contents of the MQ are inclusively ORed with the contents of the AC.

MQ V AC => AC

Load Multiplier Quotient (MQL) – The MQL instruction (7421) clears the MQ at logical sequence 1 and loads the contents of the AC into the MQ at logical sequence 2; then the AC is cleared. AC to MQ ENABLE is developed by a four-input NAND gate, as shown on Drawing D-BS-EP12-0-RCS. AC to MQ ENABLE causes the AC bits to be loaded into the data inputs of the MQ flip-flops. RCL MQ LOAD is developed in the same manner as in the CAM instruction during sequence 2.

0 => MQ, AC => MQ, 0 => AC

Step Counter Load into Accumulator (SCA) – The SCA instruction (7441) loads the contents of the SC into the AC. The AC should be cleared prior to this instruction, or the CLA instruction may be combined with the SCA to clear the AC during logical sequence 1. The transfer SC to AC occurs during logical sequence 2; both instructions, therefore, can be combined. RCL AC LOAD is developed in this instruction exactly as in the MQA instruction.

SC V AC => AC

7.4.6 EAE Operate Instructions

There are five EAE instructions that set CPT T5 RECYCLE: ASR, LSR, SHL, MUY, and DVI. Instructions that require T5 RECYCLE are referred to as EAE timing in this section. These instructions are coded in bits 08 through 10 of the instruction word; this coding transfers timing from the CP to EAE timing at TP3 of the FETCH cycle, with the operation being performed during the EXECUTE cycle.

Arithmetic Shift Right (ASR) – The ASR instruction (7415_8) causes the combined contents of the AC and MQ to shift right one more position than the number contained in the next sequential core memory location. The following occurs:

1. The contents of MQ11 are lost.
2. AC00 is loaded into the L.
3. AC00 is loaded into AC00.

During the FETCH cycle, the contents of MB bits 08 through 11 are loaded into the N counter (IR bits 08 through 11), Link is zeroed, the contents of AC are added directly back into AC, and the EXECUTE flip-flop is set. During the EXECUTE cycle, the contents of MB bits 08 through 11 are complemented and loaded into the step counter (SC). EAE timing is then started.

Logical Shift Right (LSR) – The LSR instruction (7417_8) causes the combined contents of the AC and the MQ to shift right one position more than the number in the next sequential core memory location. The following occurs:

1. MQ11 is lost.
2. L is loaded with a zero.

During the FETCH cycle, the contents of MB bits 08 through 11 are loaded into the N register, the L is zeroed, the contents of the AC are loaded directly back into the AC, and the EXECUTE flip-flop is set. The contents of MB 08 through 11 are complemented and loaded into the SC during the EXECUTE cycle. Then EAE timing is started, and CP timing is suspended in T5 RECYCLE. The contents of the AC and MQ are then shifted the number of times to the right as specified by the SC.

Shift Left (SHL) – The SHL instruction (7413_8) causes the combined contents of the AC and the MQ to shift left one position more than the number in the next sequential core memory location after the instruction. The following occurs for each shift:

1. The content of the L is lost,
2. The content of AC00 is loaded into the L,
3. All other AC bits move one place left,
4. MQ00 is loaded into AC11; all other MQ bits move one place left, and
5. MQ11 is loaded with a 0.

During the FETCH cycle, the contents of MB08 through 10 are decoded in the N-register, the contents of the PC are loaded into the MQ, and the EXECUTE flip-flop is set. The contents of MB07 through 11 are complemented and loaded into the SC during the EXECUTE cycle. Then the T5 RECYCLE is set and EAE timing is started. The contents of the AC and MQ are then shifted left the required number of times.

The logic is the same as for the ASR and LSR instructions during the FETCH cycle. During the EXECUTE cycle, the logic is the same as the ASR and LSR until the EAE timing is started.

Multiply (MUY) – The MUY instruction (7405_8) multiplies the number in the MQ by the number held in the next successive core memory location after the MUY instruction. At the end of the instruction, the twelve most significant bits of the product are contained in the AC and the twelve least significant bits of the product are contained in the MQ. During the FETCH cycle, the usual operations necessary for an EAE instruction are performed. During the EXECUTE cycle for MUY, the SC is cleared to receive the number of Multiply operations. For each operation, the SC is incremented. The SC does not have to count more than 11_{10}.

Before discussing the actual logic of the MUY instruction, some discussion of binary multiplication as performed by the EAE is necessary. To multiply two numbers together (12_{10} times 12_{10}) with the EAE, the following operations are necessary. These numbers in binary form are:

$001\ 100_2$.

Step 1		
	001 100	
	x001 100	
	000 000	Partial Product
Step 2	001 100	
	x001 100	
	000 000	(from Step 1)
	0000 00	
	0000 000	Second Partial Product

(continued on next page)

Step 3

```
   001 100
  x001 100
  --------
   000 000
  0000 00          (from Step 2)
 00110 0
 -------
 00110 000         Third Partial Product
```

Step 4

```
   001 100
  x001 100
  --------
   000 000
  0000 00
 00110 0           (from Step 3)
001100
------
010010 000         Fourth Partial Product
```

Steps 5 and 6

```
   001 100
  x001 100
  --------
   000 000         (from Step 1)
  0000 00          (from Step 2)
 00110 0           (from Step 3)
001100             (from Step 4)
000000             (from Step 5)
000000             (from Step 6)
----------
00010010 000       Final Product (144_10)
```

In Step 1, the least significant bit in the multiplier is multiplied by the multiplicand and forms a partial product. Because binary numbers (0s and 1s) are being used by the computer, some definite rules can be established. When the least significant bit in the multiplier is a 1, the multiplicand is added to the partial product. When the least significant bit in the multiplier is a 0, then 0s are added to the partial product. The number of operations necessary to perform this multiplication is six. When the same multiplication is performed using 12-bit numbers, the number of operations necessary is twelve. Because the SC (D-BS-KE12-0-EAES) must increment one time less than the number of operations performed, the SC reaches 12 to terminate EAE timing. Even when multiplying the largest binary numbers that can be expressed in 12-bits, the number of operations is the same, and the number of bits in the product does not exceed 24. When an addition of either 0s or the multiplicand is made to the partial product, the least significant bit does not change. To determine whether to add 0s or to add the multiplicand to the partial product the least significant bit of the multiplier is examined; if it contains a 1, the signal EAE MB ENABLE H must be generated. If it contains a 0, then EAE MB ENABLE must be inhibited, allowing 0 to be added to the contents of the AC. The EAE actually shifts the contents of the AC and MQ to the right to add 0s. The least significant bit (LSB) is examined, and if it is a 0, then the L, AC, and MQ are shifted to the right. If the LSB is a 1, then the MB is added to the AC, and the L, AC, and MQ are shifted to the right.

Divide (DVI) – The DVI instruction (7407_8) divides the 24-bit dividend contained in the AC (12 most significant bits) and the MQ (12 least significant bits) by the number located in the next sequential core location after the DVI instruction.

The simplest Divide algorithm, although not the one used in the KE12, is that of subtraction test, subtractions, and shifts. Assume that a double-precision number has been loaded into the AC and MQ, and that the dividend is present in the MEM register as read from memory. A test subtraction is taken (for example, MEM + AC is placed on the register bus). If an Overflow results, the subtraction cannot be made without causing a change in sign. The entire AC and MQ are shifted left without actually executing the subtraction, and a 0 is shifted into the MQ11 to form a part of the quotient. If Overflow does not result, the subtraction is performed before, or as a part of the shift, and a 1 is loaded into the MQ11. The process is very similar to longhand decimal division.

In the KE12, the algorithm used is slightly more complicated, although faster. A subtraction and shift is performed. A Flow Chart for the DVI instruction is shown in Figures 7-3 and 7-4. If Overflow does not result (indicated by FLK ADDER LINK), a 1 is loaded in MQ11, and the next arithmetic operation is an Add and Shift. The adding process continues until another Overflow is encountered, and 0s are loaded into MQ11 for each addition. Overflow indicates that the partial remainder is now positive, and that the arithmetic operation should become a subtraction. At the conclusion of the Divide operation (SC = 13), the remainder (now in the AC) may have to be corrected, and the LINK cleared. The correction depends on the last two bits in the quotient. If the last two bits are MQ10 = 0 and MQ11 = 1, the remainder is correct as is. If the last two bits are both 1s, the remainder must be complemented. Other combinations of MQ10 and MQ11 result in an extra addition or subtraction.

The first subtraction in any Divide must produce a negative result; otherwise, a condition known as Divide Overflow exists. If Divide Overflow occurs (division by 0 is a classical example), the resulting quotient cannot be contained in 12 bits, and the EAE signifies this by immediately exiting from the division with the Link set.

Normalize (NMI) – The NMI instruction (7411) is used, in part, to convert a binary number to a fraction (and its exponent) for use in floating point arithmetic. The AC and MQ are treated as one long register and their contents are shifted left, by this command, until the content of AC00 is not equal to AC01.

When this instruction is completed, the SC contains a number equal to the number of shifts performed. The contents of the L are lost. During the NMI instruction, EAE timing is started, but the CP stays in a FETCH cycle and the T5 RECYCLE is set. The NMI is the only EAE instruction that starts EAE timing but does not require an EXECUTE cycle. During logical sequence 1, MB bits 08, 09, and 10 are loaded into the N register and the SC is cleared. A normalize test tree (D-FD-KE12-0-2) checks the number contained in the AC and the MQ to determine if it is already in the normalized condition.

Step Counter Load from Memory (SCL) – The SCL instruction (7403) requires two sequential memory locations, one containing the number to be loaded into the SC. This instruction loads the complement of the memory word bits 07 through 11 (the word located in the next sequential memory address) into the SC. This instruction is a two-cycle instruction; it goes into an EXECUTE cycle but does not start EAE timing. During the FETCH cycle, the MA is incremented by one and loaded into the PC; then the memory word is loaded into the MB and IR. The contents of the PC are loaded into the MA and the EXECUTE state is entered. During the EXECUTE cycle, the contents of the memory word (next MA) are complemented and loaded into the SC.

7.5 KF12B MULTI-LEVEL AUTOMATIC PRIORITY INTERRUPT

The KF12B is designed to reduce the central processor overhead during the servicing of program interrupts. It is prewired in the EP section of the PDP-12 in racks P and R, and utilizes approximately 55 M-series modules.

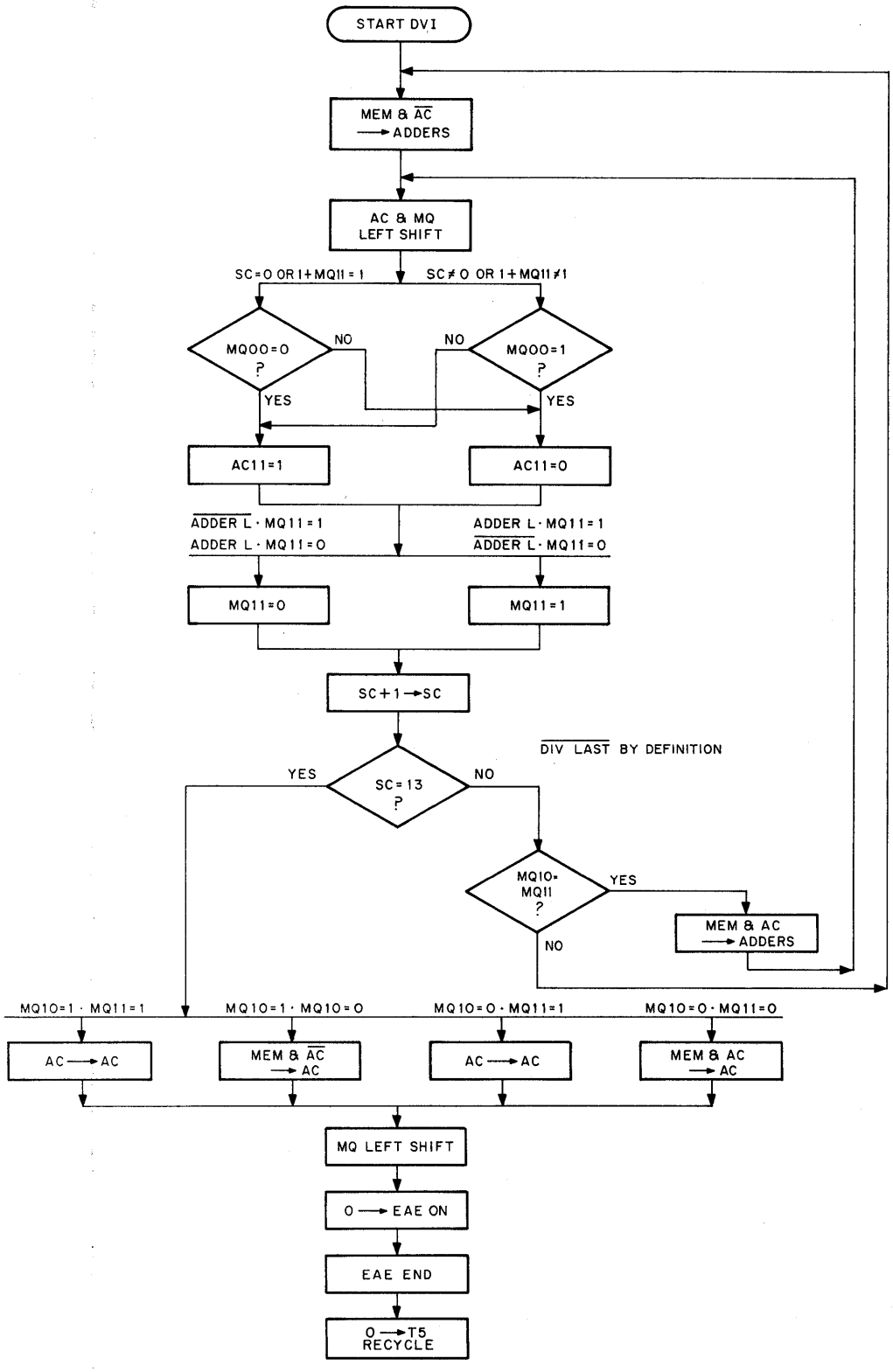


Figure 7-3 KE12 EAE DVI Flow Chart

Link	Carry	Accumulator	Multiplier-Quotient	Memory	Step Counter	Comments
0		000 000 000 000	000 010 010 001	000 000 001 100		
		111 111 111 111				AC ENABLE
		000 000 001 100				MEM ENABLE
	1	000 000 001 011				Output of ADDERS
0		000 000 010 111	000 100 100 010	000 000 001 100	00 001	AC and MQ LEFT SHIFT
						End of 1st operation
0		111 111 101 000				AC ENABLE
		000 000 001 100				MEM ENABLE
	0	111 111 110 100				Output of ADDERS
1		111 111 101 000	001 001 000 100	000 000 001 100	00 010	AC and MQ LEFT SHIFT
						End of 2nd operation
		111 111 101 000				AC ENABLE
		000 000 001 100				MEM ENABLE
	0	111 111 110 100				Output of ADDERS
1		111 111 101 000	010 010 001 000	000 000 001 100	00 011	AC and MQ LEFT SHIFT
						End of 3rd operation
		111 111 101 000				AC ENABLE
		000 000 001 100				MEM ENABLE
	0	111 111 110 100				Output of ADDERS
1		111 111 101 000	100 100 010 000	000 000 001 100	00 100	AC and MQ LEFT SHIFT
						End of 4th operation
		111 111 101 000				AC ENABLE
		000 000 001 100				MEM ENABLE
	0	111 111 110 100				Output of ADDERS
1		111 111 101 001	001 000 100 000	000 000 001 100	00 101	AC and MQ LEFT SHIFT
						End of 5th operation
		111 111 101 001				AC ENABLE
		000 000 001 100				MEM ENABLE
	0	111 111 110 101				Output of ADDERS
1		111 111 101 010	010 001 000 000	000 000 001 100	00 110	AC and MQ LEFT SHIFT
						End of 6th operation
		111 111 101 010				AC ENABLE
		000 000 001 100				MEM ENABLE
	0	111 111 110 110				Output of ADDERS
1		111 111 101 100	100 010 000 000	000 000 001 100	00 111	AC and MQ LEFT SHIFT
						End of 7th operation
		111 111 101 100				AC ENABLE
		000 000 001 100				MEM ENABLE
	0	111 111 111 000				Output of ADDERS
1		111 111 110 001	000 100 000 000	000 100 001 100	01 000	AC and MQ LEFT SHIFT
						End of 8th operation
		111 111 110 001				AC ENABLE
		000 000 001 100				MEM ENABLE
	0	111 111 111 101				Output of ADDERS
1		111 111 111 010	001 000 000 000	000 000 001 100	01 001	AC and MQ LEFT SHIFT
						End of 9th Operation
		111 111 111 010				AC ENABLE
		000 000 001 100				MEM ENABLE
	1	000 000 000 110				Output of ADDERS
0		000 000 001 100	010 000 000 001	000 000 001 100	01 010	AC and MQ LEFT SHIFT
						End of 10th operation
		111 111 110 011				AC ENABLE
		000 000 001 100				MEM ENABLE
	0	111 111 111 111				Output of ADDERS
1		111 111 111 111	100 000 000 011	000 000 001 100	01 011	AC and MQ LEFT SHIFT
						End of 11th operation
		111 111 111 111				AC ENABLE
		000 000 001 100				MEM ENABLE
	1	000 000 001 011				Output of ADDERS
0		000 000 010 110	000 000 000 110	000 000 001 100	01 100	AC and MQ LEFT SHIFT
						End of 12th operation
		111 111 101 001				AC ENABLE
		000 000 001 100				MEM ENABLE
	0	111 111 110 101				Output of ADDERS
1		111 111 110 101	000 000 001 100	000 000 001 100	01 101	MQ LEFT SHIFT ONLY
						End of 13th operation
		111 111 110 101				Since SC=13 and MQ10=0
		000 000 001 100				and MQ11=0 MEM and AC→AC.
	1	000 000 000 001				
0		000 000 000 001	000 000 001 100	000 000 001 100	01 101	NO SHIFT

Figure 7-4 KE12 EAE DVI Algorithm

There are three major services provided by the KF12B:

a. Automatic determination of device priority and vectoring of interrupt service routines.

b. Automatic saving and restoring of all major registers and machine status, which include the following: PC, AC, IF, DF, MQ, LINK, FLOW, UF, MODE, and the current processor level.

c. Automatic stacking of the saved parameters, permitting multiple levels of interrupts.

7.5.1 Block Diagram Description

When a new, higher interrupt level is asserted, the storing or stacking of parameters is called *pushing* and restoring the CP to its previous state prior to the interrupt is called *popping.* The KF12B can accommodate up to 15 levels of interrupts, which come in on the external level (EXT LEV 0–14) lines (see Figure 7-5). The interrupts can be accepted from any of nine prewired options and from up to six external devices. In addition to these 15 interrupt levels, the processor itself constitutes the sixteenth level. Octal 17 is the processor (CP) level and has the lowest priority; the octal 0 level has the highest priority.

When an interrupt request is received from a device on one of the external lines, a corresponding level (PA1 LEV) flip-flop is set. The outputs of the 15 level flip-flops are fed to an encoder circuit. The encoder circuit outputs are gated to form trial levels (PA2 TRIAL 0–2), the configuration of which designates the highest requesting interrupt level. The present machine priority level was stored in the KF12B circuits when the previous priority request was processed. This present machine priority level is subtracted from the complement of the level designated by the trial levels, and if the result is negative, priority is granted to the new device. In other words, if the new interrupt level minus the present interrupt level is less than zero, the interrupt is granted. Example: Present level = 5, new level = 4; 4 -5 = -1 which is less than zero; therefore the interrupt is granted.

When an interrupt request is granted, the PA2 PRIORITY OK level becomes true, and after gating with other signals, results in setting the push (CON PUSH) flip-flop. The setting of the push flip-flop causes the API break request (CON API BRK REQ) flip-flop to set, and an output from this flip-flop is gated with signals that indicate the status of the CP. If the CP is not in a critical state (i.e., INTERRUPT INHIBIT (0); SAVE PC (1); and not DJR, DIF, or LIF), the API request OK (CON API REQ OK) level is true.

In the Multiplexer Control section (see Figure 7-5) the API request OK level sets the MUX API flip-flop, which indicates that the KF12B is requesting a break cycle as a result of an interrupt. The Multiplexer Control for the KF12B can handle three data break options when the DM12 Data Break Multiplexer Option to the KF12B is included. The DM12 is prewired in the PDP-12, and the KF12B is a prerequisite for the DM12 because both use the same timing and control signals, which originate in the KF12B. The three external break request lines (EXT 0 BREAK RQST through EXT 2 BREAK RQST) each service one data break device; line 0 has the highest priority and line 2 the lowest priority. All three lines have higher priority than the KF12B, so that an API break request is granted only when there are no requests on the external break request lines.

When a break request is granted to the KF12B, the MUX priority API (MUX PRI API) signal becomes true. This signal enables the stack address in core memory and the controlling signals for the API.

When a break request is granted to one of the three data break devices, a similar signal, MUX priority (MUX PRI) becomes true. This signal later qualifies the MUX enable break (MUX EN BRK) levels, which gate the B BREAK and WC OVERFLOW signals to the device that is granted control of the bus.

The MUX priority API signal is gated with another signal to form the MUX enable stack address (MUX I ENABLE STACK ADD) signal, which gates the outputs of stack register to the external data address bus. The stack register contains the starting address for storage of the parameters (of the present machine status) that are to be saved. The stack address can be specified by the program by use of the SMLV (IOT 6772) and SSTK (IOT 6776) instructions. Stack bits 0, 1, and 2 specify the memory field and are loaded with AC bits 0, 1, and 2 by the SMLV

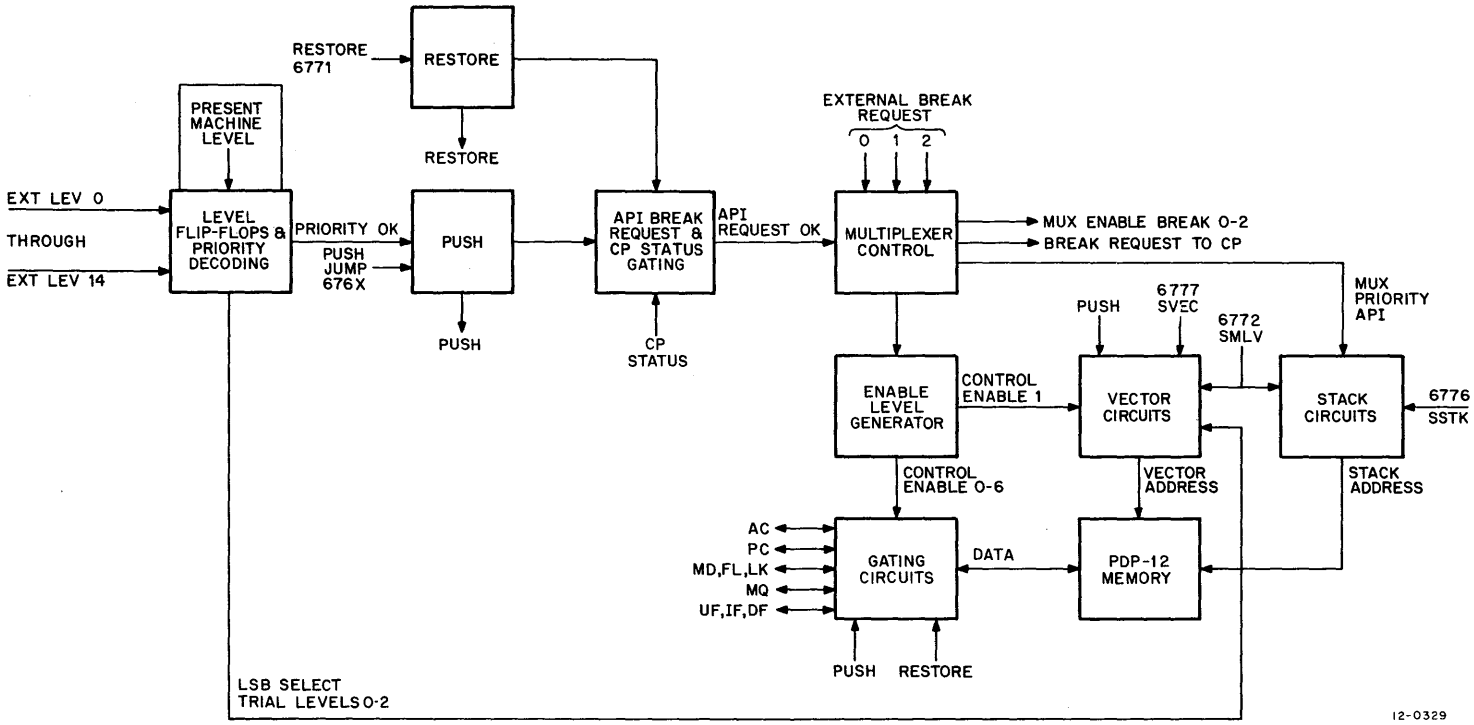


Figure 7-5 KF12B Functional Block Diagram

instruction. Stack bits 3 through 14 specify the 12-bit memory address within the field and are loaded with AC bits 0 through 11 by the SSTK instruction.

The MUX priority API signal from the Multiplexer Control section also enables the API ADD ACC signal, which results in the setting of the break done (CON BRK DONE) and control advance (CON ADV) flip-flops; a CON CLOCK PULSE is then sent to the Enable Level Generator.

The parameters of the present machine status are stored in five consecutive memory locations by five consecutive memory cycles. For each cycle the stack register is incremented by one; thus the address is increased to the next location. Also, for each memory cycle, the Enable Level Generator is incremented by another CON CLOCK PULSE, and thus different control enable levels (CON ENAB 0–6) are true for each of the five memory cycles. These control enable levels control the gating circuits that gate the various active registers and status levels into the memory buffer for storage in the memory.

Each of the 15 API interrupt levels has a vector address that specifies the starting memory address of the service routine for the device on that interrupt level. This vector address is transferred to the PC during the second memory cycle (CON ENAB 1) of the push operation; thus the CP can go into the service routine when storage of the parameters of the present machine level is completed. Vector bits 0, 1, and 2 specify the memory field and can be loaded with AC bits 3, 4, and 5 by the SMLV instruction (IOT 6772). Vector bits 3 through 9 specify memory address bits 0 through 6 and can be loaded with AC bits 0 through 6 by the SVEC (IOT 6777) instruction. The API interrupt level specifies memory address bits 7 through 10 by the trial levels (PA2 TRIAL 0–2) and the LSB select (PA1 LSB SEL) level. A vector address is always an even numbered address so that each interrupt level can have a two word vector address. Once the CP starts on the service routine, interrupts of a higher priority can take control after the execution of the first instruction in the present interrupt service routine.

Every interrupt service routine should terminate with a Restore (IOT 6771) command. This command restores the major registers and machine status by popping this information from the stack, and programming is resumed where it left off at the beginning of the interrupt. When a Restore instruction is received, the restore (CON RESTORE) flip-flop is set and the KF12B circuits operate in a manner similar to the push operation. If the CP status is OK and no requests are coming in on the external break request lines, the control enable levels are generated and the last used stack pointer provides the memory address for the first parameter to be restored. The control enable levels gate the parameters from memory back to the CP registers, in the reverse order in which they were pushed onto the stack. The stack register is decremented by one for each of the five memory cycles of the restore operation; thus, the memory address is decreased by one each time to restore the next parameter. When the restore operation is completed, the address of the next program step after the occurrence of the interrupt is in the PC, and the CP proceeds with the program that was interrupted.

A two-word instruction called push jump (PUSHJ address, IOT 676X) is featured in the KF12B. This instruction permits jumping to subroutines across memory field boundaries in both LINC and PDP-8 modes. The push flip-flop is set directly without going through the priority decoding logic. The active registers and machine status are pushed onto the stack and the CP automatically jumps to the memory location specified by the 15-bit address associated with the instruction. The X in the instruction code defines the new memory field, and the location following the instruction specifies the 12-bit memory address of the subroutine. The PC address stored on the stack during the execution of the PUSHJ instruction points to the location following the two-word PUSHJ instruction; thus the program is resumed at this location following the restore operation.

The logic circuits for the KF12B are shown on 14 block schematics (the D-BS-KF12-0-XXX drawings) that appear in the *PDP-12 Maintenance Manual, Volume III, System Drawings.* Also in Volume III are four flow and block diagrams (the D-FD-KF12-0-XXX drawings) that depict the general functional operation of the KF12B option in conjunction with the PDP-12 system.

7.5.2 Priority Levels and Priority Decoding

The priority logic is shown on engineering drawings D-BS-KF12-0-PA1 and -PA2. The 15 EXT LEV lines are connected to the PA1 LEV flip-flops through the M905 Jumper Module in location R16. The interrupt priority order for the prewired options is determined by the jumper connections that are made on the M905 Module. For the external options, the interrupt priority order is determined by the jumper connections and by the M905 terminal to which the option is connected. The PA1 LEV flip-flops are clocked by the CON CLOCK PRIORITIES pulse, which occurs at TP2. The outputs of the 15 flip-flops are connected to two encoder ICs (M192 Module in location P39) that determine the interrupt request with the highest priority. Only one encoder is active at a time due to the enable output (E OUT) level. When the enable output level is low, all the other outputs from the encoder are high or disqualified (refer to Table 7-5). This signal is used to enable the other encoder, which handles the next less significant 8 levels of priority.

Table 7-5
Priority Encoder Truth Table

E IN	0	1	2	3	4	5	6	7	GS	A_0	A_1	A_2	E OUT
H	X	X	X	X	X	X	X	X	H	H	H	H	H
L	H	H	H	H	H	H	H	H	H	H	H	H	L
L	X	X	X	X	X	X	X	L	L	L	L	L	H
L	X	X	X	X	X	X	L	H	L	H	L	L	H
L	X	X	X	X	X	L	H	H	L	L	H	L	H
L	X	X	X	X	L	H	H	H	L	H	H	L	H
L	X	X	X	L	H	H	H	H	L	L	L	H	H
L	X	X	L	H	H	H	H	H	L	H	L	H	H
L	H	L	H	H	H	H	H	H	L	L	H	H	H
L	L	H	H	H	H	H	H	H	L	H	H	H	H

H = high voltage level L = low voltage level X = irrelevant

The outputs of the encoder ICs are gated (PA2 drawing) to generate three trial levels (PA2 TRIAL 0–2). These three trial levels and the encoder output PA1 LSB SEL specify one of 16 possible interrupt levels and also specify bits 7 through 10 of the vector address (D-BS-KF12-0-PUSH). The four levels that define the interrupt priority level are compared with the present machine level (PA2 PRES LEV 0–3) by the M159 4-bit Arithmetic Unit in location P35 (PA2 drawing). If the trial level is less than the present machine level (higher priority), the signal PA2 PRIORITY OK is generated by the 4-bit arithmetic unit.

The present machine level was defined by the four outputs (PA2 PRES 0–3) from the M238 Up/Down Counter in location R36 (PA2 drawing). The three trial levels PA2 TRIAL 0 through 2 and encoder output PA1 TRIAL 3 are gated to provide inputs to the M238 Up/Down Counter; thus the interrupting level is stored as the present machine level for the next interrupt sequence. The M238 Up/Down Counter is loaded with the new machine level by the PA2 LOAD LEVEL pulse, which occurs at CON ENAB 5 of a push or a restore operation.

7.5.3 Control Circuits

The control logic is shown on engineering drawing D-BS-KF12-0-CON and includes the functions covered by the push and the API Break Request and CP Status Gating blocks on the functional block diagram (see Figure 7-5). The PA2 PRIORITY OK signal generated by the 4-bit arithmetic unit (PA2 drawing) is gated with other signals

including TP3 to generate the CON INTERRUPT (1) signal (CON drawing). This signal becomes the CON SET PUSH signal, which sets the CON PUSH flip-flop. The setting of the CON PUSH flip-flop causes the CON API BRK REQ flip-flop to set. Gating of the 1-side output of this flip-flop with signals that are true when the CP can accept a break request generates the CON API REQ OK signal.

7.5.4 Multiplexer Control

On the Multiplexer Control Drawing (D-BS-KF12-0-MUX) the CON API REQ OK signal sets the MUX API flip-flop. The Multiplexer Control with the DM12 option included can handle a total of three data break devices and the KF12B. The three data break request lines come in as the EXT BREAK RQST lines and each one goes to a MUX LEV flip-flop. The MUX LEV and MUX API flip-flops are clocked by the MUX CLOCK LEV pulse, and when a device requests a break, the corresponding flip-flop is set. The flip-flop outputs are decoded to determine priority; the KF12B has the lowest priority on the bus and device 0 has the highest priority. The MUX PRI levels enable the data address on the bus and allow the corresponding MUX EN BRK flip-flop to be set at TP1. These flip-flops gate the B BREAK signal to the device that has control of the bus.

When a KF12B break is granted, the MUX PRI API level (MUX drawing) becomes true. This signal is used to gate the MUX API ADD ACC (1) signal as shown on the MUX 1 engineering drawing. The trailing edge of the MUX API ADD ACC pulse sets the CON BRK DONE flip-flop (CON drawing), which causes the CON ADV flip-flop to be set by the trailing edge of the CON CLOCK pulse. The M401 Variable Clock generates CON CLOCK pulses every 200 ns (5 mHz rate). The CON CLOCK and the 1-side output of the CON ADV flip-flop are gated together to generate the CON CLOCK PULSE. The CON ADV and the CON BRK DONE flip-flops are cleared and the CON CLOCK PULSE increments the 4-bit up/down counter of the enable level generator (CON drawing). Only one CON CLOCK PULSE is generated per API break cycle, except during the second break cycle of an interrupt-push. In this case the CON ENAB 2 level is not used and the counter is incremented twice. Figure 7-6 is a timing diagram for the KF12B control circuits.

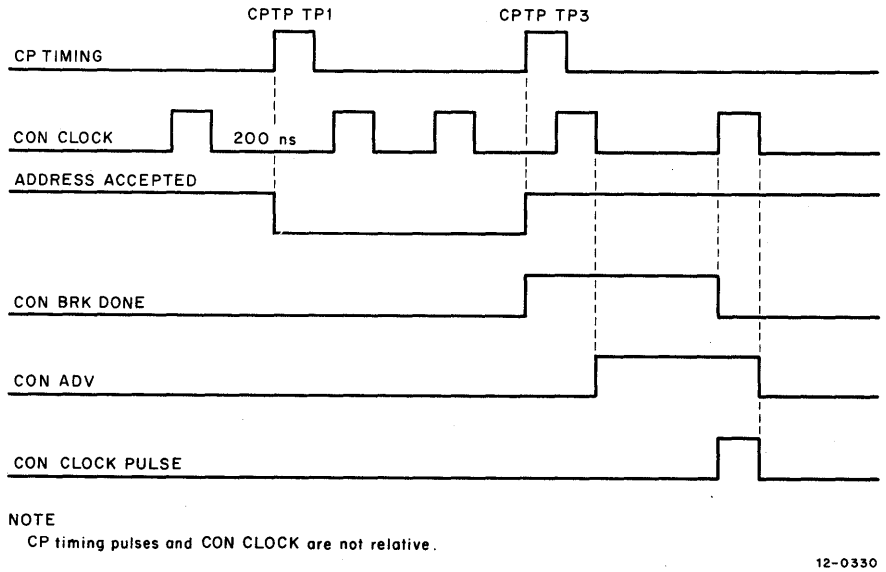


Figure 7-6 Control Timing

7.5.5 Enable Level Generator

To perform either a push or a restore operation, the KF12B requires five consecutive memory cycles. For each of these five cycles, a CON CLOCK PULSE is generated to increment the 4-bit up/down counter; thus a different enable level is generated for each of the five memory cycles. Drawing D-FD-KF12-0-MRG shows the enable levels, in combination with push or restore signals, that are used to gate the various machine parameters in and out of memory. The detailed logic that comprises the gating circuits is shown in the PRG, PUSH, RES, SRF, and SSB block schematic drawings in the engineering drawing set.

7.5.6 Stack Circuits

The stack register consists of four M238 ICs shown on engineering drawing D-BS-KF12-0-PTRS. The register is set up with the correct starting stack address by program commands that read AC bits into the register to designate memory field and memory address. Signals PTRS INC STACK PTR and PTRS DEC STACK PTR automatically increment and decrement the address contained in the stack register for push and restore operations. The stack register outputs PTRS STACK 00 (1) through 14 (1) are routed to the word A and word B ICs shown on the D-BS-KF12-0-I01 drawing. These circuits read the stack address onto the external I/O bus by program command. The stack register outputs are also routed to the group of gates shown on drawing D-BS-KF12-0-MUX1. This group of gates is enabled by the MUX 1 ENABLE STACK ADD signal, which is formed by the gating of the MUX PRI API signal and a signal that inhibits the gate during a push jump operation. Therefore, when the Multiplex Control generates the MUX PRI API signal and it is not a push jump operation, these gates read the contents of the stack register onto the external data address lines, which route the data to the MB and PC registers. The memory address is now set up for storage or read-out of the next machine parameter.

7.5.7 Vector Circuits

The vector memory field bits (vector bits 0, 1, and 2) and vector bits 3 through 9 are designated by program commands and stored in three M238 ICs shown on engineering drawing D-BS-KF12-0-PTRS. The outputs of these ICs (PTRS VECTOR 00 (1) through 09 (1)) are routed to the word A and the word C ICs shown on drawing D-BS-KF12-0-I01. The PA2 TRIAL levels and the PA1 LSB SEL signal are also used as inputs to the word C IC for vector bits 7 through 10. These circuits read the vector address onto the external I/O bus by program command.

The PTRS VECTOR 00 (1) through 02 (1) signals that designate the memory field are inverted and routed to gates shown on the D-BS-KF12-0-SRF drawing. The PUSH EN VECTOR signal gates these bits to the memory field register. The PTRS VECTOR 03 (1) through 09 (1) signals are routed to a group of gates shown on drawing D-BS-KF12-0-PUSH. The PA2 TRIAL levels and the PA1 LSB SEL signal provide inputs to the last four gates of the group. Vector bits 03 through 09, PA1 LSB SEL, and PA2 TRIAL levels 0 through 2 are gated onto the external data bus by signal PUSH VECTOR TO PC.

The CON PUSH (1), CON ENAB 1, and CPT TS1 signals are gated to generate the PUSH EN VECTOR signal, which is used to gate the memory field bits to the memory field register. The PUSH EN VECTOR signal is then gated with CON ADV and a disqualifying signal for a push jump operation to generate the PUSH VECTOR TO PC signal. Thus the vector address is gated to the external data bus during the second memory cycle of a push operation (CON ENAB 1) after the previous PC has been stored on the stack. The PRG LOAD PC signal then clocks the vector address from the external data bus to the PC and the vector address of the subroutine to service the interrupting device now in the PC.

7.5.8 Maintenance Logic

The KF12B includes maintenance logic, shown on drawing D-BS-KF12-0-MAIN, to provide for checking of the major portion of the option logic for proper operation. Two IOT instructions are used to simulate the 15 interrupt levels, to check the priority logic, and to initiate a push operation. The simulated levels remain set for only one computer cycle, which allows enough time for a push operation when the selected level has priority and the API is enabled.

7.6 KP12 POWER FAILURE RESTART

A M703 Power Fail Module located at the memory section of the Central Processor location F03, is utilized for this option, which protects an operating program in the event of power failure. When a power failure occurs, an Interrupt Flag is decoded as shown on drawing D-BS-KP12-0-PWF. When the power goes below the specified operative level, the MCT SHUT DOWN signal goes low and sets the PWF PWR LOW flip-flop. The zero side of the flip-flop is gated onto the common interrupt bus in the CP, which is enabled at this time. The PWF STOP OK signal goes low for 1 ms and is connected to the G826 Voltage Regulator to inhibit the regulator from shutting off for this 1 ms duration. This allows the subroutine time to store all the active registers and initialize the peripheral devices before halting the computer.

A toggle switch located on the M703 Module controls the automatic restart feature of the option. When the switch is in the ENABLE or on position, the following functions occur upon restoration of primary power. A 200 ms delay is initiated and at the end of the delay, the PWF RESTART pulse is generated. This pulse initiates the manual function timing chain which sets the RUN flip-flop and simulates a START key function. The RCL START PC pulse is generated and the PC is cleared allowing the program to start at address 0.

7.6.1 Programming

A skip circuit provides programmed sensing of the condition of the Power Low Flag by adding the following instructions to the computer's repertoire:

SPL Skip on Power Low

Octal Code:	6102
Event Time:	2
Execute Time:	4.25 μs
Operation:	The condition of the Power Low Flag is sampled; if set (indicating a power failure has occurred), the contents of the PC are incremented by one, skipping the next sequential instruction.
Symbol:	If Power Low Flag = 1, then PC + 1 → PC

Figures 7-7 and 7-8 illustrate the Automatic Restart Program Events and Typical Power Failure Program Service Routine, respectively.

7.7 KT12 TIME-SHARING OPTION

The KT12 Time-Sharing Option prewired in the processor section of the PDP-12 provides the additional logic circuits required for the TSS/12 Time-Sharing System. Certain configurations of I/O devices and other options must also be used with the TSS/12 Time-Sharing System. The minimum equipment required for a four-user Time-Sharing System is:

a. PDP-12 with KT12 Time-Sharing Option

b. MC12 Memory Extension Control and 8K of memory

c. RF08 Disk Control

d. RS08 Disk File

e. Asynchronous Serial Line Interfaces Full-Duplex, Dual Channel (four required) DC02E, DC02F, or PT0C

f. PR12 High-Speed Tape Reader – 300 characters per second

g. KE12 Extended Arithmetic Element

h. PDP-12 Option Cabinet

The TSS/12 Time-Sharing System permits up to 16 users to operate their individual programs in an apparently simultaneous manner. Operation and reaction time of some I/O devices and of human operators is slow compared to the speed of the CP; time-sharing allows the CP to proceed to other tasks rather than wait for slower

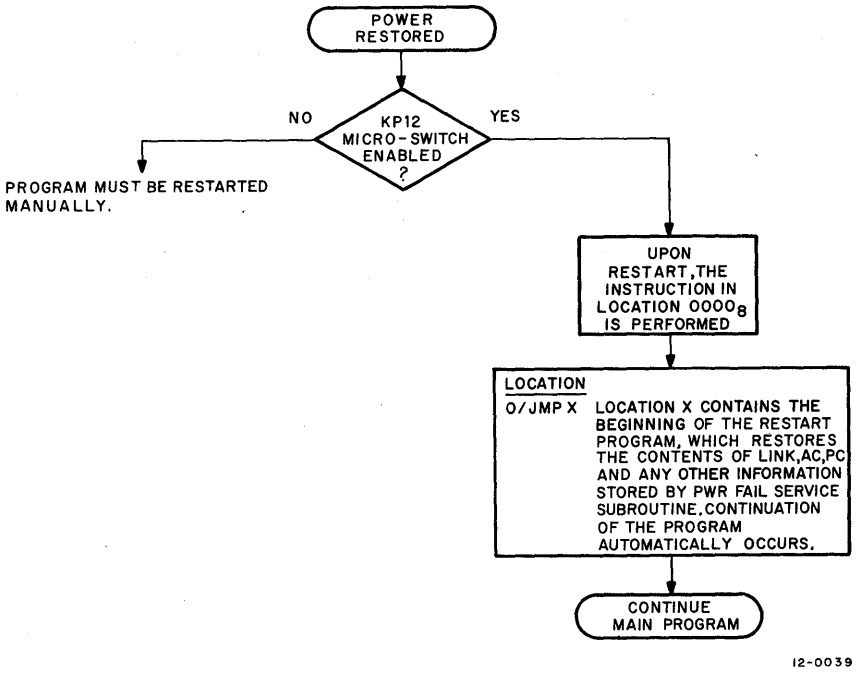


Figure 7-7 Typical Power Failure Program Service Routine

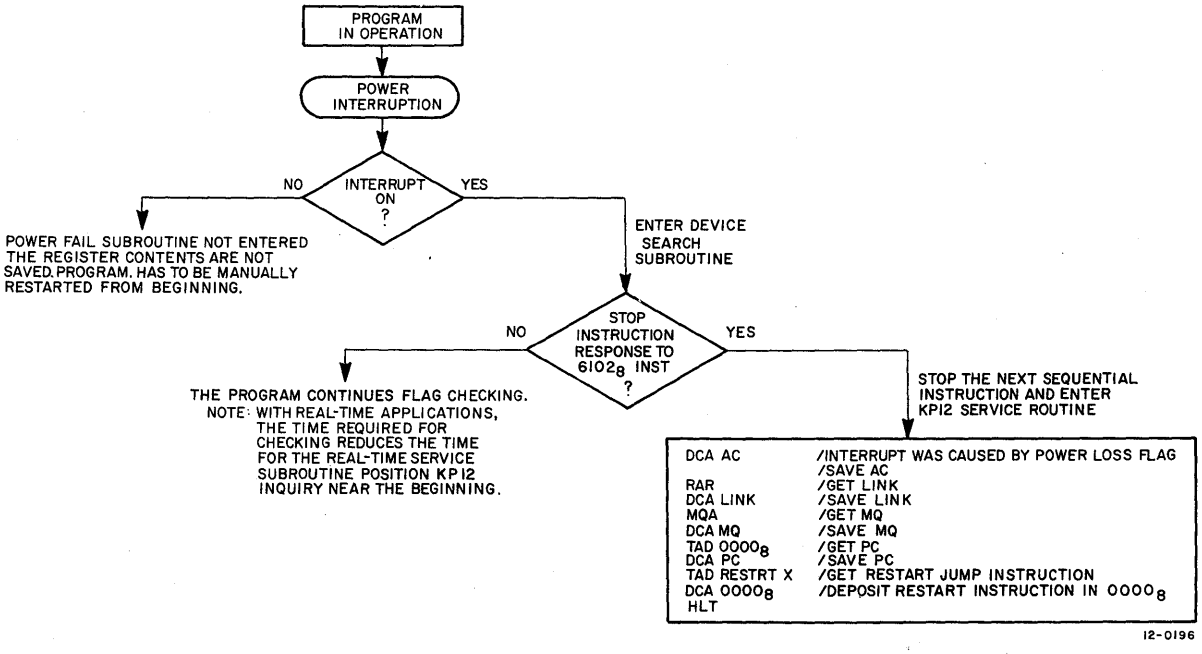


Figure 7-8 Automatic Restart Program Events

operations. As a result, it seems to each user as if each had full use of the processor. The program of each user is executed for only a fraction of a second at a time; the different programs are interspersed without interfering with each other and without noticeable delays in the response to each user.

7.7.1 TSS/Monitor Program

Time-sharing of the PDP-12 Central Processor by a group of users is controlled by a group of subprograms called TSS/12 Monitor, which coordinates the operation of various I/O devices, allocates CP time and services to the users, and controls user access to the CP. User programs are usually stored in disk memory and are transferred into core memory; activation of the user programs is handled by a sequential loop algorithm under the control of the TSS/Monitor. The user program is allowed to run for a fixed period of time and is then stopped. The contents of the PC and the various registers are stored at the time execution is stopped; the program is returned to disk storage; and the next user program is read into core memory for processing.

User programs are terminated by the TSS/12 Monitor for various reasons other than the expiration of their allotted time period. They are terminated when the output buffer is filled, when an input is requested and the input buffer is not filled, and under certain other conditions.

User programs are not allowed to perform HLT, OSR, or IOT instructions in the usual manner because normal processing of these instructions would disrupt the operation of the CP or interfere with the operation of I/O devices shared with other users. When one of these instructions appears in a user program, a User Interrupt (UINT) takes place and the TSS/Monitor takes control of the CP. Three instructions are added by the KT12 to permit the TSS/Monitor to process UINTs caused by HLT, OSR, or IOT instructions.

7.7.2 KT12 Program Instructions

The KT12 uses three additional instructions to permit the TSS/Monitor to handle UINTs and to control the UINT logic circuits. These instructions are listed with their octal codes and descriptions in Table 7-6.

Table 7-6
KT12 Program Instructions

Mnemonic Code	Octal Code	Description
CINT	6204	Clears User Interrupt. Resets the UINT flip-flop to 0 state.
SINT	6254	Skip on UINT. When the UINT flip-flop is in the 1 state, the PC is incremented by one and the program skips the next instruction.
SUF	6274	Sets the User Flag. Sets the User Buffer (UB) flip-flop and inhibits Processor Interrupts until after the next JMP or JMS instruction. The contents of the UB are transferred to the UF with the next JMP or JMS instruction.
SADC	6145	Sample A/D Channel **NOTE** **Processor Interrupts are inhibited between the SUF instruction and the loading of the UF flip-flop.**

The KT12 operates in two modes, as denoted by the UF flip-flop. When the UF flip-flop is in the logic 1 state, operation is in the user mode and a user program is executed in the CP. When the UF flip-flop is in the logic 0 state, operation is in the executive mode and the TSS/Monitor is in control of the CP. The three added instructions are used by the TSS/Monitor only in the executive mode, and are never used by the user program. If a user program attempted to use one of these instructions, execution would be blocked and a UINT would result, because these are IOT instructions (octal code 6XXX). The LINCtapes are used the same way as DECtape is used on the TSS8. Also, a new set of IOTs (8 Mode) has been implemented for time-share mode users.

The KT12 requires two additional modules. Additional gates from modules used with other prewired options are also required. The additional modules are shown in Table 7-7.

Table 7-7
KT12 Modules

Quantity	Module Type No.	Use	Location Row	Location Slot
1	M216	Flip-Flops	H	29
1	M117	NAND Gates	H	27
1	M121	NAND/NOR Gate	M	28*
1	M623	Bus Driver	M	21*
1	M111	Inverter	M	40*

* These modules are shared with other prewired options.

7.7.3 General Logic Description

The PDP-12 CP operates in the executive mode (UF(0) high) so that the TSS/12 Monitor can perform the necessary housekeeping and service routines, or in the user mode (UF(1) high) so that an individual user can perform its programmed tasks. Either mode of operation is subject to Program Interrupts that are handled by the Interrupt logic of the CP. When a user program is operating (user mode), the decoding of HLT, OSR, or IOT instructions causes an Interrupt, and the execution of these instructions is aborted. A simplified diagram of the logic circuits that generate an Interrupt in the user mode when an HLT, OSR, or IOT is programmed is shown in Figure 7-9.

When either an HLT, OSR, or IOT instruction is decoded in the CP Instruction Register (IR), and the UF flip-flop is set, the Int Req Bus level changes to a true state and the following functions are performed:

1. The UINT, and the INTERRUPT SYNC Save User Flag (SUF) flip-flops are set.
2. The CP enters the INTERRUPT state at the next CPTP1 pulse, the UF flip-flop is cleared at TS3 and the standard functions for the INTERRUPT cycle are executed.
3. The Int Req Bus level remains true until the CINT instruction is given; therefore, this instruction should be executed in the Interrupt handling subroutine.

IOT 6254 SINT checks the status of the UINT flip-flop. When this instruction is executed and the UINT flip-flop is set, the I/O SKIP BUS level becomes true, and the PC is incremented by skipping the next sequential instruction.

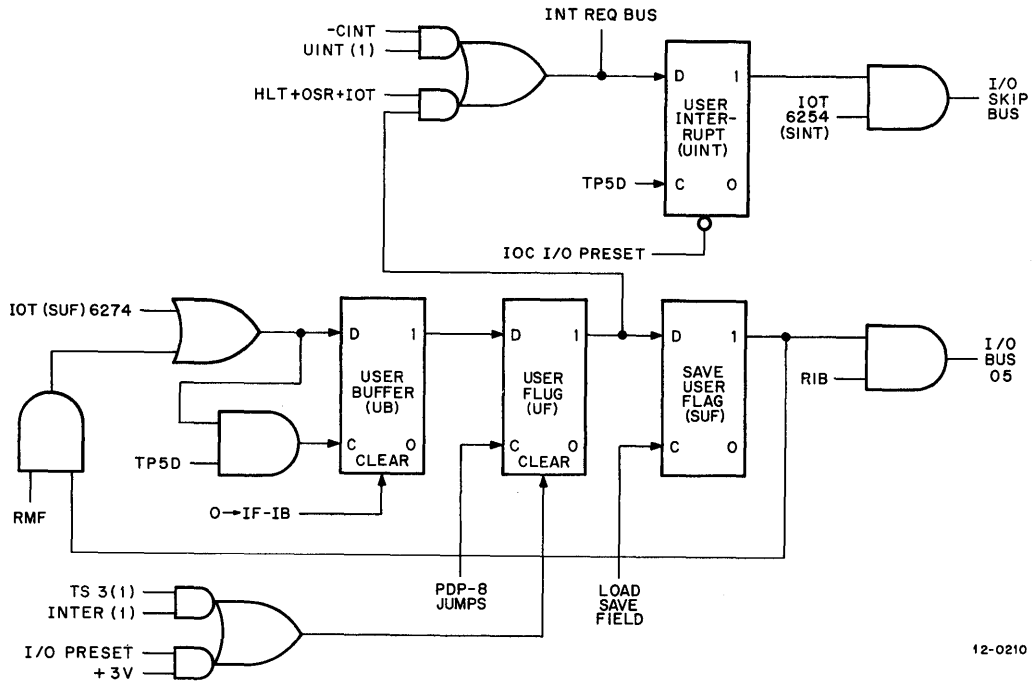


Figure 7-9 KT12 Simplified Block Diagram

7.7.4 Detailed Logic Description

Engineering drawing D-BS-KT12-0-TSS is a detailed block schematic of the logic circuits that are added to the PDP-12 for the KT12. The TSS User Buffer (UB), the TSS User Flag (UF), Save User Flag (SUF), and the UINT flip-flops with their associated gating circuits are shown on this diagram. The TSS UB, TSS UF, and TSS SUF flip-flops can be considered as 1-bit extensions of the Instruction Buffer (IB) register, the Instruction Field (IF) register, and the Save Field (SF) register, respectively. These registers are part of the extended memory logic and are shown on engineering drawing D-BS-MC12-0-MXR. The signals used to clock these registers are also used to clock the TSS UB flip-flop, TSS UF flip-flop, and TSS SUF flip-flop.

TSS User Buffer (UB) – This is a D-type flip-flop shown in location H29. The data input level is derived from the output of a M121 AND/NOR gate and is true when the signal is low (ground), when the SUF instruction is initiated, or when the RMF instruction is given and the SUF flip-flop is in the 1 state. The flip-flop is clocked at TP5D of both of these instructions.

The flip-flop is cleared with the signal MXF0-IF-IB L shown on engineering drawing *TSS*. This signal is generated with key I/O PRESET, or when entering the INTERRUPT cycle.

TSS User Flag (UF) – The Clock Input is clocked at TP4 of a JMP or a JMS instruction. This strobes the contents of the TSS UB flip-flop into the TSS UF flip-flop. The flip-flop is cleared with key I/O PRESET, or at TS3 of the INTERRUPT cycle. The output of the TSS UF flip-flop is gated with the instructions HLT, IOT, and OSR. When the TSS UF flip-flop is in the 1 state, normal operation of these instructions is aborted, and an Interrupt Request is sent to the CP.

TSS Save User Flag (SUF) – This flip-flop is used to store the contents of the TSS User Flag and thus enable the monitor to restore normal operation after an Interrupt. The Clock Input is clocked when entering the INTERRUPT cycle, thus loading the SUF with the contents of the TSS UF flip-flop.

User Interrupt (UINT) – The data input level TSS INT REQ is derived from the output of an M121 AND/NOR gate and is true when low (ground), i.e., when one of the instructions HLT, OSR, or IOT is decoded from the CP instruction register (IR) and the TSS UF is set. It is also true when the TSS UINT flip-flop is set and the instruction being executed is not the CINT instruction. The flip-flop is clocked with the TP5D pulse and cleared with I/O PRESET or the CINT instruction.

Skip and Interrupt Levels – TSS INT SKIP L is derived from a NAND gate M117 and is true when low (ground). The inputs to the NAND gate are true when executing the SINT instruction and the TSS UINT flip-flop is in the 1 state. This level is gated onto the I/O SKIP BUS with the M623 Bus Driver. The common input to the bus driver, pin M2, is true when the KT12 flip-flop is inserted (M216) in location H29. The TSS INT REQ Bus level is true when low, and is also gated on the I/O Interrupt with the M623 Bus Driver.

7.8 KW12-A REAL TIME INTERFACE AND KW12-B, -C SIMPLE CLOCKS

7.8.1 KW12-A – Real-Time Interface

The KW12-A can be used to generate Program Interrupts over a range of intervals of 2.5 μs to 40.96s (see Table 7-7 below) in time base applications; detect external and internal events in order to count them, measure them against a time base, measure the interval between them, use them as a time base standard, or control sample times of A/D conversions. Figures 7-10 and 7-11 illustrate the basic relationships between the KW12-A and the PDP-12 System; for instance, the A/D sample feature.

Table 7-8 lists the time base parameters available with the KW12-A. The time base, generated by a 400 kHz crystal clock, is under program control as determined by the rate field of the clock control word (first word) (XNXX).

Table 7-8
KW12-A Clock Time Base Ranges

Time Base Range*	Resolution	Frequency	Rate Field
(Stopped)	0	0	0 or 7
0.0025 – 10.2400 ms	2.5 μs	400 kHz	1
0.01 – 40.96 ms	10 μs	100 kHz	2
0.1 – 409.6 ms	100 μs	10 kHz	3
0.001 – 4.096s	1 ms	1 kHz	4
0.01 – 40.96s	10 ms	100 Hz	5
1 – 4096 counts of Input T Events			

* Time base within each range selected, using mode field of clock control word.

The three external events inputs, INPUTS 1 through 3, may be simulated by program/subroutines on the occurence of events detected in I/O or DMA peripherals. The appropriate simulated input is transferred from the AC register to the clock along with the appropriate rate and mode information by the CLLR instruction (6132).

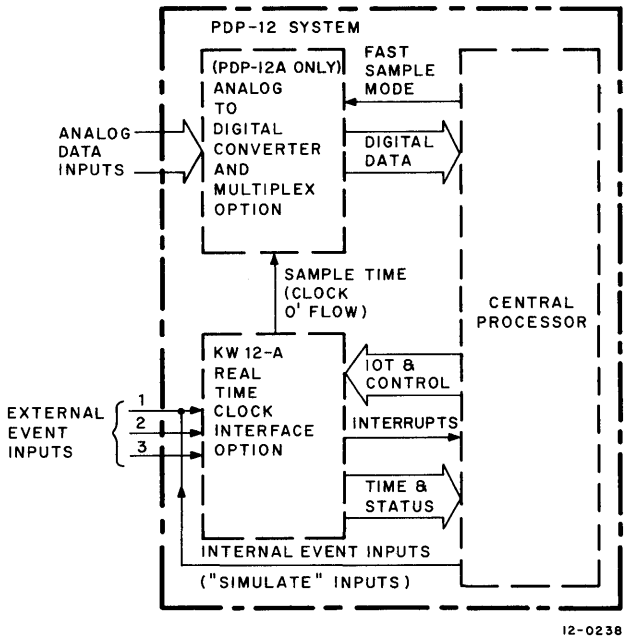


Figure 7-10 KW12-A Real Time Interface Functional Relationship

A clock enable word (CLEN instruction) is required to enable the desired Input Event lines and the clock conditions selected to produce a Clock Interrupt Request to the CP. If an external input (an instrument or the power line waveform) and the corresponding Interrupt are both enabled, a Program Interrupt is generated by the Event.

A clock status word (CLSA instruction) informs the program of the clock condition in a clock-initiated Program Interrupt. A summary of IOT instructions associated with a KW12-A is presented in the *PDP-12 System Reference Manual,* Paragraph 6.2.

The KW12-A is a prewired PDP-12 option with an input control panel, which mounts behind the vertical door on the left front of the PDP-12. The Real-Time Interface can be used to synchronize the Central Processor to external events, count external events, measure intervals of time between events, or provide program interrupts at programmable intervals from 2.5 μs to over 40s. Some of the above-mentioned operations can be done simultaneously.

INPUT channel 1 may be used to enable an external source to drive the counter. This external source may be switch-selected to be either the power line waveform or an actual signal from a laboratory instrument. The programmable selection of the rate is accomplished with the three rate-bits of the clock control register (CLLR instruction).

Input Synchronizers

Three input channels (INPUTS 1 through 3) are used to convert external events into a synchronized control and status signal for the clock. Each input channel consists of an input Schmitt trigger with pulse generator, five flip-flops, and associated control gating. The Schmitt trigger and pulse generator convert the preselected voltage threshold crossing by an external signal into a single event (pulse). This Schmitt trigger has level and slope

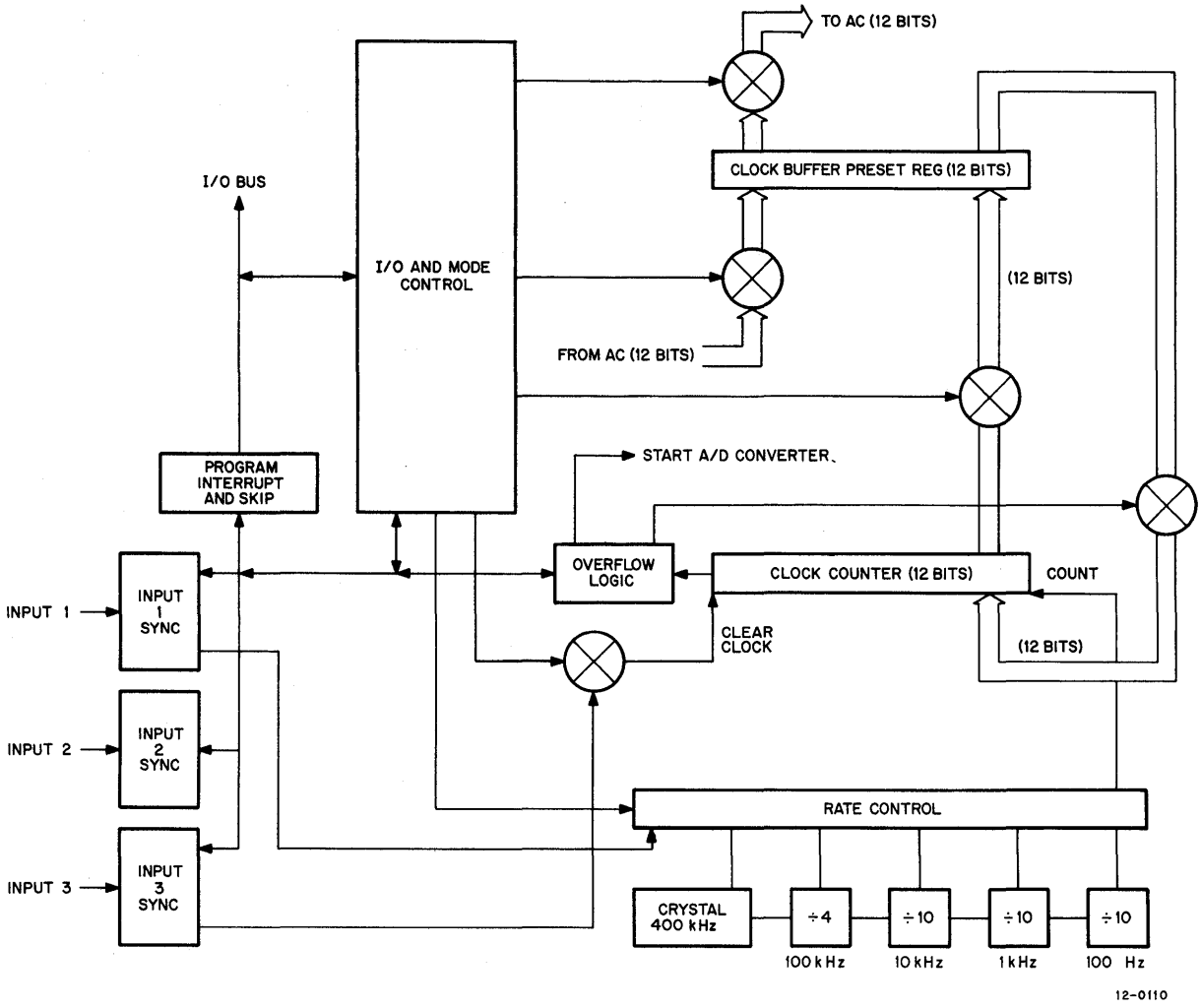


Figure 7-11 KW12-A Organization

selection controls available on the front panel. They provide selection of only threshold between ±6V and either positive- or negative-going slope. The Schmitt trigger has a hysteresis of 0.3V.

The five input circuit flip-flops are: INPUT ENABLE, INPUT, PRE EVENT, EVENT, and ENABLE EVENT INTERRUPT (see Figure 7-12). Figure 7-13 shows a basic timing of the Input Synchronizer.

Not shown in Figure 7-12 is the logic which clears the EVENT and PRE EVENT flip-flops when the CLSA instruction is given. This logic ensures that events occurring during the execution of the present CLSA instruction are indicated when the next CLSA instruction is given.

External signals (from up to three instrument/sources) are connected to the INPUT jacks on the front panel via standard 3-conductor telephone plugs. Each INPUT and OUTPUT jack is wired in parallel to permit convenient connection of the external input signals to the KW12-A, and if desired, the Analog-to-Digital Converter or external devices. The input is differential, ±5V range, input resistance greater than 10,000Ω, and protected against inputs up to ±50V. A level control and a slope control are associated with each input. The level control selects the threshold voltage at which the trigger pulse is generated. The slope determines the polarity of the input signal causing a trigger pulse. The trigger pulse sets the associated input flip-flop of the clock if that input channel is enabled.

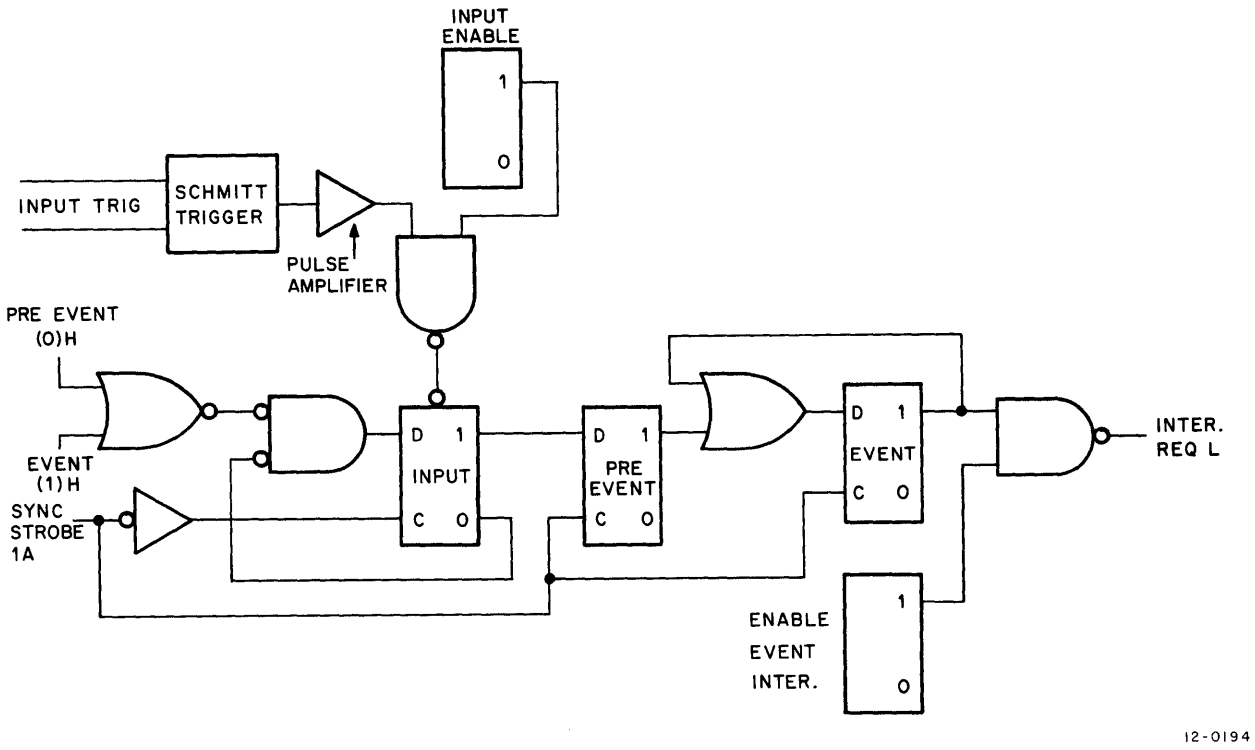


Figure 7-12 Simplified Input Synchronizer Logic Diagram

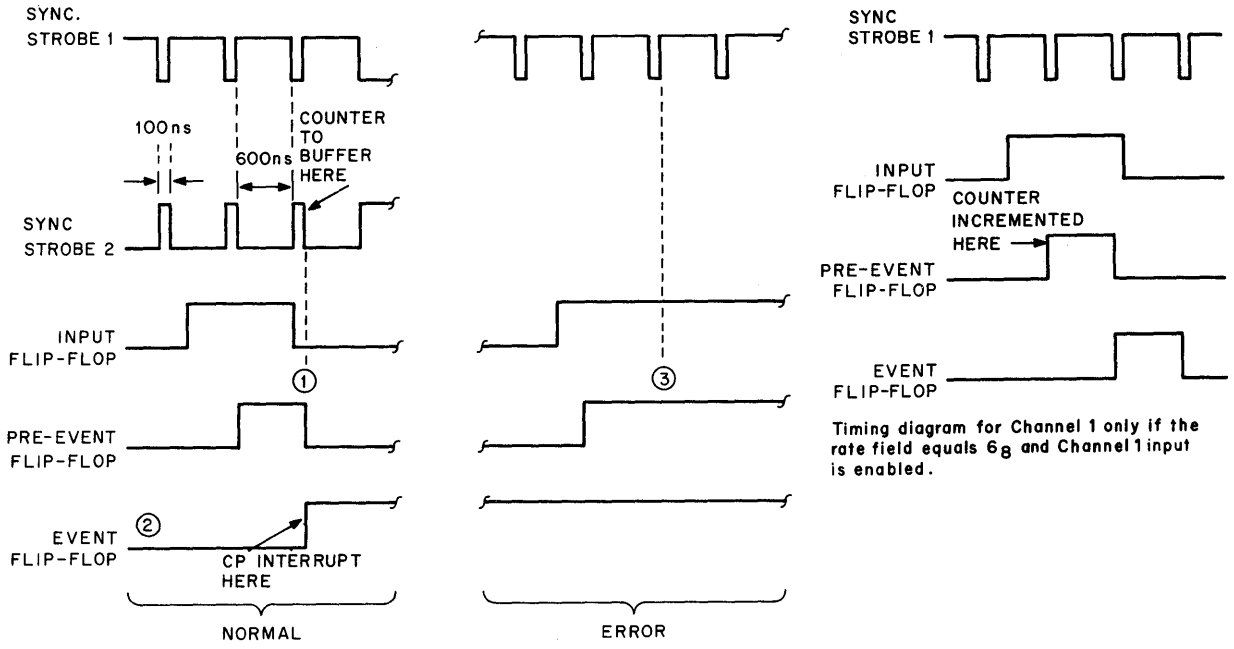
KW12-A Real-Time Interface Instructions – The KW12-A is controlled by PDP-12 IOT instructions. These instructions can be used with either 8 or LINC Mode. Execution time for the IOTs is 4.25 μs when in 8 Mode, and 5.9 μs when in LINC Mode (using IOB). Refer also to Paragraph 6.2.1.3 of the *System Reference Manual.*

Detailed Discussion

Clock control is accomplished by programmed IOT instructions. As with any device utilizing the I/O Bus, the most significant digit of the IOT instruction is $6nnn_8$. The center two octal digits (bits 3 through 8 of the IOT instruction) form the clock's device code, which must be 13_8. The device decoder within the clock enables the clock IOP gates when this device code occurs in an IOT instruction (Drawing D-BS-KW12-0-CLC). The IOP pulses, recombined with the bits of the three least significant octal digits of the IOT instruction, produce the complete clock instruction set ($613n_8$).

The clock control word (CLLR instruction) is transferred from the AC to the clock control registers (a 6132 instruction). The most significant digit of this word is stored in the RATE register (Drawing D-BS-KW12-0-CLR). These three bits are decoded to select one of the five internal clock rates, or no rate (clock stopped) to the clock counter (Drawing D-BS-KW12-0-CLKA, B). The clock MODE control register is loaded with the second most significant digit of the clock control word (bits 3, 4, and 5). Bits 7, 9, and 11 of the clock control word, if set, simulate events on clock INPUTS CHAN-1, 2, and 3, respectively (Drawing D-BS-KW12-0-CLEA, B, and C). These bits are not stored, but act in the same manner as external events on these inputs. Bits 6, 8, and 10 of the clock control word are not used.

The clock enable word is transferred from the AC to the clock enable flip-flops with a CLEN (6134) instruction. The flip-flops of this register are shown as the EN INPUT and EN EV (N) T (Enable Event Interrupt) flip-flops on the three input channel block schematics (Drawing D-BS-KW12-0-CLEA, B, and C) and the EN OV IN T



NOTES:

1. Normally PRE-EVENT is reset. If however, EVENT was already set, PRE-EVENT remains set as an error flag; indicating that more than one input occured between clearing of the EVENT flip-flop.
2. EVENT is cleared only by the CPU using the 6135 (CLSA) instruction with the following exception: Inputs to Channel 1 may be used to increment the counter. For this case the timing diagram is shown below.
3. Error indication that 6135 (CLSA) instruction was not issued soon enough following an EVENT.

12-0205

Figure 7-13 KW12-A Timing Diagram

(Enable Overflow Interrupt) flip-flop on the D-BS-KW12-0-CLIO block schematic. Bit 4 of the clock enable word, if set, causes transfer of the contents of the clock buffer register to the clock counter if the CLR MODE register contains 1 or 5. This bit (4) is not stored in the clock; the transfer occurs when the clock enable word is transferred from the AC. Gating for bit 4 is shown at coordinates A,6 on block schematic D-BS-KW12-0-CLR.

Transfers from the AC to the buffer preset register are accomplished using a CLAB (6133) instruction. The buffer register is shown across the bottom of block schematics D-BS-KW12-0-CLKA and D-BS-KW12-0-CLKB as CLKA BUF00-05 and CLKB06-11.

Clock status data is transferred from the clock to the AC, using the CLSA (6135) instruction. The status register in the clock consists of the EVENT (N) and PRE EVENT flip-flops of each of the three input channels and the OVERFLOW flip-flop. The relationship and timing of the EVENT and PRE EVENT flip-flops are discussed below (see also Figure 7-13). Drawing D-BS-DW-12-0-CLIO shows the gates from the status register flip-flops to the CP internal I/O bus at C, D-2.

Clock Interrupts are produced by one or more of four conditions set up in the clock enable flip-flops (see Figure 7-12). These conditions are an EVENT on an interrupt-enabled input channel and/or clock counter OVERFLOW if enabled.

Internal clock data transfers are controlled by the MODE register. The most significant bit of this register (MODE 0) is independent of the lesser two bits (MODE 1 and 2), and is used to enable the OVERFLOW signal

to the A/D Converter option to initiate a Sample. The lesser two bits of the MODE register control transfers between the clock counter (CNT) and the clock buffer preset register (BUF). If MODE bit 1 is set, the count in the clock counter (CNT) at the time an enabled EVENT occurs is transferred to the clock buffer preset register. The details of this logic are shown on D-BS-KW12-0-CLC and on each of the three input channel drawings (D-BS-KW12-0-CLEA, B, and C) of Volume III. If MODE bits 1 and 2 (least significant bits) are set and INPUT 3 is enabled, the shift counter (clock counter) contents are transferred to the clock buffer preset register and the clock continues to count. This logic is also shown on D-BS-KW12-0-CLC. If MODE register bit 2 is set and MODE bit 1 reset, the contents of the clock buffer preset register are transferred to the clock counter when the counter overflows, or when a clock enable word with bit 4 set (1) is transferred from the CP accumulator to the clock. This logic is shown on D-BS-KW12-0-CLR.

Except for minor differences that permit external events on the input channels to increment the clock counter, the three INPUT channels are identical. INPUT CHAN 2 is discussed here as typical of the three; it is shown in Volume III on block schematic D-BS-KW12-0-CLEB. Events outside the PDP-12 System are accepted at a Schmitt trigger which drives a pulse generator (M503 Module). The time relationship of the EVENT and PRE EVENT flip-flops is shown in Figure 7-13. The SYNC generator is a ring counter (shown on block schematic D-BS-KW12-0-CLC). INPUT CHAN 1 (CLEA) differs from INPUT CHAN 2 (CLEB) and CHAN 3 (CLEC) in that it may be used to increment the clock counter from an external source to count events when the RATE field of the clock control word (CLLR instruction) is 6_8 (AC01 + 02 (1)). On block schematic D-BS-KW12-0-CLEA it can be seen that a counter-to-buffer transfer (CNT TO BUF) is inhibited and the EVENT 1 flip-flop is set when the RATE field is 6_8. This logic prevents the clock buffer preset register from copying the contents of the clock counter (CNT) each time the counter is incremented by an event on channel 1. Because the EVENT flip-flop is set each time this mode of operation is initiated, the clock counter is also incremented.

The SYNC generator serves two purposes:

a. Primarily, it prevents transfer from the clock counter register into the clock buffer before the increment has had time to propagate through all 12 bits of the shift counter. This time is the limiting factor for the maximum rate at which the clock may be incremented.

b. Because of its relationship to the maximum clock rate, the SYNC generator strobes the PRE EVENT and EVENT flip-flops (Drawing D-BS-KW12-0-CLEA, B, and C). An error condition is indicated if two Events occur on the same input channel at a rate faster than the clock can increment. This feature is particularly important for INPUT CHAN 1 when it is used to increment the clock counter. The SYNC generator is inhibited during CLBA (6136) and CLCA (6137) instructions which transfer data from the clock buffer register and the clock counter to the CP. This function is performed by the I/O SYNC flip-flop shown on D-BS-KW12-0-CLC near the SYNC ring-counter, and prevents counter-to-buffer transfers while a clock buffer-to-AC transfer is occurring.

7.8.2 KW12-B and KW12-C Simple Clocks

The KW12-B and KW12-C clocks can provide Program Interrupts at manually selected clock rates. The KW12-B Interrupt rate is set by a jumper and potentiometer on the M401 Oscillator Module which uses an RC time base. The KW12-C uses a crystal M405 Oscillator Module. The only program control for these clocks is ON and OFF. The only data available from the KW12-B and KW12-C is a Clock Interrupt occurring at the preset time-base rate. The only clock status information available to the program is clock ON or OFF. A simplified block diagram, showing the relationships of the KW12-B and KW12-C to the CP is shown in Figure 7-14. Descriptions of time base ranges and the instruction set are presented in Paragraph 6.4.2 of the *PDP-12 System Reference Manual.*

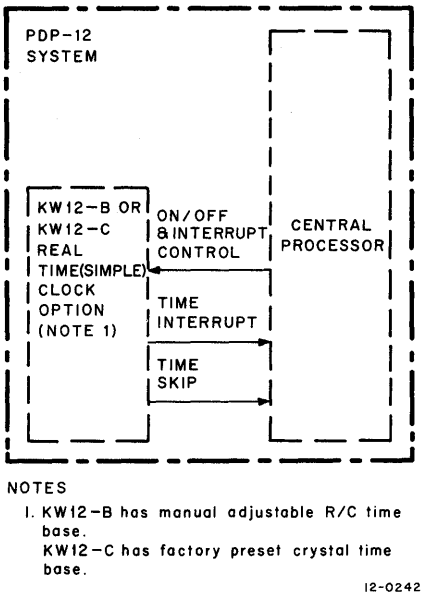


Figure 7-14 KW12-B and KW12-C Simple Clocks, PDP-12 System Functional Relationships

KW12-B

The KW12-B provides a simple means of interrupting the CP at intervals determined by a variable TC Oscillator (M401 Clock) Module. The KW12-B may be turned ON or OFF only under program control. However, any variations in frequency require a manual operation; physically changing the oscillator by the use of wire jumpers and a potentiometer. Refer to Volume III, Drawing D-BS-KW12-0-CLUB.

The KW12-B uses the M401 Clock in slot location F18, thus precluding the installation of any combinations of KW12-A, -B, or -C together in the same PDP-12 cabinet. Paragraph 6.2.2 of the *System Reference Manual* lists the frequencies available for the KW12-B Simple Clock.

KW12-C

The KW12-C Simple Clock also uses the M405 Clock Module (in slot location F18), as does the KW12-A Real-Time Interface. However, there are no decade counters providing different frequencies. Any fixed frequency (as ordered by the user; between 5 kHz and 50 kHz) provides frequency stability of .01 percent between 0° and +55°C. As with the KW12-B Simple Clock, the KW12-C can be turned ON or OFF only under program control. Any variations to the frequency would require changing the crystal CR1 on the M405 Module.

7.8.3 Block Diagram Discussions

Clock I/O Control (D-BS-KW12-0-CLR)

The logic shown on this block schematic basically comprises the device decoder, at coordinates D6 and D7; the IOP Transfer Decoder, at C and D2 and C and D3; and the SYNC generator at B4–7.

The SYNC timing chain generator produces the SYNC STROBE which: 1) clocks the input event signal to the EVENT and PRE EVENT flip-flops and 2) increments the Clock Counter. The timing chain generator is a ring-counter which disables momentarily, the IOT timing chain which allows the clock counter to settle out after each increment. At coordinates A6 and A4 are One-Shot Delay flip-flops that provide delays of 600 and 100 ns, respectively, to allow the clock counter time to settle.

The clock device decoder senses the second and third most significant bits, tests them for 13_8, and allows the IOP pulses to be gated to the clock.

The IOP Transfer logic provides the IOP signals to the correct clock functional logic areas, depending upon the coding of the least significant bits of the clock IOT instruction.

INPUT CHAN 1 (D-BS-KW12-0-CLEA)

This block schematic shows the logic discussed in the Detailed Discussion of the Input Synchronizers and the basic timing (see Figure 7-13). External events (either pulse or sinusoidal signals) are applied to the Schmitt trigger shown at D7 (an M503 Module). An adjustable THRESHOLD control (front panel) applies only preselected voltage levels to the Schmitt trigger inputs. The Schmitt trigger produces a pulse of 1 μs duration. The pulse is then output through an emitter follower which provides isolation to switching transients. The signal is applied to a pulse amplifier (at S2) which shapes the output to the proper logic level (a 1 μs ±5V signal). The EN Input (Enable Input) gates on and off input signals to the clock. It is set and cleared under program control. The INPUT flip-flop is set by an external signal from the Schmitt trigger input or under program control with the CLLR instruction. The INPUT flip-flop provides synchronization between external timing and internal clock rates.

The PRE EVENT flip-flop is strobed into the EVENT flip-flop by the STROBE 1 signal. This clears the INPUT flip-flop if EVENT is cleared.

The EVENT flip-flop is loaded with the PRE EVENT flip-flop on the next STROBE 1 and set to a 1 by the second STROBE 1, following the setting of the INPUT flip-flop. Subsequent to EVENT being loaded, PRE EVENT is cleared.

The EN EV INT (Event Enable Interrupt) flip-flop permits external events to cause program interrupts. It is set and cleared by the CLEN instruction.

The occurrence of CLC STROBE 2 with EVENT (0) and PRE EVENT (1) is the actual single event used by other parts of the clock logic, such as counting and transfers from counter to buffer-preset register.

The status of the EVENT and PRE EVENT flip-flops is loaded into the AC under program control. When this transfer occurs, the corresponding INPUT, PRE EVENT, and EVENT flip-flops are cleared. If a second input occurs before the EVENT flip-flop is cleared, then both the PRE EVENT and EVENT flip-flops will remain set, indicating an error.

INPUT CHAN 2 (D-BS-KW12-0-CLEB)
INPUT CHAN 3 (D-BS-KW12-0-CLEC)

The logic shown on these drawings is similar to that on D-BS-KW12-0-CLEA, except that there is no extra gating (coordinates C4 and D3, on -CLEA) to permit incrementing the event count (CLKA, B, or C CNT) without the IOT.

CLOCK IO INPUT (D-BS-KW12-0-CLIO)

Shown on the right side (C2 and D2) of this drawing are the clock status bits. These can be examined by the program by issuing a CLSA (6135) instruction.

The EN OVF INT (Enable Overflow Interrupt) flip-flop enables a program interrupt if set by the CLEN (6134) instruction. Note that if the clock skip is enabled (CLIO INT SKIP BUS), the clock interrupt is also enabled (CLIO INT RQST BUS). A clock skip flag cannot be enabled without the interrupt also being enabled.

CLOCK & BUFFER (D-BS-KW12-0-CLKA, B)

The two registers shown on the CLKA and CLKB drawings are the clock counter register (CNT00 through 11), a shift register that holds the current count, and the clock preset buffer, which interfaces the clock with the AC.

The buffer preset register (BUF) is used to buffer the current count in the clock register at the occurrence of an event when operating with CLR MODE 1 set. With MODE 1 cleared (0) and MODE 2 set (1), the buffer preset register folds the number to be transferred into the counter when overflow occurs. The buffer preset register can be loaded into the AC, or the AC can be transferred into the buffer preset register.

The counter register (CNT) is a 12-bit shift counter that is loaded from the buffer preset register or can be transferred into the buffer preset. The counter may be used to count events, measure intervals of time between events, or provide processor interrupts at program-selected intervals from 2.5 μs to over 40s.

CLOCK RATE (D-BS-KW12-0-CLR)

The logic on this drawing consists of: the clock RATE control register (upper left), the MODE control register (upper right), and the clock rate (COUNT) decoder.

The logic across the bottom of the drawing shapes and amplifies an External A/D Initialize signal and generates the COUNT register load signal (CLR LOAD CNT).

The RATE control flip-flops are set by the clock control word (CLLR instruction) and are applied to the clock rate decoder logic (coordinates B and C6) to establish the clock interval.

The MODE control flip-flops, also set by the clock control word, establish the clock action as determined by bits 3 through 5 of the clock control word. Refer to the *System Reference Manual,* Paragraph 6.2.1.3. The MODE 0 flip-flop enables an initialize signal to the A/D option which will start a conversion, or if MODE 0 is not set, the A/D option will be initialized by the clock at the occurrence of OVERFLOW.

The clock rate decoder selects the rate at which the clock will operate. The logic shown at coordinates B and C6 and B and C7 form inhibit logic. That is, when any CLTB NNN HERTZ H signal (where N = frequency) goes low, the associated gate no longer qualifies, thereby causing the CLR COUNT flip-flop to increment the clock counter. These load pulses will occur at the rate selected by the RATE flip-flops.

The CLR OVERFLOW flip-flop is set when the clock counter (CLKA, B) is full (up to 4096 counts) and will result in a clock interrupt request if the CLIO EN OVF INT flip-flop is set (see KW12-0-CLIO).

CLOCK TIME BASE (D-BS-KW12-0-CLTB)

The logic shown on this drawing comprises three decade counters (four flip-flops), binary counters modified by use of hardwired preset pulses to count by 10 instead of 16. The counters provide the frequency intervals (2.5 μs to 4s) and are driven by a 100 kHz signal generated by an M405 (400 kHz clock) Module through two frequency dividers (flip-flops CLTB 00, 01). Note that the logic for the counters is located on the same module as the Input Synchronizers (Drawings D-BS-KW12-0-CLEA, B, C).

7.9 TC12-F – 8 TAPE CONTROL

The TC12-F DECtape option extends the capability of the TC12 LINCtape System to read and write DECtapes that are formatted on the PDP-8, PDP-9, PDP-10, or PDP-15 computers. This option is utilized with the PTYC12-F user's program (see program document DEC-12-YJYA-D for the program description and the difference between the LINCtape and DECtape formats).

TC12-F Operation

When the LTP8 TAPE flip-flop is set (1) (see dwg. LTP8), the TC12-F is selected. This will inhibit the LWN EN WIND DECODE signal and invert the timing track input. Data from the MARK Track are shifted into the RC12-F mark window register and decoded. When reading, data are assembled in the Read/Write Buffer (RWB) and loaded into the TAC register, and subsequently transferred to the AC with the TAC-to-AC (MSC 3) instruction. When writing, data are transferred from the AC to the TB with IOT 6154 and loaded into the RWB and written on tape. The Tape Break state is not entered for a tape operation employing the TC12-F option.

7.9.1 Logic Description (TC12-F-LTP8)

The control logic is shown on the TC12-F-LTP8 block schematic (see Volume III of the *Maintenance Manual*).

LTP8 TAPE – When this flip-flop is set (1), the TC12-F option is selected and portions of the TC12 LINCtape Control are disabled. The LWN EN WIND DECODE level is disabled and the input from the timing track is inverted.

WINDOW REGISTER – (LTP8 W1 through W9) Data from the mark track are shifted into this 9-bit window register with every TP3 pulse. The register bits are decoded to give the signal LTP8 BM. This signifies that a block mark has been detected on the tape and the block number (BM) is in the Read/Write Buffer (RWB). (It is assumed here that the reader is familiar with the LINCtape Control theory.)

BLOCK – When the signal LTP8 BM is true, the BLOCK flip-flop is set (1) with TP4. The status of the flip-flop is checked with the SKL 14 instruction.

LTP8 WRITE – When the WRITE flip-flop is set, the LTP8 WRITE SEL signal is true. This controls the status of the LCS WRITE SYNC and LCS WRITE flip-flops, in order to accomplish the writing of data on tape.

LTP8 TAPE WORD – The line counter (LCS) is decoded to generate this signal when every fourth line is encountered on tape. This signifies that a 12-bit word is assembled in the RWB register and is ready for transfer to the TAC with the next TP4 pulse (LTP8 LD TAC).

7.9.2 Instructions

The following instructions are used in conjunction with the TC12-F option:

IOT 6152	
AC Bit	**Functions**
AC 4 (1)	Clear Block
AC 5 (1)	Set Backward

(continued on next page)

IOT 6152	
AC Bit	**Functions**
AC 6 (1) AC 7 (1) AC 9 (1) AC 10 (1)	Select Unit 1 Set Forward Set 8 Motion and Forward if Motion = 0 Sel 8 Tape and AC 11 → Write
SKL 14 = Skip on 8 Block SKL 17 = Skip on 8 Word Tape Preset = 0 → Write 0 → Motion Deselect 8 Tape	

7.10 XY12 INCREMENTAL PLOTTER CONTROL

The XY12 Incremental Plotter Control generates the necessary levels to control the plotters listed in Table 7-9. The control logic accepts IOT instructions from the PDP-12 and converts them to specific operational commands that are transmitted to the plotter, producing the desired pen and drum movements. Discrete points may be plotted; and vertical, horizontal, or diagonal lines may be produced in any combination by application of the proper programmed commands. The control is contained on one M704 Double-Width Integrated-Circuit Module located in the CP mainframe, location MN05. The control logic is compatible with the CalComp Digital Incremental Plotter Model 563 (30 in.-wide) and Model 565 (12 in.-wide). Each of these models can be purchased with the incremental step size of 0.010 in., 0.005 in., and 0.10 mm.

Table 7-9
XY12 Incremental Plotter Specifications

Name	Model	Paper Width (inches)	Speed (step/minute)	Step Size (inches)
CalComp	563	30	12,000	.01 .003 .1-mm
	565	12	18,000	.01-in. .003-in. .1-mm
Complot	DP-1-1 DP-1-5 DP-1-M2 DP-1-M1	12 12 12 12	18,000 18,000 18,000 18,000	.01-in. .005-in. .25-mm .1-mm

When the XY12 option, a cable assembly (7005543) is provided which has a W023 Card Connector at one end, and a 25-pin Cannon Connector at the other. This cable, which is 10 ft long, fits either the 12 in. or 30 in. model. The PDP-8 CalComp Diagnostic (MainDEC 08-D6CC-D(D) 0) tests any CalComp or Houston Plotter.

Figure 7-15 illustrates the functional relationship between the PDP-12, the XY12 control logic, and the incremental plotter.

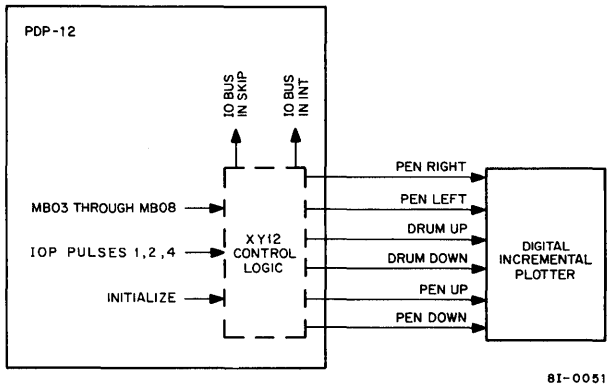


Figure 7-15 XY12 Block Diagram

The paragraphs that follow describe the operation of the control logic portion of the plotter as it relates to both the PDP-12 and the mechanism. Maintenance procedures and a description of the plotter mechanism and its operation are provided in the *CalComp Digital Incremental Plotter Maintenance Manual* and *Houston Plotter Manual* supplied with the system. The maintenance procedures pertaining to the PDP-12 also apply to the XY12 control logic. The recommended maintenance procedures for the CalComp Plotters are described in the respective California Computer Products Inc. manuals, and Houston Plotter manuals.

7.10.1 Logic Description

The XY12 control logic interprets a PDP-12 instruction in the memory buffer register to generate control pulses that set the motion control flip-flops. These flip-flops initiate plotter motion by moving the pen up, down, left, or right and/or moving the drum up or down.

The PDP-12 instructions are decoded to initiate plotter functions in the following manner. Memory buffer bits 0 through 2 contain operation code 6_8, indicating an I/O instruction. The PDP-12 Memory Buffer bits 3 through 8 contain the device code to select the XY12. Three device codes have been assigned to the plotter control. These codes (dwg. *PCL*) and their functions are as follows: 50_8 (Plotter Flag Up and Pen Up); 51_8 (Pen Right and Drum Down); and 52_8 (Pen Left and Down). Bits 9, 10, and 11 of the instruction generate the IOP pulses in the PDP-12. The IOP pulses combine with levels in the XY12 control logic to generate specific signals that initiate plotter motions.

7.10.2 Logic Operation

The following paragraphs describe the operation of the XY12 control logic. Logic circuitry for this option appears on engineering drawing D-BS-XY12-0-PCL. Whenever the computer is turned on, IOC IO PRESET H is generated, which in turn generates PCL IO PRESET L. This signal direct-clears the PLTR FLAG (Plotter Flag) flip-flop and generates the signal PC L FAST OP DONE (Fast Operate Done) to clear the Pen Right, Pen Left, Drum Up, and Drum Down motion control flip-flops.

The PC PLTR IOT H and the PCL 50 INST H level are the decoded select levels used in the XY12 logic. PCL PLTR IOT H is generated whenever the IOT contains 50_8, 51_8, or 52_8 (from memory buffer bits 3 through 8). When active, this level is at +3V.

The PCL 50 INST H level is generated and active only when the PDP-12 Memory Buffer bits 7 and 8 are cleared. The inverse of this signal, PCL 50 INST H, is also used, and is generated whenever MB07 or MB08 is set, e.g., 50_8 or 52_8 address code. The IOT levels (PCL PLTR IOT H, PCL 50 INST H and PCL 50 INST H) combine with the IOP pulse to generate specific control pulses.

The plotter operations are classified as being either fast (Pen Right, Pen Left, Drum Up, Drum Down) or slow (Pen Up, Pen Down) motion. When a fast-motion instruction is executed, the following events occur in sequence. With any one of the fast-motion control flip-flops set, plotter motion is initiated and the first 2.5 ms delay is triggered. After this second delay time (5.0 ms total), the Plotter Flag sets. When a slow-motion instruction is performed, the events occur in a manner similar to the fast-motion operation. The slow-motion (either Pen Up or Pen Down) flip-flop and the 35 ms delay are triggered; after this delay period, PCL SLOW OP DONE H is generated, and the second 35 ms delay is triggered. PCL SLOW OP DONE H direct-clears the Pen Up motion-control flip-flop. This long delay time allows the drum to settle in position and also prevents erroneous plots.

The Plotter Flag is set by the execution of any plotter-motion instruction. When this occurs, the INT INT RQST BUS L activates (0V), and the INT SKIP BUS L gate is partially enabled. INT INT RQST BUS L indicates to the PDP-12 that the device is requesting service if the Program Interrupt facility is enabled. Under program control, the computer then enters a Search subroutine to determine which device caused the Interrupt.

7.10.3 XY12 Instructions

Table 7-10 contains the instruction set for XY12 operations.

Table 7-10
XY12 Instructions

Mnemonic	Octal Code	Operation
PLSF	6501	Skip on Plotter Flag. This instruction decodes to generate PLTR IOT, 50 INST, and IOP1 to check the flag status. If the flag is set, IO BUS IN SKIP is generated.
PLCF	6502	Clear the Plotter Flag. This instruction decodes to generate PLTR IOT, 50 INST, and IOP2 to clear the flag.
PLPU	6504	Pen Up. This instruction decodes to generate PCL PLTR IOT, H PCL 50 INST H, and IOC IOP4 H to raise the plotter pen from the surface of the paper. PLPU is a slow operation.
PLPD	6524	Pen Down. After the drum and/or pen are moved, a PLPD instruction decodes to generate PCL PLTR IOT H, IOC IOP4 H, and PRD MB07(1) H. This lowers the pen to the surface of the paper. PLPD is a slow operation.
PLPR	6511	Pen Right. This instruction decodes to generate PCL PETR IOT H, IOC IOP F H, and PRE MB08(1) H to move the plotter pen one increment to the right. It is a fast operation. This instruction can be combined with the Drum Up (6512) or Drum Down (6514) instruction to produce a diagonal plot.

(continued on next page)

Table 7-10 (Cont)
XY12 Instructions

Mnemonic	Octal Code	Operation
PLPL	6521	Pen Left. This instruction decodes to generate PCL PETR IOT H, IOC IOP F H, and PRD MB07(1) H. It is a fast operation. PLPL moves the plotter pen one increment to the left. This instruction can be combined with the Drum Up (6522) instruction to produce a diagonal plot.
PLDU	6512	Drum Up. This instruction decodes to generate PCL PLTR IOT H, PCL 50 INST H, and IOC IOP2 H. The drum is moved up one increment. This instruction can be combined with Pen Right.
PLUD	6522	Drum Up. Same as above, only this can be combined with Pen Left. The drum is moved one increment up.
PLDD	6514	Drum Down. This instruction decodes to generate PCL PLTR IOT H, IOC IOP4 H, and PRE MB08(1) H. PLDD is a fast operation that moves the drum down one increment. It can combine with the PLPR instruction.

INDEX

C (Cont)

D

D (Cont)

E

F

G

H

I

I (Cont)

I (Cont)

I (Cont)

J

K

L

L (Cont)

M

M (Cont)

M (Cont)

N

O

P

P (Cont)

Q

R

R (Cont)

S

S (Cont)

T

T (Cont)

U

V

W

W (Cont)

X

Y

READER'S COMMENTS

PDP-12 MAINTENANCE MANUAL, VOLUME 1
DEC-12-HR1B-D

Your comments and suggestions will help us in our continuous effort to improve the quality and usefulness of our publications.

What is your general reaction to this manual? In your judgment is it complete, accurate, well organized, well written, etc.? Is it easy to use? _____

_____

_____

_____

What features are most useful? _____

_____

_____

_____

What faults do you find with the manual? _____

_____

_____

_____

Does this manual satisfy the need you think it was intended to satisfy? _____

Does it satisfy *your* needs? _____ Why? _____

_____

_____

_____

Would you please indicate any factual errors you have found. _____

_____

_____

_____

Please describe your position. _____

Name _____ Organization _____

Street _____ Department _____

City _____ State _____ Zip or Country _____

Fold Here

Do Not Tear - Fold Here and Staple

FIRST CLASS
PERMIT NO. 33
MAYNARD, MASS.

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

Postage will be paid by:

digital

Digital Equipment Corporation
Technical Documentation Department
146 Main Street
Maynard, Massachusetts 01754