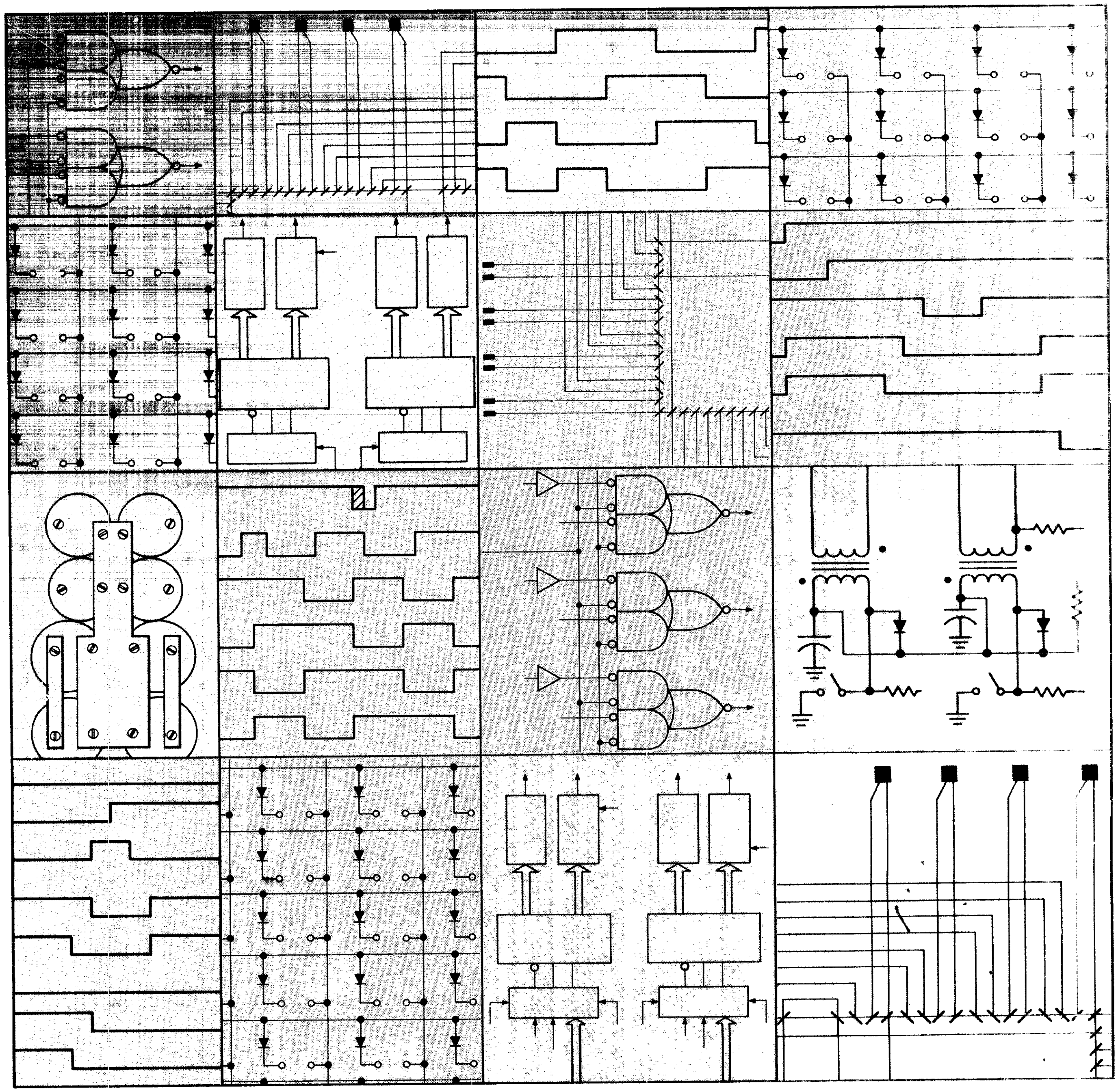
pdp8/e

pdp8/F & pdp8/m

internal bus options maintenance manual volume 2



digital

pdp8/e

MAINTENANCE MANUAL
VOLUME 2

For additional copies order No. DEC-8E-HR2C-D from Direct Mail, Digital Equipment Corporation, Maynard, Massachusetts 01754 Price: $25.00

digital equipment corporation · maynard. massachusetts

1st Edition May 1972
2nd Printing (Rev) February 1973

CONTENTS

Page

PART 1 EXTENDED ARITHMETIC ELEMENT OPTION

CONTENTS (Cont)

CONTENTS (Cont)

CONTENTS (Cont)

CONTENTS (Cont)

CONTENTS (Cont)

CONTENTS (Cont)

ILLUSTRATIONS

ILLUSTRATIONS (Cont)

ILLUSTRATIONS (Cont)

ILLUSTRATIONS (Cont)

TABLES

PREFACE

This manual, the second in a series of three, describes nonperipheral options of the PDP-8/E and PDP-8/M.

The content of this manual includes installation procedures, theory of operation, and maintenance procedures for the options described. It is assumed that the reader is thoroughly familiar with Volume 1 of this series, and with the applicable sections of the *1972 PDP-8/E & PDP-8/M Small Computer Handbook.*

PART 1
EXTENDED ARITHMETIC ELEMENT OPTION

CHAPTER 1
KE8-E EXTENDED ARITHMETIC ELEMENT

SECTION 1 INTRODUCTION

1.1 GENERAL DESCRIPTION

The KE8-E Extended Arithmetic Element option enables the PDP-8/E to perform arithmetic operations at high speeds by incorporating EAE components with the existing central processor logic so that they operate asynchronously. All logic is contained on two quad-size modules, designated M8340 and M8341, which plug directly into the OMNIBUS. The two modules are interconnected by one H851 Connector. A second H851 Connector interconnects the M8341 to the major registers control module (Figure 1-1). This connector carries register gating and controls from the EAE modules to the register controls module. A third H851 Connector interconnects the processor's M8330 Timing Generator Module with the EAE control, supplying clock and IOT functions to the EAE. The basic OMNIBUS signals connect to each module.

1.2 SOFTWARE

The following programs are used in the maintenance of the KE8-E option.

a. KE8-E EAE Test Part 1 (MAINDEC-8E-DOLB) – This program tests all EAE instructions except MUY and DVI.

b. KE8-E EAE Test Part 2 (MAINDEC-8E-DOMB) – This program tests the MUY and DVI instructions.

c. KE8-E EAE Extended Memory Exerciser (MAINDEC-8E-DORA) – The KE8-E Extended Memory Exerciser is a test of the KE8-E "B Mode" instructions which, during the DEFER cycle, use the word following the instruction to obtain the operand. The capability of each instruction to access every memory field from every memory field through nonautoindex and auto-index is tested.

1.3 COMPANION DOCUMENTS

The following documents and publications are necessary in the operation, installation, and maintenance of this option:

a. *PDP-8/E & PDP-8/M Small Computer Handbook* – DEC, 1972
b. *PDP-8/E Maintenance Manual* – Volume 1
c. *Introduction to Programming* – DEC, 1972
d. DEC engineering drawings M8340-0-1 and M8341-0-1
e. KE8-E EAE Test Part 1, MAINDEC-8E-DOLB-D
f. KE8-E EAE Test Part 2, MAINDEC-8E-DOMB-D
g. KE8-E EAE Extended Memory Exerciser, MAINDEC-8E-DORA-D

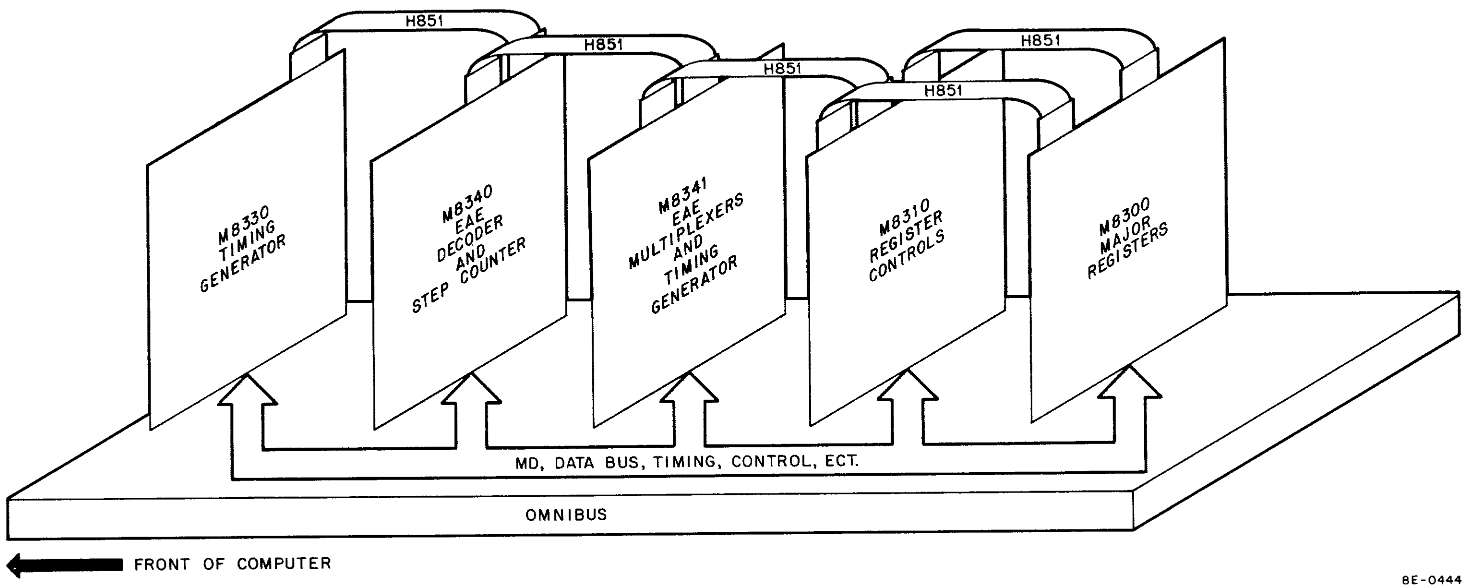


Figure 1-1 EAE Interconnections

SECTION 2 INSTALLATION

The KE8-E EAE option is installed on site by DEC field service personnel. The customer should not attempt to unpack, inspect, install, checkout, or service the equipment.

1.4 INSTALLATION

Perform the following procedures to install the KE8-E options:

Step	Procedure
1	Remove the modules from the shipping containers.
2	Inspect the modules for any apparent damage.
3	Connect the modules as follows:

a. Insert the EAE modules between the Timing Generator and the CP Major Registers and Register Control as follows:

M8330	Timing Generator
M8340	EAE Decoder and Step Counter
M8341	EAE Multiplexers and Timing Generator
M8310	CP Major Registers Control
M8300	CP Major Registers

The five modules *must* be installed in this order with no vacant slots between them for the H851 Connectors to fit properly.

b. Install H851 Connectors (five total) to connect the five modules. All connectors at the top of these modules will be utilized when all H851s are installed.

NOTE

The EAE is a complex instruction decoder that extends the basic PDP-8/E instruction set. It is intimately connected with the basic central processor and relies heavily on an M8300 and M8310 in good condition. Many potential problems can be avoided by running Instruction Test I (MAINDEC-8E-DOAB) and Instruction Test II (MAINDEC-8E-DOBB) before installing the EAE to verify the condition of the CPU. These tests should be run again after EAE installation to verify that the EAE is not malfunctioning and thereby modifying the basic instruction set.

1.5 CHECKOUT

Perform the following procedures to checkout the KE8-E option:

Step	Procedure
1	Verify that both EAE modules have been installed
2	Perform acceptance test procedures provided in Volume 1, Paragraph 2.3.
3	Load MAINDEC-8E-DOLB and perform EAE Test – Part I.
4	Load MAINDEC-8E-DOMB and perform EAE Test – Part II.

(continued on next page)

Step	Procedure
5	Load MAINDEC-8E-DORA and perform EAE extended memory exerciser (even if 4K machine).
6	Make entry on user's log that the acceptance test for the KE8-E was performed satisfactorily.

SECTION 3 SYSTEM DESCRIPTION

The organization of the EAE system block diagram (Figure 1-2) follows the organization of the detailed logic description. The detailed logic is organized by source, route and destination and contains logic diagrams representing each block illustrated in Figure 1-2.

Signals generated within the EAE control the operation of the M8300 Major Registers Module during EAE instructions. The processor timing extension logic causes the processor to halt at TP3 and at the same time starts the EAE Timing Generator. This extends TS3 to enable data to be applied to the adders a number of times. The EAE selects which register is to go into the adders by asserting a combination of signals shown in the EAE source control logic block. What happens to the data when it is on route to its destination is accomplished by asserting a combination of signals in the EAE route control signals block. The destination of the data is either the AC Register or the MQ Register.

The EAE was designed for hardware compatibility with old programs that were written for the PDP-8/I. The MODE flip-flop is cleared, selecting Mode A, when the computer is turned on. Mode A is the PDP-8/I compatible mode. Mode B is selected (via the mode-change instruction) only when using programs developed specifically for this EAE.

To better understand how the EAE functions, 13 of the EAE instructions are described in terms of functional flow to illustrate how the EAE completes each instruction.

NOTE

EAE operation is more integrated with the CPU than most options. Before attempting to study EAE theory of operation, the reader should thoroughly understand CPU theory and review sections of Volume 1 as he is reading this chapter.

1.6 STEP COUNTER LOADING OPERATION (Figure 1-3)

The Step Counter controls the number of shifts performed during the ASR, LSR, and SHL instructions. It also controls the number of steps taken during MUL or DVI, and records the number of shifts required to normalize a number.

The KE8-E provides two methods of loading the Step Counter. The ACS instruction is used by the new, or Mode B, instruction set; the SCL instruction is used by the old, or Mode A, instruction set. The SCL instruction is of interest because this same method of step counter loading is used within the SHL, LSR, and ASR instructions in both modes.

The ACS instruction takes place in a manner similar to an I/O transfer to a peripheral. The contents of the AC are placed on the DATA BUS during TS3. C0 is grounded so that the AC will be cleared. At the leading edge of TP3 the five least-significant bits are loaded into the Step Counter.

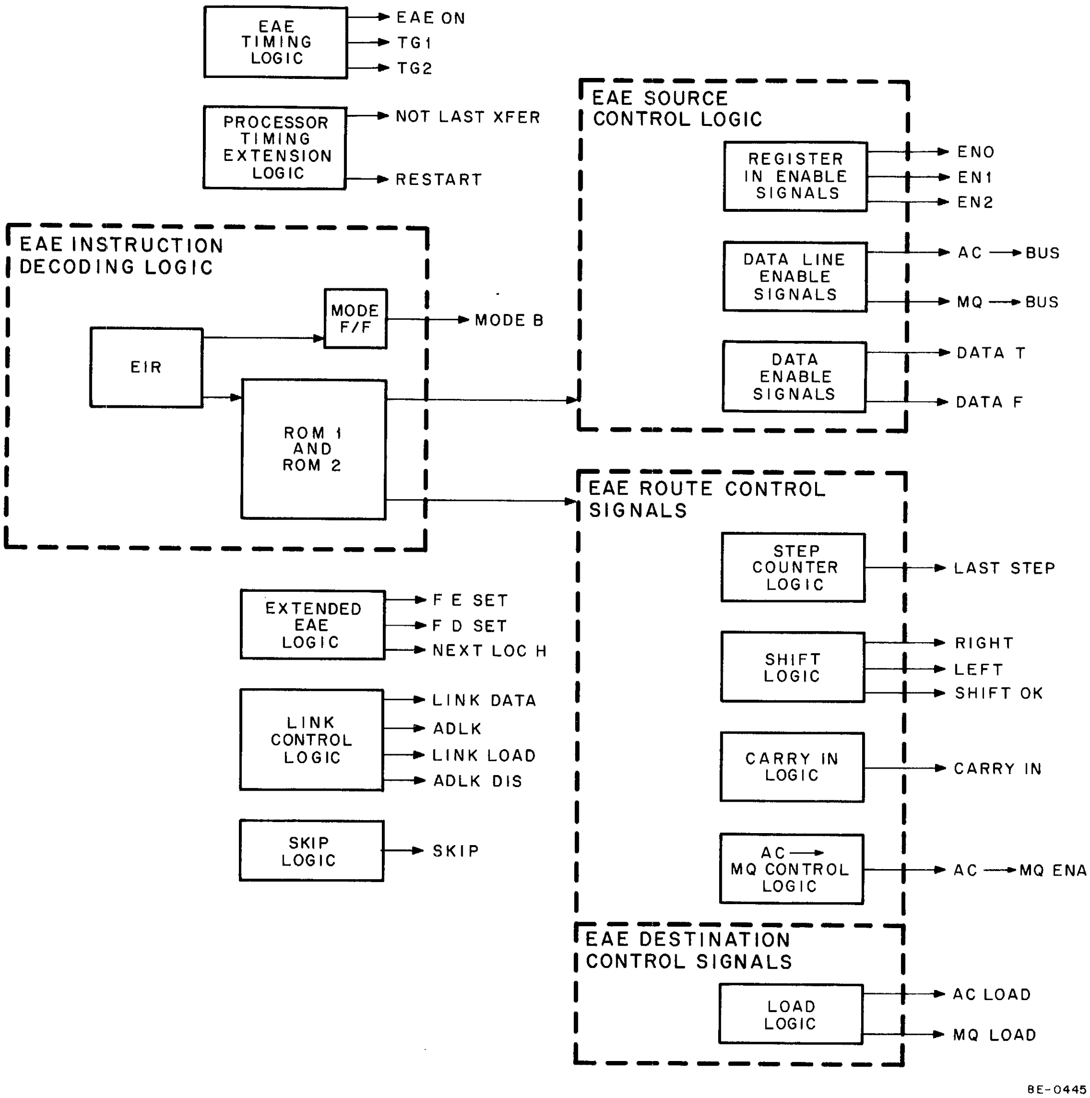


Figure 1-2 EAE System Block Diagram

The SCL instruction is somewhat more complicated. The Step Counter is to be loaded with the 1's complement of the next word in memory. As soon as the instruction is decoded, the SKIP line on the OMNIBUS is grounded. As explained in Paragraph 3.38 of Volume 1, the SKIP line is tested during IOT and OPERATE instructions. Grounding the SKIP line causes the next location (which, in this instance, contains the data for the Step Counter) to be skipped as an instruction.

During TS4 of the FETCH cycle, several control lines into the M8300 Major Registers Module are asserted by the KE8-E via the H851 Connectors. These signals (ENO and CARRY IN) cause the next location in memory to be addressed and treated as an operand (F E SET). During TS2 of the EXECUTE cycle, the contents of the five least-significant MD lines are inverted and applied to the inputs of the Step Counter. At TP2 the Step Counter is loaded.

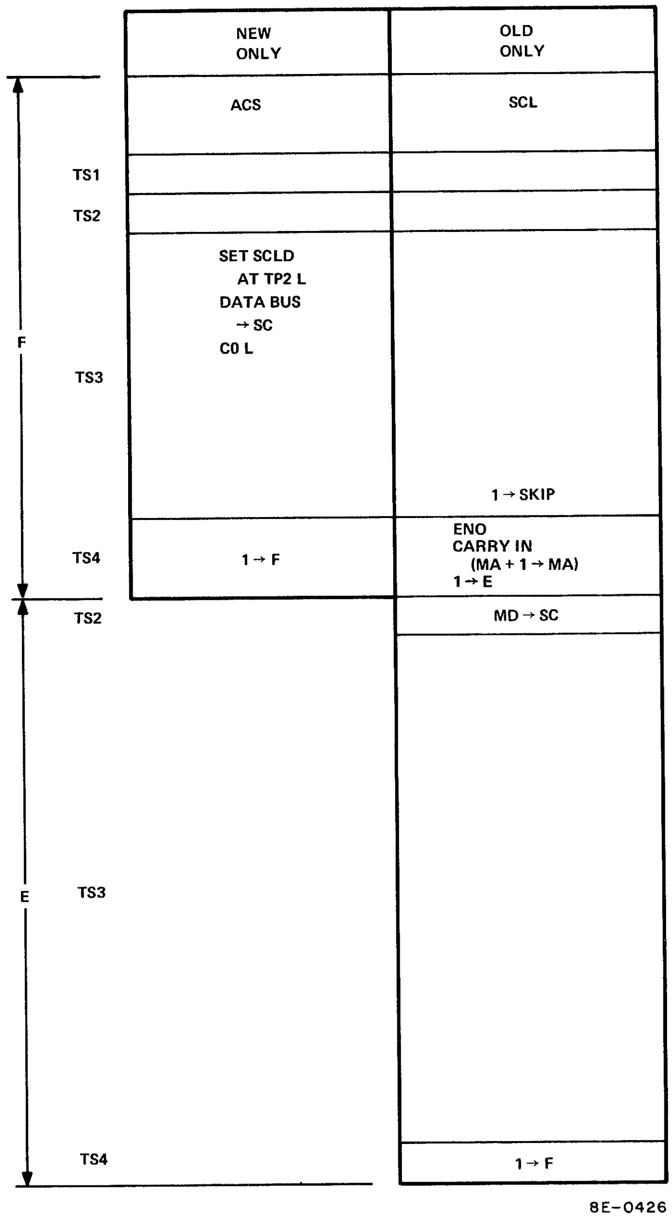


Figure 1-3 Step Counter Loading, Flow Diagram

1.7 STEP COUNTER TO ACCUMULATOR LOADING OPERATION (Figure 1-4)

The contents of the Step Counter are ORed with the contents of the AC and the result transferred to the AC. The entire operation is so similar to an IOT input OR transfer that it will not be discussed further.

1.8 THE SHIFT LEFT OPERATION

The shift left (SHL) instruction (Figure 1-5) is a 2-cycle instruction. The first cycle fetches the instruction word; the second cycle fetches a number that specifies the number of shifts that are to occur. The entire operation is identical to the SCL instruction up to and including TP2 of the EXECUTE cycle.

At the start of TP3 of the EXECUTE cycle, the EAE must shift the contents of the AC, MQ, and Link left by the number of places specified in the Step Counter. Normal machine timing stops at TP3 and EAE timing begins; one shift operation occurs with each clock pulse until the last shift has been performed.

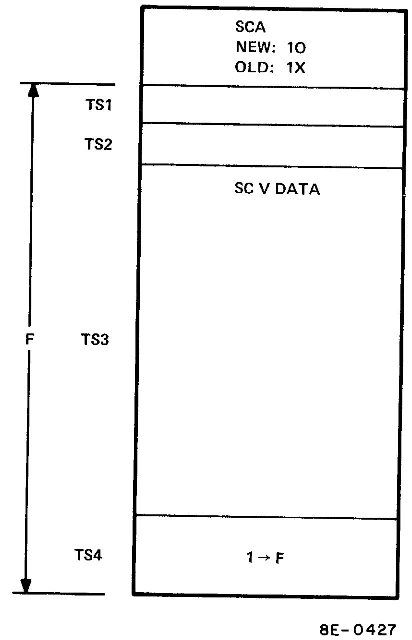


Figure 1-4 Step Counter to AC, Flow Diagram

Once the EAE is on, the following signals to the M8300 are asserted:

Signal	Function
LEFT L	Enables left shift gates at output of adders.
SHL + LD EN L	Enables MQ left shift path.
ADLK DIS L	Disconnects the normal Link-AC11 shift path. Also disconnects the AC0-Link shift path.

The following logic functions also occur:

Signal	Function
MQ0 → ADLK L	Establishes shift path from MQ0 to AC11.
AC0 → LINK DATA L	Establishes shift path from AC0 to Link.

MQ DATA is negated (high) so 0 is shifted into the MQ11. For each shift AC LOAD, MQ LOAD and LINK LOAD are developed and 1 is added to the step count. When the step count reaches 37, the EAE starts its shut-down process.

If the instruction mode is A, the EAE merely performs its last shift with NOT LAST XFER high. The processor restarts, and the total number of shifts is one more than the number in the second core location. If the instruction mode is B, a special line within the EAE, SHIFT OK H, is negated. Negating SHIFT OK H negates the signals required to cause AC shifts, and inhibits LINK LOAD and MQ LOAD. Thus, the processor starts without taking the final shift; the number of places shifted is equal to the number in the second core location.

1.9 RIGHT SHIFT OPERATIONS (Figure 1-5)

Two right-shift instructions, ASR and LSR, are available in the EAE option. The only difference between the two instructions is how the Link is handled. The Link is loaded at TP3 of the FETCH cycle via the OMNIBUS. If the LSR instruction (logical right shift) is being processed, no data is placed on the LINK DATA line and thus the Link is cleared. If the ASR instruction (arithmetic right shift) is being processed, AC0 is placed on the LINK DATA line and the Link is thus made equal to AC0.

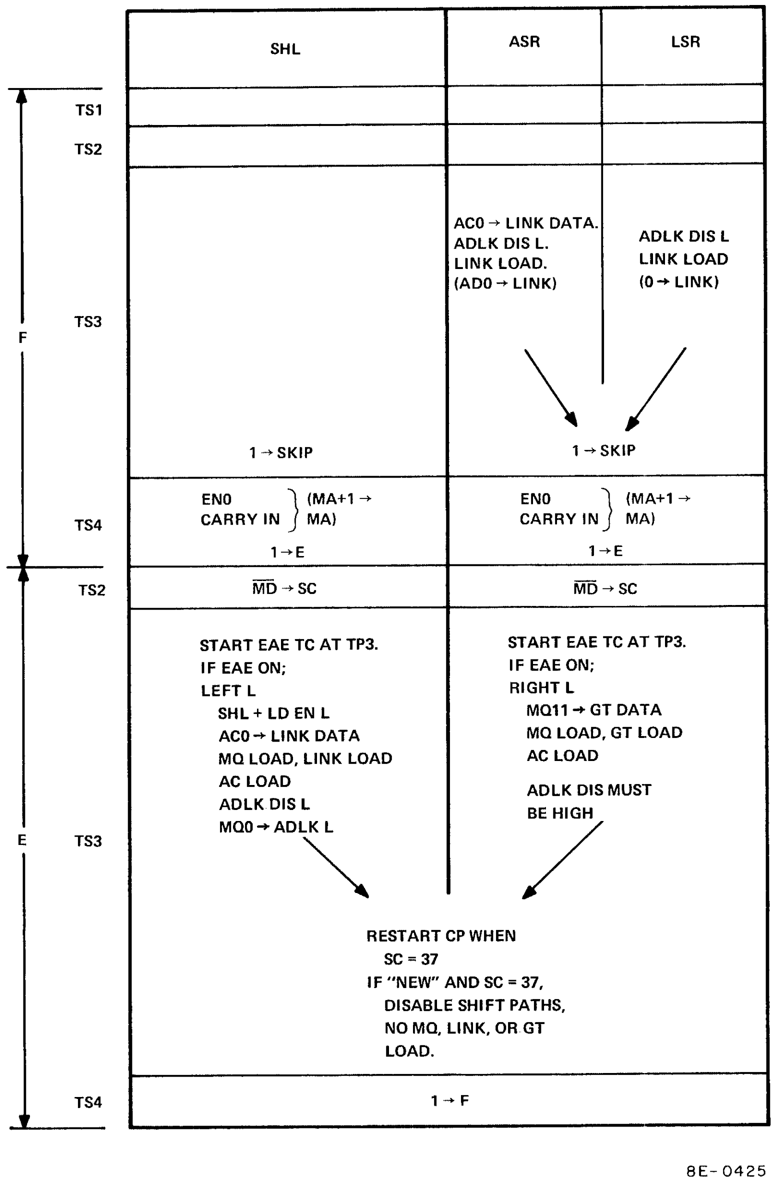


Figure 1-5 SHIFT Operations, Flow Diagrams

As in the case of SHL, the computer enters the EXECUTE cycle to obtain step-count information. When the EAE is turned on at TP3 of the EXECUTE CYCLE, the following signals are asserted:

Signal	Function
RIGHT L	Enables MQ right shift, enables right shift gates at adder outputs.
MQ11 → GT DATA	Enables path from MQ11 to the GT flag.

Like the SHL instruction, AC LOAD and MQ LOAD are generated for each shift and the step count is incremented. GT LOAD is also generated for each shift, although the GT flag is held cleared if it is in Mode A. Notice that the Link is **not** loaded, and that ADLK DIS L is high. These two conditions mean that the Link is not modified during shifting, but that the output of the Link is coupled to the input of AC0. All other details are similar to those given for the SHL instruction.

1.10 NORMALIZE INSTRUCTION (Figure 1-6)

Normalization is the process by which the 24-bit fixed-point word in the AC and MQ Registers is converted to floating-point format and expressed as a fraction and the corresponding power of two.

The 1-cycle NORMALIZE instruction is completely implemented during TS3 of the FETCH state. Because the final shift count is important to this operation, the Step Counter is initially cleared (zeroed). OMNIBUS signal NOT LAST XFER L is asserted and, at TP3, processor timing comes to a halt and the EAE Timing Generator is started. The Normalize operation only occurs if SHIFT OK H remains H, as determined by comparing AC0 to AC1. If the two are not equal, SHIFT OK H is grounded, thereby causing the EAE timing to halt and restart the processor timing.

As long as AC0 and AC1 are equal, AC, MQ, and Link will shift one place to the left as if they were one long register, as explained for the SHL instruction. Each time a shift occurs, 1 is added to the Step Counter. This continues until the EAE finds AC0 not equal to AC1. Another condition for which the Normalization process is terminated is when AC2-MQ11 are all equal to 0 (the word cannot be normalized). The Normalization process also terminates in Mode B if the 24-bit word in the AC and MQ equals 40000000 (only AC0 is a 1); C0 is grounded during TS3 so that the AC is cleared.

1.11 DOUBLE-PRECISION SKIP IF ZERO (DPSZ) (Figure 1-7)

The 24-bit number in the AC and MQ is tested. If all bits are 0, the next instruction is skipped. If any bit is a 1, the next instruction is executed.

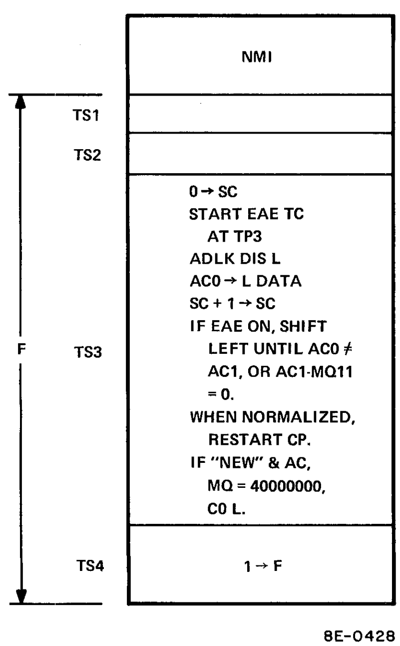


Figure 1-6 NORMALIZE Operation, Flow Diagram

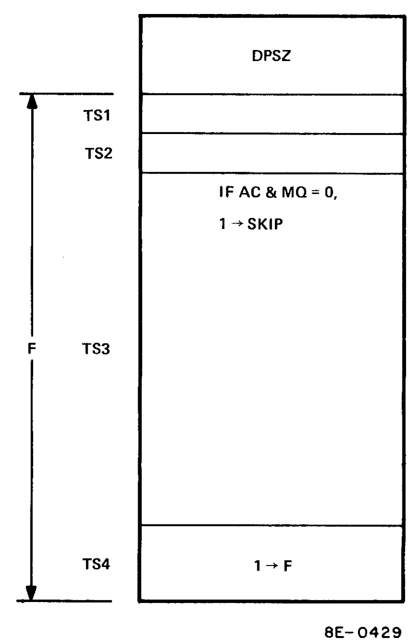


Figure 1-7 DPSZ Instruction, Flow Diagram

1.12 DOUBLE-PRECISION COMPLEMENT (DCM) (Figure 1-8)

The objective of the DCM instruction is to form the 2's complement of the 24-bit word in the AC and MQ. Since the M8300 Major Registers Module is capable of only 12-bit arithmetic, the complete DCM operation requires two passes through the adders. These passes are labeled Step 1 and Step 2 in Figure 1-8 and in the following paragraphs. The entire operation takes place in the FETCH cycle.

The DCM instruction uses the SWP instruction built into the M8310. (One requirement of the DCM instruction is that bit 5, controlling the MQ → AC path, and bit 7, controlling the AC → MQ path, both be 1s.) Thus, as the DCM instruction is decoded, the M8310 causes MQ → BUS L and AC → MQ ENA L (described in Volume 1, Paragraph 3.40). At the KE8-E, two other lines to the M8300 are being controlled. These lines are DATA F L, which is asserted for both operations and CARRY IN L, which is unconditionally asserted for Step 1 and asserted if Link is 1 for Step 2. The KE8-E also disables normal Link gating and places CARRY OUT L from the adders onto the LINK DATA L line of the OMNIBUS. The Link, AC, and MQ are loaded at the conclusion of each step. (AC LOAD and MQ LOAD occurs at the end of Step 1 because of the SWP portion of the instruction.) The processor's timing chain is stopped and the EAE's timing chain is run for one step to provide the extra time and load pulses for Step 2.

One of the more severe tests of the DCM instruction is to perform this operation on a cleared AC and MQ, since such a task requires the carry to propagate through all 24 bits.

1.13 DOUBLE-PRECISION INCREMENT (DPIC) (Figure 1-9)

The DPIC instruction adds 1 to the 24-bit word in the AC and MQ in the same manner as the DCM instruction. The only difference is that DATA F L is not asserted, allowing the contents of the DATA BUS to be applied to the adders without being complemented.

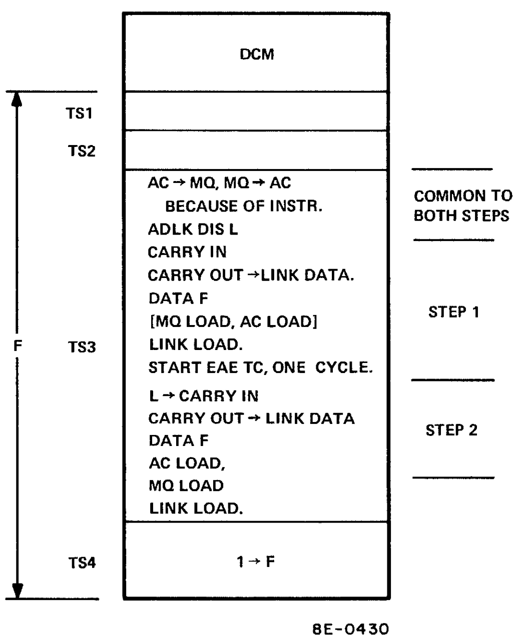


Figure 1-8 DCM Instruction, Flow Diagram

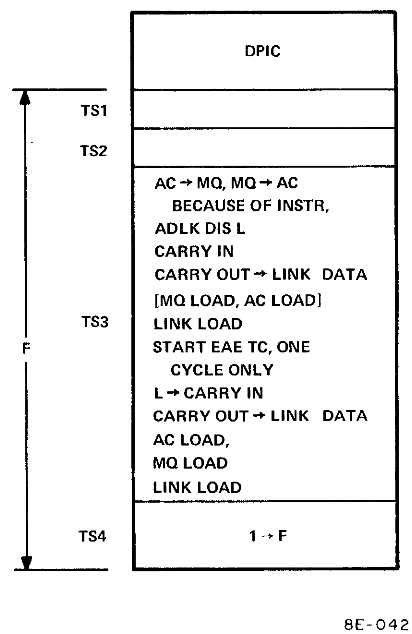


Figure 1-9 DPIC Instruction, Flow Diagram

1.14 DOUBLE-PRECISION STORE (DST) (Figure 1-10)

The contents of MQ and AC are stored at the double-precision location (two consecutive memory locations). The AC, MQ, and Link are not changed by this instruction. When the EAE decodes the DST instruction (Figure 1-10), the next location is accessed in a manner similar to the SCL instruction. Instead of grounding F E SET, however, the DST instruction grounds F D SET, thereby causing the computer to enter the DEFER major state and treat the next location as an address.

At the conclusion of the DEFER cycle, the computer enters the EXECUTE CYCLE. Simultaneously, a flip-flop within the EAE sets. This flip-flop (EX1) grounds F E SET, causing the processor to perform two consecutive EXECUTE cycles and forcing MA + 1 → MA at the end of the first EXECUTE cycle. EX1 is cleared at the end of the first of these EXECUTE cycles, allowing normal processing to resume at the conclusion of the second EXECUTE.

During each of the EXECUTE cycles, the following processor signals are asserted during TS2:

Signal	Function
MQ → BUS DATA T	Gates MQ Register to MB.
MD DIS	Removes the MD, which is normally applied to the MB via the adders.

(AC → MQ ENA L is also grounded, but has no effect since the MQ is not loaded at TP2.)

During TS3, the usual gating is set up to swap the contents of the AC and MQ. Hence, the sequence of events during the EXECUTE cycle is:

a. Store the least-significant twelve bits, presently in the MQ.

b. Swap the AC and MQ.

c. Address the next memory location.

d. Store the most-significant twelve bits presently in the MQ.

e. Swap the AC and MQ to return the bits to their original locations.

1.15 DOUBLE-PRECISION ADD (DAD) (Figure 1-11)

The DAD instruction has many similarities to the DST and DCM instructions. Like the DST instruction, it uses a second memory word as a deferred address. It also requires two EXECUTE cycles to obtain data from two consecutive memory locations. The DAD instruction handles its carry to and from the Link in a manner similar to the DCM instruction.

During TS3 of the FETCH cycle, ADLK DIS is grounded, enabling the OMNIBUS LINK DATA and LINK LOAD inputs. At TP3, LINK LOAD is generated. Since no data was placed on LINK DATA, the Link is cleared. Other than clearing the Link, the DAD instruction process is identical to that of the DST instruction for the first two machine cycles.

During each of the two EXECUTE cycles, a word is obtained from memory and applied to the adders via the MD lines. During TS3, the output of the Link is applied to the carry input of the adders. The contents of the MQ are gated to the other inputs to the adders. The carry output of the adders is applied to the LINK DATA input

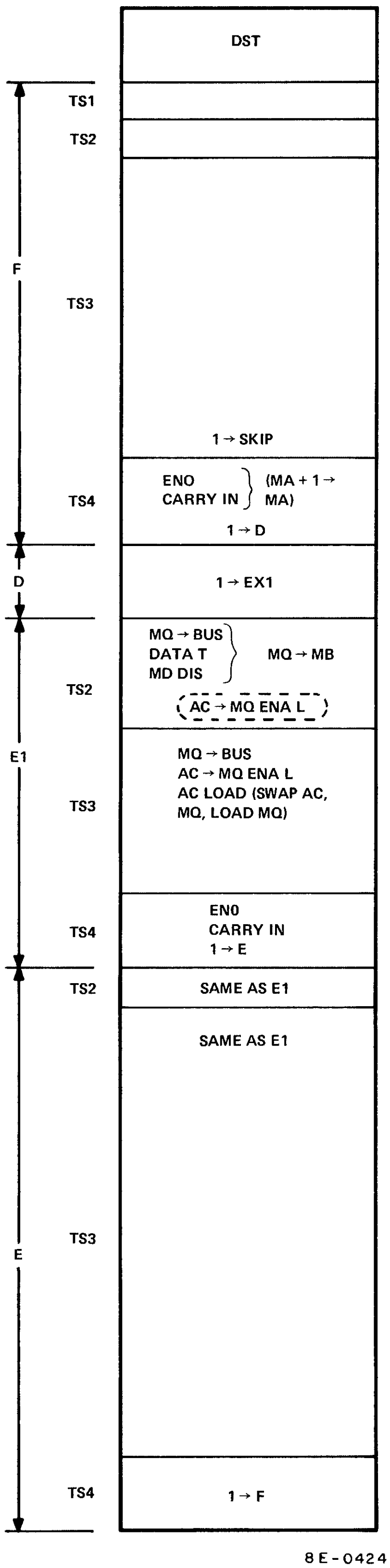


Figure 1-10 DST Instruction, Flow Diagram

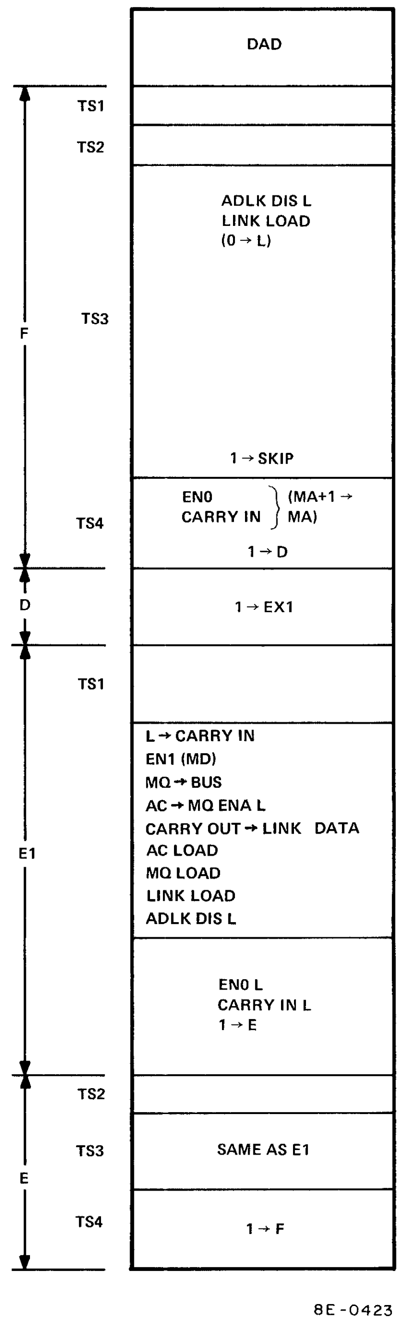


Figure 1-11 DAD Instruction, Flow Diagram

of the Link; the path from the AC to MQ is enabled. At TP3 the AC, Link, and MQ are loaded. Hence, the old AC is moved to the MQ, while the sum of the old MQ and the MD is loaded into the AC. The Link provides and receives carry information.

1.16 SUBTRACT AC FROM MQ (SAM) (Figure 1-12)

The SAM instruction subtracts AC from MQ and places the result in the AC, Link, and GT flag. The MQ is not modified. The entire operation takes place during the FETCH cycle.

The MQ is gated to the adders by grounding source control lines EN0 and EN2 at the H851 Connectors. As listed in Table 3-4 of Volume 1, the MQ Register is gated to one of the sets of adder inputs. The AC is complemented and introduced to the other set of adder inputs via the DATA BUS by grounding AC → BUS, DATA F, and CARRY IN. A "Greater Than" signal is generated and applied to the GT flag. The carry from the address is applied to the Link inputs. The AC, Link, and GT are loaded, completing the operation.

The "Greater Than" signal is derived as follows:

a. If the MQ and the old AC are of different signs, the MQ is greater than the AC if the MQ is positive. The MQ is less than the AC if the MQ is negative.

b. If the MQ and the old AC are of the same sign, the MQ is greater than (or equal to) the old AC if the output of the most-significant bit of the adder is positive. Otherwise, the MQ is less than the AC.

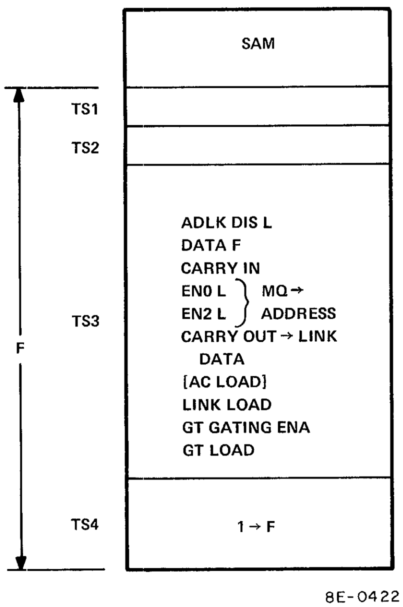


Figure 1-12 SAM Instruction, Flow Diagram

Logic at the input of the GT flag computes the "Greater Than" signal.

1.17 MULTIPLY INSTRUCTION (MUY) (Figure 1-13)

The MUY instruction combines the multiplicand (which was previously loaded into the MQ Register) with a multiplier (obtained from memory by the MUY instruction), using the rules of binary multiplication. The result is left in the AC and MQ. The multiplication requires twelve (decimal) steps which are counted by the Step Counter. At each step, MQ11 is examined. If it is a 1, the multiplier is added to the AC. Regardless of the state of MQ11, the AC and MQ are shifted right in the same manner as is done for the LSR instruction, except that the GT flag is not loaded. This same process is repeated for the new MQ11 until the twelve steps have been completed. At this point, the AC and MQ contains the 24-bit product.

The MUY instruction requires one FETCH cycle to fetch the instruction, one DEFER cycle (Mode B only) to obtain the multiplier address, and one EXECUTE cycle to obtain the multiplier and accomplish the multiply operation.

The decoded instruction clears the Step Counter and places a 0 in the Link by asserting ADLK DIS L and LINK LOAD L. It then accesses the next location in memory (refer to SCL instruction). If the older, PDP-8/I compatible, Mode A instruction set is in use, the next sequential address contains the multiplier. The EAE, therefore, grounds F E SET L and goes directly to the EXECUTE cycle. If the EAE is in Mode B, F D SET L is grounded and the processor enters the DEFER state for the address of the multiplier. At the conclusion of the DEFER cycle, the processor automatically enters the EXECUTE state.

During the first part of the EXECUTE cycle, the multiplier is read onto the MD lines. At TS3, NOT LAST XFER L is asserted on the OMNIBUS; at TP3, processor timing halts. The EAE timing chain is then started. The right-shift signals are asserted (refer to the LSR instruction for further details). Each time MQ11 is a 1, EN1 is grounded by the EAE. If EN1 is ground, EN0 is grounded by the M8310 Major Registers Control Module. Grounding EN1 and EN0 causes the word on the MD lines to be added to the partial product. This process continues for the twelve steps necessary to complete the multiplication. The last step is made with NOT LAST XFER high, causing the processor to resume its timing.

The data paths for the MUY instruction are illustrated in Figure 1-14. This figure also illustrates the control signals that must be enabled to make this instruction possible.

1.18 DIVIDE INSTRUCTION (DVI) (Figure 1-15)

There are two common methods of doing binary division:

a. Restoring divide (the standard long-hand method): Try to subtract. If the result is +, place a 1 in the quotient. If the result is –, the subtraction does not take place; place a 0 in the quotient. In either case, shift left.

b. Nonrestoring divide (the method used in PDP-8/E): Always make the subtraction and always shift left. If the result is +, place a 1 in the quotient; the next step will also be a subtract. If the result is –, place a 0 in the quotient; the next step will be an add. This method requires a final correction step if the final remainder is –.

Figure 1-15 illustrates the DVI Instruction. This instruction requires one FETCH, one DEFER (Mode B only), and one EXECUTE cycle. The instruction clears the Step Counter at TP3 of the FETCH cycle. The next memory location is accessed, as explained for the SCL instruction. If the instruction mode is B, the CPU must obtain the operand address by entering the DEFER major state. Otherwise, the CPU goes directly into the EXECUTE state.

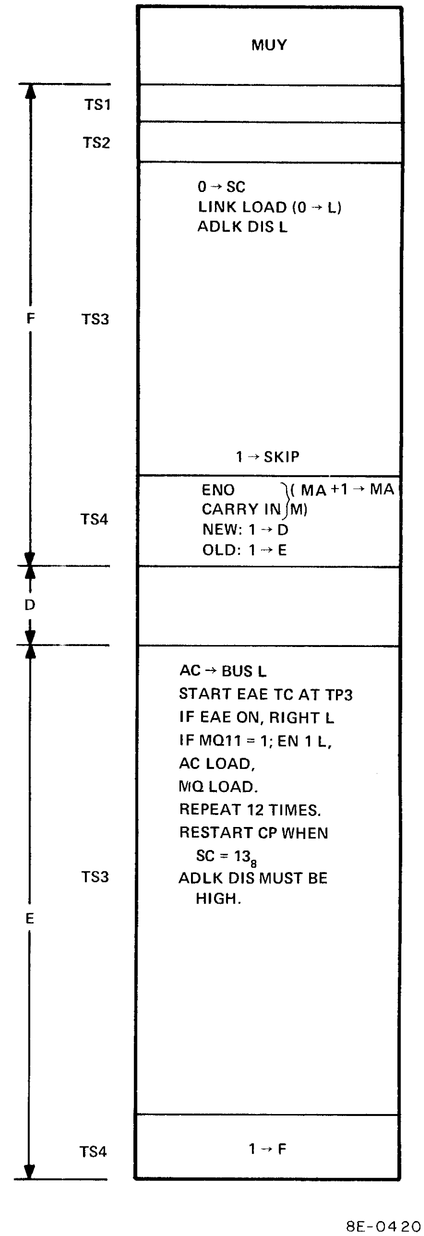


Figure 1-13 MUY Instruction/ Operation, Flow Diagram

The first subtraction takes place at TP3 of the EXECUTE cycle, *before the EAE is turned on.* At the same time, the Link* is set. The EAE is turned on only if there is a carry from the most-significant bit of the adder. Otherwise, a condition known as divide overflow exists, and the quotient cannot be contained in the 12 bits available. If the EAE is turned on, the last divide step clears the Link. Thus, the Link is used as a program flag to indicate whether or not divide overflow occurred.

*Link refers to Processor Link, DIV LNK refers to EAE Link.

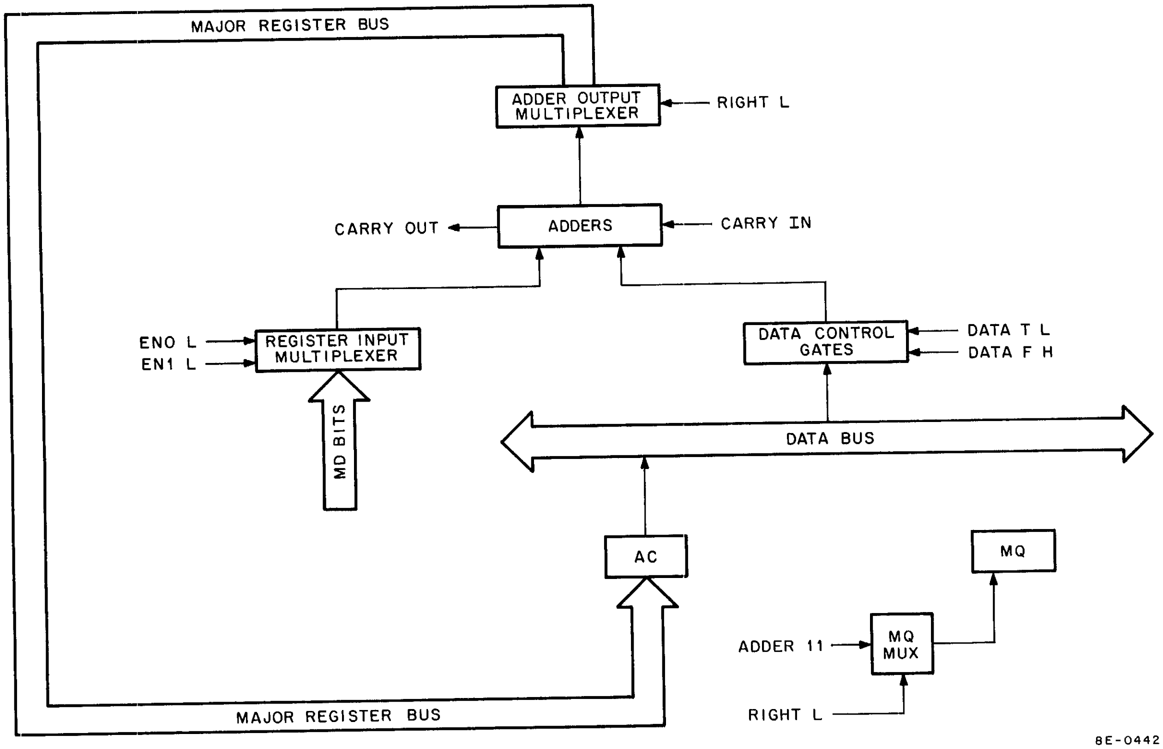


Figure 1-14 MUY Instruction Data Paths

The Major Registers are shown in Figure 1-16. This figure, an expansion of Figure 3-79 of Volume 1, shows the signals that are important to the divide process. Notice that the M8300 has no provision for complementing the MD. The only means of complementing is via the data control gates which are in the AC shift path. In order to subtract, the KE8-E must cause $\overline{AC}$ plus MD → AC. The AC now contains the complement of the result. Successive subtractions merely cause AC plus MD → AC, since the AC is already in complemented form. To change from subtraction to addition, the KE8-E must cause $\overline{AC}$ plus MD → AC. Of course, successive adds are performed by AC plus MD → AC. Complementing is accomplished by grounding DATA F, and must be performed each time the quotient bit changes. The logic merely grounds DATA F if MQ10 ≠ MQ11 after the first two divide steps have established quotient bits in MQ10 and MQ11. DATA F is grounded for the first two steps.

As explained above, AC may be in its true or complemented form as the divide operation progresses. The MQ is always in its true form, and is the source of unprocessed dividend bits that are shifted into AC11. If the word being loaded into the AC is in complemented form, MQ0 must be complemented as it is shifted into AC11. The logic merely examines MQ11. If it is a 1, MQ0 is complemented as it is shifted into AC11.

The fundamental rule governing the quotient bit is as follows:

> If the sign of the dividend does not change, MQ11 → MQ11.
>
> If the sign of the dividend changes, $\overline{\text{MQ11}}$ → MQ11.
>
> But the sign might have changed because the logic grounded DATA F, so the fundamental must be expanded to:
>
> If DATA F H and no sign change, or if DATA F L and the sign changes, MQ11 → MQ11.

If DATA F H and the sign changes, or if DATA F L and no sign change,
MQ11 → MQ11.

Since DATA F L is caused by MQ10 ≠ MQ11, a little Boolean manipulation yields:

If MQ10 = 0 and no sign change, or if MQ = 1 and the sign changes, 0 → MQ11.

If MQ10 = 0 and the sign changes, or if MQ10 = 1 and no sign change, 1 → MQ11.

The sign change is derived from DIV LNK (the AC sign bit after the shift) XORed with CARRY OUT. All combinations of MQ10, MQ11, DIV LNK and CARRY OUT are shown in Table 1-1, together with the resulting quotient bit.

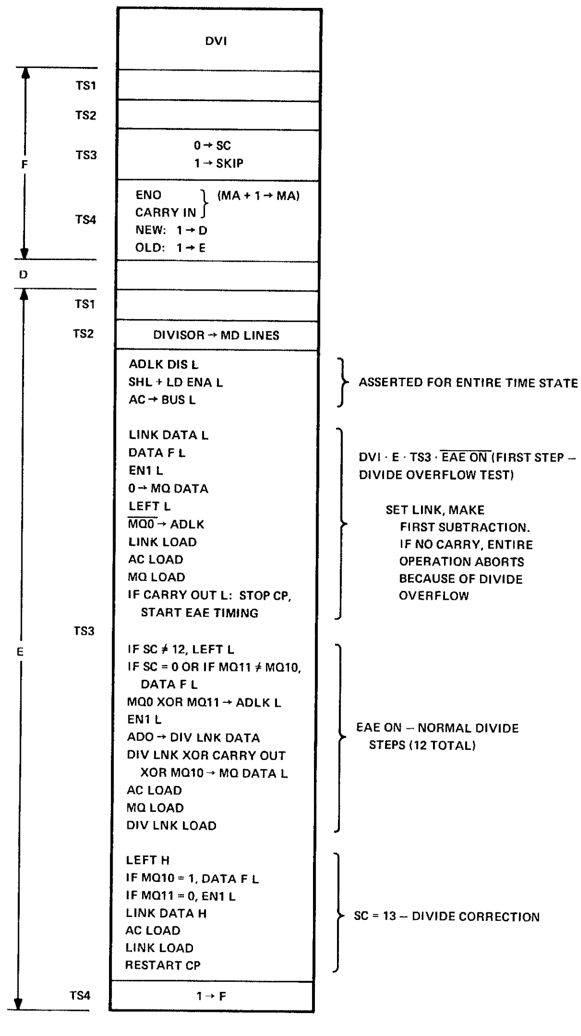


Figure 1-15 DIVIDE Instruction, Flow Diagram

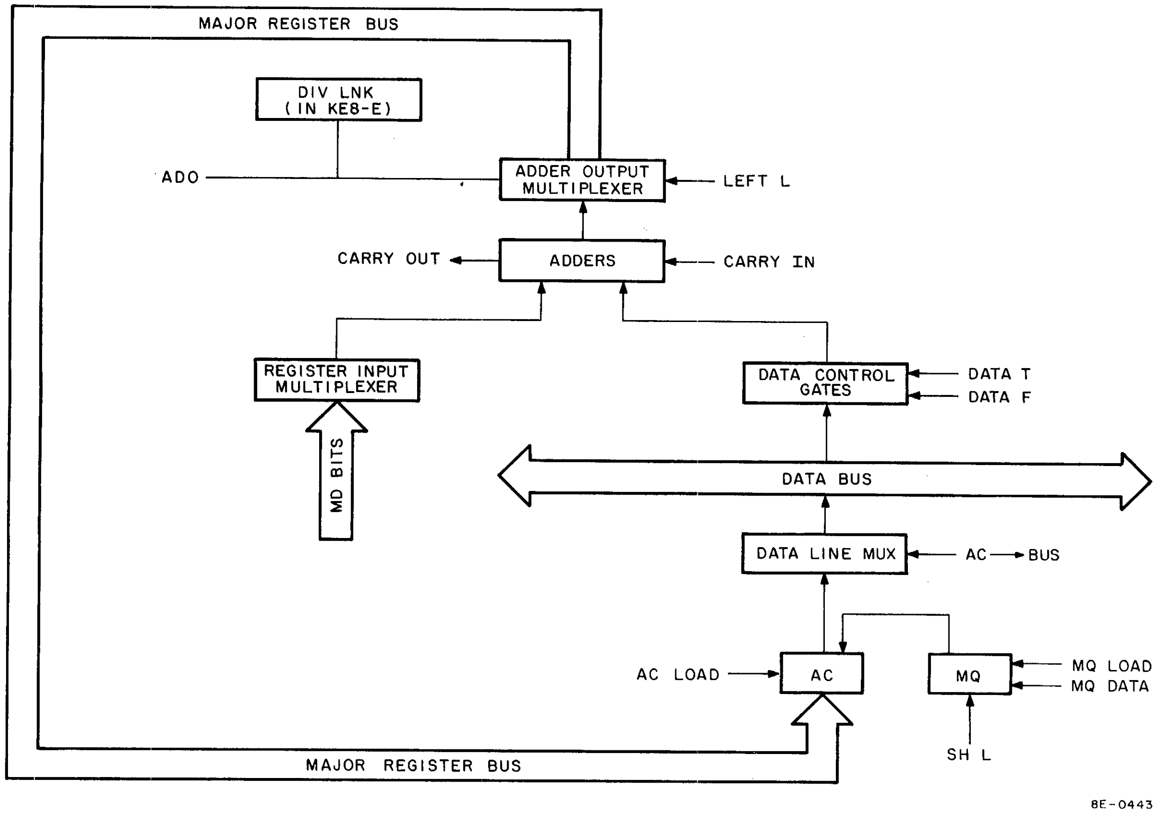
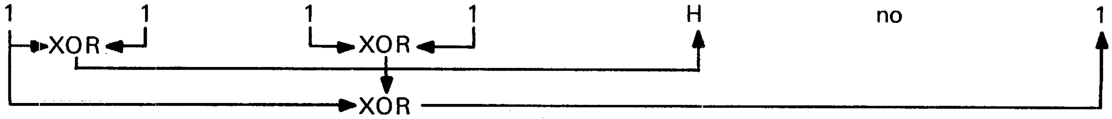


Figure 1-16 Major Registers

Table 1-1
Divide Instruction Table of Combinations

MQ10	MQ11	DIV LNK	CARRY OUT	DATA F	SIGN CHANGE	What goes into MQ11 (MQ DATA)
0	0	0	0	H	no	0
0	0	0	1	H	yes	1
0	0	1	0	H	yes	1
0	0	1	1	H	no	0
0	1	0	0	L	no	0
0	1	0	1	L	yes	1
0	1	1	0	L	yes	1
0	1	1	1	L	no	0
1	0	0	0	L	no	1
1	0	0	1	L	yes	0
1	0	1	0	L	yes	0
1	0	1	1	L	no	1
1	1	0	0	H	no	1
1	1	0	1	H	yes	0
1	1	1	0	H	yes	0
1	1	1	1	H	no	1



Carry	Link	Accumulator	Multiplier Quotient	Step Counter	Comments
	0	000 000 000 000	000 010 010 001		
		111 111 111 111			$\overline{AC}$ → ADDERS (FIRST STEP)
		000 000 001 100			MD → ADDERS
1		000 000 001 011			
	0	000 000 010 111	000 100 100 010	00 000	MQ0 → AC11
	0	111 111 101 000			
		000 000 001 100			
0		111 111 110 100			
	1	111 111 101 000	001 001 000 100	00 001	
		111 111 101 000			AC → ADDERS (MQ10 = MQ11)
		000 000 001 100			
0		111 111 110 100			
	1	111 111 101 000	010 010 001 000	00 010	
		111 111 101 000			
		000 000 001 100			
0		111 111 110 100			
	1	111 111 101 000	100 100 010 000	00 011	
		111 111 101 000			
		000 000 001 100			
0		111 111 110 100			
	1	111 111 101 001	001 000 100 000	00 100	
		111 111 101 001			
		000 000 001 100			
0		111 111 110 101			
	1	111 111 101 010	010 001 000 000	00 101	
		111 111 101 010			
		000 000 001 100			
0		111 111 110 110			
	1	111 111 101 100	100 010 000 000	00 110	
		111 111 101 100			
		000 000 001 100			
0		111 111 111 000			
	1	111 111 110 001	000 100 000 000	00 111	
		111 111 110 001			
		000 000 001 100			
0		111 111 111 101			
	1	111 111 111 010	001 000 000 000	01 000	
		111 111 111 010			
		000 000 001 100			
1		000 000 000 110			
	0	000 000 001 100	010 000 000 001	01 001	
		111 111 110 011			
		000 000 001 100			$\overline{AC}$ → ADDERS (MQ10 ≠ MQ11)
0		111 111 111 111			
	1	111 111 111 111	100 000 000 011	01 010	
		111 111 111 111			AC → ADDERS (MQ10 = MQ11)
		000 000 001 100			MQ0 → AC11 (MQ11 = 1)
1		000 000 001 011			
	0	000 000 010 110	000 000 000 110	01 011	
		111 111 101 001			$\overline{AC}$ → ADDERS (MQ10 ≠ MQ11)
		000 000 001 100			
0		111 111 110 101			
	1	111 111 110 101	000 000 001 100	01 100	
		111 111 110 101			Since SC = 15_8, MQ10 = 0, and MQ11 = 0;
		000 000 001 100			MQ plus AC → AC
1		000 000 000 001			(Divide Correction).
	0	000 000 000 001	000 000 001 100	01 101	NO SHIFT

Figure 1-17 Divide Example — Divides 221_8 by 14_8

Thirteen divide steps take place (the first step tests for divide overflow; the next twelve steps determine the quotient). A final remainder correction step is made as the CPU is restarted and the Link is cleared. For the correction step, the left-shift signals are all negated. If MQ10 = 1, the last regular divide step was a subtract. The AC is in complemented form before the correction step; hence DATA F must be grounded to re-complement the AC to its true form as a part of the correction process. If MQ11 = 0, the divisor must be added to the remainder (this is the correction step mentioned in the first part of this section).

Figure 1-17 shows an example of the division process.

SECTION 4 DETAILED LOGIC

1.19 EAE INSTRUCTION DECODING LOGIC

The EAE instruction decoding logic consists of the EAE Instruction Register (EIR), the MODE flip-flop, and ROM 1 and ROM 2. The decoding logic recognizes EAE commands from the processor and interprets them in terms of EAE instructions.

1.19.1 EIR Register

The EIR Register (Figure 1-18) comprises 12 D-type flip-flops (IC 74H74). It is loaded at TP2 of the FETCH major state with the 12 Memory Data bits (MD0—11) and provides outputs of EIR N(1) or EIR N(0), where N corresponds to the EIR bit designation. The most active EIR bits correspond to bits MD7—10 and play a dominant role in the EAE coding scheme. If the system is about to answer an interrupt and is not doing a data break, all flip-flops are cleared at TP4. Otherwise, the flip-flops are cleared at TP1 of FETCH.

1.19.2 MODE Flip-Flop Logic

The MODE flip-flop (Figure 1-19) comprises one J-K flip-flop (IC 74H106) which responds to SWAB and SWBA instructions. Two modes of operation were designed into the EAE to accommodate the user having programs that were written for a PDP-8/I or to accommodate the new user. Mode A corresponds to the PDP-8/I type software; Mode B corresponds to the new instructions that are provided. The EAE always starts in Mode A. The MODE flip-flop allows the programmer to switch modes at his convenience. The flip-flop is clocked at the trailing edge of TP2 whenever the basic EAE instruction in a FETCH state is decoded.

1.19.3 ROM Logic

The ROM logic (Figure 1-20) consists of two ICs, each containing a 32 8-bit word capability and selected by the combination of 5 inputs.

Figures 1-21 and 1-22 illustrate ROM operation. ROM 1 is enabled during either a FETCH or EXECUTE cycle, when an EAE instruction has been decoded and the instruction is not a mode-swapping instruction. ROM 2 is enabled during a FETCH cycle, when an EAE instruction has been decoded and the instruction is not a mode-swapping instruction. Each EAE instruction can be easily traced to eight output ROM signals, each representing a specific command to somewhere in the EAE logic. An indication of what each output is doing and where it is going can be seen at the bottom of each matrix. For the purpose of ROM decoding a 0 can be considered active and function as a 1 in normal logic terminology. For example, ROM 26 L causes an MA plus 1 to the MA. The specific EAE instructions causing this ROM instruction can be seen on the matrix.

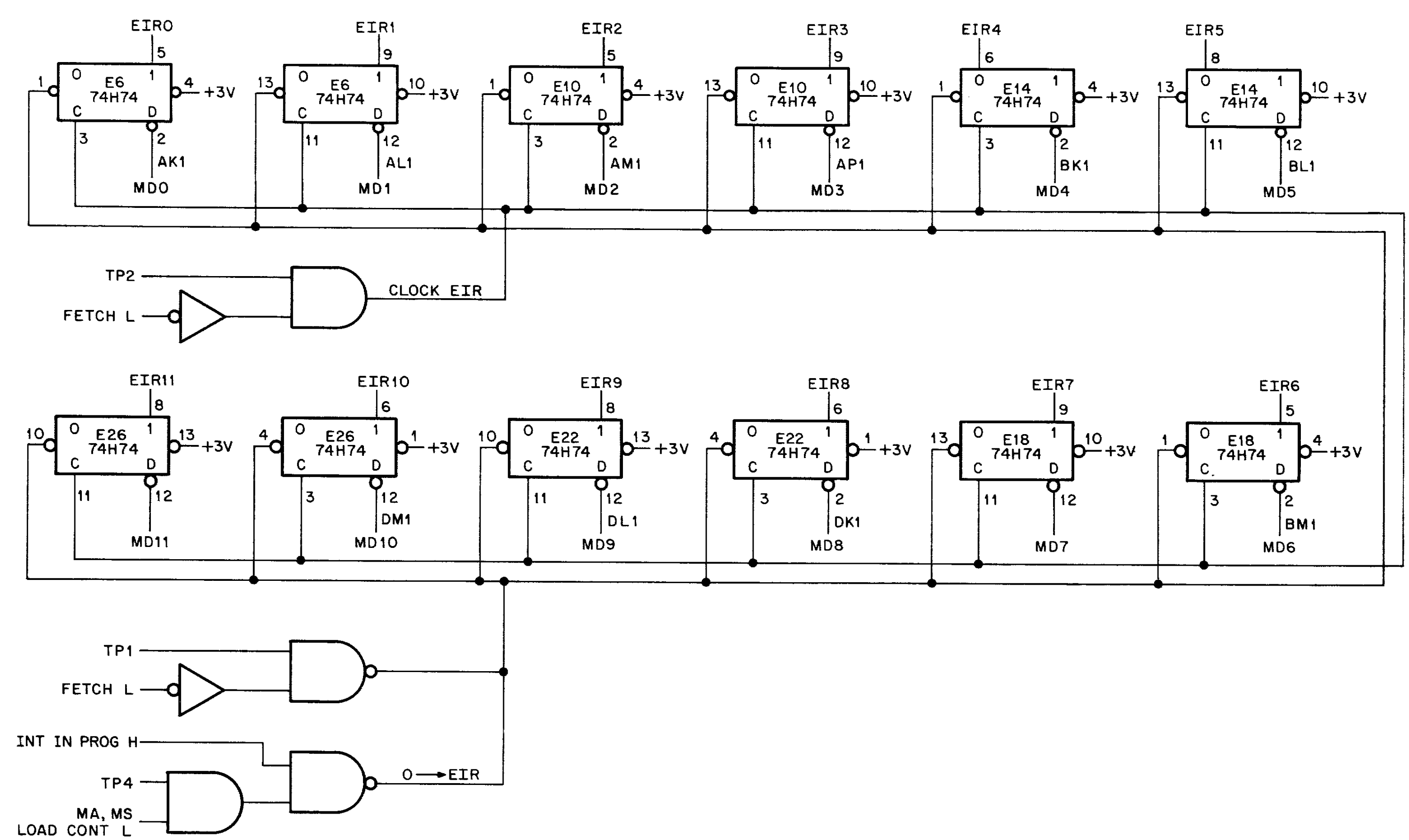


Figure 1-18 EIR Register and Controls

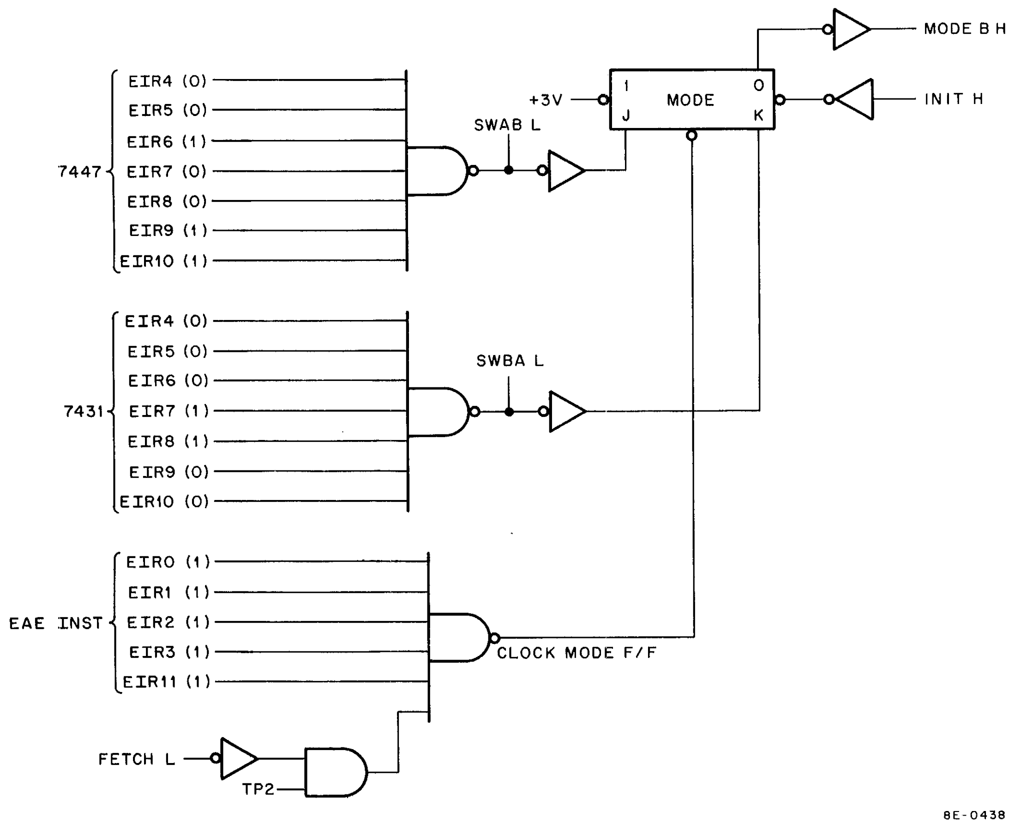


Figure 1-19 MODE Flip-Flop Logic

1.20 EAE TIMING LOGIC

The EAE timing logic is illustrated in Figure 1-23. The components consist of six D-type flip-flops, TG1 through TG4, an E SYNC flip-flop, and an EAE ON flip-flop, plus a variety of control and input gates. Flip-flops TG1 through TG4 are configured as a switch-tail ring counter. TG2(1), TG3(1) and TG4(0) are used to clock major events in the EAE logic. For example, TG2(1) L and TG3(1) L are combined to form ETP (EAE Time Pulse), which is the primary clock pulse to step the Step Counter and load registers.

The length of the switch-tail ring counter is controlled by ROM 12 L, which indicates whether an add (and possibly a shift) or merely a shift operation is taking place. If adds are taking place, the EAE Timing Generator must run at a slower rate to allow time for carries to propagate in the adders of the M8300 Major Registers Module. If ROM 12 L is high, TG1 is disabled and TG4 shifts (complemented) into TG2. Six clock pulses are required to complete the timing generator cycle; hence, ETPs are 300 ns from leading edge to leading edge. If ROM 12 L is low, TG1 is in the Shift Register. The ETPs are then 400 ns from leading edge to leading edge.

1.20.1 EAE Timing Generator Timing Diagram

A timing diagram (Figure 1-24) relates the transition from processor timing to EAE timing. The signal NOT LAST XFER L, which is grounded when the processor is to stop, is not shown on the diagram (refer to Paragraph 1.25 for information on the EAE start/stop logic). NOT LAST XFER L is asserted at TP2D; at the leading edge of

TP3, processor timing halts. The EAE Timing Generator operation begins on the leading edge of TP3, which dc sets the E SYNC flip-flop. EAE timing begins when flip-flop EAE ON is set; the timing chain is started on the next 20-MHz clock input from the processor timing generator. The first ETP occurs when TG4 is set and TG2 is reset.

ETP occurs once every 300 ns or 400 ns (depending upon ROM 12 L) and continues as long as LAST STEP, or SHIFT OK, or DCM + DPIC is not low. Any one of these signals will cause a 0 to be clocked into the E SYNC flip-flop, thus beginning a series of events that ends the EAE timing and restarts the processor timing.

Each time an ETP is generated by timing, the Step Counter is stepped one more time until the total number of shifts have been completed.

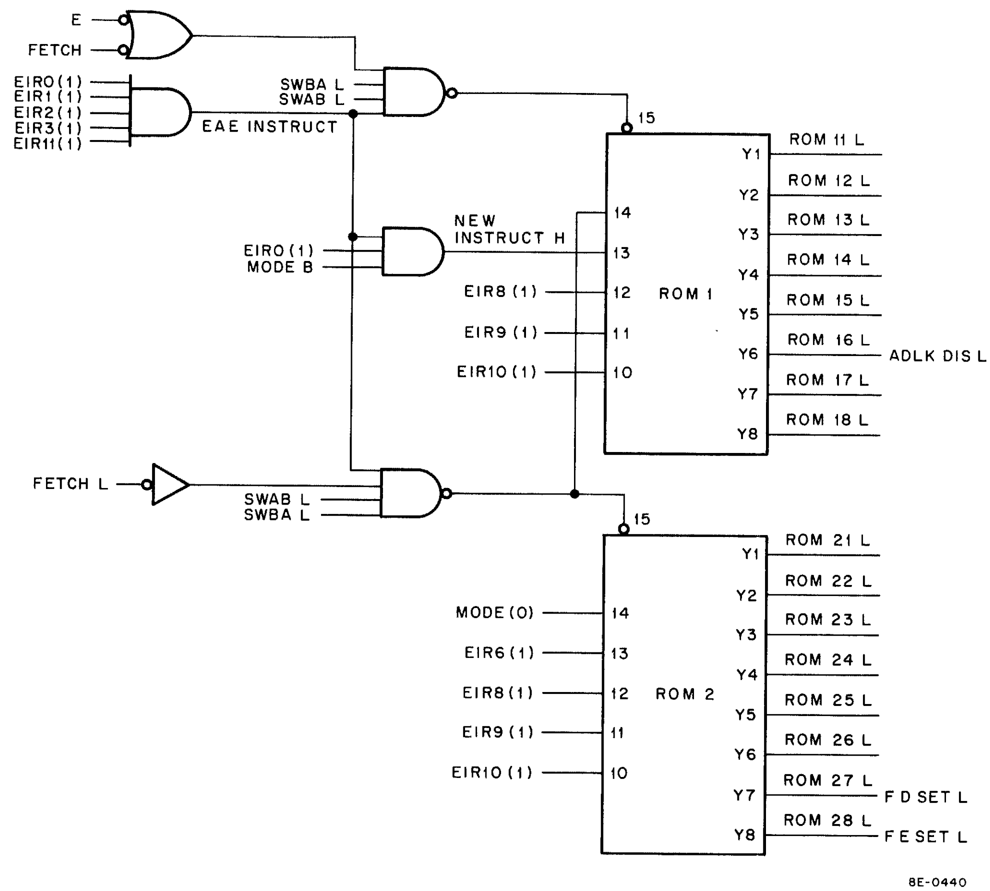


Figure 1-20 ROM Logic

1.21 EAE SOURCE CONTROL SIGNALS

EAE demands on the processor are more extensive than most other options. Data can be selected from the AC, MQ, MD, MB, PC or from the CPMA Register. How the data is selected and its source are illustrated in Figure 1-25. The AC or the MQ can be applied to one set of adder inputs via the DATA BUS. The MD, MQ, PC, or the CPMA Register can be applied to the second set of adder inputs.

E D C B A	FUNCTION	Y1	Y2	Y3	Y4	Y5	Y6	Y7	Y8	OCTAL
ØØ	F· NOP	1	1	1	1	1	1	1	1	377
Ø1	F·(ACS + SCL)	1	1	1	1	1	1	1	1	377
Ø2	F· MUY	1	1	1	1	1	0	0	1	371
Ø3	F· DVI	1	1	1	1	1	1	1	1	377
Ø4	F· NMI	0	1	1	0	0	0	1	1	143
Ø5	F· SHL	1	1	1	1	1	1	1	1	377
Ø6	F· ASR	0	1	1	1	1	0	0	1	171
Ø7	F· LSR	1	1	1	1	1	0	0	1	371
1Ø	F· SCA	1	1	1	1	1	1	1	1	377
11	F· DAD	1	1	1	1	1	0	0	1	371
12	F· DST	1	1	1	1	1	1	1	1	377
13	NOP	1	1	1	1	1	1	1	1	377
14	F·DPSZ	1	1	1	1	1	1	1	1	377
15	F· DPIC	1	0	0	1	1	0	0	1	231
16	F· DCM	1	0	0	1	1	0	0	1	231
17	F· SAM	1	1	0	1	1	0	0	1	331
2Ø	E· NOP	1	1	1	1	1	1	1	1	377
21	E· SCL	1	1	1	1	1	1	1	0	376
22	E· MUY	1	0	1	1	0	1	1	1	267
23	E· DVI	1	0	1	0	0	0	0	1	241
24	NOT USED									X
25	E· SHL	0	1	1	0	0	0	1	0	142
26	E· ASR	1	1	1	1	0	1	1	0	366
27	E· LSR	1	1	1	1	0	1	1	0	366
30	NOT USED									X
31	E· DAD	1	1	0	1	1	0	0	1	331
32	E· DST	1	1	1	1	1	1	1	1	377
33	NOT USED									X
34	NOT USED									X
35	NOT USED									X
36	NOT USED									X
37	NOT USED									X

- Y1: Ø INDICATES ACØ→LINK DATA AT TS3
- Y2: Ø INDICATES TG SLOW—ADD (NOT MERELY SHIFT)
- Y3: Ø INDICATES CARRY COUPLE AT TS3 (L→CARRY IN, CARRY OUT→L DATA)
- Y4: Ø INDICATES LEFT SHIFT
- Y5: Ø INDICATES A SHIFT OPERATION
- Y6: Ø DISABLES CPU ADDER LINK GATING
- Y7: Ø INDICATES LINK LOAD AT TP3
- Y8: Ø INDICATES LOAD SC AT TP2

Figure 1-21 ROM 1 Instructions

FUNCTION	Y1	Y2	Y3	Y4	Y5	Y6	Y7	Y8	OCTAL
NOP	1	1	1	1	1	1	1	1	377
ACS	0	1	1	1	1	1	1	1	177
NEW MUY	1	1	0	1	1	0	0	1	331
NEW DVI	1	1	0	1	1	0	0	1	331
NMI	1	1	0	1	1	1	1	1	337
SHL	1	1	1	1	1	0	1	0	372
ASR	1	1	1	1	1	0	1	0	372
LSR	1	1	1	1	1	0	1	0	372
SCA	1	1	1	1	0	1	1	1	367
DAD	1	1	1	1	1	0	0	1	371
DST	1	1	1	1	1	0	0	1	371
NOP	1	1	1	1	1	1	1	1	377
DPSZ	1	1	1	0	1	1	1	1	357
DPIC	1	0	1	1	1	1	1	1	277
DCM	1	0	1	1	1	1	1	1	277
SAM	1	0	1	1	1	1	1	1	277
NOP	1	1	1	1	1	1	1	1	377
SCL	1	1	1	1	1	0	1	0	372
OLD MUY	1	1	0	1	1	0	1	0	332
OLD DVI	1	1	0	1	1	0	1	0	332
NMI	1	1	0	1	1	1	1	1	337
SHL	1	1	1	1	1	0	1	0	372
ASR	1	1	1	1	1	0	1	0	372
LSR	1	1	1	1	1	0	1	0	372
SCA	1	1	1	1	0	1	1	1	367
SCA•SCL	1	1	1	1	0	0	1	0	362
SCA•OLD MUY	1	1	0	1	0	0	1	0	322
SCA•OLD DVI	1	1	0	1	0	0	1	0	322
SCA • NMI	1	1	0	1	0	1	1	1	327
SCA • SHL	1	1	1	1	0	0	1	0	362
SCA • ASR	1	1	1	1	0	0	1	0	362
SCA • LSR	1	1	1	1	0	0	1	0	362

Ø INDICATES ACS

Ø INDICATES DCM+SAM+DPIC

Ø INDICATES Ø→SC AT F • TP3

Ø INDICATES DPSZ

Ø INDICATES SCA

Ø INDICATES MA+ 1→MA, 1→SKIP

Ø INDICATES DSET

Ø INDICATES ESET

Figure 1-22 ROM 2 Instructions

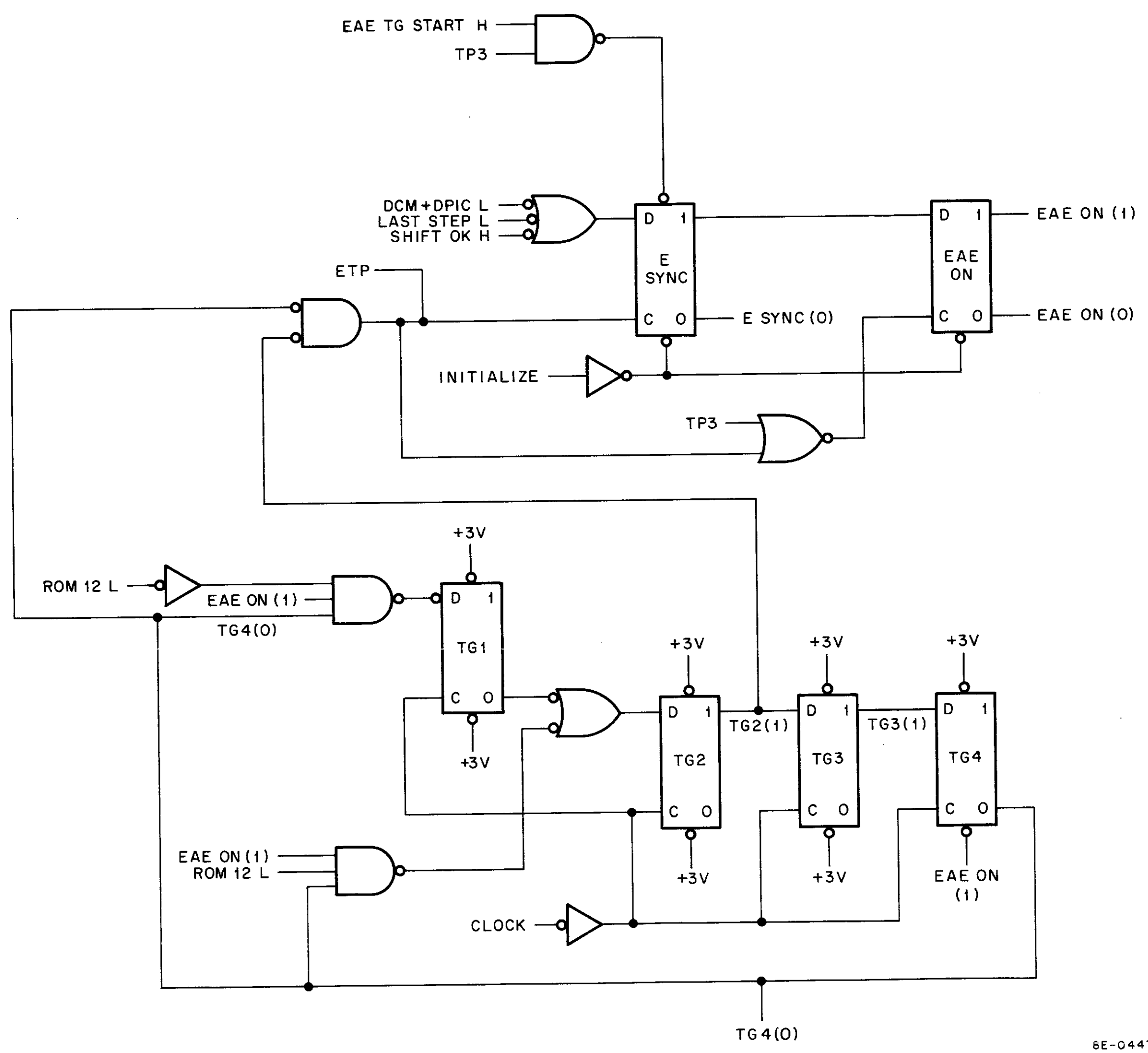


Figure 1-23 EAE Timing Logic

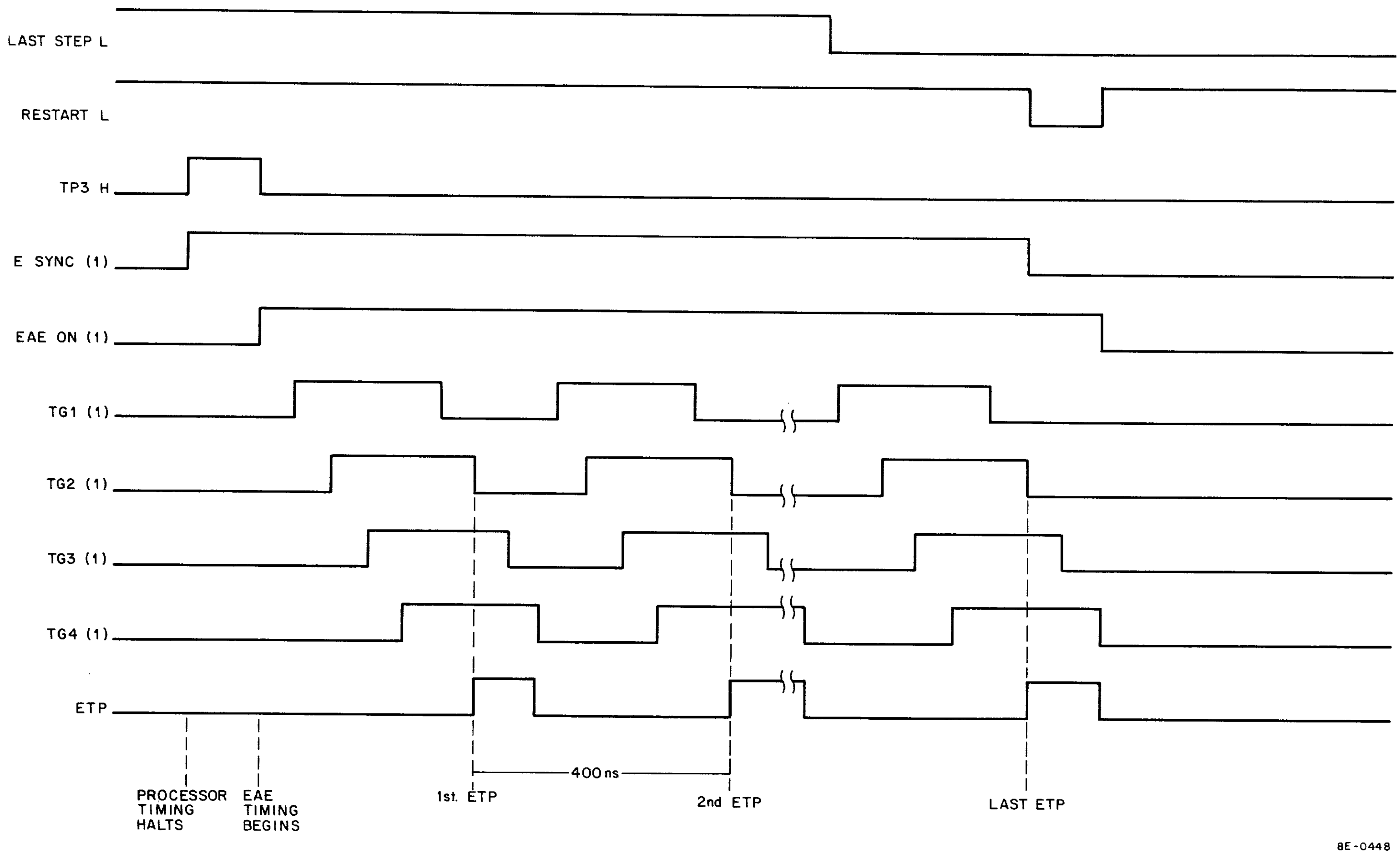


Figure 1-24 EAE Timing Generator Timing Diagram

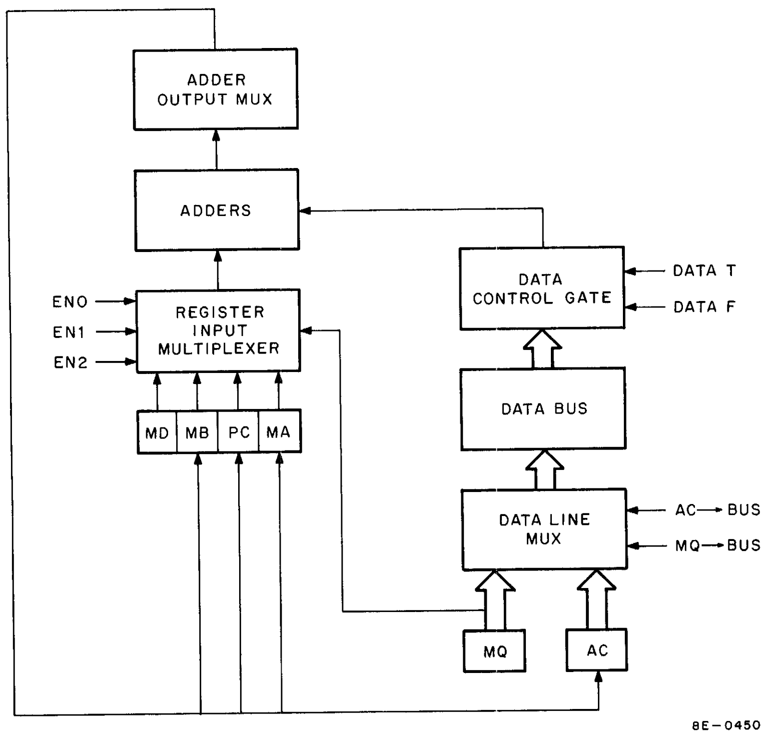


Figure 1-25 Source Control Data Path, Simplified Block Diagram

1.21.1 Register Input Enable Signals

The processor register selection logic is illustrated in Figure 1-26. Signals EN0, EN1, and EN2 determine what data will pass through the Register Input Multiplexer. The data that is selected is illustrated by the decoding scheme shown in Table 1-2.

Table 1-2
Register Select Decoding Scheme

EN0	EN1	EN2	Register or Data Selected
low	low	low	PC Register (not selected by KE8-E)
low	low	high	MD Lines
low	high	low	MQ Register
low	high	high	CPMA Register

NOTE
When EN1 is grounded by EAE, gating within the M8310 Major Registers Control automatically grounds EN0. Thus, the EAE need only ground EN1 to select MD to the adders.

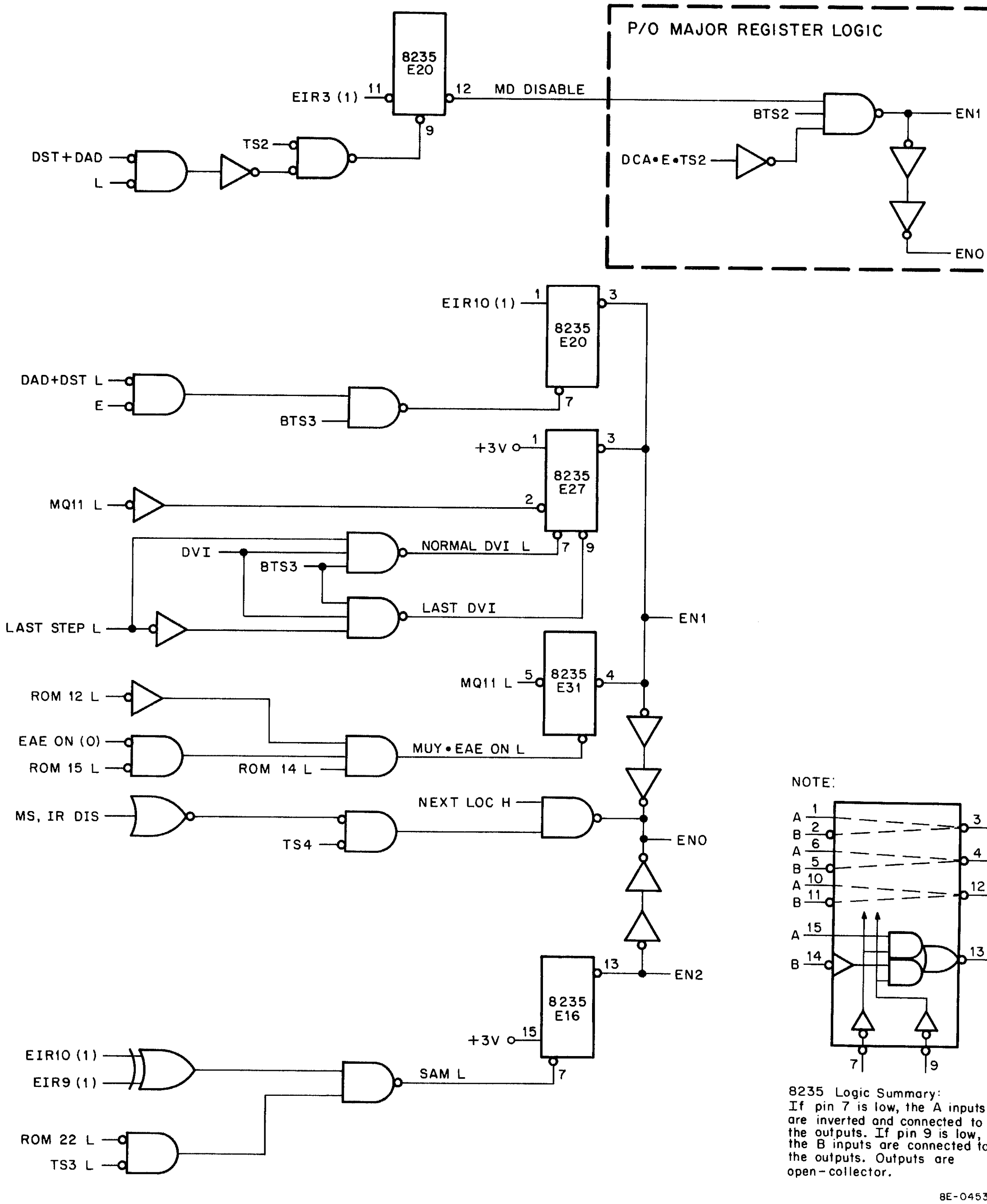


Figure 1-26 Processor Register Selection Logic

1.21.2 Data Line Enable Signals

As illustrated in Figure 1-25, signals AC → BUS and MQ → BUS are used to gate either the contents of the AC Register or the MQ Register to the DATA BUS. The generation of these two signals is shown in Figure 1-27. Signal AC → BUS occurs during all shift instructions, including MUY and DVI, as dictated by signal ROM 15 L during TS3. Signal MQ → BUS is asserted by E and NEW INSTR. E and NEW INSTR also generate AC → MQ ENA L. This arrangement automatically transfers the contents of the MQ Register to the DATA BUS and the contents of the AC Register to the MQ. The DAD and DST instructions follow this procedure. There are other times (DCM and DPIC) when MQ → BUS and AC → MQ L are asserted, but these situations are handled in the M8310 by the MQA and MQL bits (refer to Volume 1).

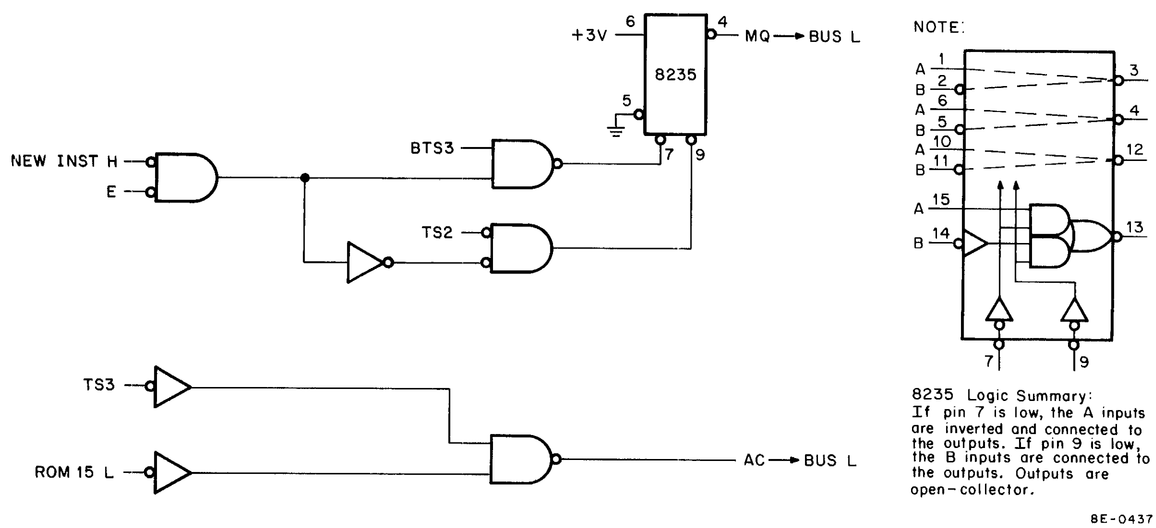


Figure 1-27 Data Line Enable Signals

1.21.3 Data Enable Signals

Signals DATA T and DATA F allow data to move from the DATA BUS to the adders. The type of data being applied to the adders is illustrated in Table 1-3.

Table 1-3
EAE Combinations of DATA T and DATA F

Signal		Type of Data Applied to Adders
DATA T	DATA F	
low	low	Complement of contents of DATA BUS
low	high	Contents of DATA BUS
high	low*	Zero

*DATA F is grounded by a gate in the M8310 Register Control if DATA T is high.

The data usually placed on the DATA BUS by the EAE option will be the contents of the AC Register or the MQ Register. Signals DATA T and DATA F can be asserted by either the Major Registers Control logic or the EAE data control logic (Figure 1-28). Signal DATA T is brought low during TS2 when a DAD + DST instruction is being executed during an EXECUTE cycle. However, DATA T is also brought low during TS3 of all EAE cycles, as described in Paragraph 3.35.3 of Volume 1.

DATA F is pulled low by a SAM, DCM, or DPIC instruction during TS3. During a normal DVI, DATA F is low if MQ10 and MQ11 are alike. During the last divide step (the correction step), LAST STEP L tests MQ10 for a 1.

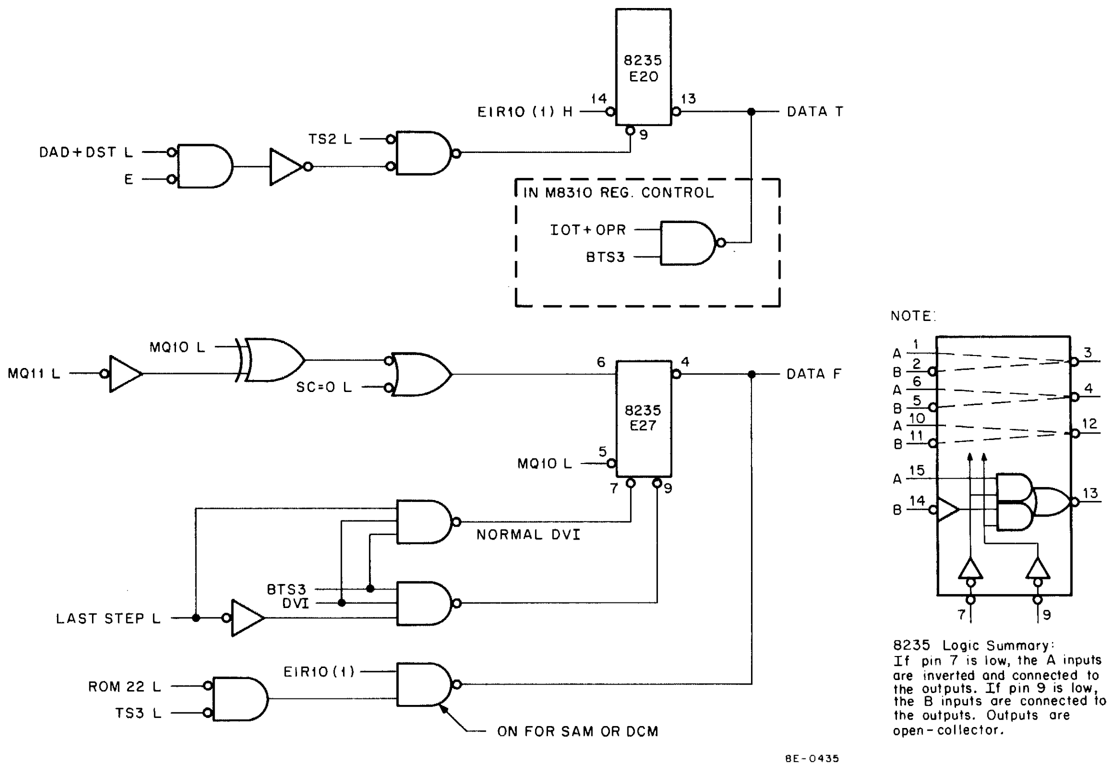


Figure 1-28 EAE Data Control Logic for Processor Data Control Gates

1.22 EAE ROUTE CONTROL SIGNALS

The EAE route control signals control shifting right, shifting left, carry in, and carry out. These elements are represented in the simplified block diagram given in Figure 1-29.

1.22.1 Step Counter Loading and Control Logic

The Step Counter loading and control logic is illustrated in Figure 1-30. The logic controls loading, reading, and incrementing the Step Counter.

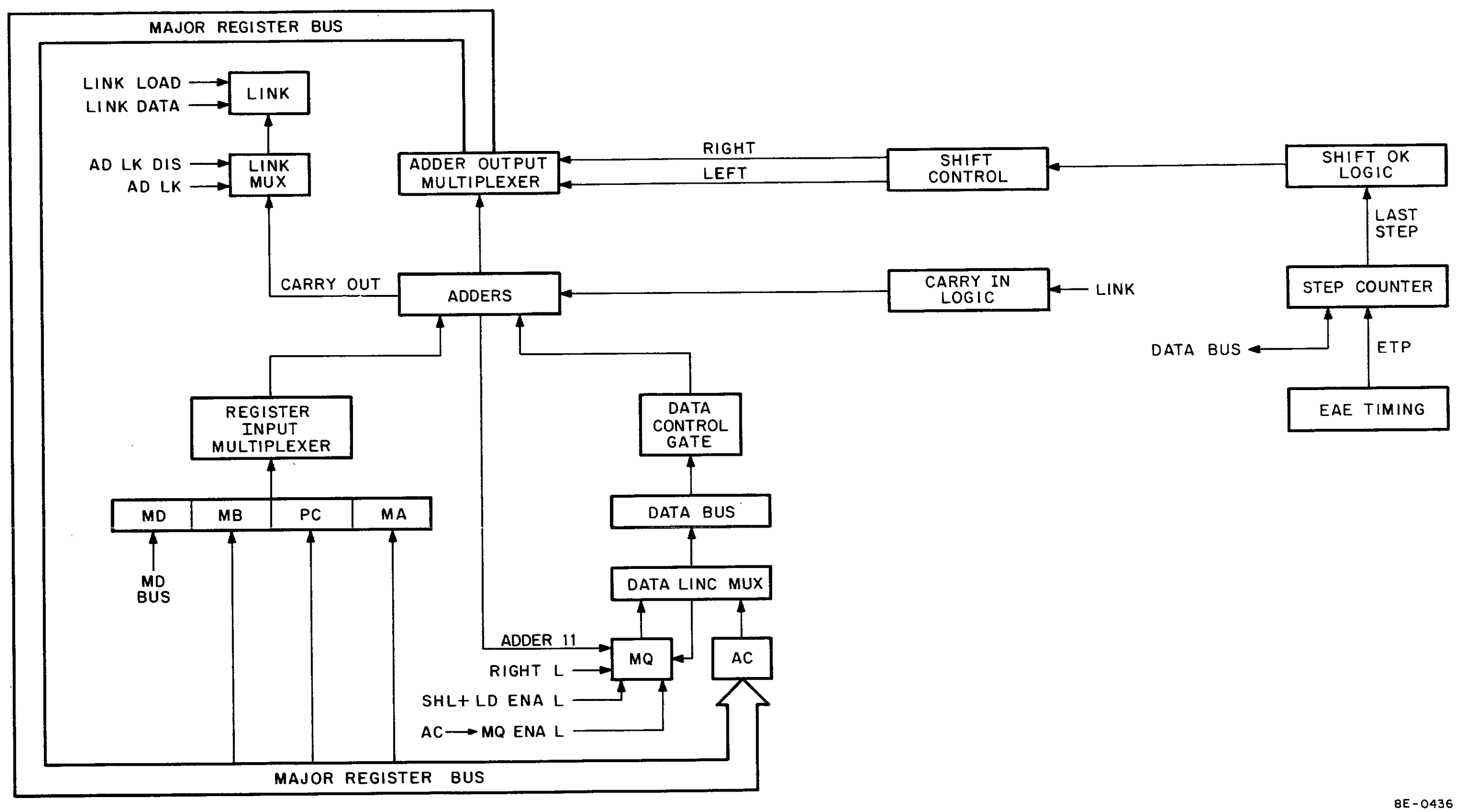


Figure 1-29 EAE Route Control Signal Block Diagram

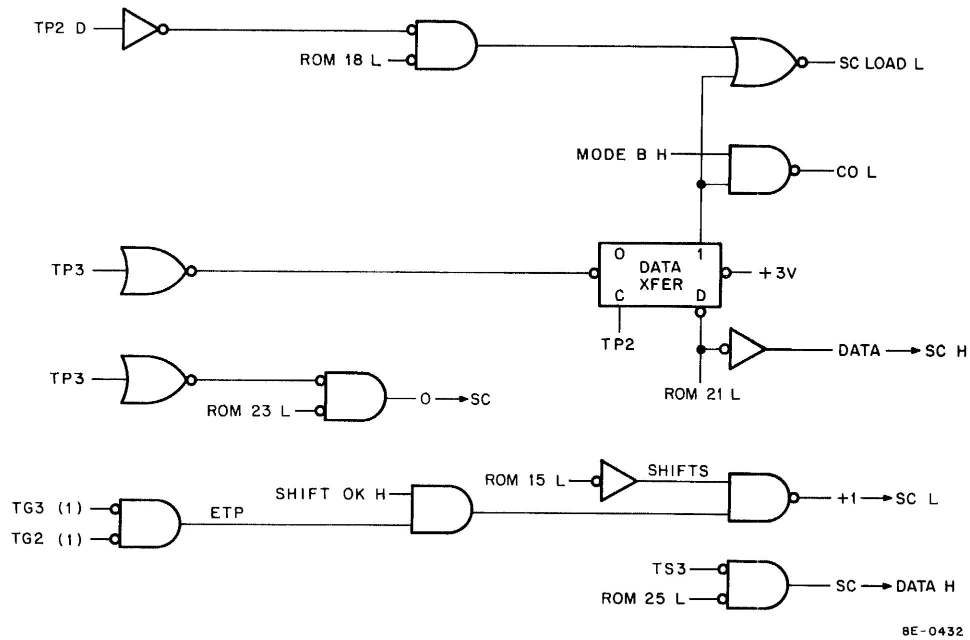


Figure 1-30 Step Counter Loading and Control Logic

Two sources of data (the AC and MD) can be loaded into the SC by two different instructions. If the last five bits of the AC are to be loaded into the Step Counter, the ACS instruction generates ROM 21 L, which sets the DATA XFER flip-flop at TP2. Because ACS is a Mode B instruction, MODE B H and DATA XFER (1) H generate CO L and DATA XFER (1) asserts SC LOAD L. Signal CO L is used to clear the AC Register at the same time the SC is loaded. Signal DATA → SC H is used to gate the contents of the DATA BUS to the Step Counter. At TP3, DATA XFER is cleared and the SC is loaded.

If an SCL, SHL, ASR, or LSR instruction is decoded and the major state is EXECUTE, ROM 18 L is asserted. At TP2D, SC LOAD L is asserted, which causes the complement of the last five MD bits to be loaded into the Step Counter. Signal +1 → SC L is generated at EAE Timing Pulse (ETP) time by SHIFT OK H and ROM 15 L. ROM 15 L is decoded when a shift operation is to take place.

When it is desired to load the AC Register with the contents of the Step Counter, instruction SCA asserts ROM 25 L. During TS3 L, SC → DATA H is asserted, gating the contents of the Step Counter to the DATA BUS.

1.22.2 Step Counter Logic

The Step Counter logic is illustrated in Figure 1-31. IC 8266 transmits the complement of DATA 7 through 11 and the uncomplemented MD bits. When signal DATA → SC goes high, the DATA BUS bits (low for 1) are applied to the Step Counter (high for 1). Otherwise, bits MD7–11 (low for 1) will be complemented and applied to the Step Counter (high for 1). The Step Counter loads the contents of the four input lines when SC LOAD L is received and increments when it receives the signal +1 → SC L. The Step Counter is IC 74193. It is usually used to count up to zero. When the count reaches 0, signal SC = 0 L is asserted.

Signal LAST STEP L is generated when SC = 13_8 during an MUL, when SC = 14_8 during a DVI, and when SC = 37 for all operations.

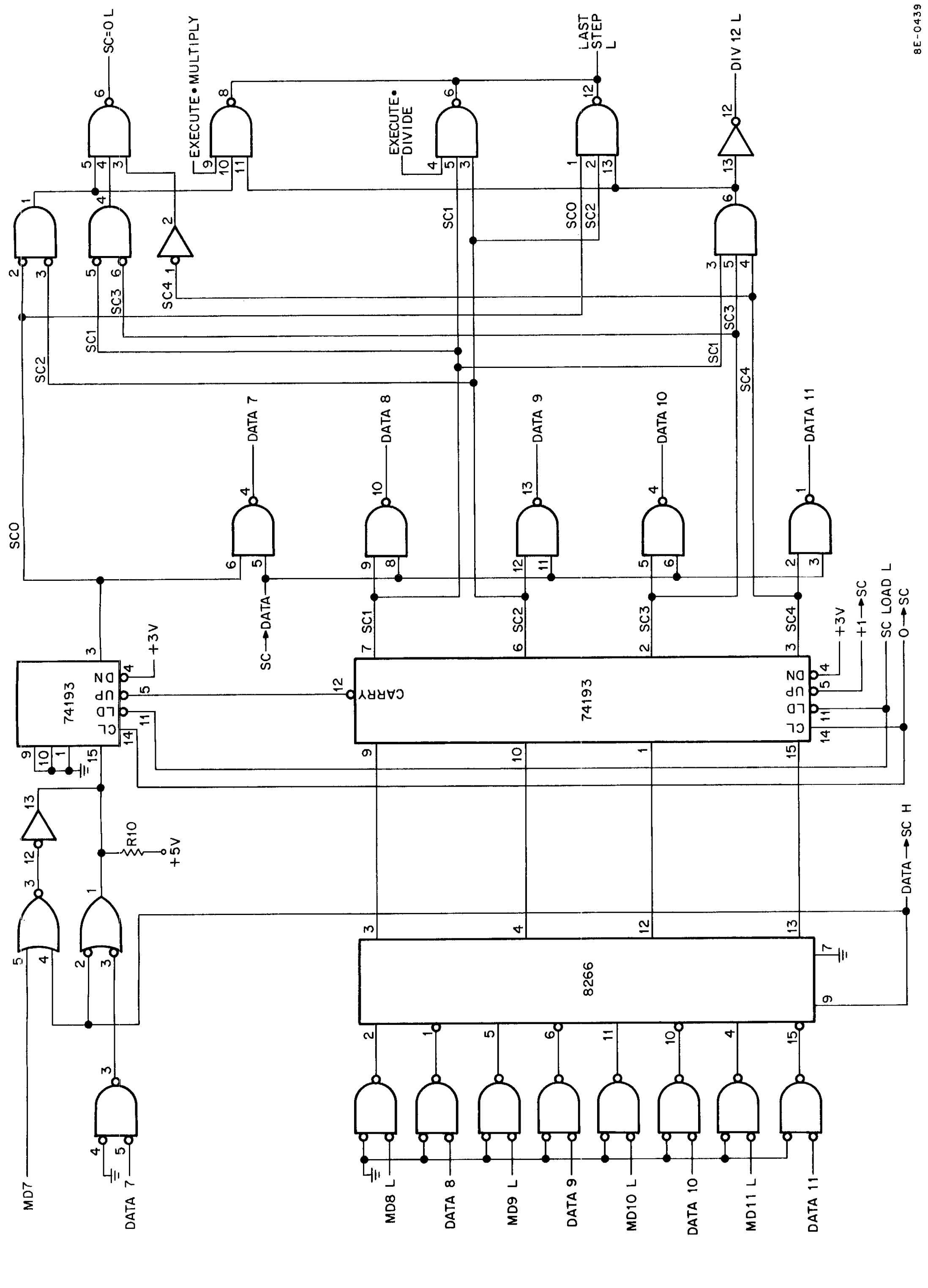


Figure 1-31 EAE Step Counter

1.22.3 Shift Right/Shift Left Control Logic

The shift right/shift left control logic is illustrated in Figure 1-32. The 8235 ICs are multiplexers that receive select signals at pins 7 or 9 to select signals being received at pins 1, 2, 10, or 14. LEFT L is enabled during an SHL, NORMALIZE, or DVI instruction. The shift-right (ASR or LSR) and multiply instructions cause RIGHT L to be asserted.

1.22.4 Shift OK Logic

The shift OK logic (Figure 1-33) monitors the contents of the AC and MQ during the NMI instruction and checks for LAST STEP L. When LAST STEP L becomes low during an SHL, LSR, or ASR instruction, SC = 37. If the MODE flip-flop is set, indicating the new or Mode B instruction set is in use, SHIFT OK H is grounded to prevent the last shift from occurring. During an NMI instruction, SHIFT OK H is grounded when the number becomes normalized, to prevent an extra shift from taking place as the processor is restarted.

1.22.5 EAE Carry In Logic

Signal CARRY IN L is developed by the EAE under the conditions shown in Figure 1-34. ROM output, ROM 22 L is decoded when an SAM, DCM, or DPIC instruction is to be performed. If the EAE is off (SAM or Step 1 of DPIC and DCM) CARRY IN L is generated. ROM 13 L controls the coupling of carries, and introduces a CARRY IN L if the Link is set during Step 2 of DPIC and DCM and the two EXECUTE cycles of the DAD instruction.

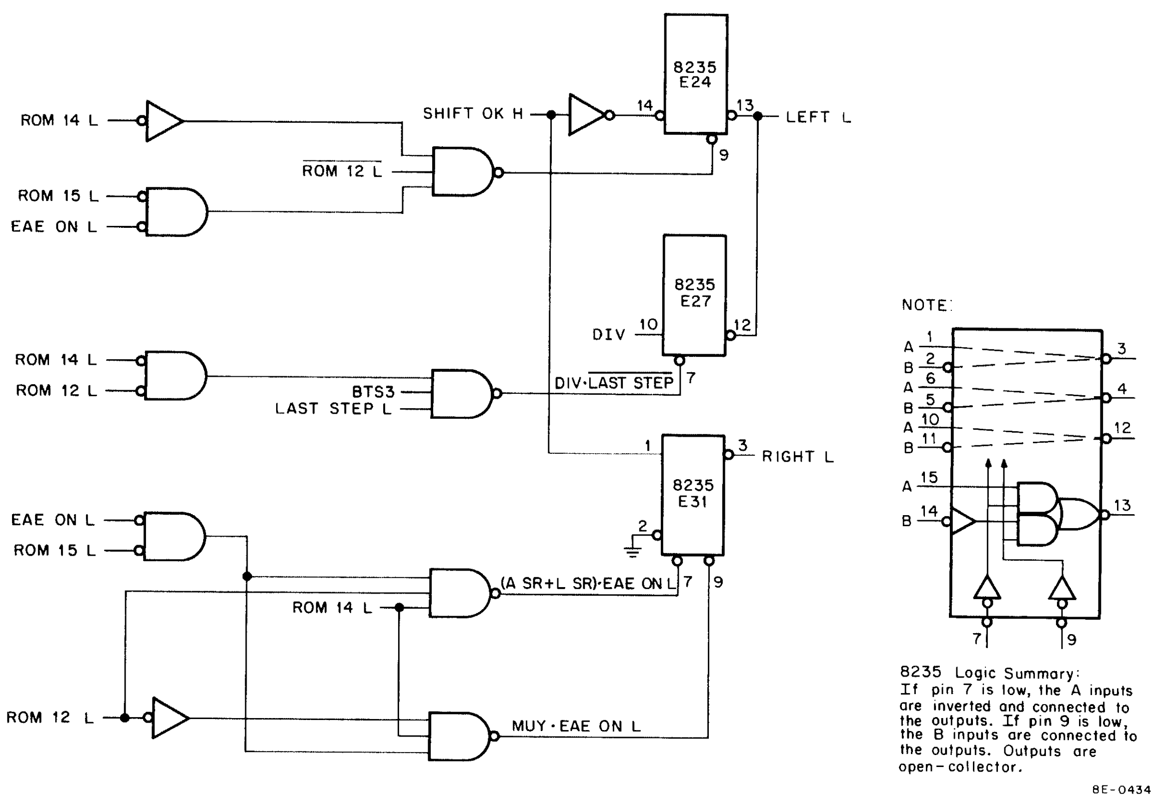


Figure 1-32 EAE Shift Right/Shift Left Control Logic

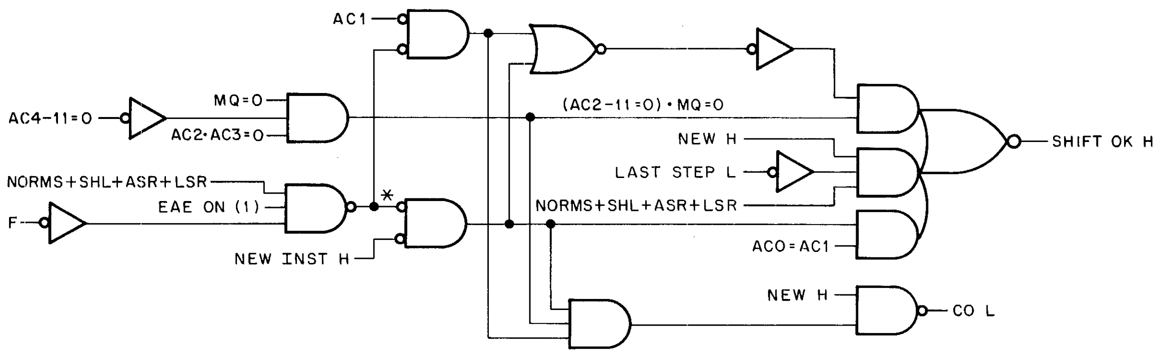


Figure 1-33 Shift OK Logic

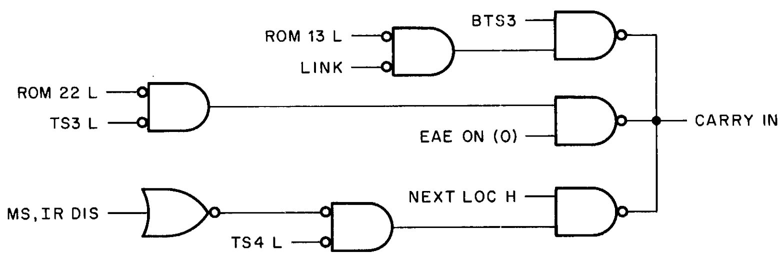


Figure 1-34 EAE Carry In Logic

1.22.6 MQ Register Shift Left Logic

The MQ Register shift left logic is illustrated in Figure 1-35. Decoded outputs ROM 14 L and ROM 15 L, with EAE, select such signals as SHIFT OK H and CARRY OUT L, etc. Signal MQ DATA L provides quotient information to MQ11 during a divide. Otherwise, MQ DATA L remains high, shifting zero into MQ11 for NMI and SHL instructions. Signal SHL + LD EN L forces the MQ Register to shift one place to the left. Also shown with this logic is the DIV LINK and other gating required to generate the quotient bit.

1.22.7 AC to MQ Transfer Signals

Figure 1-36 illustrates the conditions when the signal AC → MQ ENA L can be asserted. This gating is used during the DAD and DST instruction as a part of the AC → MQ swapping process. AC → MQ ENA L is also generated for the DPIC and DCM instructions in the M8310, as described in Volume 1, Paragraph 3.40.

1.23 DESTINATION CONTROL SIGNALS

The signals that actually cause register loading are called destination control signals. In the case of the EAE, only the AC LOAD L and MQ LOAD L signals are developed in the EAE logic. Other loading signals, including MB LOAD L, are asserted by the processor. Figure 1-37 illustrates how the AC, MQ, and MB Register loading signals are generated.

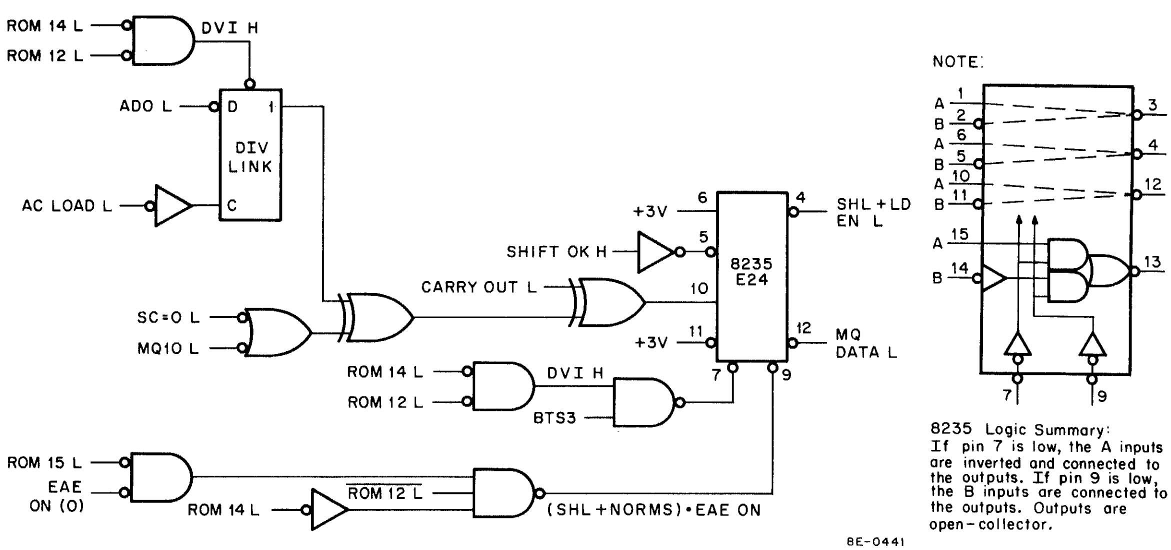


Figure 1-35 MQ Register Shift Left Logic

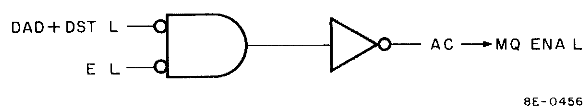


Figure 1-36 AC → MQ TRANSFER Signal

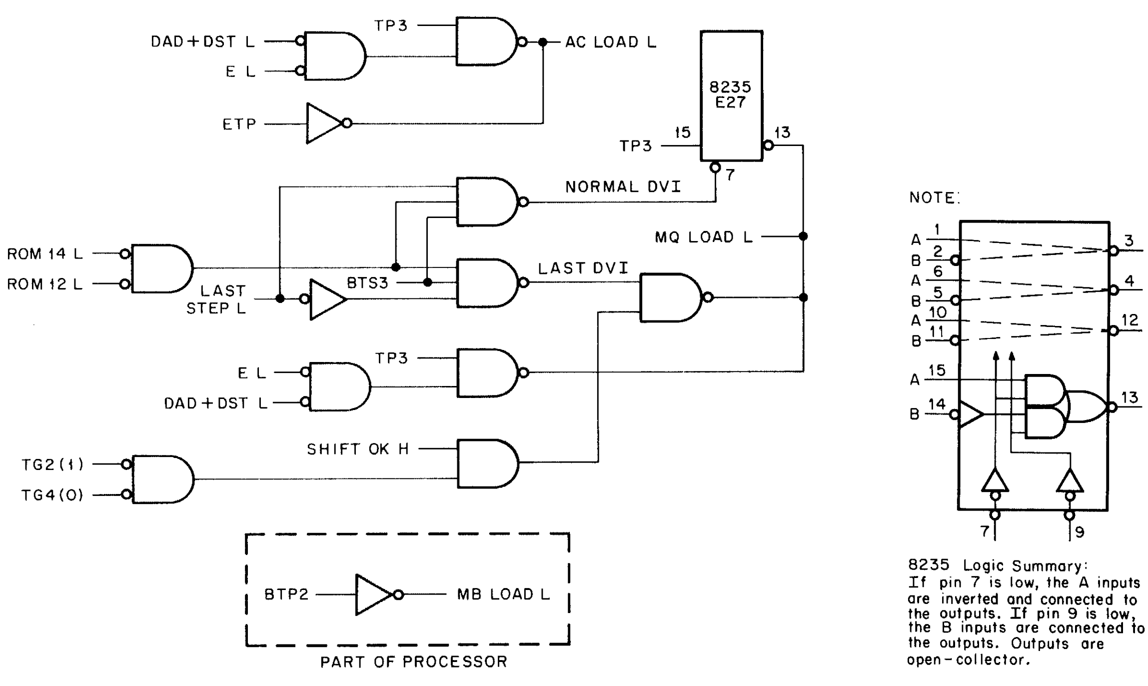


Figure 1-37 Destination Control Logic

1.24 EAE START/STOP LOGIC

The EAE start/stop logic, shown in Figure 1-38, transfers timing generation from the CPU to the EAE. Up to TP3 of certain EAE cycles the generation of all timing signals is under CPU control. At TP3, timing control can be transferred to the EAE to allow high-speed multiple shifts and/or adds. At the conclusion of these special operations, timing is returned to the CPU.

The EAE grounds OMNIBUS signal NOT LAST XFER L before TP3 when the current instruction and major state requires running the EAE timing chain. The NLX flip-flop is clocked 100 ns after the trailing edge of TP2, after the ROMs and associated decoding have had ample time to settle. If the D input to NLX is high, one of the following instructions has been decoded; the major state is the one in which the EAE operation is to take place.

Instructions which start the EAE Timing Chain:

ASR, LSR, SHL, NMI, MUY, DVI, DPIC, DCM

The output of the NLX flip-flop is applied to one input of a two-input NAND gate (labeled A in Figure 1-38) whose output grounds NOT LAST XFER L. At the same time, the other input of gate A is high, unless a Divide Overflow situation is detected by gate B. If NOT LAST XFER L is low at the leading edge of TP3, CPU timing is interrupted as described in Volume 1, Paragraph 3.21. EAE TG START H is ANDed with TP3 and the result used to set the E SYNC flip-flop. At the trailing edge of TP3, the EAE ON flip-flop is clocked and sets. The EAE's timing chain is now running.

The EAE continues to run until some condition within the EAE causes EAE STOP H to go high. At the leading edge of the next ETP, E SYNC clears. At the trailing edge of the same ETP, EAE ON clears and stops the EAE. Signal RESTART L has the same effect on the Timing Generator of the CPU (but not the Major Registers) as does BUS STROBE L – it starts the CPU if NOT LAST XFER L is high. RESTART L is generated twice, once when the EAE starts (it has no effect then), and once when the EAE stops. NOT LAST XFER L is high by the time the second RESTART L signal is generated, because NLX is cleared by one of the EAE's timing generator flip-flops (TG2) a short time after the trailing edge of TP3 and well before the leading edge of the first ETP.

1.25 EXTENDED EAE LOGIC

The extended EAE logic (Figure 1-39) consists of a D-type flip-flop called EX1 and its associated logic. When set, EX1 forces a second EXECUTE cycle and causes the processor to access the next sequential memory location. Signal NEXT LOC H is used to generate CARRY IN L and to ground EX0, which causes MA + 1 to the MA Register.

The logic gating for the EX1 data input is limited to either a DAD or DST instruction. DAD, DST, MUY, and DVI are the only EAE instructions that enter a DEFER cycle. For both MUY and DVI instructions, EIR6(1) is low and, therefore, prevents the EX1 flip-flop from being set.

1.26 EAE LINK CONTROL LOGIC

The EAE link control logic (Figure 1-40) contains all of the Link Control elements required to load the Link and to disable the Link so that it is not affected by certain processor-EAE operations. For a better understanding of Link operation within the processor, refer to Volume 1, Paragraph 3.39.

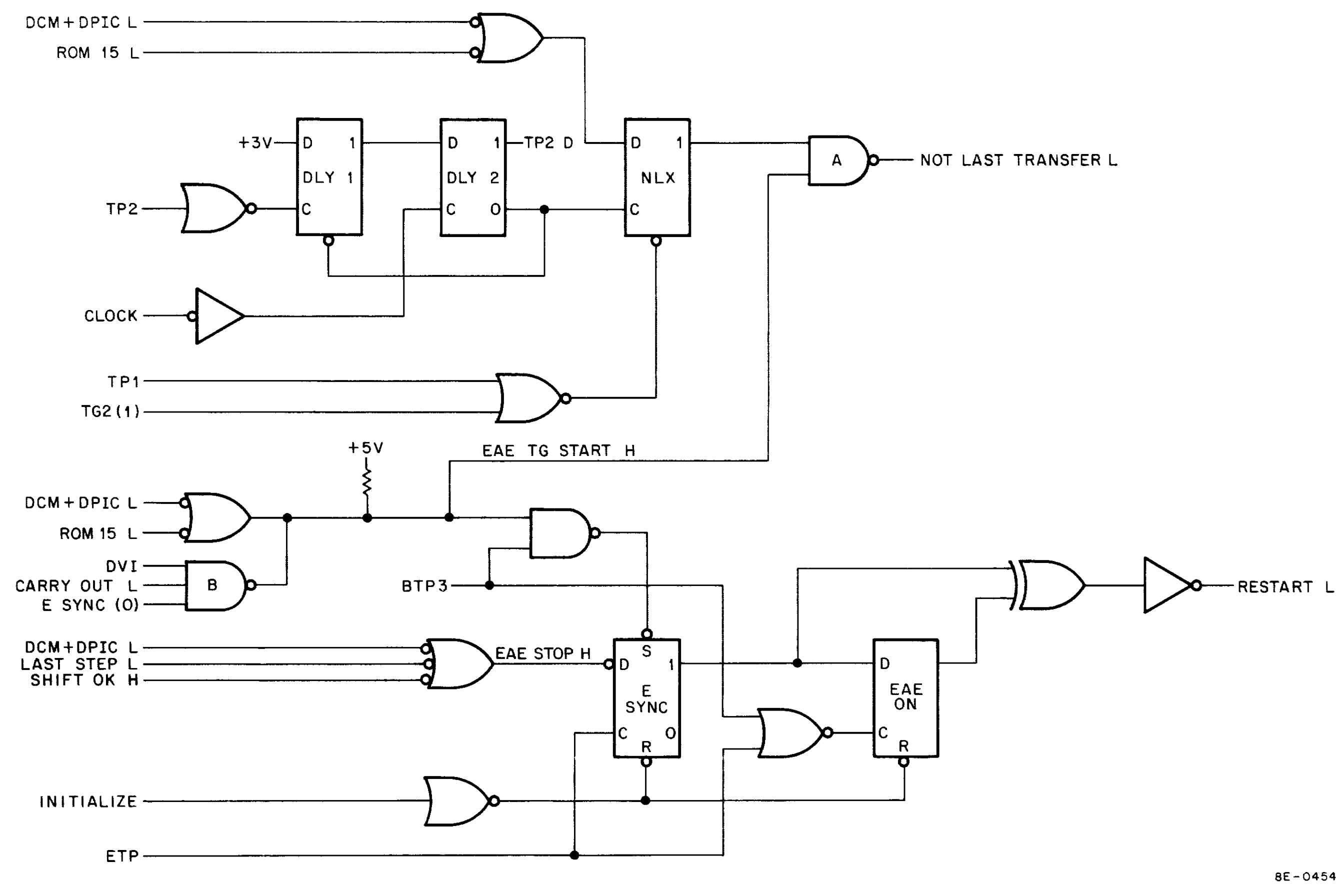


Figure 1-38 EAE Start/Stop Logic

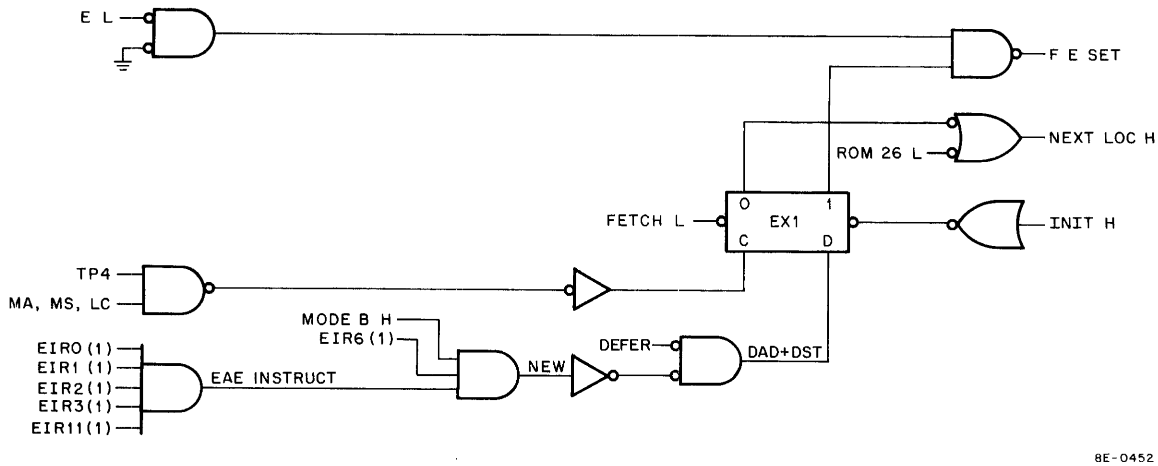


Figure 1-39 Extended EAE Logic

Signal LINK DATA L provides information to be loaded into the Link by the LINK LOAD L pulse. This information may be one of the following:

Enable	Result if Link Loaded	Used By
None	Zero	DVI (to clear overflow indication) MUY, LSR, DAD
ROM 11 L	Carry from adders	DAD, DPIC, DCM
ROM 13 L	AC0	ASR, NMI
DVI EAE ON (0)	One	DVI (to anticipate overflow)

The LINK LOAD L signal is generated at TP3 if ROM 17 is low (usually to preset the Link) and at ETP time during left shifts. During the execution of DVI, the Link is set at TP3 (for possible overflow indication) and cleared if the DVI process reaches LAST STEP L (meaning a legal divide has occurred). During the right shift and MUY instructions, the Link is not modified.

During left shifts, data is introduced from MQ0 to AC11 via a line called ADLK L. During SHL or NMI instructions, MQ0 is gated directly to ADLK L. For DVI, MQ0 is sometimes inverted before being applied to ADLK L. The gating logic, which depends on MQ11 and whether the first divide step has taken place (EAE ON), is also shown in Figure 1-40.

The ADLK DIS L signal disables the normal Link gating described in Volume 1, Paragraph 3.39. ADLK DIS L must be low any time the LINK LOAD L/LINK DATA L inputs are used or whenever left shifts are performed in order to avoid conflict with the normal link gating. This line is grounded by ROM 16 L.

1.27 EAE SKIP LOGIC AND GT FLAG

The EAE skip logic is illustrated in Figure 1-41. There are two methods of generating a SKIP L signal. Signal NEXT LOC H, which is generated by the extended EAE logic, is inverted and applied to the SKIP L line. Signal NEXT LOC H also generates CARRY IN L.

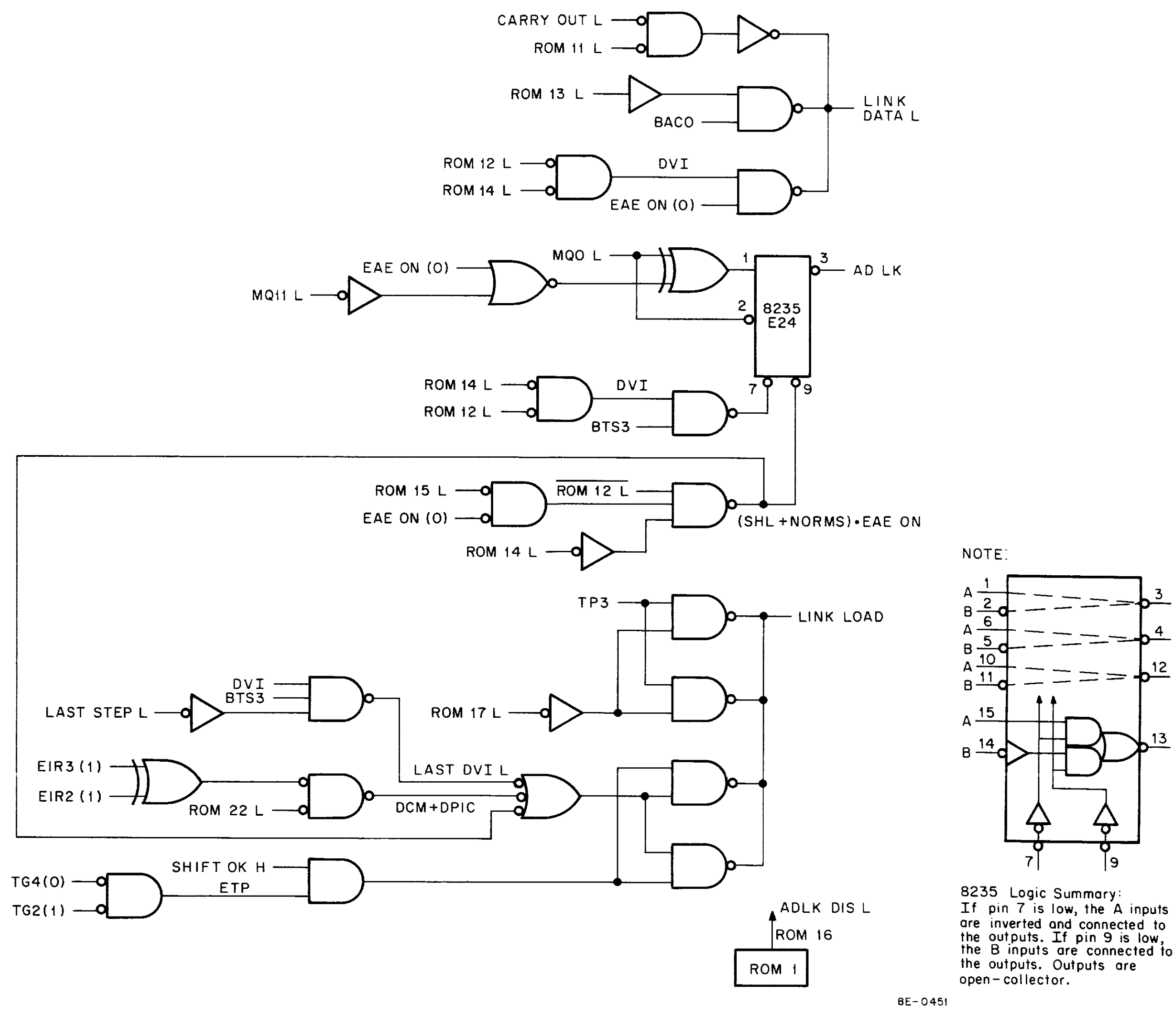


Figure 1-40 EAE Link Control

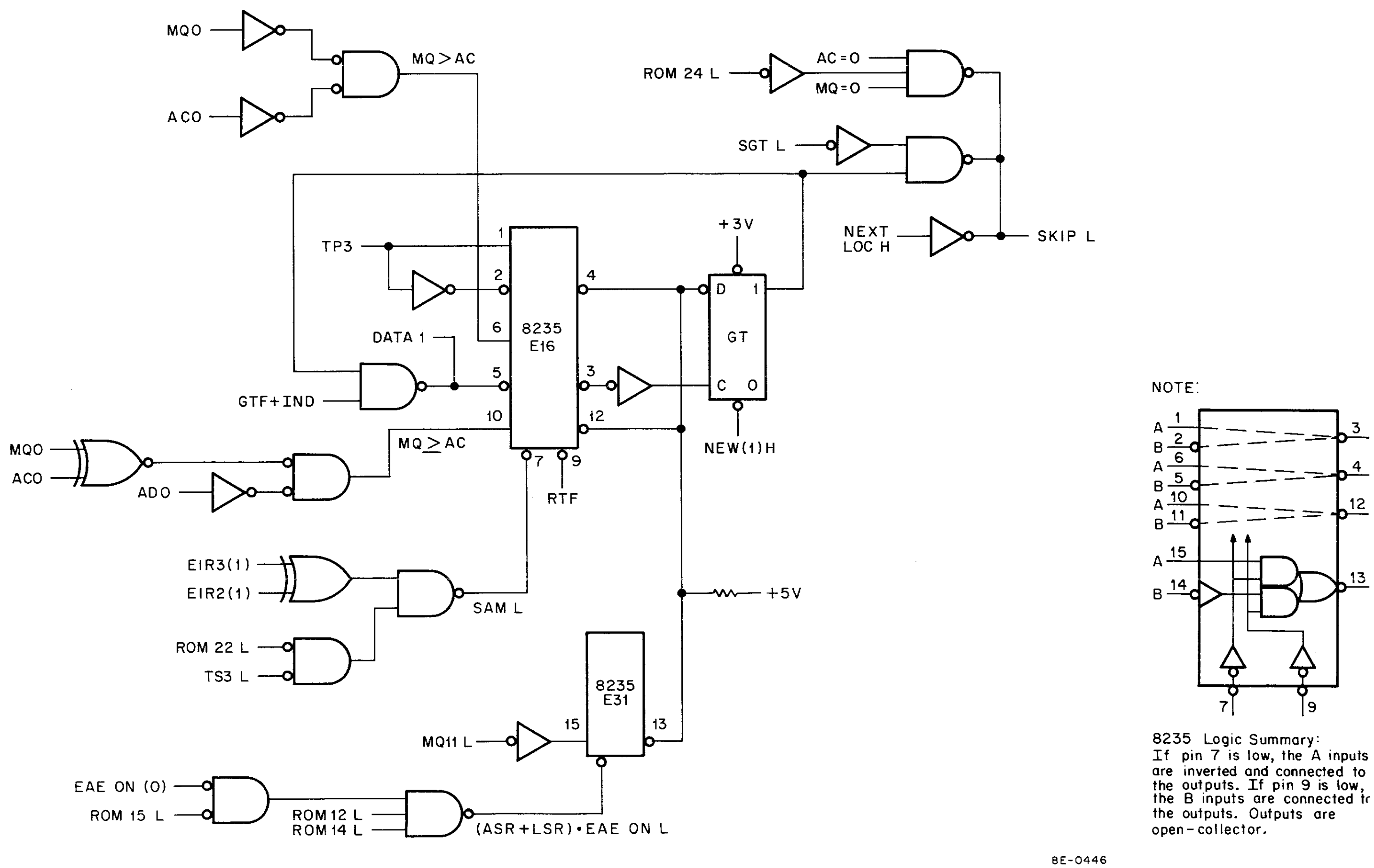


Figure 1-41 EAE Skip Logic

The SKIP line is also grounded when a DPSZ instruction is being performed. This instruction tests the AC and the MQ for 0. If both registers equal zero, the SKIP L signal will be asserted. SKIP L is used to set the SKIP flip-flop in the processor's logic at TP3. For information on the processor's skip logic, refer to Volume 1, Paragraph 3.38.

The more complex part of the skip logic involves the Greater Than (GT) flag. When the GT flip-flop is set, instruction SGT forces the SKIP line to go low. The GT flag can be changed by one of three methods:

Method	Source of Information
RTF instruction	DATA 1
SAM instruction	Set if MQ greater than or equal to AC; cleared otherwise.
ASR or LSR	Previous MQ11

The GT flag is cleared if the EAE is in Mode A; hence, the flag is active only for Mode B instructions.

SECTION 5 MAINTENANCE

Since the EAE is physically connected to the Central Processor Timing Generator and OMNIBUS, a definite possibility exists that some EAE malfunctions will not be caused by the EAE modules. For this reason, the following procedure is suggested.

Step	Procedure
1	Remove M8340 and M8341 from the OMNIBUS.
2	Perform processor-related diagnostics to ensure processor reliability.
3	Insert M8340 in OMNIBUS and connect it to the Timing Generator via the "J" top connector.
4	Perform PDP-8/E Instruction Tests 1 and 2.
5	Insert M8341 in OMNIBUS and connect it to M8340 connector H.
6	Perform PDP-8/E Instruction Tests 1 and 2.
7	Connect M8341 to M8310 via "F" connector.
8	Perform PDP-8/E Instruction Tests 1 and 2.

NOTE

If problems are encountered during this procedure, they can be isolated by troubleshooting the processor, and tracing the malfunction back to the last module or connector that was added to the system.

When this procedure fails to isolate a malfunction, perform EAE Instruction Tests 1 and 2, and the EAE Extended Memory Test. One of these tests should give some idea of the problem.

Once the problem is pinpointed, write and toggle in a simple program using the malfunctioning instruction.

The following basic ideas can be built upon or modified to suit any special purpose:

1. Mode Changing Instructions

```
0/  7431     SWAB
    7447     SWBA
    5000     JMP 0
```

2. SCL or ACS (dependent on mode)

Mode A

```
0/  7604     LAS
    3003     DCA.+2
    7403     SCL
    XXXX     OPERAND
    5000     JMP 0
```

Mode B

```
0/  7431     SWAB
    7604     LAS
    7403     ACS
    5001     JMP .–2
```

3 SCA or SCA, CLA

Mode A

```
0/  7604     LAS
    3003     DCA .+2
    7403     SCL
    XXXX     OPERAND
    7441 or  SCA or SCA, CLA
    7641
    7000  \
     ↓     } 5 or 6 NO OPS WILL HOLD THE AC
    7000  /  FOR OBSERVATION
    5000     JMP 0
```

Mode B

```
0/  7431     SWAB
    7604     LAS
    7403     ACS
    7441 or  SCA or SCA, CLA
    7641
    7000  \
     ↓     } NO OPS TO HOLD AC FOR OBSERVATION
    7000  /
    5001     JMP 1
```

4. SHL Shift Left

Mode A

```
0/  7604     LAS (Shift Count = one more than the last five bits of the location following SHL)
    3006     DCA
    1050     TAD MQ
    7421     MQL
```

(continued on next page)

```
      1051    TAD AC
      7413    SHL
      XXXX    SHIFT COUNT
      7000 ⎫
       ↓   ⎬ NO OPS WILL HOLD AC AND MQ FOR OBSERVATION
      7000 ⎭
      7621    CAM
      5000    JMP 0
```

Mode B

```
  0/  7604    LAS (Shift Count = last five bits of location following SHL)
      3007    DCA
      7431    SWAB
      1050    TAD MQ
      7421    MQL
      1051    TAD AC
      7413    SHL
      XXXX    SHIFT COUNT
      7000 ⎫
       ↓   ⎬ NO OPS WILL HOLD AC AND MQ FOR OBSERVATION
      7000 ⎭
      7621    CAM
      5001    JMP 1
```

5. NMI

```
  0/  7431    SWAB (Start here if Mode B)
  1/  1050    TAD MQ (Start here if Mode A)
      7421    MQL
      1051    TAD AC
      7411    NMI
      7000 ⎫
       ↓   ⎬ HOLDS AC AND MQ FOR OBSERVATION
      7000 ⎭
      5001
```

6. ASR or LSR

```
  0/  7604    LAS (Shift Count = One more than this number in location following ASR or LSR
              if Mode A)
      3007    DCA (Shift Count = Number in location following ASR or LSR if Mode B)
      7431    SWAB (Start here if Mode B)
      1050    TAD MQ (Start here if Mode A)
      7421    MQL
      1051    TAD AC
      7415 or ASR or LSR
      7417
      XXXX    SHIFT COUNT
```

(continued on next page)

```
7000 )
↓     } NO OPS TO HOLD AC AND MQ FOR OBSERVATION
7000  )
5000 )
```

7. MUY

Mode A

```
0/  7406    LAS (Multiplier)
    3006    DCA
    1050    TAD MQ
    7421    MQL
    1051    TAD AC
    7405    MULTIPLY
    XXXX    MULTIPLIER
    7000 )
     ↓   }  HOLD AC AND MQ FOR OBSERVATION
    7000 )
    5000    JMP
```

Mode B

```
0/  7431    SWAB
    7604    LAS (Multiplier)
    3100    DCA 100
    1050    TAD MQ
    7421    MQL
    1051    TAD AC
    7405    MULTIPLY
    100     ADDRESS OF MULTIPLIER
    7000 )
     ↓   }  HOLD AC AND MQ FOR OBSERVATION
    7000 )
    5000    JMP
```

8. DVI

Mode A

```
0/  7604    LAS (Divisor)
    3006    DCA
    1050    TAD MQ
    7421    MQL
    1051    TAD AC
    7407    DVI
    XXXX    DIVISOR
    7000 )
     ↓   }  HOLD AC AND MQ FOR OBSERVATION
    7000 )
    5000    JMP
```

(continued on next page)

Mode B

```
0/  7431
    7604      LAS (Divisor)
    3100      DCA 100
    1050      TAD MQ
    7421      MQL
    1051      TAD AC
    7407      DIVIDE
    100       ADDRESS OF DIVISOR
    7000  ⎫
     ↓    ⎬   HOLD AC AND MQ FOR OBSERVATION
    7000  ⎭
    5001      JMP
```

9. SAM

```
0/  7431      SWAB
    1050      TAD MQ
    7421      MQL
    1051      TAD AC
    7457      SAM
    7000  ⎫
     ↓    ⎬   HOLD AC, MQ AND STATUS FOR OBSERVATION
    7000  ⎭
    5001      JMP
```

10. DAD or DLD

```
0/  7431      SWAB
    1050      TAD MQ
    7421      MQL
    1051      TAD AC
    7443 or   DAD or DLD
    7763
    XXXX      ADDRESS OF WORD
    7000  ⎫
     ↓    ⎬   HOLD AC AND MQ FOR OBSERVATION
    7000  ⎭
    5001
```

11. DST   CONFIGURE AC AND MQ

```
0/  7431      SWAB
    7445      DST
    100       LOCATION OF WORD
    7763      DLD
    100
    7000  ⎫
     ↓    ⎬   HOLD AC AND MQ FOR OBSERVATION
    7000  ⎭
    5001
```

(continued on next page)

12. DPIC

```
0/  7431   SWAB
    7573   DPIC
    5001   JMP
```

13. DCM

```
0/  7431   SWAB
    1050   TAD MQ
    7421   MQL
    1051   TAD AC
    7575   DCM
    1051   TAD Original AC
    7440   SZA
    7402   HLT
    7521   SWP
    1050   TAD Original MQ
    7440   SZA
    7402   HLT
    7621   CAM
    5001   JMP
```

14. DPSZ

```
0/  7431   SWAB
    7621   CAM
    7451   DPSZ
    7402   HLT
    5001   JMP
```

The programs listed above, when used in conjunction with the flowchart, ROM encoding matrix, and print set, provide simple, repetitive troubleshooting instructions.

Use the following check list as a guideline.

a. Instruction was properly loaded into the Op-decoder.

b. ROM address is correct.

c. ROM outputs are correct.

d. Step Counter decodes last step properly.

e. Control and loading signals listed on the flow chart are occurring at the correct time in relation to time states and bit configurations.

SECTION 6 SPARE PARTS

Table 1-4 lists recommended spare parts for the KE8-E. These parts can be obtained from a local DEC office or from DEC, Maynard, Massachusetts.

Table 1-4
Recommended KE8-E Spare Parts

DEC Part No.	Description	Quantity
19-05585	IC DEC 7476	1
19-05576	IC DEC 7410	1
19-09955	IC DEC 7412	1
19-10018	IC DEC 74193	1
19-09934	IC DEC 8266	1
19-09267	IC DEC 74H11	1
19-05635	IC DEC 74H20	1
19-05586	IC DEC 74H40	1
19-09486	IC DEC 384	1
19-09004	IC DEC 7402	1
19-09667	IC DEC 74H74	1
19-09059	IC DEC 74H30	1
19-09973	IC DEC 97401	1
19-09485	IC DEC 380	1
23-001A1	IC Encoded ROM (Drives ROM 11–18)	1
23-002A1	IC Encoded ROM (Drives ROM 21–28)	1
19-09930	IC DEC 7405	1
19-09705	IC DEC 8881	1
19-05515	IC DEC 7400	1
19-07686	IC DEC 7404	1
19-09062	IC DEC 74H53	1
19-10011	IC DEC 7486	1
19-09935	IC DEC 8235	1
13-00295	Resistor 330Ω 1/4W, 5%	1
13-00365	Resistor 1K, 1/4W, 5%	1
13-00317	Resistor 470Ω, 1/4W, 10%	1
10-00067	Capacitor 6.8 μF, 5V, 20% Solid Tantalum	1
10-01610	Capacitor 0.01 μF, 100V, 20% Ceramic Disk	1

PART 2
MEMORY EQUIPMENT OPTIONS

CHAPTER 2
KM8-E MEMORY EXTENSION AND TIME-SHARE

SECTION 1 INTRODUCTION

The KM8-E Memory Extension and Time-Share option generates a 3-bit address for the extended memory address lines. This address allows the use of more than 4K of memory and, when required, is used as a prerequisite in a time-sharing system.

All logic is contained on one M837 quad-size module that plugs directly into the OMNIBUS.

All signals enter and leave the module via the OMNIBUS.

2.1 MEMORY EXTENSION DESCRIPTION

Memory extension hardware is required when more than 4K of memory is to be addressed. Except for data break devices, the KM8-E Memory Extension is the only means by which extended memory addresses can be applied to the three extended memory address lines called EMA 0-2. Memory is divided into 4K fields, starting with field 0, for the basic 4K memory, up to field 7, when 32K of memory is employed. Each 4K of memory receives and decodes the EMA signals. This provides the addressing capability of up to 32,768 memory locations.

There are two types of fields: the Instruction Field, which acts as an extension to the PC and direct argument addresses; and the Data Field, which augments the address of indirectly obtained arguments. When the programmer desires to use one field for instructions and a different field for data, he directs the corresponding field address to either the Instruction Field Register or the Data Field Register contained on the KM8-E. The field addresses are applied to the EMA lines by specific instructions and conditional logic. Safeguards are provided so that during unplanned events, such as interrupts and data breaks, no field addresses are lost. Program instructions allow any field address to be stored; this is particularly important to the programmer desiring to nest interrupts.

A simplified block diagram showing the basic transfer paths dealing with memory addressing and field addressing is shown in Figure 2-1. The KM8-E is the only route by which fields above field 0 may be selected. The only exception is the data break device that has the capability of selecting its own memory field. The programmer has two methods of selecting memory fields. One method is via the Console Switch Register; the second method is via an IOT instruction. In either event, field information is loaded into the appropriate register in the extension control. The extension control automatically responds to the appropriate instructions and major states by placing the contents of the correct field register onto the EMA lines.

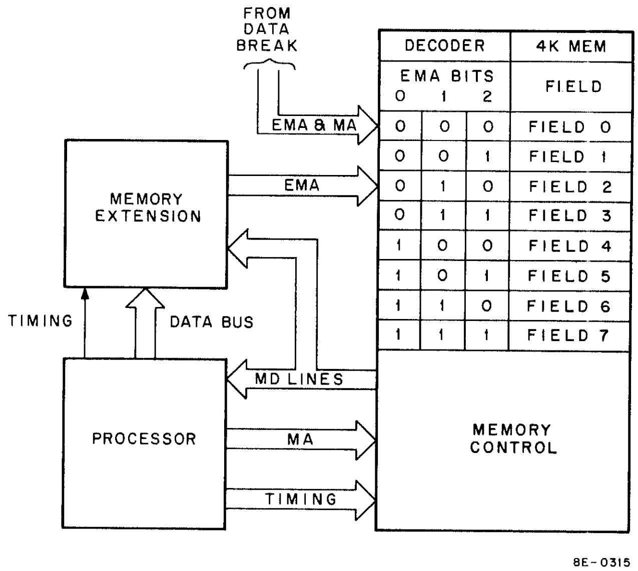


Figure 2-1 Memory Extension, Simplified Block Diagram

2.2 TIME-SHARE DESCRIPTION

The time-share portion of the KM8-E is used only when a time-sharing system is to be employed. The KM8-E Module contains a jumper in the inhibit logic that prevents the operation of the time-share function. Unless this jumper is to be removed from the module, the reader need not be concerned with the time-share description in this chapter.

The KM8-E Memory Extension and Time-Share option provides the necessary additional hardware for a general-purpose time-share system. This option, coupled with a time-sharing program, such as TSE, and 12K to 32K of core, allows a maximum of 16 users to independently run programs. This creates the appearance that each user has the computer to himself.

As a system, the user can be considered operating in one of three levels: a) not logged-in level, b) monitor level, and c) user level. A typical system is illustrated in Figure 2-2. The user interface to the system is the KL8-E Teleprinter Control. The monitor program performs a dominant role in controlling operation between the processor and the KM8-E option and between the users and the processor.

The monitor is a complex of subprograms to coordinate the operations of various programs and user consoles. The monitor allocates the computer's time and services to various users; it grants a slice of processing (computing) time to each job, and schedules jobs in sequential order to make the most use of the mass-storage device. The monitor also handles user requests for hardware operations (reader, punch, etc.), swaps (moves) programs between memory and mass storage, and manages the user's private files. Thus, the primary time-sharing capability is provided by the system monitor and the PDP-8/E Processor. However, certain additional hardware not provided by the PDP-8/E Processor is needed to accommodate the special requirements of time-sharing and extended addressing capability.

For more information on the monitor and user programming, refer to Chapter 10 of *Introduction to Programming.*

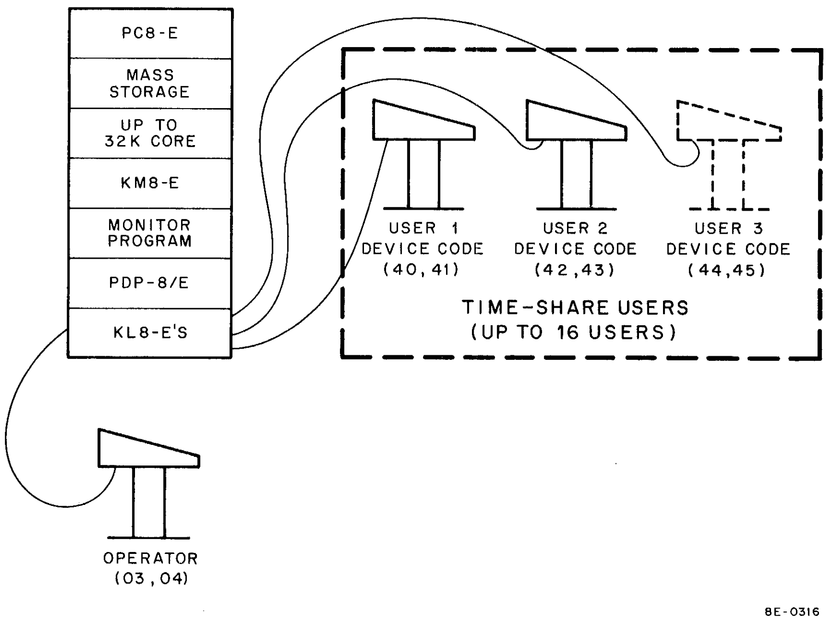


Figure 2-2 Typical Time-Share System

2.3 EXTENDED MEMORY AND TIME-SHARE SUMMARY

As a memory extension control, the KM8-E provides:

a. Hardware to allow the programmable selection of the extended memory field (fields 1 through 7), allowing the extended addressing capability of the processor from a basic system of 4096 addresses to an extended memory system of up to 32,768 addresses.

b. Hardware to prevent an interrupt or data break from interferring with the extended addressing scheme.

c. Hardware to save and restore a field return address.

As a time-share control, the KM8-E provides:

a. Hardware to distinguish between user and monitor modes.

b. Hardware to trap certain instructions, causing an interrupt and placing the time-share system in monitor mode.

c. Ability to establish user mode.

2.4 SOFTWARE

The following programs are used in time-sharing and memory extension operations:

a. System Programs

1. TSE Time-Sharing Monitor (DEC-T8-MRFB) – TSE (Time-Sharing System for the PDP-8/E Computer) is a general-purpose, stand-alone, time-sharing system. TSE offers each of a maximum of 16 users a comprehensive library of programs for compiling, assembling, editing, loading, saving, calling, debugging, and running user programs on-line.

b. Diagnostic Programs

1. Extended Memory Address Test (MAINDEC-8E-D1FA) – This program tests all of memory (up to 32K) not occupied by the program to verify that each location can be uniquely addressed.

2. Extended Memory Checkerboard (MAINDEC-8E-D1BA) – This program is designed to provide worst-case half-select noise conditions to determine the operational status of core memory. The patterns generate worst-case noise conditions in all used fields of a PDP-8/E equipped with at least 8K of core memory.

3. Extended Memory Control and Time-Share Test (MAINDEC-8E-D1HA) – This program tests the Extended Memory Control and Time-Share option logic for proper operation. The program exercises and tests the control IOT's, time-share instruction trapping, and the ability to address all fields, program interrupt, and auto-index.

2.5 COMPANION DOCUMENTS

The following documents and publications are necessary for the operation, installation, and maintenance of this option:

a. *PDP-8/E & PDP-8/M Small Computer Handbook* – DEC, 1972

b. *PDP-8/E Maintenance Manual – Volume 1*

c. *Introduction to Programming* – DEC, 1972

d. DEC engineering drawing, Memory Extension and Time-Share Option, number E-CS-M737-0-1

e. Extended Memory Address Test, MAINDEC-8E-D1FA-D

f. Extended Memory Control and Time-Share Test, MAINDEC-8E-D1HA-D

SECTION 2 INSTALLATION

The KM8-E Memory Extension and Time-Share option is installed on-site by DEC Field Service personnel. The customer should **not** attempt to unpack, inspect, install, checkout, or service the equipment.

2.6 INSTALLATION

Perform the following procedures to install the KM8-E option:

Step	Procedure
1	Remove the module from the shipping container.
2	Inspect the module for any apparent damage.
3	Connect the module to a convenient OMNIBUS slot.

2.7 CHECKOUT

Perform the following procedure to checkout the KM8-E option:

Step	Procedure
1	Verify that the extended memory modules have been installed.
2	Verify that the corresponding jumpers have been installed to reflect the appropriate memory fields.

(continued on next page)

Step	Procedure
3	Perform acceptance tests provided in Volume 1, Chapter 2, Paragraph 2.3.
4	Load MAINDEC-8E-D1FA, Extended Memory Address Test. This program tests all of memory (up to 32K) not occupied by the program to verify that each location can be addressed uniquely.
5	Load MAINDEC-8E-D1BA, Extended Memory Checkerboard. This diagnostic program provides worst-case half-select noise conditions and verifies the operational status of core memory.
6	Load MAINDEC-8E-D1HA, Extended Memory Control and Time-Share Test. This program tests the Extended Memory Control and Time-Share option logic for proper operation. The program exercises and tests the control IOT's time-share instruction trapping; and, if time sharing is implemented, the ability to address all fields, program interrupt, and auto-index.
7	Make entry on user's log that the acceptance test for the KM8-E option was performed satisfactorily.

SECTION 3 PRINCIPLES OF OPERATION

2.8 INTRODUCTION

The KM8-E system description is given in terms of its functional operation. From the functional point of view, instructions that make either the memory extension or time-share portion do something must be considered, as well as the philosophy of why the events happen as they do. The system description, therefore, is a composite treatment of the hardware, represented in block diagram form, and of the flow of events, represented in flow diagram form. The events that occur in the Memory Extension and Time-Share option are considered fully; the events within the processor are considered only partially. Such areas as major register gating and how the processor functions are completely described in Volume 1.

2.9 SYSTEM DESCRIPTION

A block diagram representing all of the functional elements of the KM8-E Memory Extension and Time-Share option is given in Figure 2-3. The logic can be considered divided into three groups: the Control (located on the left portion of the illustration), the Instruction Field Registers (located in the center of the illustration), and the Data Field Registers (located in the far right of the illustration). The only interface is the OMNIBUS. All signals entering and leaving the system, therefore, are directed from and to the OMNIBUS. Data paths between the KM8-E and the processor are via the DATA BUS. Data is directed to the console status indicators via the DATA BUS during TS1 to tell the operator which instruction and data fields have been addressed. When the data field and/or the instruction field are to be stored in memory, the data path is from the DATA BUS to the AC Register. A DCA instruction is then used to store information in memory. The data paths between the processor and the KM8-E are via the DATA BUS and MD lines.

Special inhibiting features are designed into this option. For example, during a data break operation, when some peripheral such as a disk is being operated, control lines such as CPMA DIS prevent the transmission of data to the EMA lines. For the case of programmed interrupts, the logic provides the means of holding an interrupt back until a CIF or RMF instruction has been completely processed.

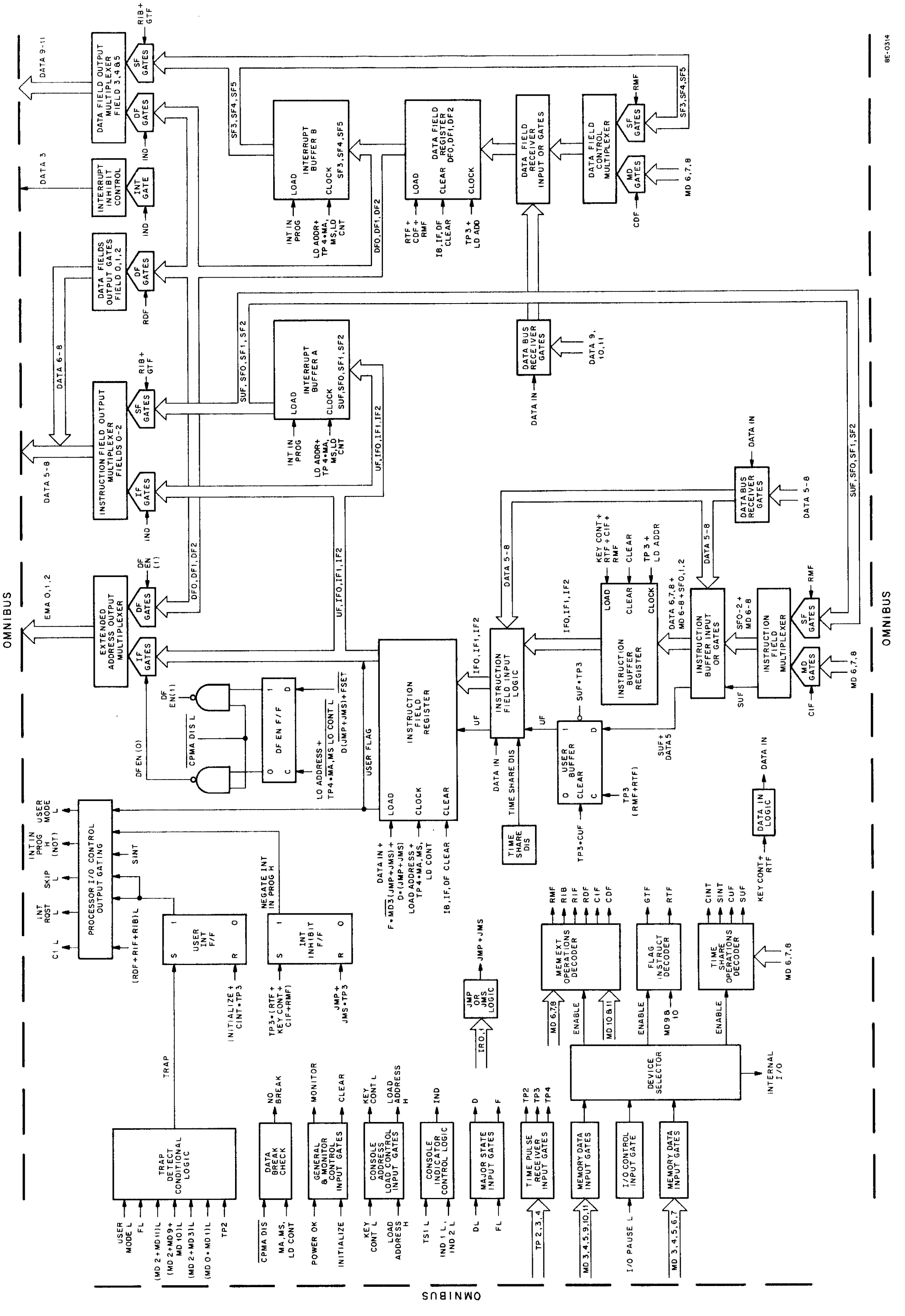


Figure 2-3 Memory Extension and Time-Share Control, System Block Diagram

Another design feature is the time-share trap logic that signals the monitor when certain instructions are used by the user.

2.9.1 Control Logic

The control logic (Figure 2-3) for the Memory Extension and Time-Share option contains elements similar to most I/O devices. These are the device selector logic, operations decoders, and the C1, INT RQST, and SKIP signal lines. Before operation begins, an IOT instruction addressed to this option is required. The INT RQST and SKIP lines are directly controlled by the USER INT flip-flop that functions to detect a TRAP signal. This flip-flop signals the monitor that a TRAP has been detected; the monitor then evaluates and takes appropriate action. The USER MODE L control signal is used in the processor to prevent the processor from responding to HLT, OSR, and IOT instructions. The INT INHIBIT flip-flop serves to ground the INT IN PROG line, thereby preventing an interrupt from occurring when instruction field changes are being processed.

For the user not desiring to use the time-share portion of the option, a simple jumper on the module prevents the User Flag (UF) flip-flop from being loaded. The instructions used with the Memory Extension and Time-Share option are listed in Table 2-1. For a detailed explanation of each instruction, refer to the KM8-E Memory option description in Chapter 7 of the *PDP-8/E & PDP-8/M Small Computer Handbook.*

Table 2-1
KM8-E Extended Memory and Time-Share Option Instructions

MEMORY EXTENSION OPERATIONS

Data Field and Instruction Field Operations	
RMF	Restore Memory Field
RIB	Read Interrupt Buffer
Data Field Operations	
CDF	Change to Data Field
RDF	Read Data Field
Instruction Field Operations	
RIF	Read Instruction Field
CIF	Change to Instruction Field
FLAG OPERATIONS	
GTF	Get the Flags
RTF	Restore the Flags
TIME-SHARE OPERATIONS	
CINT	Clear User Interrupt
SINT	Skip on User Interrupt
CUF	Clear User Flag
SUF	Set User Flag

2.9.2 Instruction Field Register and Controls

The Instruction Field Register receives new instruction field information from any one of three sources: a) Memory Data Lines, b) DATA BUS, c) Interrupt Buffer. The main registers are the Instruction Buffer Register and the Instruction Field Register. The three output lines of the Instruction Field Register are connected, via the Extended Address Output Multiplexer, to the three EMA lines.

The control logic (Figure 2-3) associated with the Instruction Field Register governs data flow, providing the necessary gating and the primary control signals to enable gating, loading, clocking, etc. This logic selects either the Save Field, the DATA BUS, or the Memory Data lines as input and outputs this information to either the Extended Address Output Multiplexer or the DATA BUS. The flow of data is from the bottom to the top of the illustration. When the contents of the Interrupt Buffer are transferred to the Instruction Field Register, an RMF instruction gates the Save Field bits through the Instruction Field Multiplexer. When the contents of the Instruction Register were previously stored in some memory location, the CIF instruction gates bits MD 6–8 through the Instruction Field Multiplexer.

The DATA BUS Receiver Gates receive data from the AC Register or the Console Switch Register. During TS3 of the FETCH state, the contents of the AC are applied to the DATA BUS, while the KM8-E Operations Decoder is decoding the RTF instruction. The RTF instruction immediately gates bits 5–8 into the Instruction Buffer Input OR Gates. This information may have been fetched from memory during the previous memory cycle or input from the Console Switch Register.

The contents of either the MD lines or the Save Field are gated through the Instruction Field Multiplexer by instruction RMF or CIF. The Instruction Buffer Input OR Gates, in turn, apply either DATA 5–8, MD 6–8, or the Save Field to the Instruction Buffer Register and apply the UF flip-flop, obtained from the Save Field, to the User Buffer. The purpose of the Instruction Buffer is to prevent the logic from prematurely loading the Instruction Field Register.

If a CIF, RTF, or RMF instruction is issued, the actual change of field does not take place until the next JMP instruction has been completed or until the EXECUTE cycle of a JMS instruction has been entered. The new instruction field is first loaded into the Instruction Buffer and then transferred from the Instruction Buffer to the Instruction Register at TP4 so that memory is not disturbed.

Although data is available for input to the Instruction Buffer Register and the User Buffer, loading does not occur until the loading lines are asserted. To load the Instruction Buffer, signal line KEY CONTROL must be grounded or one of the three loading instructions (RTF, CIF, RMF) must be asserted (grounded) and clocked in by TP3. Manual loading of data at the Console Switch Register creates LOAD ADDRESS L for the Instruction Buffer Clock input and KEY CONTROL for the buffer loading. Bits IF 0–2, representing one of eight possible instruction fields, are on the buffer output lines when the buffer is loaded. The instruction field input logic receives IF 0–2 and the UF flip-flop. Bits IF 0–2 are then ORed with DATA 5–8 unless the operator is manually loading data into the Switch Register or an RTF instruction is executed. These conditions allow only DATA 5–8 to enter the Instruction Field Register. The UF flip-flop is inhibited if the TIME SHARE DIS signal is asserted.

Program loading of the Instruction Field Register is accomplished by a directly addressed JMP or JMS instruction at the end of FETCH, or by a JMP or JMS at the end of DEFER, or by an RTF instruction. Manual loading is accomplished by signal KEY CONTROL, which is developed when the operator loads data into the Console Switch Register. The clocking for programmed loading occurs at TP4; for manual input, clocking occurs any time LOAD ADDRESS L is asserted.

Once the Instruction Field Register is loaded, the contents are ready to be loaded into either the Extended Address Output Multiplexer or the Interrupt Buffer.

2.9.3 Interrupt Buffer A

Interrupt Buffer A functions to save the contents of the instruction field when an interrupt occurs. Signal INT IN PROG H loads the UF flip-flop and bits IF 0–2 at TP4. When the interrupt has been serviced and the memory extension is again activated, the contents of the Interrupt Buffer are input to the Instruction Field Multiplexer by an RMF instruction and the field change sequence of events will be repeated. Should the programmer wish to nest interrupts, he can store the contents of the Interrupt Buffer using the GTF instruction (or RIB instruction). The machine can be restored to its original condition using the RTF (or CIF and CDF) instruction.

2.9.4 Data Field Register And Controls

The Data Field Register and Controls function to receive a new data field from any one of three sources: a) Memory Data Lines; b) DATA BUS; c) Interrupt Buffer. The output lines of the Data Field Register are connected to the Extended Address Output Multiplexer and, when selected, address the data field by means of a combination of three bits on the EMA lines.

The simplified blocks (Figure 2-3) concerned with the Data Field Register and Data Field Register operation represent the flow of data, the necessary gating, and the primary control signals to enable gating, loading, clocking, etc. They function to select either the Save Field, DATA BUS, or memory data lines as an input and output this information to the EMA lines or the DATA BUS.

The flow of data is from the bottom to the top of the illustration. When the contents of the Interrupt Buffer are transferred to the Data Field Register, an RMF instruction gates the Save Field bits through the Data Field Control Multiplexer. When a CDF instruction is executed, bits MD 6–8 are gated through the Data Field Control Multiplexer.

The DATA BUS Receiver Gates receive data from the AC Register or the Console Switch Register. During TS3 of the FETCH state, the contents of the AC are applied to the DATA BUS while the KM8-E Flag Instruction Decoder decodes the RTF instruction. The RTF gates bits 9 through 11 into the Data Field Register.

The contents of either the MD lines or the Interrupt Buffer (RMF instruction) are gated through the Data Field Control Multiplexer. At TP3, the new data field is loaded into the Data Field Register. The manual loading operation is similar to manual loading of the Instruction Field Register.

2.9.5 Interrupt Buffer B

Interrupt Buffer B saves the contents of the data field whenever an interrupt occurs. Signal INT IN PROG H loads data field bits DF 0–2 at TP4. When the interrupt has been serviced and the memory extension is to be activated, the data field is restored to the Data Field Register by the RMF instruction.

2.9.6 Extended Memory Addressing Output Control

The three EMA 0–2 lines address any one of 8 memory fields. When any combination of these lines is grounded, the result is a selected memory field. The Extended Address Output Multiplexer with the DF EN flip-flop determines whether the EMA bits will be the data field or the instruction field (Figure 2-3). A simplified flow diagram illustrating the conditions which determine the selection is shown in Figure 2-4.

Beginning at the top of the flow diagram, the logic tests for a JMP or JMS instruction. If the JMP or JMS instruction is activated, the instruction field is addressed. Otherwise, the logic tests the DEFER state. The DF EN flip-flop is clocked by TP4 or LOAD ADDRESS if the field is applied at the console switches. If a data break is in operation, the extended address will not be applied to the EMA lines at that time. As soon as the data break ends, the Extended Address Output Multiplexer will be enabled to gate either the instruction field or data field bits out to the EMA lines and thereby select a memory field.

The rule for data field usage is as follows: If the current instruction is an AND, TAD, ISZ, DCA or EAE instruction, and if the processor is currently in the DEFER state, the next EXECUTE cycle will use the data field. All other machine cycles that are not data break cycles use the instruction field.

Notice that the DEFER state is tested at the end of the current processor cycle. The decision whether the processor is to go to the EXECUTE state or the FETCH state is clearly indicated in Figure 3-17 of Volume 1. The same type of decisions that determine if the next state is to be EXECUTE or FETCH determine if the instruction field or data field is to be addressed.

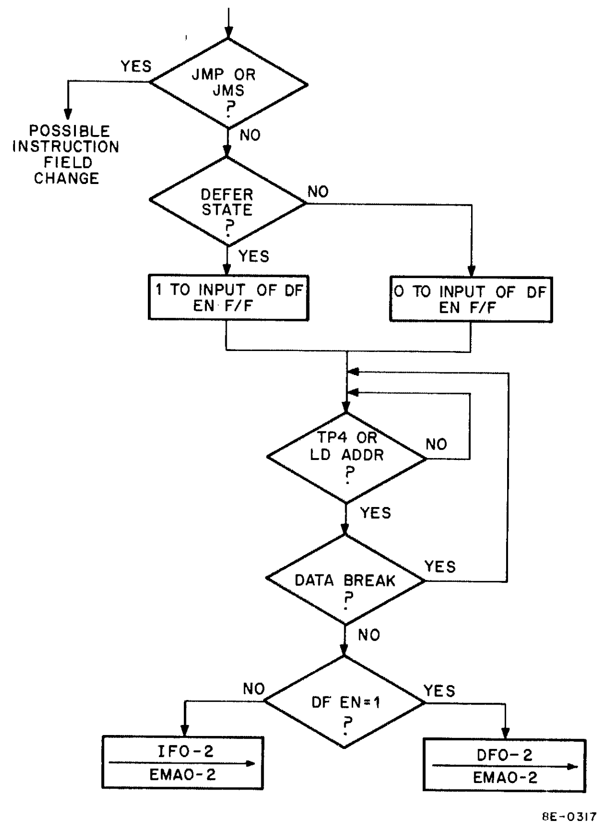


Figure 2-4 Extended Memory Addressing, Flow Diagram

2.10 OPERATING FUNCTIONS

The following paragraphs provide some examples of KM8-E operation. These descriptions reflect the flow of events for illustrative purposes and do not reflect the method by which this option is to be programmed.

2.10.1 Status Operation

Occasionally the user wants to select the STATUS position on the Console Selector Switch. When he does this, IND L will be generated at the start of TS1 and remain until TS2. Signal IND is used in the memory extension and time-share logic to gate the contents of the IF and DF Registers through the output multiplexers and onto the DATA BUS. Refer to the system block diagram given in Figure 2-3 and to Figure 2-5 for the bit arrangement illustrating the status. Only bit 3 and bits 5 through 11 represent the status of the Memory Extension and Time-Share Control.

Bit 3 represents the interrupt status of the INT INHIBIT flip-flop. Signal INT INHIBIT is generated at TP3 whenever an RTF, KEY CONTROL, CIF, or RMF signal is asserted, and negated at TP3 whenever a JMP or JMS instruction occurs. This prevents any memory field from being lost during a program interrupt. Thus, whenever bit 3 of the status indicator is illuminated, indicating a program interrupt inhibiting condition, the processor has not started a JMP or JMS instruction since the last instruction field change instruction was performed. When the IF and DF Registers are loaded and INT IN PROG H is asserted, the contents of the registers are loaded into the Interrupt Buffers at TP4.

Bits 5 through 8 represent the contents of the instruction field and the flag in the IF Register; bits 9 through 11 represent the contents of the data field in the DF Register.

2.10.2 Interrupt Buffer Transfer to Memory and Restoration

A simplified explanation of how the Interrupt Buffers could be stored in memory and restored to the Instruction Field and Data Field Registers is illustrated in Figures 2-6 and 2-7. An RIB or GTF instruction creates the necessary gating signals to allow the instruction field and data field to pass through the corresponding output multiplexers and be applied to the DATA BUS. The same instruction grounds the C1 line, which causes the contents of the DATA BUS to be loaded into the AC Register. A DCA instruction transfers the contents of the AC Register to the corresponding addressed memory location. The next time the field is to be addressed, the TAD instruction returns the data to the AC Register and an RTF instruction allows the corresponding bits to be loaded into the Instruction Buffer and Data Field Registers at TP3. The addition of PC, AC, and MQ handling allows nesting of interrupts.

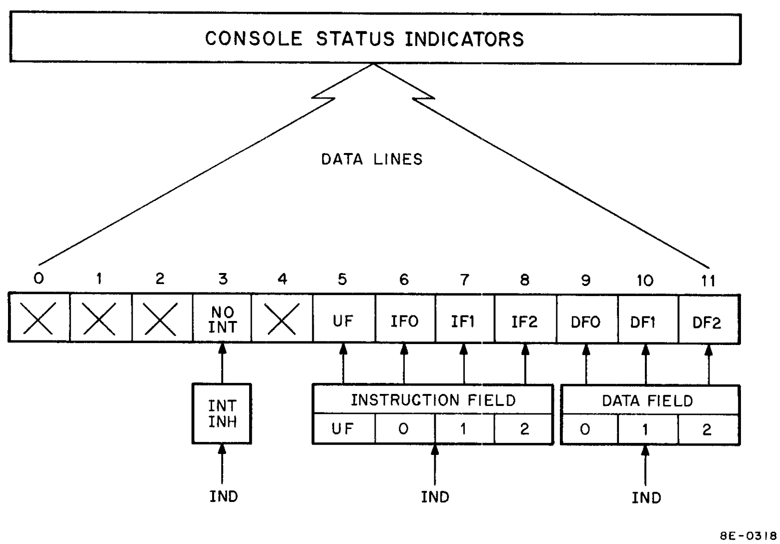


Figure 2-5 IF and DF Display Status During TS1

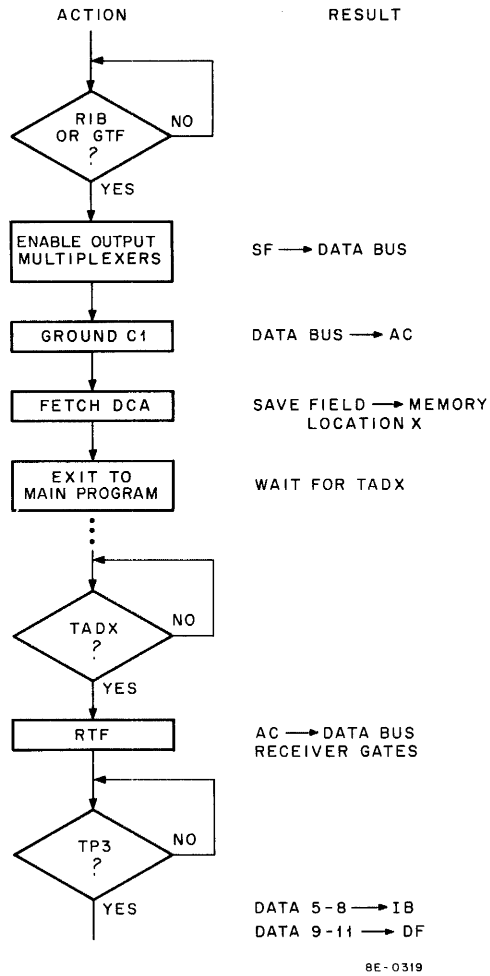


Figure 2-6 Interrupt Buffer Transfer to Memory and Restoration, Flow Diagram

2.10.3 Instruction Field Register Loading Operation

The Instruction Field Register loading operation is illustrated in the Instruction Field Register loading flow diagram (Figure 2-8). The first four decision blocks represent a **go** condition if any one of the blocks contain a status of **yes**. For example, KEY CONTROL is developed when the operator depresses the EXTD ADDR LOAD key. The corresponding clock input is LOAD ADDRESS. If the loading of the Instruction Field Register is under program control, the next three conditions are tested. An RTF instruction allows the Instruction Field Register to be loaded at TP4. Otherwise, the major states are tested. If the processor is in the FETCH state and not performing an RTF instruction, JMP or JMS is tested and finally MD3L is tested. When MD3H is present (indicating direct addressing), the Instruction Field Register will be loaded at TP4. Otherwise, a DEFER state with JMP or JMS is tested. Note that the Instruction Field Register is not loaded during a data break. The input is at TP4 to ensure that there is no address mixup during the current memory cycle.

2.10.4 Time-Share Operation of the System

Because much of the logic is shared between the time-share operation and the memory extension operation, it may not be obvious what logic is specifically dedicated to the time-share function.

The time-share portion operates in two modes as denoted by the UF flip-flop (refer to the system block diagram presented in Figure 2-3). When the UF flip-flop is in the logic 1 state, the system is operating in the user mode and a user program is running in the central processor. When the UF flip-flop is in the logic 0 state, the system is operating in the executive mode and the time-sharing monitor is in control of the central processor.

The four instructions developed by the Time-Share Operations Decoder are used by the monitor in the executive mode and are never executed by a user program. The trap logic is a monitoring device to assure the system that the user is programming valid instructions. When TRAP is developed, INT RQST is grounded and SKIP is enabled. The monitor then takes over and examines the invalid instruction and determines what action must be taken.

The UF flip-flop also plays an important role in monitoring for valid instructions. When the UF flip-flop is a 1, USER MODE L is developed and the grounded line, which goes to the processor, inhibits STOP and I/O PAUSE. The processor is operating in the executive mode during the time that memory extension or time-sharing instructions are being processed. The user mode begins when an SUF instruction has been completed. This sets the USER BUFFER flip-flop and inhibits the processor interrupt until the next JMP or JMS instruction. At the conclusion of either of these instructions, the UF flip-flop is transferred to the Instruction Field Register. At that time, the UF signal is applied to the processor I/O control output gating where USER MODE L is developed.

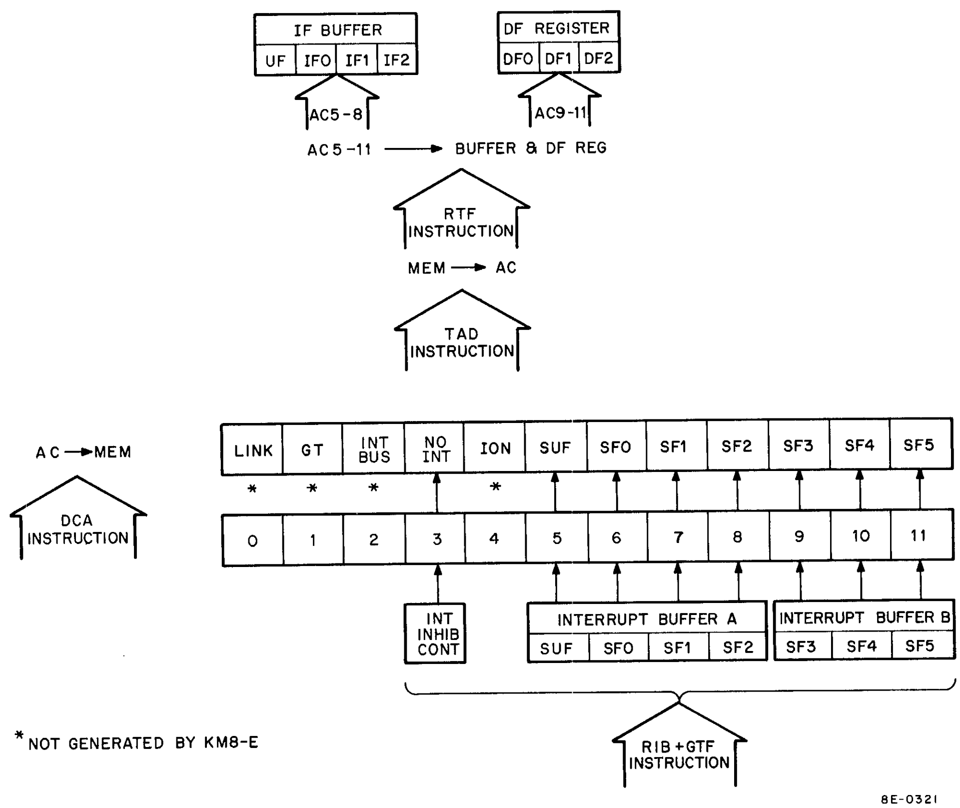


Figure 2-7 Interrupt Buffer Transfer to Memory and Restoration

The following is a summary concerning valid and trapped instructions:

FUNCTION NORMALLY:

AND
TAD
ISZ
DCA
JMP
JMS
most OPERATES

TRAPPED INSTRUCTIONS:

HLT	Monitor Return. Machine must not stop because all users would be shut down.
OSR, LAS	Requires special action. A user does not have his own Switch Register.
IOT	Requires interpretation (usually a device code change) by the monitor. For example, any user can use a KSF instruction (octal 6031). If executed by the computer, this instruction would test the flag of the console TTY (the operator). However, the monitor alters this instruction by changing the middle 6 bits to the device code of the user's TTY.

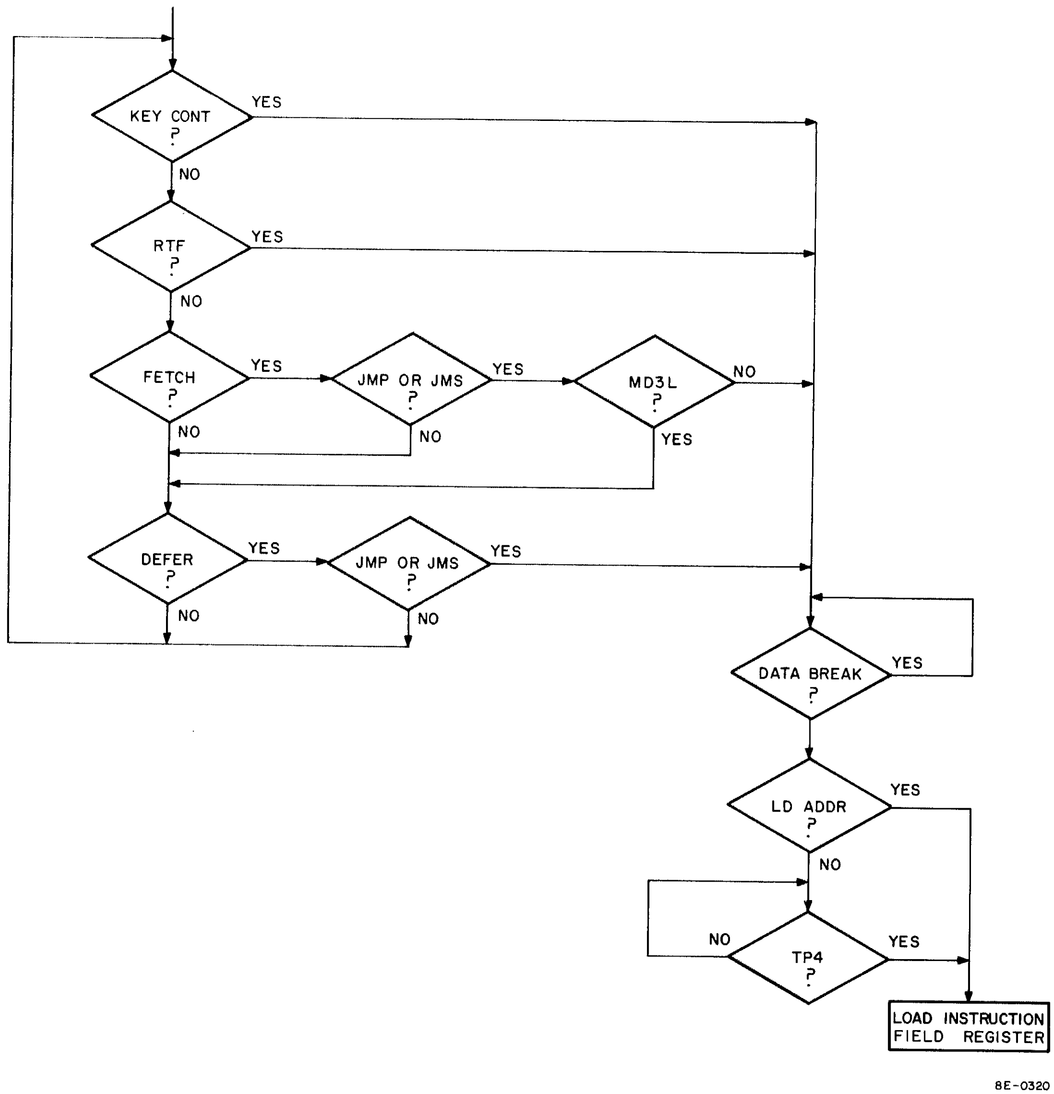


Figure 2-8 Instruction Register Loading, Flow Diagram

SECTION 4 DETAILED LOGIC

The following description represents an expansion of the Memory Extension and Time-Share Control system block diagram given in Figure 2-3.

2.11 INSTRUCTION FIELD REGISTERS

The Instruction Field Registers and gating logic are illustrated in Figure 2-9. The major parts are the Input Multiplexer, Instruction Buffer, and Instruction Field Register.

If the instruction field is to be changed, instruction CIF causes MD bits 6-8 to pass through the Input Multiplexer. If the instruction field is to be restored, instruction RMF causes the save field bits from Interrupt Buffer A to pass through the Input Multiplexer. If the data is to come from the DATA BUS, the signal DATA IN gates in bits DATA 5–8. Bit 5 holds the content of the UF flip-flop. The Instruction Buffer receives three bits. The same signals that were used to gate the bits through the Input Multiplexer are used to load the Instruction Buffer at TP3. The Instruction Field Register loads the inputs by DATA IN or during a JMP or JMS. The outputs are applied to the input gates of the Extended Address Output Multiplexer and Interrupt Buffer A.

2.12 INSTRUCTION FIELD OUTPUT MULTIPLEXER

The Instruction Field Output Multiplexer is illustrated in Figure 2-10. It consists of an 8235 IC and various control gates. Data selection lines select either the instruction field or the save field which, in turn, is applied to the DATA BUS. Signal IND L or instruction RIF selects the instruction field. Instruction GTF or RIB gates the contents of SUF and SF0 through SF2 to the DATA BUS. Bit DATA 5 is used to indicate the status of the UF flip-flop.

2.13 DATA FIELD LOGIC

The data field logic is illustrated in Figure 2-11. It consists of an 8266 IC Input Multiplexer and an 8271 IC Data Field Register. Instruction CDF is used to select bits MD 6–8; instruction RMF selects save field bits SF3 through SF5, which are to be restored from the Interrupt Buffer B. Signal DATA IN L gates in three DATA BUS bits, DATA 9–11.

The data field bits are loaded into the Data Field Register by DATA IN L or by CDF or RMF at LOAD ADDRESS time or TP3.

2.14 DATA FIELD OUTPUT MULTIPLEXER

The Data Field Output Multiplexer is illustrated in Figure 2-12. Data Field bits DF0–2 and INT INHIBIT (1) H are selected by IND L. This places these bits onto the DATA BUS during TS1 and into the Status Display on the front panel. The contents of the Interrupt Buffer can be selected by the GTF or RIB instruction.

2.15 DATA FIELD OUTPUT GATES

The Data Field Output Gates are illustrated in Figure 2-13. The contents of data field bits DF0–2 are applied to DATA 6–8 when the RDF instruction is used.

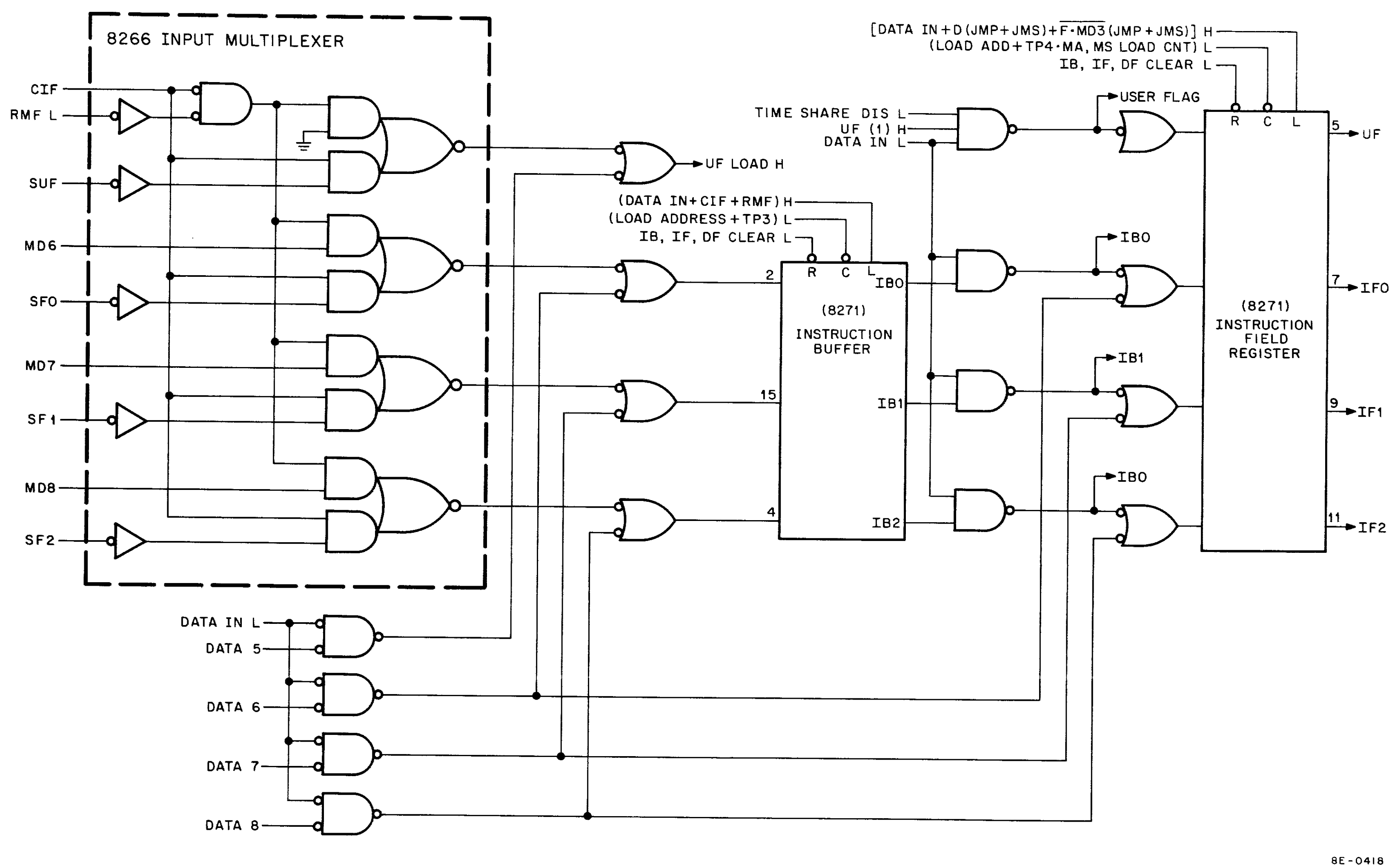


Figure 2-9 Instruction Field Registers and Gating Logic

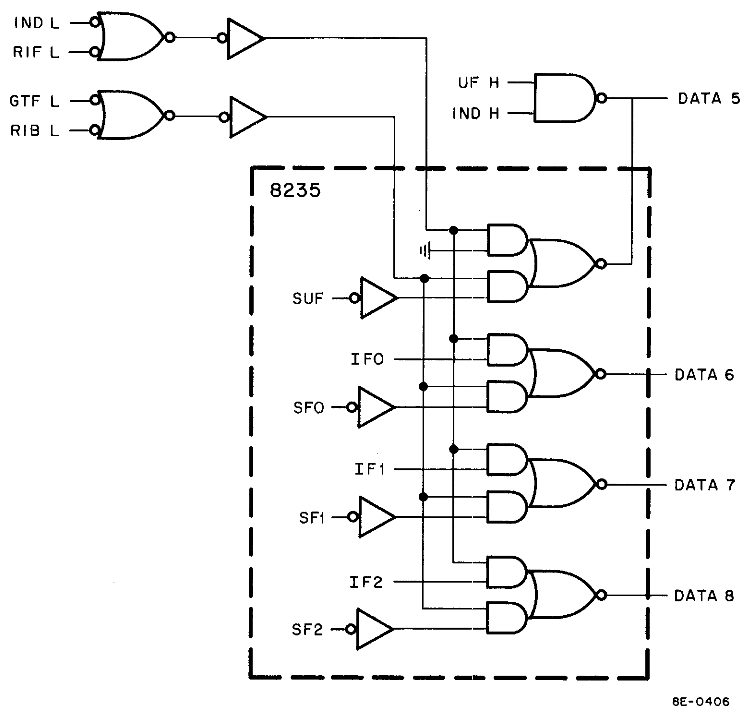


Figure 2-10 Instruction Field Output Multiplexer

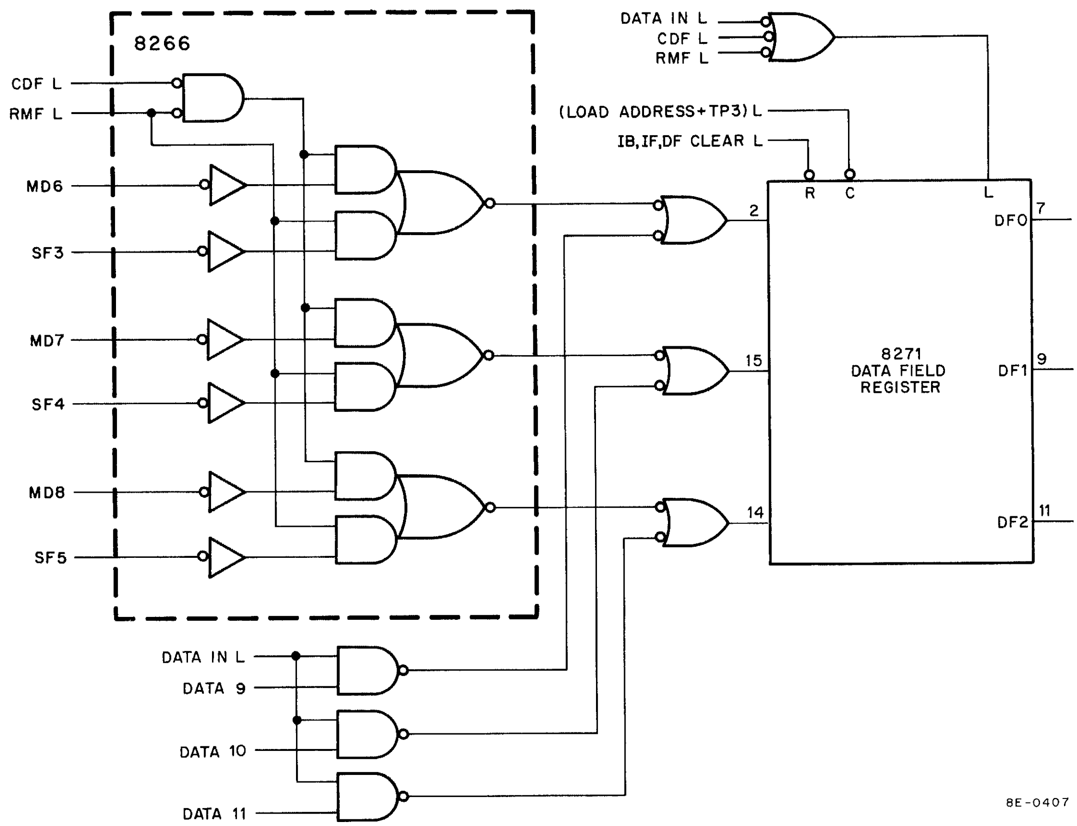


Figure 2-11 Data Field Logic

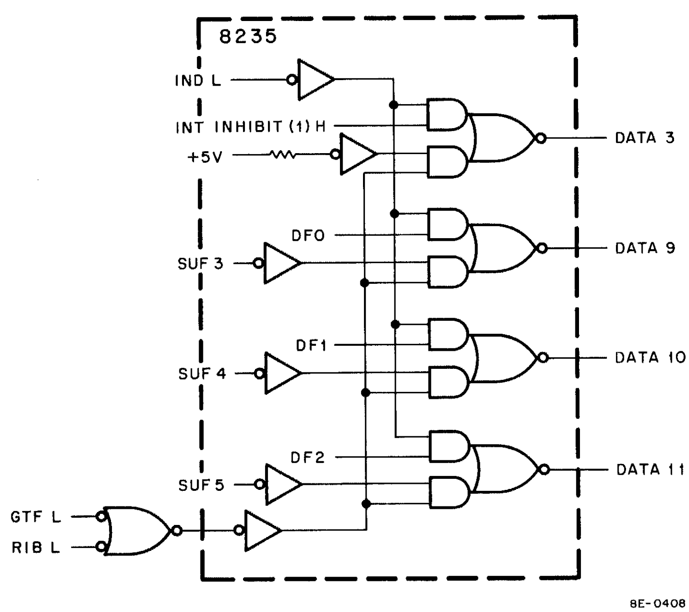


Figure 2-12 Data Field Output Multiplexer

2.16 INTERRUPT BUFFERS LOGIC

Interrupt Buffers A and B are illustrated in Figure 2-14. Buffer A is used to store the contents of the instruction field in the event of an interrupt; Buffer B is used to store the contents of the data field in the event of an interrupt. Each time an interrupt occurs, the buffers are loaded at TP4.

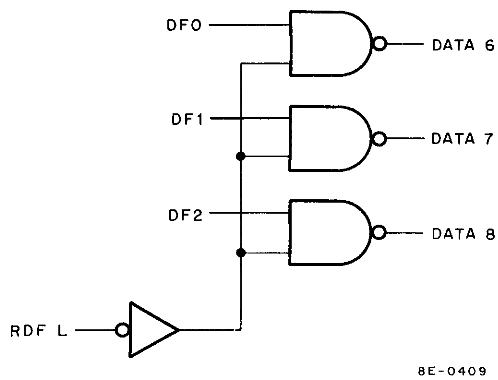


Figure 2-13 Data Field Output Gates

2.17 INTERRUPT INHIBIT LOGIC

When the processor is honoring an interrupt, INT IN PROG H is asserted. This signal is asserted at +5V, and is driven positive by a load resistor. Thus, anytime INT IN PROG H is to be negated, the signal line is simply grounded. Within the memory extension logic there is a period of time in which INT IN PROG H must not be asserted; for example, when the processor has issued a CIF instruction and has not yet encountered a JMP or JMS. These logical conditions are represented in the logic diagram illustrated in Figure 2-15. Signal INT INHIBIT (1) H is carried to DATA 3 of the DATA BUS to report to the Status Display that signal line INT IN PROG H has been negated, thereby inhibiting any interrupts.

2.18 INTERRUPT-BREAK DETECT LOGIC

The interrupt-break detect logic ensures that no critical operation, such as clearing of the IB, IF or DF registers or loading the Save Field Register, is accomplished while a data break is taking place. At the time an interrupt is being honored, both of these operations must take place. The two signal lines representing these activities (INT IN PROG H and MA, MS, LD CONT) are tested. Refer to Figure 2-16 for the detailed logic.

2.19 REGISTER CLEAR LOGIC

The Instruction Buffer, the Instruction Field Register, and the Data Field Register will be cleared at TP4 (Figure 2-17) if there is an interrupt and no data break is occurring, or if the machine is powered up.

2.20 DATA IN LOGIC

The data in logic develops DATA IN L to load the memory extension registers or to allow bits 5–8 of the DATA BUS to be gated into the instruction field logic and bits 9–11 of the DATA BUS to be gated into the data field logic. Signal DATA IN L is asserted by the program when an RTF instruction is decoded, or under manual control during TS1. Refer to Figure 2-18 for the logic diagram.

2.21 USER FLAG LOGIC

The User Flag (UF) is used only when the time-sharing portion of this option is implemented. The UF logic (Figure 2-19) is a D-type flip-flop that acts as a buffer in the same manner as the Instruction Buffer. Signal UF LOAD H is generated by the instruction field loading logic when DATA 5 is a 1 or SUF is asserted and clocked in by either an RMF or RTF instruction at TP3. The UF flip-flop can be set by instruction SUF and cleared by the instruction SUF and cleared by the logical conditions shown in Figure 2-19.

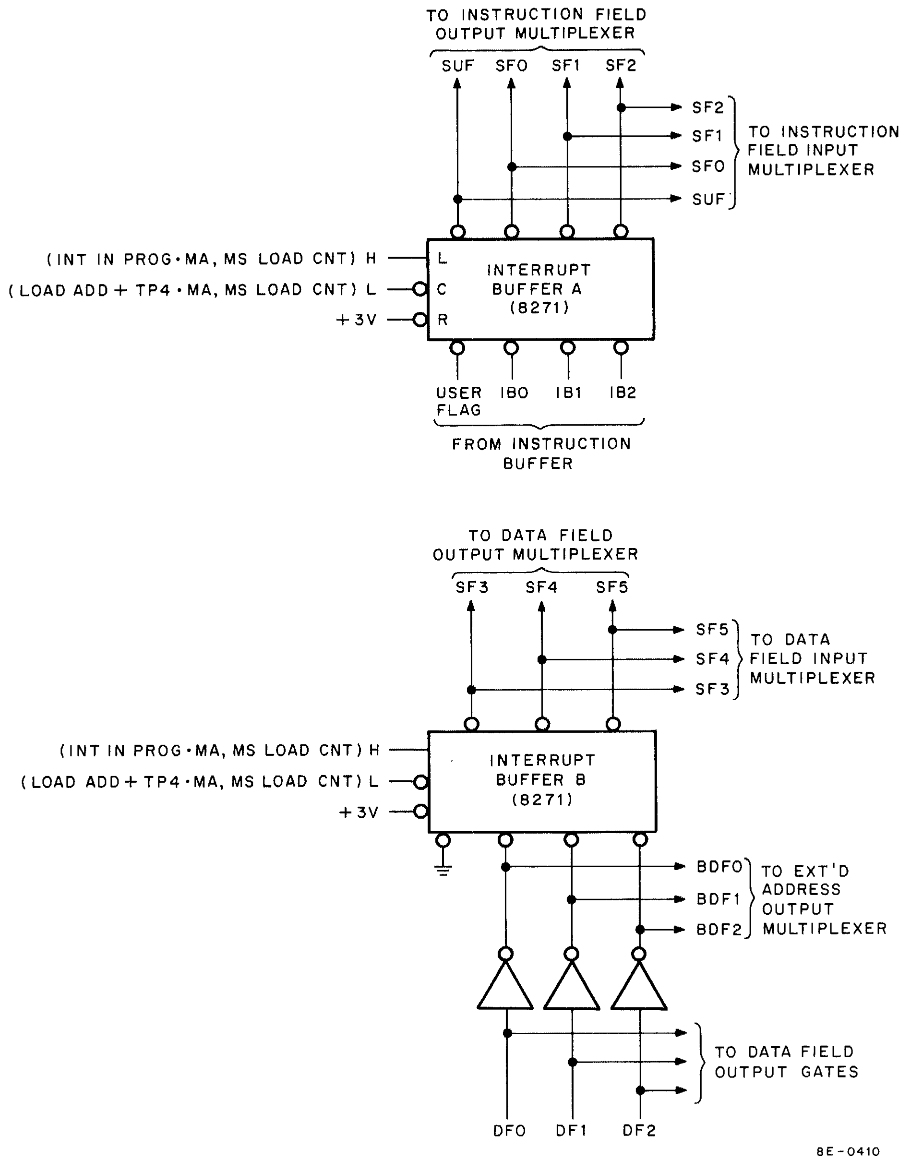


Figure 2-14 Interrupt Buffers Logic

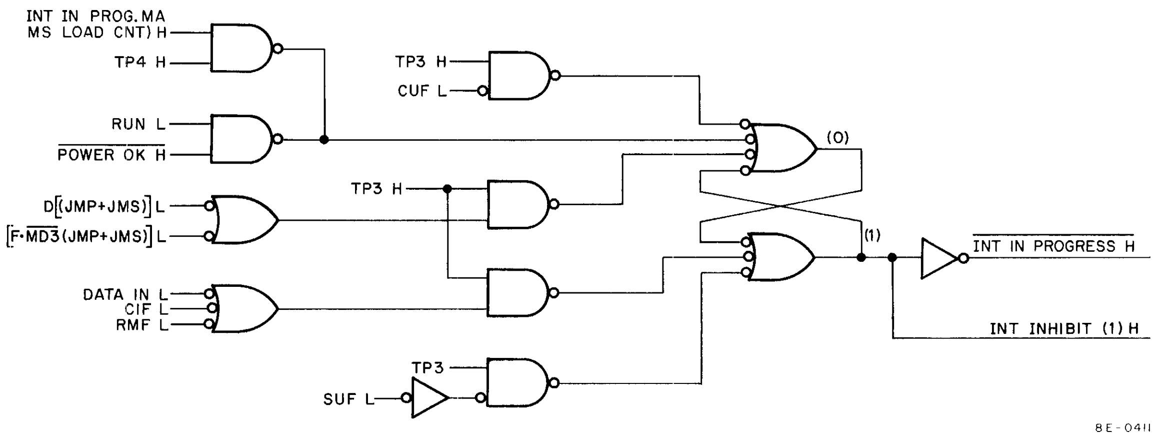


Figure 2-15 Interrupt Inhibit Logic

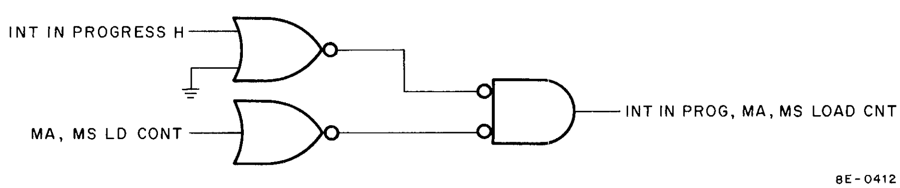


Figure 2-16 Interrupt Break Detect Logic

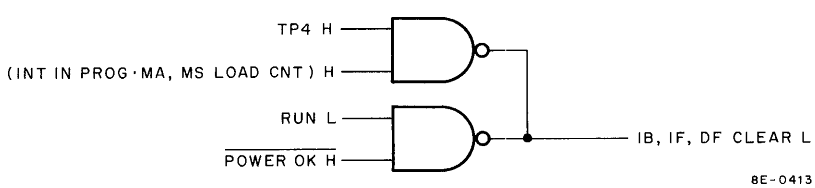


Figure 2-17 Register Clear Logic

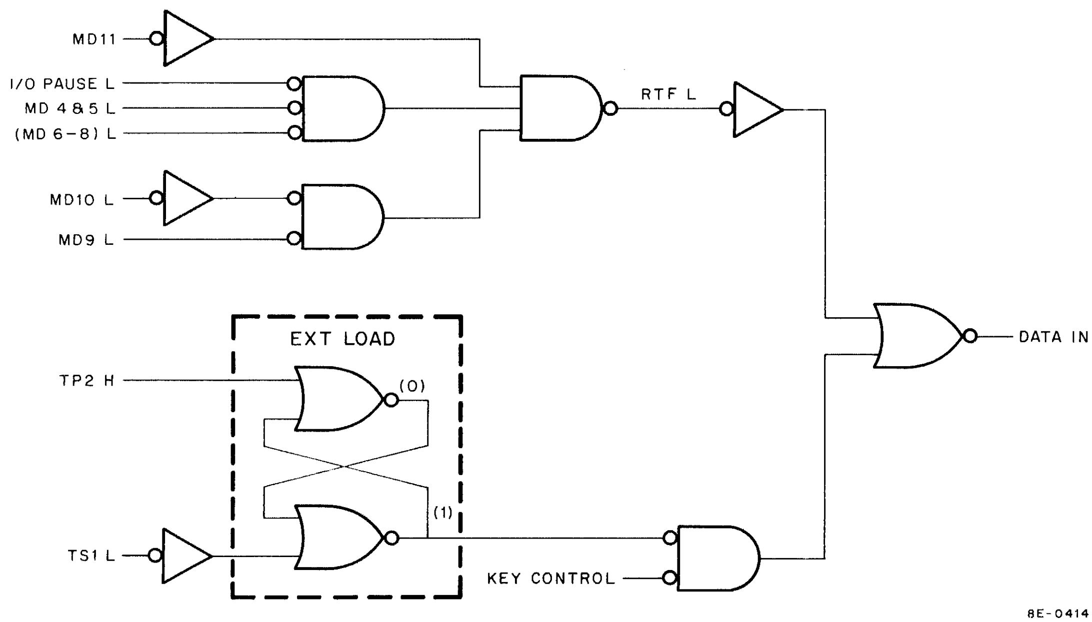


Figure 2-18 Data In Logic

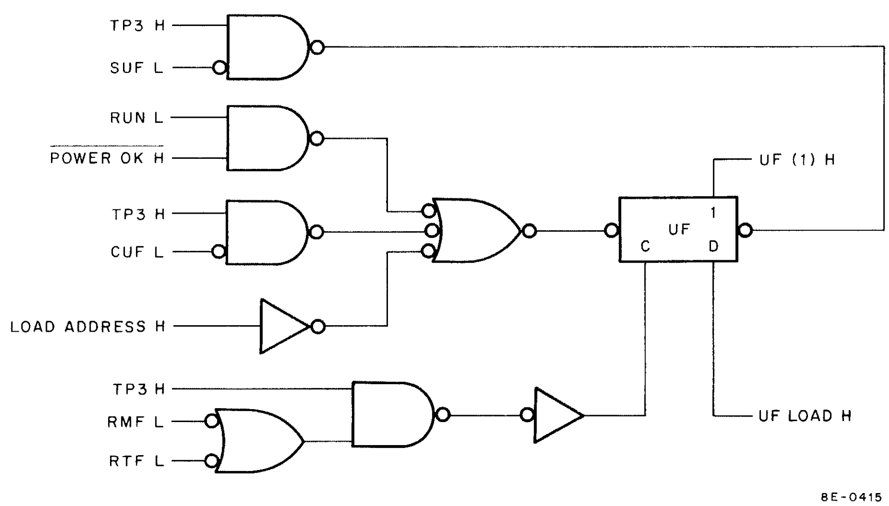


Figure 2-19 User Flag Logic

2.22 TRAP DETECT LOGIC

The trap detect logic (Figure 2-20) is used when the time-sharing portion of this option is implemented. When it is not implemented, the time-share disable circuit will contain jumpers that will hold the (0) side high and the (1) side low. Signal USER MODE L will be high because UF (1) will be low. Signal TRAP is high if the MD lines indicate an IOT (F.UM.6xxx) or special operate (F.UM.74x1, where bits 9 or 10 are 1s).

2.23 EXTENDED MEMORY ADDRESS LOGIC

The extended memory address logic (Figure 2-21) includes the output gates within the 8235 IC and the DF EN and EMA DISABLE flip-flops with corresponding control logic. The output gates, IC 8235, are a multiplexer of which the data selected for output is determined by the EMA DISABLE and DF EN flip-flops. The EMA DISABLE flip-flop serves the same function as the MAC flip-flop in the M8310 CPU Control Module, and is used to disable the output multiplexer at TP4 in the event of a data break.

The interrupt break logic signal clears the EMA DISABLE flip-flop at TP4 or when the machine is in the manual mode. The DF EN flip-flop determines if the output data is the data field or the instruction field. If the processor is not doing a JMP or JMS instruction, but is in the DEFER state, the data field will be addressed during the next machine cycle. If the processor is doing a JMP or JMS or is not in the DEFER state, the instruction field will be addressed.

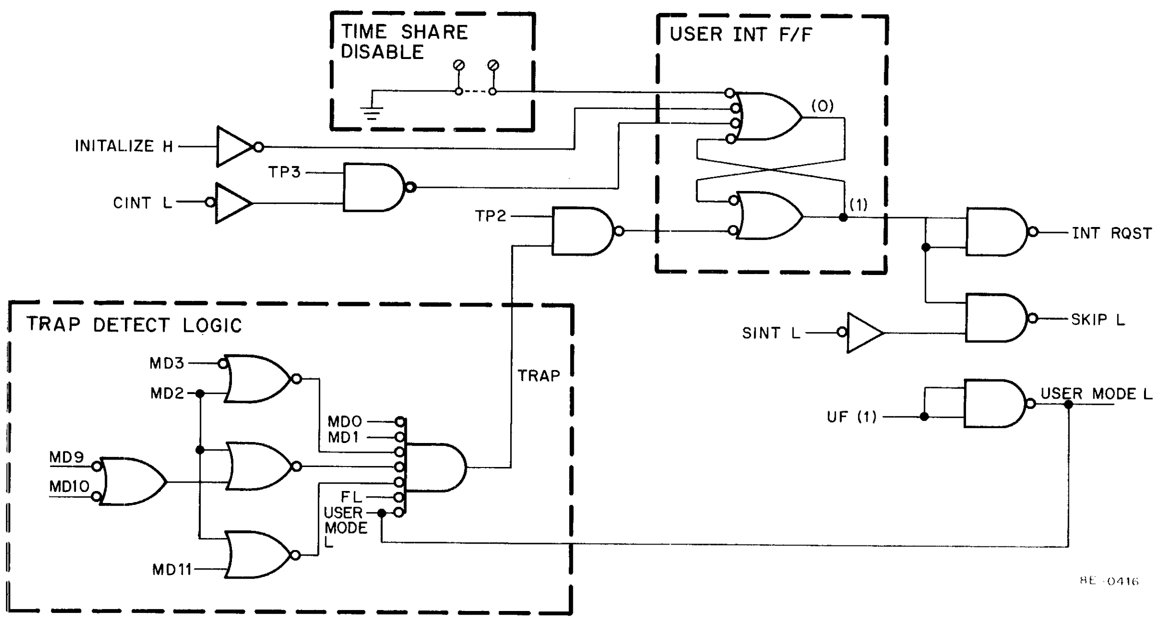


Figure 2-20 Time-Share Trap Logic

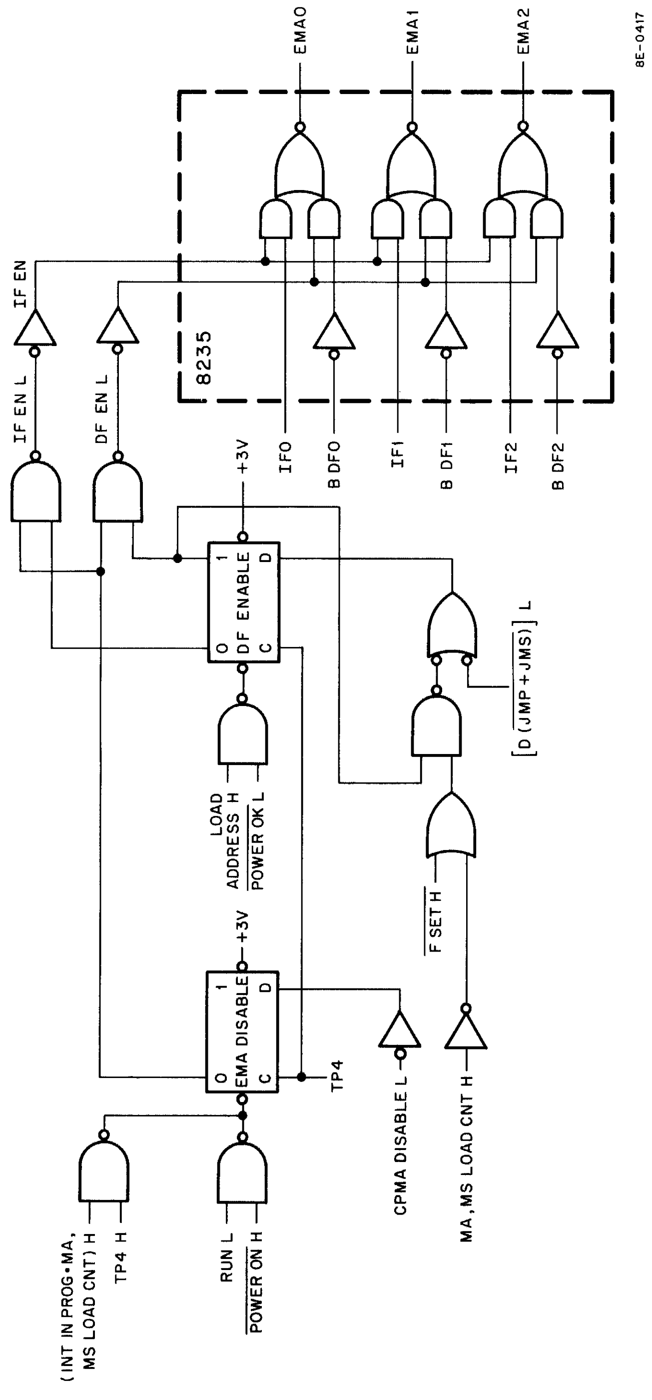


Figure 2-21 Extended Memory Address Logic

SECTION 5 MAINTENANCE

The general procedures concerning preventive and corrective maintenance are given in Volume 1, Chapter 4. When corrective maintenance is required, the technician should use the maintenance programs given in Section 2 of this chapter to determine the nature of the problem. Refer to the option schematic, drawing number E-CS-M737-0-1, for IC locations and pin numbers. Test points have been provided on the module to facilitate troubleshooting.

SECTION 6 SPARE PARTS

Table 2-2 lists recommended spare parts for the KM8-E. These parts can be obtained from any local DEC office or from DEC, Maynard, Massachusetts.

Table 2-2
Recommended KM8-E Spare Parts

DEC Part No.	Description	Quantity
19-05575	IC DEC 7400	1
19-09705	IC DEC 8881	1
19-09615	IC DEC 8271	1
19-09935	IC DEC 8235	1
19-09934	IC DEC 8266	1
19-09594	IC DEC 8251	1
19-09667	IC DEC 74H74	1
19-05547	IC DEC 7474	1
19-05577	IC DEC 7420	1
19-05576	IC DEC 7410	1
19-09686	IC DEC 7404	1
19-09056	IC DEC 74H00	1
19-09486	IC DEC 384A	1
19-09972	IC DEC 6314A	1
13-00365	Resistor 1K, 1/4W, 5%	1
10-01610	Capacitor 0.01 MF DISK, 20%	6
10-05306	Capacitor 6.8 μF, 35V, 10%	1

CHAPTER 3
MR8-E READ-ONLY MEMORY

SECTION 1 INTRODUCTION

3.1 READ-ONLY MEMORY DESCRIPTION

The MR8-E is a 256-word Read-Only Memory (ROM) option used in the PDP-8/E. The MR8-E consists of an M880 Quad Module that is inserted into the OMNIBUS and an H241 Braid Board mounted on the M880 Module. The thickness of the module and braid board requires that two spaces be allotted for this option on the OMNIBUS, although the MR8-E plugs into only one of these spaces. Each ROM option occupies two pages (400_8 locations) of the 32_{10} pages (7777_8 locations) in each field. The MR8-E can be located starting at the beginning of any even-numbered page in any field, such as 0000_8, 00400_8, or 64400_8. Note that the corresponding core memory locations cannot be used by the software while the MR8-E is installed in the OMNIBUS. When a memory location assigned to the MR8-E is addressed, ROM ADD L will be asserted which, in turn, disables core memory.

From a programming point of view, and as viewed from the OMNIBUS, the MR8-E is addressed the same way as core memory (Paragraph 3.24, Volume 1). Within the MR8-E these 400_8 words are organized as 200_8 lines, each running through or around 24 ferrite cores (two 12-bit words). Each drive line is terminated by a diode on one end and by a switch tied to a decoder on the other end. The memory locations are selected by MA00 to MA11 and EMA00 to EMA02. The MR8-E is interfaced to the processor by the OMNIBUS.

SECTION 2 INSTALLATION

The MR8-E will be installed on site by DEC Field Service personnel. The customer should **not** attempt to unpack, inspect, install, checkout, or service the MR8-E Module.

3.2 INSTALLATION

Perform the following steps to install the MR8-E Read-Only Memory:

Step	Procedure
1	Remove power from the PDP-8/E by turning Power Switch to OFF.
2	Ensure proper diodes are installed in the M880 Module to select the starting address of ROM (drawing CS-M880-0-1).
3	Insert the MR8-E (M880 and H241) into the OMNIBUS (refer to Table 2-3, Volume 1) for module installation priority.

3.3 ACCEPTANCE TEST

Perform the following steps to check the MR8-E Modules:

Step	Procedure
1	Load ROM Test Tape Low (MAINDEC-8E-D1JA-PB1) or ROM Test Tape High (MAINDEC-8E-D1JA-PB2). Refer to diagnostic write up for correct loading procedures.
2	Allow the diagnostic to run for 20 minutes with no errors.

NOTE
Refer to Section 5 for the procedure to change ROM contents if errors are found.

SECTION 3 SYSTEM DESCRIPTION

3.4 MR8-E BLOCK DIAGRAM

The MR8-E consists of an M880 Drive and Sense Module and an H241 Braid Board Assembly. The M880 Module contains the logic for addressing 256 words of memory located on the H241 Braid Board Assembly. The MR8-E is addressed in the same way as PDP-8/E Core Memory (Paragraph 3.24, Volume 1). If the MR8-E is placed in the same field with a 4K core memory, the two pages of core memory with the same addresses as ROM cannot be accessed by the program. If the software tries to write in ROM, the new information will be lost and the contents of ROM will not be changed. When the MR8-E detects an address to which it must respond it asserts ROM ADD L (Figure 3-1) which disables core memory. The 400_8 words of the MR8-E memory are organized as 200_8 lines with each line containing two words. The 200_8 lines run through or around 24_{10} ferrite cores and sense windings (Figure 3-2). Each of the 200_8 lines is terminated by a diode at one end and in 8 groups of 16_{10} lines tied to the outputs of a BCD decoder at the other end.

3.5 ROM ADDRESSING

Addressing of the 400_8 locations is accomplished as follows.

a. The three EMA lines and the four most-significant MA lines (MA00–MA03) are decoded in Address Selection AND gate within the MR8-E to determine whether the currently addressed location is within the option. The Address Selection gate consists in part of 14_{10} diodes, seven of which must be removed to define the combination of EMA and most-significant MA bits for which the MR8-E is active. If the MR8-E is selected, a gate at the output of the Address Selection gate asserts (grounds) the ROM ADD L line on the OMNIBUS, thereby disabling the core memory that would normally respond to that address. In addition, a second output of the Address Selection gate (labeled FIELD L on the logic diagrams) enables decoding of the remaining MA lines.

b. MA04 through MA07 are decoded and select one of 20_8 (16_{10}) drivers, each of which is connected to 10_8 (8_{10}) diodes.

c. MA08 through MA10 select one output of the line select decoder which is a BCD-to-decimal decoder. One of 10_8 outputs from the BCD-to-decimal decoder is selected and forward biases the diode selected by MA04 through MA07 so that current will flow through the selected line and induce a signal in the sense winding if it passes through the core. Data is taken from the MR8-E using high-permability ferrite U- and I-cores mated to form a closed magnetic path. The 200_8 lines run through or around the 24_{10} U-cores, the 24 I-cores are wound with a 50-turn sense winding. When current is driven through a selected line, which passes through the mated core, a 2.5V to 3.0V signal is magnetically induced in the winding. The signal induced in the sense winding is fed to a DTL-type gate and clocked by a strobe pulse. If the selected line passes around (not through) the mated core, no signal is induced in the sense winding.

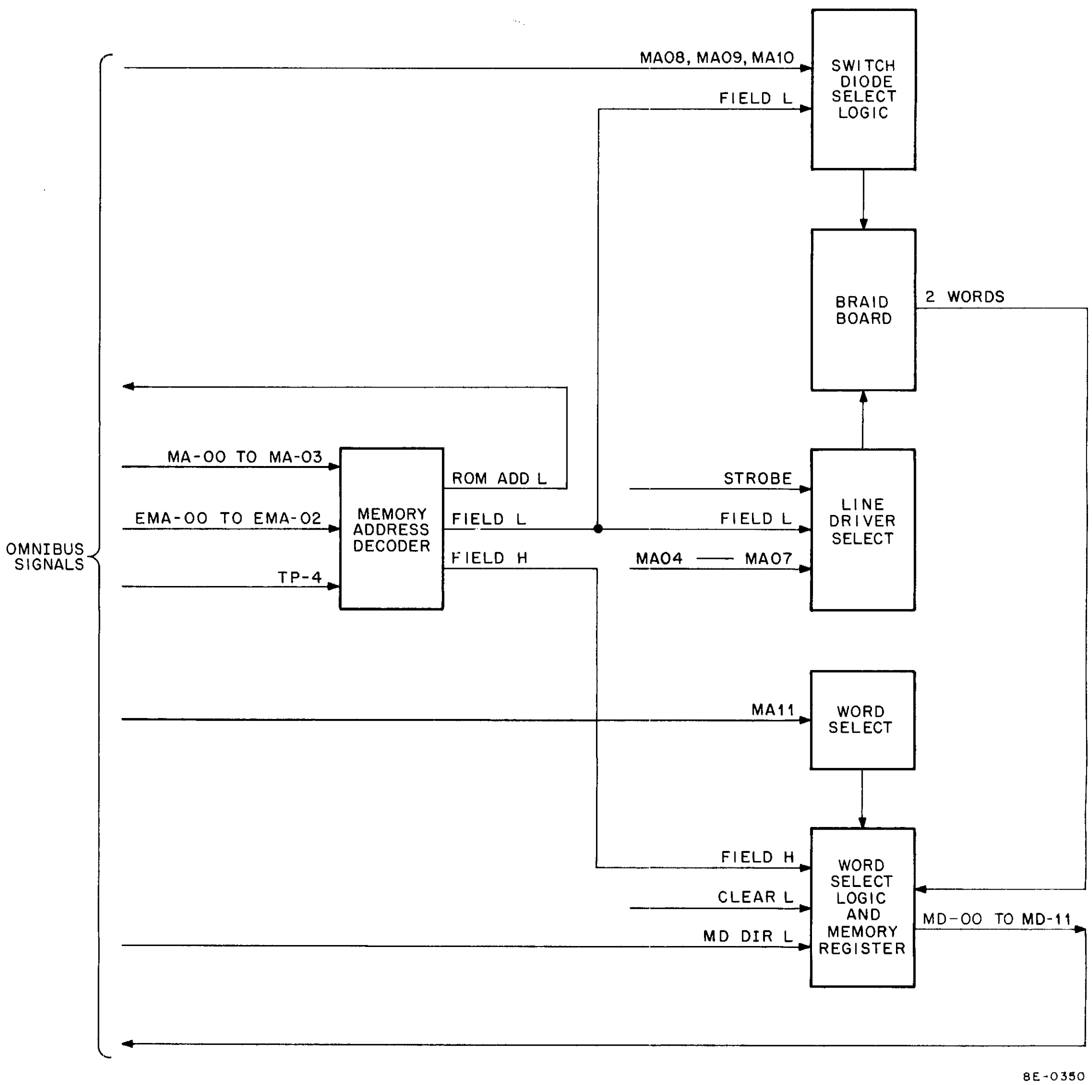


Figure 3-1 Read-Only Memory Block Diagram

d. The selected line causes a 24-bit word to be read. MA11 controls which half of the addressed word is applied to the Memory Register. The output of the DTL gates is clocked into a 12-bit Memory Buffer Register.

e. FIELD H, NOT CLEAR L and MD DIR L enable the output of the Memory Register to the MD lines.

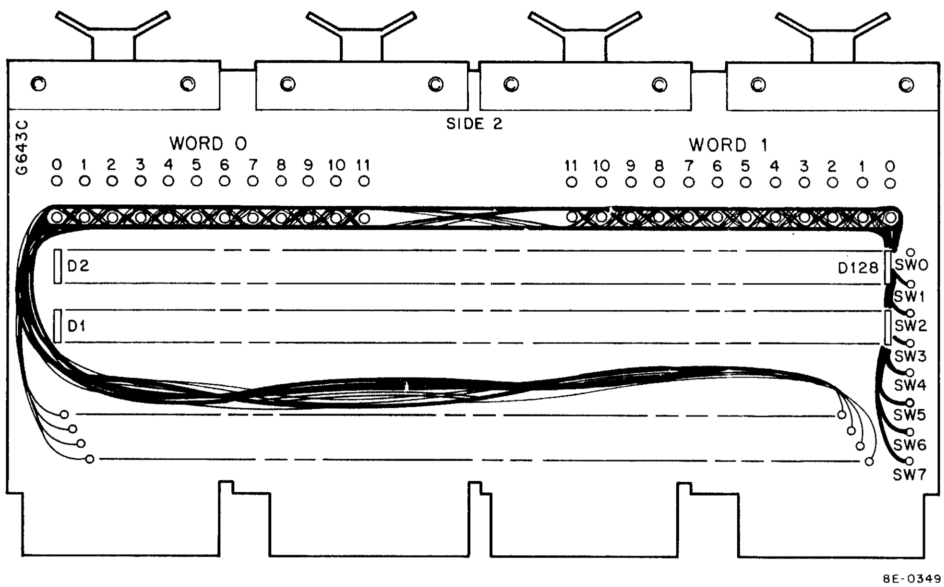


Figure 3-2 H241 Braid Board

SECTION 4 DETAILED LOGIC

The logic in the MR8-E is broken into functional groups for discussion purposes. The block diagram in Figure 3-1 should be used to understand the interaction of the groups of logic.

3.6 ADDRESS DECODER

The Address Decoder (Figure 3-3) receives bits MA00 through MA03 and EMA00 through EMA02 for decoding. The 14 diodes (7 of which are removed) select the high-order address of ROM. If the MA and EMA bits indicate the selected address is within ROM FIELD L, ROM ADD will be asserted, so that only ROM can be accessed by the program. FIELD L enables gates to allow MA04 through MA11 to be decoded and used to select a line driver, switch, and word 0 or 1.

3.7 SWITCH SELECT LOGIC

The switch select logic (Figure 3-4) decodes MA08 through MA10 and selects one of the groups of diodes on the H241 Braid Board Assembly using the proper switch line. FIELD L is used to enable E26 AND gates and the output of the AND gates is applied to the 74145 IC. The 74145 IC is a BCD-to-decimal decoder that pulls one of the output lines low when the 3-bit input is decoded. The output of E27 is applied to the 8 switches that enable the selection of the proper diode and one of the 16 line drivers (Figure 3-5).

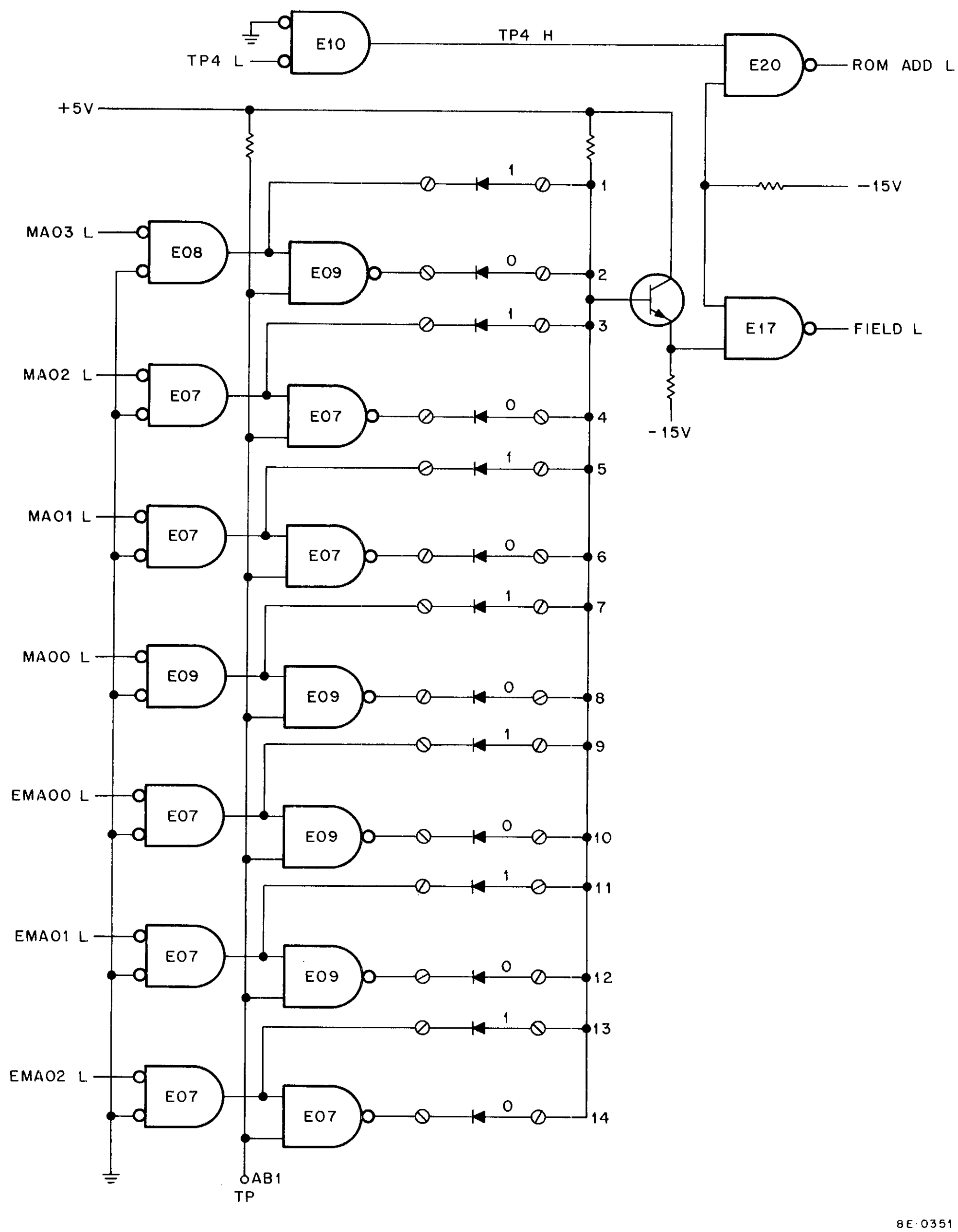


Figure 3-3 Address Decoder Logic

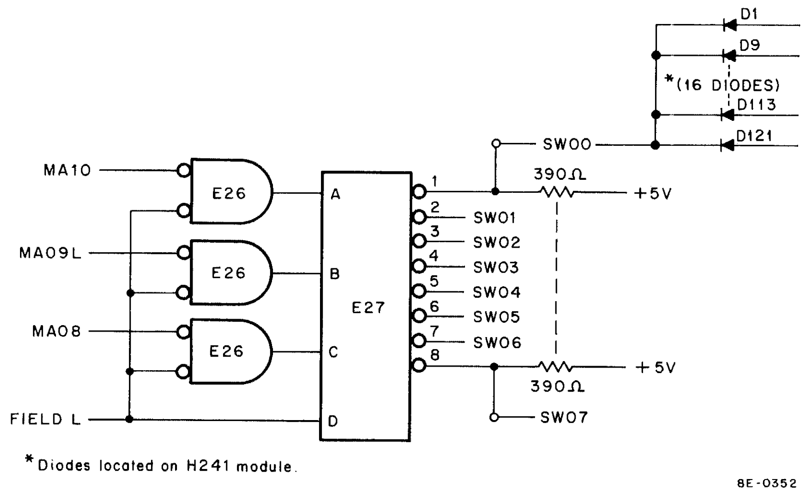


Figure 3-4 Switch Select Logic

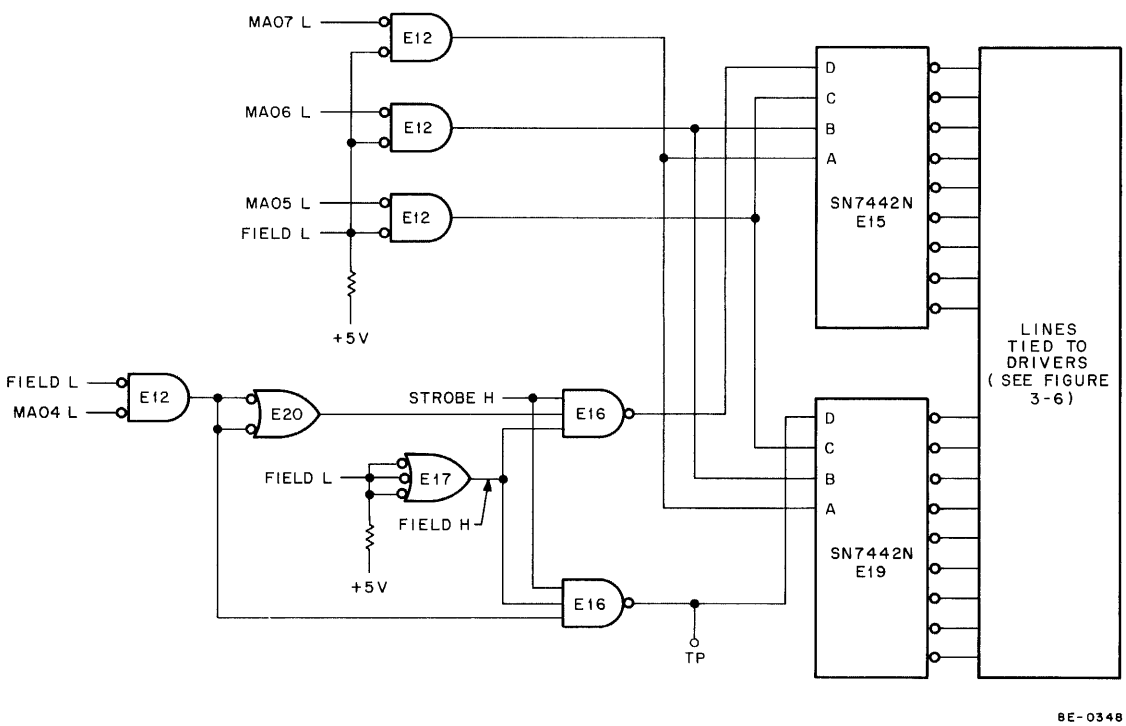


Figure 3-5 Line Driver Select Logic

3.8 LINE DRIVER SELECT LOGIC AND LINE DRIVERS

The line driver select logic (Figure 3-5) decodes MA04 through MA07 and selects one of the 16 line drivers. Each line driver has 8 diodes in its collector circuit, each of which is tied to one end of the 8 switch lines on the braid board (Figure 3-6). There are 16 line drivers, with 8 diodes each, to select 128 lines (256 words) on the braid board. The E12 AND gates are enabled by FIELD L; MA05 through MA07 are applied to E15 and E19. E15 and E19 are BCD-to-decimal decoders that assert one output line for each input and cause the line driver to forward bias 8 diodes. MA04 is used to select a control input at pin D on E15 or E19. When MA04 is 1 (low), E20 is disabled and input D of E19 has a low enabling input that allows E19 to pull one output low and select one of its line drivers. When MA04 is 0 (high), E20 and E16 are enabled and input D of E15 has a low (enabling) input. Note that each line driver has 8 diodes tied to 8 different switch lines in the diode matrix (Figure 3-6). When the diode is selected by a line driver and the switch on the other end of a line is selected, current flows through that line and induces a voltage in the sense windings of those cores that have a line passing through them. Table 3-1 lists the lines in the braid board and the two words associated with each line.

The diode tied to each line has the same numerical designation as the line, i.e., line 1 has D1 tied to one end.

3.9 SENSE LOGIC AND MEMORY REGISTER

The sense windings of the 24 ferrite cores are tied to DTL AND gates (Figure 3-7) on the M880 Module. As stated previously, when current flows through the selected line a voltage is induced in the sense windings shown as A1 through A11 for word 0, and B1 through B11 for word 1. MA11 applied to E26 allows one of the words to be applied to the Memory Register when FIELD L and STROBE L are applied to E24. If MA11 is 0, word 0 (sense winding A) is applied to the Memory Register; if MA11 is 1, word 1 (sense winding B) is applied to the Memory Register.

The Memory Register is made up of 12 set-reset type flip-flops that hold the 12 bits of data until MD DIR L, NOT CLEAR L and FIELD H are received. The result enables the AND gates and applies the contents of the Buffer Register to the MD lines.

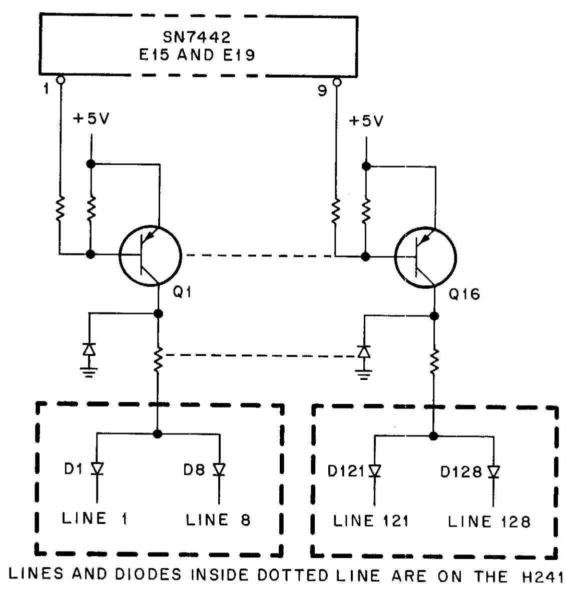


Figure 3-6 Line Drivers and Diodes

Table 3-1
Line and Switch Identification for ROM Addresses

Line Number	Addresses*	Sw. Term	Line Number	Addresses*	Sw. Term
1	000,001	0	41	120,121	0
2	002,003	1	42	122,123	1
3	004,005	2	43	124,125	2
4	006,007	3	44	126,127	3
5	010,011	4	45	130,131	4
6	012,013	5	46	132,133	5
7	014,015	6	47	134,135	6
8	016,017	7	48	136,137	7
9	020,021	0	49	140,141	0
10	022,023	1	50	142,143	1
11	024,025	2	51	144,145	2
12	026,027	3	52	146,147	3
13	030,031	4	53	150,151	4
14	032,033	5	54	152,153	5
15	034,035	6	55	154,155	6
16	036,037	7	56	156,157	7
17	040,041	0	57	160,161	0
18	042,043	1	58	162,163	1
19	044,045	2	59	164,165	2
20	046,047	3	60	166,167	3
21	050,051	4	61	170,171	4
22	052,053	5	62	172,173	5
23	054,055	6	63	174,175	6
24	056,057	7	64	176,177	7
25	060,061	0	65	200,201	0
26	062,063	1	66	202,203	1
27	064,065	2	67	204,205	2
28	066,067	3	68	206,207	3
29	070,071	4	69	210,211	4
30	072,073	5	70	212,213	5
31	074,075	6	71	214,215	6
32	076,077	7	72	216,217	7
33	100,101	0	73	220,221	0
34	102,103	1	74	222,223	1
35	104,105	2	75	224,225	2
36	106,107	3	76	226,227	3
37	110,111	4	77	230,231	4
38	112,113	5	78	232,233	5
39	114,115	6	79	234,235	6
40	116,117	7	80	236,237	7

*These addresses are within the MR8-E. To get the absolute address, add the starting address of the MR8-E to the MR8-E addresses, i.e., MR8-E starts at 4400, line 123 contains absolute addresses 4764 and 4765. 4764 is word 0 and 4765 is word 1. The word will be selected by Bit 11 applied to the MR8-E.

Table 3-1 (Cont)
Line and Switch Identification for ROM Addresses

Line Number	Addresses*	Sw. Term	Line Number	Addresses*	Sw. Term
81	240,241	0	105	320,321	0
82	242,243	1	106	322,323	1
83	244,245	2	107	324,325	2
84	246,247	3	108	326,327	3
85	250,251	4	109	330,331	4
86	252,253	5	110	332,333	5
87	254,255	6	111	334,335	6
88	256,257	7	112	336,337	7
89	260,261	0	113	340,341	0
90	262,263	1	114	342,343	1
91	264,265	2	115	344,345	2
92	266,267	3	116	346,347	3
93	270,271	4	117	350,351	4
94	272,273	5	118	352,353	5
95	274,275	6	119	354,355	6
96	276,277	7	120	356,357	7
97	300,301	0	121	360,361	0
98	302,303	1	122	362,363	1
99	304,305	2	123	364,365	2
100	306,307	3	124	366,367	3
101	310,311	4	125	370,371	4
102	312,313	5	126	372,373	5
103	314,315	6	127	374,375	6
104	316,317	7	128	376,377	7

*These addresses are within the MR8-E. To get the absolute address, add the starting address of the MR8-E to the MR8-E addresses, i.e., MR8-E starts at 4400, line 123 contains absolute addresses 4764 and 4765. 4764 is word 0 and 4765 is word 1. The word will be selected by Bit 11 applied to the MR8-E.

3.10 CLEAR LOGIC

The clear logic (Figures 3-8 and 3-9) clears the Memory Register and enables gates to place data from the Memory Register on the MD lines of the OMNIBUS. The Memory Register is cleared when RETURN H goes high (100 ns into the memory cycle) clearing the CLEAR flip-flop.

At STROBE TIME, the CLEAR flip-flop is set, enabling the Memory Register output gates (Figure 3-7) to transfer the contents of the Memory Register to the MD lines, when MD DIR is asserted (MD DIR on the OMNIBUS is low).

3.11 LINE SELECT DIODE MATRIX

The line select diode matrix (Figure 3-10) selects one of the lines which pass through or around the ROM core and allows current to flow through one line for each address. One end of each line is tied to a switch; the other end is tied to a diode. The diode is in the collector of a line driver (DR0 to DR15) and the selected diode is

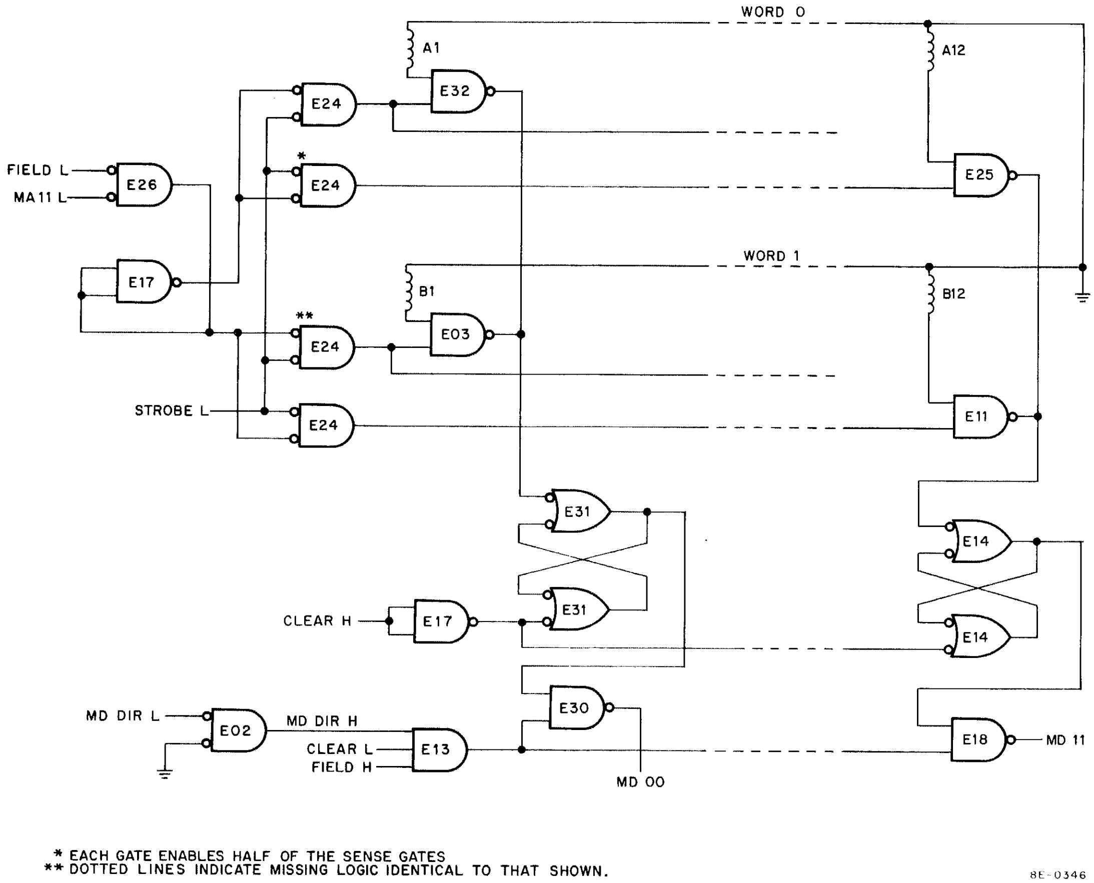


Figure 3-7 Sense Logic and Memory Register

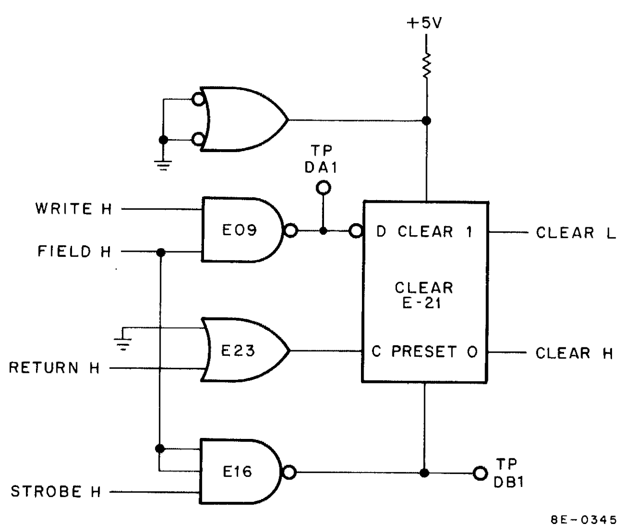


Figure 3-8 Clear Logic

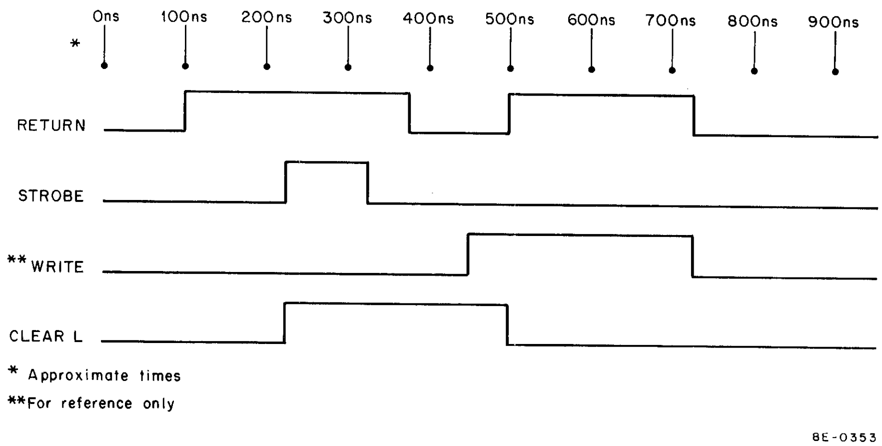


Figure 3-9 Read Timing Diagram

forward biased by the switch select logic to allow current flow through the line. Current flow through the line causes a voltage to be induced in the sense windings of the cores through which the line passes. This voltage is applied to the DTL logic (Figure 3-7) and gated to the Memory Register by MA11 and STROBE L.

SECTION 5 MAINTENANCE

The MR8-E diagnostic (MAINDEC-8E-D1JA-PB1 or MAINDEC-8E-D1JA-PB2) should be run when a ROM malfunction is suspected. Use the following procedure to change the contents of ROM or to correct errors.

Step	Procedure
1	Remove the H241 with cover and standoffs from M880.
2	With Side 2 of the H241 up, find the line that corresponds to the address of the word to be changed. Each line contains two words, so the 128 lines contain 256 addresses (Table 3-1). Cut the line to be changed.
3	Solder the new wire to the lug as follows: a. If you are placing new words in these addresses, string the wire through all 24 cores using all the tie down jumpers. The 24 cores correspond to the two 12-bit words of the line (Figure 3-2). For a logical 1, string the wire inside the "U" core; for a 0, string the wire outside the core. Terminate the wire on the proper switch (Table 3-1). b. If you are correcting an error in the MR8-E, the diagnostic program will provide all the information needed. A typical typeout would be as shown in Table 3-2. For this error, cut Line 5, replace it, and string wire through the cores as shown in the insert portion of the typeout. The 24 bits shown correspond one-to-one with the 24 cores on the board, from left to right. The line should be terminated on SW4. If an error occurs in the field, check the diode before replacing the line.
4	Check electrical continuity.

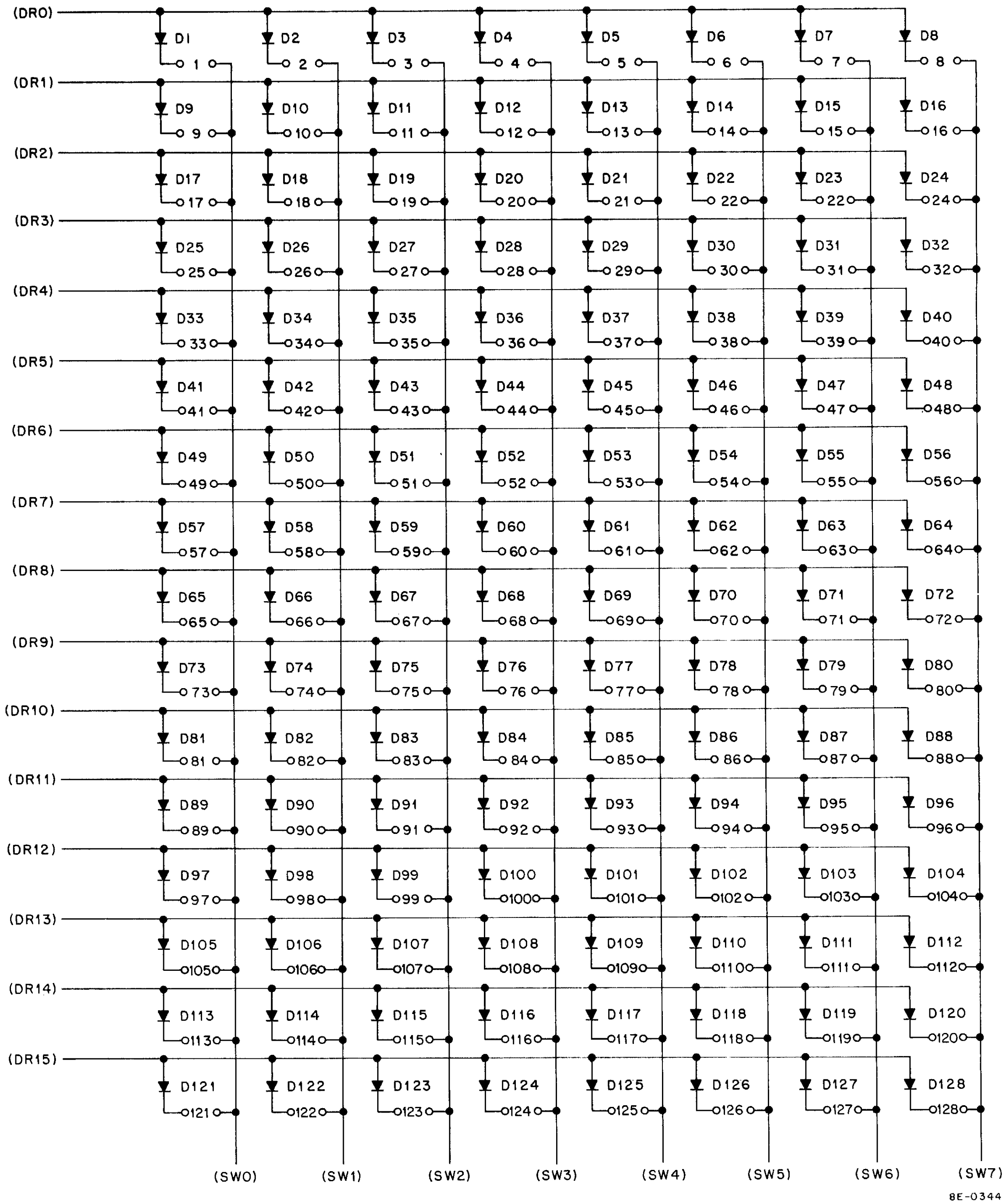


Figure 3-10 Line Select Diode Matrix

Table 3-2
Typical Error Typeout

ADDR	Good	Bad	Driver	Line	Diode	Insert	Term
7010	3635	3637	00	05	05	011110011101 010101010100	SW4

SECTION 6 SPARE PARTS

Table 3-3 lists recommended spare parts for the MR8-E. These parts can be obtained from any local DEC office or from DEC, Maynard, Massachusetts.

Table 3-3
Recommended MR8-E Spare Parts

DEC Part No.	Description	Quantity
15-05321	Transistor DEC 4258	2
15-03100	Transistor DEC 3009B	1
11-60114	Diode D664	10
19-10047	IC DEC 74145	1
19-10046	IC DEC 7442	1
19-09705	IC DEC 8881	1
19-09688	IC DEC 846	1
19-09667	IC DEC 74H74	1
19-09486	IC DEC 384	1
19-09971	IC DEC 6380	2
19-09267	IC DEC 74H11	1
19-09056	IC DEC 7402	1
19-05576	IC DEC 7410	1
19-05575	IC DEC 7400	2

CHAPTER 4
MI8-E HARDWARE BOOTSTRAP LOADER

SECTION 1 INTRODUCTION

4.1 GENERAL DESCRIPTION

The MI8-E Bootstrap Loader option uses a 32-word Read-Only Memory (ROM) with diodes that can be arranged to accommodate any program up to 32 words in length. The Bootstrap Loader option is available in the following configurations:

Option Designation	RIM Program
MI8-E	Unencoded
MI8-EA,B	Paper Tape
MI8-EC	TC08 DECtape
MI8-ED	RK8
MI8-EE	Typeset
MI8-EF	Edu System (low)
MI8-EG	Edu System (high)
MI8-EH	TD8-E DECtape

Each configuration contains a uniquely encoded ROM in the form of a specific Read-In Mode (RIM) program. Without this option, RIM would be toggled into memory by the operator at the programmer's console. The RIM loader instructs the computer to receive and store, in core, data from any of the above peripherals in RIM-coded format.

The MI8-E Bootstrap Loader is contained on one quad-size module, designated M847, that plugs directly into the OMNIBUS. All signals enter and leave the module via the OMNIBUS.

4.2 EQUIPMENT REQUIREMENTS

The following basic equipment is necessary to operate and maintain the MI8-E Bootstrap Loader:

a. PDP-8/E Computer
b. ASR-33 Teletype or Equivalent
c. Low- or High-Speed Paper-Tape Reader
d. Low- or High-Speed Paper-Tape Punch
e. MI8-E Bootstrap Diagnostic
f. Bootstrap Loader Option

4.3 COMPANION DOCUMENTS

The following documents and publications are necessary for the operation, installation, and maintenance of this option:

a. *PDP-8/E & PDP-8/M Small Computer Handbook* – DEC, 1972
b. *PDP-8/E Maintenance Manual* – Volume 1
c. *Introduction to Programming* – DEC, 1972
d. DEC engineering drawing, Bootstrap Loader Option, number M847-0-1
e. *MI8-E Bootstrap Loader Diagnostic Manual,* MAINDEC-8E-D1IA-D-(D)

4.4 SOFTWARE

The MAINDEC-8E-D1IA-D diagnostic is used to troubleshoot and verify the operation of the MI8-E Bootstrap Loader option in all configurations shown in Paragraph 4.1.

The diagnostic is available in a low- and high-core version. The version used to test an MI8-E Module will depend on the memory locations utilized by that particular module. The low-core version occupies and uses memory locations 0200-1777, the high-core version occupies and uses memory locations 4200-5777. Use the version that does not conflict with the memory locations of the bootstrap block for the MI8-E Module under test.

SECTION 2 INSTALLATION

The MI8-E Bootstrap Loader option is installed on-site by DEC Field Service personnel. The customer should **not** attempt to unpack, inspect, install, checkout, or service the equipment.

4.5 INSTALLATION

Perform the following procedures to install the MI8-E option:

Step	Procedure
1	Remove the module from the shipping container.
2	Inspect the module for any apparent damage.
3	Verify that the initial address, field address, and starting address jumpers are correct according to Table 4-1.
4	Verify that the diode matrix is properly cut.
5	Connect the module to a convenient OMNIBUS slot. The module should be located reasonably close to the MM8-E Modules.

4.6 CHECKOUT

Perform the following procedures to checkout the MI8-E option:

Step	Procedure
1	Perform acceptance tests provided in Paragraph 2.3, Volume 1.
2	Load MAINDEC-8E-D1IA-D. This verifies the correct operation of the MI8-E Bootstrap Loader in all of its standard configurations.

(continued on next page)

Step	Procedure
3	If this verification was not performed satisfactorily, refer to Section 5 for troubleshooting procedures.
4	Make proper entry on user's log that the acceptance test for the MI8-E option was performed satisfactorily.

Table 4-1
MI8-E Bootstrap Loader Option Encoding Scheme

Option	MI8-E Unencoded	MI8-EA Paper Tape	MI8-EC TC08 DECtape	MI8-ED RK8	MI8-EE Typeset	MI8-EF Edu Sys Low	MI8-EG Edu Sys High	MI8-EH TD8-E Dectape
INITIAL	0	7737	7554	0023	7756	7737	6007	7300
ADDRESS		6014	7600	6007	7771	6007	6007	1312
Data		0776	6774	6751	6014	7604	7604	4312
		7326	1374	6745	6011	7510	7510	4312
		1337	6766	5025	5360	3343	3343	6773
		2376	6771	7200	7106	6766	6766	5303
		5340	5360	6733	7106	6771	6771	6777
		6011	7240	5031	6012	5344	5344	3726
		5356	1354	7777	7420	1376	1376	2326
		3361	3773	7777	5357	5343	5343	5303
		1361	1354	"	5756	7600	7600	5732
		3371	3772	"	4356	6603	6603	2000
		1345	1375	"	3373	6622	6622	1300
		3357	6766	"	4356	5352	5352	6774
		1345	5376	"	7777	5752	5752	6771
		3367	7754	"	"	7577	7577	5315
		6032	7755	"	"	6032	6014	6776
		6031	0600	"	"	6031	6011	0331
		5357	0220	"	"	5357	5357	1327
		6036	6771	"	"	6036	6016	7640
		7106	5376	"	"	7106	7106	5315
		7006	7777	"	"	7006	7006	2321
		7510	"	"	"	7510	7510	5712
		5374	"	"	"	5357	5374	7354
		7006	"	"	"	7006	7006	7756
		6031	"	"	"	6031	6011	7747
		5367	"	"	"	5367	5367	0077
		6034	"	"	"	6034	6016	7400
		7420	"	"	"	7420	7420	7777
		3776	"	"	"	3776	3776	"
		3376	"	"	"	3376	3376	"
		5356	"	"	"	5356	5357	"
Data	0	0	"	"	"	0220	0220	"
Starting Address		7737	7754	7703	7770	7737	7737	7300

SECTION 3 OPERATING PROCEDURES

The operating procedures for the Bootstrap Loader are as follows:

Step	Procedure
1	If the RUN lamp is on, depress the HLT key and observe that the RUN lamp is off.
2	Depress and raise the SW key.
3	Observe that the RUN lamp is again illuminated.

SECTION 4 PRINCIPLES OF OPERATION

4.7 GENERAL DESCRIPTION

The relation of the Bootstrap Loader and the CPU is similar to that of the operator's console and the CPU. Both the loader and the panel must:

a. Initialize the CPU.

b. Load Address.

c. Load Extended Address.

$\left.\begin{array}{l} \\ \\ \end{array}\right\}$ to define the first address in which to deposit instructions

d. Deposit instructions in sequential locations.

e. Load starting address of the program just deposited.

f. Start the program.

Because the operation of the Bootstrap Loader is closely tied in with the operation of the processor, the reader should have a thorough understanding of the processor control signals. Detailed theory of the processor and memory is given in Chapter 3, Volume 1; the control signal description is provided in Chapter 9 of the *PDP-8/E & PDP-8/M Small Computer Handbook.*

A summary description of the control signals necessary to accomplish the six operations listed above is given in the following:

Operation	Assert	Affect on CPU
Initialize (Bootstrap)	Ground: POWER OK H (Pin BV2)	Causes the CPU's MA Control flip-flop to clear. Also causes the CPU's timing generator to generate.
Initialize (Panel)	INITIALIZE H (Pin CR1)	INITIALIZE H, clearing the AC, Link, all flags, and the Interrupt and Break systems.
Load Initial Address	Ground: LA ENABLE L (pin BM2) Place initial address onto the data lines of the OMNIBUS Pulse: PULSE LA H (Pin DR2)	Loads the address placed on the data lines into the CPMA register.

(continued on next page)

Operation	Assert	Affect On CPU
Load Extended Address	Ground: KEY CONTROL L (Pin DU2) Place extended address bits onto Data 6–8 and Data 9–11 Pulse: PULSE LA H (Pin DR2)	Loads the addresses placed on bits 6–8 and 9–11 of the DATA BUS into the IF and DF of the Memory Extension Control, Type KM8-E, if one is in the machine.
Deposit (used by panel)	Ground: KEY CONTROL L (Pin DU2) MS, IR DISABLE L (Pin CV1) Place bits to be deposited onto DATA BUS. Pulse (low): MEM START L (Pin AJ2)	Causes one memory cycle to occur. Word on DATA BUS is deposited, and PC is incremented. Machine stops at end of memory cycle.
Deposit (used by bootstrap)	Same as Deposit above, except KEY CONTROL L (Pin DU2) goes high during TS3 of computer's cycle.	Same as above, except that the machine continues to run.
Load Starting Address	Same operation as Load Initial Address except for the word placed on the DATA BUS.	
Start Program	Pulse (low): MEM START L (Pin AJ2)	Starts program.

To minimize the logic needed for the MI8-E, an extra Load Extended Address operation takes place just before starting the program. Hence the complete sequence of operations is:

a. Initialize the CPU, Bootstrap, and all system devices.

b. Load Initial Address.

c. Load Extended Address.

d. Deposit 32 words into memory.

e. Load Starting Address of the bootstrap program just deposited.

f. Load Extended Address.

g. Start the bootstrap program.

4.8 MAJOR PORTIONS OF THE MI8-E

A block diagram of the major portions of the MI8-E is shown in Figure 4-1. When power is applied to the computer, critical control flip-flops within the MI8-E are initialized so that the MI8-E is inoperative and the computer can run normally. If the switch labeled SW on the operator's console is moved to the "down" position and then to the "up" position, the MI8-E will operate.

NOTE

SW should be left in the "up" position when the MI8-E is not being used. If SW is left in the "down" position, the MI8-E will operate if the machine is stopped and the console's OFF/POWER/PANEL LOCK switch is moved to the PANEL LOCK position.

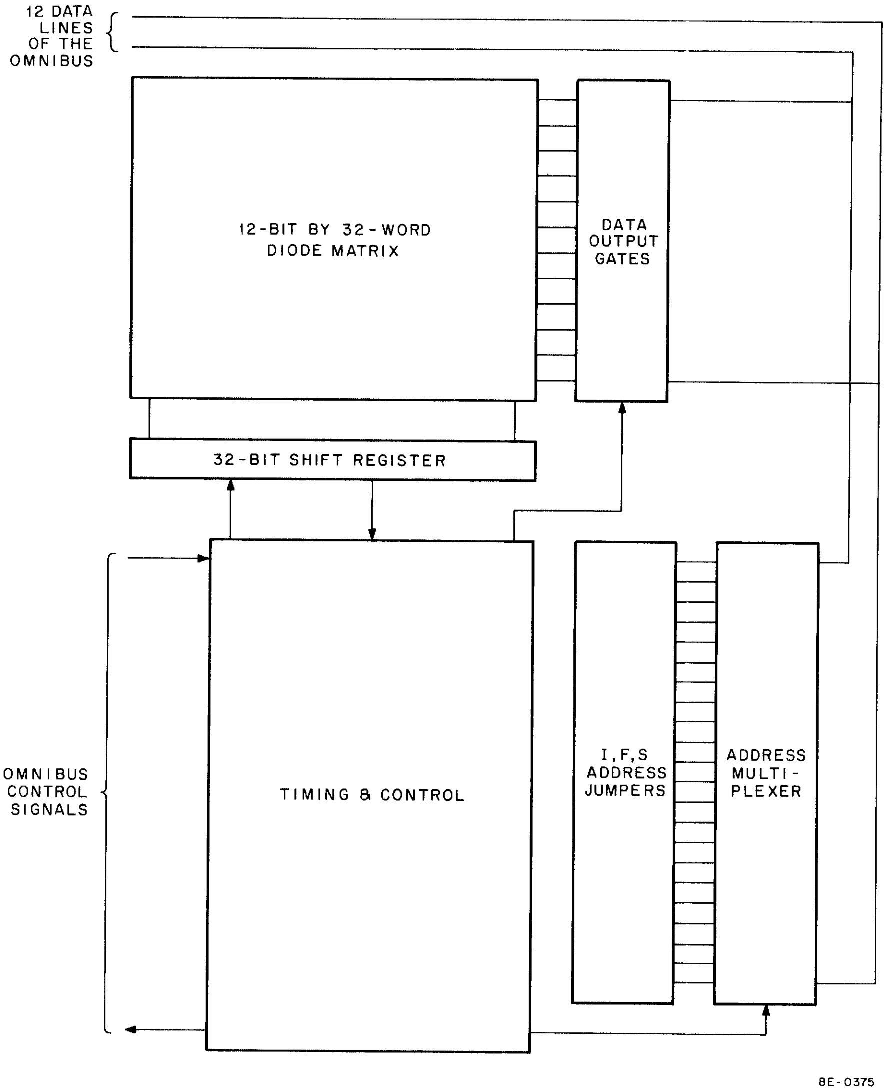


Figure 4-1 MI8-E Block Diagram

4.8.1 Address and Data Information

Address information is stored in three sets of jumpers denoted as follows:

Designation	Function
I0–I11	Initial address. I0 is the most-significant bit. The address encoded in the I jumpers is the first of 32 successive locations into which instructions will be loaded by the MI8-E.
S0–S11	Starting address. S0 is the most-significant bit. This address is the address at which the MI8-E will start the program after the bootstrap program has been stored.
F2, F1, F0	Field bits. F2 is the most-significant bit. These bits are loaded into both the IF and DF of the Memory Extension Control, Type KM8-E.

In all cases, the presence of address jumpers indicates a binary 0; the absence of a jumper indicates a binary 1. Unencoded MI8-E boards are shipped with all jumpers in place.

Data to be deposited in memory is encoded by the presence (binary 0) or absence (binary 1) of diodes in a 12-bit by 32-word matrix. The positions of words and bits within words are clearly indicated in etch on the board; the presence or absence of diodes and jumpers for standard bootstraps is shown on the engineering drawings.

4.8.2 Sequence of Operations

When SW is operated while the computer is on but not running, the MI8-E's timing and control logic is activated. The MI8-E first grounds POWER OK H to initialize the computer. Simultaneously, it clears the 32-bit shift register that drives the diode matrix. The timing and control logic then loads I and F addresses into the CPU and Extension Control, respectively, by enabling the appropriate address multiplexer and then placing signals onto the OMNIBUS. These signals have already been described in Paragraph 4.7. The MI8-E then shifts a 1 into the first bit of the 32-bit long shift register, enables the matrix data output gates, and starts the processor's timing chain. Thus, the contents of the first word in the matrix are placed on the DATA BUS. Control signal MS, IR DISABLE L, applied to the OMNIBUS, causes this word to be written into memory. The KEY CONTROL L signal causes the next sequential address to enter the CPMA, KEY CONTROL L is allowed to go high during TS3 of the processor's cycle so that the processor continues to run. At the end of each memory cycle (TP4) the single 1 is shifted down the shift register, and a 0 is shifted into the first bit of the register. This process continues until all 32 words have been deposited.

When the 1 reaches the last bit of the shift register, control circuitry causes the processor to stop. The timing of the MI8-E is restarted. After loading the "S" and "F" addresses into the CPU and Extension Control, the processor is again started and the MI8-E's job is done.

4.8.3 Bootstrap Timing (Figure 4-2)

A timing diagram illustrating the primary timing and control signals within the bootstrap, as well as the processor timing, is shown in Figure 4-2. The development of each signal is shown in the detailed logic.

4.9 DETAILED LOGIC DESCRIPTION

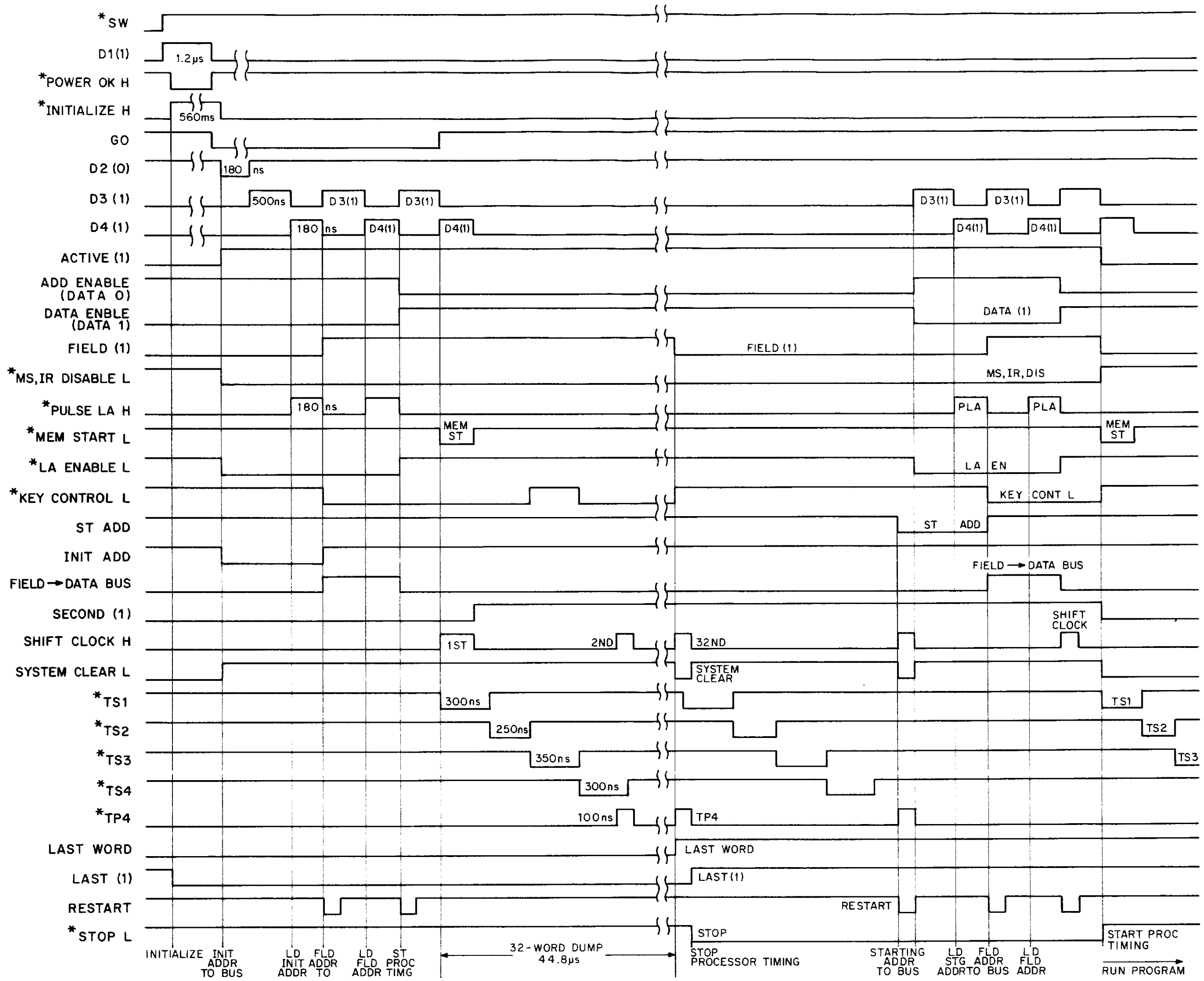
The following paragraphs present portions of the MI8-E logic. All illustrations are interrelated and, therefore, should be considered collectively. The sequential operation of each circuit is presented in the system description.

4.9.1 Bootstrap Timing Logic (Figure 4-3)

A timing chain is created in the timing logic by connecting, in series, four time-delay flip-flops designated D1 through D4. The 74123 IC has a retriggering characteristic that makes an automatic restart capability possible. However, the restart loop is concerned only with D3 and D4 and is controlled by the logical conditions of one flip-flop designated ACTIVE and a second flip-flop designated DATA.

Refer to Figure 4-4 for details about the 74123 IC. The restart operation makes use of the last line in the truth table, which states that the one-shot will initiate a timing cycle if the A and B inputs are enabled and the Master Reset input (C) is brought from low to high.

Again referring to Figure 4-3, the logic to the right of D1 is used to ground POWER OK H and serves to control the GO flip-flop. GO will set when D1 times out only if RUN L is not asserted (the machine is stopped). RUN L also prevents the bootstrap from grounding POWER OK H, which would otherwise stop the processor timing. Therefore, once the bootstrap is activated, GO remains set until processor timing begins. RUN L results and clears the GO flip-flop.



NOTE:
Signals marked * appear on the omnibus. All other signals are internal to the MI8-E.

Figure 4-2 Bootstrap Timing

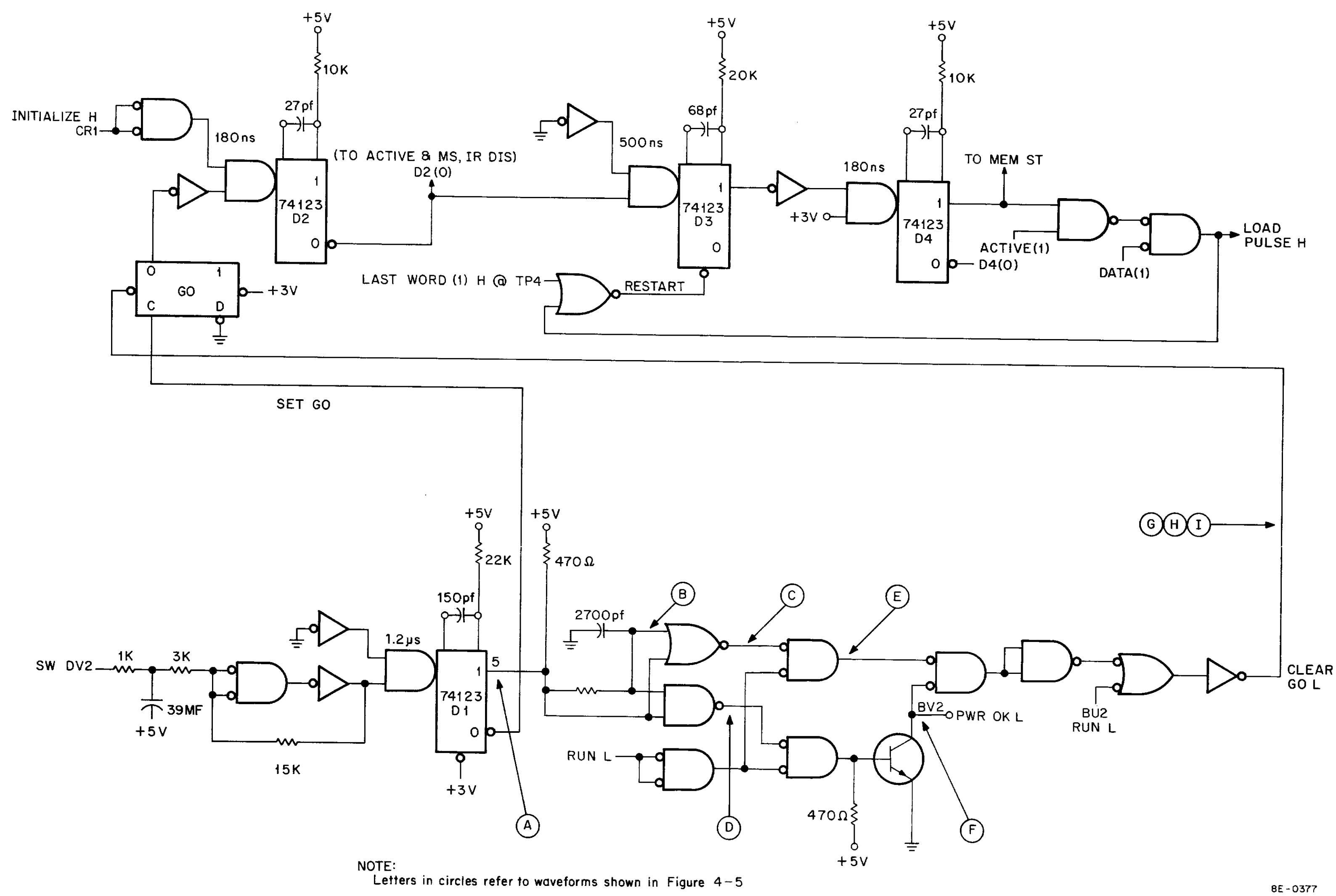
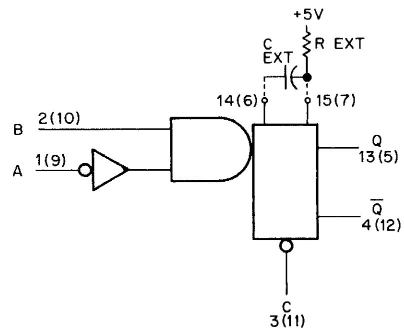


Figure 4-3 Bootstrap Timing Logic



NOTE:
Pin numbers not in parentheses are for delay 1; those within parentheses are for delay 2.

INPUTS			OUTPUTS	
A	B	C	Q	Q̄
H	X	X	L	H
X	L	X	L	H
X	X	L	L	H
L	↑	H		
↓	H	H		
L	H	↑		

↓ = TRANSITION FROM HIGH TO LOW

↑ = TRANSITION FROM LOW TO HIGH

= OUTPUT WAVEFORMS(DURATION DETERMINED BY THE R AND C ATTACHED TO THE ONE-SHOT)

8E-0378

Figure 4-4 74123 Logic Diagram and Truth Table

A positive-going transition on the SW line sets D1 for a period of 1.2 μs. On the input line, the RC network and feedback loop remove any switch contact bounce. On the output of D1 (1), the circuitry asserts POWER OK L (if RUN L is not asserted) and asserts CLEAR GO L when RUN L is asserted. Because CLEAR GO L is connected to the clear side of GO, the net result is:

a. If the computer is running, ignore the output of D1 (0).

b. If the computer is stopped and D1 triggers, set GO.

The conditions upon which CLEAR GO L is generated are shown graphically in Figure 4-5.

Signal POWER OK H goes low shortly after D1 (1) goes high. This causes the M8330 Timing Module to create a 560-ms INITIALIZE H pulse to clear the processor and options. In the meantime, D1 becomes reset because the 1.2-μs delay times out. This causes D1 (0) to go high; this transition clocks GO. GO (0) L half-qualifies the input to D2. At the end of the 560-ms INITIALIZE H pulse, the trailing edge sets D2 for a period of 180 ns. The D2 (0) negative-going output sets the ACTIVE flip-flop and the DEP flip-flop and the trailing edge of D2 (0) sets D3 for a period of 500 ns. At the trailing edge of D3 (1), the negative-going transition sets D4. A restart path beginning with D4 (1) H and gated by ACTIVE (1) H and DATA (1) L generates LOAD PULSE H. Signal LOAD PULSE H (trailing edge) clocks the FIELD flip-flop on the first pass and clocks the DATA flip-flop on the second pass. The trailing edge of each LOAD PULSE H causes D3 to start another timing cycle. Once the DATA flip-flop is clocked, DATA (1) H inhibits LOAD PULSE H and hence another restart operation until LAST is set (following the 32-word dump operation). The last D4 (1) H with DATA (1) H asserts MEM START L. MEM START L

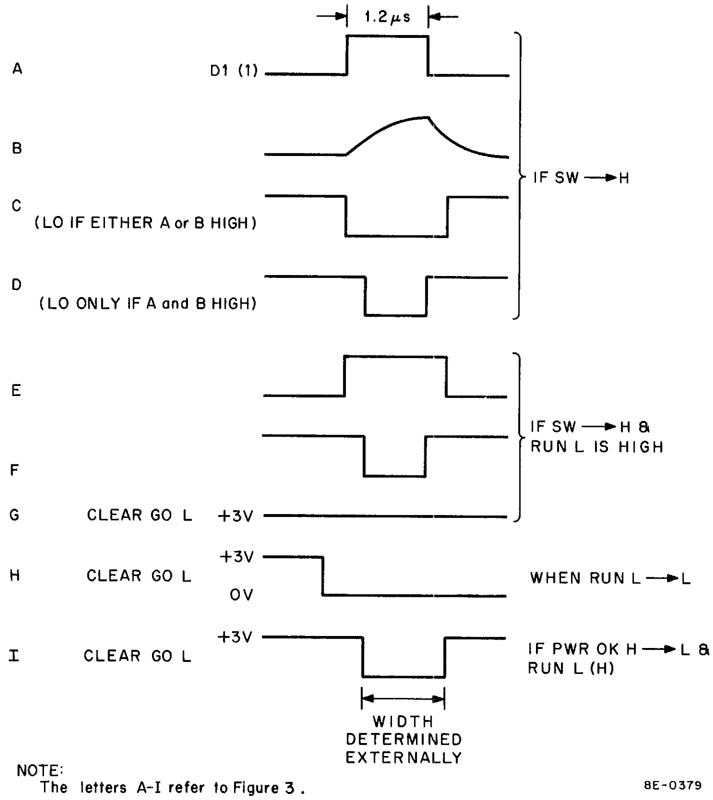


Figure 4-5 CLEAR GO Signal Waveform Analysis

asserts RUN L in the timing module and RUN L asserts CLEAR GO L, which then resets the GO flip-flop. The timing chain remains inactive until LAST WORD H sets the LAST flip-flop and with TP4 creates RESTART.

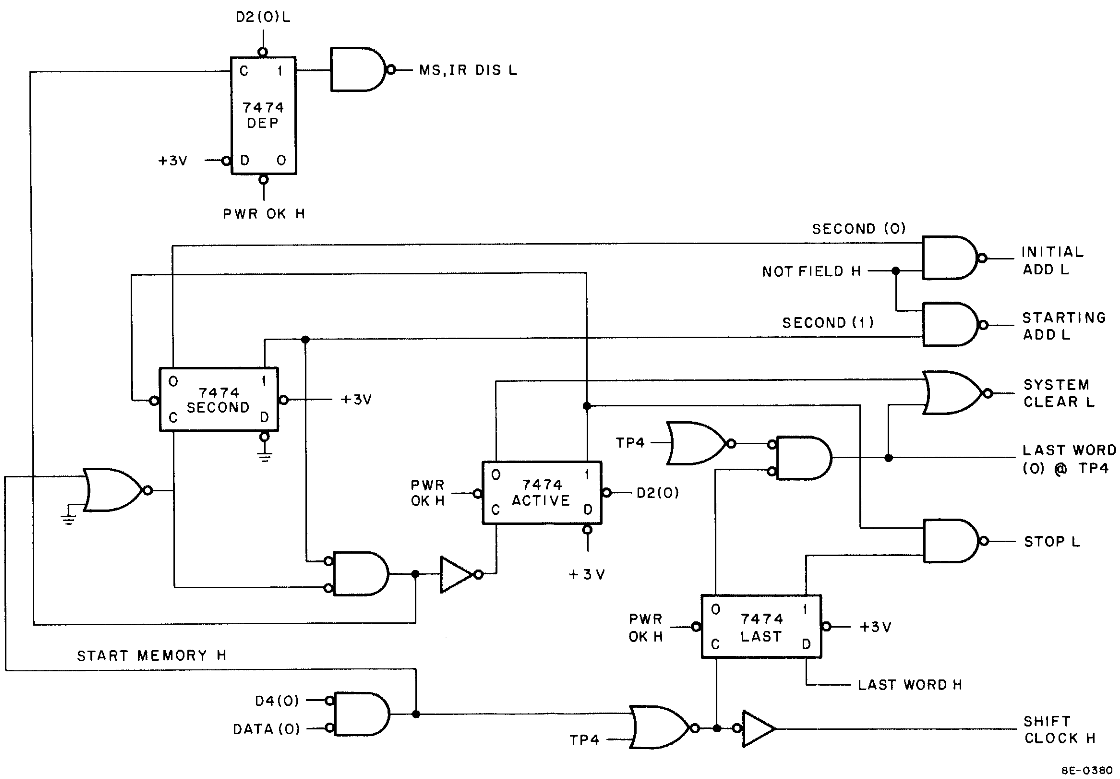
4.9.2 Bootstrap Control Logic (Figure 4-6a and b)

Four flip-flops are shown in Figure 4-6a. LAST, ACTIVE, and DEP are cleared by POWER OK H being low. ACTIVE (1) low clears SECOND. ACTIVE (0) now high generates SYSTEM CLEAR L. SYSTEM CLEAR L clears the FIELD flip-flop, the DATA flip-flop, and the 32-bit shift register.

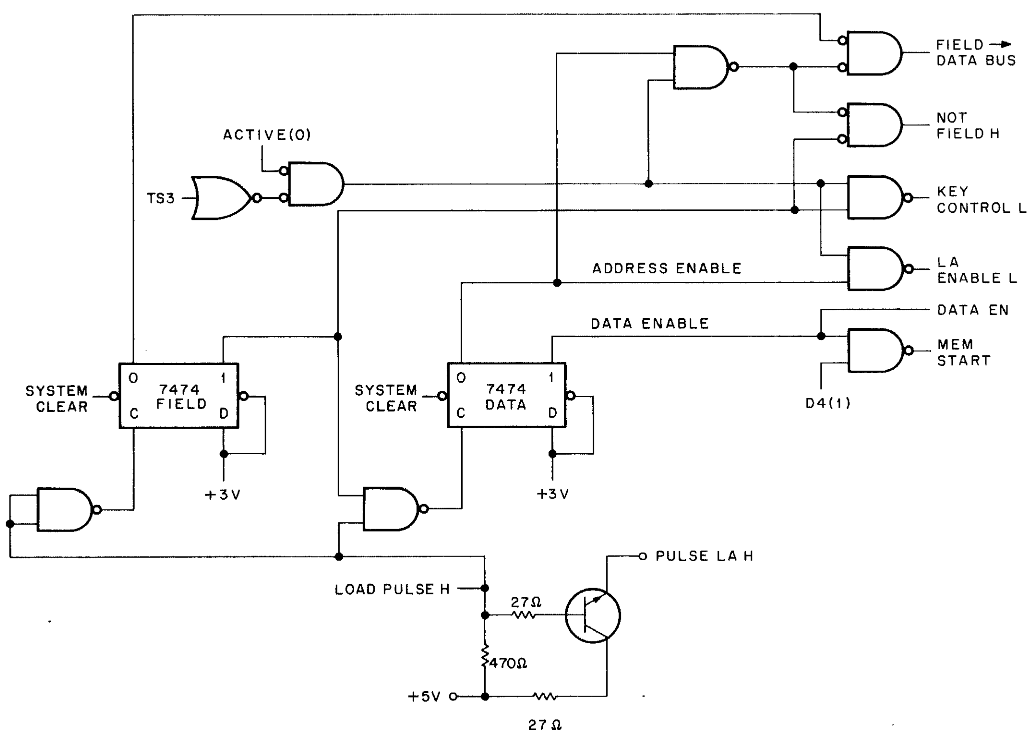
When the timing chain reaches D2, D2 (0) sets flip-flops DEP and ACTIVE. ACTIVE (1) H now half-qualifies STOP L and DEP (1) H generates MS, IR DISABLE L. FIELD (1) L and ACTIVE (0) L generate NOT FIELD H. This signal, combined with SECOND (0) H, asserts INITIAL ADD L. Signal ACTIVE (0) L, combined with DATA (0) H, asserts LA ENABLE L.

When the timing chain reaches D4 and generates LOAD PULSE H, PULSE LA H is asserted. This loads the initial address into the CPMA. When D4 times out (at the trailing edge of LOAD PULSE H), FIELD is clocked. FIELD (1) H half-qualifies the DATA flip-flop clock input and FIELD (0) L qualifies FIELD TO DATA BUS.

With FIELD set, ACTIVE set and DATA reset, FIELD TO DATA BUS, KEY CONTROL L and LA ENABLE L cause the memory field to be addressed. The trailing edge of LOAD PULSE H also asserts RESTART and the timing chain is again activated.



a.



b.

Figure 4-6 Bootstrap Control Logic

The next LOAD PULSE H at the trailing edge clocks the DATA flip-flop and again asserts RESTART. DATA (1) H now half-qualifies MEM START. When D4 is again set, D4 (1) asserts MEM START L. D4 (0) L, combined with DATA (0) L generates START MEMORY H and SHIFT CLOCK H. SECOND is clocked by the trailing edge of START MEMORY H in preparation for STARTING ADDRESS. Signal MEM START L starts the processor timing and SHIFT CLOCK H activates the 32-bit shift register. The 32-word dump operation now begins. Signal LAST WORD H is asserted when the 32nd word is reached. LAST is clocked at TP4 and LAST (1) qualifies STOP L. LAST (0) at TP4 asserts SYSTEM CLEAR L and restarts the timing chain. SYSTEM CLEAR L again clears flip-flops DATA and FIELD.

With SECOND set, the next address must be the starting address. When the starting address is loaded into the CPMA, the timing chain goes through two additional restarts for the field address and Memory Start. When MEM START L is again qualified, START MEMORY H is asserted. The trailing edge of START MEMORY H this time clocks the ACTIVE flip-flop and ACTIVE (0) asserts SYSTEM CLEAR L. This prevents a second 32-word dump and removes all bootstrap signals from the OMNIBUS.

4.9.3 Initial/Starting Address Jumper Network and Output Logic (Figure 4-7)

The initial and starting addresses, determined by jumpers I for initial address and S for starting address, provide a 12-bit word for each type of address. Removing a jumper causes a 1 of the corresponding bit to be placed onto the DATA BUS. Otherwise a 0 will be submitted. The 7235 IC multiplexer selects either the initial address or the starting address, depending upon which control line is asserted low.

4.9.4 Extended Memory Field Output Logic (Figure 4-8)

The extended memory field output logic is jumper-selectable to provide either a 1 or 0 on each corresponding bit. Signal FIELD → DATA BUS H, developed in the bootstrap control logic, is used to apply 1s and 0s to the DATA BUS.

4.9.5 12 x 32 Diode Matrix and Control Logic (Figure 4-9)

A partial illustration of the 12 x 32 diode matrix and control logic is shown in Figure 4-9. The diode matrix is arranged in 32 columns and 12 rows to accommodate 32 12-bit words. Each word is applied to the output logic gates by 8-bit parallel-out serial shift registers with the first word being applied by SHIFT CLOCK H during TS1. Four 74164 ICs, each providing 8 outputs, are used to sequentially bias the diodes corresponding to the selected row. Each time SHIFT CLOCK H is received, the 74164 IC shifts to the next sequential output line. Carry to the next IC is accomplished by the last IC output line, except for the output line labeled "31". When 31 has been reached, LAST WORD H is generated and applied to the LAST flip-flop. Signal SYSTEM CLEAR L is generated in the control logic to clear the shift registers.

SECTION 5 MAINTENANCE

The general procedures concerning preventive and corrective maintenance are given in Chapter 4, Volume 1. When corrective maintenance is required, the technician should use the maintenance program given in Section 2 of this chapter to determine the nature of the problem.

4.10 TROUBLESHOOTING

The option schematic, drawing number E-CS-M847-0-1, must be referred to for IC locations and pin numbers. Test points are provided on the module to facilitate troubleshooting.

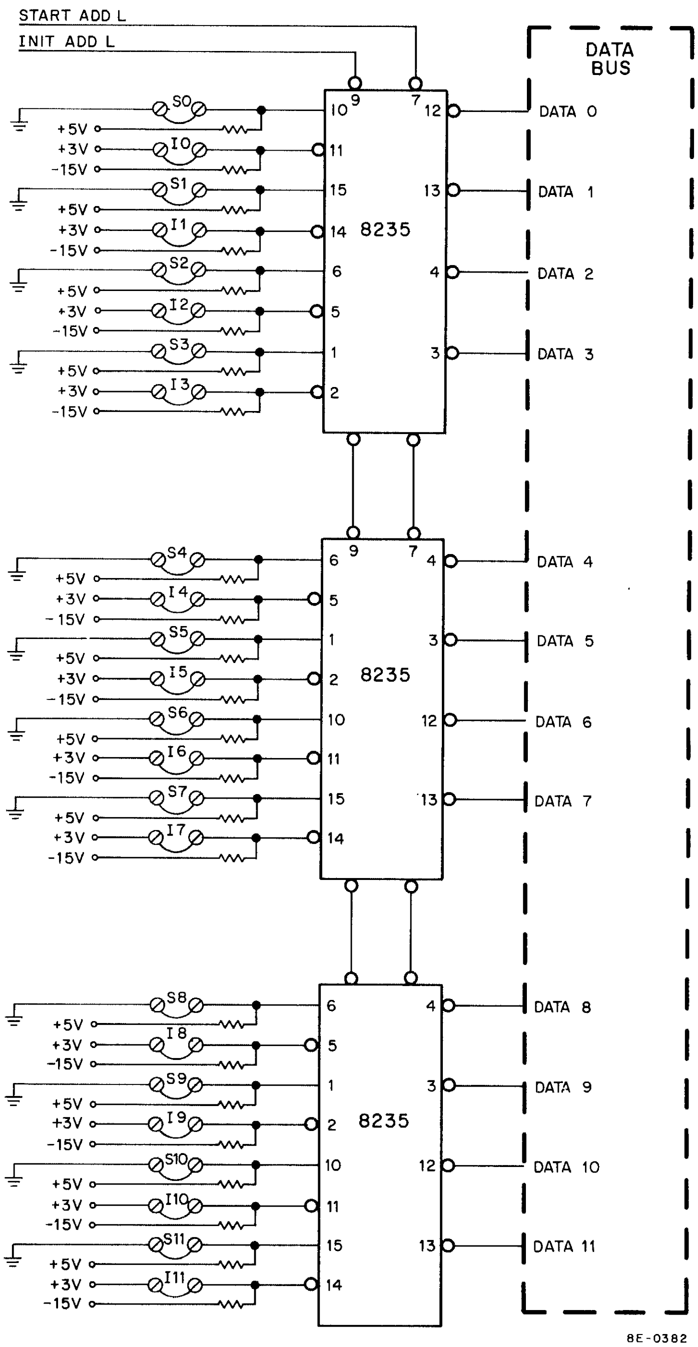


Figure 4-7 Initial/Starting Address Jumper Network

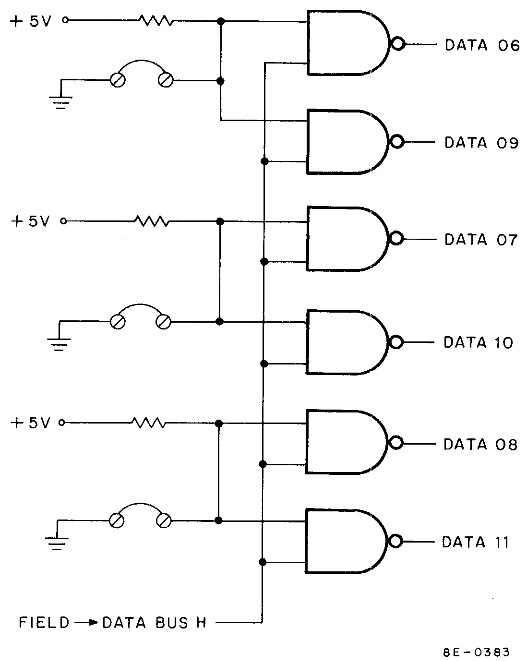


Figure 4-8 Extended Memory Field Output Logic

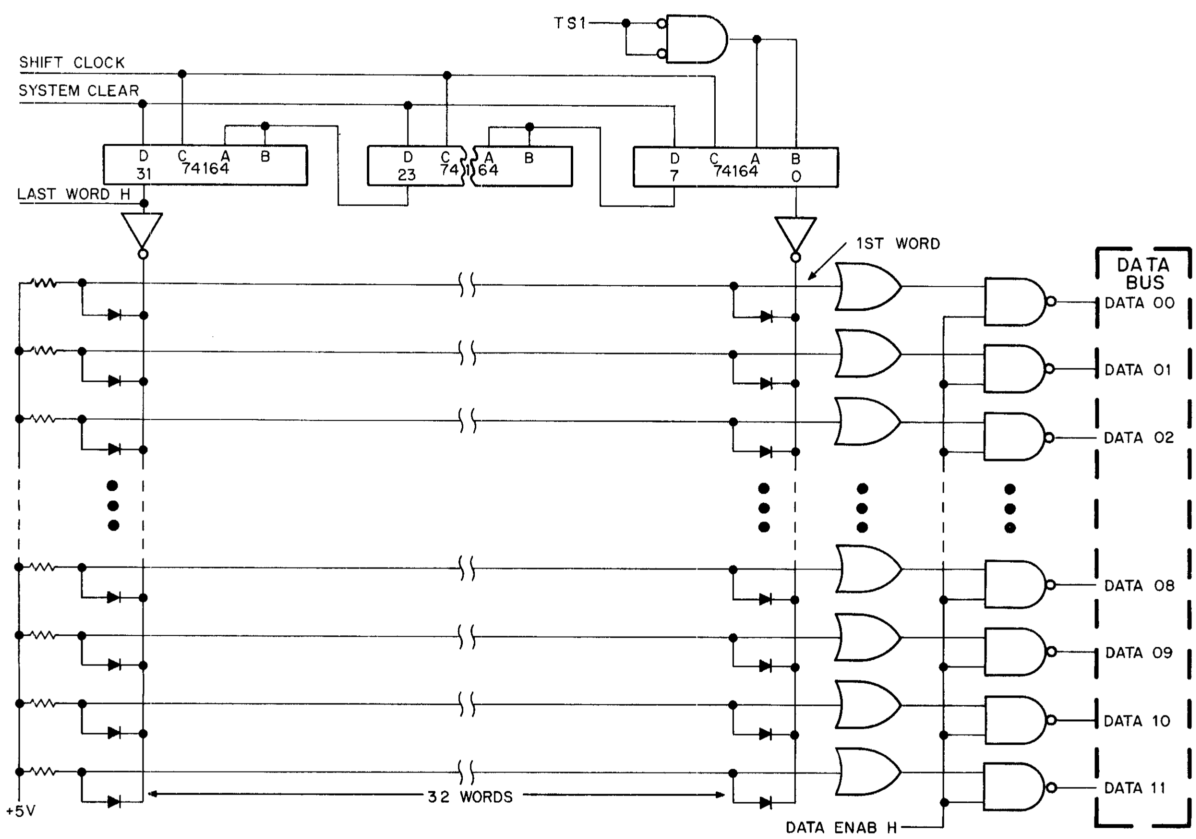


Figure 4-9 12 X 32 Diode Matrix

4.10.1 MI8-E Bootstrap Diagnostic Program

The operation of the MI8-E Bootstrap Loader should first be verified by the MI8-E Bootstrap Diagnostic Program, MAINDEC-8E-D11A-D, with corresponding MAINDEC operating procedures.

4.10.2 Direct Memory Access Control Signal Verification

Because the bootstrap loader uses direct memory access control signals in a manner similar to the operator's console (front panel), addressing memory and depositing information must be accomplished in the same manner. Proper operation of signals such as PULSE LA H, LA ENABLE L, KEY CONTROL L, MEM START L, and STOP L can be verified by performing the following procedure at the programmer's console:

Step	Procedure
1	Load Address 7777.
2	Load Address 0000.
3	Load Extended Address 7.
4	Load Extended Address 0.
5	Deposit bit 11, then 10 . . . 0 individually.
6	MA should equal 0014.
7	Load Address 0000.
8	Exam. You should see bit 11, then 10 . . . 0 until the MA = 0014.
9	Turn Rotary Switch to the state position.
10	Depress and hold LOAD ADDRESS. No major state should be lit (F,D,E).
11	Put SW up. SW indicator should be off.
12	Put SW down. SW indicator should be on.
13	Deposit in location 0/7240 1/7402.
14	Load and start location 0.
15	Turn Rotary Switch to AC position. AC should = 7777.
16	Depress CLEAR. AC should = 0000.
17	Load location 0. Deposit 5000.
18	Load and start location 0. Run light should be on.
19	Depress SINGLE STEP. Run light should be out.

If the above procedures were completed successfully, the problem is in the MI8-E Bootstrap Loader hardware. If the procedures indicate a malfunction, the MI8-E should be removed and the procedure tried again; the problem is in the programmer's console, the CPU or the Memory Extension Control.

4.11 BOOTSTRAP HARDWARE TROUBLESHOOTING ANALYSIS

A very convenient troubleshooting aid is the Programmer's Console Status Display. The mere presence or absence of any one of the addresses used in the Bootstrap Loader option can localize the problem by the process of elimination. If the problem is that the option fails to continue after the initial address is loaded and the RUN lamp does not come on, the probable cause is the FIELD flip-flop or associated circuitry. However, if the field address is loaded but the RUN lamp does not come on, the probable cause is the ENABLE flip-flop or associated circuitry. It can be assumed that all of the logic leading up to the logic that controls MEM START functions properly.

If the RUN lamp comes on but the 32-word dump operation does not begin, it can be assumed that the DATA flip-flop functions and the problem is somewhere in the 12 x 32 diode matrix logic. No dump might indicate that the first shift register is malfunctioning.

If the RUN lamp comes on and the dump operation takes place but then fails to stop, the problem is in the Shift Register or LAST flip-flop. The problem also causes the most-significant MA lights to run at half intensity. Examination of memory shows that all locations have been cleared. The reason for the failure is that the single 1 which should shift down the 32-bit shift register has been lost, probably because of a malfunctioning 74164 IC.

A malfunction will also occur if a 74164 IC picks up a bit. In this situation, fewer than 32 words will be deposited. Some of these words will be the OR of two or more of the words encoded in the diode matrix.

When an incorrect word is deposited, the faulty diode can be located by determining which word and bit fails. This is easily accomplished by addressing the bootstrap initial address and depressing the EXAM key. When the rotary switch is in the MD position, each of the 32-bit words can be examined.

If bootstrap timing operates properly, the SINGLE STEP and CONTINUE keys may be used to single-step the bootstrap. This technique may be used to find out if data is being deposited correctly.

SECTION 6 SPARE PARTS

Table 4-2 lists the recommended spare parts for the MI8-E. These parts can be obtained from any local DEC office or from DEC, Maynard, Massachusetts.

Table 4-2
Recommended MI8-E (M847) Spare Parts

DEC Part No.	Description	Quantity
19-10436	IC DEC 74123	1
19-09004	IC DEC 7402	1
19-05547	IC DEC 7474	1
19-09935	IC DEC 8235	1
19-10041	IC DEC 74164	1
19-05575	IC DEC 7400	1
19-09705	IC DEC 8881	2
19-09686	IC DEC 7404	2
19-09485	IC DEC 380	1
19-09486	IC DEC 384	1
15-03100	Transistor DEC 3009B	1
11-00114	Diode D664	10
13-01423	Resistor 6.8K, 1/4W, 5%	2
10-00006	Cap., 0.01 μF, 100V, 20%	2

CHAPTER 5
MP8-E MEMORY PARITY

SECTION 1 INTRODUCTION

5.1 MP8-E DESCRIPTION

The MP8-E Memory Parity option adds the circuits required to generate, store, and check the parity of memory words. The MP8-E consists of three quad boards, which are inserted into the OMNIBUS. An H220 Memory Stack is used to store parity for all words in memory. A G227 Memory X-Y Driver is used to provide word select currents. A special G105 Sense-Inhibit Module, which also contains IOT decoding, is used.

This option expands the memory system from 12 to 13 bits per word. When a word is written into memory, its parity is computed, and odd parity (odd number of binary ones in the 13-bit word) is generated and stored in the 13th, or parity, bit. When a word is retrieved from memory, the parity of the 13-bit word is checked. If even parity is detected, a memory error has occurred.

Much of the MP8-E is identical to the MM8-E 4K Core Memory described in Chapter 3, Section 4, Volume 1. The MP8-E is discussed in Chapter 7 of the *PDP-8/E & PDP-8/M Small Computer Handbook.* The reader should have a thorough understanding of this material before proceeding.

One major difference exists between the MP8-E and its description in the *PDP-8/E & PDP-8/M Small Computer Handbook.* The MP8-E is disabled whenever it encounters Read-Only memory. Thus, the initialization process described in the handbook is unnecessary. The CEP instruction (used for diagnostic purposes) will work, as described, as long as the next EXECUTE cycle deals with a location that does not ground the ROM ADDRESS L line on the OMNIBUS. (This EXECUTE cycle can even be the result of an indirect reference to a nonexistent memory field, if desired.)

SECTION 2 INSTALLATION

The MP8-E will be installed on site by DEC Field Service personnel. The customer should **not** attempt to unpack, inspect, install, test, or service the equipment.

5.2 INSTALLATION

Use the following procedure to install the MP8-E:

Step	Procedure
1	Ensure power is off.
2	Install the MP8-E on the position indicated in Table 2-3 of Volume 1.
3	Install the four H851 Edge Connectors between the three modules.

5.3 ACCEPTANCE TEST

To check the MP8-E, run the MP8-E diagnostic program, MAINDEC-8E-D1DA. The program should be run 15 minutes for each 4K MM8-E in the machine; e.g., if the machine has 16K of memory, run the diagnostic for 1 hour. Refer to the program writeup for instructions on how to run the program.

SECTION 3 PRINCIPLES OF OPERATION

Parity in the PDP-8/E is handled in a somewhat unconventional manner. The conventional method of handling parity is to use a special 13-bit memory stack and sense-inhibit system in each of the memory fields. In the PDP-8/E, however, the standard memory, Type MM8-E, is used for data and instruction storage, regardless of whether the system contains the parity option. When parity is desired, an additional memory is added. This special memory handles the parity bit for all possible memory fields. The advantage of this method of handling parity is that no special memory stack is required.

Figure 5-1 is a block diagram of the MP8-E. The lower half of this figure is identical to the lower half of Figure 3-33 in Volume 1; only the portion above the broken line is different. Eight of the twelve memory bits are used, corresponding to the eight possible fields. The FIELD signal into the G227 X-Y Driver is permanently enabled. Hence, during each memory cycle all eight parity bits are read and rewritten into the parity core memory.

Field select gating decodes the extended address bits (EMA 0, 1, and 2) and MD DIR L. The three EMA lines determine which of the eight parity bits are to be examined and possibly modified. All other bits are automatically rewritten from the local parity sense register. If MD DIR L is low, the selected bit is rewritten from the Sense Register. If MD DIR L is high, however, the parity of the twelve MD lines is written into the parity memory.

The selected bit read from the parity core memory is combined with the bits on the twelve MD lines. If parity is erroneous, the ERROR flip-flop is set. The ERROR flip-flop can be interrogated by a SKIP instruction, and can cause an interrupt. IOTs permit clearing of the Error flag, enabling the disabling of parity interrupt, and intentional reading of even parity for diagnostic purposes.

SECTION 4 DETAILED LOGIC

The G227 X-Y Driver and H220 Memory are discussed thoroughly in Volume 1. No attempt will be made in this chapter to duplicate that discussion. In addition, certain portions of the G105 are identical to corresponding parts of the G104 described in Volume 1. These portions are:

a.	Strobe Control Logic	(Vol. 1, Paragraph 3.27.8)
b.	Sense Amplifiers	(Vol. 1, Paragraph 3.27.9)
c.	Sense Flip-flops	(Vol. 1, Paragraph 3.27.10)
d.	Inhibit Drivers	(Vol. 1, Paragraph 3.27.12)
e.	Current Control Circuit	(Vol. 1, Paragraph 3.27.13)
f.	–6V Supply and Slice Control	(Vol. 1, Paragraph 3.27.14)

These parts will not be described here in detail.

5.4 FIELD SELECT GATING

The field select gating (Figure 5-2) connects the output of a SENSE flip-flop to the input of the corresponding Inhibit Driver. The output of gate A must be low in order to rewrite a 1 into the parity memory.

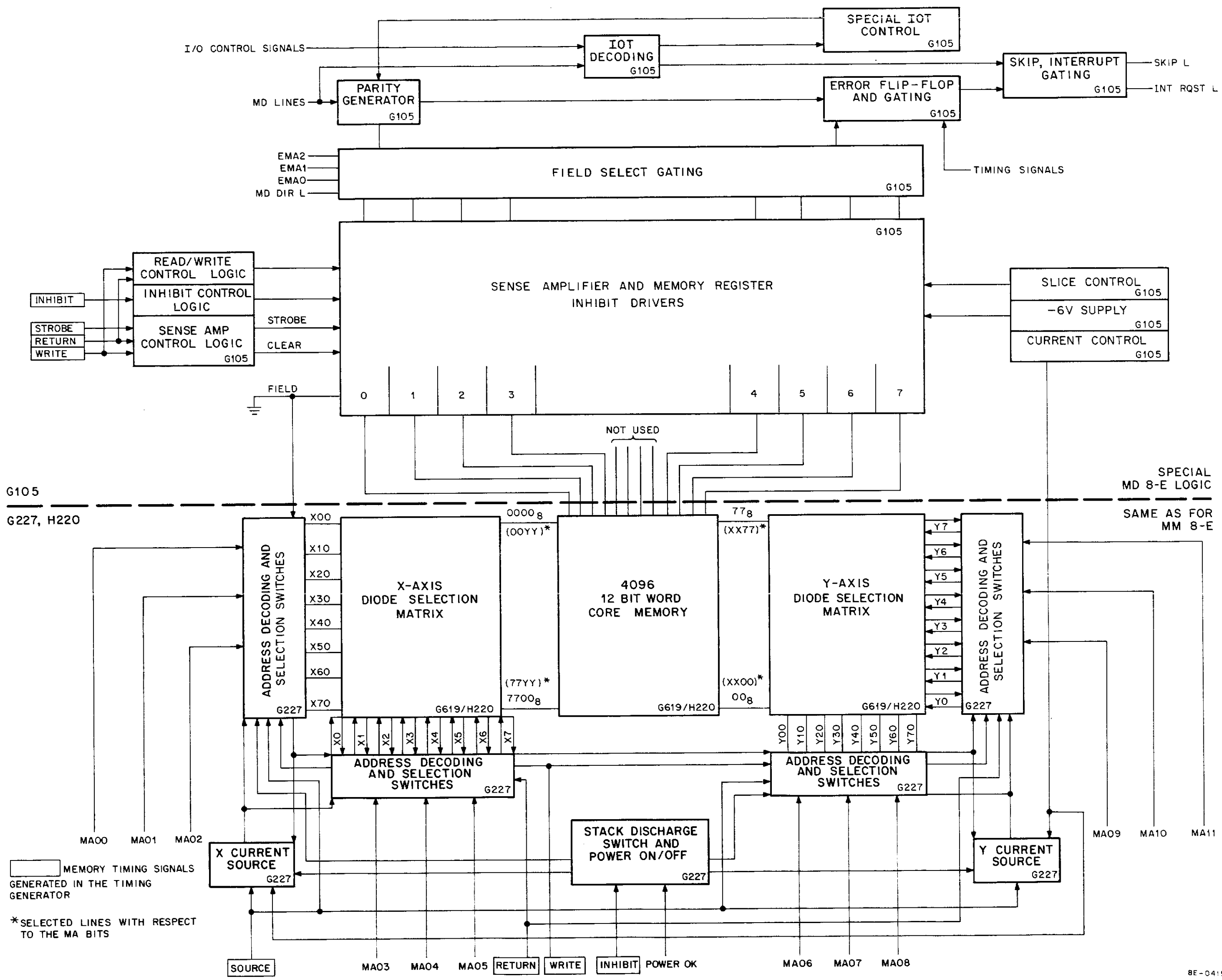


Figure 5-1 MP8-E Block Diagram

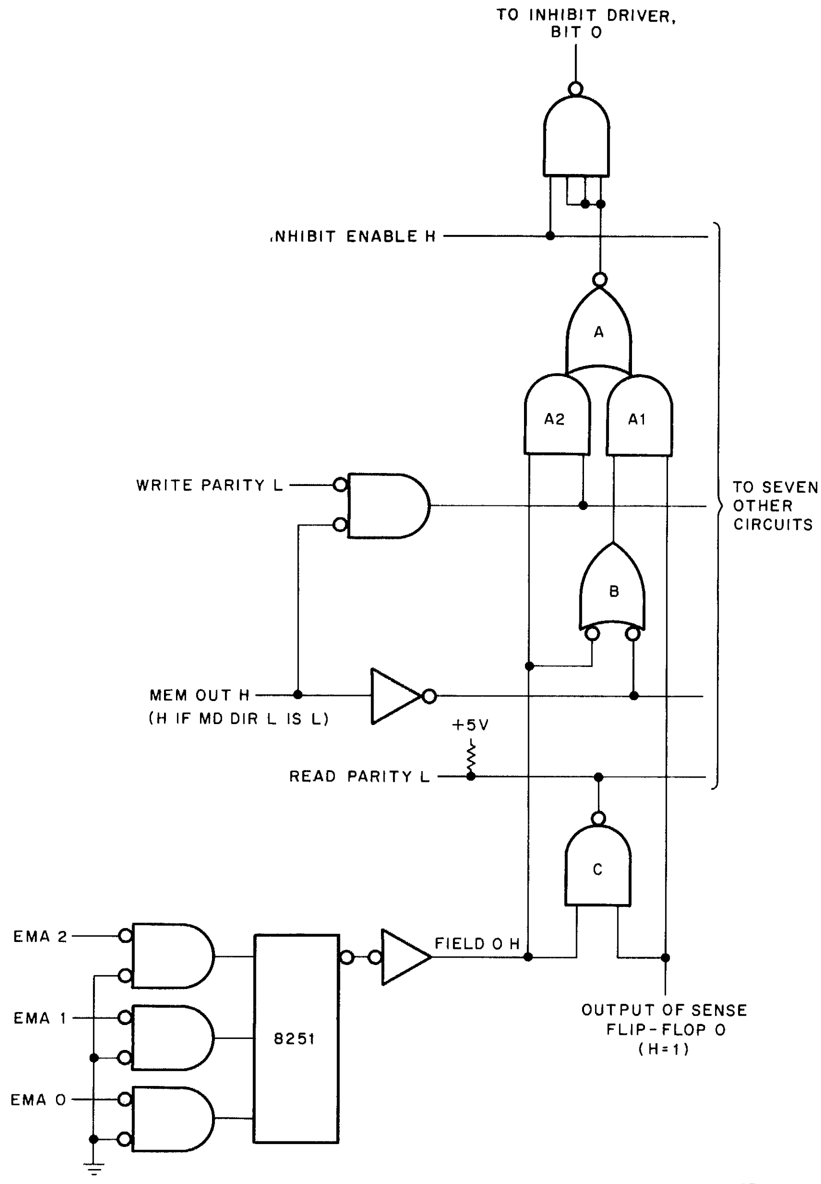


Figure 5-2 Field Select Gating

The EMA bits are buffered from the OMNIBUS and applied to the input of an 8251 Binary-to-Octal Decoder. One of the output lines of the 8251 is low for any combination of EMA bits. All outputs of the 8251 are inverted and applied to portions of the field select gating logic.

If the FIELD 0 H line in Figure 5-2 is low (field 0 is not selected), gates A2 and C are disabled. The output of gate B is high, enabling gate A1. Thus, the output of SENSE flip-flop 0 is applied, via gate A, to the input of the Inhibit Driver, writing the previously read bit back into memory.

If the FIELD 0 H line is high, Field 0 has been selected. The output of SENSE flip-flop 0 is gated onto the READ PARITY L bus via gate C. The source of information for the Inhibit Driver is now a function of MD DIR L. If MD DIR L is low during the latter half of the memory cycle (as it is for all fetches and non-autoindexed defers), gate B is enabled and the SENSE flip-flop provides the rewrite data. If MD DIR L is high, gate A1 is disabled, gate A2 is enabled, and WRITE PARITY L provides the data for the Inhibit Driver. A low on WRITE PARITY L writes a 1 in the parity bit.

5.5 PARITY GENERATOR

The heart of the Parity Generator is the 74180 8-bit parity generator IC which is shown in Figure 5-3. Two of these ICs are cascaded to form the 12-bit gated parity generator shown in Figure 5-4. Line A is high if a 1 has been read as the parity bit from memory. As will be discussed in Paragraph 5.6, the ODD PARITY H line is sampled part way through the memory cycle, before MD DIR L can go high. During the latter half of the memory cycle, if MD DIR L goes high, odd parity is sent to the selected Inhibit Driver via the WRITE PARITY L line. (The state of WRITE PARITY L is ignored by the field select gating if MD DIR L is low.) EVEN PARITY CONTROL H is normally low, as will be described in Paragraph 5.8.

5.6 ERROR FLIP-FLOP AND GATING

Before discussing this logic in detail, a review of MD DIR L and its ramifications in the machine cycle is necessary. MD DIR L is a signal that indicates the source of information on the MD lines. At the beginning of every memory cycle MD DIR L is low, causing the contents of the currently active memory's Sense Register to be placed on the MD lines. At TP2, MD DIR L may or may not change. It will not change during an F or non-autoindexed D cycle. It will change on all other cycles unless grounded by an option other than the CPU (for example, during a BREAK cycle with data direction from memory to peripheral). In any event, if MD DIR L remains low after TP2, no modification of memory is possible and the PDP-8/E may well be performing a "fast" 1.2-μs cycle. If MD DIR L goes high after TP2, the PDP-8/E is definitely performing a "slow" 1.4-μs cycle and is most likely modifying memory.

In a fast cycle, the MD lines have just about changed state by the leading edge of TP2, and there is insufficient time for the Parity Generator to settle. During such cycles, the parity decision must be deferred to the trailing edge of TP2 in order to gain 100 ns more time for the Parity Generator to settle. Since MD DIR L cannot change, there is no danger that the information on the MD lines will change.

In a slow cycle, the MD lines have set up 200 ns before TP2. The parity decision must be made at the leading edge of TP2, since MD DIR L may well change and place new information (the contents of CPU's MB Register) on the MD lines.

The ERROR flip-flop and its gating are shown in Figure 5-5. Assume that the ERROR flip-flop is cleared, and that the odd parity is received from the parity generator logic. At the leading edge of TP2, the SLOW flip-flop is clocked. At the trailing edge of TP2, the FAST flip-flop is clocked. Since no parity error has occurred, both flip-flops remain cleared.

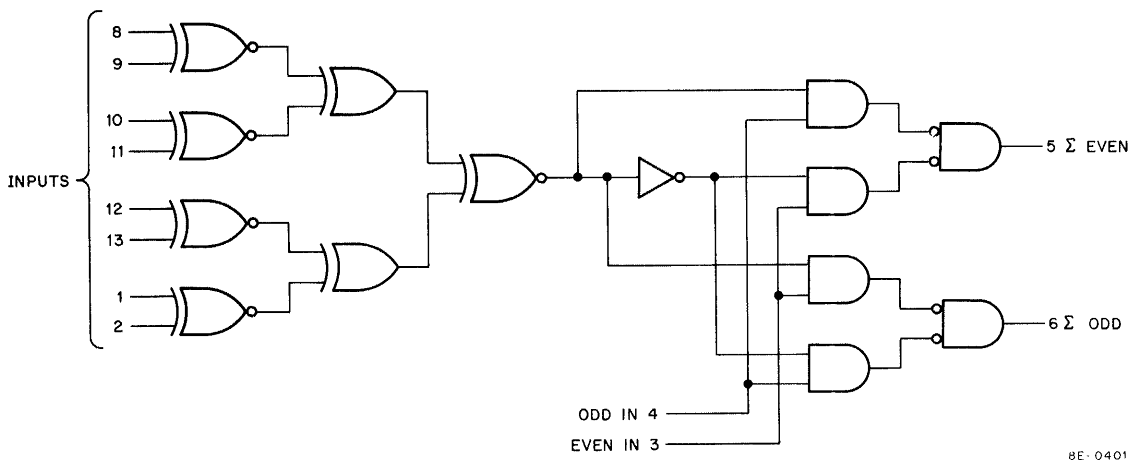


Figure 5-3 74180 8-Bit Parity IC

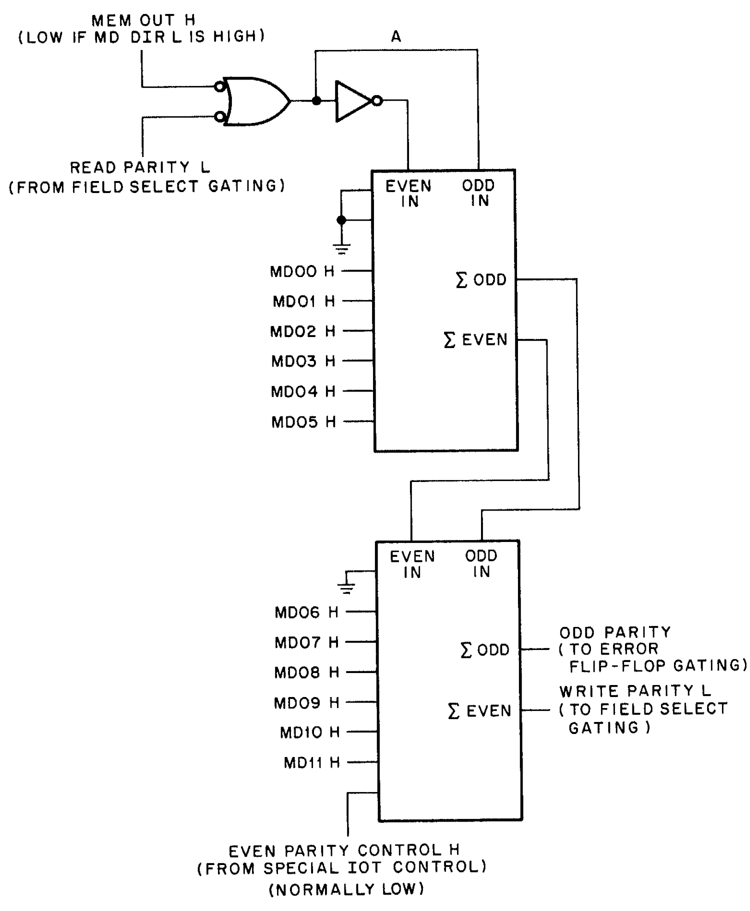


Figure 5-4 Parity Generator

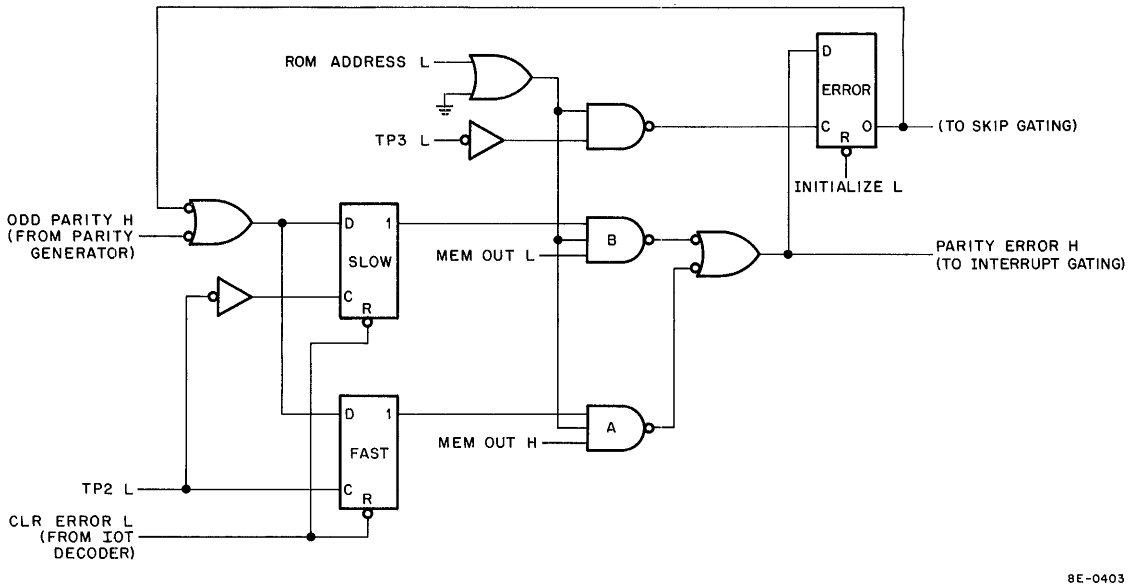


Figure 5-5 ERROR Flip-Flop and Gating

Now assume that the ODD PARITY H line becomes low during a fast cycle. MEM OUT L is low and MEM OUT H is high, since these lines are controlled by MD DIR L. The ODD PARITY H line, therefore, is low at the trailing edge of TP2, when the FAST flip-flop is clocked. Gate A is enabled, presenting a high signal to the D input of the ERROR flip-flop, and simultaneously requesting an interrupt. The SLOW flip-flop may have been set at the leading edge of TP2, but gate B is disabled because MEM OUT L is low.

Last, assume that the ODD PARITY H line becomes low during a slow cycle. At the leading edge of TP2, the SLOW flip-flop is clocked and set. MD DIR L goes high, the result of a flip-flop in the CPU's timing generator being clocked by TP2. Therefore, MEM OUT H goes low, disabling gate A; MEM OUT L goes high, enabling gate B. The MD lines change, the Parity Generator starts to respond, and the FAST flip-flop is clocked at the trailing edge of TP2. The state of the FAST flip-flop is ignored because gate A is now disabled.

Regardless of the type of cycle, the D input of the ERROR flip-flop is high well before TP3, allowing an interrupt request to be generated during the current memory cycle (if the interrupt system is enabled). At the trailing edge of TP3, the ERROR flip-flop is clocked, setting the ERROR flip-flop. During all succeeding memory cycles, the D inputs to both the FAST and SLOW flip-flops are high, maintaining the error condition. The three flip-flops remain set until CLR ERROR L is generated in the IOT decoder logic (discussed in Paragraph 5.7). CLR ERROR L clears FAST and SLOW at the leading edge of TP3. At the trailing edge of TP3, ERROR (its D input is now low) is clocked and thus cleared.

If ROM ADDRESS L is low, gates A and B are both disabled and PARITY ERROR H, therefore, is low. No interrupt request is made. Also, the clock input to the ERROR flip-flop is disabled so that any previously detected error will not be lost.

5.7 IOT DECODING AND SKIP, INTERRUPT GATING

A summary of the IOTs for the MP8-E is given in Table 5-1. The logic is shown in Figure 5-6. The device code is decoded by a 314 gate in a manner similar to any OMNIBUS decoding scheme. As in any internal device, the device must ground INTERNAL I/O L when it sees its device code to inhibit operation of the KA8-E. Decoding of the IOT operation is done by an 8251 Binary-to-Octal Decoder.

The EPI L and DPI L signals are used to set and clear, respectively, the IRE (Interrupt Enable) flip-flop. The CMP L and the SMP, CMP L signals are ORed together, gated with TP3, and the result used to clear both the FAST and SLOW flip-flops in the ERROR flip-flop and gating logic. As was explained in Paragraph 5.6, the ERROR flip-flop is then cleared at the trailing edge of TP3. The SMP L and the SMP, CMP L signal is ORed together and the result ANDed with ERROR (1) to drive the SKIP L line of the OMNIBUS. The SKIP L line is also grounded any time the SPO L line is grounded.

The one remaining output of the 8251, the CEP L line, is used to enable the special IOT control described in the following paragraph.

NOTE
This IOT decoding scheme is somewhat special, and should not be used as an example of general I/O decoding design.

Table 5-1
MP8-E IOT Summary

Octal	Mnemonic	Description
6100	DPI	Disable MP8-E interrupts.
6101	SMP	Skip if no parity error.
6102	---	Not used.
6103	EPI	Enable MP-E interrupts.
6104	CMP	Clear parity error flag.
6105	SMP, CMP	Skip if no parity error, clear error flag.
6106	CEP	Check for even parity. Complement the read parity but write odd parity in the next EXECUTE cycle only. Used for diagnostic purposes. Execution to a ROM address results in no operation.
6107	SPO	Skip if MP8-E is in machine.

5.8 SPECIAL IOT CONTROL

The CEP instruction gating requires additional comment. As shown in Figure 5-7, CEP ANDed with TP3 sets the DAE flip-flop, causing one input of gate A to go low. The next time the computer's major state becomes EXECUTE, OMNIBUS signal EL goes low. Gate A is fully enabled, generating EVEN PARITY CONTROL H and causing the output of gate B to go low. At the leading edge of TP2, the output of gate B goes high again; DAE clears; and EVEN PARITY CONTROL H is negated.

EVEN PARITY CONTROL H drives one input of the Parity Generator, causing the MP8-E to intentionally read wrong parity to test the MP8-E logic. Since EVEN PARITY CONTROL H is negated at TP2 (before the memory starts to rewrite), odd parity is written into the parity memory.

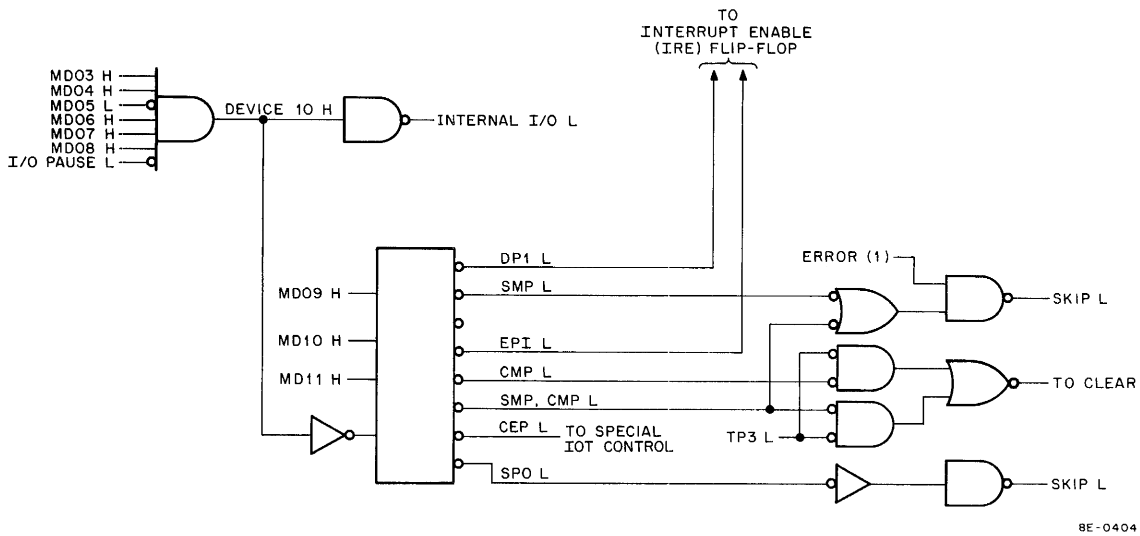


Figure 5-6 IOT Decoding and Skip, Interrupt Gating

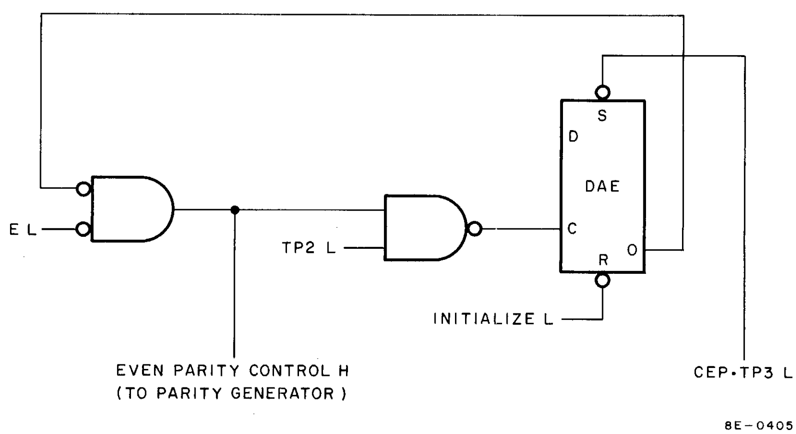


Figure 5-7 Special IOT Control

SECTION 5 MAINTENANCE

The MP8-E diagnostic program is a particularly good test of core memory stacks. If a stack operates properly in the MP8-E, it is virtually certain to operate properly in any MM8-E, unless there is a problem with one of the four bits that the MP8-E does not use. The MP8-E diagnostic should be run as a regular part of system maintenance. Diagnostic messages and the program listing should serve to pinpoint any commonly encountered malfunctions.

The same procedures as described under memory troubleshooting in Paragraph 4.7, Volume 1 are applicable. Note that the standard memory checkerboard programs are insufficient tests of the MP8-E, since a checkerboard pattern of all 1s and all 0s is used. The odd parity of all 0s is 1. Likewise, the odd parity of all 1s is 1. The strobe adjustment procedure for the MP8-E is the same as for the MM8-E, with the exception of the checkerboard program used.

SECTION 6 SPARE PARTS

Table 5-2 lists recommended spare parts for the MP8-E. These spare parts can be obtained from any local DEC office or from DEC, Maynard, Massachusetts.

Table 5-2
Recommended MP8-E Spare Parts

DEC Part No.	Description	Quantity
12-10043	Rotary Switch	1
16-09651	Transformer 8010	2
16-09996	Transformer 6501	1
16-09478	Transformer 1775	2
16-10031-0	Delay Line, 100 ns	1
13-10032	Resistor, 16.9 ohm, 6W, 1%	1
13-02858	Resistor, 100 ohm, 1/8W, 1%	2
13-02956	Resistor, 196 ohm, 1/8W, 1%	1
13-04858	Resistor, 348 ohm, 1/8W, 1%	1
13-02953	Resistor, 750 ohm, 1/8W, 1%	1
13-03114	Resistor, 1K ohm, 1/8W, 1%	1
13-02871	Resistor, 1.21K ohm, 1/8W, 1%	1
13-04833	Resistor, 1.96K ohm, 1/8W, 1%	1
13-04856	Resistor, 4.64K ohm, 1/8W, 1%	1
13-04885	Resistor, 9.09K ohm, 1/8W, 1%	1
13-02941	Resistor, 14.7K ohm, 1/8W, 1%	1
13-03156	Resistor, 34.8K ohm, 1/8W, 1%	1
13-05128	Resistor, 56.2K ohm, 1/8W, 1%	1
13-05252	Resistor, 68.1K ohm, 1/8W, 1%	1
13-10071	Thermistor, 1K, 1%	1
11-05275	Diode D672	7
11-00114	Diode D664	10
11-09991	Zener Diode 1/4M6, 8AZ1	1
19-10010	Diode Pack DEC 2501	2

(continued on next page)

Table 5-2 (Cont)
Recommended MP8-E Spare Parts

DEC Part No.	Description	Quantity
15-02155	Transistor DEC 1008	1
15-01881	Transistor DEC 2219	1
15-03100	Transistor DEC 3009B	1
15-10062	Transistor DEC 3734	2
15-09649	Transistor DEC 3762	1
15-10015	Transistor DEC 4008	2
15-05312	Transistor DEC 4258	1
15-03409-01	Transistor DEC 6534B	2
19-05575	IC DEC 7400	1
19-05590	IC DEC 7401	1
19-09004	IC DEC 7402	1
19-09686	IC DEC 7404	1
19-05580	IC DEC 7450	1
19-05547	IC DEC 7474	1
19-10724	IC DEC 74180	1
19-09056	IC DEC 74H00	1
19-09057	IC DEC 74H10	1
19-09267	IC DEC 74H11	1
19-05586	IC DEC 74H40	1
19-09967	IC DEC 74H74	1
19-09704	IC DEC 314	1
19-09485	IC DEC 380	1
19-09486	IC DEC 384	1
19-09594	IC DEC 8251	1
19-09705	IC DEC 8881	1

PART 3
REAL-TIME CLOCK OPTIONS

CHAPTER 6
REAL-TIME CLOCK OPTIONS

SECTION 1 INTRODUCTION

Three real-time clock options are available for use with the PDP-8/E. The DK8-EA and DK8-EC are similar; each consists of a clock frequency source and control logic contained on a single quad module that plugs into the OMNIBUS. These two clock options cause program interrupts of the PDP-8/E at predetermined intervals that are not subject to program control. The options can be used by the programmer to sample processes or to count events. The DK8-EP, a more sophisticated clock option, provides a large amount of program control. It provides the means to measure and/or count intervals and events in different ways. This option is discussed in detail in the *LAB 8-E Maintenance Manual.* Only the DK8-EA and DK8-EC are discussed here.

SECTION 2 BLOCK DIAGRAM

The difference between the DK8-EA and DK8-EC options is the way in which the clock frequency source operates. The DK8-EA derives a clock frequency of 100 Hz (for 50-Hz primary power) or 120 Hz (for 60-Hz primary power) from a power supply ac voltage. The DK8-EC derives a clock frequency of 1 Hz, 50 Hz, 500 Hz, or 5 kHz from a 20-MHz crystal-controlled oscillator. Because the options are similar in all other respects, the logic description pertains to both, except when the clock frequency logic is discussed. Reference designations on logic symbols, E3, E12, for example, are for reference only. They may or may not coincide with the reference designations on a specific schematic drawing.

Figure 6-1 is a block diagram of the DK8-EA/C. The control logic consists of the select logic, the INTERRUPT flip-flop logic, and the clock flag logic. The clock frequency is provided by the clock logic.

When an IOT instruction causes I/O PAUSE L to be asserted by the CPU Timing Generator, the option select logic decodes bits MD-03 through MD-11. The INTERNAL I/O L signal is asserted to direct the positive I/O bus interface to ignore the IOT instruction. The three instructions that pertain to the DK8-EA/C are 6131, 6132, and 6133. 6131 and 6132 are used in the INTERRUPT flip-flop logic where they set and clear, respectively, the INTERRUPT flip-flop. If this flip-flop is set, the clock flag is logically connected to the interrupt system. Upon receipt of an interrupt request, the computer begins to execute the interrupt servicing routine to determine the identity of the requesting device. When the 6133 instruction in the servicing routine is decoded, the clock flag logic asserts SKIP L. At the same time, the CLOCK FLAG flip-flop is cleared. The computer then proceeds to the subroutine associated with the Real-Time Clock option. Both the INT RQST L and SKIP L signals are negated. The next clock pulse asserts INT RQST L again and the procedure is repeated. Each succeeding clock pulse causes an interrupt request until the 6132 instruction clears the INTERRUPT flip-flop.

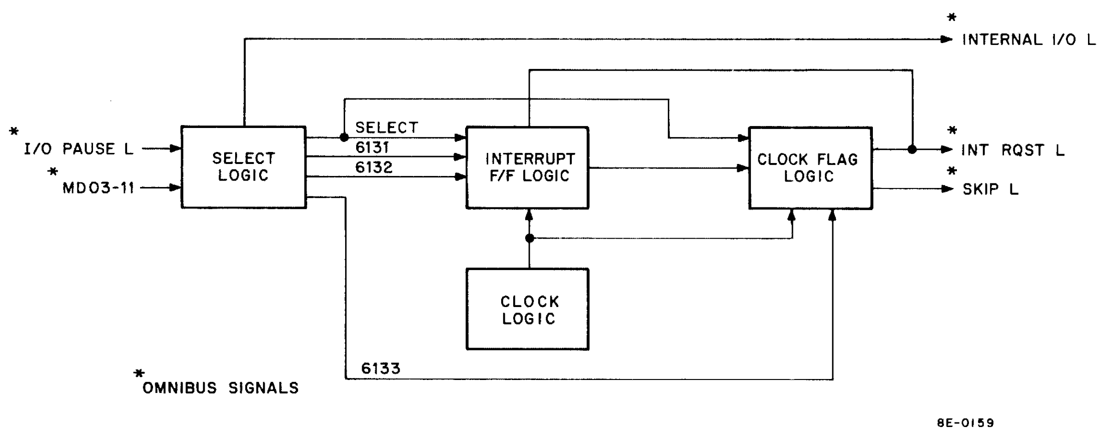


Figure 6-1 Real-Time Clock (DK8-EA, DK8-EC), Block Diagram

SECTION 3 DETAILED LOGIC

6.1 SELECT LOGIC

The select logic is shown in Figure 6-2. The SELECT signal is asserted by NAND gate E5 when a 613X instruction is decoded. The SELECT signal, in turn, asserts INTERNAL I/O L, which causes the positive I/O bus interface to ignore the IOT instruction. The SELECT signal is gated with bits MD-09 through MD-11 to provide instruction 6131, 6132, or 6133.

6.2 INTERRUPT FLIP-FLOP LOGIC

The INTERRUPT flip-flop logic is shown in Figure 6-3. The flip-flop, E11B, is cleared by INITIALIZE; it can be set by the first clock pulse following power turn-on. This flip-flop remains set until cleared by the 6133 instruction, as is discussed in Paragraph 6.3.

The first TP1 pulse to be generated after E11A is set causes flip-flop E7 to be set; the 1-output of E7 provides the desired high level at one input of NAND gate E9. The other input of the gate is from the 1-output of the INTERRUPT flip-flop. If the select logic decodes 6131, the INTERRUPT flip-flop is set at TP3 and E9 asserts INT RQST L.

The 6132 instruction is used to disable the interrupt capability of the option. It, like the 6131 instruction, is contained in the background program and, when decoded by the select logic, clears the INTERRUPT flip-flop at TP3.

6.3 CLOCK FLAG LOGIC

The clock flag logic is shown in Figure 6-4. When 6133, which is usually located in the interrupt servicing routine, is decoded by the select logic, NAND gate E9B in the clock flag logic asserts SKIP L (E7 is set at TP1 following power turn-on). At TP3, the CLOCK FLAG flip-flop is cleared. Simultaneously, the CPU SKIP flip-flop, which is conditioned by SKIP L, is set. This action causes the program counter to be incremented during TS4. Thus, the instruction following 6133 is skipped and the next instruction that is performed is the first instruction of the real-time-clock subroutine. At TP1 of this instruction, latch E7 in the clock flag logic is cleared, negating both

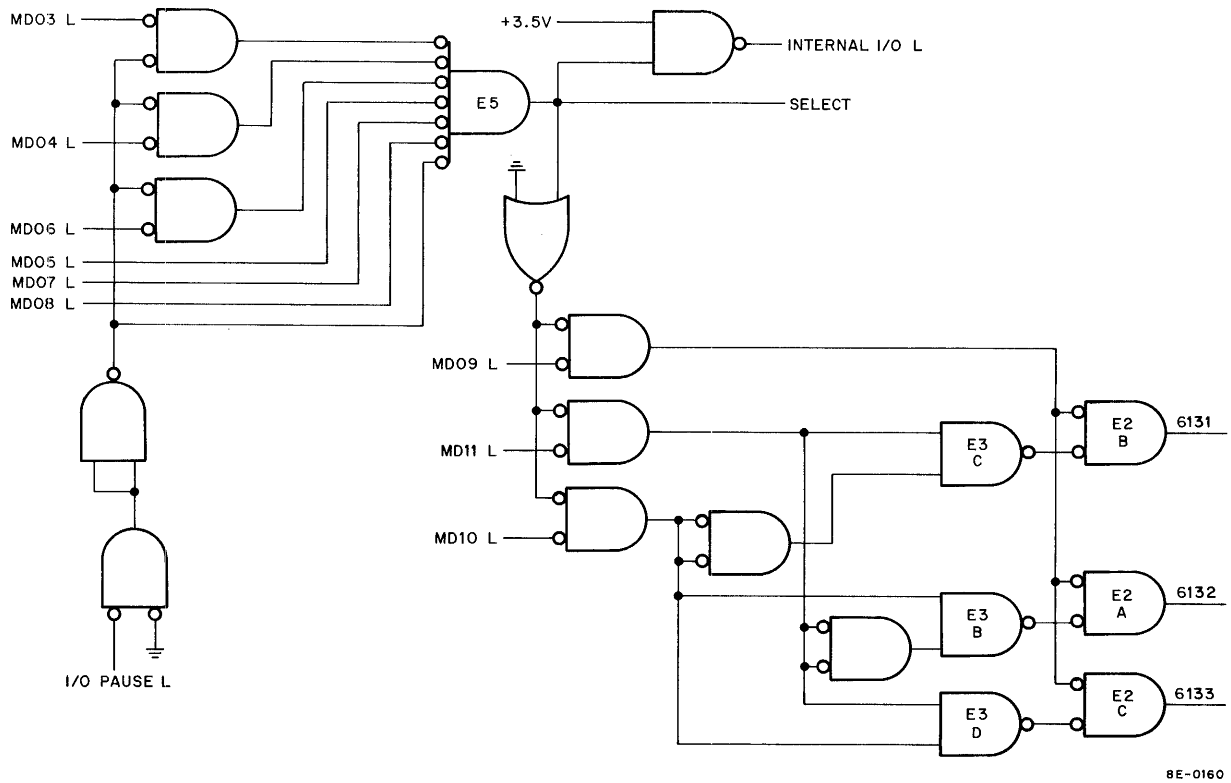


Figure 6-2 Select Logic

INT RQST L and SKIP L. The next interrupt request occurs when the CLOCK FLAG flip-flop is again set by a clock pulse and the following TP1 pulse causes E7 to be set.

Naturally, the time required for the service loop must be less than the clock period. Furthermore, the clock is free-running. Hence, the program must be able to handle two nearly simultaneous clock interrupts when the CLOCK INTERRUPT flip-flop is enabled. Consider what happens if the clock is enabled just before it is ready to generate a clock pulse. The enabling process generates an interrupt and the CPU may still be executing the first clock subroutine when the option requests the second program interrupt. If this occurs, the interrupt servicing routine is entered immediately after the first clock subroutine is exited, and the second execution of the subroutine occurs just after the first. Thus, an uncertainty of one count is always present because of the unknown clock phase.

6.4 CLOCK LOGIC

6.4.1 DK8-EA

The clock logic for the DK8-EA is shown in Figure 6-5. The DK8-EA connects to the 28-Vac output of the H724 or H724A Power Supply via the cable supplied with the option. The cable connects to the option with an 8-pin connector (2 pins are not used). Each ac input is wired, via the board etch, to an adjacent pin so that the 28 Vac is not dead-ended (the KP8-E Power Fail and Auto Restart option also uses the 28-Vac output of the power supply; if both options are in the system, one is connected directly to the power supply, while the second is connected to J1 of the first).

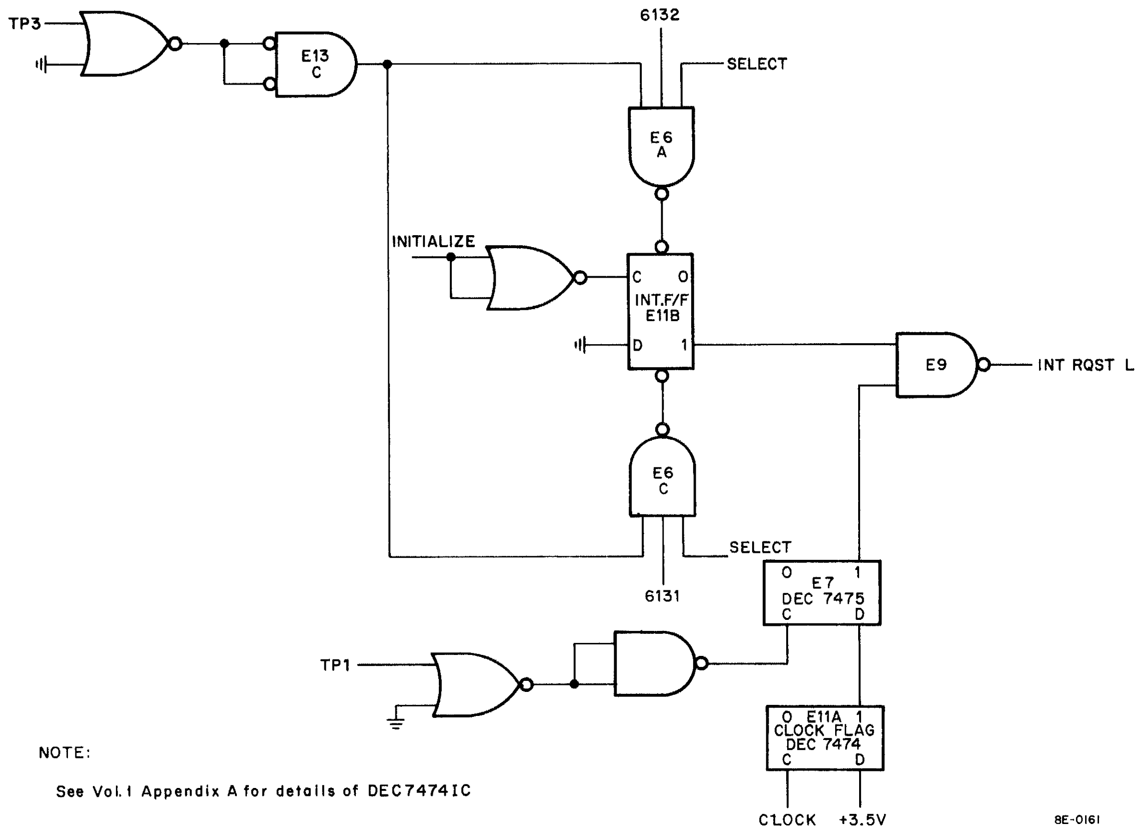


Figure 6-3 INTERRUPT Flip-Flop Logic

The circuit consists of a full-wave rectifier, a clamp, and a Schmitt trigger. The rectified ac voltage is clamped to +5V by D1 to prevent damage to the E12 IC. E12 is wired as a Schmitt trigger and its output is a reasonably well-shaped square wave with a pulse repetition frequency that is twice the frequency of the primary power source.

6.4.2 DK8-EC

The clock logic for the DK8-EC is shown in Figure 6-6. The basic clock frequency, 20 MHz, is provided by a crystal-controlled oscillator (see the option schematic). This frequency is divided by a factor of four by the two J-K flip-flops, E17 and E18 (when both the J and the K inputs are high, the 1 output is changed with each positive transition at the C input). The 5-MHz clock frequency is applied to a chain of DEC 7490 decade counters, each counter except the last is wired to divide by ten; the last counter is wired to divide by five. The output of counter E4, E8, E12, or E21 can be selected by connecting a jumper wire, as illustrated in Figure 6-6. (The option is manufactured with an etch connection from the output of E21 to the terminal that connects to the CLOCK FLAG flip-flop; if a clock frequency of other than 1 Hz is desired, cut the etch connection and wire a jumper from the terminal to the desired flip-flop output.)

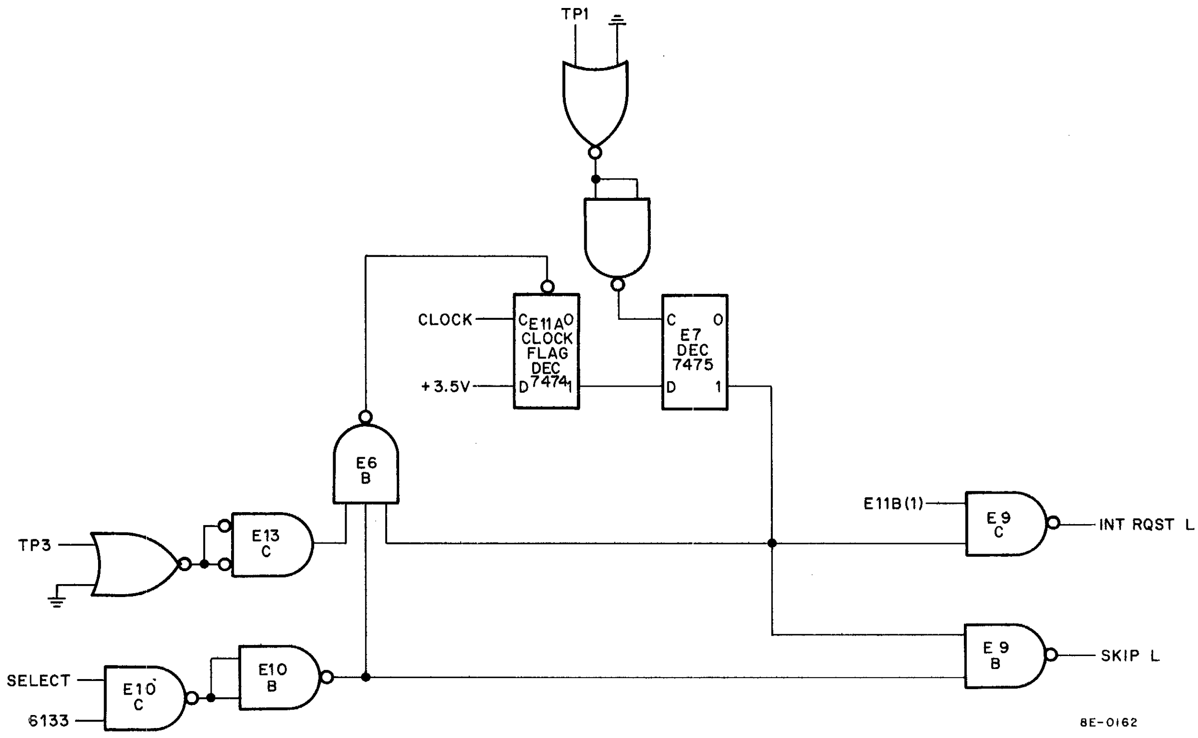


Figure 6-4 Clock Flag Logic

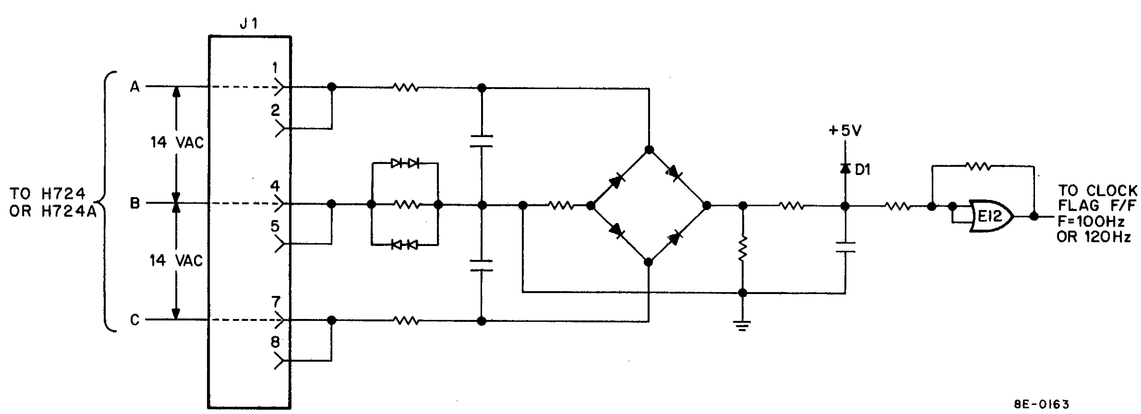


Figure 6-5 Clock Logic DK8-EA

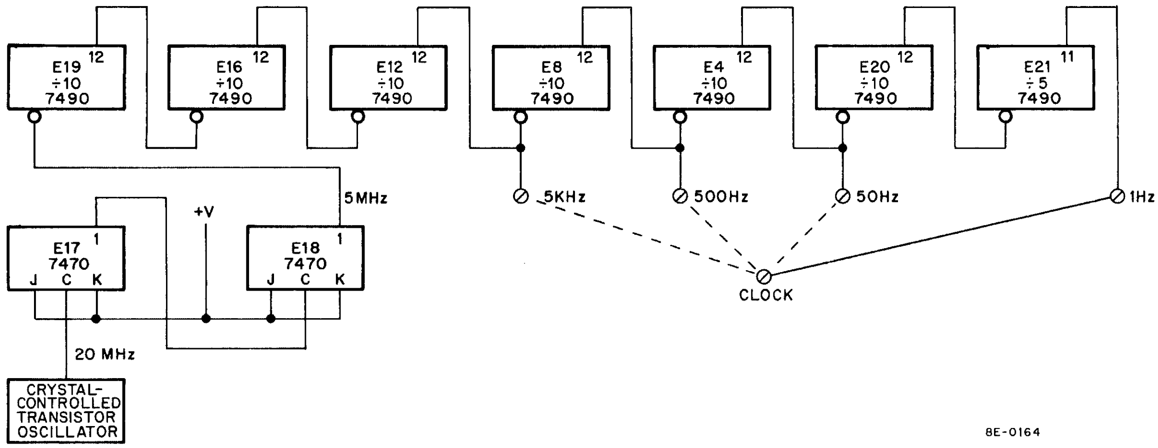


Figure 6-6 Clock Logic, DK8-EC

SECTION 4 MAINTENANCE

General instructions concerning preventive and corrective maintenance are given in Chapter 4, Volume 1. When corrective maintenance is required, use the MAINDEC-8E-D8AA maintenance program to determine the nature of the problem. The option schematics, E-CS-M883-0-1, E-CS-M882-0-1, E-CS-M860-0-1, and E-CS-M518-0-1, must be referred to for IC locations and pin numbers. Test points are provided on the option to facilitate trouble-shooting.

SECTION 5 SPARE PARTS

Table 6-1 lists the M882/M883 spare parts. Spare parts can be obtained from any local DEC office or from DEC, Maynard, Massachusetts.

Table 6-1
Recommended DK8-EA/DK8-EC – (M882/M883) Spare Parts

DEC Part No.	Description	Quantity
19-9705	DEC 8881	1
19-9704	DEC 314	1
19-9485	DEC 380	1
19-9051	DEC 7490	1
19-9050	DEC 7475	1
19-9004	DEC 7402	1
19-5589	DEC 7470	1
19-5576	DEC 7410	1
19-5575	DEC 7400	1
19-5547	DEC 7474	1
19-9486	DEC 384	1

(continued on next page)

Table 6-1 (Cont)
Recommended DK8-EA/DK8-EC – (M882/M883) Spare Parts

DEC Part No.	Description	Quantity
18-9880	Crystal (M883 only)	1
16-9651	Pulse Transformer (M883 only)	1
10-9678	Capacitor 0.047 μF, 16-15 + 20%	1
10-1610	Capacitor 0.01 μF, 100V, 20%	1
10-0016	Capacitor 100 pF, 100V, 5%	1
10-0014	Capacitor 68 pF, 100V, 5%	1
10-0011	Capacitor 47 pF, 100V, 5%	1
10-0006	Capacitor 10 pF, 100V, 5%	1
10-1765	Capacitor 0.005 μF,	1

PART 4
POWER-FAIL OPTION

CHAPTER 7
KP8-E POWER-FAIL AND AUTO-RESTART

SECTION 1 INTRODUCTION

The KP8-E Power-Fail and Auto-Restart option monitors the computer's primary power source and initiates a controlled shut-down sequence if a power failure occurs. This power-fail sequence protects the operating program by storing the contents of the PC Register, AC Register, MQ Register, and the Link in known memory locations. When normal primary power is restored, the KP8-E automatically restarts the computer in location 0000.

The Power-Fail and Auto-Restart option consists of the M848 quad module which is inserted into the OMNIBUS and connected to the computer ac power supply by a 7007128 power cable. The PDP-8/E supplies 28 Vac and the PDP-8/F and PDP-8/M supply 56 Vac. The 56 Vac input is reduced to 28 Vac by removing a jumper (W2) on the M848 module.

SECTION 2 M848 BLOCK DIAGRAM

The KP8-E block diagram is shown in Figure 7-1. The power monitor logic checks the PDP-8/E power supply 28 Vac output or the PDP-8/F and PDP-8/M 56 Vac power supply output which reflects the condition of the primary power source. If line voltage drops below a predetermined minimum value, the UP signal is negated and the PWR LOW flip-flop is set, asserting the OMNIBUS INT RQST L signal. Filter capacitors in the power supply guarantee continued operation for 1 ms; this is sufficient time for the interrupt request to be recognized and the program interrupt routine to be carried out (because of the time limitation, the KP8-E SPL instruction, Skip on PWR LOW flag, 6102, should be the first status check made by the program interrupt routine).

SECTION 3 M848 DETAILED LOGIC

7.1 SELECT LOGIC

The select logic is shown in Figure 7-2. When the SPL instruction (6102) is decoded, the logic asserts the INTERNAL I/O L signal that causes the positive I/O bus interface to ignore the IOT instructions. The status of the PWR LOW flip-flop is checked; if the flag is set, indicating a power failure has occurred, the SKIP L signal is asserted. The program then skips the next sequential instruction and jumps to a subroutine that begins executing the power-fail routine.

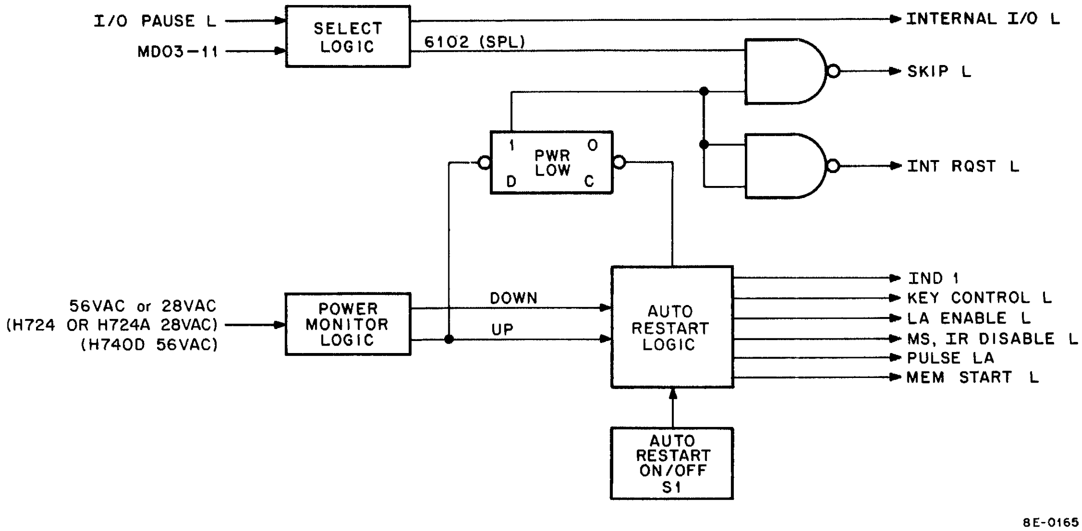


Figure 7-1 M848 Power Fail and Auto-Restart Option, Block Diagram

7.2 POWER MONITOR LOGIC

The power monitor logic is shown in Figure 7-3. The logic directly monitors the ac output of the computer's power supply, to which the option is connected by a cable. The cable connects to the option with an 8-pin connector (2 pins are not used). Each ac input at J1 is wired via the board etch to an adjacent pin so that the ac is not dead-ended (the DK8-E Real-Time Clock option (line frequency) also uses the ac output of the power supply; if both options are in the system, one is connected directly to the power supply while the second is connected to J1 of the first).

NOTE

The W2 jumper must be removed when the KP8-E is installed in a PDP-8/F or PDP-8/M.

The ac is full-wave rectified and applied to two comparator circuits. One of these circuits includes transistor pair Q4/Q5 and initiates the auto-restart sequence. The second circuit includes transistor pair Q2/Q3 and initiates the power-fail sequence. There are two thresholds for power fail. An upper threshold, 105 Vac, that is used to start the RESTART logic, and a lower threshold, 95 Vac, that sets the PWR LOW flip-flop. Q4 and Q5 detect the 105 Vac threshold; Q2 and Q3 detect the 95 Vac threshold. The upper and lower thresholds have a tolerance of ±15%. Q2 provides a trigger for one-shot E7 when the amplitude of the line voltage, and hence, the amplitude of the ac input of J1, is above the desired minimum. The −10.3 Vdc reference voltage is generated by a precision voltage regulator that is not shown (see the option schematic for this and for resistor values).

Figure 7-4 presents idealized waveforms to illustrate how E7 is controlled by the comparator circuit. If the line voltage is above the selected minimum value, the positive transition at the collector of Q2 will be of sufficient amplitude to trigger E7. Because the period of the collector waveform is less than the triggered delay time of E7, the one-shot will remain active. If the line voltage falls below the minimum value for as little as one-half cycle, as illustrated, E7 times out and an interrupt request is generated. Even though the power recovers almost instantaneously, the power-fail sequence is carried out.

The timing diagram shows the UP signal being asserted by the half-cycle immediately following the missing half-cycle. Thus, the auto-restart sequence begins 1500 ms later, as detailed in the following paragraphs.

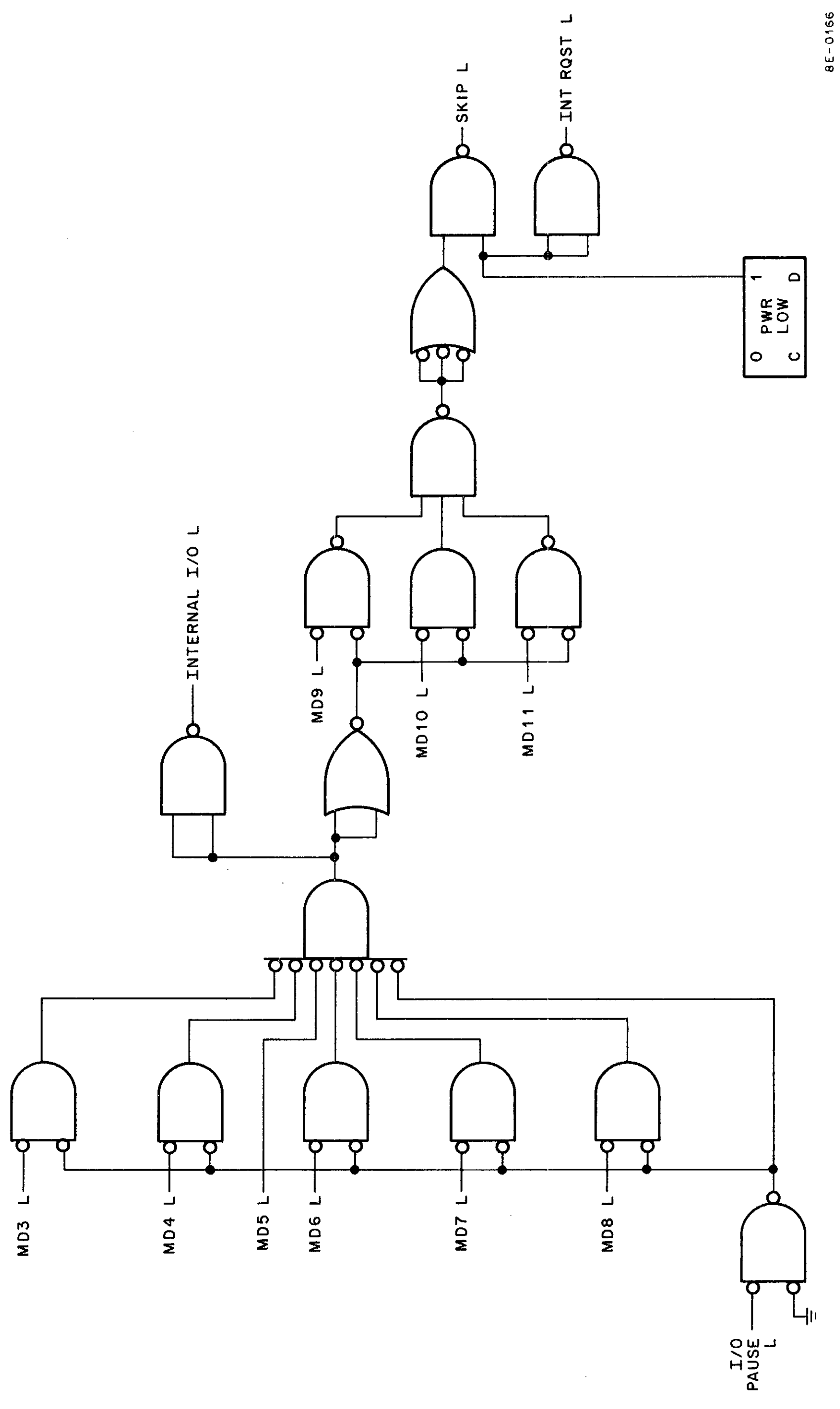
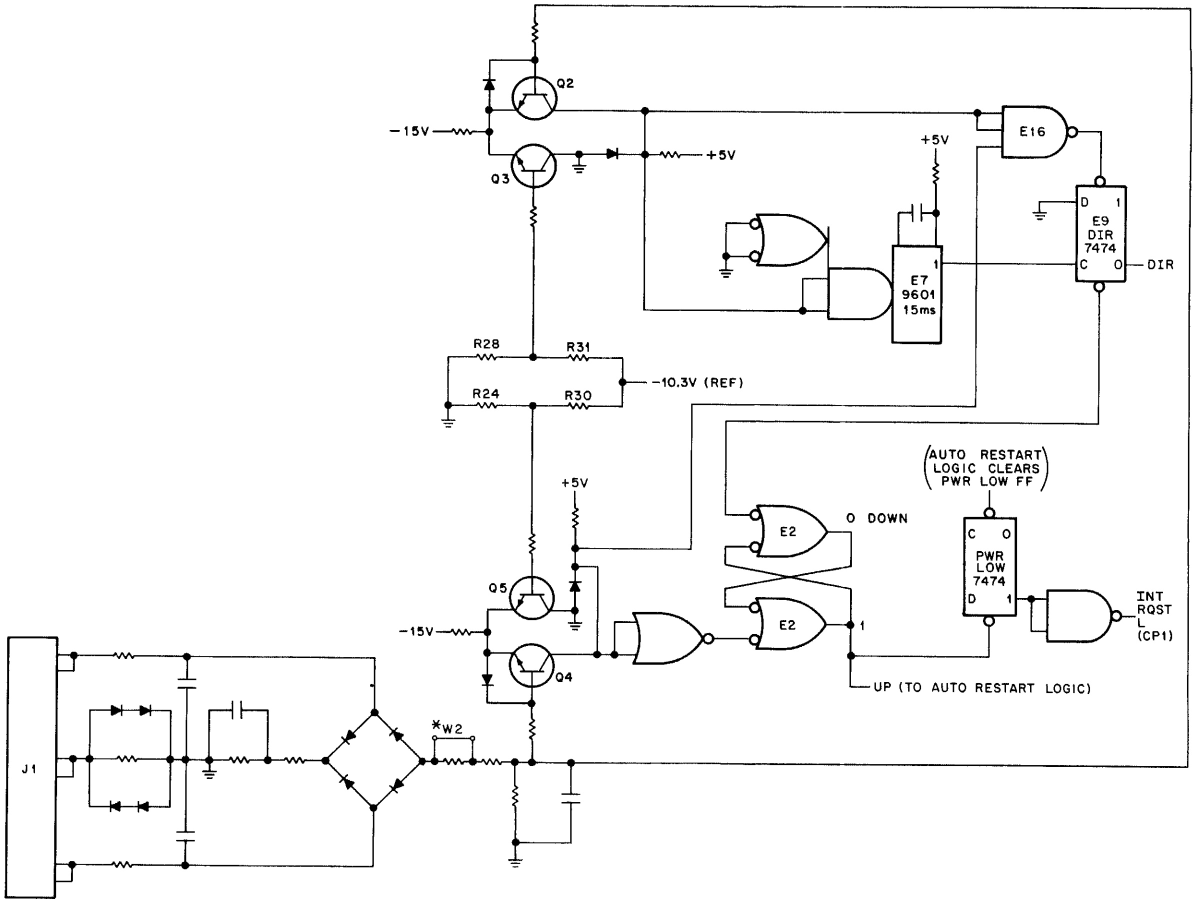


Figure 7-2 Select Logic



* W2 MUST BE REMOVED WHEN INSTALLED IN PDP-8/M COMPUTER TO REDUCE 56VAC TO 28VDC.

8E-0167

Figure 7-3 Power Monitor Logic

7.3 AUTO-RESTART LOGIC

The computer must resume operation by executing the instruction that was stored in location 0000, field 0, by the power-fail routine. Consequently, the CPMA Register and the IF and DF Registers of the KM8-E option must be loaded with 0s before CPU timing is renewed. Furthermore, the CPU Major State Register must be manipulated so that a FETCH cycle is entered when timing begins. The auto-restart logic meets these requirements by simulating some of the operations that normally occur when the programmer's console is being used.

Thus, to load the CPMA Register with 0s, the auto-restart logic first asserts the LA ENABLE L signal (at the same time, the MS, IR DISABLE L signal is asserted; this signal places the CPU in the DMA state, ensuring that the first timing cycle begins in the FETCH state). The LA ENABLE L signal:

a. ensures that only "bus" information is placed on the DATA 0-11 lines; because nothing has access to "bus" at this time, the DATA 0-11 lines carry 0s (Volume 1, Section 5, Paragraph 3.3.3, for clarification);

b. causes the 0s on the DATA 0-11 lines to be gated through the CPU Major Register gating to the MAJOR REGISTERS BUS.

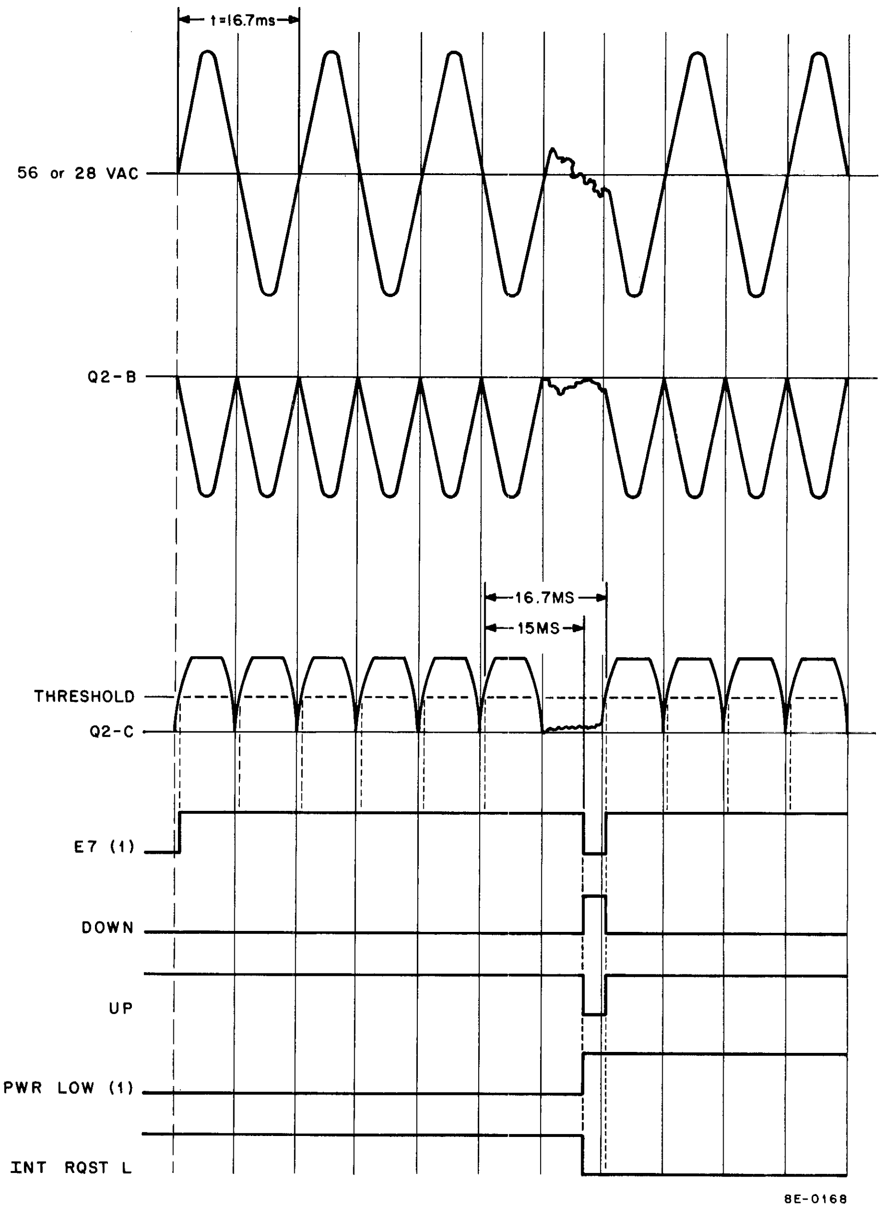


Figure 7-4 Power Fail Timing

After a delay to ensure that the control lines have settled, the logic asserts PULSE LA. This signal causes the CPU to generate the CPMA LOAD L signal that loads the CPMA Register with address 0000. CPMA LOAD L also sets the F flip-flop of the Major State Register; thus, a FETCH cycle will be entered when timing begins. IND1 is asserted at the same time as LA ENABLE to ensure only 0s are on the bus if the programmer's console is not installed (Volume 1, Paragraph 3.33.10).

When the CPMA Register has been loaded and the F flip-flop set, the auto-restart logic asserts the KEY CONTROL L signal. After a delay that enables the control line to settle, PULSE LA is asserted again. However, because KEY CONTROL L is true, CPMA LOAD L is not generated by this assertion of PULSE LA, nor is the Major State Register clocked.

Rather, the 0s on the DATA 0-11 lines are loaded into the IF and DR Registers of the KM8-E option (the auto-restart logic allows for the KM8-E option even if the option is not contained in the system). After this assertion of the PULSE LA signal, the MS, IR DISABLE L, IND1, and LA ENABLE L signals are negated and a final delay period is allowed. When this delay times out, MEM START L is asserted, initiating CPU timing, and the computer fetches the instruction from location 0000.

The auto-restart logic is shown in Figure 7-5; the relative timing of the logic is shown in Figure 7-6. The ENABLE/DISABLE switch, S1, must be in the ENABLE position (up) if the automatic restart is to function after a power interrupt has occurred. If S1 is in the DISABLE position, the PWR LOW flip-flop is cleared when ac power comes up, but the program must be restarted manually.

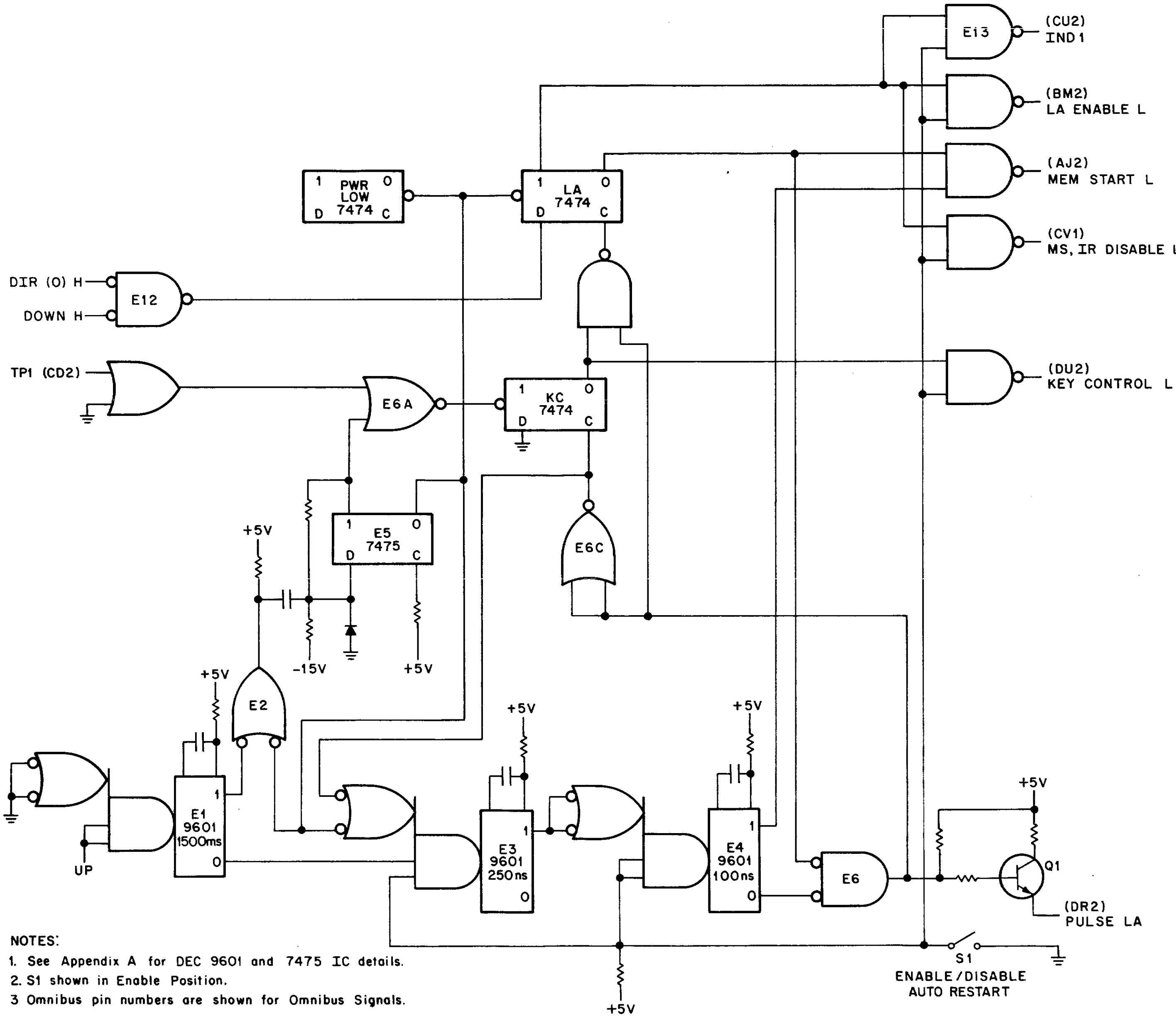
The auto-restart sequence begins when the power monitor logic asserts the UP signal. The positive transition of this signal triggers one-shot multivibrator E1. During the active 1500 ms of this one-shot, all system equipment (computer, peripherals, options) can complete operations initiated by the OMNIBUS INITIALIZE signal. At the end of the 1500 ms delay, bistable latch E5 is triggered and the LA flip-flop is set. This latch sets the KC (key control) flip-flop, clears the PWR LOW flip-flop, and triggers the E3 one-shot (the 0-output of E1 provides a required high signal at one input of E3; as the timing diagram illustrates, this high precedes the E3 trigger signal by an appreciable amount of time). The 1-output of the LA flip-flop asserts the LA ENABLE L, IND1 and MS, IR DISABLE L signals. The E3 one-shot, when it times out after 250 ns, triggers one-shot E4, which is active for 100 ns. During the 100 ns period, NAND gate E6 is enabled and PULSE LA is asserted. Thus, the CPMA Register is loaded and the FETCH flip-flop is set. The E3 one-shot is retriggered, via NOR gate E6C, coincidently with the leading edge of the PULSE LA signal; while the KC flip-flop is cleared 100 ns later to assert the KEY CONTROL L signal. When E3 times out the second time, PULSE LA is produced again. Thus, the IF and DF Registers are loaded at this time. As before, E3 is retriggered coincidently with the leading edge of PULSE LA. At the end of PULSE LA, when E4 times out for the second time, the LA flip-flop is cleared, and LA ENABLE L and MS, IR DISABLE L are negated.

When E3 times out for the third time, it once again triggers E4. However, because the LA flip-flop is now clear, no PULSE LA signal is produced. Instead, the 1-output of E4 asserts MEM START L and CPU timing begins by fetching the instruction in location 0000. At TP1 time of this FETCH cycle, the KC flip-flop is set by NOR gate E6A and the KEY CONTROL L signal is negated.

The DIR (Direction) flip-flop ensures that a Restart Sequence is initiated only when ac voltage is rising, through the Restart threshold, toward normal line Voltage Condition.

SECTION 4 MAINTENANCE

General instructions concerning preventive and corrective maintenance are given in Volume 1, Chapter 4. When corrective maintenance is required, the technician should use the maintenance program, MAINDEC-8E-DOKC-D (D), to determine the nature of the problem. The option schematic, drawing no. E-CS-M848-0-1, must be referred to for IC locations and pin numbers. Tests points have been provided on the option to facilitate troubleshooting.



NOTES:
1. See Appendix A for DEC 9601 and 7475 IC details.
2. S1 shown in Enable Position.
3 Omnibus pin numbers are shown for Omnibus Signals.

Figure 7-5 Auto-Restart Logic

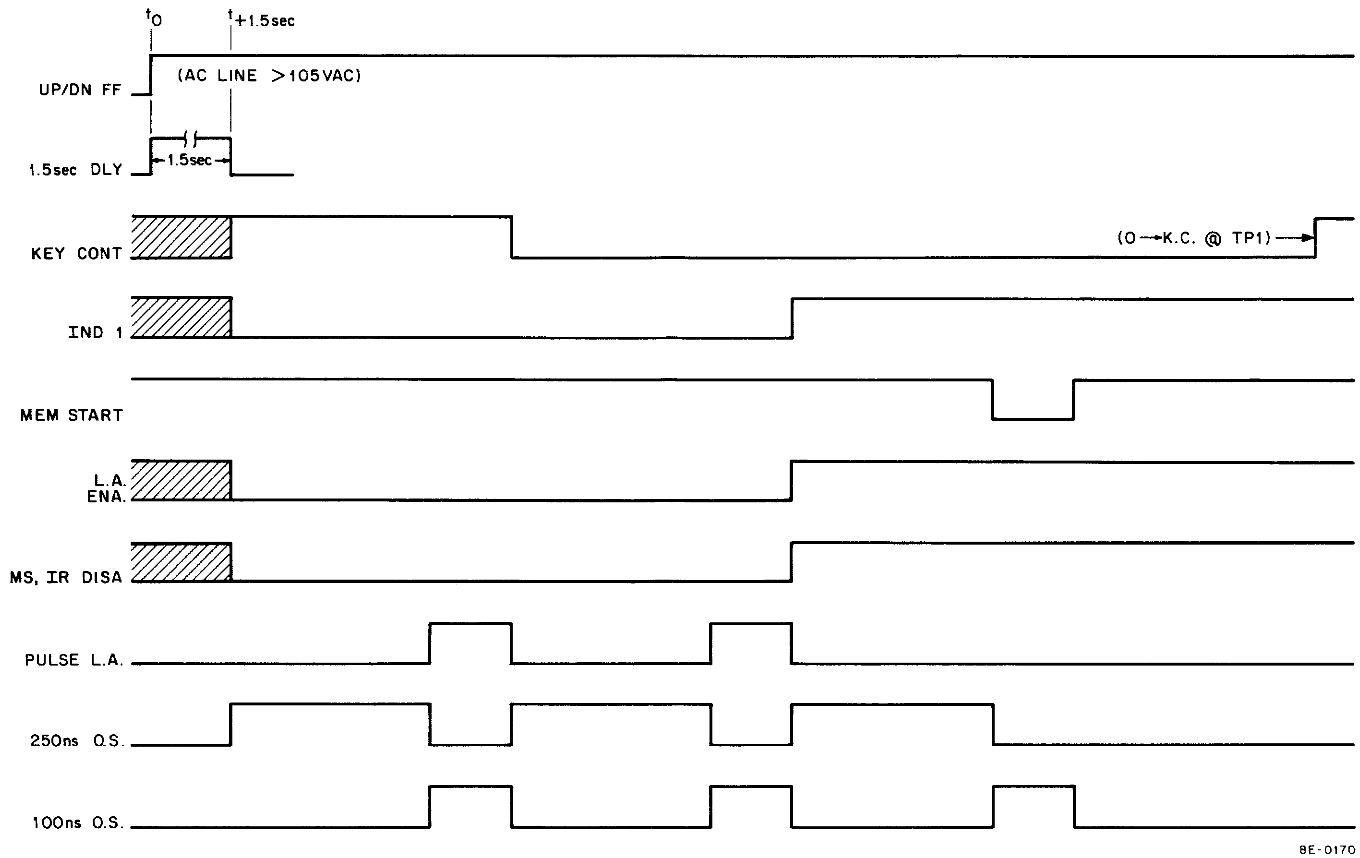


Figure 7-6 Auto-Restart Timing

SECTION 5 SPARE PARTS

Table 7-1 lists recommended spare parts for the KP8-E. These parts can be obtained from any local DEC office or from DEC, Maynard, Massachusetts.

Table 7-1

Recommended KP8-E Spare Parts

DEC Part Number	Description	Quantity
11-09991	Diode, AZ1-1/4M, 6.8V	1
11-00114	Diode, D664	1
11-00275	Diode, D672	1
15-03100	Transistor, 3009B	1
15-03409-01	Transistor, DEC 6534B	1
19-05547	IC DEC 7474	1
19-05575	IC DEC 7400	1
19-05576	IC DEC 7410	1
19-09004	IC DEC 7402	1
19-09050	IC DEC 7475	1
19-09373	IC DEC 9601	1
19-09486	IC DEC 384	1
19-09705	IC DEC 8881	1
19-09971	IC DEC 6380	1
19-09972	IC DEC 6314	1

PART 5
INTERPROCESSOR OPTION

CHAPTER 8
DB8-E INTERPROCESSOR BUFFER

SECTION 1 INTRODUCTION

The DB8-E Interprocessor Buffer is designed to plug directly into the PDP-8/E OMNIBUS. This option allows two PDP-8/E Computers to transfer data between themselves, one 12-bit word at a time, at a software-limited rate of 50 kHz. The DB8-E can also be used to transfer data to user-designed logic on single-ended data lines.

The basic DB8-E option has one M8326 Module and one BC08-R cable (up to 100-ft long). The DB8-EB has one M8326 Module, two BC08-R cables, and two 5409209 Module Adapters for connection to user-designed logic.

Device codes on the module are jumper-selected between 50 and 57, allowing a maximum of eight Interprocessor Buffers to be used in one PDP-8/E.

SECTION 2 INSTALLATION

The DB8-E Interprocessor Buffer is installed on site by DEC Field Service personnel. The customer should **not** attempt to unpack, inspect, install, checkout, or service the equipment.

8.1 INSTALLATION

To install the Interprocessor Buffer, remove power from PDP-8/E No. 1 and insert the M8326 Module into the OMNIBUS. Refer to Table 2-3, Volume 1, for information about recommended module priorities (the DB8-E is a non-memory option). Remove power from PDP-8/E No. 2 and insert the second M8326 Module into its OMNIBUS. Connect one of the BC08-R cables between J1 of the first and J2 of the second DB8-E Module. Connect the second BC08-R cable between J2 of the first DB8-E and J1 of the second DB8-E. This will connect the output of PDP-8/E No. 1 buffer to the input of PDP-8/E No. 2 module and the output of PDP-8/E No. 2 to the input of PDP-8/E No. 1. Table 8-1 shows the pin-to-pin connections of J1 and J2.

Table 8-1
Connection from J1 to J2

J1 Pin No.	Adapter Module Pins	J2 Pin No.	Adapter Module Pins
J1TT	A1	J2C	V2
J1RR	B1	J2E	U2
J1D	V1	J2SS	A2
J1F	U1	J2PP	B2
J1J	T1	J2MM	C2

(continued on next page)

Table 8-1 (Cont)
Connection from J1 to J2

J1 Pin No.	Adapter Module Pins	J2 Pin No.	Adapter Module Pins
J1L	S1	J2KK	D2
J1N	R1	J2HH	E2
J1R	T1	J2EE	F2
J1T	N1	J2CC	H2
J1V	M1	J2AA	J2
J1X	L1	J2Y	K2
J1Z	K1	J2W	L2
J1BB	J1	J2U	M2
J1DD	H1	J2S	N2

NOTE: All pins not listed are tied to ground.

8.2 ACCEPTANCE TEST

The acceptance test should be performed when the DB8-E Modules are installed and periodically after installation to check the operation of the DB8-E logic. A working M8326 Test Module is needed to perform this test. The programs in Section 5 can be used for preliminary operational checks.

Perform the following steps to check the DB8-E Modules installed in the PDP-8/E OMNIBUS.

NOTE
The PDP-8/E under test will be referred to as PDP-8/E No. 1; the PDP-8/E with test module used to check the DB8-E Module in PDP-8/E No. 1 will be referred to as PDP-8/E No. 2.

Step	Procedure
1	Remove the DB8-E Module from PDP-8/E No. 2 and install the test module in its place; connect the cables from the DB8-E Module to the test module.
2	Load binary loader in PDP-8/E No. 1 and No. 2.
3	Load diagnostic MAINDEC-8E-DOPA-PB in PDP-8/E No. 1 and No. 2.
4	Run Part 1 and Part 2 of the test for 5 minutes each. There should be no errors.
5	If there are no errors, remove the test module from PDP-8/E No. 2 and reinstall the DB8-E Module.
6	To check the DB8-E Module in PDP-8/E No. 2, repeat Steps 1 through 5 with test module in PDP-8/E No. 1.

SECTION 3 SYSTEM DESCRIPTION

The Interprocessor Buffer (Figure 8-1) receives or transmits data under control of programmed instructions from the CPU. Data can be put on the DATA BUS of the OMNIBUS to be transferred to the accumulator, or data can be taken from the DATA BUS and transferred to another PDP-8/E or user's equipment.

The following instructions are used to program interprocessor data transfers.

Skip on Receive Flag (DBRF)

Octal Code:	65X1
Operation:	Skip if the RECEIVE FLAG equals one.

Read Incoming Data (DBRD)

Octal Code:	65X2
Operation:	Read the incoming data into the AC, clear the RECEIVE FLAG, and set DONE flip-flop.

Skip on Transmit Flag (DBTF)

Octal Code:	65X3
Operation:	Skip if the DONE FLAG equals one.

Transmit Data (DBTD)

Octal Code:	65X4
Operation:	Transfer the contents of the AC Register to the transmit buffer. Transmit data and set the FLAG.

Enable Interrupt (DBEI)

Octal Code:	65X5
Operation:	Enable the interrupt request line.

Disable Interrupt (DBDI)

Octal Code:	65X6
Operation:	Disable the interrupt line.

Clear Done Flag (DBCD)

Octal Code:	65X7
Operation:	Clear the DONE FLAG.

To transfer data from PDP-8/E No. 1 to PDP-8/E No. 2, data is loaded into the AC of PDP-8/E No. 1 and transferred to the output buffer of its DB8-E using the 65X4 IOT. In addition to loading the output buffer, IOT 65X4 also sets the FLAG flip-flop in PDP-8/E No. 2. When the program in PDP-8/E No. 2 has sensed its FLAG flip-flop, data is gated into the AC of PDP-8/E No. 2 using IOT 65X2. IOT 65X2 then sets the DONE flip-flop of PDP-8/E No. 1, indicating the transfer was completed, and clears the FLAG flip-flop of PDP-8/E No. 2. PDP-8/E No. 1 then clears its DONE flip-flop using IOT 6507.

SECTION 4 DETAILED LOGIC

The logic in the Interprocessor Buffer will be broken into functional groups for discussion purposes. Figure 8-1 should be used to understand the relationship between each group of logic.

8.3 DEVICE SELECT LOGIC

The device select logic is shown in Figure 8-2. Bits MD03 through MD11 are gated by I/O PAUSE when a 65XX instruction is decoded. An INT I/O L and SELECT L will be asserted to allow the operation decoder to receive its input and to cause the positive I/O bus interface to ignore the IOT instruction.

8.4 OPERATION SELECT LOGIC

SELECT L in Figure 8-3 will enable the gates for bits MD09 through MD11 and allow inputs to the operation decoder which is a BCD-to-decimal decoder. (Refer to Appendix A, Volume 1, for details about the 8251 IC.) The decoder will supply signals that represent instructions 65X1 through 65X7. When a 65X2 instruction is decoded, the C line select logic will pull C0 and C1 low to allow data to be transferred from the DATA BUS to the AC.

8.5 INTERRUPT AND SKIP LOGIC

The interrupt and skip logic is used to interrupt the program when a data transfer is required. To allow an interrupt to occur, the INT ENA flip-flop must be set (Figure 8-4) by a 65X6 instruction. The 1-output of the INT ENA flip-flop will go to E17 and assert one side of the AND gates allowing them to generate INT RQST L. A more detailed explanation of interrupt and skip logic can be found in Volume 1.

Table 8-2 shows the signal and data flow required to transfer information between two PDP-8/E Computers.

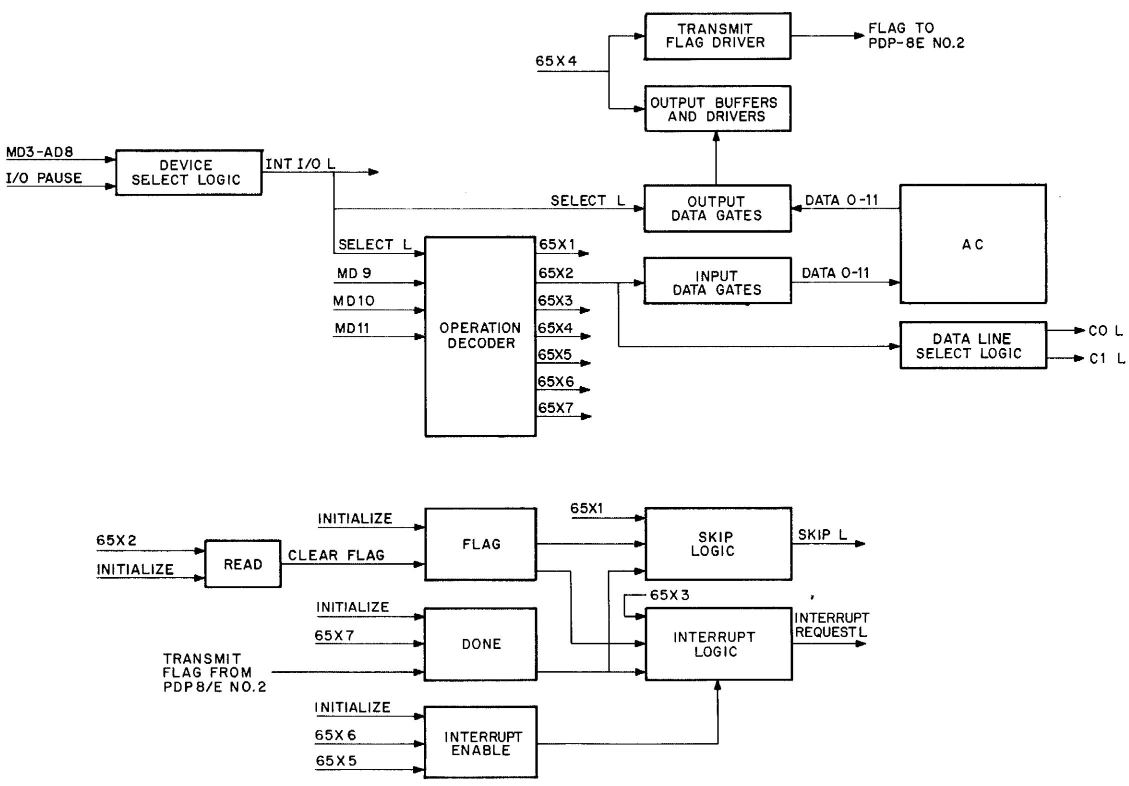


Figure 8-1 Interprocessor Buffer, Block Diagram

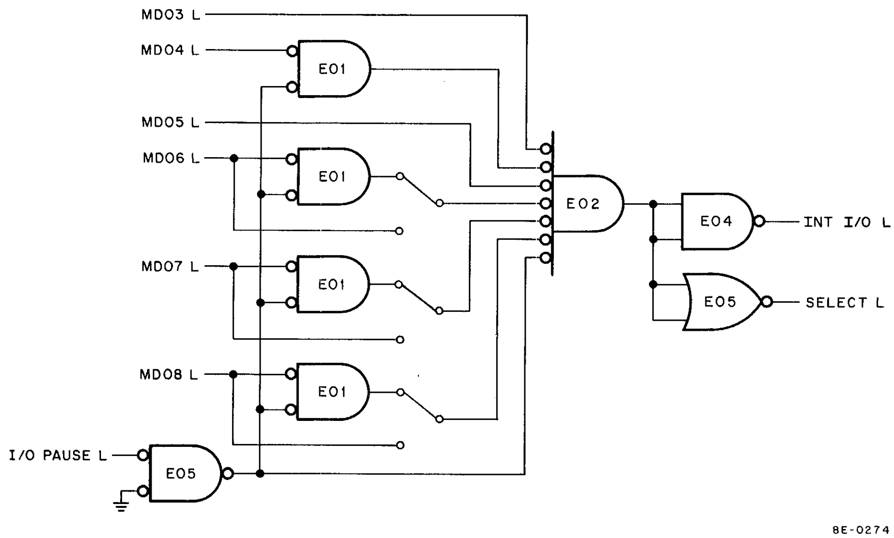


Figure 8-2 Device Select Logic

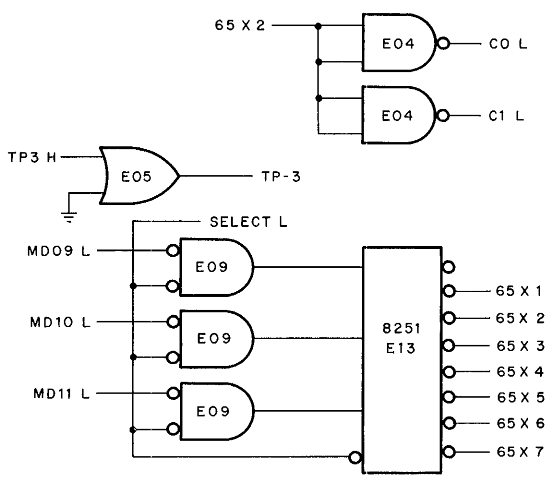


Figure 8-3 Operation Select Logic

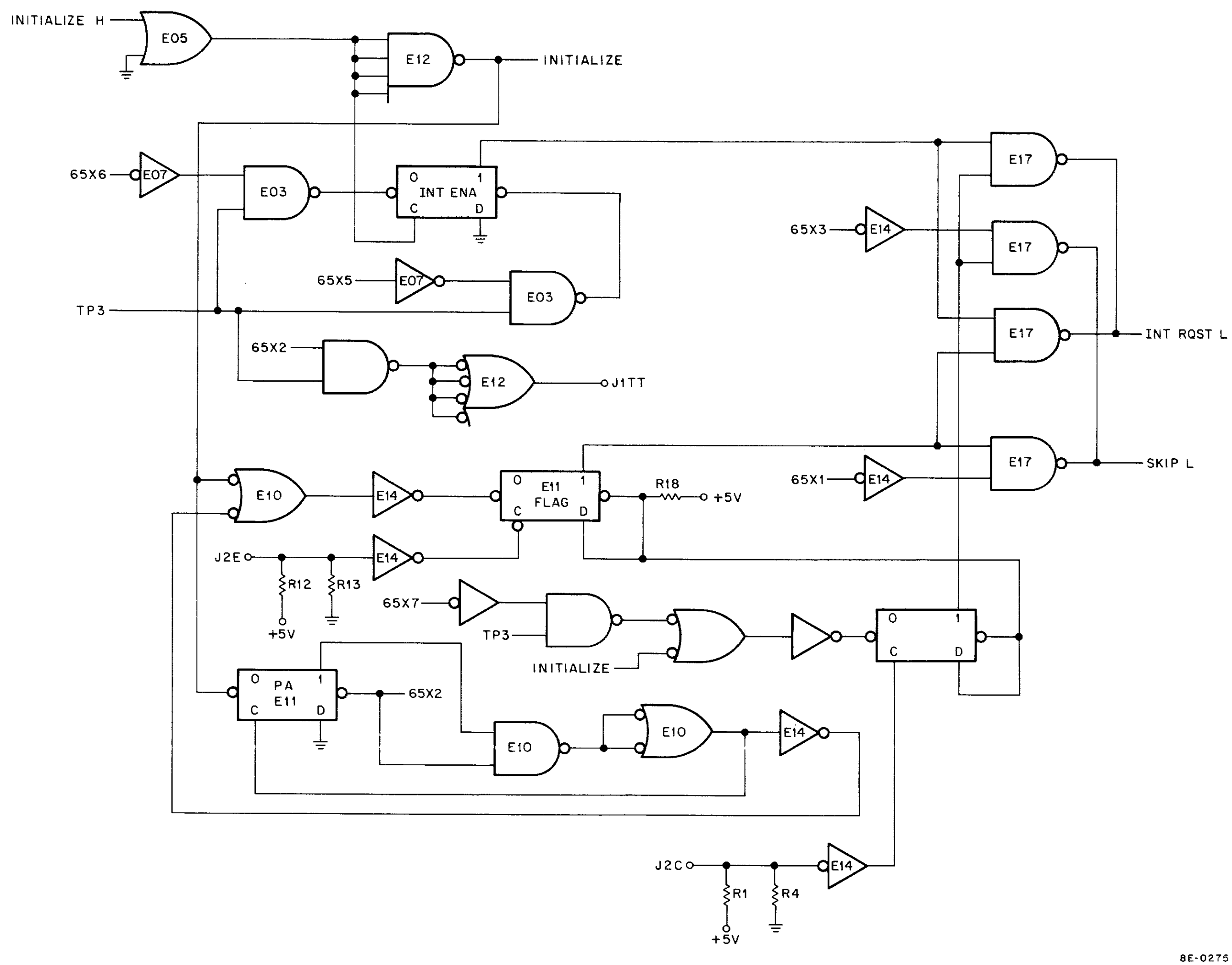


Figure 8-4 Interrupt and Skip Logic

8.6 INPUT DATA GATES AND OUTPUT BUFFERS

The input data gates (Figure 8-5) will be enabled by a 65X2 instruction and transfer data to the AC via the DATA BUS.

Table 8-2
Simultaneous Receive and Transmit Operation by
Two PDP-8/E Computers Using Interprocessor Buffer

PDP-8/E No. 1 Data Transmitter	PDP-8/E No. 2 Data Receiver
Data is transferred from AC to the DATA BUS and applied to the data inputs of buffer flip-flops by programmed instructions.	
A 6504 instruction clocks the buffer flip-flops and transmits information to PDP-8/E No. 2 and simultaneously sends a signal from J1RR of PDP-8/E No. 1 to J2E of PDP-8/E No. 2.	Data is received and FLAG is set by signal at J2E. INT RQST if ENA is set, and SKIP generated and subroutine to read data is started.
The trailing edge of the signal at J2C sets the DONE flip-flop to indicate PDP-8/E No. 2 has read data.	6502 instruction gates data to DATA BUS. TP3 and 6502 instruction enable signal to J1TT to be transmitted to J2C of PDP-8/E No. 1.
Signal is removed from J2C when 6502 signal is removed from gate in PDP-8/E No. 2.	The Flag is cleared by trailing edge of 6502 instruction.
DONE is cleared by 6507 instruction.	
PDP-8/E No. 1 is ready to transmit or receive data.	PDP-8/E No. 2 is ready to receive or transmit data.

The SELECT L signal applied to the output data gates will allow 1s to be transferred from the DATA BUS to the data input side of flip-flops in the output buffer. When a 65X4 instruction is generated, the clock input of the flip-flop will be pulsed and the data will be transferred to the input gates of PDP-8/E No. 2. This instruction will also cause J1RR to go high and set the FLAG in PDP-8/E No. 2. The data transferred from the Interprocessor Buffer is +3V for true (1) signals and 0.0V for false (0) signals.

SECTION 5 MAINTENANCE

Refer to Volume 1 for maintenance information about PDP-8/E Computers. The acceptance test given in Section 2 should be performed when an error is suspected in the DB8-E.

8.7 DATA TRANSFER TEST

Perform the following programs to transfer data from the switch register of PDP-8/E No. 1 to the AC of PDP-8/E No. 2.

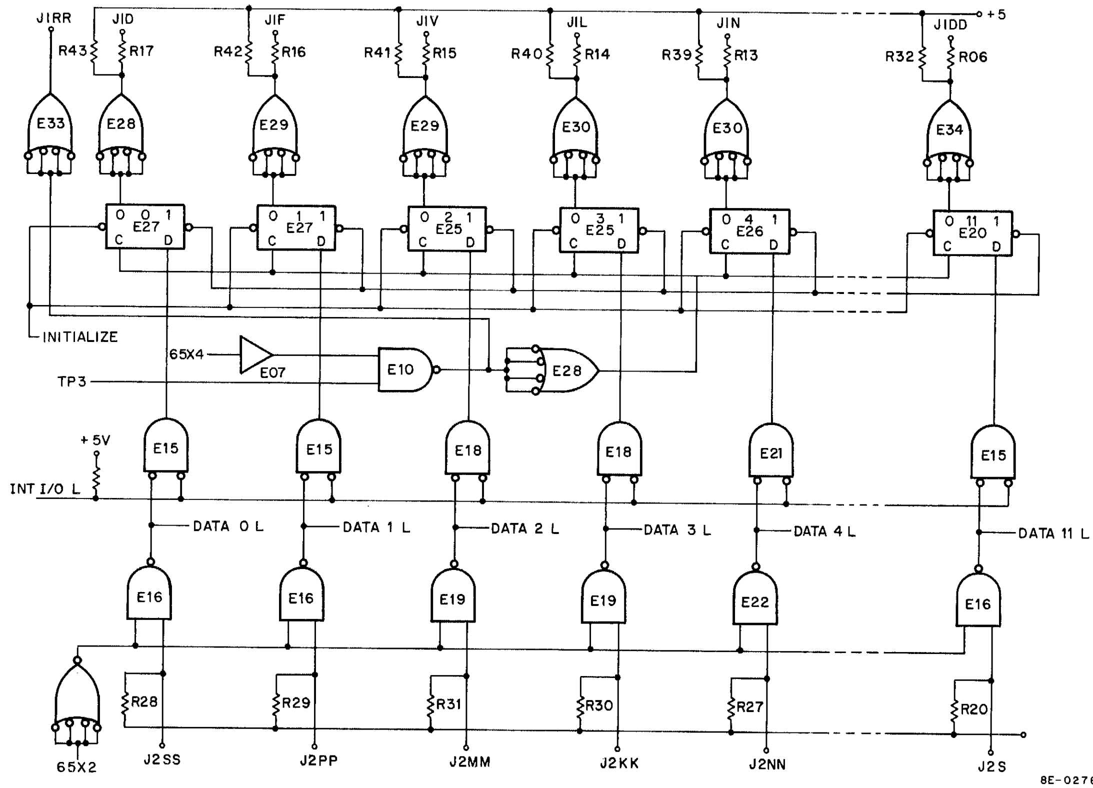


Figure 8-5 Input Data Gates and Output Data Gates and Drivers

PDP-8/E No. 1

Address	Contents
200	7604
201	6504
202	6503
203	5202
204	6507
205	5200

PDP-8/E No. 2

Address	Contents
200	6501
201	5200
202	6502
203	5200

After the programs are loaded, load address 200 and start both PDP-8/Es. The AC lights of PDP-8/E No. 2 will display contents of SR on PDP-8/E No. 1. It is recommended that the following be tried.

a. All 1s
b. All 0s
c. A single 1 in each bit position
d. A single 0 in each bit position.

SECTION 6 SPARE PARTS

Table 8-3 lists recommended spare parts for the DB8-E. These spare parts can be obtained from any local DEC office or from DEC, Maynard, Massachusetts.

Table 8-3
DB8-E Recommended Spare Parts

DEC Part No.	Description	Quantity
19-05547	IC DEC 7474	1
19-05575	IC DEC 7400	1
19-05579	IC DEC 7440	1
19-09486	IC DEC 384	1
19-09594	IC DEC 8251	1
19-09686	IC DEC 7404	1
19-09973	IC DEC 97401	1
19-09971	IC DEC 6380	1
19-09972	IC DEC 6314	1

PART 6
EXTERNAL BUS INTERFACE CONTROL OPTIONS

CHAPTER 9
KA8-E POSITIVE I/O BUS INTERFACE

SECTION 1 INTRODUCTION

The KA8-E Positive I/O Bus Interface permits use of a PDP-8/I or PDP-8/L type peripheral with the PDP-8/E. If the peripheral is a data break device, a KD8-E data break interface must also be in the system. The concept of data transfers and the interrelationship of the KA8-E Positive I/O Bus Interface, the KD8-E Data Break Interface, and the OMNIBUS are explained in Chapters 6 and 10 of the *PDP-8/E & PDP-8/M Small Computer Handbook.* A detailed discussion of CPU operation during a programmed I/O transfer is presented in Volume 1, Chapter 3, Section 6. The reader should be thoroughly familiar with this referenced information to benefit from the detailed logic discussion presented in this chapter.

SECTION 2 BLOCK DIAGRAM

Figure 9-1 is a functional block diagram of the Positive I/O Bus Interface. When an IOT instruction is placed on the OMNIBUS MD lines, the I/O PAUSE L signal is asserted by the CPU timing generator. If INTERNAL I/O L is not asserted by an internal peripheral, I/O PAUSE L causes the interface IOP timing to assert the NOT LAST TRANSFER L signal. Thus, CPU timing is suspended at TP3 time. Simultaneously, IOP timing is initiated and the IOP signal that is subsequently generated enables the BIOP pulse generator to produce one or more pulses. These pulses are used by the peripheral in conjunction with BMB bits to decode IOT instructions.

The IOT instruction can clear and set flags and registers within the peripheral, or it can direct a data word transfer or a SKIP operation. Data words are transferred between the data gating logic and the CPU via the DATA0-11 lines; between the peripheral and data gating, the data path depends on the direction of transfer, as shown in the block diagram. The OMNIBUS C lines are asserted within the data gating logic in combinations that depend on the type and direction of transfer.

If a SKIP operation is directed by the IOT instruction, the peripheral asserts the external SKIP L signal when conditions warrant. IOP timing clocks the skip counter and either the DATA10 or DATA11 line, or both, is activated.

SECTION 3 DETAILED LOGIC

9.1 BIOP PULSE GENERATOR LOGIC

Figure 9-2 shows the BIOP pulse generator logic, which converts bits MD9—11 to BIOP pulses 4, 2, and 1, respectively. A logic 1 on any MD line conditions a corresponding NAND gate for enabling at TP2 time. When the NAND gate is enabled, it dc-sets a flip-flop that, in turn, conditions another NAND gate. Simultaneously, the

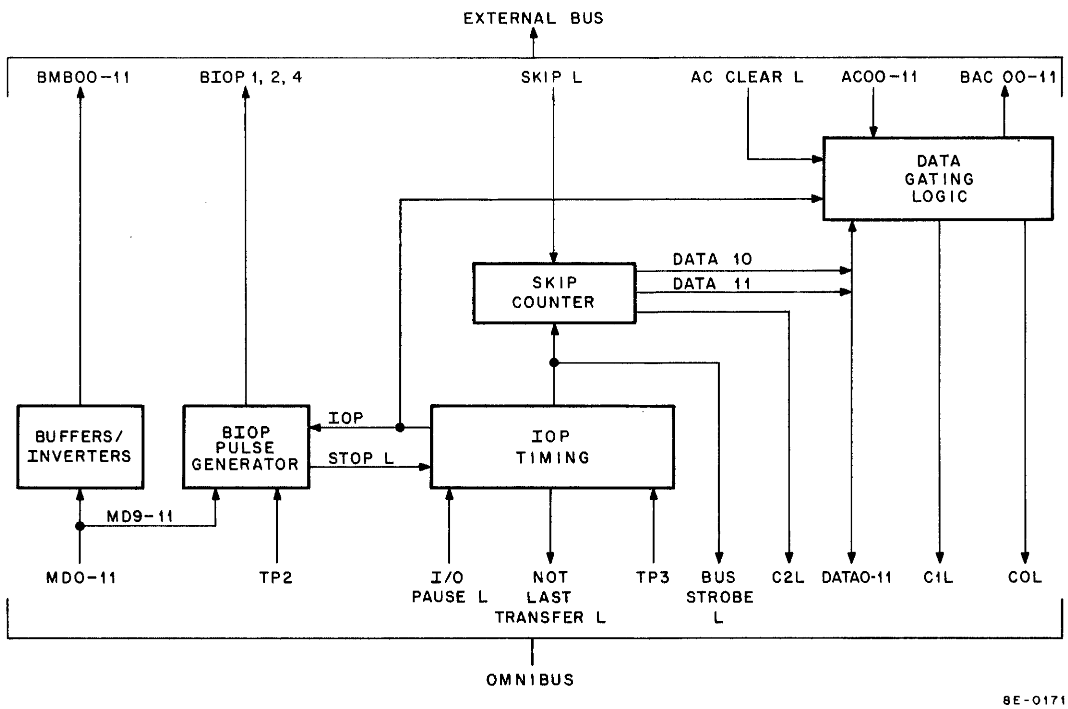


Figure 9-1 Positive I/O Bus Interface, Block Diagram

flip-flop causes the STOP L signal to be negated. If an I/O transfer involving an external bus peripheral is in progress, the STOP L signal enables TP3 to initiate the IOP timing operation. The IOP timing logic (Paragraph 9.3) responds by asserting a signal (IOP) that enables the flip-flop conditioned NAND gate. The resulting signal is buffered and designated BIOP4, BIOP2, or BIOP1.

For example, if MD bit 9 is a logic 1, flip-flop IO4 is dc-set at TP2 time (note that the IO flip-flops are cleared at each TP1 time). The 0-output of the flip-flop causes STOP L to be asserted and, providing the IO1 and IO2 flip-flops are cleared (MD bits 10 and 11 are logic 0), the 1-output conditions NAND gate E24 for enabling by the IOP signal. When the IOP timing asserts the IOP signal, E24 is enabled and the BIOP4 pulse is generated. The width of the pulse can be varied by adjusting a potentiometer in the IOP timing logic, thereby asserting the IOP signal for the desired amount of time (Paragraph 9.3). When the IOP signal is negated, E24 is disabled. Because the output of E24 is connected to the clock input of IO4, the flip-flop is cleared when E24 is disabled (note that the D inputs of the IO flip-flops are connected to ground; thus, a positive transition at a clock input clears the flip-flop). The 0-output of IO4 negates the STOP L signal; this action causes the IOP timing logic to terminate the I/O dialogue.

This example stipulated that bits MD10 and MD11 were logic 0. Suppose, instead, that all three MD bits are logic 1. All three IO flip-flops are then set at TP2. The STOP L signal is asserted and the 1-output of IO1 conditions E11C for enabling by the IOP signal. Observe that the 0-output of IO1 disables both NAND gate E8B and NAND gate E24. Thus, the IOP signal enables E11C first and the BIOP1 pulse is generated. When the IOP signal is negated, E11C is disabled and IO1 is cleared. This action removes the disabling signal from E8B; however, E24 remains disabled because the 0-output of IO2 is one of its inputs. NAND gate E8B is now conditioned for enabling by the IOP signal. When this signal is again asserted by the IOP timing logic, BIOP2 is generated. When BIOP2 ends, IO2 is cleared, removing the disabling signal from E24. Now the BIOP4 pulse is generated, as detailed earlier.

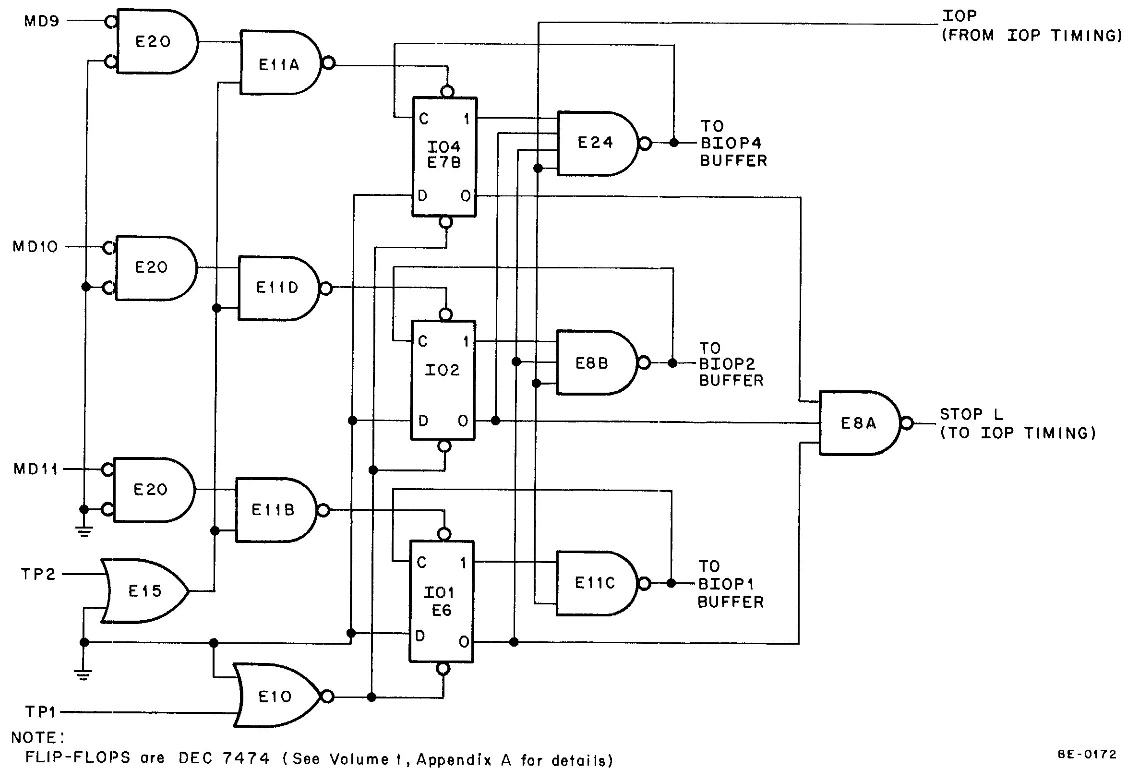


Figure 9-2 BIOP Pulse Generator Logic

Thus, the BIOP Pulse Generator logic operates in such a way that the BIOP pulses are not assigned specific time slots. If only one of the MD bits is a logic 1, then the corresponding BIOP pulse, whether 4, 2, or 1, is generated at the first assertion of the IOP signal. If more than one MD bit is logic 1, the least-significant bit is selected first and its corresponding BIOP pulse is generated; the most-significant bit is selected last.

9.2 BMB BUFFERS/INVERTERS

The preceding paragraphs discussed the BIOP pulse generator. A peripheral uses these BIOP pulses in conjunction with BMB bits to decode IOT instructions. The BMB bits are derived from the OMNIBUS MD bits, which are buffered and inverted by the interface. Figure 9-3 shows the buffer/inverter and network.

BMB bits 03–08 are used in peripheral device selection logic. Both the true and the false states of these bits are derived from the corresponding MD bit; this minimizes the device selection network in the external peripheral. Although programmed transfer peripherals use the BMB bits only for device selection, data break peripherals receive output (from the CPU) data via the BMB00–11 lines. Thus, all 12 BMB bits are derived, as shown in Figure 9-3.

9.3 IOP TIMING LOGIC

Figure 9-4 shows the IOP timing logic, which determines the duration of BIOP pulses and the separation between individual pulses, if more than one is programmed. Separation and duration can be varied individually by potentiometers that are indicated on the logic diagram and on the KA8-E etch as IOP SEP and IOP WIDTH. These

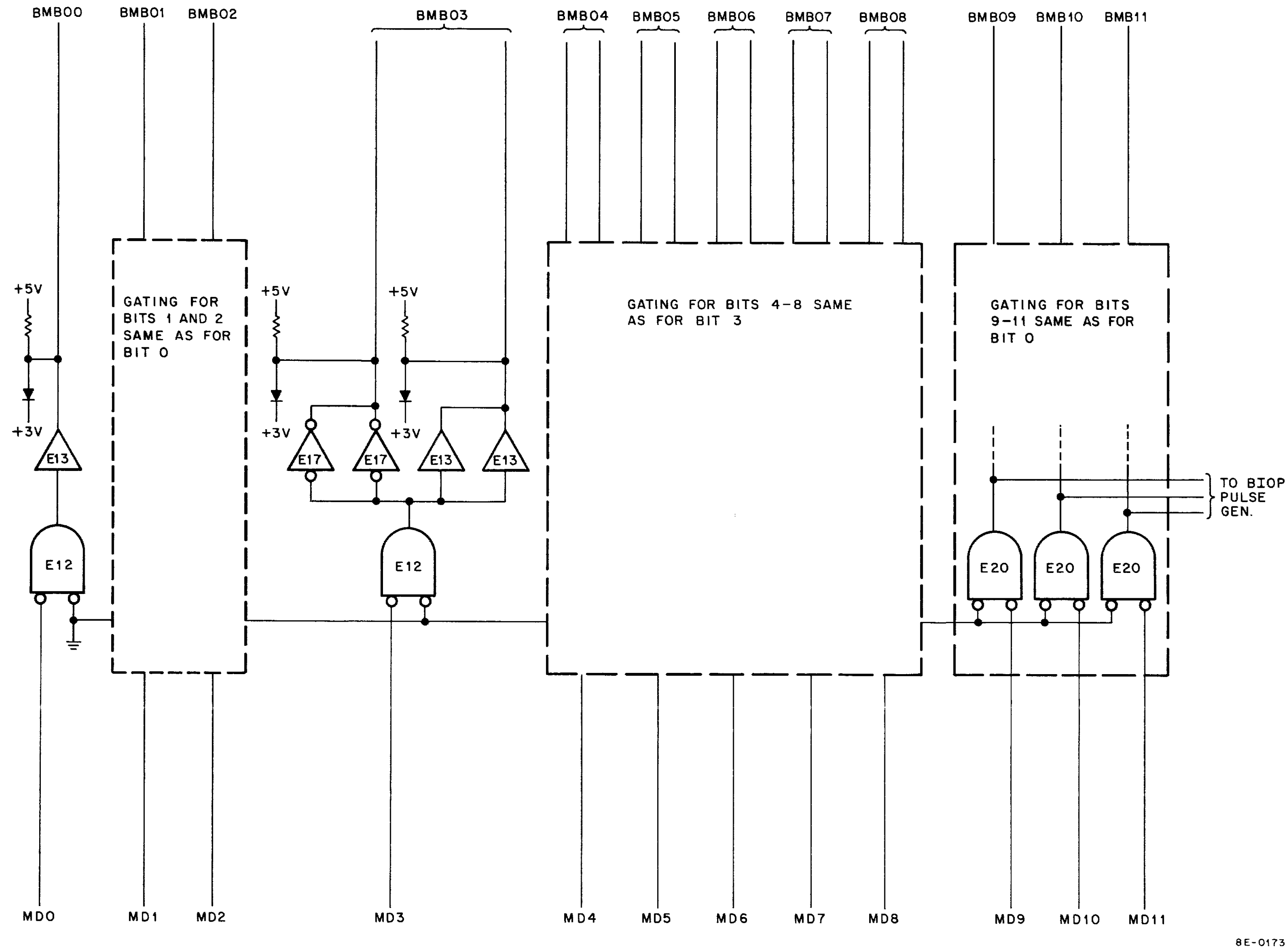


Figure 9-3 BMB Buffers/Inverters

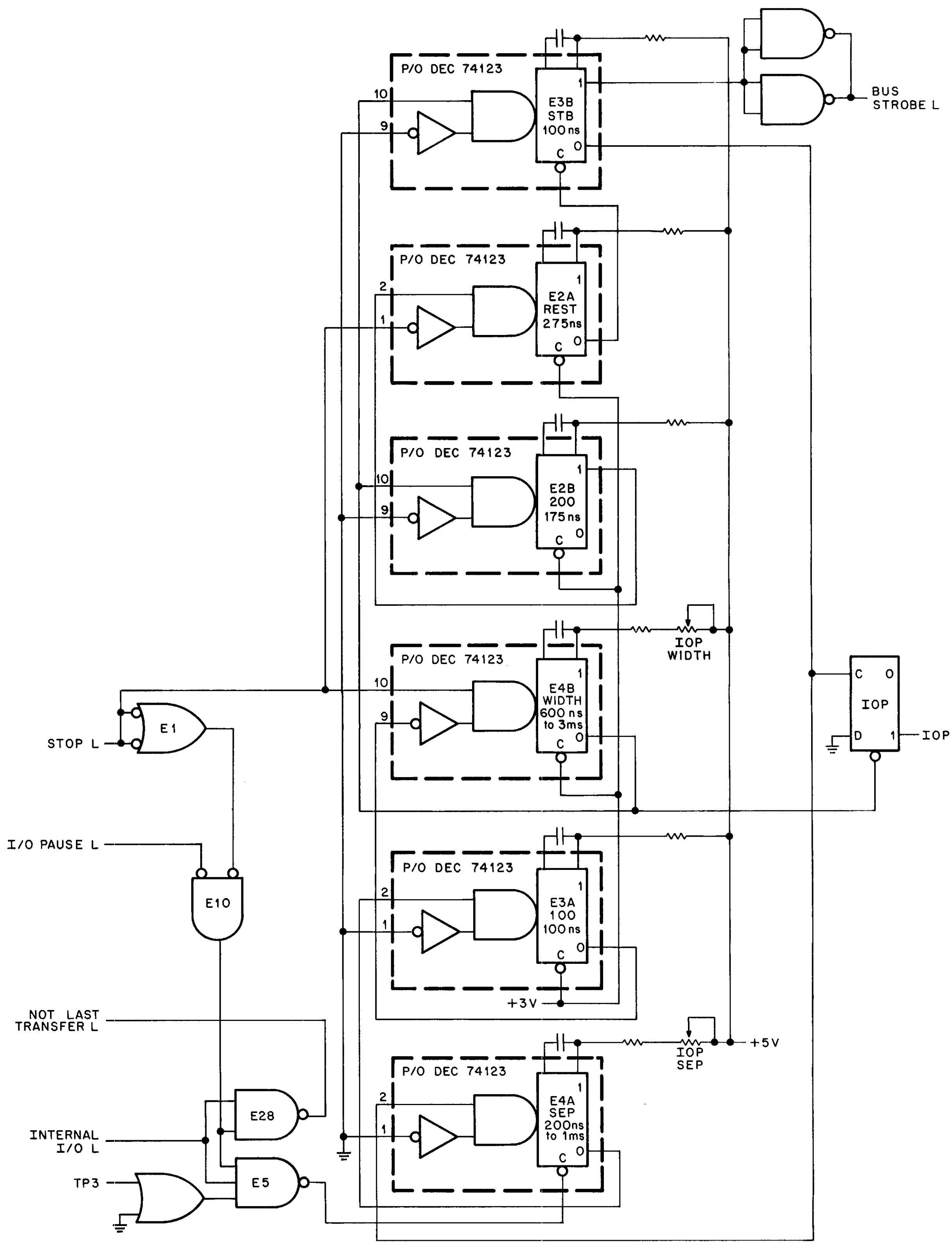


Figure 9-4 IOP Timing Logic

potentiometers determine the triggered delay time of associated one-shot multivibrators, shown as part of DEC 74123 ICs. Briefly, the one-shots contained within a 74123 can be triggered by:

a. a positive transition at pin 2, if, prior to the transition, pin 1 is low and the clear (C) input is high (for the "B" half of the IC, substitute pin 10 and pin 9 for pin 2 and pin 1, respectively),

b. a negative transition at pin 1, if, prior to the transition, pin 2 is high and C is high,

c. a positive transition at the C input, if, prior to the transition, pin 1 is low and pin 2 is high.

Figure 9-5 shows the IOP timing for a typical I/O transfer. The IOP signal is asserted twice during the time that I/O PAUSE is active; thus, two BIOP pulses are generated by the BIOP pulse generator (the identity of the BIOP pulses does not affect the waveform relationship). The waveforms representing SEP and WIDTH are shown for the minimum allowable triggered delay time, viz., 200 ns for SEP and 600 ns for WIDTH. The potentiometer values allow these delay times to be increased to five times the minimum value. Refer to both figures while studying the following description.

If the I/O transfer involves an external bus peripheral (INTERNAL I/O L remains negated) and the BIOP pulse generator negates the STOP L signal, and NAND gate E28 asserts the NOT LAST TRANSFER L signal; this signal indicates to the CPU timing generator the impending interruption of normal timing. At TP3 time, IOP timing is initiated, while CPU timing is suspended in TS3.

IOP timing begins when the SEP one-shot is triggered by the positive transition at its C input. SEP (0) is used to trigger the 100 one-shot, which, in turn, triggers WIDTH. WIDTH (0) sets the IOP flip-flop; the resulting IOP signal enables the BIOP pulse generator to begin the BIOP pulse. The STB (Strobe) one-shot, triggered by the end of WIDTH (0), clears the IOP flip-flop; thus, the duration of the BIOP pulse is 100 ns longer than the duration of WIDTH (0). A BUS STROBE L signal is generated by STB (1) at the end of each BIOP pulse to execute the instruction represented by the BIOP pulse (BUS STROBE L causes the AC LOAD L signal to be asserted in the CPU; see Volume 1, Chapter 3, Section 6 for details). When the first BIOP pulse has ended, the sequence outlined begins again, this time with STB (0) triggering the SEP one-shot. When the last BIOP pulse ends, the STOP L signal is asserted by the BIOP pulse generator. This signal ensures that the WIDTH one-shot is not triggered by the next transition of 100 (0). Therefore, the IOP flip-flop remains clear through the remainder of the IOP timing. When REST (0) (the Restart one-shot), which is triggered by the STOP L signal transition, goes positive after 275 ns, STB is triggered again, producing a final BUS STROBE L signal. This BUS STROBE terminates I/O dialogue and reinstates CPU timing.

The 100 one-shot is necessary for proper triggering of WIDTH. It provides a negative transition at pin 9 of WIDTH when SEP times out. If the negative transition were supplied by the 0-output of SEP, WIDTH would trigger at the same time as SEP. On the other hand, if the negative transition were supplied by the 1-output of SEP, WIDTH would trigger at TP2 time, when the STOP L signal is negated. Consequently, the 100 one-shot is quite important to the timing operation.

The 200 one-shot is also important to the timing logic. Note on Figure 9-5 that the STOP L signal is shown to have a spike that is coincident with the trailing edge of the first BUS STROBE signal (if three BIOP pulses were generated, there would be two spikes shown). This spike is a representation of the tendency of the STOP L signal to go low at the end of the IOP signal (NAND gate E8A in the BIOP pulse generator becomes momentarily indecisive at this point in the timing). The 200 one-shot brackets this spike in time and, thus, prevents the REST one-shot from triggering prematurely.

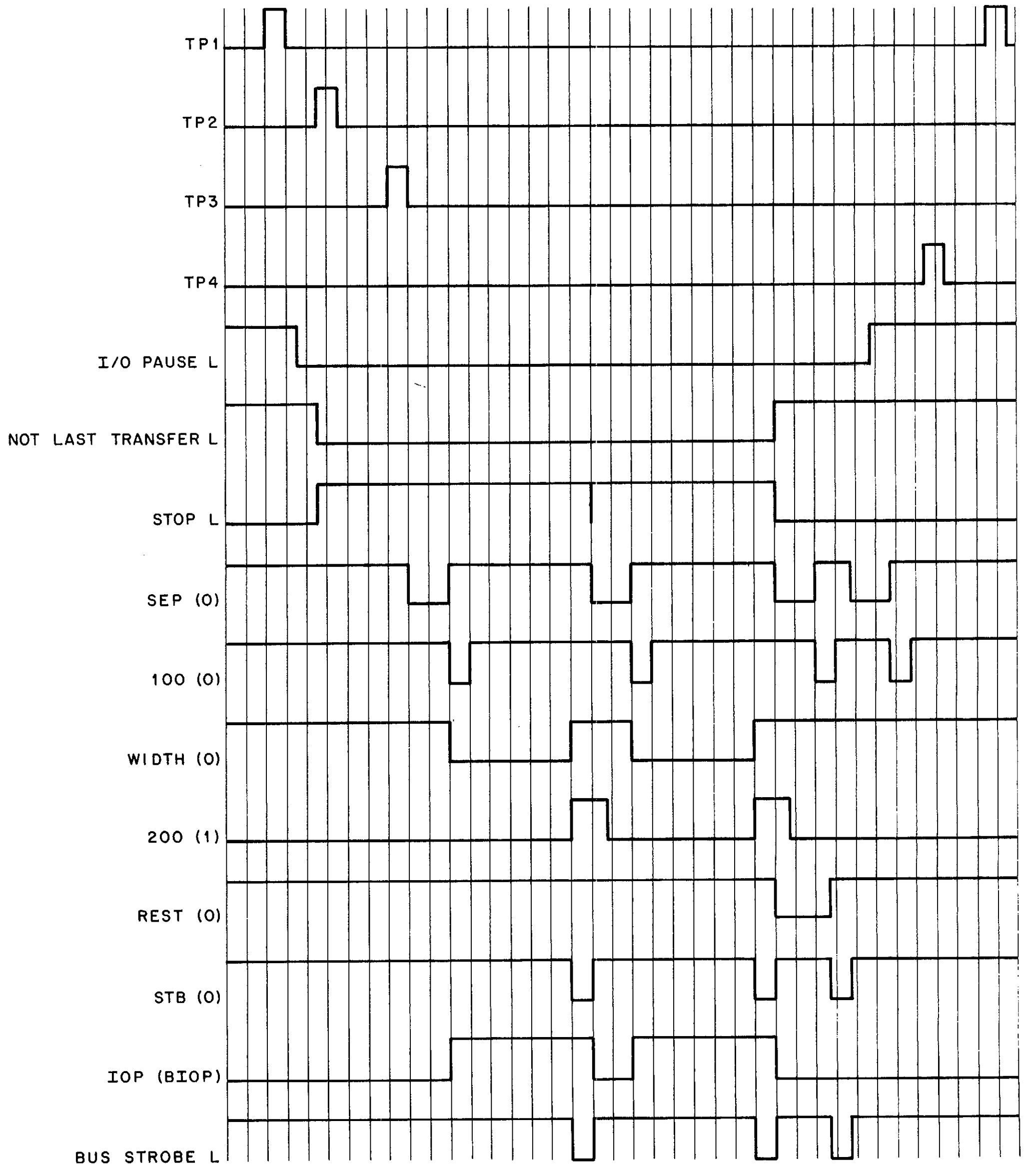


Figure 9-5 Waveforms, IOP Timing Logic

9.4 DATA GATING LOGIC

The data gating logic is shown in Figure 9-6. During a programmed I/O transfer, data is transferred to or from the CPU on the OMNIBUS DATA0–11 lines. Output data (from the CPU) is gated from the DATA lines through an interface buffer/inverter network (illustrated in Figure 9-6 for bits 0 and 11) to the external bus BAC00–11 lines. If the output transfer is to be accompanied by a clearing of the CPU AC Register, the peripheral is directed (by the BIOP pulse) to assert the OMNIBUS C0 L signal. The peripheral does this indirectly by grounding the external bus AC CLEAR line. The AC CLEAR L signal causes NAND gate E25D on the interface to assert the C0 L signal. If the AC Register does not have to be cleared, the C-lines remain negated high throughout the transfer.

On the other hand, an input transfer must always be accompanied by the assertion of at least one C-line. Note that when a data word is transferred from the peripheral on the external bus AC00–11 lines, the interface DATA IN L signal is asserted by NOR gate E29. If the data is placed on the AC lines during the BIOP pulse (as it must be), NAND gate E25C asserts the C1 L signal. Simultaneously, the AC INPUTS → DATA BUS signal gates the data onto the OMNIBUS DATA0–11 lines. The result of these actions is an OR operation of the AC contents and the data on the DATA0–11 lines. The peripheral can cause a jam input by grounding the AC CLEAR line, thereby asserting the C0 L signal. Thus, only the information on the DATA0–11 lines is placed in the AC Register. Note that if data word 0000_8 is transferred from the peripheral, the DATA IN L signal is not asserted. To transfer 0000_8, the peripheral must ground the AC CLEAR line and, in effect, cause a jam input of zeros.

These are the four types of data transfers that can be made by the PDP-8/I type peripherals. Another form of I/O transfer, other than a 12-bit data word, utilizes the interface skip logic to update the CPU PC Register. This type of transfer is discussed in Paragraph 9.5.

9.5 SKIP COUNTER LOGIC

The peripheral, when directed by an IOT instruction, can cause a skip of 1, 2, or 3 program instructions. It initiates a SKIP operation by grounding the external bus SKIP line. The interface skip logic, shown in Figure 9-7, asserts the OMNIBUS DATA10 and/or DATA11 lines, depending on the number of instruction skips required (during a SKIP operation the peripheral does not place data on the AC00–11 lines). At the same time, the OMNIBUS C lines are manipulated to provide a path for the DATA bits through the CPU major register gating to the PC Register. The DATA bits are added to the contents of the PC Register, increasing the program count in the register by 1, 2, or 3.

The timing diagram of a typical SKIP operation is shown in Figure 9-8. Refer to this diagram and to Figure 9-7 while studying the description that follows. The timing diagram shows that two BIOP pulses, 1 and 2, are generated during the IOT instruction. The imaginary peripheral that applies to this example decodes these two BIOP pulses and responds by grounding the SKIP line. Keep in mind that this is an example, only, and that this imaginary peripheral does not necessarily exist. The combination of BIOP1 and BIOP2 can produce a variety of operations, depending on how a peripheral decodes the pulses.

Nevertheless, when TP3 starts IOP timing, this peripheral is directed to initiate a SKIP operation by asserting SKIP L. The SKIP line controls the D-input of flip-flop E9, which is clocked when the STB one-shot is triggered. Because STB is triggered at the end of each BIOP pulse, E9 can be clocked twice during IOP timing. Note that the 100 one-shot dc-sets E9. Thus, E9 is set at the first triggering of the 100 one-shot, or (as shown in Figure 9-8) during a previous IOP timing cycle. Thereafter, 100 and STB are griggered alternately. Thus, E9 is alternately cleared and set, as long as SKIP L is asserted. Each time E9 is cleared, SKIP 1, the first stage of the 2-stage binary counter, is clocked. In this example, the binary counter is clocked twice, indicating that two instructions are to

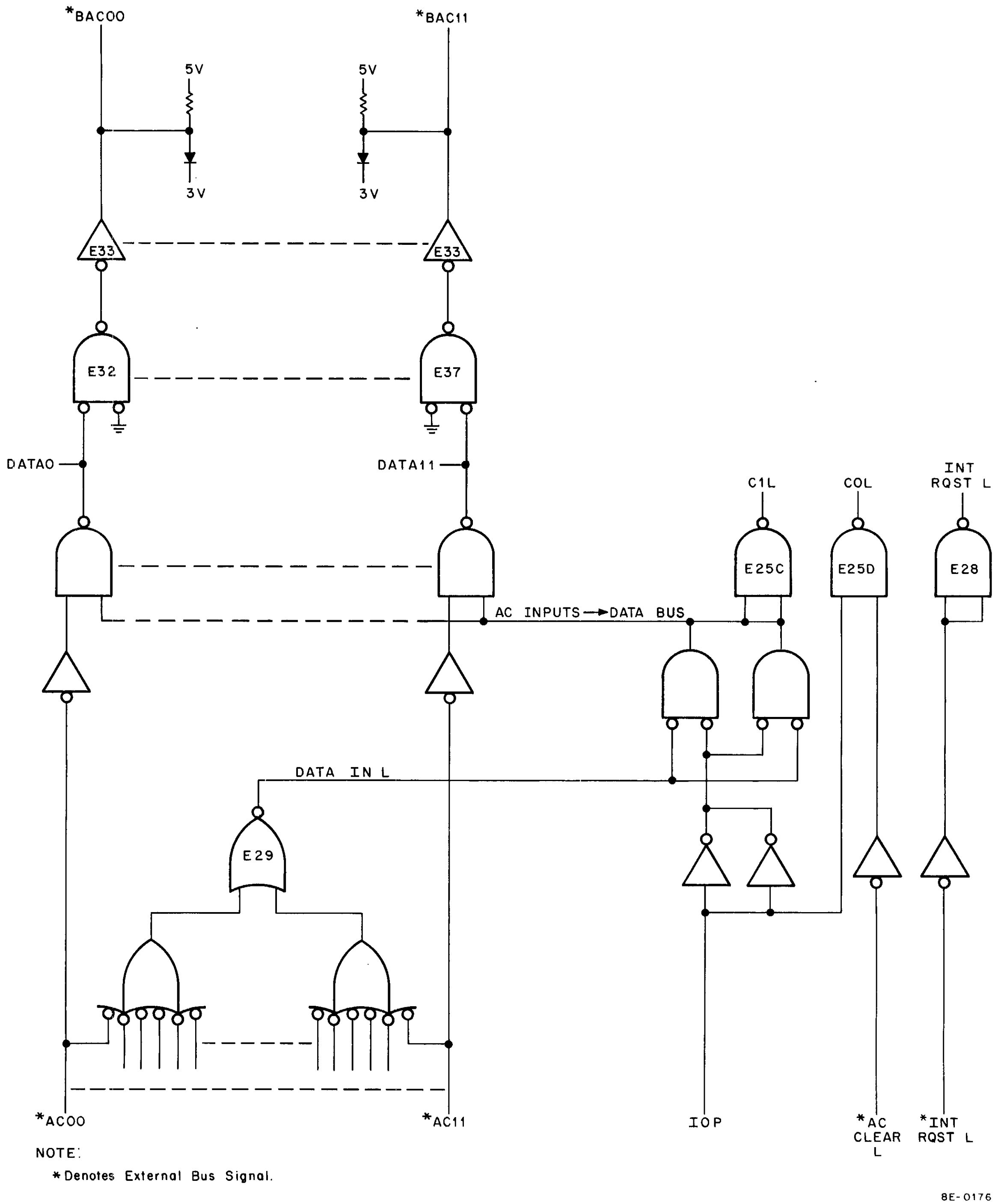


Figure 9-6 Data Gating Logic

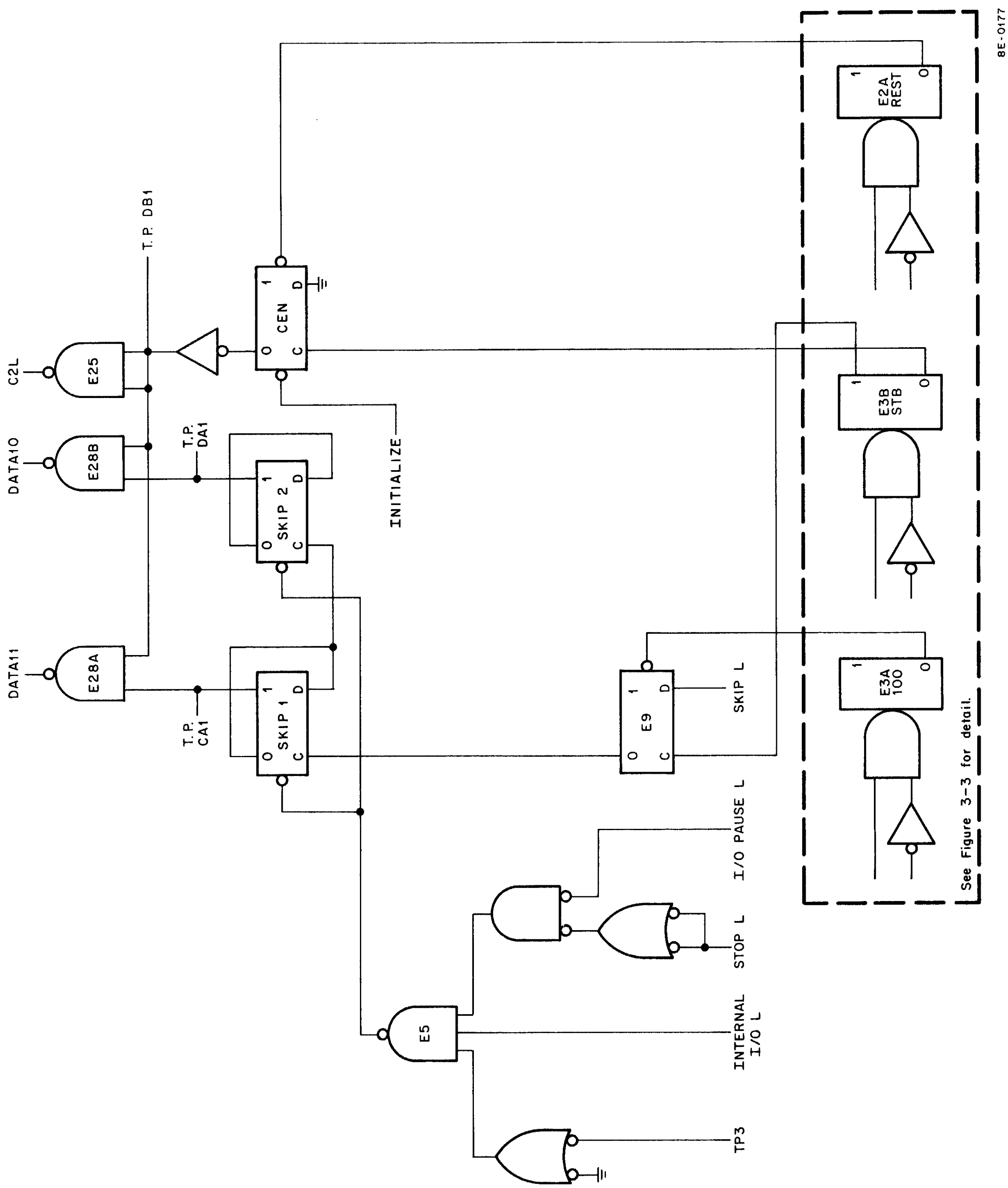


Figure 9-7 Skip Logic

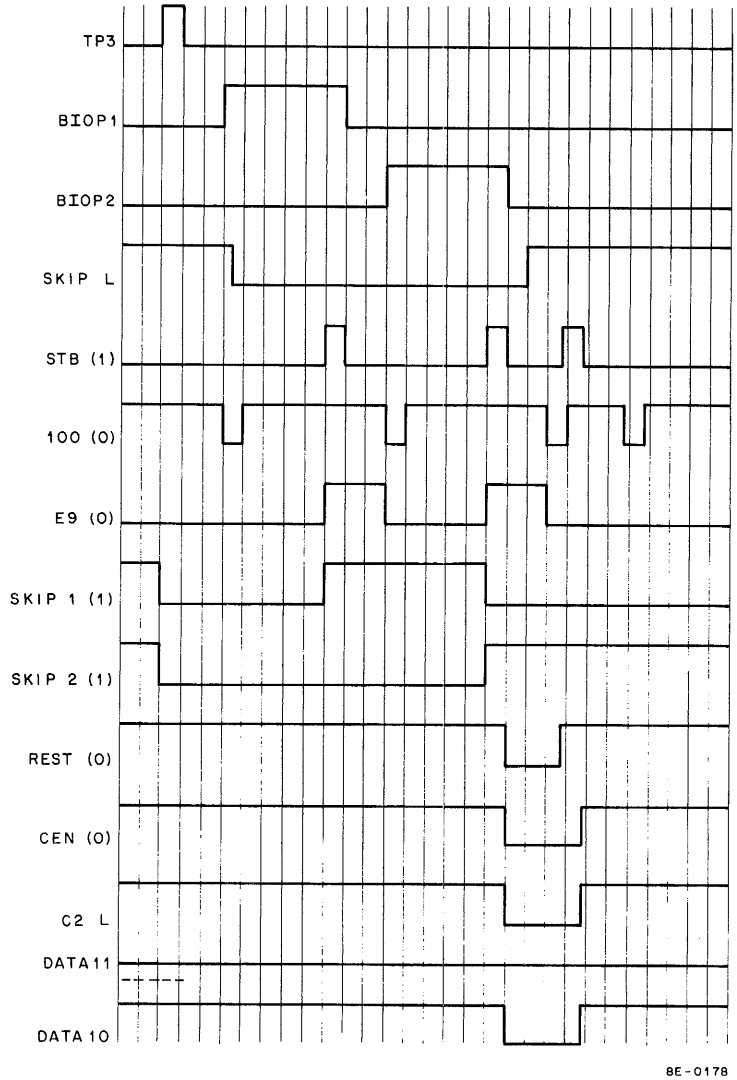


Figure 9-8 Timing, Skip Logic Application

to be skipped. SKIP 2 is set at the second triggering of STB. The C-line enable (CEN) flip-flop is dc-set by the REST one-shot, enabling NAND gates E25 and E28B (note that CEN is clocked by STB at the same time that it is dc-set by REST; the dc-input takes precedence in such a case). The assertion of C2 L, while C1 L and C0 L remain negated, provides a path for DATA10 through major register gating to the PC Register. C2 L and BUS STROBE (generated when REST times out) assert PC LOAD L and DATA10 is added to the contents of the PC Register, updating it by two. Note that the SKIP line must be negated before STB is triggered by the trailing edge of REST (0). If not, the binary counter is erroneously clocked one more time.

SECTION 4 MAINTENANCE

There are no specific maintenance procedures for the KA8-E itself. Each DEC peripheral that connects to the KA8-E has an associated MAINDEC or exerciser program that enables the technician to maintain both the option and the KA8-E interface. Because all of these peripherals use the one interface, a fault in the interface can be isolated by running a number of MAINDEC programs. If all programs result in errors, one can reasonably conclude that the KA8-E is at fault.

General information concerning corrective maintenance is included in Volume 1, Chapter 4. The technician will find this material helpful. The interface schematic, E-CS-M8350, indicates important test points, IC locations, and pin numbers and should be used whenever maintenance is being performed.

The KA8-E connects directly to a single peripheral via three cables that are supplied with the interface (refer to the *PDP-8/E & PDP-8/M Small Computer Handbook,* Chapter 10, for cabling rules and suggestions). Each cable connects to the interface with a 40-pin Berg connector and to the peripheral with a DEC M953A cable connector. From-To information for the cable is given in Table 9-1 (the cables are identical). (Refer to the *PDP-8/E & PDP-8/M Small Computer Handbook* for details concerning cable connections.)

Table 9-1
KA8-E Cable Information

From (M953A Cable Conn.)	To (Berg Conn.)	From (M953A Cable Conn.)	To (Berg Conn.)
Gnd	A	Gnd	Y
Gnd	B	M2	Z
Gnd	C	Gnd	AA
B1	D	L1	BB
Gnd	E	Gnd	CC
D2	F	P2	DD
Gnd	H	Gnd	EE
D1	J	M1	FF
Gnd	K	Gnd	HH
E2	L	S2	JJ
Gnd	M	Gnd	KK
E1	N	P1	LL
Gnd	P	Gnd	MM
H2	R	T2	NN
Gnd	S	Gnd	PP
H1	T	S1	RR
Gnd	U	Gnd	SS
K2	V	V2	TT
Gnd	W	Gnd	UU
J1	X	Gnd	VV

Pins A2, B2, U1, and V1 on M953A not used.
Pins A1, C1, F1, K1, N1, R1, T1, C2, F2, J2, L2, N2, R2, and U2 on M953A are ground pins.

SECTION 5 SPARE PARTS

Table 9-2 lists recommended spare parts for the KA8-E. These spare parts can be obtained from any local DEC office or from DEC, Maynard, Massachusetts.

Table 9-2
KA8-E Recommended Spare Parts

DEC Part No.	Description	Quantity
15-03100	Transistor, DEC 3009B	1
19-09705	IC DEC 8881	1
19-10010	IC DEC 2501	1
19-09971	IC DEC 6380	1
19-09921	IC DEC 7417	1
19-09928	IC DEC 7416	1
19-09686	IC DEC 7404	1
19-09373	IC DEC 9601 (M835 only)	1
19-09486	IC DEC 384	1
19-09004	IC DEC 7402	1
19-05578	IC DEC 7430	1
19-05577	IC DEC 7420	1
19-05576	IC DEC 7410	1
19-05575	IC DEC 7400	1
19-05547	IC DEC 7474	1
11-00114	Diode D664	4
11-00113	Diode D662	2
BC08J-10	Cable, 10 ft.	1
	IC DEC 74123 (M8350 only)	1

CHAPTER 10
KD8-E DATA BREAK INTERFACE

SECTION 1 INTRODUCTION

The KD8-E Data Break Interface is used by peripherals to transfer large blocks of data between the peripheral and memory. This interface cannot provide all the necessary signals for such a data transfer. Consequently, the positive I/O bus interface must also be used in the system. The concept of data transfers and the interrelationship of the data break interface, the positive I/O bus interface, and the OMNIBUS are explained in Chapters 6 and 10 of the *PDP-8/E & PDP-8/M Small Computer Handbook,* DEC 1972. A detailed discussion of CPU operation during a data break transfer is presented in Volume 1, Chapter 3, Section 6. The reader should be thoroughly familiar with this referenced information to benefit from the detailed logic discussion presented in Section 3.

SECTION 2 BLOCK DIAGRAM

Figure 10-1 is a functional block diagram of the KD8-E Data Break Interface. OMNIBUS signals are indicated by asterisks (on the block diagram only). When an interface receives a BRK RQST L signal from its peripheral, the request accept logic uses the next INT STROBE to assert signals that indicate acceptance of the request. These signals are used by the CPU and the peripheral in preliminary operations and by the interface priority network. The priority network compares the priority ranking of a peripheral with that of all other peripherals that make a break request at the same time. The interface of the highest ranking peripheral generates a PRIORITY signal that allows its CP Control logic to assert CP control lines. This action enables the peripheral to assume control of the CP Major Register gating and to directly address, via the BKMA Register logic, memory locations associated with the data transfer. The direction of transfer and the type of transfer are controlled by the interface data transfer logic. When the cycle select logic indicates that the true BREAK (BK) cycle is in progress, the data word is transferred to, or from, the address indicated by the BKMA Register logic. If the data transfer is from the peripheral, the data word is placed on the DATA 0–11 lines of the OMNIBUS by the data transfer logic.

SECTION 3 DETAILED LOGIC

10.1 CP REGISTER CONTROL LOGIC

Figure 10-2 shows the CP Register control logic. The NBR flip-flop determines if the interface can assert the CP Register control lines in response to a break request from the peripheral. This determination is based on the state of the NEW BRK OK L signal, which is asserted within the interface when conditions allow a data transfer (the NEW BRK OK L signal is discussed fully in Paragraph 10.2). If NEW BRK OK L has been asserted by the interface and the BRK RQST L signal has been asserted by the peripheral, the NBR flip-flop is set by the INT STROBE

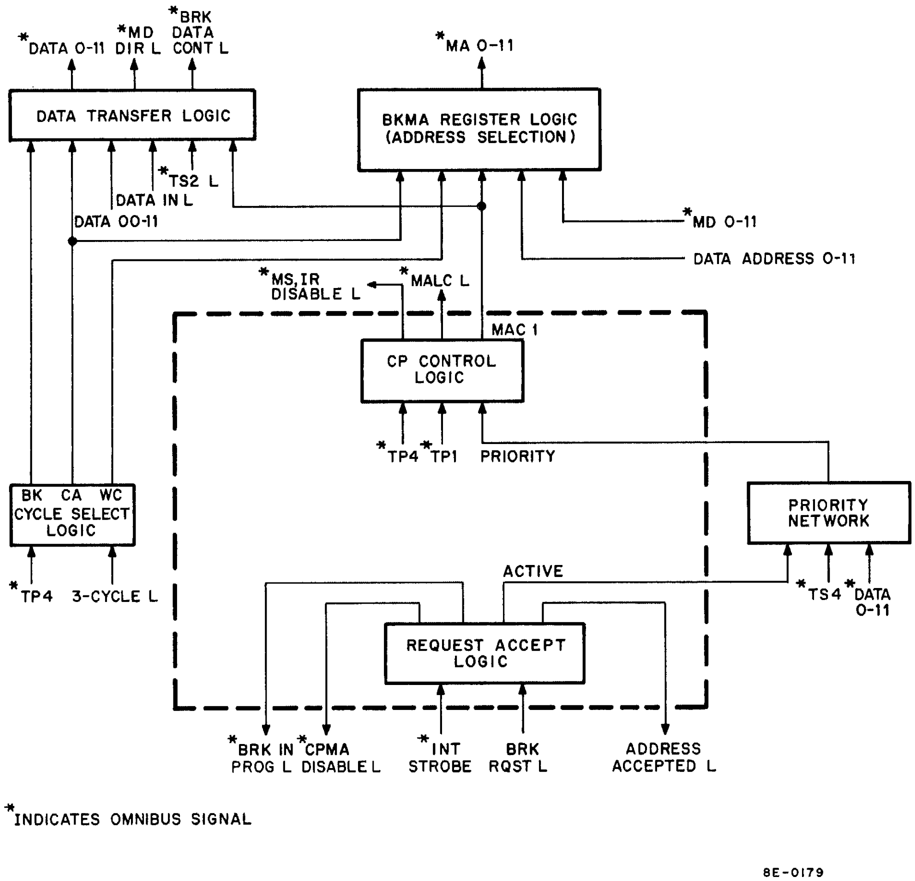


Figure 10-1 Data Break Interface, Block Diagram

signal (Figure 10-2). The 0-output of NBR then sets the ACTIVE flip-flop, which asserts the BRK IN PROG L and CPMA DISABLE L signals. The CPMA DISABLE L signal conditions a flip-flop on the Major Register Control module (Volume 1, Figure 3-102) so that TP4 can clear the flip-flop; the resulting signal, MAC L, removes the CPMA Register outputs from the MA lines. The BRK IN PROG L signal, which can be displayed on the programmer's console, ensures that only data break devices place priority information on the DATA lines during TS4.

When TS4 is entered, the 1-output of the ACTIVE flip-flop asserts the priority signal for this (*our*) interface. If other peripherals have made break requests at INT STROBE time, each peripheral's interface asserts a priority signal at TS4. The priority network (Paragraph 10.3) in each active interface examines these signals to determine if it has the highest priority of all the devices currently attempting to use the data break system. If our peripheral is not of sufficiently high priority, it must wait until the next TS4 signal; at that time the priority signals are again compared. If *our* peripheral has highest priority, the D-inputs of the MAC1 and MAC2 flip-flops are taken to a positive voltage; the flip-flops are then set at TP4. The 1-output of MAC1 not only asserts the MS, IR DISABLE L signal, which places the CP in the DMA state, but also conditions the MALC flip-flop so that it can be set when the TP1 pulse occurs. The 1-output of MALC then asserts the MALC L signal, which ensures that the CPMS Register and the CPMA Register will resume normal operation in the correct major state and at the correct memory address, respectively.

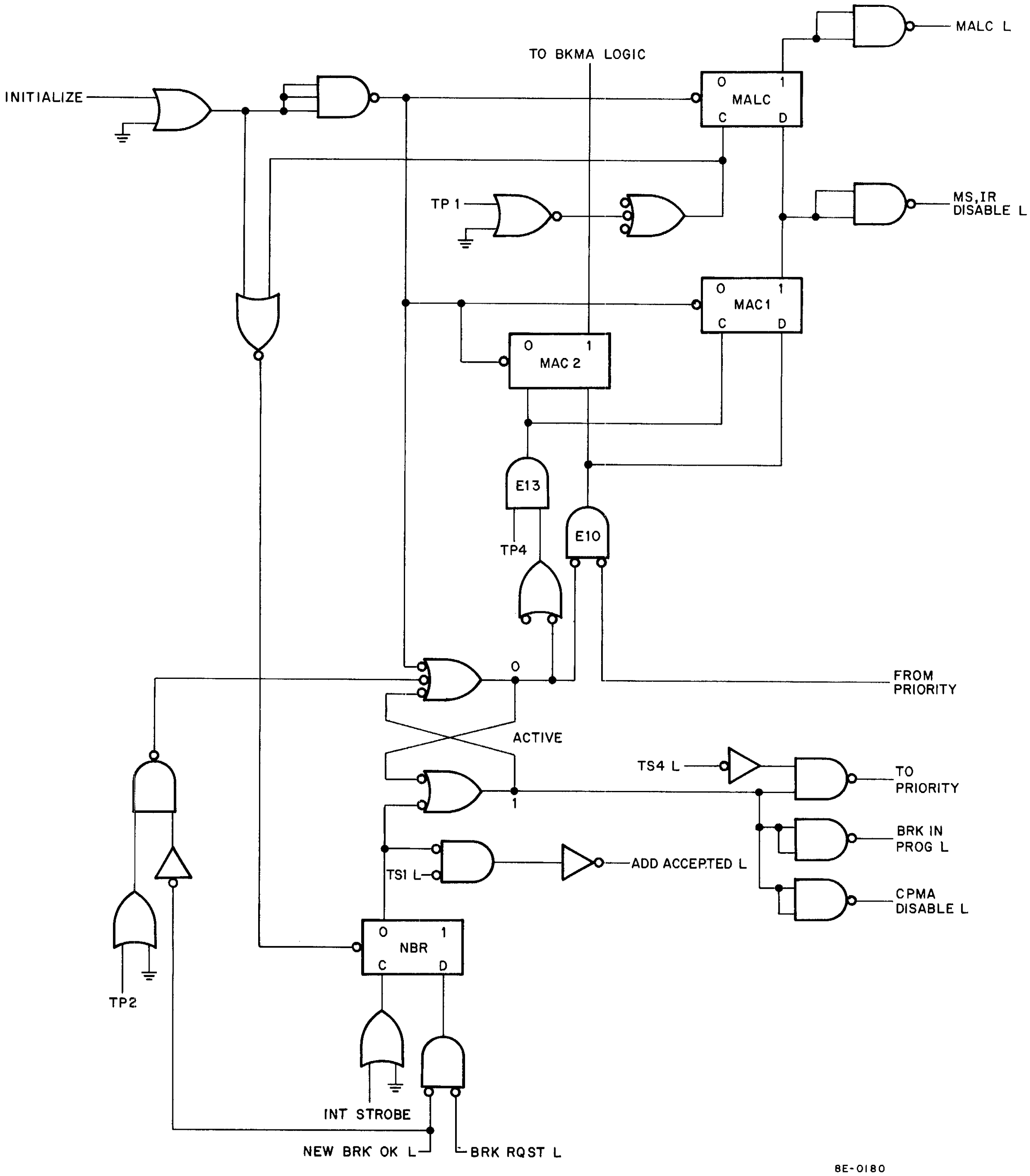


Figure 10-2 CP Register Control Logic

Note that ADD ACCEPTED L is asserted during TS1 of the first cycle following INT STROBE, regardless of the outcome of the priority check. This signal clears the break request flip-flop in the peripheral, allowing the peripheral to make another request when it is ready. At TP1, the NBR flip-flop is cleared, also regardless of what occurs in the priority network. Thus, this flip-flop is active for only the short time necessary to indicate acceptance of the break request. In contrast, the ACTIVE flip-flop is cleared only at TP2 of a true break cycle (defined and explained in Paragraph 10.2), which can be delayed for some time by priority considerations. Both NBR and ACTIVE are used extensively as control signals in other functional sections of the interface.

10.2 CYCLE SELECT LOGIC

After *our* interface takes control of the CP, a data transfer can be made. The CP is in the DMA state and remains in this state as long as MS, IR DISABLE L remains low. Each timing cycle (a "slow" cycle of 1.4 μs) that occurs during the DMA state is used by the interface/peripheral to accomplish tasks necessary for the data transfer. The actual data word transfer takes place during the Break (BK) cycle of operation. For 1-cycle peripherals only the BK cycle is necessary; however, 3-cycle devices require a Word Count (WC) cycle and a Current Address (CA) cycle before the BK cycle. This section describes the method of selecting each of the three cycles of operation; the details of what occurs during each cycle are presented in succeeding sections, except that certain BK cycle operations are detailed here.

Figure 10-3 shows the cycle select logic. When one of the three flip-flops shown (WC, CA, and BK) is set at TP4, the respective operation cycle is entered. If the peripheral is a 3-cycle device, the 3 CYCLE line is wired to ground within the peripheral. Thus, pin 2 of the DEC 8271 IC is positive voltage (high). This high is gated to the D-input of the WC flip-flop, providing the 8271 load (L) input is also high and the shift (S) input is low (see Volume 1, Appendix A, for details about the 8271 IC). This provision is met each time the interface accepts a break request from the peripheral (the 1-output of NBR goes high at INT STROBE time). The WC flip-flop is set at TP4 – the same TP4 at which MAC1 and MAC2 are set (because the D-input of both the CA and BK flip-flops is low, TP4 clears these flip-flops). Note that TP4 is NANDed with the output of a NOR gate (E18) that can be enabled by the 0-output of the ACTIVE flip-flop. This means that the WC flip-flop can be set even though access to the CP has been claimed by another peripheral during TS4 (the ACTIVE flip-flop is set prior to the priority check and remains set until TP2 of the BK cycle). If this happens, the operations that normally occur during the WC cycle are suspended for at least one cycle. One of these normal operations is the clocking of the 8271 flip-flops at TP4. To suspend this clocking process, the 8271 is placed in the "hold" mode by the NBR and MALC flip-flops (Table 10-1). The NBR flip-flop is cleared at TP1 of the suspended cycle, while the MALC flip-flop remains in the clear state (MALC is set at TP1 of a normal cycle). The "hold" condition remains in effect until *our* peripheral has priority; at this time a normal WC cycle is entered.

At TP1 of the normal WC cycle NBR is cleared and MALC is set (NBR remains clear until a new break request is accepted). As Table 10-1 shows, the 8271 IC is placed in the right-shift mode. In this mode, the three flip-flops comprise a shift register that is right-shifted by clock pulses. Thus, TP4 of the WC cycle causes the high at the 1-output of WC to be shifted into the CA flip-flop (CA is set, while WC is cleared). The CA flip-flop, also, can be set regardless of the results of the TS4 priority check (as before, the 0-output of ACTIVE enables TP4 to clock the flip-flop). The normal CA cycle operations are then suspended for at least one timing cycle. At TP1 of the suspended cycle, MALC is cleared; the 8271 is placed in the "hold" mode, preventing TP4 of the suspended CA cycle from clocking the flip-flops. When *our* peripheral has priority, the normal CA cycle is entered, MALC is set at TP1 time, and the 8271 IC is again placed in the right-shift mode. The CA cycle operations are carried out and, at TP4, the BK flip-flop is set, while the CA flip-flop is cleared.

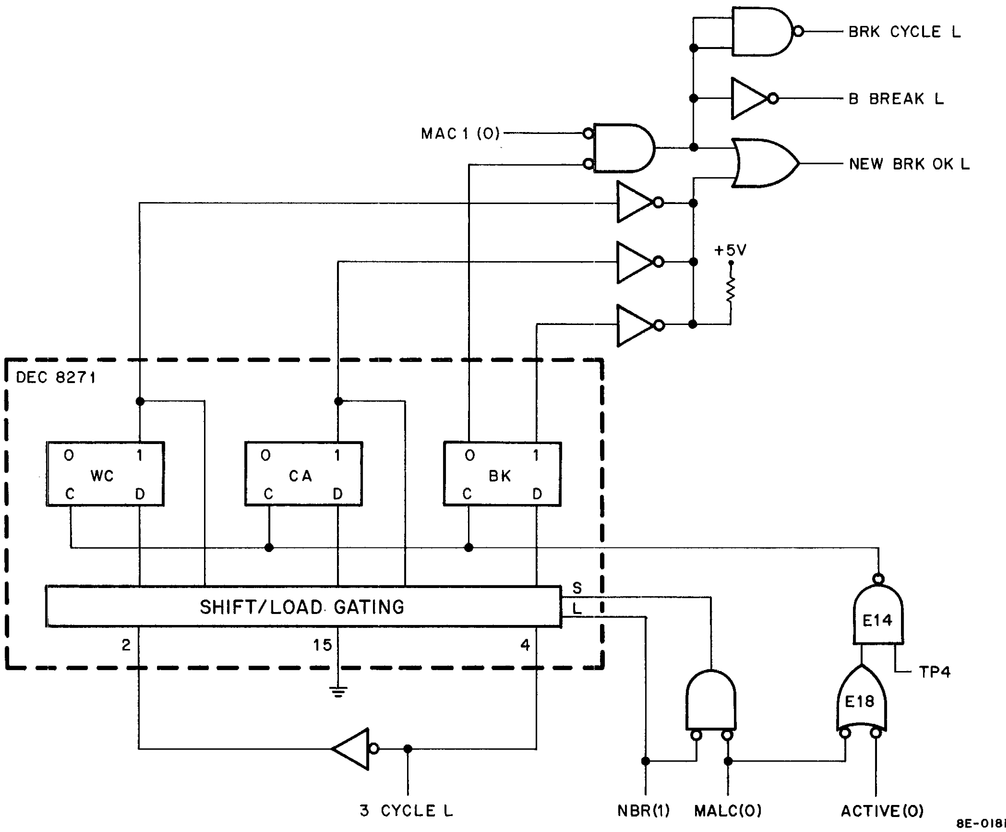


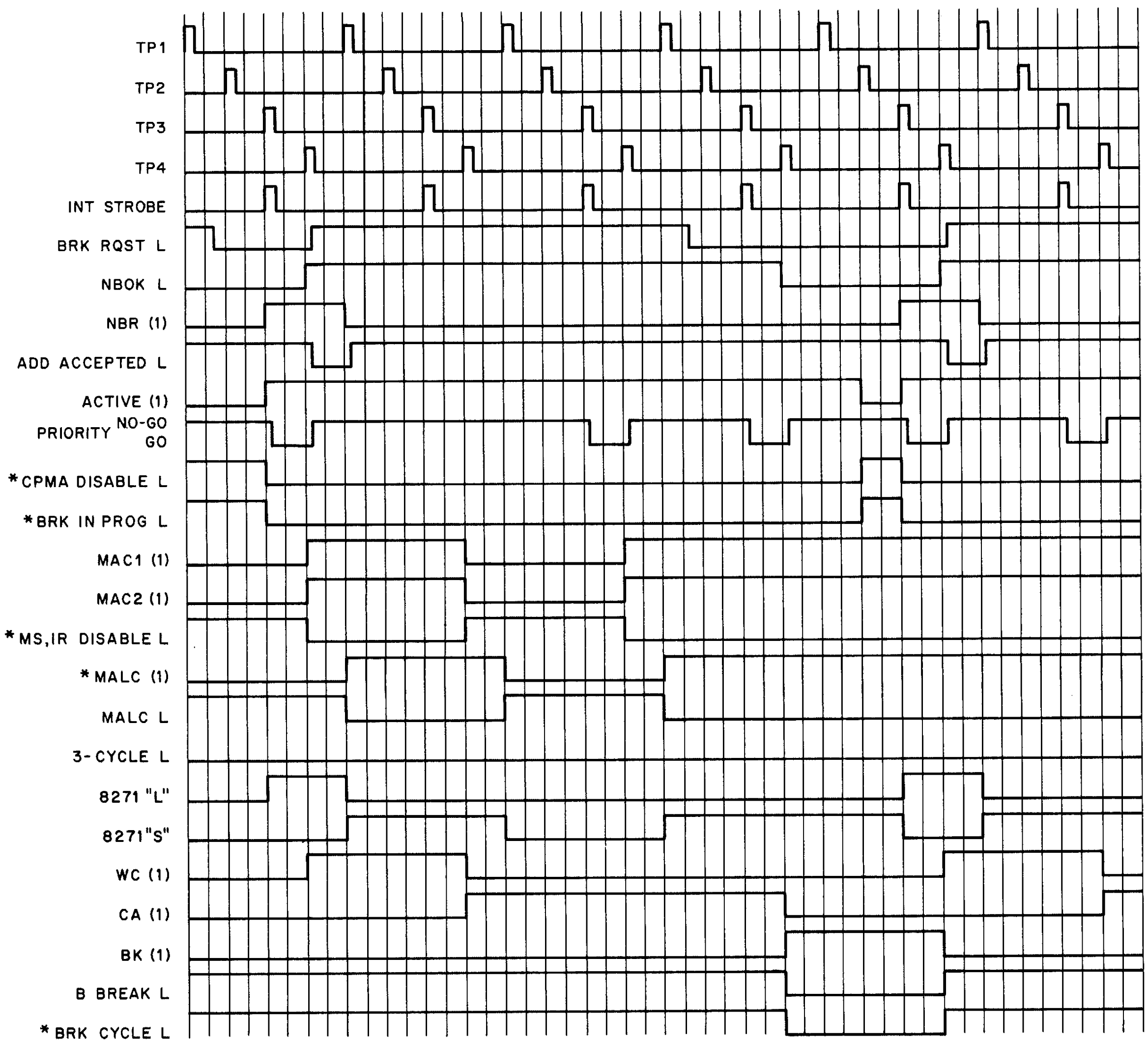
Figure 10-3 Cycle Select Logic

Table 10-1
8271 IC Control Signals

NBR F/F	MALC F/F	S	L	8271 IC Control State
Clear	Clear	Low	Low	Hold
Set	–	Low	High	Parallel Load
Clear	Set	High	Low	Right Shift

The BK flip-flop, too, is set regardless of priority. If another peripheral has priority, the normal BK cycle operations are suspended and the 8271 IC is placed on the "hold" condition for at least one cycle. When *our* peripheral has priority, the normal BK cycle is entered at TP4. The 0-output of BK if NANDed with the 0-output of MAC1 to assert NEW BRK OK L, BRK CYCLE L, and B BREAK L (Figure 10-3). Because MAC1 is cleared at TP4 if another peripheral has priority, these signals are asserted only during a normal break cycle. NEW BRK OK L tells the NBR flip-flop that data is about to be transferred and that a break request can be accepted at the next INT STROBE time; BRK CYCLE L is applied to the programmer's console display to indicate the normal or "true break" cycle; B BREAK L clocks the data into or out of the peripheral's buffer register, depending upon the direction of transfer. At TP2 of the true BK cycle the ACTIVE flip-flop is cleared, negating the CP Register control lines. If the peripheral has not asserted the BRK RQST L signal before INT STROBE time of this cycle, NBR and ACTIVE remain clear and the CP Register control lines remain negated. MAC1, MAC2, and BK are cleared at TP4, while MALC is cleared at TP1 of the next timing cycle (because ACTIVE is clear when this TP4 occurs, the 0-output of MALC is used to enable TP4 to clock the BK flip-flop). If a break request was made before INT STROBE time, the WC flip-flop is set at the same time that the BK flip-flop is cleared; the break operation is repeated as many times as necessary.

Figure 10-4 is a timing diagram relating the signals discussed in this section and in the preceding section, CP Register Control Logic (the time scale does not reflect true processor timing; refer to Volume 1, Chapter 3, for timing information).



NOTE:
3-Cycle device xfer showing 1 word xfer with a priority break during CA cycle; WC cycle is shown for second word xfer.

8E-0182

Figure 10-4 Timing, Register Control and Cycle Select Logic

10.3 PRIORITY LOGIC

Figure 10-5 shows the priority logic. Each peripheral interface contains a nearly identical circuit; differences exist only in the placement of jumper wires, which are designated A0—A11 and B0—B11 in Figure 10-5. The priority of a peripheral is established on the interface by removing a particular A jumper (all A jumpers are wired in place during production of the interface) and installing the corresponding B jumper. For example, to establish a "0" priority for *our* peripheral, remove A0 and install jumper B0. Note that this action disables all 12 NAND gates (ICs E17, E34, and E49). Thus, the output from NOR gate E18 is low. When the interface accepts a break request from the peripheral, the ACTIVE flip-flop is set at INT STROBE time. The 0-output of ACTIVE enables NAND gate E10 to assert the PRIORITY signal; therefore, *our* interface/peripheral begins the data break operation. Because *our* peripheral has only to request a break for that request to be granted, *our* peripheral has been assigned highest priority. No other interface can have its A0 jumper removed, or its B0 jumper connected.

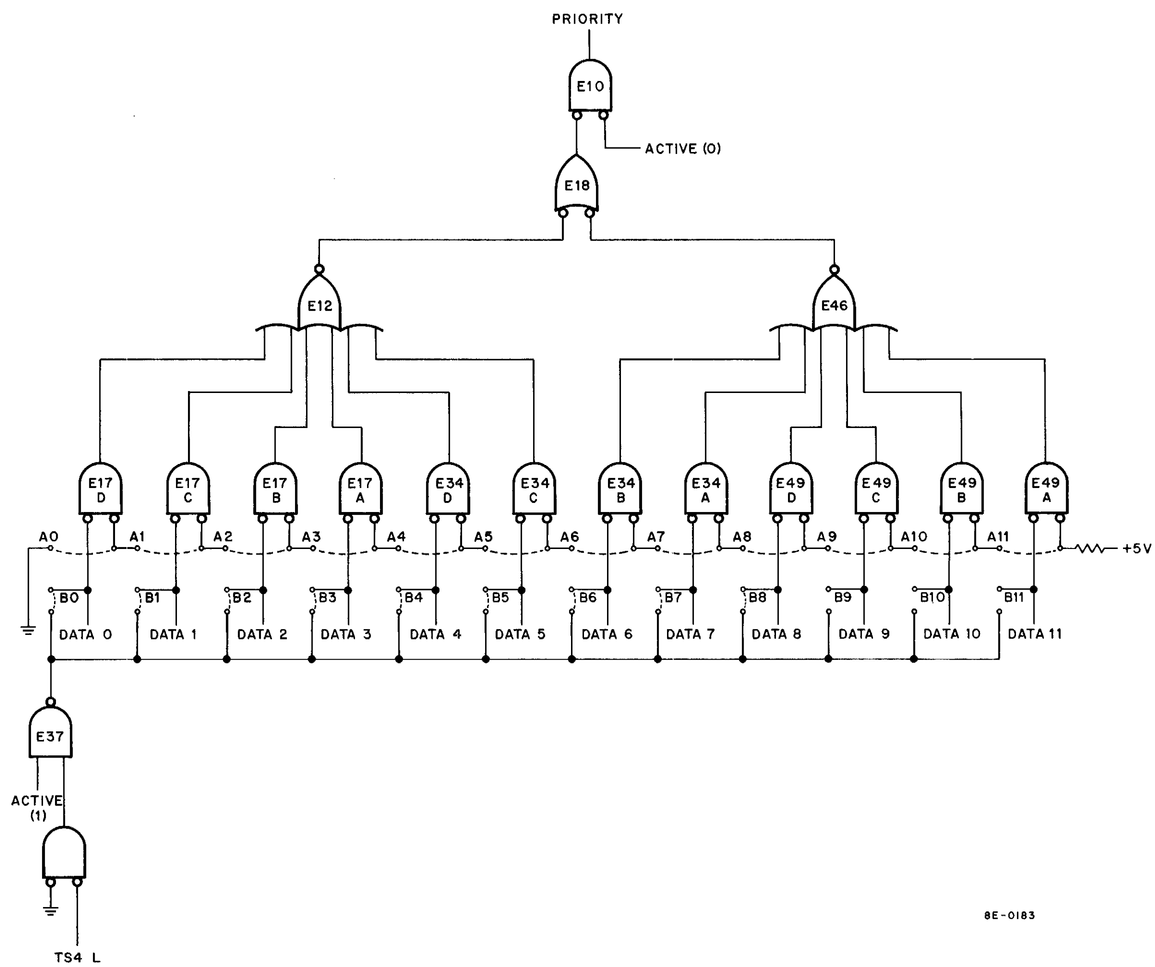


Figure 10-5 Priority Logic

As many as 11 other peripherals can have a break request accepted by their respective interfaces at INT STROBE time. No matter what priority has been established for these peripherals, each of the 11 interfaces has an A0 jumper in place. Note that when *our* interface ACTIVE flip-flop is set, NAND gate E37 brings the DATA 0 line low (*our* B0 jumper is in place) during TS4. Because all interfaces monitor the DATA lines, NAND gate E17D of each other interface is enabled. Thus, these interfaces cannot assert their PRIORITY signals as long as *our* peripheral requests data breaks.

As another example, consider what happens if *our* peripheral ranks only third highest in the peripheral priority structure. This priority is established on *our* interface by removing jumper A2 and connecting jumper B2. Because jumpers A0 and A1 are left in place, two other interfaces can keep *our* peripheral from beginning a data break operation. If the second highest priority peripheral has a break request accepted at the same time *our* peripheral's request is accepted, its interface brings the DATA 1 line low during TS4 time. NAND gate E17C on our interface is enabled and the PRIORITY signal remains negated. Until this other peripheral has completed the data break operation, our peripheral remains inactive. As has been implied, priority decreases from left to right, i.e., the lower the priority of the peripheral, the higher the number of the A jumper removed and B jumper installed. Thus, to establish priority on the lowest ranking peripheral's interface, remove jumper A11 and install jumper B11.

10.4 BKMA REGISTER LOGIC

Figure 10-6 shows the BKMA Register logic. This logic enables the peripheral to reference memory locations associated with the data break transfer. A peripheral that has made a break request must provide its interface with a memory address via the DATA ADDRESS 0–11 lines (Figure 10-6). This address is gated through DEC 8266 (refer to Volume 1, Appendix A, for details of this IC) to the parallel-load inputs of the 8271 ICs (the gating for bits 0 through 10 is identical; thus, the description and the illustration detail events for only bit 0 and bit 11). When the interface accepts the peripheral's request at INT STROBE time, the 0-output of the NBR flip-flop enables NOR gate E15, placing the 8271 ICs in the "load" condition (Table 10-1). At the same time, one input of NAND gate E14 is sent high by the 0-output of the ACTIVE flip-flop. At TP4, E14 is enabled and the BKMA Register flip-flops are loaded with the address on the DATA ADDRESS lines. If the peripheral has priority, the MAC2 flip-flop is set, also at TP4; the 1-output of the flip-flop enables NAND gate E1, placing DATA ADDRESS bit 0 on the MA0 line. If the peripheral does not have priority, the address is retained in the BKMA Register but not gated onto the MA lines until MAC2 is set at a later TP4 time. The NBR flip-flop is cleared at TP1, regardless of the outcome of the priority check, and NOR gate E15 places the 8271 ICs in "hold".

If the peripheral is a 1-cycle device, the address placed on the MA lines at TP4 is that of the memory location to or from which the data word is to be transferred. Note that NAND gate E26 is enabled in this situation (the BK flip-flop of the cycle select logic is set, the WC flip-flop is clear). Thus, the full 12- or 15-bit address supplied by the peripheral is placed on the MA lines. However, if the peripheral is a 3-cycle device, the WC cycle is entered first after a break request is accepted. NAND gate E26, disabled during the WC cycle, in turn disables NAND gate E40. Thus, DATA ADDRESS bit 11 is not gated onto the MA11 line; rather, MA11 is high, logic 0, during the cycle. The address placed on the MA lines during the WC cycle is that of the memory location containing the peripheral's WC Register; this address is hard-wired in the peripheral and is even (bit 11 is logic 0) so that the CA Register can be easily referenced during the next timing cycle, as is explained in the following paragraph.

During the WC cycle, the count in the WC Register is transferred from memory to the CP, incremented, and returned to the register. At TP4, the WC flip-flop is cleared and the CA flip-flop is set. If the peripheral still has priority, NAND gate E37 is enabled, pulling the MA11 line low. The MA0–10 lines carry the same address as during the WC cycle. Thus, the peripheral's hard-wired address is incremented and the new address is that of the memory location containing the peripheral's CA Register.

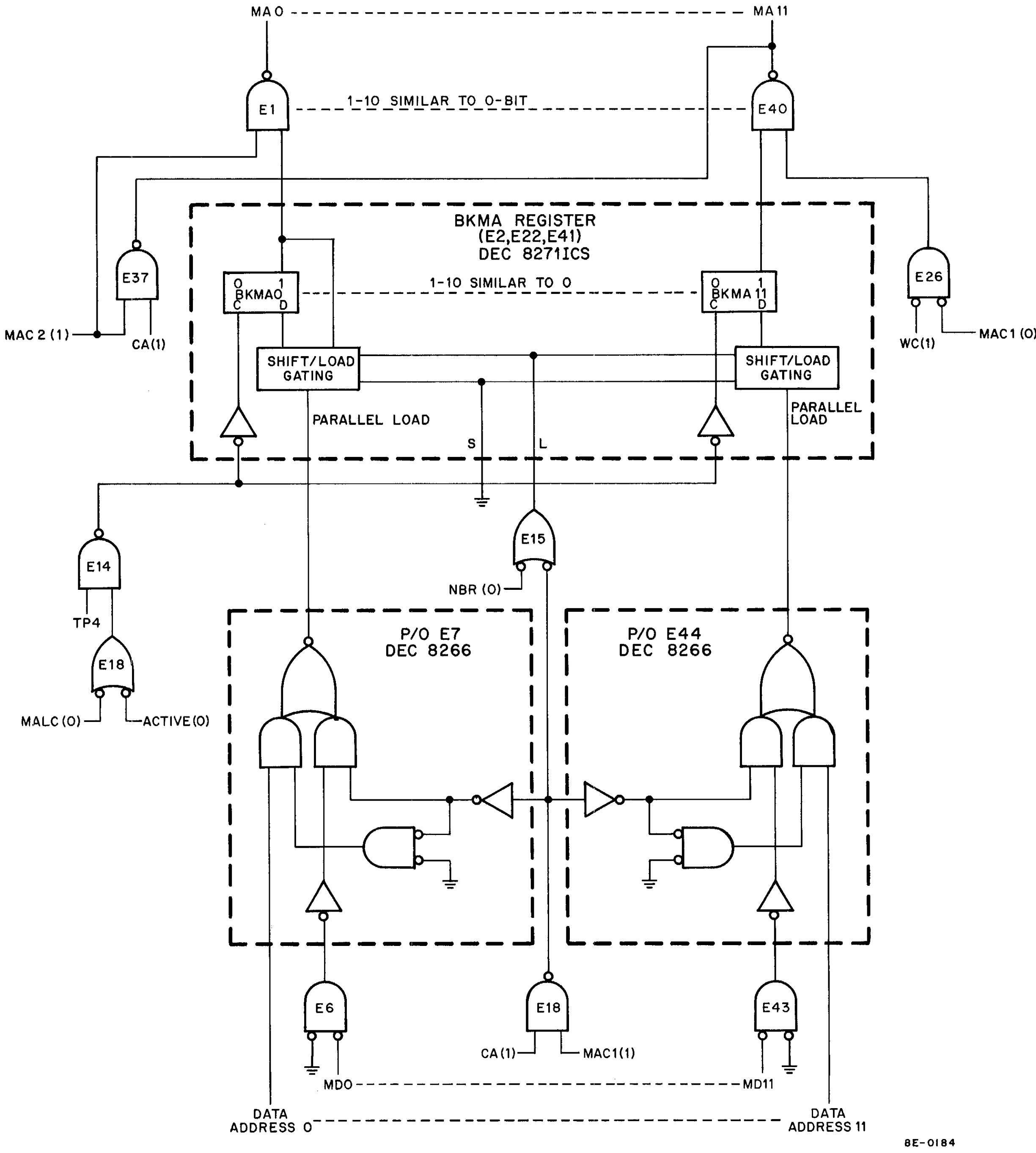


Figure 10-6 BKMA Register Logic

At the same time (TP4) that the CA Register address is placed on the MA lines, the CA flip-flop enables NAND gate E18. This gate, in turn, enables NOR gate E15, which places the BKMA Register in the "load" condition. Also, NAND gate E18 removes the DATA ADDRESS lines from the parallel-load inputs of the BKMA Register, substituting the MD0–11 lines. Note that these actions do not take place prior to the negative transition at the BKMA Register flip-flop clock inputs. Thus, the BKMA Register retains the WC Register address until TP4 of the CA cycle.

During the CA cycle, the address in the CA Register is transferred from memory to the CP, incremented, and returned to the register. The address, after incrementation (the current address), is that of the location to or from which the data is to be transferred. Thus, this current address must be placed in the BKMA Register, and transreffed from there to the MA lines at the beginning of the BK cycle. The current address is sent via the MD lines to the CP where it is incremented. At TP2, the result is returned to the MD lines and remains on these lines for the latter portion of the CA cycle. Therefore, at TP4 of the CA cycle, the BKMA Register parallel-load inputs reflect the current address. Because the BKMA Register is in the "load" condition, the current address is loaded into the register. At approximately the same time, the BK flip-flop is set, while the CA flip-flop is cleared. Because E37 is disabled and E26 is enabled, the contents of BKMA0–11 are placed on the MA0–11 lines.

Also at TP4, NAND gate E18 is disabled. This action removes the MD lines from the parallel-load inputs, selects the DATA ADDRESS lines, and places the BKMA Register in "hold". Note that the MALC flip-flop keeps NOR gate E18 enabled throughout the BK cycle (ACTIVE is cleared at TP2), if the peripheral has priority. These two gates ensure that the bKMA Register is ready to begin a new transfer, if the interface accepst a break request at INT STROBE time of the BK cycle.

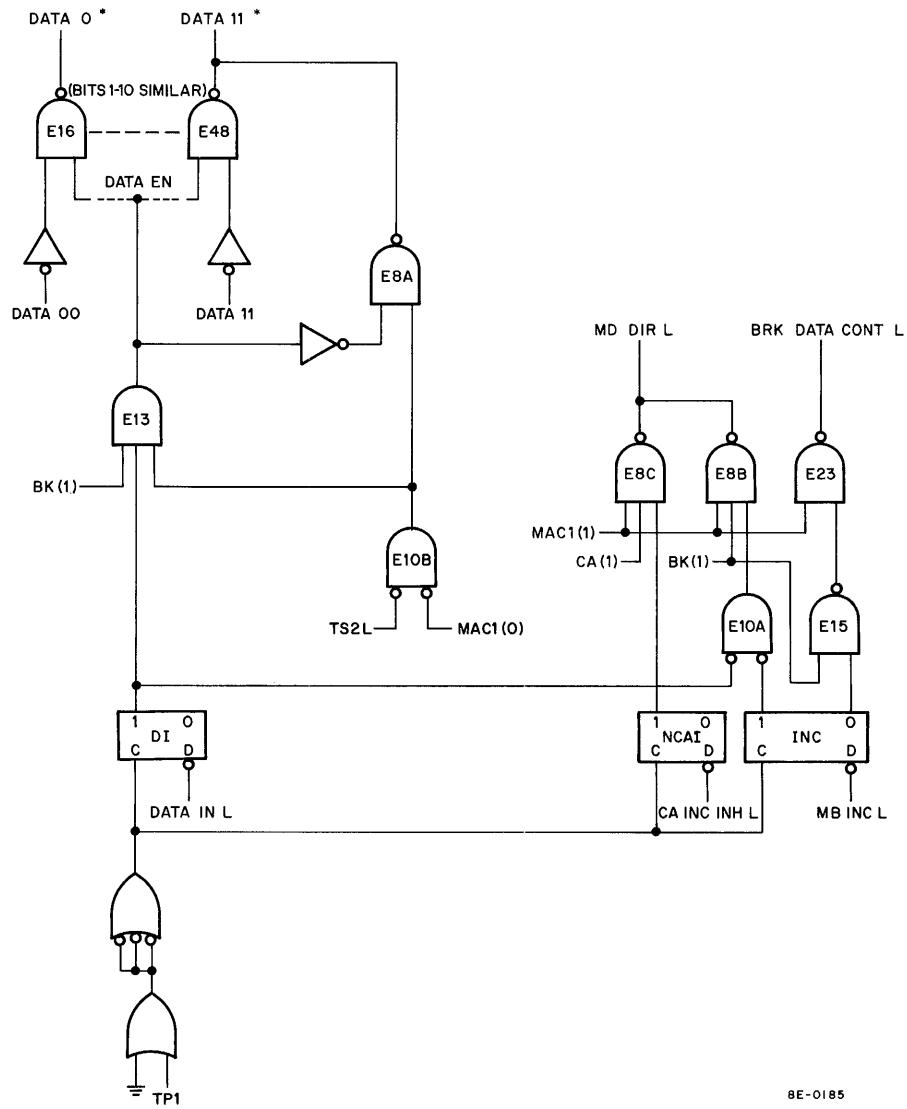
During the BK cycle, the data word is transferred to or from the location specified by the current address. The logic that accomplishes this transfer through the interface is covered in the following section, which also details the special operations of all three cycles.

10.5 DATA TRANSFER LOGIC

The data transfer logic (Figure 10-7) controls the direction and type of data transfer. Table 10-2 shows the relationship between the type and direction of transfer and the signal levels of the various control lines. The table should be used with Figure 10-7 for a good understanding of the logic details.

If the transfer is to be from the memory to the peripheral, a 12-bit data word is transferred from the addressed location via the positive I/O bus interface. The Data Break Interface asserts the MD DIR L signal so that the data word is rewritten in the memory location during the write half of the timing cycle. NAND gate E8B (Figure 10-7) is used to ground the MD DIR line. The BK (1) and MAC1 (1) signals ensure that the true break cycle is in progress. The third input to E8B is high because NAND gate E10A is enabled. E10A is controlled by the DI (Data In) and INC (INCrement) flip-flops, which are both cleared at TP1 time of an output (from memory) transfer.

Note that the peripheral need not assert any control lines for an output transfer. However, if an input (to memory) transfer is to be carried out, the peripheral usually grounds the DATA IN line (exceptions are noted in the discussion). The DI flip-flop is set at TP1 of the first cycle of the data break. If the peripheral is a 3-cycle device, this first cycle is a WC cycle. Since the BK flip-flop is clear, NAND gate E13 is disabled (note that the DI flip-flop is significant in the operation of E13 only during the BK cycle; thus, the decision to set or clear this flip-flop, by asserting or negating the DATA IN L signal, need not be made during either the WC or CA cycles). One input of NAND gate E8A is high. If this is a normal WC cycle (this interface's peripheral has priority), E8A is enabled during TS2 by NAND gate E10B, pulling the DATA 11 line low. This single bit of data is transferred to the CP and added to the word count, which is brought to the CP from the WC Register, providing the BRK DATA CONT L signal is asserted by the interface. NAND gate E23 is used to assert this signal. Because this is a WC cycle, NAND gate E15 is disabled, enabling E23 (MAC1 is set because the peripheral has priority). Therefore, the word count is incremented. Because the MD DIR L signal is negated (both E8C and E8B are disabled during the WC cycle), this new word count is placed in the WC during the write half of the memory cycle.



*OMNIBUS SIGNALS

Figure 10-7 Data Transfer Logic

During the CA cycle of the 3-cycle transfer, the current address is incremented in much the same way as the word count. Again, DATA 11 is pulled low by E8A during TS2 if the peripheral has priority. E23 asserts the BRK DATA CONT L signal, which enables DATA 11 and the current address to be added in the CP. If the MD DIR L signal is negated, the new address is sent to the CA Register during the write half of the cycle. Note that the MD DIR L signal is negated only if the NCAI (No Current Address Increment) flip-flop is clear. This flip-flop is clocked at TP1 of a timing cycle and is normally clear. However, the peripheral can ground the CA INC INH line, causing the NCAI flip-flop to be set at TP1. NAND gate E8C then asserts the MD DIR L signal during the CA cycle. Thus, although the current address is incremented as usual, the original address (the one that was in the CA Register at the beginning of the cycle) is returned to the CA Register during the memory write.

Table 10-2
Control Signals, Cycle, Type, and Direction of Transfer

Cycle	WC	CA		BR			
Direction of Transfer	In	In		In			Out
Type of Transfer	Word Count Increment	Current Address Increment	No Current Address Increment	MB Increment	ADM	12-Bit Data Word	12-Bit Data Word
MD DIR L	High	High	Low	High	High	High	Low
BRK DATA CONT L	Low	Low	Low	Low	Low	High	High
DATA IN L	Low	Low	Low	High	Low	Low	High
MB INC L	High	High	High	Low	Low	High	High
CA INC INH L	High	High	Low	High	High	High	High

During the BK cycle of operation, whether of a 1- or 3-cycle operation, an input or output transfer can take place. As described at the beginning of this section, there is only one type of output data transfer, i.e., the transfer of a 12-bit data word from the addressed location. However, there are three types of input transfers that can be carried out. One of these is similar to the transfer that takes place during the WC and CA cycles, and is designated MB Increment. To accomplish this transfer, the peripheral grounds only the MB INC line. At TP1 the INC flip-flop is set, while the DI flip-flop is cleared. NAND gates E10A and E15 are disabled by the INC flip-flop. Thus, E32 asserts the BRK DATA CONT L signal and, because E8C is disabled during the BK cycle, E8B negates the MD DIR L signal. During TS2 the DATA 11 line is pulled low by E8A. This single bit is transferred to the CP, where it is added to the data word that is brought from the addressed memory location. The incremented data is then sent back to the addressed location during memory write.

Another type of input transfer, similar to the MB Increment, is designated Add to Memory (ADM). The peripheral grounds the MB INC and DATA IN lines so that both the DI flip-flop and the INC flip-flop are set at TP1. During TS2 of the true break cycle, a 12-bit data word carried on the peripheral's DATA 00—11 lines is gated through the interface to the OMNIBUS DATA lines. This data is added in the CP to the data brought from the addressed memory location, and the result is rewritten in the memory location.

The third type of input transfer is that of a 12-bit data word to the addressed memory location. The peripheral grounds only the DATA IN line; thus, TP1 sets the DI flip-flop, while clearing the INC flip-flop. NAND gate E15 is enabled and causes NAND gate E23 to negate the BRK DATA CONT L signal. NAND gate E10A is disabled by the 1-output of DI and, in turn, disables E8B. Because E8C is also disabled, the MD DIR L signal is again high. During TS2, a 12-bit data word is placed on the OMNIBUS DATA lines and transferred to the addressed memory location.

10.6 WC OVERFLOW LOGIC AND EMA REGISTER LOGIC

Figure 10-8 shows the WC Overflow logic and the EMA Register logic. The WC OVERFLOW L signal is generated by the interface during either a normal WC cycle or a true BK cycle if the OMNIBUS OVERFLOW L signal is asserted by the CP. OVERFLOW L is asserted during a WC cycle to indicate that the last word of a block is about to be transferred. WC OVERFLOW L is then used by the peripheral to terminate the data break operation. During a BK cycle the OVERFLOW L signal is asserted to indicate that an input transfer has resulted in assertion of the CP CARRY OUT L signal. In this case, WC OVERFLOW L is used in the peripheral as directed by the program.

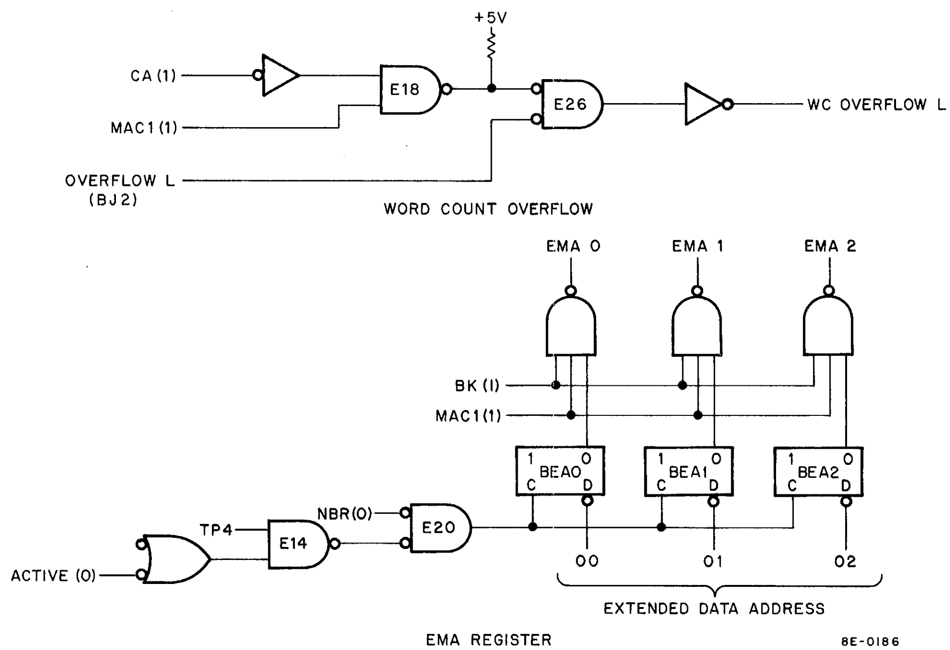


Figure 10-8 Word Count Overflow and EMA Register Logic

The EMA Register logic is used to specify the complete 15-bit memory address to or from which data is to be transferred. EMA0 is the MSB of the 15-bit address, while MA11 is the LSB (see Chapters 9 and 10 of the *PDP-8/E & PDP-8/M Small Computer Handbook* for definitions of OMNIBUS and External bus signals relating to extended memory). If the computer contains only the basic 4K memory, EMA 0, EMA 1, and EMA 2 are logic 0. When memory is extended (up to 32K, if desired), these three most significant bits are used to indicate which memory field is to take part in the data transfer. The peripheral specifies the memory field via the External bus Extended Data Address 00–02 lines (Figure 10-8). This field address is loaded into the BEA (Break Extended Address) register at TP4 of the first cycle of the break operation (Figure 3-4). If the peripheral is a 3-cycle device, the first cycle is the WC cycle. However, note that BEA Register information is placed on the EMA lines only during a true BK cycle. Thus, for a 3-cycle device the WC and CA Registers must be located in memory field 0, the basic 4K. The location to or from which data is to be transferred can be contained in an extended memory field.

SECTION 4 MAINTENANCE

There are no specific maintenance procedures for the KD8-E itself. Each DEC peripheral has an associated MAINDEC or exerciser program that enables the technician to maintain both the option and the KD8-E Interface.

General information concerning corrective maintenance is included in Volume, Chapter 4. The technician will find this material helpful. The interface schematic, E-CS-M8360-0-1, indicates important test points, IC locations, and pin numbers; it should be used when maintenance is being performed.

The KD8-E connects directly to a single peripheral via two cables that are supplied with the interface (refer to *PDP-8/E & PDP-8/M Small Computer Handbook,* Chapter 10, for cabling rules and suggestions). Each cable connects to the interface with a 40-pin Berg connector and to the peripheral with a DEC M953A cable connector. From-To information for the cable is given in Table 10-3 (the cables are identical; refer to Chapter 10 of the *PDP-8/E & PDP-8/M Small Computer Handbook* for details concerning proper connection of the cables).

Table 10-3
KD8-E Cable Information

From (M953A Cable Conn.)	To (Berg Conn.)	From (M953A Cable Conn.)	To (Berg Conn.)
Gnd	A	Gnd	Y
Gnd	B	M2	Z
Gnd	C	Gnd	AA
B1	D	L1	BB
Gnd	E	Gnd	CC
D2	F	P2	DD
Gnd	H	Gnd	EE
D1	J	M1	FF
Gnd	K	Gnd	HH
E2	L	S2	JJ
Gnd	M	Gnd	KK
E1	N	P1	LL
Gnd	P	Gnd	MM
H2	R	T2	NN
Gnd	S	Gnd	PP
H1	T	S1	RR
Gnd	U	Gnd	SS
K2	V	V2	TT
Gnd	W	Gnd	UU
J1	X	Gnd	VV

Pins A2, B2, U1 and V1 on M953A not used.

Pins A1, C1, F2, K1, N1, R1, T1, C2, F2, J2, L2, N2, R2, and U2 on M953A are ground pins.

SECTION 5 SPARE PARTS

Table 10-4 lists recommended spare parts for the KD8-E. These spare parts can be obtained from any local DEC office or from DEC, Maynard, Massachusetts.

Table 10-4
KD8-E Recommended Spare Parts

DEC Part No.	Description	Quantity
10-01610	Capacitor 0.01 μF, 100V, 20%	2
11-00113	Diode D662	1
19-05575	IC DEC 7400	1
19-05579	IC DEC 7440	1
19-09004	IC DEC 7402	1
19-09057	IC DEC 74H10	1
19-09267	IC DEC 74H11	1
19-09971	IC DEC 6380	1
19-09486	IC DEC 384	1
19-09615	IC DEC 8271	1
19-09667	IC DEC 74H74	1
19-09686	IC DEC 7404	1
19-09972	IC DEC 6314	1
19-09973	IC DEC 97401	1
19-09928	IC DEC 7416	1
19-09934	IC DEC 8266	1
19-09955	IC DEC 7412	1
19-10010	IC DEC 2501	1

HOW TO OBTAIN SOFTWARE INFORMATION

Announcements for new and revised software, as well as programming notes, software problems, and documentation corrections are published by Software Information Service in the following newsletters.

Digital Software News for the PDP-8 Family
Digital Software News for the PDP-9/15 Family
PDP-6/PDP-10 Software Bulletin
Digital Software News for the PDP-11 Family

These newsletters contain information applicable to software available from Digital's Program Library.

Please complete the card below to place your name on the newsletter mailing list.

Questions or problems concerning DEC Software should be reported to the Software Specialist at your nearest DEC regional or district sales office. In cases where no Software Specialist is available, please send a Software Trouble Report form with details of the problem to:

Software Information Service
Digital Equipment Corporation
146 Main Street, Bldg. 3-5
Maynard, Massachusetts 01754

These forms, which are available without charge from the Program Library, should be fully filled out and accompanied by teletype output as well as listings or tapes of the user program to facilitiate a complete investigation. An answer will be sent to the individual and appropriate topics of general interest will be printed in the newsletter.

New and revised software and manuals, Software Trouble Report forms, and cumulative Software Manual Updates are available from the Program Library. When ordering, include the document number and a brief description of the program or manual requested. Revisions of programs and documents will be announced in the newsletters and a price list will be included twice yearly. Direct all inquiries and requests to:

Program Library
Digital Equipment Corporation
146 Main Street, Bldg. 1-2
Maynard, Massachusetts 01754

Digital Equipment Computer Users Society (DECUS) maintains a user Library and publishes a catalog of programs as well as the DECUSCOPE magazine for its members and non-members who request it. For further information please write to:

DECUS
Digital Equipment Corporation
146 Main Street
Maynard, Massachusetts 01754

Send Digital's software newsletters to:

Name ______________________________

Company Name ______________________________

Address ______________________________

(zip code) __________

My computer is a ☐ **PDP-8/E** ☐ **PDP-9** ☐ **PDP-8/L**
☐ **PDP-8/I** ☐ **PDP-10** ☐ **PDP-12**
☐ **LINC-8** ☐ **PDP-11** ☐ **PDP-15**
Other ______________ **Please specify**

My system serial number is ______________ (if known)

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 Equipment Corporation
Technical Documentation Department
146 Main Street
Maynard, Massachusetts 01754

READER'S COMMENTS

PDP-8/E MAINTENANCE MANUAL
VOLUME II
DEC-8E-HR2C-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 Equipment Corporation
Technical Documentation Department
146 Main Street
Maynard, Massachusetts 01754

DIGITAL EQUIPMENT CORPORATION WORLD-WIDE SALES AND SERVICE

MAIN OFFICE AND PLANT

146 Main Street, Maynard, Massachusetts, U.S.A. 01754 • Telephone: From Metropolitan Boston: 646-8600 • Elsewhere: (617)-897-5111
TWX: 710-347-0212 Cable: DIGITAL MAYN Telex: 94-8457

UNITED STATES

NORTHEAST

REGIONAL OFFICE
275 Wyman Street, Waltham, Massachusetts 02154
Telephone: (617)-890-0320/0330 TWX: 710-324-6919

WALTHAM
15 Lunda Street, Waltham, Massachusetts 02154
Telephone: (617)-891-1030 TWX: 710-324-6919

CAMBRIDGE/BOSTON
899 Main Street, Cambridge, Massachusetts 02139
Telephone: (617)-491-6130 TWX: 710-320-1167

ROCHESTER
130 Allens Creek Road, Rochester, New York 14618
Telephone: (716)-461-1700 TWX: 710-253-3078

CONNECTICUT
240 Pomeroy Ave., Meriden, Conn. 06450
Telephone: (203)-237-8441/7466 TWX: 710-461-0054

MID-ATLANTIC — SOUTHEAST

REGIONAL OFFICE:
U.S. Route 1, Princeton, New Jersey 08540
Telephone: (609)-452-2940 TWX: 510-685-2338

NEW YORK
95 Cedar Lane, Englewood, New Jersey 07631
Telephone: (201)-871-4984, (212)-594-6955, (212)-736-0447
TWX: 710-991-9721

NEW JERSEY
1259 Route 46, Parsippany, New Jersey 07054
Telephone: (201)-335-3300 TWX: 710-987-8319

PRINCETON
U.S. Route 1
Princeton, New Jersey 08540
Telephone: (609) 452-2940 TWX: 510-685-2338

LONG ISLAND
1 Huntington Quadrangle
Suite 1SO7 Huntington Station, New York 11746
Telephone: (516)-694-4131, (212)-895-8095

PHILADELPHIA
Station Square Three, Paoli, Pennsylvania 19301
Telephone: (215)-647-4900/4410 Telex: 510-668-8395

WASHINGTON
Executive Building
6811 Kenilworth Ave., Riverdale, Maryland 20840
Telephone: (301)-779-1600/752-8797 TWX: 710-826-9662

DURHAM/CHAPEL HILL
2704 Chapel Hill Boulevard
Durham, North Carolina 27707
Telephone: (919)-489-3347 TWX: 510-927-0912

ORLANDO
Suite 130, 7001 Lake Ellenor Drive, Orlando, Florida 32809
Telephone: (305)-851-4450 TWX: 810-850-0180

ATLANTA
2815 Clearview Place, Suite 100,
Atlanta, Georgia 30340
Telephone: (404)-451-3734/3735/3736 TWX: 810-757-4223

MID-ATLANTIC — SOUTHEAST (cont.)

KNOXVILLE
6311 Kingston Pike, Suite 21E
Knoxville, Tennessee 37919
Telephone: (615)-588-6571 TWX: 810-583-0123

CENTRAL

REGIONAL OFFICE:
1850 Frontage Road, Northbrook, Illinois 60062
Telephone: (312)-498-2500 TWX: 910-686-0655

PITTSBURGH
400 Penn Center Boulevard
Pittsburgh, Pennsylvania 15235
Telephone: (412)-243-9404 TWX: 710-797-3657

CHICAGO
1850 Frontage Road, Northbrook, Illinois 60062
Telephone: (312)-498-2500 TWX: 910-686-0655

ANN ARBOR
230 Huron View Boulevard, Ann Arbor, Michigan 48103
Telephone: (313)-761-1150 TWX: 810-223-6053

INDIANAPOLIS
21 Beachway Drive — Suite G
Indianapolis, Indiana 46224
Telephone: (317)-243-8341 TWX: 810-341-3436

MINNEAPOLIS
Suite 111, 8030 Cedar Avenue South,
Minneapolis, Minnesota 55420
Telephone: (612)-854-6562-3-4-5 TWX: 910-576-2818

CLEVELAND
Park Hill Bldg., 35104 Euclid Ave.
Willoughby, Ohio 44094
Telephone: (216)-946-8484 TWX: 810-427-2608

ST. LOUIS
Suite 110, 115 Progress Pky., Maryland Heights,
Missouri 63043
Telephone: (314)-878-4310 TWX: 910-764-0831

DAYTON
3101 Kettering Blvd., Dayton, Ohio 45439
Telephone: (513)-299-7377 TWX: 810-459-1676

MILWAUKEE
8531 W. Capitol Drive, Milwaukee, Wisconsin 53222
Telephone: (414)-463-9110 TWX: 910-262-1199

DALLAS
8855 North Stemmons Freeway
Dallas, Texas 75247
Telephone: (214)-638-4880 TWX: 910-861-4000

HOUSTON
3417 Milam Street, Suite A, Houston, Texas 77002
Telephone: (713)-524-2961 TWX: 910-881-1651

CENTRAL (cont.)

NEW ORLEANS
3100 Ridgelake Drive, Suite 108
Metairie, Louisiana 70002
Telephone: 504-837-0257

WEST

REGIONAL OFFICE
310 Soquel Way, Sunnyvale, California 94086
Telephone: (408)-735-9200

ANAHEIM
801 E. Ball Road, Anaheim, California 92805
Telephone: (714)-776-6932/8730 TWX: 910-591-1189

WEST LOS ANGELES
1510 Cotner Avenue, Los Angeles, Californa 90025
Telephone: (213)-479-3791/4318 TWX: 910-342-6999

SAN DIEGO
3444 Hancock Street
San Diego, California 92110
Telephone: (714)-298-0591, 0593 TWX: 910-335-1230

SAN FRANCISCO
1400 Terra Bella
Mountain View, California 94040
Telephone: (415)-964-6200 TWX: 910-373-1266

PALO ALTO
560 San Antonio Rd., Palo Alto, California 94306
Telephone: (415)-969-6200 TWX: 910-373-1266

OAKLAND
7850 Edgewater Drive
Oakland, California 94621
Telephone: (415) 635-5453/7830 TWX: 910-366-7238

ALBUQUERQUE
6303 Indian School Road, N.E.
Albuquerque, N.M. 87110
Telephone: (505)-296-5411/5428 TWX: 910-989-0614

DENVER
2305 South Colorado Blvd., Suite #5
Denver, Colorado 80222
Telephone: (303)-757-3332/758-1656/758-1659
TWX: 910-931-2650

SEATTLE
1521 130th N.E., Bellevue, Washington 98005
Telephone: (206)-454-4058/455-5404 TWX: 910-443-2306

SALT LAKE CITY
431 South 3rd East, Salt Lake City, Utah 84111
Telephone: (801)-328-9838 TWX: 910-925-5834

PHOENIX
4358 East Broadway Road
Phoenix, Arizona 85040
Telephone: (602) 268-3488 TWX: 910-950-4691

PORTLAND
Suite 168
5319 S.W. Canyon Court, Portland, Ore. 97221
Telephone: (503) 297-3761/3765

INTERNATIONAL

EUROPEAN HEADQUARTERS

Digital Equipment Corporation International Europe
81 Route de l'Aire
1211 Geneva 26, Switzerland
Telephone: 42 79 50 Telex. 22 683

FRANCE

Equipment Digital S.A.R.L.

PARIS
327 Rue de Charenton, 75 Paris 12 EME, France
Telephone: 344-76-07 Telex: 21339

GRENOBLE
10 rue Auguste Ravier, F-38 Grenoble, France
Telephone: (76) 87 87 32 Telex: 32 882 F (Code 212)

GERMANY

Digital Equipment GmbH

MUNICH
8 Muenchen 13, Wallensteinplatz 2
Telephone: 0811-35031 Telex: 524-226

COLOGNE
5 Koeln, Bismarckstrasse 7,
Telephone: 0221-522181 Telex: 888-2269
Telegram: Flip Chip Koeln

FRANKFURT
6078 Neu-Isenburg 2
Am Forsthaus Gravenbruch 5-7
Telephone: 06102-5526 Telex: 41-76-82

HANNOVER
3 Hannover, Podbielskistrasse 102
Telephone: 0511-69-70-95 Telex: 922-952

AUSTRIA

Digital Equipment Corporation Ges.m.b.H

VIENNA
Mariahilferstrasse 136, 1150 Vienna 15, Austria
Telephone: 85 51 86

UNITED KINGDOM

Digital Equipment Co., Ltd.

U.K. HEADQUARTERS
Arkwright Road, Reading, Berks.
Telephone: 0734-583555 Telex: 84327

READING
The Evening Post Building, Tessa Road
Reading, Berks.

BIRMINGHAM
29/31, Birmingham Road, Sutton Coldfield, Warwicks.
Telephone: (0044) 21-355 5501 Telex: 337 060

MANCHESTER
13 Upper Precinct, Walkden, Manchester M28 5AZ
Telephone: 061-790-8411 Telex: 668666

LONDON
Bilton House, Uxbridge Road, Ealing, London W.S.
Telephone: 01-579-2334 Telex: 22371

EDINBURGH
Shiel House, Craigshill, Livingston,
West Lothian, Scotland
Telephone: 32705 / Telex: 727113

NETHERLANDS

THE HAGUE
Digital Equipment N.V.
Sir Winston Churchilllaan 370
Rijswijk/The Hague, Netherlands
Telephone: 070-995-160 Telex: 32533

BELGIUM

BRUSSELS
Digital Equipment N.V./S.A.
108 Rue D'Arlon
1040 Brussels, Belgium
Telephone: 02-139256 Telex: 25297

SWEDEN

Digital Equipment Aktiebolag

STOCKHOLM
Vretenvagen 2, S-171 54 Solna, Sweden
Telephone: 98 13 90 Telex: 170 50
Cable: Digital Stockholm

NORWAY

Digital Equipment

OSLO
c/o Firma Service
Waldenmarthranesgate 84-B-86
Oslo 1, Norway
Telephone: 37 19 85, 37 02 30 Telex: 166 43

DENMARK

Digital Equipment Corporation

COPENHAGEN
Vesterbrogade 140, 1620 Copenhagen V

SWITZERLAND

Digital Equipment Corporation S.A.

GENEVA
81 Route de l'Aire
1211 Geneva 26, Switzerland
Telephone: 42 79 50 Telex: 22 683

ZURICH
Scheuchzerstrasse 21
CH-8006 Zurich, Switzerland
Telephone: 01/60 35 66 Telex: 56059

ITALY

Digital Equipment S.p.A.

MILAN
Corso Garibaldi 49, 20121 Milano, Italy
Telephone: 872 748 694 394 Telex: 33615

SPAIN

MADRID
Atalo Ingenieros S.A., Enrigue Larreta 12, Madrid 16
Telephone: 215 35 43 / Telex: 27249

BARCELONA
Atalo Ingenieros S.A., Ganduxer 76, Barcelona 6
Telephone: 221 44 66
Digital Equipment Corporation Ttd.

AUSTRALIA

Digital Equipment Australia Pty. Ltd.

SYDNEY
P.O. Box 491, Crows Nest
N.S.W. Australia 3065
Telephone: 439-2566 Telex: AA20740
Cable: Digital, Sydney

MELBOURNE
60 Park Street, South Melbourne, Victoria, 3205
Telephone: 696-142 Telex: AA40616

PERTH
643 Murray Street
West Perth, Western Australia 6005
Telephone: 214-993 Telex: AA92140

BRISBANE
139 Merivale Street, South Brisbane
Queensland, Australia 4101
Telephone 444-047 Telex: AA40616

ADELAIDE
6 Montrose Avenue
Norwood, South Australia 5067
Telephone: 631-339 Telex: AA82825

NEW ZEALAND

Digital Equipment Corporation Ltd.

AUCKLAND
Hilton House, 430 Queen Street, Box 2471 A,
Auckland, New Zealand
Telephone: 75-533

CANADA

Digital Equipment of Canada, Ltd.

CANADIAN HEADQUARTERS
150 Rosamond Street, Carleton Place, Ontario
Telephone: (613)-257-2615 TWX: 610-561-1651

OTTAWA
120 Holland Street, Ottawa 3, Ontario K1Y 0X7
Telephone: (613)-725-2193 TWX: 610-562-8907

TORONTO
230 Lakeshore Road East, Port Credit, Ontario
Telephone: (416)-274-1241 TWX: 610-492-4306

MONTREAL
9675 Cote de Liesse Road
Dorval, Quebec, Canada 760
Telephone: 514-636-9393 TWX: 610-422-4124

EDMONTON
5531 - 103 Street
Edmonton, Alberta, Canada
Telephone: (403)-434-9333 TWX: 610-831-2248

VANCOUVER
Digital Equipment of Canada, Ltd.
2210 West 12th Avenue
Vancouver 9, British Columbia, Canada
Telephone: (604)-736-5616 TWX: 610-929-2006

ARGENTINA

BUENOS AIRES
Coasin S.A.
Virrey del Pino 4071, Buenos Aires
Telephone: 52-3185 Telex: 012-2284

VENEZUELA

CARACAS
Coasin S.A. (Sales only)
Apartado 50939
Salana Grande No. 1, Caracas
Telephone: 72-9637 Cable: INSTRUVEN

CHILE

SANTIAGO
Coasin Chile Ltda. (sales only)
Casilla 14588, Correo 15, Santiago
Telephone: 396713 Cable: COACHIL

JAPAN

TOKYO
Rikei Trading Co., Ltd. (sales only)
Kozato-Kaikan Bldg.
No. 18-14, Nishishimbashi 1-chome
Minato-Ku, Tokyo, Japan
Telephone: 5915246 Telex: 781-4208
Digital Equipment Corporation International
Kowa Building No. 17, Second Floor
2-7 Nishi-Azabu 1-Chome
Minato-Ku, Tokyo, Japan
Telephone: 404-5894/6 Telex: TK-6428

PHILIPPINES

Stanford Computer Corporation
P.O. Box 1608
416 Dasmarinas St., Manila
Telephone: 49-68-96 Telex: 742-0352

INDIA

H.S. Sonawala Mg Director (Sales Only)
HINDITRON SERVICES PUT LTD.
69/A Nepean Sea Road
Bombay, India