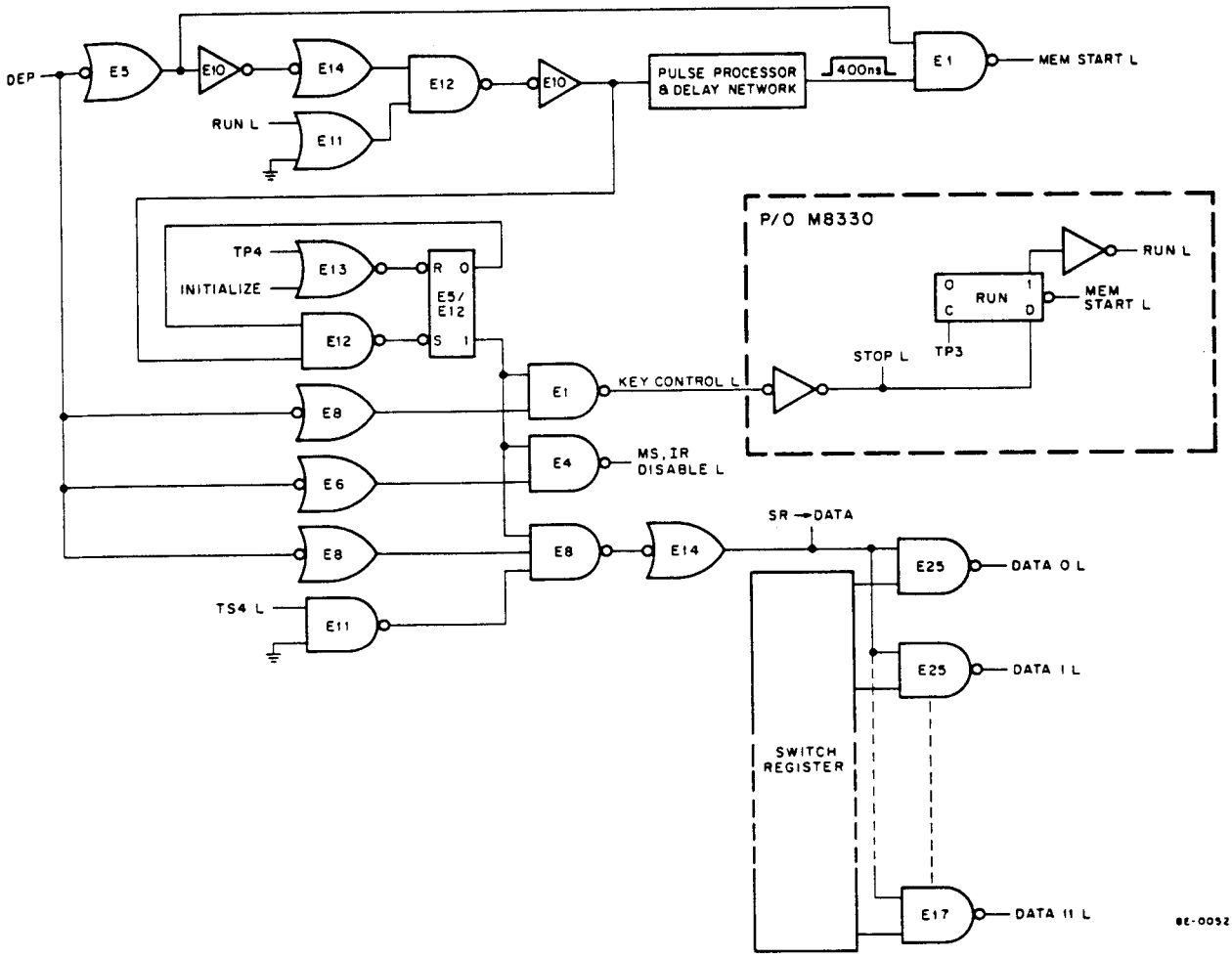


Figure 3-71 DEP Key Logic

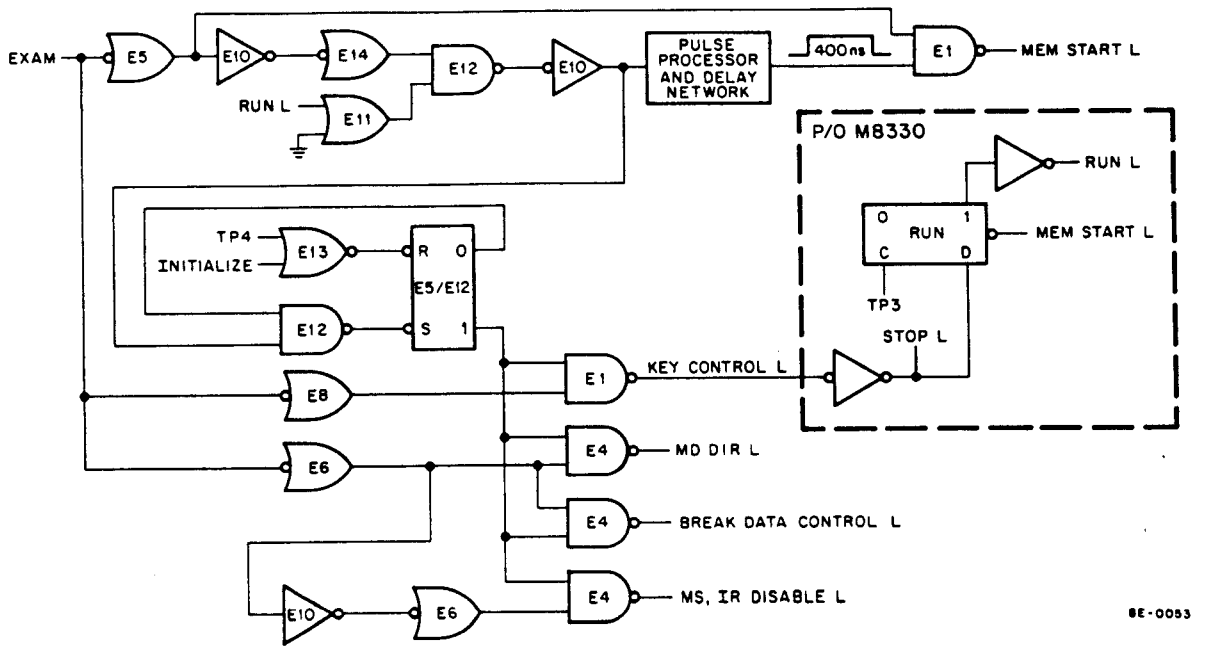


Figure 3-72 EXAM Key Logic

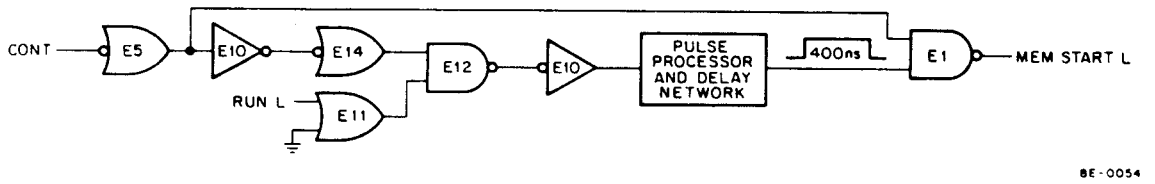


Figure 3-73 CONT Key Logic

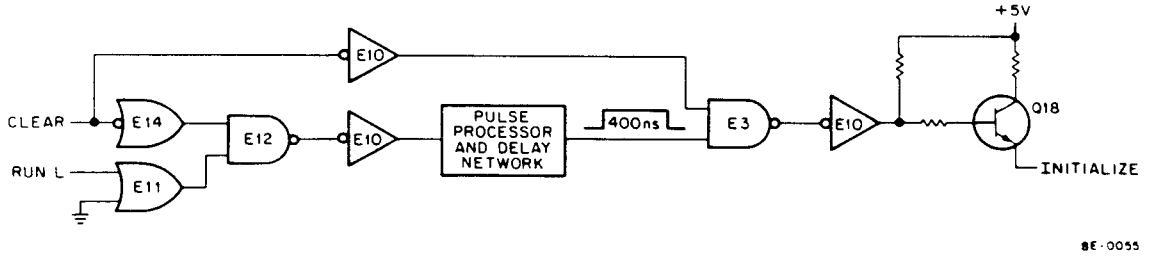


Figure 3-74 CLEAR Key Logic

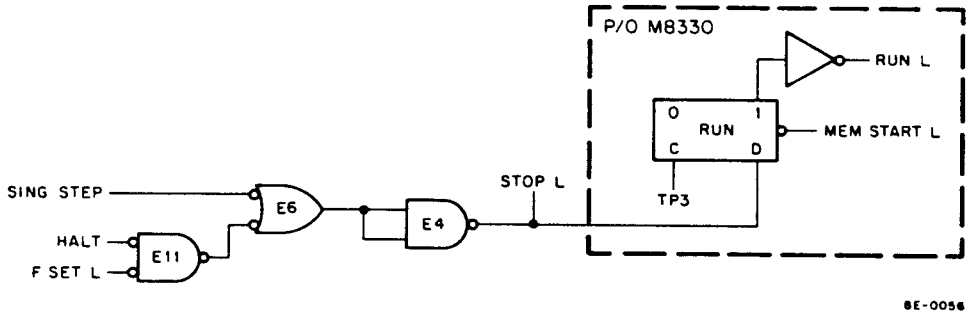
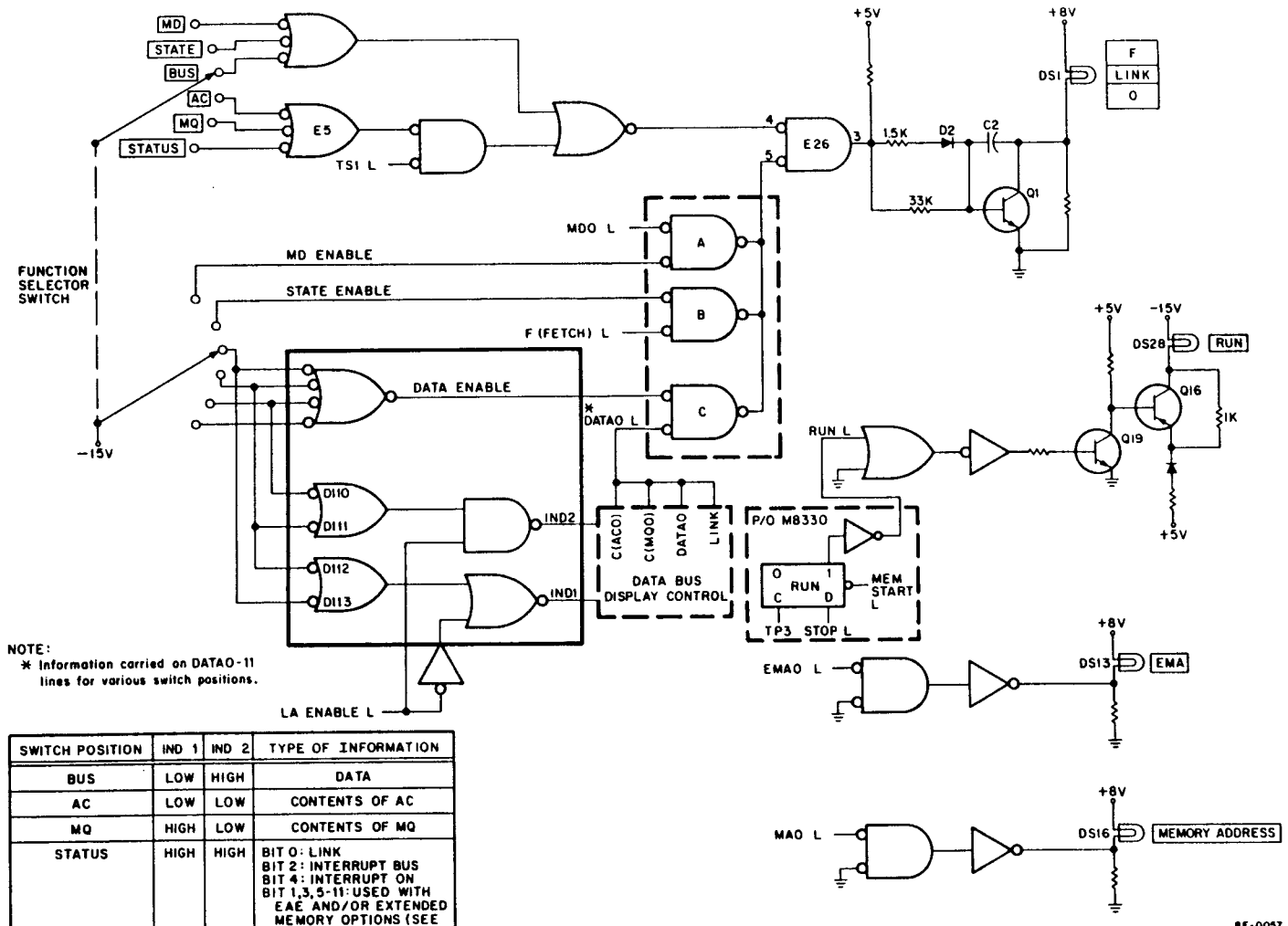


Figure 3-75 SING STEP and HALT Switch Logic

3.33.1.2 Display – The display logic and circuits are shown in Figure 3-76. There are three groups of indicator lamps and a single lamp (RUN) which, when lit, indicates that timing cycles are being generated. Each of the three groups is represented by its 0 bit, e.g., EMA0 L, MA0 L, LINK. The circuits used to display EMA and MEMORY ADDRESS are relatively simple; for example, if MA0 L is a logic 1, the NAND gate is enabled. The output of the inverter drops to a ground level. The full 8V supply voltage appears across the lamp and causes it to light. The lamp has a small voltage across it when MA0 L is a logic 0 (this condition extends the life of the lamp by eliminating the full-on/full-off cycle that often burns out lamp filaments). Consequently, it is lit at this time; however, the lamp is so dim that it is not visible from the front panel.

The circuit used to display RUN is shown above the EMA display circuit. When timing cycles are not being generated, the RUN flip-flop in the Timing Generator is cleared. Q19 and Q16 are in the nonconducting state. The conducting path for the lamp voltage includes the 1000Ω resistor; thus, a small voltage appears across DS28, causing it to be dimly lit. When RUN L is asserted by MEM START L, Q19 and Q16, in turn, are turned on. The conducting path from +5V to -15V takes the low impedance route through Q16 (from emitter to collector), rather than through the 1000Ω resistor. Thus, the RUN lamp is brightly lit, indicating that timing cycles are being generated.



SWITCH POSITION	IND 1	IND 2	TYPE OF INFORMATION
BUS	LOW	HIGH	DATA
AC	LOW	LOW	CONTENTS OF AC
MQ	HIGH	LOW	CONTENTS OF MQ
STATUS	HIGH	HIGH	BIT 0: LINK BIT 2: INTERRUPT BUS BIT 4: INTERRUPT ON BIT 1,3,5-11: USED WITH EAE AND/OR EXTENDED MEMORY OPTIONS (SEE SMALL COMPUTER HANDBOOK-1972)

Figure 3-76 Display Logic

The RUN indicator uses -15V and +5V, rather than +8V, to operate the lamp. The +8V supply is removed from the display panel, thus extending the lamp life, when the panel is locked. The -15V supply is not removed; thus, there is an indication of whether or not the computer is running.

The last group of indicators uses the greatest amount of the display logic and circuitry. This group reflects the data appearing on the MD 00–11 lines, the data appearing on the DATA 00–11 lines, and the state of selected registers and OMNIBUS control lines.

The type of information displayed by this group of indicators is selected at the front panel by a six-position rotary switch, labeled "Function Selector Switch" on Figure 3-76. Note that this switch produces one of three enable signals, depending on its position. If MD ENABLE is asserted, data on the MD 00–11 lines is displayed on the front panel; if STATE ENABLE is asserted, data on selected OMNIBUS signal lines is displayed; if DATA ENABLE is asserted, data on the DATA 00–11 lines is displayed.

If the operator wants to monitor the information on the MD lines, he selects the MD position of the rotary switch. This action asserts MD ENABLE, which is ANDed with MD0 L (the actual circuits within this dashed line are presented in a following paragraph of this section). If MD0 L is a logic 1, the AND gate brings pin 5 of NAND gate E26 to a virtual ground. Because pin 4 is also at ground, E26 is enabled, and pin 3 goes to +5V. Transistor Q1 turns on rapidly because its base drive is supplied through the low-impedance diode path. The switching action of Q1 places a ground on its collector; thus, approximately 8V appear across DS1, causing it to be brightly lit. When E26 is again disabled, pin 3 goes to ground, and diode D2 is reversed-biased. Capacitor C2 begins to discharge through the 33 kΩ resistor. This long RC time constant ensures that Q1 turns off slowly, increasing the visibility of DS1.

If the operator wants to monitor the content of the AC Register, he selects the AC position; the DATA ENABLE signal is then asserted. In addition, NOR gates D110/D111 and D112/D113 are enabled (the actual circuit within this solid line is presented in a following paragraph). When these gates are enabled, both IND1 and IND2 are asserted, provided that LA ENABLE L is not asserted. These control lines cause the content of the AC Register to be placed on the DATA 00–11 lines. If DATA 0 is a logic 1, E26, pin 5 is grounded; pin 4 is also grounded, but only during TS1, and the content of DATA 0 is displayed on DS1.

Similarly, the content of the MQ Register, STATUS information, and BUS data can be displayed. The relationship between IND1/IND2 and the type of information displayed is indicated on Figure 3-76. Figure 3-77 shows the logic used to generate control signals in response to IND1 and IND2. This logic is contained within the block designated DATA BUS DISPLAY CONTROL in Figure 3-76.

Figure 3-78 shows the circuit represented by AND gate C, enclosed within the dashed line in Figure 3-76 (gate C was arbitrarily selected for discussion; the following can apply to gates A and B, if the signal names and component designations are changed accordingly). Both the DATA ENABLE line and the DATA 0 line must be asserted if pin 5 of E26 is to be true (logic 1, or 0V). If the DATA ENABLE line is not asserted, it is at +5V. The junction of diodes D61 and D36 is +5V; neither diode conducts current; thus, pin 5 is +5V. When the DATA ENABLE line is asserted, both diodes can conduct current. If the DATA 0 line is negated (at 3V), the diode junction goes to approximately 2.3V. Pin 5 then goes to 3V, and E26 remains disabled. When DATA 0 becomes true (0V), the junction goes to -0.7V, pin 5 goes to ground, and E26 is enabled.

Figure 3-79 shows the circuit represented by the logic gates and enclosed within the solid line in Figure 3-76. Each of the four switch positions asserts DATA ENABLE by switching the DATA ENABLE line to -15V through a diode. Thus, the line, when asserted, is at approximately -14V.

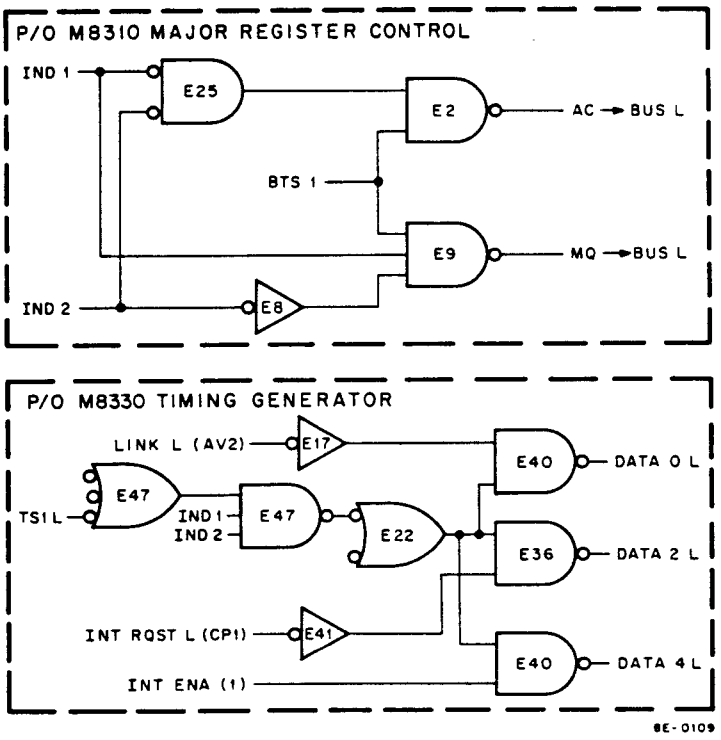


Figure 3-77 Data Bus Display Control Signals

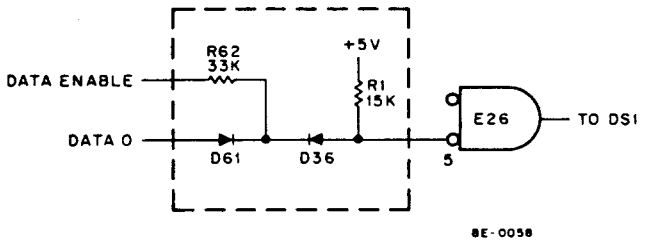


Figure 3-78 Enable Circuit

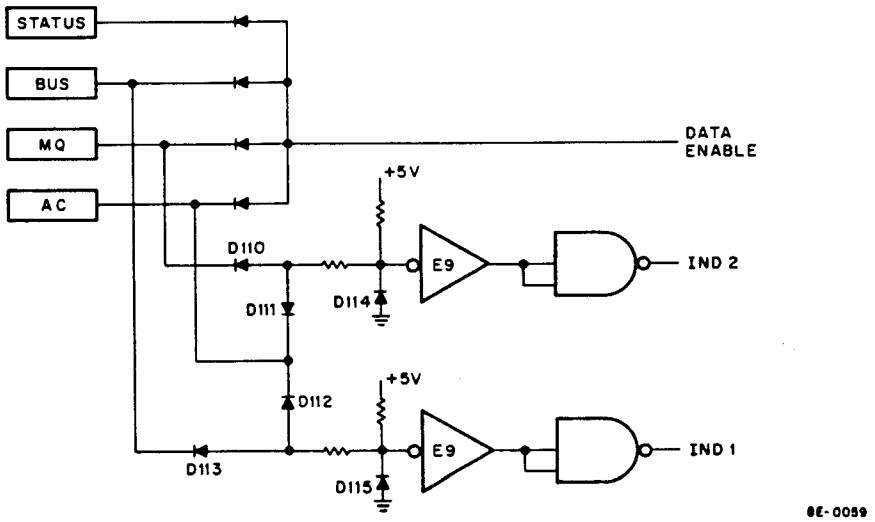


Figure 3-79 DATA ENABLE/IND SELECT

Only STATUS asserts DATA ENABLE without affecting either the IND1 or IND2 control lines. Each of the remaining three positions causes one or both of these control lines to be asserted. For example, if BUS is selected, the junction of diodes D113 and D112 goes to approximately - 14V. D115 has a conduction path and, thus, clamps the input of E9 at -0.7V. The resulting positive voltage at the output of E9 enables the following NAND gate to assert IND1. If neither D112 nor D113 conducts, D115 remains in the nonconducting state. The input to E9 remains at +5V, and IND1 is negated.

The pulse processor and delay network is shown in Figure 3-80. The input to the network, from inverter E10, is a positive-going level that results from closure of a front panel key. The output from the network is a noise-free, 400 ns gate that is used to assert either MEM START L, PULSE LA, or INITIALIZE.

When a front panel key, EXAM, is depressed, the resulting negative-going edge at the input to inverter E10 may appear as shown in Figure 3-80; i.e., noise spikes appear because of contact bounce. The integrator at the output of E10 is designed to remove the noise spikes. The large capacitance is unresponsive to noise, and, therefore, smooths the edge, while also greatly increasing its rise time. The integrating action, along with the shaping and filtering accomplished by the three transistors, which comprise a Schmitt trigger, produces a positive transition at the collector of Q15. This transition is delayed approximately 20 ms from the negative transition at the input of E10.

The differentiator converts the transition to a positive spike, which triggers a one-shot. This spike is inverted by NOR gate E2; the leading edge of the negative spike is coupled through the 330 pF capacitor and enables NAND gate E2. Simultaneously, the leading edge enables a charging path for the capacitor. This path is sustained by bringing the NAND gate output back to keep the NOR gate enabled. The capacitor charges toward +5V on a long time constant, keeping the NAND gate enabled for approximately 400 ns. When the capacitor has charged to a voltage sufficiently high to disable NAND gate E2, the charging path is removed. Thus, the output of E2 is a 400 ns positive gate and, in this instance, asserts MEM START L for that length of time.

3.33.2 KC8-FL Programmer's Console

3.33.2.1 Manual Operation

Switch Register – The switch register comprises 12 switches that allow the operator to load the CPMA Register with a 12-bit memory address; to load the extended address bits, if more than 4K of memory is used; to deposit a 12-bit data word in a selected memory location; and to change the content of the AC Register. To carry out these functions, the switch register is operated in conjunction with the DEP, ADDR LOAD, and EXTD ADDR LOAD keys (Figure 3-67).

Figure 3-81 illustrates the logic used to set data into the switch register and to place it on the DATA 0–11 lines. The operator selectively closes switches S11 through S22 (designated "0" through "11", respectively, on the front panel). He then causes NOR gate E20 to assert SR → DATA; this signal gates the information represented by the switch register keys onto the DATA lines.

There are three ways for the operator to assert SR → DATA. If he wants to load the CPMA Register, he depresses ADDR LOAD, causing NAND gate E13A to be enabled. (The signals designated A0 L, A1 L, A2 L, and E4 (1) are enable signals that are selectively generated when the operating keys are activated; these signals are described in the *Operating Keys* section.) If he wants to load the extended address bits, he depresses EXTD ADDR LOAD, again causing E13A to be enabled. If he wants to deposit data in a memory location, he raises the DEP key, causing NAND gate E9C to be enabled during TS2 (a variety of information must be carried by the DATA lines;

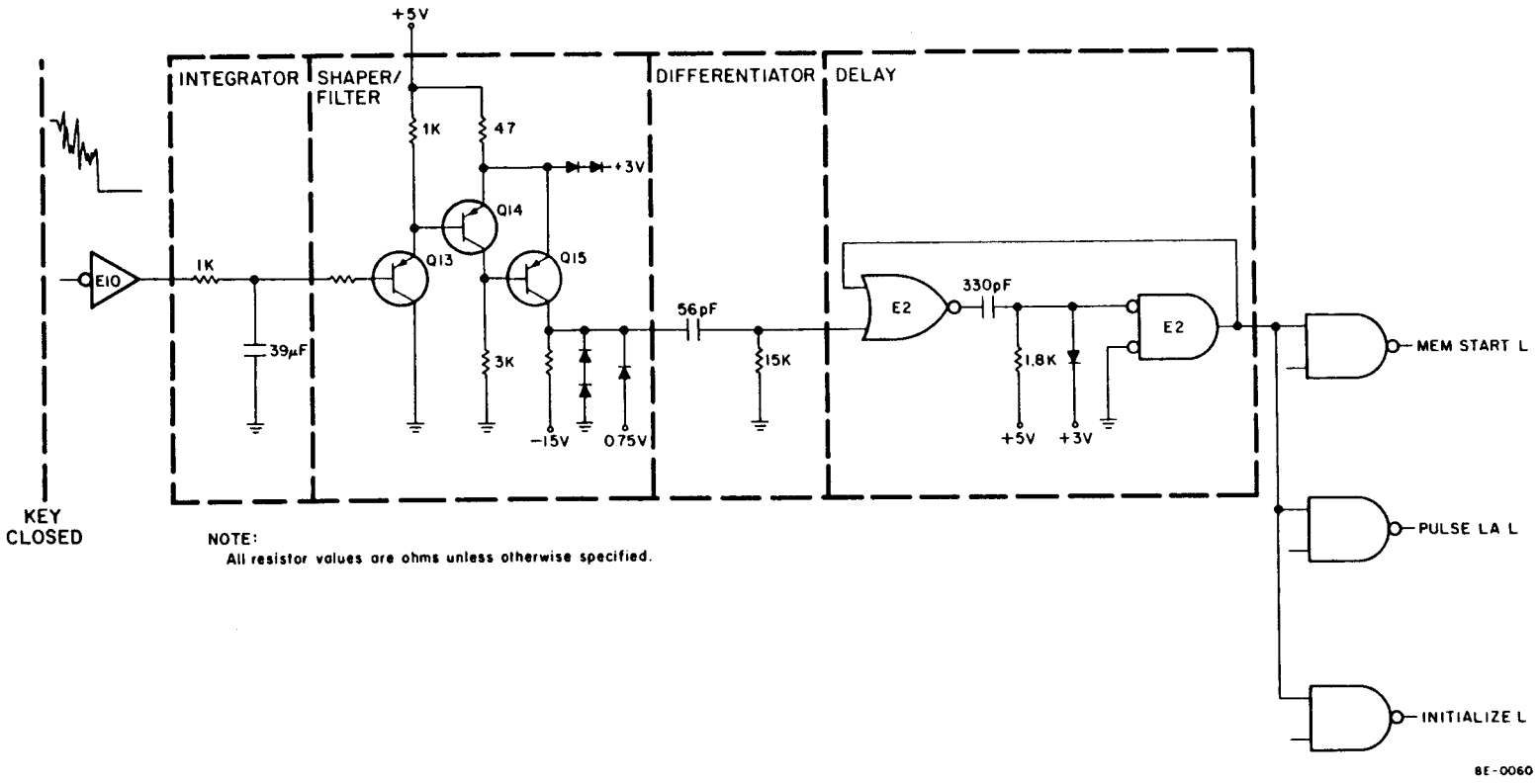


Figure 3-80 Pulse Processor and Delay Network, Schematic

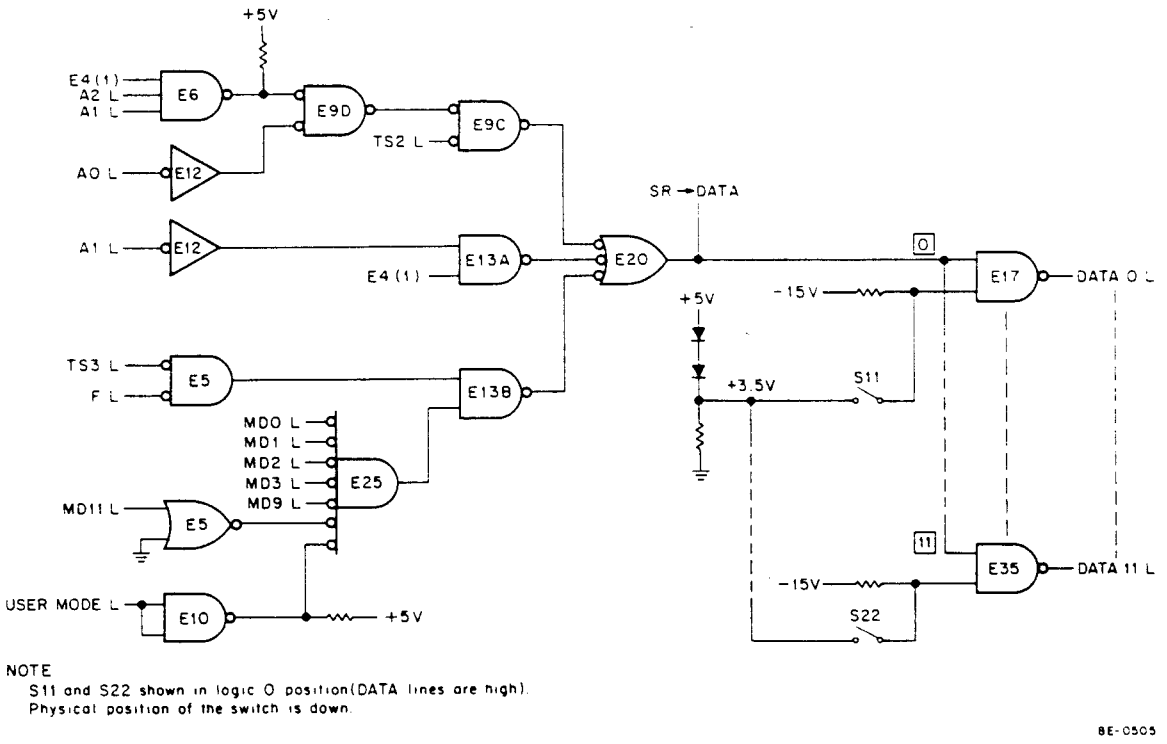
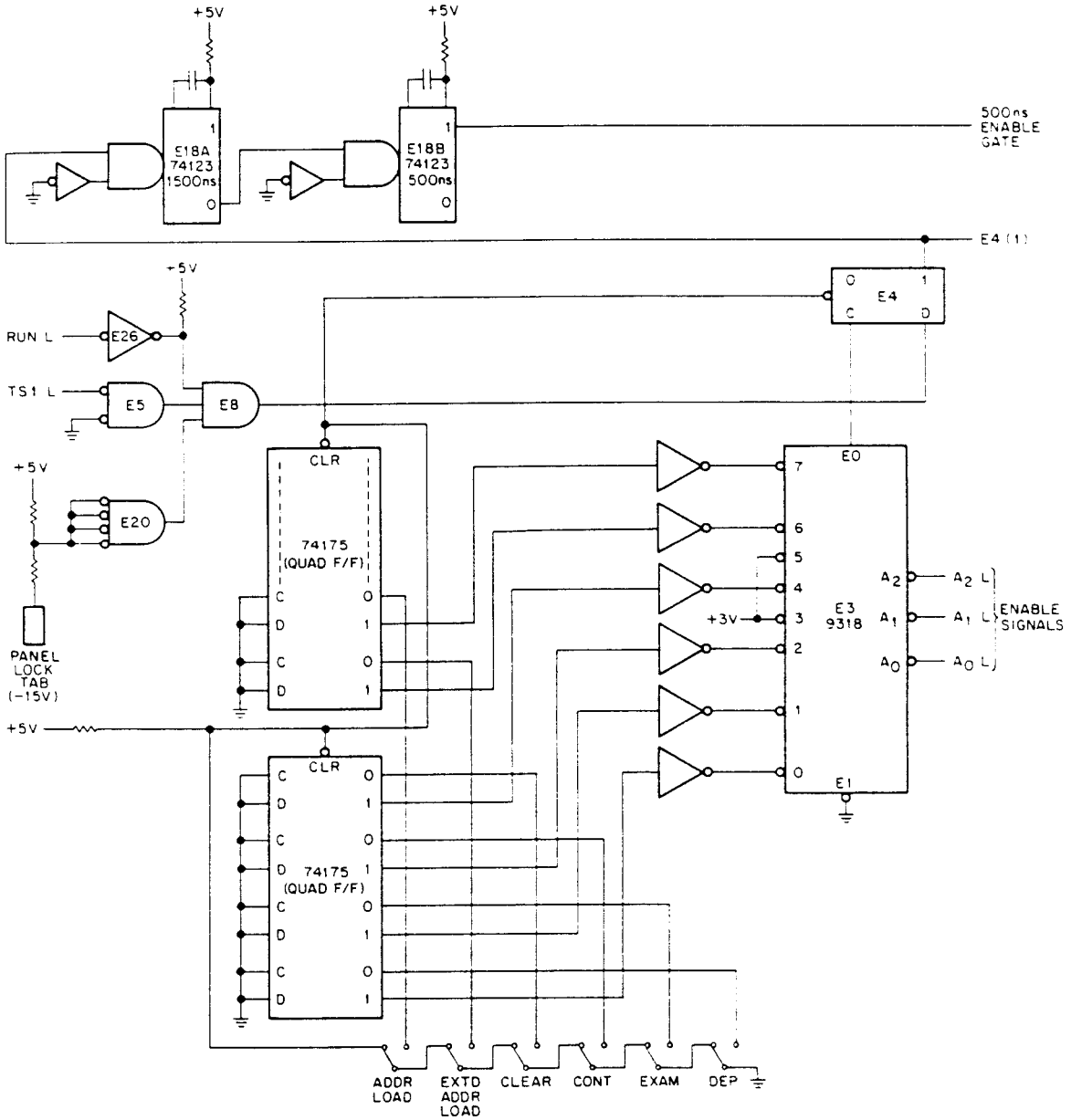


Figure 3-81 Switch Register Control Logic

time sharing of the lines must be employed to maintain the identity of each type of information). Finally, if the operator wants to change the content of the AC Register under program control, he can program either the OSR instruction (inclusive OR, switch register with AC) or the LAS instruction (load AC with switch register). Either instruction causes gate E25 to be enabled, provided USER MODE L is not asserted by the KM8-E option. When NAND gate E5 is enabled during TS3, E13B is enabled, in turn, and the SR → DATA signal is asserted. NANDing the F L signal in gate E5 ensures that E13B is not enabled during the DEFER or EXECUTE cycle of a multicycle instruction.

Operating Keys – The operating keys selectively generate enable signals when they are activated. The enable signals, in turn, selectively assert control signals that cause the processor to carry out the intended key function (all but one of the control signals, SR → DATA, are on the OMNIBUS). The logic used to generate the enable signals is shown in Figure 3-82.

Each key controls one of the D-type flip-flops of a DEC 74175 quad flip-flop IC. The normally open contact of each key is connected to the 0 output of a flip-flop. When a key is activated, the ground placed on the 0 output of the associated flip-flop forces the flip-flop to the set state. The 1 output is applied through an inverter to a DEC 9318 8-input priority encoder. Each input of the encoder is assigned a value from 0 to 7. When an input line is activated, the 9318 encoder provides a binary representation on the active low outputs A0, A1, and A2 (A0 is the LSB). At the same time, the 9318 active high output, EO, goes high, clocking flip-flop E4. If the computer is stopped (power is on but timing has not been initiated), the RUN L signal is high and the D input of E4 is high. Thus, E4 is set and the E4 (1) signal is asserted (if the front panel OFF/POWER/PANEL LOCK switch is in the PANEL LOCK position, the - 15V supply voltage is removed from the panel lock tab, and NAND gate E8 cannot be enabled; this effectively disables the operating keys and switches and prevents manual operation of the switch register). The E4 (1) signal triggers 1-shot E18A; 1500 ns later the 500 ns enable gate is generated when E18B is triggered.



7	6	5	4	3	2	1	0	E0	A_2	A_1	A_0	KEY
H	H	H	H	H	H	H	H	L	H	H	H	—
L	X	X	X	X	X	X	X	H	L	L	L	ADDR LOAD
H	L	X	X	X	X	X	X	H	L	L	H	EXTD ADDR LOAD
H	H	H	L	X	X	X	X	H	L	H	H	CLEAR
H	H	H	H	H	L	X	X	H	H	L	H	CONT
H	H	H	H	H	H	L	X	H	H	H	L	EXAM
H	H	H	H	H	H	H	L	H	H	H	H	DEP

H = High voltage level
L = Low voltage level
X = Dont care

8E-0506

Figure 3-82 Operating Keys, Enable Signal Logic

The table in Figure 3-82 relates the keys and the 9318 inputs and outputs. The table reflects the dominant feature of the 9318 priority encoder, viz., if two or more inputs are simultaneously active, the input with the highest priority is represented by the binary output. Input 7 is assigned highest priority; thus, the ADDR LOAD key takes precedence over all other keys. If the operator depresses the ADDR LOAD key, for example, the 9318 IC asserts the A0 L, A1 L, A2 L, and EO signals. Until this key is released, no other key can affect the enable signals. When the key is released, the 74175 CLR input is returned to ground, clearing the flip-flop that was set by the ADDR LOAD key. Now, the flip-flop associated with a key of lower priority causes the 9318 output to change. Note that flip-flop E4 is also cleared when a key is released. Thus, the enable signals and, as a result, the control signals are asserted only as long as the operator depresses a key. Manual operation with the KC8-FL is, therefore, as fast as the operator can make it.

The logic used to generate the control signals is shown in Figure 3-83. The gating of the logic in response to the enable signals is not discussed; this task is left to the reader. Table 3-4, which will facilitate this task, relates the control signals, the enable signals, and the keys. The state of each of the control inputs of the 8235 IC is also tabulated. A functional description of each of the keys follows. Refer to the flow diagram, the logic diagrams, and the tables while reading the functional descriptions.

ADDR LOAD Key and EXTD ADDR LOAD Key – The ADDR LOAD key is used to load the CPMA Register with the memory address specified by the switch register. When the operator depresses this key, switch register information is placed on the DATA 0–11 lines. At the same time, LA ENABLE L and MS, IR DISABLE L are asserted. These two control signals enable a path for the DATA lines through the adder network to the MAJOR REGISTERS BUS. The LOAD ADDRESS signal is then asserted. This signal produces the CPMA LOAD L pulse, which loads the CPMA Register with the information on the DATA 0–11 lines. The memory location specified by the CPMA Register can now be operated on by the DEP key or the EXAM key.

Note that the ADDR LOAD key does not initiate a timing cycle. The purpose of this key is to establish a memory location at which some operation will take place. If a timing cycle were to be initiated, the CPMA address would be incremented, and the operation would take place at the desired address, plus 1. Therefore, to avoid confusion, ADDR LOAD does not initiate a timing cycle.

The EXTD ADDR LOAD key, likewise, does not initiate a timing cycle. This key is used to load switch register bits 6–11 into the Instruction Field (IF) and Data Field (DF) Registers of the KM8-E Memory Extension and Time Share option (bits 0–5 of the switch register are used for IOT designation and device selection code). When the operator depresses the EXTD ADDR LOAD key, the switch register information is placed on the DATA 0–11 lines. The LA ENABLE L and KEY CONTROL L signals are asserted, providing a path to the KM8-E option for the DATA 6–11 lines. The LOAD ADDRESS signal is then asserted and the DF and IF Registers are loaded with the extended address information. The address specified by the CPMA Register and the DF and IF Registers can now be operated on by other keys.

DEP Key – The DEP key is used to deposit the data represented by the switch register in a specified memory location. When the DEP key is lifted, two control signals, MS, IR DISABLE L and KEY CONTROL L, are asserted. KEY CONTROL L selects the MA lines for gating into one leg of the adder network, while MS, IR DISABLE L provides an arithmetic 0 at the other adder input. KEY CONTROL L also asserts a processor signal that increments the CPMA Register. 1500 ns later, the MEM START L signal is asserted, the RUN flip-flop is set, and a timing cycle begins. At TP1 time the PC Register is loaded with the next consecutive memory address (note that the nonincremented address remains in the CPMA Register until TP4 time of the cycle).

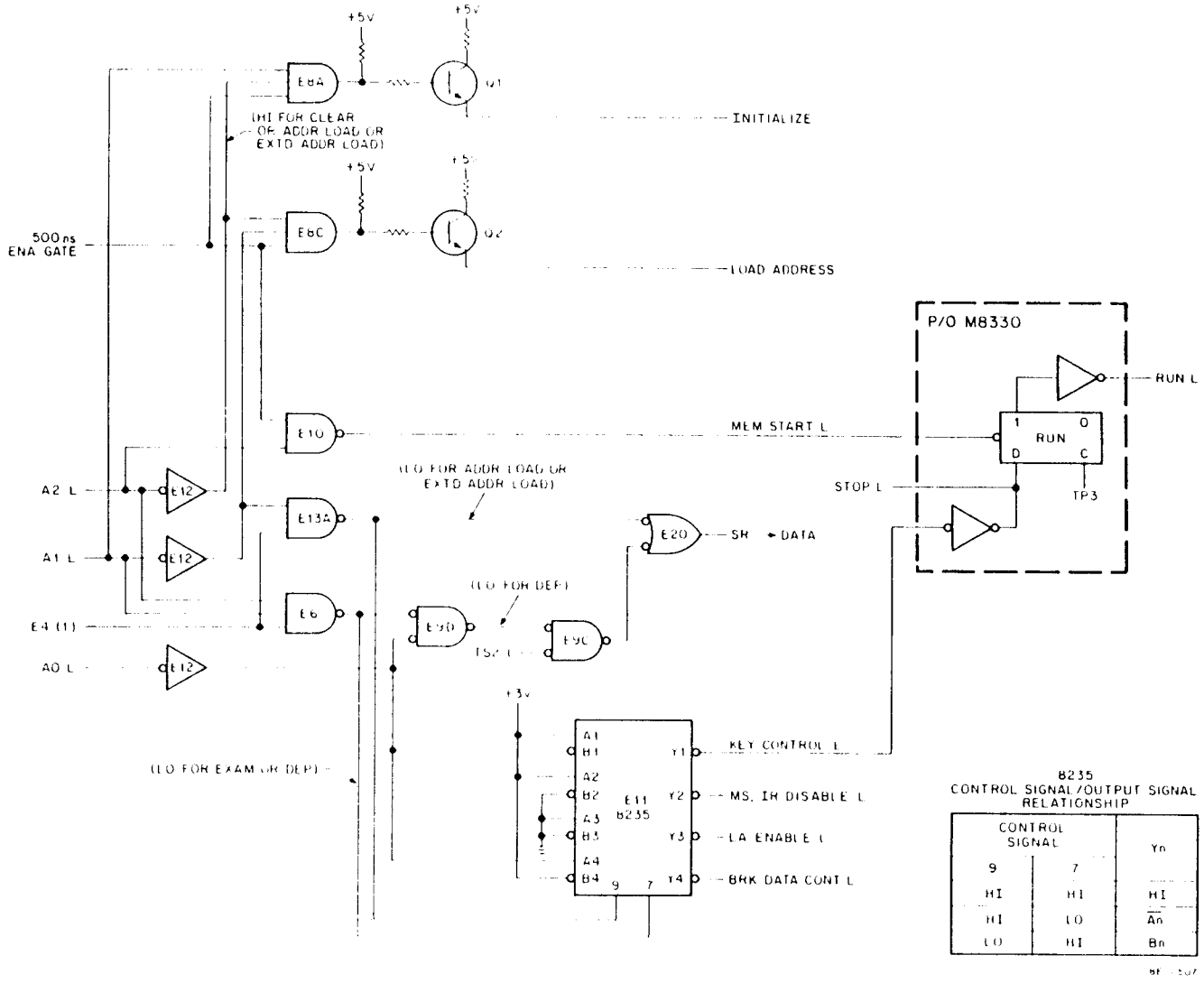


Figure 3-83 Operating Keys, Control Signal Logic

Table 3-4
KC8-FL CONTROL/ENABLE Signals

Key	ENABLE Signals					E11		CONTROL Signals
	A0 L	A1 L	A2 L	E4 (1)	500 ns ENA Gate	9	7	
ADDR LOAD	LO	LO	LO	HI	HI	LO	HI	LOAD ADDRESS, SR → DATA, MS, IR DISABLE L, LA ENABLE L
EXTD ADDR LOAD	HI	LO	LO	HI	HI	LO	HI	LOAD ADDRESS, SR → DATA, MS, IR DISABLE L, LA ENABLE L, KEY CONTROL L
CLEAR	HI	HI	LO	HI	HI	HI	HI	INITIALIZE
CONT	HI	LO	HI	HI	HI	HI	HI	MEM START L
EXAM	LO	HI	HI	HI	HI	HI	LO	MEM START L, KEY CONTROL L, MS, IR DISABLE L, BRK DATA CONT L
DEP	HI	HI	HI	HI	HI	HI	LO	MEM START L, KEY CONTROL L, MS, IR DISABLE L, SR → DATA

During TS2 of the timing cycle, the switch register data is gated onto the DATA 0–11 lines and the adder control signals enable a path through the adder network, adding an arithmetic 0 to each DATA bit. The switch register information is placed on the MAJOR REGISTERS BUS and loaded into the MB Register at TP2 time. At the same time, the MD DIR L signal is negated causing the content of the MB Register to be placed on the MD 0–11 lines. The data is then read into the memory location specified by the content of the CPMA Register.

At TP3 time the RUN flip-flop is reset (the STOP L signal is asserted when KEY CONTROL L is generated). The timing continues through TS4 and TP4 to halt in TS1 of some as yet unspecified cycle. Before the cycle is completed, one other operation is performed. The next consecutive address must be transferred from the PC Register to the CPMA Register. Therefore, the information in the PC Register is gated to one leg of the adder, and a 0 is added to each bit; the new address is placed on the MAJOR REGISTERS BUS and loaded into the CPMA Register at TP4 time. The cycle then ends.

EXAM Key – The EXAM key also initiates a timing cycle. By depressing this key the operator causes the content of a selected memory location to be brought from memory and loaded into the MB Register. Except for the absence of the SR → DATA signal, the TS1 operations are the same for the EXAM key as for the DEP key. The MS, IR DISABLE L, KEY CONTROL L, and MEM START L signals are asserted and function as described. However, the EXAM key also causes the BRK DATA CONT L signal to be asserted; this signal causes the TS2 operations to differ from those of the DEP key.

The BRK DATA CONT L signal gates the MD lines to the adder network, where a 0 is added to the data being brought from the memory location. The MD bits are placed on the MAJOR REGISTERS BUS and loaded into the MB Register at TP2 time (at the same time, the MD DIR L signal is negated, causing the register content to be again placed on the MD lines; thus, the content of the memory location can be re-written during the write half of the timing cycle). The operator can view the content of the MB Register by selecting the MD position on the front panel function selector switch. The operator can modify the data in the examined location by using the switch register and the DEP key. However, the EXAM cycle increments the PC Register to set up the next sequential memory address; therefore, to modify the data in the examined location, the switch register and ADDR LOAD key must be used to get back to the correct address.

CONT Key – The CONT key initiates a timing cycle by causing the MEM START L signal to be asserted. This is an important function because this is the *only* key that initiates repetitive timing cycles. Thus, the operator can begin automatic operation only by depressing the CONT key.

CLEAR Key – The CLEAR key generates the INITIALIZE signal. This signal clears the AC Register, the LINK, the Interrupt system, all peripheral flags, and various flip-flops within the basic system.

SING STEP Switch and HALT Switch – The operator can stop the PDP-8/E by closing either the SING STEP or the HALT switch. The logic is shown in Figure 3-84. If the SING STEP switch is used, the STOP L signal is generated and the first TP3 pulse clears the RUN flip-flop. The RUN L signal is negated, and the machine stops at the beginning of the next TS1 period. If the HALT switch is used, the STOP L signal is generated only when the F SET L signal is asserted. Because F SET L is asserted only when the next timing cycle is to be a FETCH cycle, the processor completes an entire instruction before halting in TS1. The operator can use these two switches and the CONT key to step a program one cycle or one instruction at a time.

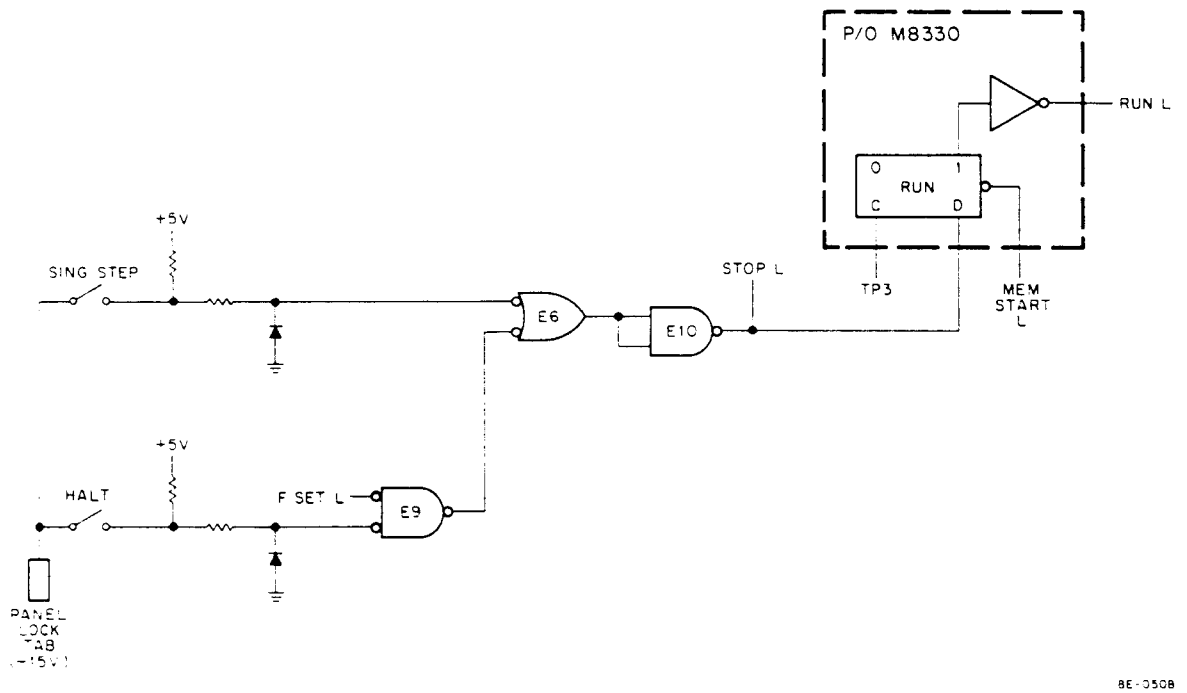


Figure 3-84 SING STEP and HALT Switches

3.33.2.2 **Display** – The KC8-FL Programmer's Console uses solid-state devices rather than filament-type lamps to indicate the state of various PDP-8/F (PDP-8/M) major registers and OMNIBUS signals. These indicators, light-emitting diodes (LEDs), are more durable than lamps and help to promote maintenance-free operation. LEDs differ from incandescent lamps in another important way. As the name implies, LEDs are diodes. Consequently, they exhibit the usual nearly constant forward voltage when conducting. A series resistor to define the forward current through the diode is necessary. In the KC8-FL, the resistor value of 330Ω establishes the LED on-current at about 10 mA. At currents of less than 500 μA, the diode does not emit detectable amounts of light.

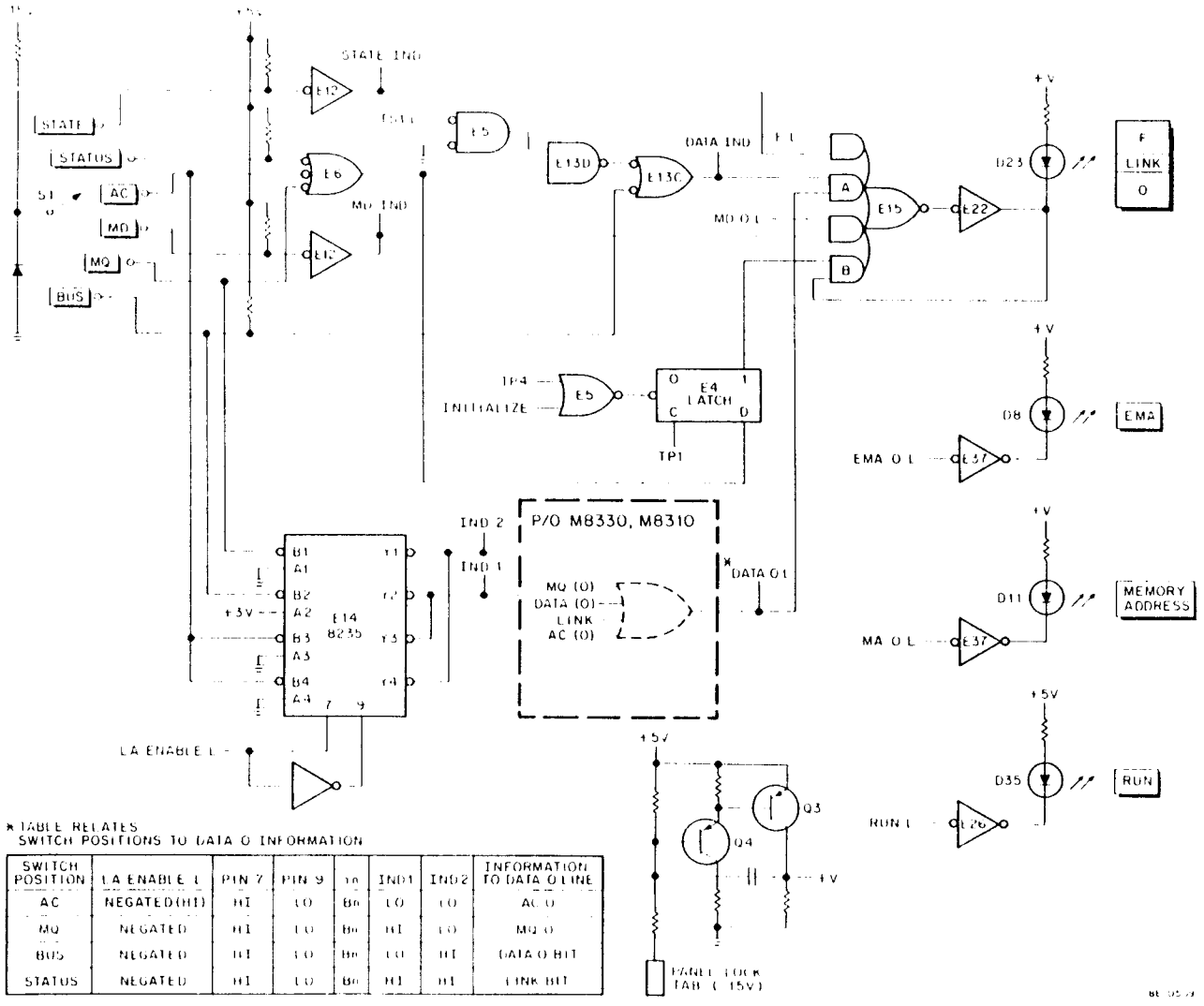
A single indicator, designated RUN, emits light when timing cycles are being generated. The light from this and all other indicators, is visible on the front panel (Figure 1-1). The other indicators are divided into two groups. One group, comprising 15 LEDs, displays the current memory address. This group is represented by the designations MEMORY ADDRESS and EMA. The other group, comprising 12 LEDs, displays the contents of selected registers and important OMNIBUS signals. A front panel switch enables the operator to select the register or type of OMNIBUS signal that he wishes displayed.

The display logic is represented in Figure 3-85. The RUN indicator is shown in the lower right of the figure. When the timing generator logic (M8330) asserts the RUN L signal, LED D35 emits light, indicating that the computer is running (power is on and timing cycles are being generated).

The two groups of indicators are represented by the 0 bit logic. For example, a complete 15-bit memory address has two 0 bits, EMA0 and MA0. Both bits are shown in the logic. When either the EMA0 L signal or the MA0 L signal is asserted, the +V voltage appears across the corresponding LED and its current determining resistor, causing it to light. The +V voltage, approximately +4.5V, is taken from the circuit that includes Q3 and Q4. When the front panel OFF/POWER/PANEL LOCK switch is in the PANEL LOCK position, the -15V supply is removed from the panel lock tab, thereby removing the +V voltage. Consequently, only the RUN indicator is lit with the switch in this position.

The majority of the logic in Figure 3-85 is used with the group of indicators that displays OMNIBUS signals and major register contents. Again, only the 0 bit logic is represented. The type of information displayed is selected on the front panel by a 6-position rotary switch, labeled S1 in Figure 3-85. This switch causes one of three signals to be asserted, depending on its position: in the MD position, MD IND is asserted; in the STATE position, STATE IND is asserted; in all other positions, DATA IND is asserted.

If the operator wants to monitor the MD lines, he switches to the MD position, causing the MD IND signal to be asserted. This signal is ANDed with the MD0 L signal in AND/NOR gate E15. The three other AND gates of E15 are disabled when S1 is in the MD position. If the MD0 L signal is asserted, E15 is disabled. Therefore, the output of inverter E22 is low. The +V voltage appears across LED D23, causing it to emit light that is visible on the front panel. The same analysis can be repeated for the F L signal when S1 is in the STATE position. However, the logic gating is more complicated when any of the other four switch positions is selected. The reason for this is that the four different types of signals must be carried on only the DATA lines; furthermore, remember that the DATA lines are time-shared and display information has access to these lines only during TS1.



* TABLE RELATES SWITCH POSITIONS TO DATA O INFORMATION

SWITCH POSITION	LA ENABLE L	PIN 7	PIN 9	to	IND1	IND2	INFORMATION TO DATA O LINE
AC	NEGATED(HI)	HI	LO	B6	LO	LO	AC O
MQ	NEGATED	HI	LO	B6	HI	LO	MQ O
BUS	NEGATED	HI	LO	B6	LO	HI	DATA O BIT
STATUS	NEGATED	HI	LO	B6	HI	HI	LINK BIT

Figure 3-85 Display Logic

Assume that the operator positions S1 at AC. The DATA IND signal is asserted during TS1. The AC0 bit must be gated to the DATA 0 line so that its state can be displayed. The table in Figure 3-85 shows that when the AC position is selected, the 8235 IC takes both IND1 and IND2 low. This combination of IND1 and IND2 causes the logic within the broken line (the actual logic is shown in Figure 3-86) to assert the AC → BUS L signal. Thus, the AC0 bit is gated to the DATA 0 line. If the AC0 bit is logic 1, AND gate A of E15 is disabled, the output of E15 stays high, and D23 emits light.

If the AC0 bit is logic 0, the LATCH flip-flop performs an important function. To illustrate, assume that the operator, while stepping the computer through a series of timing cycles (with the SINGLE STEP switch closed), causes a TAD instruction to be executed. The computer stops in TS1 of the cycle following the TAD instruction. The operator wants to check the result and switches to the AC position. D23 indicates logic 0. When the CONT key is depressed, AND gate A of E15 is disabled as soon as TS2 of the new timing cycle is entered. Therefore, if AND gate B and the LATCH flip-flop were not present, D23 would emit light for most of the timing cycle following the TAD execute cycle. Instead, the LATCH flip-flop is set at TP1 time (the D-input is high when the AC position is selected). The 1 output of the flip-flop and the high at the output of E22 keep E15 enabled; D23 remains dark throughout the timing cycle. At TP1 time the LATCH flip-flop is cleared.

Similarly, the content of the MQ Register, STATUS information, and BUS data can be displayed. The relationship between IND1/IND2 and the type of information displayed is indicated on Figure 3-85. Figure 3-86 shows the logic used to generate control signals in response to IND1 and IND2.

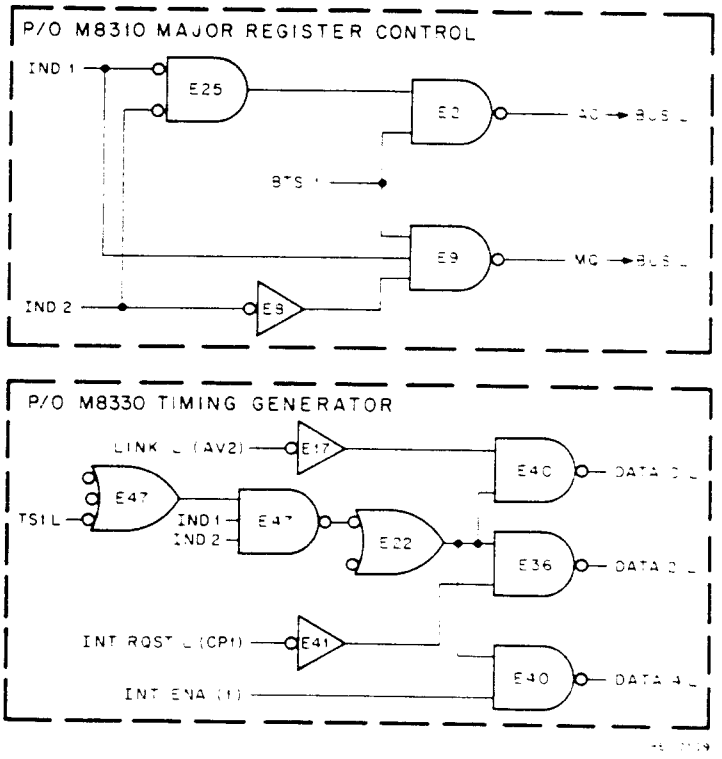


Figure 3-86 Data Bus Display Control Signals

3.34 MAJOR REGISTER GATING, BLOCK DIAGRAM DESCRIPTION

The major registers of the PDP-8/E perform all the operations required to implement program instructions. In retrospect, the PC Register keeps track of the program steps, the CPMA Register selects the memory location provided by the PC, and the AC Register uses the data in the selected memory location to carry out arithmetic operations. Information of one type or another (data, addresses, etc.) is continuously exchanged by the major registers. To effect these exchanges, the major register gating network is required. The major registers and the gating network are illustrated in the block diagram of Figure 3-87. This diagram presents the register flip-flops and the gating for one bit, bit 0, of the PDP-8/E 12-bit data word. All 12 data bits have similar gating networks; when differences exist, they will be noted in the discussion.

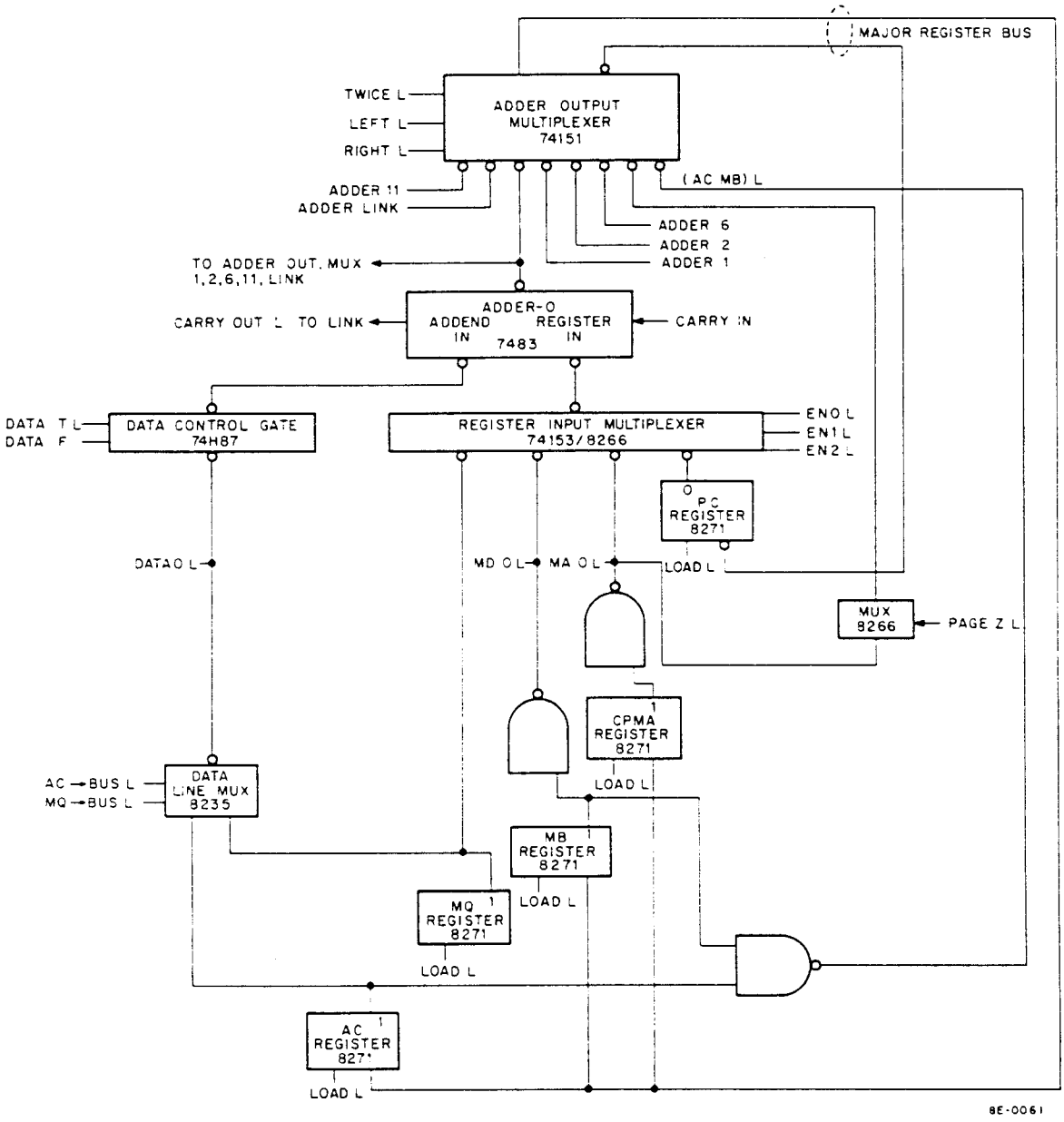


Figure 3-87 Major Registers and Gating, Block Diagram

Each major register is loaded with data that is transferred to the 0 bit flip-flop by the MAJOR REGISTERS BUS. The source of the data on this bus is the adder network. This network consists of the adder itself, a Register Input Multiplexer, an Adder Output Multiplexer, and a Data Control Gate which supplies an addend to the adder. The Register Input Multiplexer gates the output from a selected register flip-flop to the "register in" line of the adder. The data to be added to the 0 bit of this register is supplied by the Data Control Gate. The sum for the adder is transferred to the MAJOR REGISTERS BUS by the Output Multiplexer. A timing pulse then provides a load signal, which clocks the data into the desired register.

Note that the Adder Output Multiplexer can select any one of eight inputs for transfer to the register bus. Consider only the adder 0 input. Bit 0 of any register can be transferred to bit 0 of any other register. This is the case when the memory address in the PC Register is transferred to the CPMA Register at the completion of an instruction (PC → CPMA). Bit 0 of the PC is gated by the Register Input Multiplexer to the adder "register in" line. Because the bit must be transferred unaltered, an arithmetic 0 is supplied at the "addend in" line by the DATA T/DATA F control signals. The Adder Output Multiplexer places the unaltered bit on the MAJOR REGISTERS BUS; a load signal clocks the data into the 0 bit of the CPMA Register.

A similar operation takes place when the 0 bit is modified before transfer. For example, during an IOT instruction a peripheral can effect a relative jump of the program count (DATA + PC → PC). In this case, the "addend in" line must represent the state of the DATA 0 line and can be either a 0 or a 1. The sum of DATA 0 and PC0 is transferred to the MAJOR REGISTERS BUS; a load signal clocks the data into the 0 bit flip-flop of the PC.

If the Adder Output Multiplexer control signals (TWICE L, LEFT L, RIGHT L) select the "adder 1" input for transfer to the 0 bit MAJOR REGISTERS BUS, the RAL microinstruction (rotate AC and LINK left one place) is being implemented. Bit 1 of the AC Register is transferred to bit 0 of the AC Register (at the same time, AC bit 0 is transferred to the LINK). AC transfers are accomplished differently from the transfers of the other major registers. AC bit 0 is gated onto the DATA 0 L line by an AC → BUS signal, and through the data control gate to the adder. In this case, the Register Input Multiplexer provides an arithmetic 0 at the "register in" line. Bit 0 is then gated through the Adder Output Multiplexer that is associated with the LINK. Simultaneously, the output from adder 1 is gated onto the 0 bit MAJOR REGISTERS BUS. The AC LOAD L signal then clocks the data into the 0 bit of the AC Register.

Similarly, AC bits 11, 2, and 6, and the LINK bit can be transferred into bit 0 of the AC. The microinstructions implemented by those transfers are RTR, RTL, BSW, and RAR, respectively.

In addition to adder outputs, the Adder Output Multiplexer can gate two other signals onto the MAJOR REGISTERS BUS. One of these signals is (AC·MB) L, which is active when both MD0 L and AC0 L are at a positive voltage level (logic 1). When the basic AND instruction is implemented, the signal is gated through the Adder Output Multiplexer by the control signals. The logic 1 or 0 represented by the signal is then clocked into AC0.

The other signal that can be gated by the multiplexer is encountered during implementation of a Memory Reference Instruction (MRI). As the block diagram shows, this input is taken from a multiplexer that is controlled by the PAGE Z L signal. If the page address of the operand of the MRI is the same as that of the MRI, itself, the multiplexer output reflects the state of the MA0 line; however, if the operand is located on the 0 page, the multiplexer output is an arithmetic 0. In either case, the page data is placed on the MAJOR REGISTERS

BUS and loaded into CPMA0. This gating procedure only applies to register bits 0 through 4. If register bits 5 through 11 are considered, only a relative address must be considered. Therefore, the input to the Adder Output Multiplexer is taken directly from the appropriate MD line, rather than from the MA line via the PAGE Z multiplexer. A further discussion is presented when the logic diagrams are considered in detail.

3.34.1 Major State Register

PDP-8/E operations are grouped functionally into the four major states already introduced. FETCH (F), DEFER (D), and EXECUTE (E) are entered actively by setting flip-flop in the Major State Register. The fourth state, Direct Memory Access (DMA), results when none of the other three has been entered.

The Major State Register logic is shown in Figure 3-88. Flip-flops E45 and E50 are used to generate the F, D, and E states. The outputs of E45 and E50 are NANDed with a control signal called MS, IR DISABLE L. When this signal is negated (at +3V), one of the NAND gates (only one major state flip-flop may be active at any given time) is enabled; either F, D, or E is asserted. On the other hand, if MS, IR DISABLE L is asserted, none of these three states is entered. Thus, the computer is in the Direct Memory Access (DMA) state. This state is entered when data is to be transferred between memory and a data break peripheral, or between memory and the front panel. In either case, this state bypasses the processor logic that is normally used to access memory.

The major state flip-flops, E45 and E50, are clocked by the inverted CPMA LOAD L signal. This signal is produced when the front panel LOAD key is depressed during manual operation. In addition, the signal is generated by each TP4 pulse during automatic operation. Some type of manual operation always precedes automatic operation; for example, to initiate automatic operation of a stored program, the operator must load the CPMA with the starting address of the program. This address is loaded by depressing the LOAD key, an action that produces the CPMA LOAD pulse. However, this action also produces MS, IR DISABLE L. In fact, MS, IR DISABLE L is produced some 20 ms before CPMA LOAD. MS, IR DISABLE L causes the logic to generate a control signal, F SET L, which is applied to the data (D) input of the FETCH state flip-flop. Thus, when CPMA LOAD is produced, it clocks the flip-flop to its reset state, and the next cycle of operation will be a FETCH state. If the operator now depresses the CONT key, the computer begins the FETCH operation specified by the instruction just addressed.

The F SET L signal is asserted when computer operation requires that the FETCH state be entered. As previously indicated, FETCH is always entered on completion of manual operation. FETCH can also be entered on completion of a data break operation. Both manual and data break operations take place in the DMA state. Entry to and exit from this state is discussed fully in a following paragraph; a third method of entering the FETCH state is discussed in the following paragraph.

All instructions start in the FETCH state; consequently, this state can be entered from any state that completes an instruction. FETCH is entered from the EXECUTE state at the completion of a two- or three-cycle instruction; FETCH is entered from the DEFER state at the completion of a two-cycle instruction; finally, FETCH is entered from FETCH, itself, at the completion of a one-cycle instruction.

Figure 3-88 shows that F SET L is asserted when both inputs to NAND gate E49 are logic 0 (positive voltage level). Note that this condition arises only if NAND gates E30, E31, and E35 are disabled (ignore FE SET L and FD SET L for this discussion). If the computer is entering the EXECUTE state of a two- or three-cycle instruction, E is asserted at pin 4 of NAND gate E49, while both BD and BF are negated. E30, E31 and E35 are disabled, and CPMA LOAD L (produced by TP4) sets the FETCH flip-flop. If the computer is in the DEFER state of a two-cycle instruction, BF is negated, and E30 and E31 are disabled. E35 is likewise disabled in this

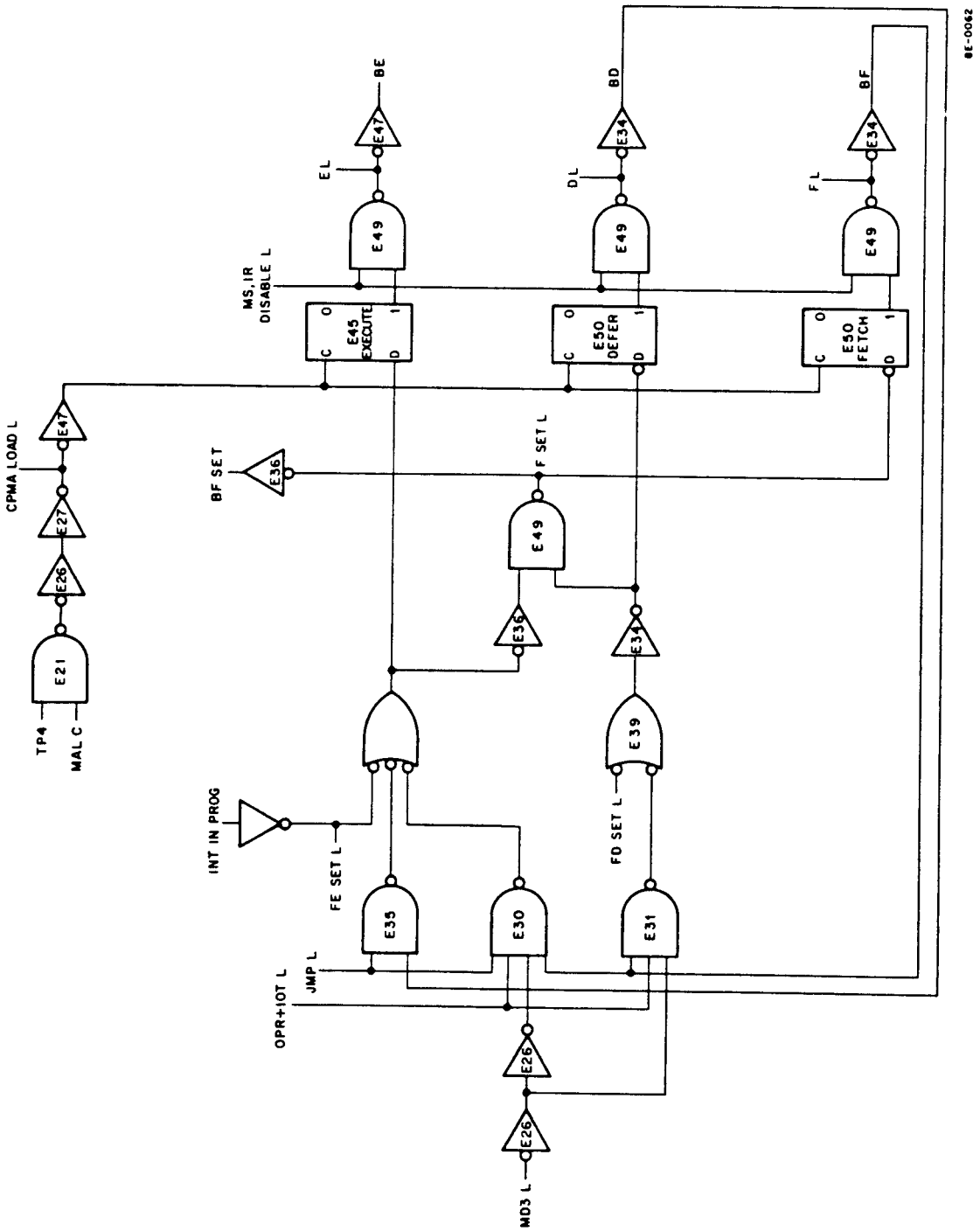


Figure 3-88 Major State Register Logic

;ase, because a two-cycle instruction involving the DEFER state can only be a JMP instruction. Therefore, the JMP line is asserted low and, again, all three NAND gates are disabled. Finally, if the computer is performing a one-cycle instruction (a FETCH state), BD is negated, thereby disabling E35. A one-cycle instruction can be an OPR microinstruction, an IOT instruction, or a directly addressed JMP instruction. If the instruction is either of the first two, the OPR + IOT line is grounded to disable gates E30 and E31. However, if the instruction is a directly addressed JMP, the JMP line is grounded to disable E30, while the negated MD3 L signal disables E31.

When the FETCH state of a multicycle instruction has been completed, either a DEFER or EXECUTE state follows, unless the operator depresses the SING STEP key or a data break device causes suspension of program control. Either circumstance can cause a halt or interruption when the FETCH state is completed; then the DMA state is entered. When the operations in this state have been carried out, control is returned to the program, and the multicycle instruction can be completed.

Consider, at this time, normal operation under program control, and assume that the instruction being processed is either a two-cycle instruction involving the DEFER state, or a three-cycle instruction. Operations are being carried out in the FETCH state. The DEFER state is the next state to be entered. The data line of the DEFER flip-flop must be asserted low, if the flip-flop is to be set at TP4 of FETCH; thus, NAND gate E35 must be enabled (again, disregard FD SET L). Because the FETCH state of a multicycle instruction is being performed, BF is high and OPR + IOT L is high. The DEFER state is entered when indirect addressing is required by the memory location of the operand. Such addressing is indicated when MD3 L is asserted, thereby providing the third high level for NAND gate E31. TP4 sets the DEFER flip-flop, and operations are then carried out in the DEFER state.

If the instruction is complete at the end of the DEFER state, the FETCH state is entered; however, a three-cycle instruction requires that EXECUTE be entered after DEFER. This requirement is met if NAND gate E35 is enabled. With BD asserted high and JMP negated (some instruction other than JMP is being performed and FE SET L is disregarded), E35 is enabled and TP4 sets the EXECUTE flip-flop. When EXECUTE is complete, the FETCH state is again entered. If the multicycle instruction being carried out does not require indirect addressing, the computer goes directly from FETCH to EXECUTE; in this event, NAND gate E30 is enabled, because MD3 L is negated high.

The preceding discussion disregarded the signals FD SET L and FE SET L. These signals are asserted by the EAE option; their functions are discussed in Volume 2 of this manual. However, note that the INT IN PROG L signal also can assert the FE SET line. The INT IN PROG L signal is asserted high when an interrupt request is honored by the interrupt logic of the timing generator module. The data line of the EXECUTE state flip-flop is then high, and TP4 forces the EXECUTE state to be entered next. FE SET can be asserted in this manner during either a FETCH, DEFER, or EXECUTE state, provided that the particular state is the final state of an instruction.

FETCH can be entered from the DMA state. The DMA state is used during manual operation and for data breaks. This state is entered when the MS, IR DISABLE L line is asserted. If this line is asserted because manual operations are being performed, the FETCH state always follows the DMA state. MS, IR DISABLE L disables NAND gates E49, thereby asserting F SET L (Figure 3-88). If the EXAM or DEP key has been depressed, TP4 and MA, MS LOAD CONT L (the latter is negated during manual operations) produce CPMA LOAD L. If the LOAD key has been depressed, PULSE LA H produces CPMA LOAD L. In either case, the FETCH flip-flop is reset and automatic operation begins in the FETCH state.

If MS, IR DISABLE L has been asserted by a data break peripheral, the FETCH state may or may not follow the DMA state. A data break operation can begin at the end of any major state. Program control is halted for one timing cycle (control may be halted for three cycles, as well; for convenience, a halt of one cycle is considered). When the data transfer has been completed, program control is re-established and the previously interrupted operation continues. An example of the interrupting processor is given in the following paragraph.

If operations are being carried out in the FETCH state of a two-cycle DCA instruction, the data line of the EXECUTE flip-flop is at a positive voltage level. If a peripheral initiates a data break during this FETCH state, MS, IR DISABLE L is asserted at TP4. At the same time, CPMA LOAD L is produced by TP4 and MA, MS LOAD CONT L and the EXECUTE flip-flop is set, but E, at the output of E49, remains negated. The DMA state is entered, instead of the EXECUTE state, and data transfer begins. Because the E49 NAND gates are disabled, F SET L is asserted. It could be assumed that the next TP4 pulse (of the DMA state) would reset the FETCH flip-flop; however, at TP1 of the DMA state the peripheral asserts MA, MS LOAD CONT L, thereby preventing the TP4 pulse in question from resetting the FETCH flip-flop. Instead, this TP4 negates MS, IR DISABLE L. The EXECUTE flip-flop is still set and, thus, E is asserted. Operations begin in the EXECUTE state of the interrupted instruction. At TP1 of this state, MA, MS LOAD CONT L is negated, completing the return to uninterrupted operation.

Consequently, because a data break can occur at the end of any major state, either FETCH, DEFER, or EXECUTE can be entered directly from the DMA state. The following section describes how the DMA state is used to suspend operation of the Instruction Register and decoder network during a data break operation.

3.34.2 Instruction Register

Operations within each major state are determined by the type of instruction that is contained in the addressed memory location. This instruction is placed on the MD lines, near the beginning of a FETCH cycle. MD bits 0, 1, and 2 are decoded by a network consisting, primarily, of an Instruction Register (IR) and a decoder IC. The decoder IC provides an output in response to the octal number represented by IR0 L, IR1 L, and IR2 L (MD0 L, MD1 L, and MD2 L, respectively). For example, the TAD instruction is represented as 1000 octal. Therefore, MD bits 0, 1, and 2 are 0, 0, and 1, respectively. The decimal 1 line is asserted by the decoder, causing TAD operations to begin. Other basic operations are initiated in the same manner. For a detailed presentation of the decoder integrated circuit, see Appendix A.

The decoding network, referred to as the IR decoder, is shown in Figure 3-89. Each of the IR decoder flip-flops, E40 and E45, is clocked by TP2 of the FETCH state. When the data (D) input of a flip-flop is grounded, indicating a logic 1 on the MD line in question, TP2 resets the flip-flop. When a flip-flop is reset, its corresponding IR line (0, 1, or 2) is asserted (grounded), provided that the MS, IR DISABLE L signal is negated. The eight possible combinations of the IR lines are decoded by the decoder module to produce one of the eight basic instructions shown.

If MS, IR DISABLE L is asserted (grounded), it removes the IR flip-flop outputs from the IR0 L, IR1 L, and IR2 L lines (it also causes the decoder to assert the AND L line; however, MS, IR DISABLE L forces the processor into the DMA state, and the basic instructions are not implemented during this major state). When MS, IR DISABLE L is negated, the IR lines return to their former state, and the interrupted instruction is again asserted.

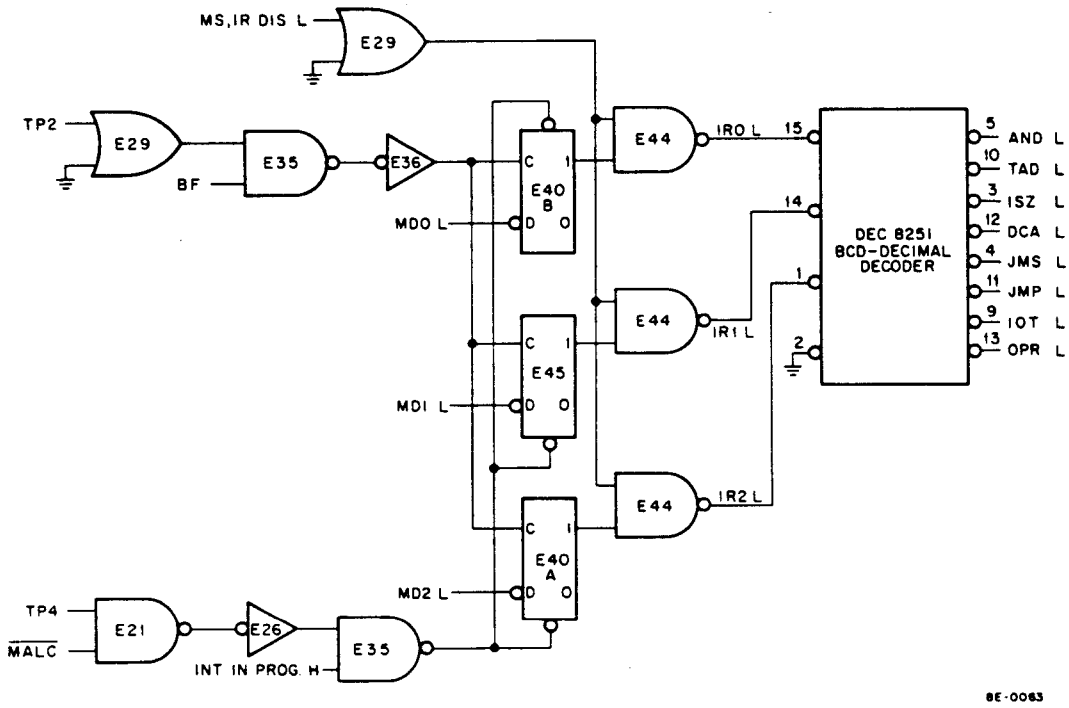


Figure 3-89 IR Decoder Logic

Note that flip-flop E40B can be dc-set and flip-flops E40A and E45 can be dc-cleared by a negative transition at the output of NAND gate E35. This transition occurs at TP4 time, if the INT IN PROG H signal has been asserted. This action forces the IR to 4 (JMS) and causes the decoder to assert the JMS line. Remember that INT IN PROG H also enables the data input of the EXECUTE flip-flop; thus, TP4 also sets the EXECUTE flip-flop. The processor enters the EXECUTE state and performs the JMS instruction, storing the program count in memory location 0, and entering the interrupt servicing routine. When the subroutine has been completed, the main program is resumed at the program count specified by memory location 0.

A data break device can request a break by asserting MS, IR DISABLE L at the same time (TP4) the IR flip-flops are dc-clocked to the JMS state. In this case, the data break device assumes control, and the processor enters the DMA state. On completion of the data break, the processor enters the EXECUTE state, performs the JMS instruction, and services the interrupting device, as outlined.

3.35 SOURCE CONTROL SIGNALS

3.35.1 Register In Enable Signals

The EN0 L, EN1 L, and EN2 L control signals are used to gate the 12 bits of the desired major register to the adder "register in" lines. Table 3-5 lists the usable combinations of these signals (in terms of voltage levels) and the register selected by each combination. The last entry of Table 3-5, which has EN0 L as a positive voltage level, gates an arithmetic 0 to the register in line. This condition occurs during TS3, when an OPR microinstruction or an IOT instruction is being carried out by the processor; in addition, a data break peripheral can cause such a condition to occur during TS2 of the DMA state.

Table 3-5
Register In Enable Signals

EN0 L	EN1 L	EN2 L	Input to Adder Register In Line
Low	Low	Low	PC Register
Low	Low	High	MD Lines
Low	High	Low	MQ Register (can be effected only by EAE option)
Low	High	High	CPMA Register
High	X	X	Arithmetic Zero

Note: X represents a "do not care" condition.

The logic gates used to generate the EN control signals are shown in Figure 3-90. To better understand processor operation, consider the logic diagrams as a means for understanding why a particular register is brought into the adder. For example, when all three signals are at a 0V level, the PC Register is gated to the adder. The processor flow diagram shows that the PC is transferred to the CPMA during TS4 of every major state. Thus, signals involved with TS4 should be examined.

The TS4 signal is the signal most likely to be used in gating the PC. This signal is applied to three separate NAND gates, E1A, E1B, and E3B. However, only E1B or E3B can, alone, cause all three EN signals to go to a ground level. E3B uses KEY CONTROL L as an enabling signal; KEY CONTROL L is asserted by the EXAM and DEP keys, both of which initiate a timing cycle. Thus, E3B produces the signals that gate the PC to the adder, thereby establishing a path from the PC to the CPMA during manual operation. E1B performs a NAND operation with two signals. One signal is a composite consisting of TS4, F SET L, and MS, IR DISABLE L. The F SET L signal indicates that the instruction is complete; MS, IR DISABLE L ensures that the current cycle is not a DMA state caused by a data break (if this is the case, the interrupted instruction may not have been completed prior to the data break; consequently, the PC must not be transferred to the CPMA). The other signal NANDed in E1B is (FE+FD) L. This signal is asserted either by the KE8-E option (operation in this circumstance is discussed in Volume 2), or by the INT IN PROG L signal. In the event that (FE+FD) L is negated, E1B produces the signals that establish a path from the PC to the CPMA during automatic operation.

If (FE+FD) L is asserted by INT IN PROG L, the PC is not transferred to the CPMA. Rather, the 0 location is loaded into the CPMA, and the processor goes into the EXECUTE state to carry out the JMS operation. During TS4 of this EXECUTE state, the PC is transferred to the CPMA.

In the process of carrying out the above JMS instruction, the processor must deposit the program count in location 0 (the PC must be transferred to the MB). Thus, the PC is gated to the adder. Inverter E8D causes the EN signals to be asserted during TS2 of the EXECUTE state of a JMS instruction. TS2 ensures that the PC data is on the MAJOR REGISTERS BUS before MB LOAD L is generated at TP2 time.

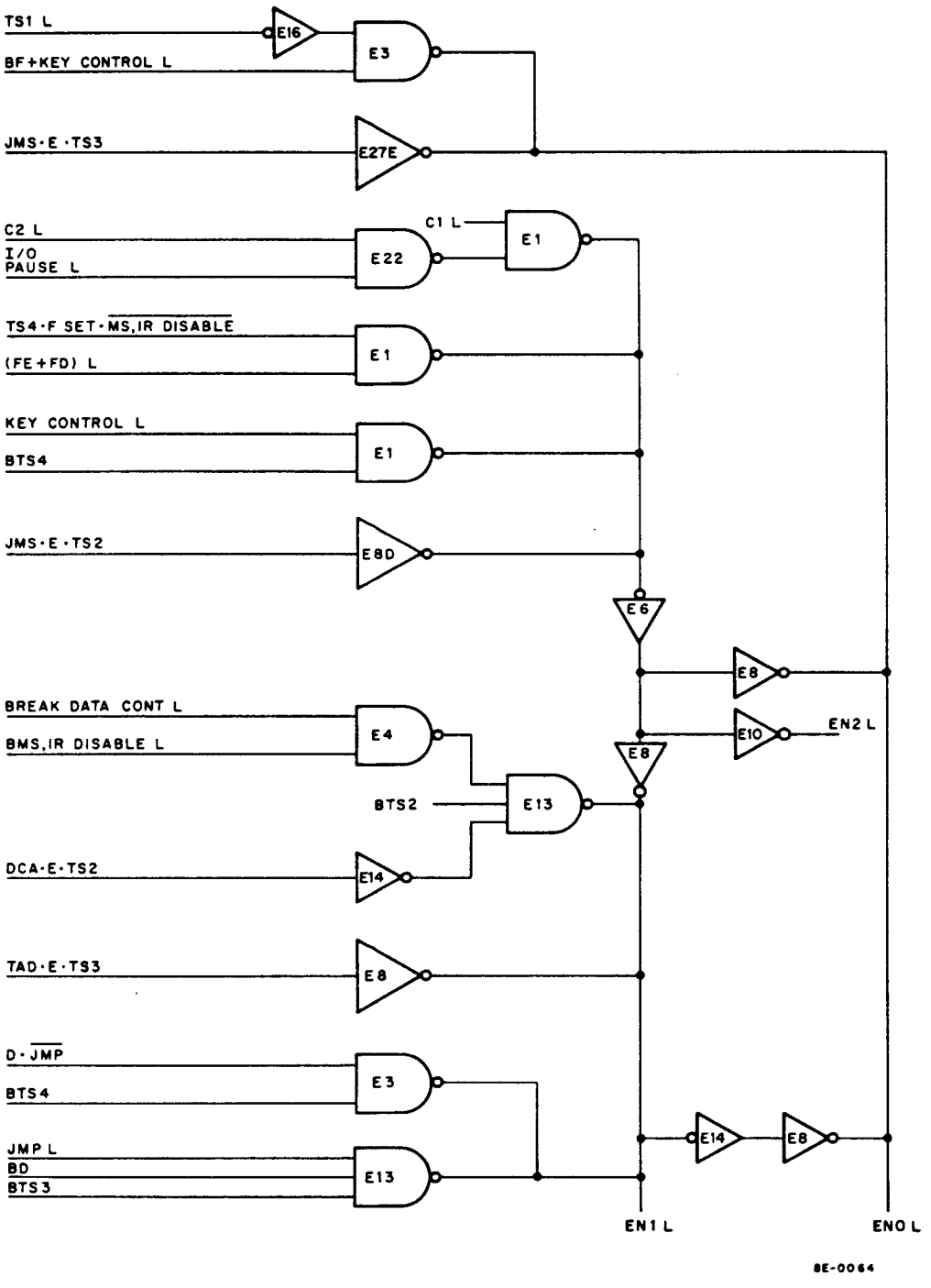


Figure 3-90 Register In Enable Logic

One remaining operation calls for the PC to be gated to the adder. A peripheral, through an IOT instruction, can change the program count in the PC Register by causing DATA+PC to the PC. NAND gates E22 and E1D assert the EN signals for this purpose. Further discussion of the C0 L, C1 L, C2 L, and I/O PAUSE L signal is presented in Section 6, I/O Transfer Logic.

If only EN0 is asserted, the CPMA Register is gated to the adder network (Table 3-5). As the logic indicates (Figure 3-90), the only way the processor can assert EN0, alone, is by enabling the wired-NOR consisting of NAND gate E3D and inverter E27. The signals used by this wired-NOR for gating the CPMA to the adder are described in the following paragraphs.

E3D is enabled during TS1 of the FETCH cycle or during TS1 when KEY CONTROL L is asserted (remember that the EXAM and DEP keys assert KEY CONTROL L). Thus, the CPMA is gated to the adder, making possible CPMA+1 to the PC during automatic and manual operation.

E27 asserts EN0 during TS3 of the EXECUTE state of a JMS instruction. The program count is stored in memory location 0000 when a JMS is forced by the INT IN PROG L signal. This sequence occurs during TS2 of the EXECUTE cycle. During TS3, the address in the CPMA, 0000, is incremented and sent to the PC. The new count, 0001, represents the address of the next instruction to be performed, and this instruction directs the processor to the starting address of the subroutine. Thus, when E27 asserts EN0, it allows CPMA+1 to be transferred to the PC during TS3 of the JMS instruction.

Table 3-5 indicates that the MQ Register can be gated to the "register in" line of the adder if EN0 and EN2 are brought to ground together. Note on the logic diagram (Figure 3-90) that this cannot be accomplished by any of the processor signals. MQ can be gated onto this line only by the KE8-E option (this feature is used by the SMA instruction). Thus, the remaining gates on the logic diagram are used to gate the MD lines to the adder, which is done by asserting both EN0 and EN1, while negating EN2. The MD lines are usually gated to the adder during TS2. The data from a memory location is transferred, either with or without modification, to the MAJOR REGISTERS BUS and loaded into the MB by TP2. NAND gate E13B causes EN1 and EN0 to be asserted during TS2 of all major states, with two exceptions. One of these exceptions occurs when the DCA instruction is being executed. To execute DCA, the processor must load the MB with the contents of the AC; thus, only the AC contents must be on the MAJOR REGISGERS BUS during TS2. The MB is loaded by TP2, the MD DIR L signal is negated, and the AC contents are written into the memory location. The other exception occurs during the DMA state (MS, IR DISABLE L is asserted), when the BRK DATA CONT L signal has not been asserted by a data break peripheral. Under these conditions, data is transferred by a peripheral to the DATA0 L–DATA11 L lines during TS2 and gated unchanged through the adder to the MAJOR REGISTERS BUS. At TP2 time, the data is loaded into the MB and then placed on the MD lines.

The processor flow diagram (Section 2) illustrates the operations that require the MD lines to be gated to the adder. MD to the MB is carried out during the FETCH state and during the non-autoindexed DEFER state, though the operation is meaningless. Because MD to the MB does not disrupt processor operations, it is more convenient to allow, rather than inhibit, its occurrence.

The MD lines are gated to the adder in three other instances: (a) During TS3, when a TAD instruction is being executed, both the MD lines and the contents of the AC are gated to the adder. The resulting sum is then placed on the MAJOR REGISTERS BUS and loaded into the AC by TP3. Inverter E8E allows EN0 and EN1 to be asserted in response to the (TAD•E•TS3) L signal. (b) When the processor is in the DEFER state of an indirectly addressed JMP instruction, the contents of the PC Register are changed to address the location of the next instruction to be performed. This address, the effective address of the JMP instruction, is placed on the MD lines.

The MD lines are gated through the adder to the MAJOR REGISTERS BUS during TS3, and then loaded into the PC by TP3. NAND gate E13A allows EN0 and EN1 to be asserted by the BTS3, BD, and JMP L signals. (c) MD → ADDER occurs also during the DEFER state; however, this instance specifically excludes the JMP instruction. If an AND, TAD, ISZ, DCA, or a JMS instruction has been indirectly addressed, the effective address is loaded in the CPMA, rather than the PC. The processor then goes to the EXECUTE state to operate on the data in the effective address. Thus, NAND gate E3B is enabled during TS4, thereby placing the effective address, contained on the MD lines, on the MAJOR REGISTERS BUS. TP4 then loads the address into the CPMA.

3.35.2 Data Line Enable Signals

The AC → BUS and MQ → BUS control signals are used to gate the 12 bits of the AC or MQ Registers to the DATA lines. The selected register can then be gated through the Data Control Gate to the "addend in" line of the adder.

The logic elements that produce MQ → BUS are shown in Figure 3-91. NAND gate E9A is used when the function select switch on the Programmer's Console is set to the MQ position. The console logic takes the IND2 line to ground and the IND1 line to a positive voltage. During TS1, the contents of the MQ are placed on the DATA 0 – DATA 11 lines and displayed on the Programmer's Console.

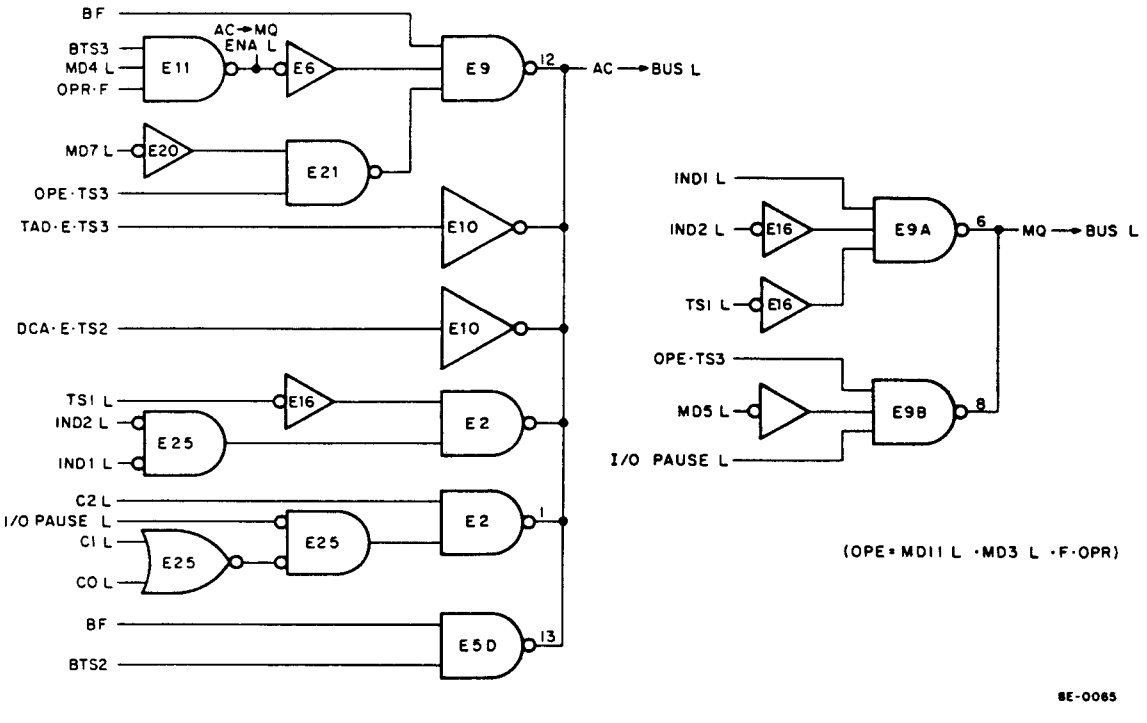


Figure 3-91 Data Line Enable Logic

NAND gate E9B is used to gate the MQ to the DATA lines for certain MQ microinstructions. Thus, this gate is enabled, provided the MQ BUS INH L signal is negated, for the following instructions: MQA, SWP, ACL, and CLA, SWP. The MQA instruction asserts both MQ → BUS and AC → BUS, a condition which causes an inclusive-OR to be performed by the Data Line Multiplexer. Each of the other MQ microinstructions asserts only MQ → BUS. The MQ contents are then gated through the data control gate to the adder, placed on the MAJOR REGISTERS BUS, and loaded into the AC by TP3.

The logic that produces AC → BUS is also shown in Figure 3-91. NAND gate E9C is used to gate the AC to the DATA lines during operate microinstructions. The AC contents are then gated to the adder, and the result of the arithmetic operation is loaded into the AC by TP3.

Inverters E10A and E10F assert AC → BUS during the EXECUTE state. E10A allows the AC to be gated to the adder during TS3 of a TAD instruction; also during TS3, the contents of the MD lines are gated to the adder by the Register In Enable logic. Thus, the 2's complement adding operation is effected, and the result is stored in the AC. E10F is used during TS2 of a DCA instruction. The contents of the AC are to be deposited in a memory location. To accomplish this, the contents of the AC must be placed on the MAJOR REGISTERS BUS before TP2. At TP2 time, the MB is loaded and the MD DIR L signal is negated, placing the AC contents on the MD lines in preparation for the WRITE operation.

NAND gate E5 asserts AC → BUS during TS2 of each FETCH cycle. This condition is not required by an instruction; however, it establishes the state of the AC lines at the positive I/O interface early in the FETCH cycle in order to maintain timing compatability with some older peripherals.

NAND gate E2C is used when the operator wishes to monitor the contents of the AC Register. During TS1, this information is placed on the DATA lines and used by the Programmer's Console logic.

E2B asserts AC → BUS when an IOT instruction directs a data transfer to a peripheral. In this case, I/O PAUSE L is asserted by the processor, the correct C lines are asserted by the peripheral, and the information on the DATA lines is strobed into a peripheral Buffer Register.

3.35.3 Data Enable Signals

The data enable signals, DATA T and DATA F, are used to determine the input for the addend in line of the adder. As indicated in the discussion of the data line enable signals, this input can be AC information, MQ information, or a combination of the two. The input can also be information that is being transferred from a peripheral to either the AC, PC, or MB. One other possibility exists: some operations require that the addend in line be an arithmetic 0. Therefore, the DATA T/DATA F signals are used to gate, to the adder, either the information carried on the DATA lines or an arithmetic 0. Table 3-6 is a truth table of these two signals (in terms of voltage levels); Figure 3-92 shows the logic that implements the truth table.

Table 3-6
Data Enable Signals

Signal		Explanation
DATA T	DATA F	Input to Adder "addend in" line
Low	Low	DATA LINES (complement of contents of AC or MQ Registers, I/O XFER DATA)
Low	High	DATA LINES (contents of AC or MQ Registers, I/O TRANSFER DATA)
High	High	7777 (not used)
High	Low	Arithmetic 0

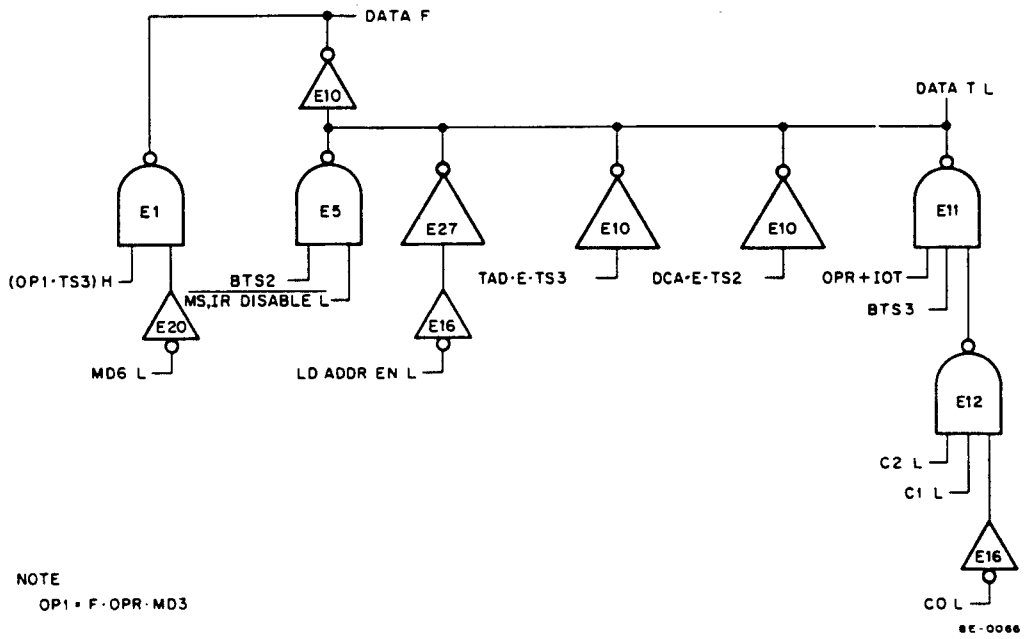


Figure 3-92 Data Enable Logic

Information on the DATA lines is often a result of implementation of an OPR or an IOT instruction. NAND gate E11 is used to assert DATA T in either case. If an IOT instruction has been decoded, DATA T is asserted, and DATA F remains negated. The transfer can be either into or out of the processor, depending on which C control line, if any, is asserted by the peripheral. If the C lines specify a transfer into the processor, the information is placed on the DATA lines by the peripheral, gated through the adder, and placed on the MAJOR REGISTERS BUS; the information is loaded into either the AC or the PC at TP3. If the transfer is out of the processor, the AC information on the DATA lines is gated into the peripheral's Buffer Register. However, the information must also be retained in the AC and, therefore, it is gated through the MAJOR REGISTERS BUS and again loaded into the AC by TP3.

In one case, an IOT instruction does not cause DATA T to be asserted via E11. If the peripheral asserts the CO L line, and leaves C1 L and C2 L negated, NAND gate E12 is enabled. Thus E11 is disabled, and, because no other gate enables DATA T under these circumstances, an arithmetic 0 is gated to the addend in line of the adder. This action results in an output data transfer and also a clearing of the AC, because 0s are loaded at TP3.

If an OPR microinstruction has been decoded, E11 is again used to assert DATA T, while DATA F is left negated. However, several OPR microinstructions (CMA, for example) are implemented by gating the complement of the signal on the DATA lines to the adder. This can be done if the DATA F signal is grounded along with DATA T. As Figure 3-92 shows, NAND gate E1 grounds DATA F; it does this only for those OPR microinstructions (CMA, CIA, CLA, CMA) that require the AC to be complemented.

DATA T can be asserted by any one of four other signals. One signal is (DCA·E·TS2) L, which asserts both AC → BUS and DATA T. Thus, the AC information is gated to the MAJOR REGISTERS BUS and loaded into the MB by TP2. After the AC information is transferred, the AC must be cleared by negating DATA T and, thereby, placing a 0 on the "addend line" line. Because a 0 is also placed on the "register in" line during TS3 of a DCA instruction, a 0 is placed on the MAJOR REGISTERS BUS. This is then loaded into the AC by TP3.

Another signal group that asserts both AC → BUS and DATA T is TAD·E·TS3. Thus, the AC information is gated to the adder, where it is added to the contents of a memory location; the result is then loaded into the AC by TP3.

The LOAD ADDR EN L signal asserts DATA T during manual loading of an address. The operator sets an address on the console switch register and depresses the LOAD ADDR key. This action places the address on the DATA lines, and DATA T gates it to the MAJOR REGISTERS BUS. The address is then loaded into the CPMA by the CPMA LOAD L signal.

The last method of asserting DATA T is with NAND gate E35. This gate provides a path to the adder for data that is being transferred to the MB, either from a data break peripheral or from the console switch register. TP2 then loads the data from the MAJOR REGISTERS BUS into the MB.

3.36 ROUTE CONTROL SIGNALS

3.36.1 Carry In Logic

The Carry In logic is shown in Figure 3-93. The CARRY IN L control signal enables the processor to increment data. The block diagram in Figure 3-87 shows that a CARRY IN L signal is applied to the adder IC (DEC 7483). This IC is a full adder and, therefore, can perform a complete addition by adding three inputs: an augend, an addend, and a carry from a previous order. In the PDP-8/E, the CARRY IN line of Figure 3-93 is applied directly to the carry input of adder 11, the LSB of the 12-bit data word. The carry from adder 11 is applied to adder 10, and so on, until the carry from adder 0 is applied to the LINK adder (Link logic is discussed in Paragraph 3.39).

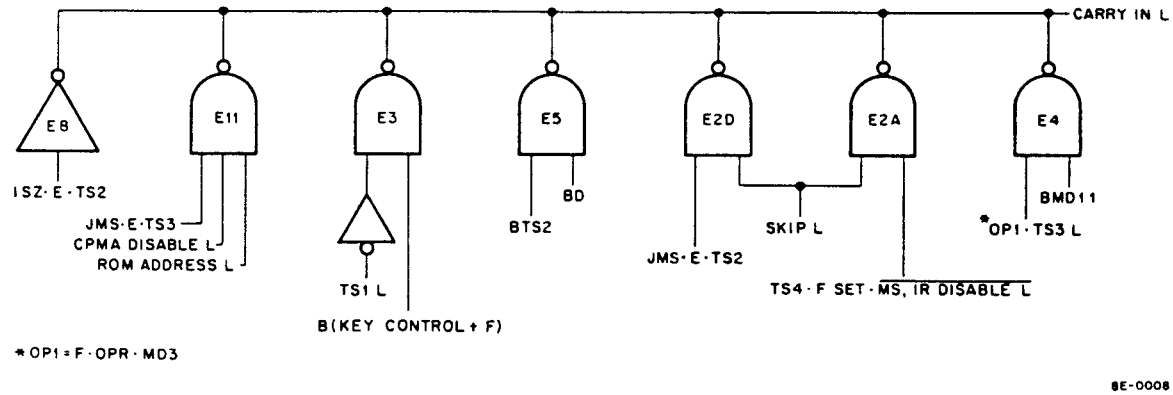


Figure 3-93 Carry In Logic

Data is incremented during various operations which were introduced in the discussion of the source control signals. For example, the address carried on the MA lines is incremented during TS1 of the FETCH cycle. This new address is then loaded into the PC Register at TP1, thereby updating the program counter. The operation is represented symbolically as MA+1 → PC and is carried out as follows: (a) the data on the MA lines is gated to the register in line of the adder (the augend); (b) a 0 is gated to the addend in line of the adder (the addend); and (c) a 1 is gated to the CARRY IN line of the adder by NAND gate E3. The result, MA+1, is placed on the MAJOR REGISTERS BUS and loaded into the PC by TP1; because this must happen at the beginning of each instruction, E3 is enabled by TS1 L and F L.

NAND gate E3 asserts CARRY IN L during automatic operation and during those manual operations that cause the KEY CONTROL L signal to be generated (Paragraph 3.33). Remember that KEY CONTROL L can be asserted by three Programmer's Console keys: DEP, EXAM, and EXTD ADDR LOAD. Because both DEP and EXAM initiate a single timing cycle, the PC is updated by 1 (MA+1 → PC) each time either of these keys is activated. The EXTD ADDR LOAD key also asserts KEY CONTROL L and, thus, CARRY IN L; however, both assertions are meaningless in the present application, because no timing cycle is initiated by EXTD ADDR LOAD L.

The program count must also be incremented during the EXECUTE cycle of a JMS instruction. The program count is stored in the memory location specified by the operand of the JMS instruction, e.g., location Y. The processor then proceeds to the first instruction of the subroutine. The address of this first instruction is contained in location Y+1; thus, the PC must be incremented and, as in the last example, the operation is represented symbolically as MA+1 → PC. This operation is accomplished during TS3 by NAND gate E11. The ROM ADDRESS L and CPMA DISABLE L signals are asserted if the MR8-E (Read-Only Memory) option is included in the system. Use of CARRY IN L with this option or with any option designed to assert CPMA DISABLE L only is discussed in Volume 2.

Inverter E8 asserts CARRY IN L in response to the signal ISZ·E·TS2. The ISZ instruction directs the processor to increment the data in the specified location and then skip the next instruction if the result of the incrementation is 0000. To increment the data, the processor performs the operation MD+1 → MB. The CARRY IN L line is asserted during TS2 of the EXECUTE cycle so that MD+1 can be loaded into the MB at TP2 time. The second portion of the ISZ instruction is carried out as described in Paragraph 3.38.

NAND gate E5 also asserts CARRY IN L. E5 does this during TS2 of a DEFER cycle and as part of the operation MD+1 → MB. Remember that this operation has significance during the DEFER cycle only if an Autoindex Register (locations 0010–0017) has been referenced by the instruction being performed. When this is the case, the content of this register is incremented before it is used as the operand. Autoindexing is discussed fully in Chapter 4 of the *PDP-8/E & PDP-8/M Small Computer Handbook.*

NAND gates E2A and E2D are used with the Skip logic (Paragraph 3.38) to assert CARRY IN L. Gate E2A asserts CARRY IN L when a skip of one program instruction (symbolized as PC+1 → CPMA) is required. This requirement arises if: (a) a Group 2 operate microinstruction such as SKP (skip unconditionally) is programmed; (b) if an IOT instruction causes a peripheral to assert the OMNIBUS SKIP L signal; and (c) as a result of the ISZ instruction. In each of these circumstances, TP3 sets the SKIP flip-flop, E13 (Paragraph 3.38), thereby asserting the SKIP (1) line high. During TS4, CARRY IN L is asserted, the PC is gated to the adder and incremented, and the result is placed on the MAJOR REGISTERS BUS and loaded into the CPMA at TP4. The F SET L portion of the enabling signal at E2A ensures that CARRY IN L is asserted only during the last cycle of the instruction. MS, IR DISABLE L is high and ensures that CARRY IN L is not asserted during TS4 of a DMA state. For example, if the program is stopped by the HALT key, the SKIP flip-flop can remain set. Then, if the operator pushes the EXAM key, CARRY IN L is asserted and PC+1, rather than PC only, is loaded into the CPMA at TP4 time of the memory cycle. MS, IR DISABLE L prevents this error from occurring because it is high.

NAND gate E2D is used to assert CARRY IN L if a program interrupt prevents gate E2A from asserting it. For example, if a Group 2 OPR microinstruction causes the SKIP flip-flop to be set at TP3, the PC should be incremented during TS4. However, if the interrupt system logic located in the Timing Generator has honored a peripheral's interrupt request, the OMNIBUS INT IN PROG H signal is also asserted at TP3. The assertion of INT IN PROG H prevents the PC from being gated to the adder network during TS4 (Paragraph 3.35.1). The CPMA is

loaded with 0s, rather than being loaded with the incremented PC contents at TP4. The processor then enters the EXECUTE state. During TS2, CARRY IN L is asserted by E2D, and the PC is gated to the adder and incremented. At TP2 the result (PC+1) is stored in location 0, and the SKIP flip-flop is reset. When the interrupt has been serviced, a JMP I to location 0 causes the main program to be resumed at location PC+1 (the original location if the interrupt had not been requested).

The last gate to consider in the Carry In logic is NAND gate E4. This gate is used during TS3 of a FETCH cycle to carry out operate instructions involving the IAC command. If IAC has been programmed, the contents of the AC are placed on the DATA lines during TS3 and gated through the Data Control Gate to the adder. A 0 is gated to the adder's register in line. E4 asserts CARRY IN L; the result, AC+1, is placed on the MAJOR REGISTERS BUS and loaded into the AC by TP3.

If CIA has been programmed, the operation is similar to that just outlined. However, the data in the AC is complemented by the Data Control Gate before it is applied to the addend in line of the adder. Thus, the result, $AC_{complement}+1$, is loaded into the AC at TP3.

When CLA IAC is programmed, 0s are provided at the addend in and register in lines of the adder. When CARRY IN L is asserted, the resulting output from the adder is 0001_8. This is loaded into the AC at TP3, setting AC=1. Similar operations occur when the IAC command is microprogrammed with other Group 1 operate instructions.

3.36.2 Shift Control Signals

The shift logic is shown in Figure 3-94. The control signals produced by this logic, RIGHT L, LEFT L, TWICE L, and PAGE Z L, gate data to the MAJOR REGISTERS BUS. This data can be taken from an adder, from a NAND gate, or from the PAGE Z multiplexer, as illustrated by the block diagram of Figure 3-87.

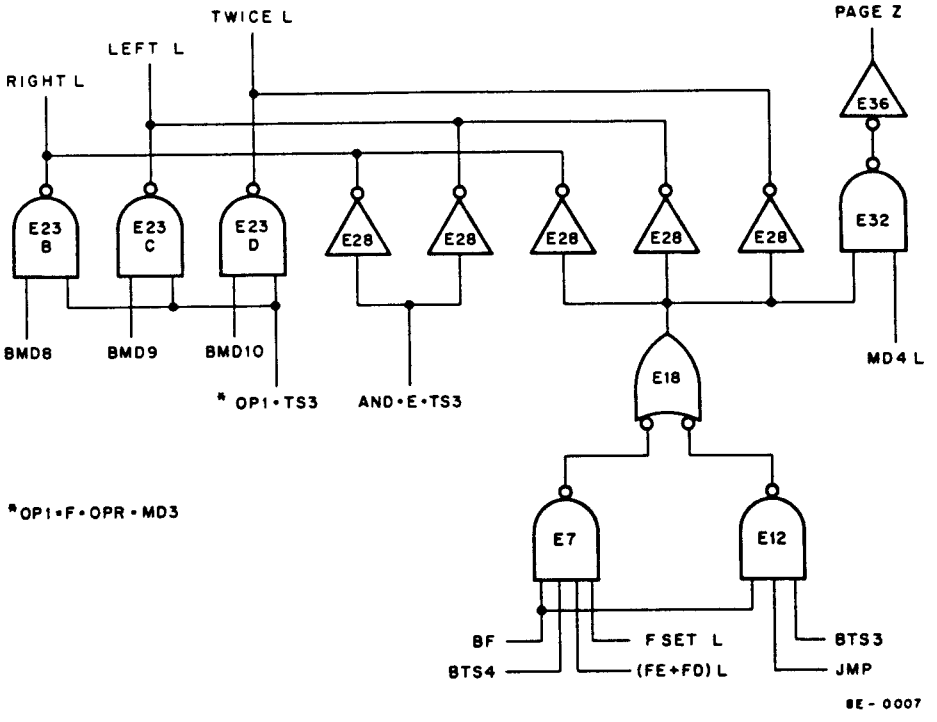


Figure 3-94 Shift Logic

A data bit can be transferred from one major register to another major register (Paragraph 3.34 where bit 0 was considered). To transfer bit 0 from one register to another, a "no shift" condition is needed by the Adder Output Multiplexer. As shown in Table 3-7, this condition arises when RIGHT L, LEFT L, and TWICE L are high, and PAGE Z L is low. Because most processor operations involve no shifting, this combination of signals normally prevails. A different combination can arise during either TS3 or TS4 of a FETCH cycle, with one exception. This exception, as shown on the logic diagram (Figure 3-94), occurs during TS3 of the EXECUTE cycle of an AND instruction and is discussed in the following paragraph. Thus, unless the program specifically requires a shifting operation, the shift signals remain as shown above. Shifting is a consequence of any one of five operate microinstructions (Paragraph 3.34). These microinstructions, RAR, RAL, RTR, RTL, and BSW, enable NAND gates E23B, E23C, and E23D in combinations that cause the desired shifting to occur. For example, if RAR (7010_8) (rotate AC and LINK right one) is programmed, NAND gate E23B is enabled during TS3 of the FETCH cycle. Bit X of the AC Register is placed on the DATA X line at the beginning of TS3, and gated through Data Control Gate X to adder X. A 0 is gated to the register in line of the adder. The adder CARRY IN line is negated. The adder X output is selected by Adder Output Multiplexer X+1, in response to the RIGHT L signal, and placed on MAJOR REGISTERS BUS X+1. At TP3 time, the data is loaded into bit X+1 of the AC Register. In the same manner, AC bit X can be shifted to the right twice (RTR) to AC bit X+2, to the right by six (BSW), or to the left in response to RTL, for example. All this shifting is accomplished by NAND gate E23.

RIGHT L, LEFT L, and TWICE L signals are used in operations other than shifting. Both RIGHT L and LEFT L are asserted by the composite signal AND•E•TS3, as illustrated in Figure 3-94. This combination enables each Adder Output Multiplexer to select the output of a NAND gate for transfer to the MAJOR REGISTERS BUS. This NAND gate is used to implement the basic AND instruction. During the EXECUTE cycle of this instruction, data in the addressed memory location is brought from memory and loaded into the MB Register at TP2 time. The MB Register output is logically NANDed with the AC Register output. During TS3, the logical result is gated through the Adder Output Multiplexer by RIGHT L and LEFT L signals to the MAJOR REGISTERS BUS and loaded into the AC at TP3 time.

Seven of the eight inputs to the Adder Output Multiplexer have been considered with respect to the shift signals. The eighth input is taken from the PAGE Z multiplexer (Figure 3-87). The gating networks for bits 0 through 4 use this multiplexer to provide the page address of the operand of either an MRI or an indirectly addressed JMP instruction. If this address is on the current page, the PAGE Z multiplexer output gates the data on MA lines 0 through 4 to the Adder Output Multiplexers. RIGHT L, LEFT L, and TWICE L signals gate these (0–4) bits to the MAJOR REGISTERS BUS. They are then loaded into CPMA 0 through 4, respectively, at TP4 time (this applies to an MRI; they are loaded into PC 0–4 and CPMA 0–4 for the JMP I instruction). Table 3-7, entry 8, lists the signal levels required to carry out this operation. This condition is implemented by gates E7, E13, E28, E32, and E36, shown in Figure 3-94. NAND E7 is enabled only during TS4 of a FETCH cycle. F SET L must be negated, ensuring that the instruction is either a JMP I or an MRI (if F SET L is negated, either a DEFER or an EXECUTE cycle follows the FETCH). The (FE+FD) L signal must also be negated. If the KE8-E option is not in the system, (FE+FD) L can be asserted only if the INT IN PROG H signal is asserted. Because INT IN PROG H is asserted only if F SET L is asserted, (FE+FD) L and F SET L are negated as a pair. The KE8-E option, by asserting FE SET L or FD SET L, can cause (FE+FD) L to be asserted even though F SET L is negated. Refer to Volume 2 for a further discussion.

Table 3-7
Shift Control Operations

RIGHT L	LEFT L	TWICE L	PAGE Z L	Input To Adder Output Multiplexer	Used For
HI	HI	HI	LO	ADDER X	No-Shift Operations
HI	LO	HI	LO	ADDER (X+1)	Rotate and Byte Swap Instructions
HI	LO	LO	LO	ADDER (X+2)	
LO	HI	HI	LO	ADDER (X-1)	
LO	HI	LO	LO	ADDER (X-2)	
HI	HI	LO	LO	ADDER (X+6)	
LO	LO	HI	LO	ACX, MBX → ACX·MBX	Logical AND Instruction (0000)
LO	LO	LO	LO	PAGE Z Multiplexer 0–4 to Adder Output Multiplexers 0–4. MA 0–4 loaded into CPMA 0–4 (or into PC 0–4).	Providing Page Address of operand of either an MRI or an indirectly addressed JMP instruction.
				MD Lines 0–11 to Adder Output Multiplexers 5–11, loaded into CPMA 5–11 (or PC 5–11).	Providing Relative Address of either of above.
LO	LO	LO	HI	PAGE Z Multiplexer 0–4 to Adder Output Multiplexer 0–4. 0s loaded into CPMA 0–4 (or into PC 0–4).	Providing Page Address
				MD lines 5–11 to Adder Output Multiplexer 5–11, loaded into CPMA 5–11 (or PC 5–11).	Providing Relative Address

NOR gate E18 is enabled, and the high level at its output causes inverter E28 to assert RIGHT L, LEFT L, and TWICE L. This high level is also applied to E32, where it is NANDed with the signal representing the state of MD4 L. If MD4 L is asserted (the address of the operand is on the current page) PAGE Z L is negated. The data on MA lines 0–4 is loaded into CPMA 0–4 at TP4 time. However, if MD4 L is negated (the address of the operand is on page 0), PAGE Z L is asserted. In this case, the PAGE Z multiplexer provides arithmetic 0s for the Adder Output Multiplexer; these 0s are loaded into CPMA 0–4 at TP4 time.

The PAGE Z multiplexer is also used to provide the page address of an indirectly addressed JMP instruction. This operation is carried out in the same manner as described for the MRI. If the JMP instruction is directly addressed, it is carried out in one cycle of operation. Thus, NAND gate E12 is used to assert the RIGHT L, LEFT L, and TWICE L signal lines. As before, MD4 L can be asserted or negated. If this line is asserted, the data on MA lines 0–4 is placed on the MAJOR REGISTERS BUS and loaded into the PC Register (bits 0–4, respectively) at TP3 time. If MD4 L is negated, 0s are loaded into PC 0–4.

To specify the complete address to which the program must jump, the relative address specified by bits 5 through 11 of the JMP instruction must be loaded into PC 5–11. Bits 5 through 11 of the major register gating do not use a PAGE Z multiplexer. Instead, MD lines 5–11 are connected directly to Adder Output Multiplexers 5–11. Whenever RIGHT L, LEFT L, and TWICE L are asserted by the JMP instruction, the data carried on MD lines 5–11 is selected by the Adder Output Multiplexer, placed on the MAJOR REGISTERS BUS, and loaded into PC 5–11 by TP3. The program control is then transferred to this new location, if the JMP instruction is directly addressed. If it is indirectly addressed, the data on MD lines 5–11 is gated to the MAJOR REGISTERS BUS during TS4 and loaded into CPMA 5–11 at TP4 time.

Similarly, when the instruction is an MRI, the data on MD lines 5–11 is placed on the MAJOR REGISTERS BUS during TS4. It is then loaded into CPMA 5–11 at TP4, providing the CPMA with the absolute address of the operand.

3.37 DESTINATION CONTROL SIGNALS

The destination control signals load data into the major registers. Data is placed on the MAJOR REGISTERS BUS during time states and loaded into a specific major register by a specific time pulse. Thus, TP1 loads only the PC Register, TP2 loads only the MB Register, TP3 loads the AC Register, primarily (it also loads the PC, if certain conditions are met), and TP4 loads only the CPMA Register. The fifth major register, MQ, is also loaded by TP3, but not from the MAJOR REGISTERS BUS. Because of the unique manner in which data is transferred to the MQ, Paragraph 3.40 is devoted to this major register; the MQ LOAD L signal is also discussed there.

3.37.1 MB LOAD L

The logic used to provide the MB LOAD L signal is shown in Figure 3-95. The MB is loaded only at TP2 time, but at *every* TP2. Thus, data placed on the MAJOR REGISTERS BUS during TS2 of any cycle is loaded into the MB Register by inverter E48.

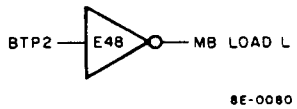


Figure 3-95 MB LOAD L Logic

3.37.2 AC LOAD L

The logic used to provide the AC LOAD L signal is shown in Figure 3-96. NAND gate E15D is used to provide the AC LOAD L signal during an IOT transfer. A detailed discussion of the use of AC LOAD L during an IOT transfer can be found in Section 6 of this chapter. The remainder of this discussion is concerned with how NAND gate E15C is used to assert AC LOAD L.

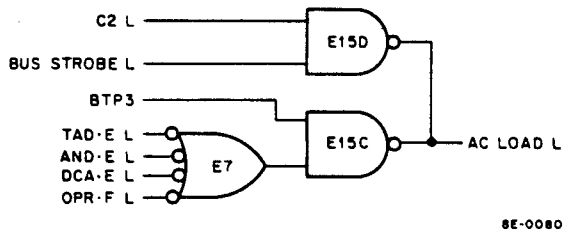


Figure 3-96 AC LOAD L Logic

Note that this load signal is generated at TP3 time. Thus, data that is to be loaded into the AC must be placed on the MAJOR REGISTERS BUS during TS3 time. If a TAD instruction is being executed, data in the AC is gated to the addend in line of the adder, while data on the MD lines is gated to the register in line of the adder (remember all gating is accomplished by the composite signal TAD·E·TS3). The result from the adder is a 2's complement sum of the two quantities. This sum is placed on the MAJOR REGISTERS BUS (all the shift signals are negated) and loaded into the AC at TP3 time.

If an AND instruction is being executed, the RIGHT L and LEFT L shift signals are asserted by AND·E·TS3, as described in Paragraph 3.36.2. The result obtained by NANDing the AC and the MB is gated to the MAJOR REGISTERS BUS and, again, the AC is loaded at TP3 time.

When a DCA instruction is being executed, the source control signals gate 0s to the adders during TS3. The 0 from the adder is placed on the MAJOR REGISTERS BUS and loaded by TP3.

Finally, all operate microinstructions cause the AC to be loaded at TP3 time, which occurs even when the microinstruction does not involve the AC (CML, for example). In this case, the AC is loaded at TP3 time with the same word that it contained prior to TP3 (the operation can be represented as AC → AC).

3.37.3 PC LOAD L

The logic used to provide the PC LOAD L signal is shown in Figure 3-97. NAND gate E15 is used during an IOT transfer and is discussed fully in Section 6 of this chapter. Note that PC LOAD L can be asserted at either TP1 or TP3 time. E53B is enabled by TP1 of a FETCH cycle or by TP1 of a key-initiated DMA cycle (KEY CONTROL is asserted by the DEP key and by the EXAM key; each key initiates a single DMA cycle). During TS1 of a FETCH cycle or of a key-initiated DMA cycle, the data on the MA lines is gated to the adder. The CAR IN L signal is asserted, and the output from the adder, MA+1, is gated through the Adder Output Multiplexer by the no-shift condition. TP1 then enables E53B to assert PC LOAD L, and MA+1 is loaded into the PC, updating the program count. NAND gates E53A and C assert PC LOAD L at TP3 time. E53A is used with the JMP housekeeping instruction. During TS3 of the FETCH cycle of a JMP instruction, an absolute address is gated onto the MAJOR REGISTERS BUS by the shift signals (Paragraph 3.36.2). This address gives the memory location from which the next program instruction is to be taken. At TP3 time, E53A asserts PC LOAD L, and the new program count is loaded into the PC Register.

NAND gate E53C asserts PC LOAD L at TP3 to load either MA+1 or MA into the PC. The address that is loaded into the PC is that of the first instruction of the subroutine to which program control is being transferred (Paragraph 3.36.1).

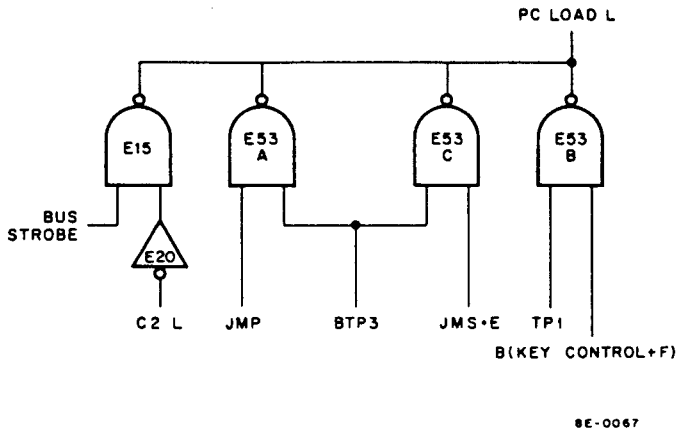


Figure 3-97 PC LOAD Logic

3.37.4 CPMA LOAD L

The logic used to provide the CPMA LOAD L signal is shown in Figure 3-98. Although only two gates are used, a variety of operations are involved. NAND gate E52 is used with the ADDR LOAD key. When the key is depressed, the address represented by the console switch register keys is gated to the MAJOR REGISTERS BUS. PULSE LA then enables E52, and the address is loaded into the CPMA.

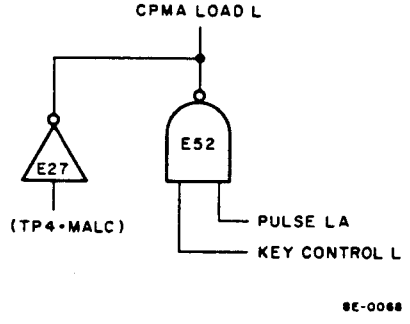


Figure 3-98 CPMA LOAD Logic

Inverter E27 asserts CPMA LOAD L when signal TP4•MA,MS LOAD CONTROL L is high. Thus, CPMA LOAD L is asserted by each TP4 pulse, with only one exception. This exception is the TP4 pulse that occurs during a peripheral-induced DMA state. The MA,MS LOAD CONTROL L signal is grounded by the peripheral to ensure that the processor returns to the correct major state at the end of the data break (see Paragraph 3.34.1 for more details).

The data loaded into the CPMA at TP4 time is always an address. If the instruction being performed is in its concluding cycle, or if the cycle is a DMA state, this address specifies the succeeding memory location. Because this address is being transferred from the PC Register, the operation is represented as PC → CPMA (it can also be represented as PC+1 → CPMA, an operation that involves the Carry In logic and the Skip logic; this operation is detailed in subsequent paragraphs).

If the request of a peripheral for a program interrupt has resulted in the INT IN PROG signal being asserted, the address loaded into the CPMA at TP4 time is 0000. The PC address is then stored in this memory location while the interrupting peripheral is serviced (Paragraph 3.35.1).

If the processor is in the FETCH state of either an MRI or an indirectly addressed JMP instruction, or is in the DEFER state of an indirectly addressed MRI, the address loaded into the CPMA at TP4 time specifies the memory location of the operand. If the processor is in the DEFER state, the operation is represented as MD → CPMA (Paragraph 3.35.1). If the processor is in the FETCH state, the operation is represented as either MA 0–4 → CPMA 0–4, MD 5–11 → CPMA 5–11, or 0 → CPMA 0–4, MD 5–11 → CPMA 5–11 (Paragraph 3.36.2).

3.38 SKIP LOGIC

The Skip logic, shown in Figure 3-99, is used to sample the contents of the AC Register and/or LINK, and the state of the OMNIBUS SKIP line. If the sampled data satisfies specified conditions, the program count is incremented before it is transferred to the CPMA at TP4 time. Thus, the next program instruction is skipped.

The Skip operation is used primarily during the implementation of operate microinstructions. As the PDP-8/E instruction list indicates, the majority of the Group 2 operate microinstructions, and nearly half of the combined operate microinstructions, involve the Skip instruction. Thus, the greater part of the logic shown in Figure 3-99 is devoted to implementation of these Skip microinstructions.

Consider the Group 2 microinstruction SMA (7500) (skip on minus AC). Remember that minus numbers in the PDP-8/E are those between 4000_8 and 7777_8, bit 0 is a 1. The AC0 bit is sampled by NAND gate E5. Therefore, when the AC contains a minus number, E5 is enabled by the SMA microinstruction (BMD5 H is asserted by SMA). The exclusive-OR gate E38 provides a high output because only one of its input signals, MD8 L (which is negated by SMA), is high. Because (OP2·TS3) is always high for the operate Skip microinstructions, NAND gate E3 is enabled, and the data (D) input of flip-flop E17 is high. At TP3 time, E17 is set, and the SKIP L (1) signal is applied to the Carry In logic. The Carry In logic causes the program count to be incremented before being transferred to the CPMA at TP4 time (Paragraph 3.36.1).

If the program specifies SPA (skip on plus AC), E5 is disabled because AC0 L is now low; E4 and E43 are also disabled. However, E38 still provides a high output, because MD8 L is now asserted. The D input of flip-flop E17 is again high, and BTP3 sets the flip-flop. Each operate Skip microinstruction that involves sampling of the AC and/or LINK, as well as the unconditional Skip microinstruction, SKP, is implemented in a manner similar to that outlined for SMA and SPA. In all cases, exclusive-OR E38 provides a high output signal if the SKIP condition is met, thereby enabling E3. NOR gate E12 provides a high input at the D line of E17, and BTP3 sets the flip-flop.

The Skip logic is also used to sample the state of the OMNIBUS SKIP line. During an I/O transfer (IOT L is asserted), a peripheral may assert the SKIP L signal. This action enables NAND gate E22, causing E17 to be set at TP3 time. This ultimately results in a skip of the next program instruction (the procedure is explained in detail in Paragraph 3.41.3). The SKIP L signal can also be asserted by the KE8-E option, as discussed in Volume 2.

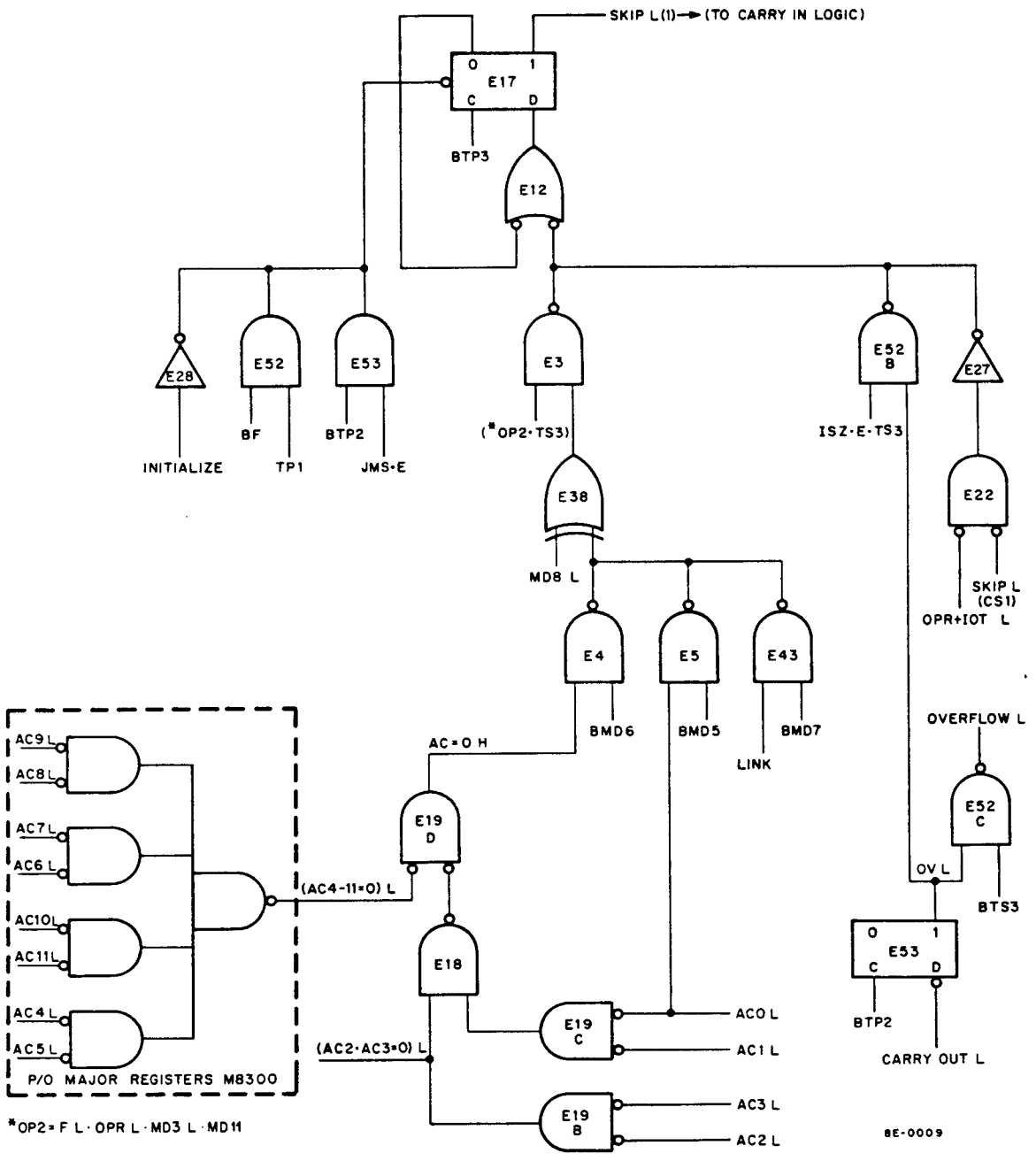


Figure 3-99 Skip Logic

Only one MRI (the ISZ instruction) causes the SKIP flip-flop, E17, to be set. The composite signal (ISZ·E·TS3) enables NAND gate E52B if E33, the OVERFLOW flip-flop, is active (Figure 3-99). The signal (ISZ·E·TS2) causes the CARRY IN L signal to be asserted, incrementing the operand of the MRI (Paragraph 3.36.1). If the result of this incrementation is 0000_8 (the operand must have been 7777_8), a carry out from adder 0 is generated. The CAR OUT L signal is applied to flip-flop E33, as shown in the logic diagram. BTP2 then sets E33. During TS3, E52B is enabled; thus, flip-flop E17 can be set by BTP3. SKIP(1) causes the CARRY IN L signal to again be asserted and to increment the program count. Thus, the ISZ instruction causes CARRY IN L to be asserted twice during the EXECUTE cycle.

Note that the 1 output of E33 is also applied to NAND gate E52C. During TS3, this gate is enabled, thereby grounding the OMNIBUS OVERFLOW line. This line is used by peripherals, and a further description can be found in Section 6 of this chapter.

Note that the SKIP flip-flop, E17, can be dc-reset in three ways: (a) it is reset whenever the INITIALIZE signal is asserted. This occurs when power is turned on or off, when the CLEAR key is depressed, or when the CAF instruction is programmed; (b) TP1 of each FETCH state resets it; (c) TP2 resets it when a JMS instruction is being executed (for details on the need for this reset see Paragraph 3.36.1, under the discussion of NAND gates E2A and E2D, see Figure 3-93).

3.39 LINK LOGIC

The Link is used with a TAD instruction or with Group 1 operate and combined operate microinstructions. The TAD instruction causes the content of the AC, a binary number, to be added to the content of a specified memory location, another binary number. Both binary numbers may be negative numbers (the MSB is a 1). Therefore, the addition may result in a carry from adder 0. Because this carry is significant to the result of the addition, it is gated to, and complements, the LINK flip-flop. The flip-flop can then be manipulated by the Group 1 operate microinstructions and used in other AC operations.

The Group 1 microinstructions can be used to clear and complement the LINK independently of the AC, or to rotate the LINK bit right or left along with the contents of the AC. In addition, the LINK can be complemented in response to the IAC microinstruction, providing the number in the AC is 7777 (-1) before incrementation.

The Link logic is shown in Figure 3-100. Note that the LINK flip-flop, E33, can be clocked if either NAND gate E43C or NAND gate E43D is enabled. The gate E43D is enabled if LINK LOAD L is asserted. LINK LOAD L and LINK DATA L are controlled by the RTF (restore interrupt flags) IOT instruction. RTF and its companion instruction, GTF (get interrupt flags), are designed primarily for use with the KE8-E and/or KM8-E options. These instructions are discussed only as they pertain to the loading of the LINK flip-flop.

The GTF instruction causes the state of the LINK bit to be placed on the DATA 0 L line. This data, and the information that is placed on DATA lines 1–11, is then loaded into the AC at TP3 time. The contents of the AC can be stored in some location until it is needed. At this later time, an RTF instruction can be issued. As the Link logic shows, the previous state of the LINK bit is placed on the Link Data line and gated through E32A to the D input of E33 (note that because RTF is an IOT instruction, NAND gate E30, at the Link Multiplexer output, is disabled). At TP3 time, LINK LOAD L is asserted and the previous LINK state is loaded back into E33.

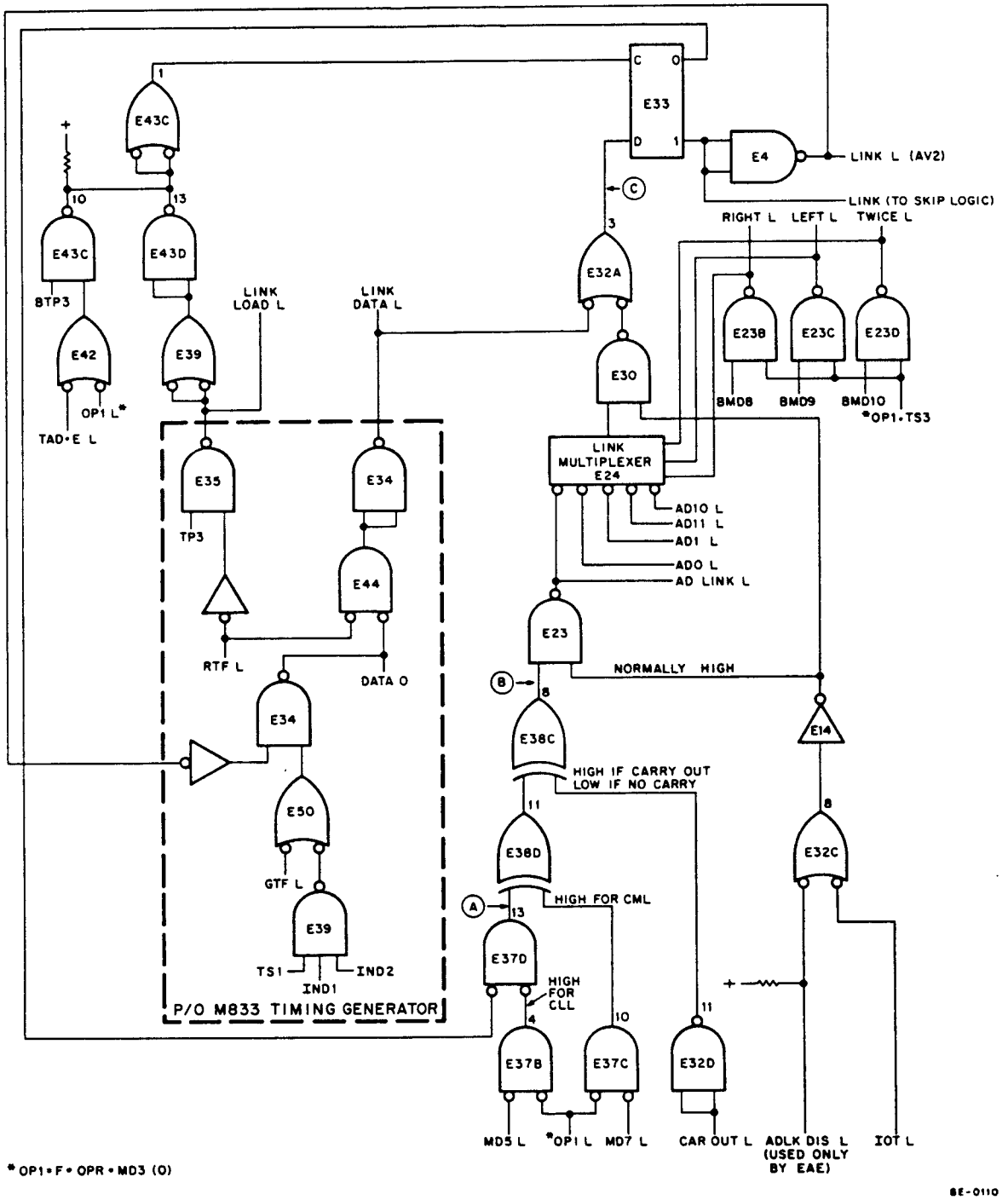


Figure 3-100 Link Logic

As previously discussed, the LINK is used with the TAD instruction and with Group 1 operate microinstructions (MD3 L is negated). Thus, when either (TAD·E) L or OP1 L is asserted, NAND gate E43C is enabled by BTP3, and E33 is clocked.

The Link can be rotated along with the AC. Therefore, the Link logic must contain a multiplexer similar to the Adder Output Multiplexers of the major register gating network. This required multiplexer is shown in Figure 3-100 as the Link Multiplexer, E24. Note that there are five distinct inputs to E24; four of these are from outputs of adders in the major register gating network. The fifth input is from the Link gating network (ICs E23, E32, E37, and E38), and is connected by the AD LINK L line to an input of Adder Output Multiplexers 0, 1, 10, and 11 in major register gating. When a rotate microinstruction is programmed, the shift signals, RIGHT L, LEFT L, and TWICE L, select only one of the four adder outputs, thereby enabling the logic level from this adder to be placed on the Link line. At the same time, the previous state of the Link line is selected by one of the Adder Output Multiplexers and placed on the MAJOR REGISTERS BUS for loading into the AC. For example, if RAR is programmed, the shift signals select the logic signal on the ADII L line (this signal reflects the contents of AC11). A signal representing this logic level is gated to the D input of the LINK flip-flop. When the LINK flip-flop is clocked, the logic level previously contained in AC11 is placed on the Link line. At the same time, the AD LINK input is selected by the shift signals at Adder Output Multiplexer. A signal representing the previous logic state of the Link line is placed on MAJOR REGISTERS BUS 0. At TP3 time, the AC is loaded and AC0 then contains the logic level previously carried on the Link line. Note that, because there is no connection between AD LINK L and Adder Output Multiplexer 5, and none between AD6 L and the Link Multiplexer, the Link line is unaffected by the BSW instruction.

To carry out the rotate instruction, the Link gating must place the state of the Link line on the AD LINK line. However, if a programmed microinstruction directs that the Link be cleared (the Link line negated), a high signal must be gated to the AD LINK line (E24's no-shift input). Finally, if a Group 1 microinstruction or a TAD instruction directs that the Link line be complemented, the complement must be placed at the AD LINK L input of E24. Because the Link gating must accomplish these tasks, the gating processes are relatively intricate. Three NAND gates, E37B, E37C, and E32D, are keys in the gating network. The various combinations of the enabled and disabled states of these gates determine whether the Link line is negated, complemented, or left unchanged. The output of E37B is low unless an operate instruction involving the CLL instruction is present; the output gate of E37C is high for the CML instruction; the output of E32D is low unless there is a carry from the adders; and the output of E14 is high for all except IOT and EAE instructions.

The outputs of the two exclusive OR gates E38C and E38D are low if their inputs are of the same logic level. If the inputs are different, the output is high. If there is no CML instruction, and no carry out from the adders, the logic level at point B is the same as the logic level at point A. If either a CML instruction or a carry out is present, B $\neq$ A. If both CML and carry out are present (an extremely rare occurrence), A = B.

Table 3-8 is a learning aid and possible maintenance tool. This table lists the various gates of the Link gating network, along with the output to be expected from each under the conditions specified at the top of the table. For example, the CMA instruction should have no effect on the state of the Link line. However, the LINK flip-flop is clocked whenever the CMA instruction is programmed; therefore, the state of the flip-flop's D input must be such that the flip-flop is unchanged by the clock pulse. The level to be expected at the D input is found in the column headed BSW, etc. This level is the complement of the level on the Link line. If LINK L is asserted (a low level), the D input is high. At TP3 time, the flip-flop is set, as it was before TP3.

Table 3-8
Link Logic Gating Levels

Intended Operation / Gate State or Output	Complement Link (Group 1 Microinstruction)	Complement Link (CAR OUT L Asserted)	Rotate Link (Shift Signals Asserted)	Clear Link	BSW, CLA, NOP, CMA
E37B NAND	Disabled	Disabled	Disabled	Enabled	Disabled
E37C NAND	Enabled	Disabled	Disabled	Disabled	Disabled
E37D NAND	Output is complement of level on Link line	Output is complement of level on Link line	Output is complement of level on Link line	Output is LO regardless of level on Link line	Output is complement of level on Link line
E38D EXCLUSIVE-OR	Output is same as level on Link line	Output is complement of level on Link line	Output is complement of level on Link line	Output is LO regardless of level on Link line	Output is complement of level on Link line
E32D NAND	Enabled	Disabled	Enabled	Enabled	Enabled
E38C EXCLUSIVE-OR	Output is present level on Link line	Output is present level on Link line	Output is complement of level on Link line	Output is LO regardless of level on Link line	Output is complement of level on Link line
E32C NOR	Disabled	Disabled	Disabled	Disabled	Disabled
AD LINK LINE	Complement of level on Link line	Complement of level on Link line	Same as level on Link line	HI level regardless of level on Link line	Same as level on Link line
LINK F/F D INPUT	Same as level on Link line	Same as level on Link line	Complement of level on selected ADX line	LO level regardless of level on Link line	Complement of level on Link line
RESULT	Link line assumes, at TP3 the complement of its present level	Link line assumes, at TP3, the complement of its present level	Link line assumes level of selected ADX line at TP3	Link line assumes a HI level at TP3	Link line remains at same level

3.40 MQ REGISTER LOGIC

The MQ Register was briefly discussed in Paragraph 3.35. The MQ acts as an extension of the AC during Extended Arithmetic Element (EAE) operations. Only in this capacity does the MQ fulfill its potential. However, because the register is included in the basic machine, it provides the programmer with a temporary storage register and increases the processing power of the AC. Consequently, the eight MQ instructions, discussed in Paragraph 3.35.2, are included in the instruction repertoire of the basic PDP-8/E.

In carrying out these eight MQ instructions, the AC and MQ must transfer information from one to the other. The MQ is represented in the block diagram in Figure 3-87. Note that the MQ does not monitor the MAJOR REGISTERS BUS; therefore, to transfer the contents of the AC to the MQ, another data path must be provided as shown in the MQ logic gating for bit 0 (Figure 3-101). The AC data is transferred to the MQ via the MQ multiplexer, E12, and the right-shift gate, E17. Data to be transferred from the MQ to the AC is gated through the major register gating network to the MAJOR REGISTERS BUS. The logic used to generate the MQ control signals is shown in Figure 3-102. Some of these signals have been discussed in relation to other processor operations. The pecularities of this logic, when related to implementation of the MQ instructions, are discussed in this section.

Consider the MQ instruction CLA SWP, 7721 – transfer MQ to the AC, clear the MQ. The MQ data must be gated through the major register gating network to the MAJOR REGISTERS BUS during TS3 and loaded into the AC at TP3. There are two paths to the major register gating for the MQ (Figure 3-101). The path that goes to Register Input Multiplexer 0 is used only with the EAE option, during implementation of the SAM instruction. Thus, CLA SWP must cause the MQ data to be placed on the DATA 0 line for transfer to Data Control Gate 0. The MQ → BUS L control signal must be asserted, which is done by NAND gate E9B (I/O PAUSE L is asserted only during an IOT instruction). Note that at the same time AC → BUS L is negated by E9C (E21 is enabled, E11 is disabled). The MQ0 bit is gated to the MAJOR REGISTERS BUS and loaded into the AC at TP3. To complete the instruction, the MQ must now be cleared (0 → MQ), which is accomplished if SHL ENA L, RIGHT L, and AC → MQ ENA L are negated. The first two signals are asserted only by the EAE option, when considering the MQ. The last, AC → MQ ENA L, is controlled by NAND gate E11; as noted previously, this gate is disabled during CLA SWP (MD4 L is asserted). Therefore, AC → MQ ENA L is negated. These three control signals, when negated, cause a 0 (positive voltage) to be applied to the data (D) input of the MQ0 flip-flop. MQ LOAD L, at TP3, then clears the flip-flop.

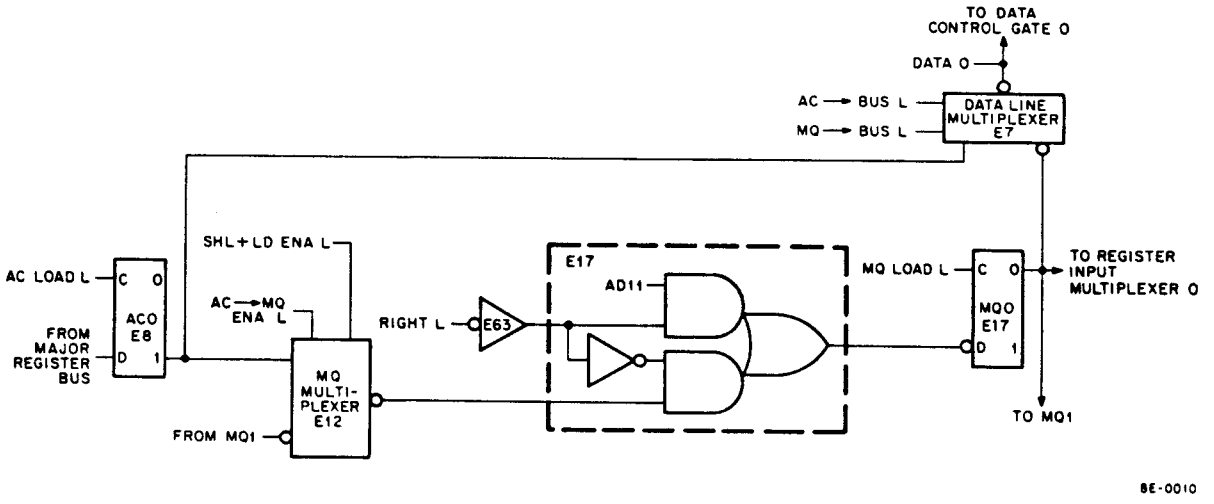


Figure 3-101 MQ Logic

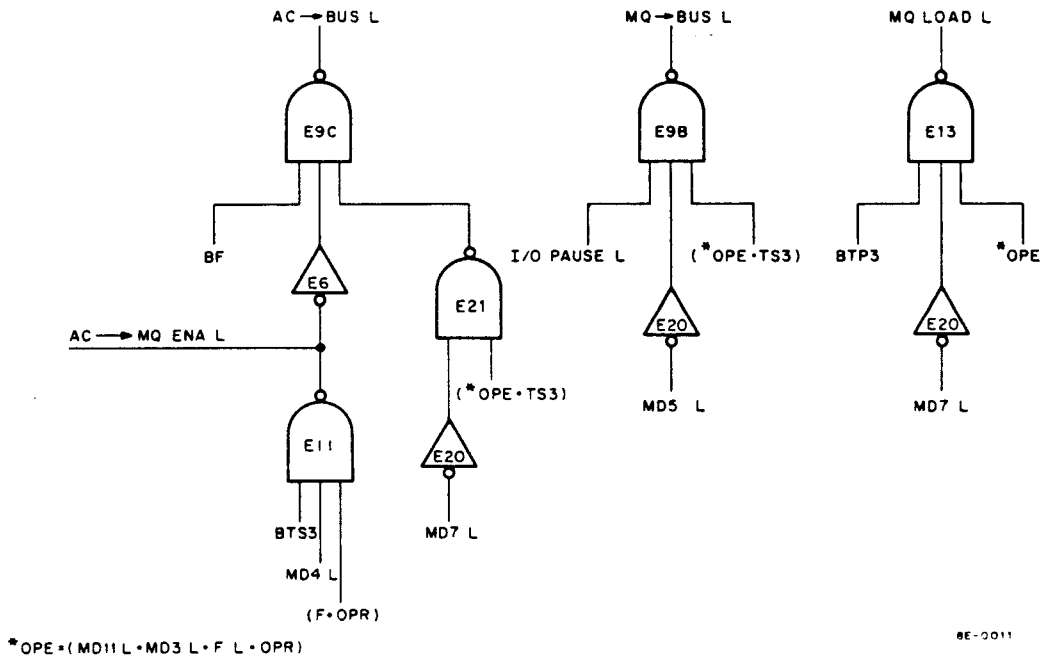


Figure 3-102 MQ Gating Control Logic

If ACL, 7701 (transfer MQ to the AC), is programmed, the operation is similar to CLA SWP. MQ → BUS L is asserted, and AC → BUS L is negated (E11 is disabled). MQ0 is loaded into AC0 at TP3. SHL ENA L, AC → MQ ENA L and RIGHT L are again negated. To prevent the MQ from being cleared, as it is during CLA SWP, the MQ LOAD L is negated (MD7 L is negated). Thus, the MQ retains the data, in addition to transferring it to the AC.

If SWP, 7521, is programmed, simultaneous transfers are required. Thus, the MQ is transferred to the AC as described, while the AC is transferred to the MQ via the MQ multiplexer, E12. This multiplexer gates the AC0 bit to E17, if AC → MQ ENA L is asserted and SHL ENA L is negated. The MQ transfer is negated unless asserted by the EAE. The AC transfer is asserted by NAND E11 in the AC → BUS L logic (AC → BUS L is negated by E21). If RIGHT L is negated (as it is, unless asserted by the EAE), the output of E17 reflects the state of the AC0 flip-flop. MQ LOAD then loads the MQ0 flip-flop with the AC0 data. At the same time, the AC0 flip-flop is being loaded with the MQ0 data.

The rest of the MQ instructions are implemented by carrying out one or another of the operations described in CLA, SWP, ACL, and SWP. These instructions are outlined in the following paragraph. The method through which the operations are implemented is not discussed.

The MQL instruction negates both AC → BUS L and MQ → BUS L, while asserting AC → MQ ENA L. The MQA instruction asserts both AC → BUS L and MQ → BUS L, while negating MQ LOAD (AC → MQ ENA L is asserted, but has no significance). The CAM instruction negates AC → BUS L, MQ → BUS L, and AC → MQ ENA L. The CLA instruction negates AC → BUS L, MQ → BUS L, AC → MQ ENA L, and MQ LOAD L.

The MQ logic is discussed in Volume 2 when the EAE option is described in detail. Signal destination and signal source notation, which were not discussed in this section, are fully explained in Volume 2.

SECTION 6 – I/O TRANSFER LOGIC

3.41 PROGRAMMED I/O TRANSFER LOGIC

3.41.1 Programmed I/O Transfer Gating and Control Logic

Programmed I/O transfers use IOT instructions to initiate data transfers between an option and the AC Register. Information is transferred from and to an option via the OMNIBUS DATA 0–11 lines. Figure 3-103 shows the major register gating for bit 0 and the programmed I/O control signal logic. The major register gating shows only the essential features of I/O transfer; thus, only the AC and the PC Registers are included in the diagram. Note that the route control signals of major register gating are always negated for I/O transfers. However, the source control signals and the two destination control signals are used in a manner similar to that encountered in transfers between major registers. These last two types of control signals are asserted during I/O transfers, largely by the signals present on three OMNIBUS control lines: C0 L, C1 L, and C2 L. The option asserts the C0 L, C1 L, and C2 L signals, thereby specifying the direction (from or to the CP) of data transfer. Two important signals that are used along with these C-line signals are asserted within the processor whenever an IOT instruction is programmed. These two signals are I/O PAUSE L, used in the source control signal logic, and BUS STROBE L, used in the destination control signal logic. These signals are discussed in detail in later paragraphs. For this discussion it is sufficient to know that I/O PAUSE L is asserted by the processor when an IOT instruction is programmed; BUS STROBE L is asserted at TP3 time of an I/O transfer.

Table 3-9 lists the six possible types of programmed I/O transfers, the C-line signal levels for each type, and the resulting source and destination control signal levels (signal levels are given in terms of HI and LO voltages). This table can be used with Figure 3-103 as both a learning aid and a maintenance tool. The following discussion is an introduction to the table, when applied to the gating diagram.

For example, consider the first entry under Type of Transfer: OUTPUT, AC UNCHANGED. The AC0 bit must be placed on the DATA 0 line (it is later strobed into a buffer register within the option). The option accomplishes this by negating the C lines. AC → BUS L is asserted and gates AC0 to the DATA 0 line. Now, the 0 bit must be gated to the MAJOR REGISTERS BUS. The DATA T/DATA F control signals gate the bit to adder 0 during TS3. A 0 is gated to the adder by Register Input Multiplexer 0; because the route control signals are negated, the unaltered AC0 bit is placed on the MAJOR REGISTERS BUS. At TP3, BUS STROBE L is asserted and the resulting AC LOAD L signal clocks AC0. Because the D input does not change states, the AC0 flip-flop does not change states. Thus, the output transfer leaves the AC unchanged. The major register gating should be examined with Table 3-9. Understanding each type of data transfer results in an excellent understanding of programmed I/O transfers.

3.41.2 I/O PAUSE L, BUS STROBE L, INT STROBE Logic

The logic used to assert the I/O PAUSE L, BUS STROBE L, and INT STROBE signals is shown in Figure 3-104. The I/O PAUSE L signal is asserted by NAND gate E7 whenever an IOT instruction is brought from memory, provided the OMNIBUS USER MODE L signal has not been asserted (this signal is asserted by the KM8-E option and is discussed in detail in Volume 2; for the present discussion, this signal is negated high). Gate E7 is enabled when bits MD0–2 indicate an IOT instruction ($6XXX_8$), the USER MODE L signal is high, and the I/O TIME flip-flop is set. This flip-flop is set during each FETCH cycle at the same time that the STROBE memory timing signal is negated, i.e., at TP1 time plus 150 ns (setting the flip-flop at this time allows the option maximum time to decode the IOT instruction and activate the necessary control lines).

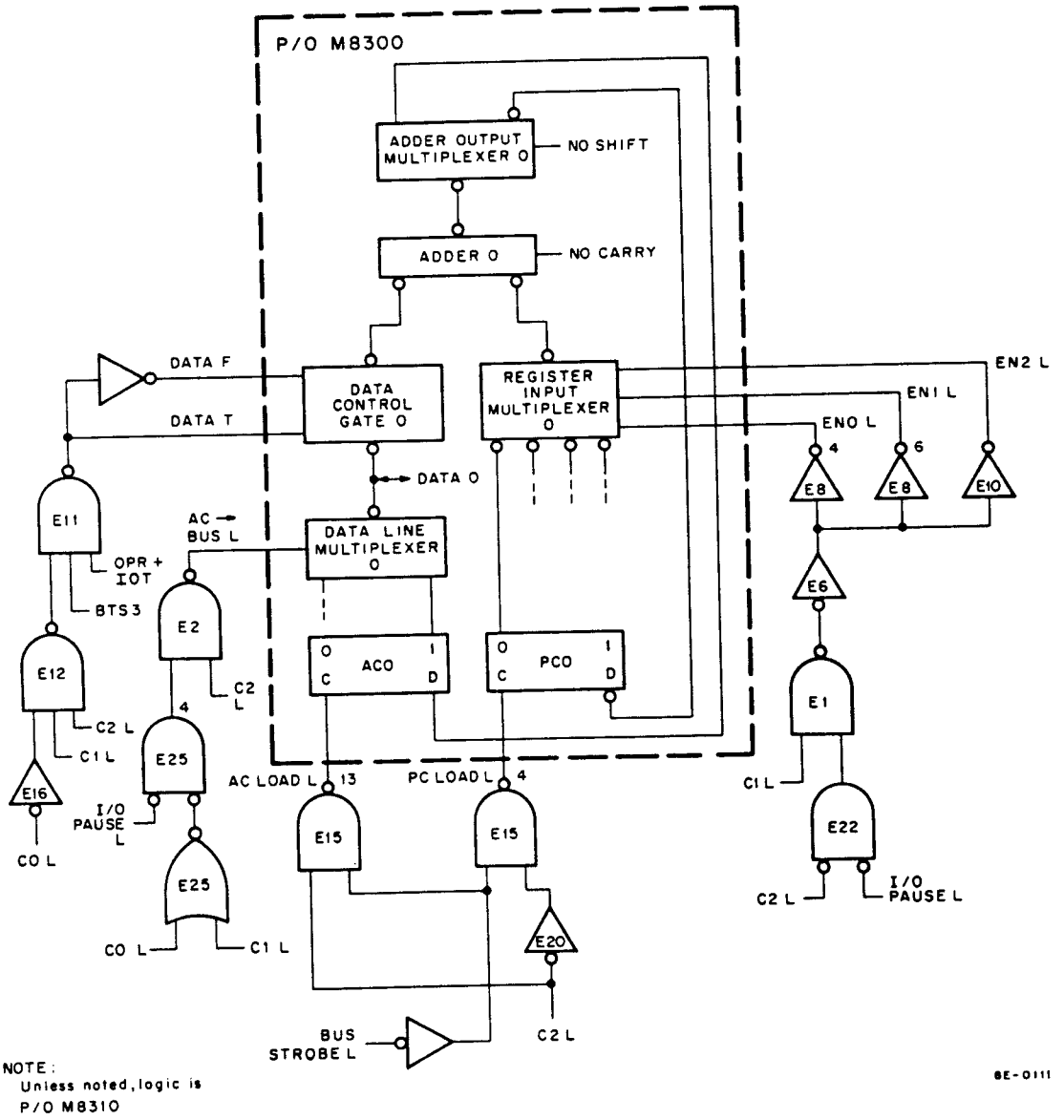


Figure 3-103 Major Register Gating (Bit 0) and Control Signal Logic, Programmed I/O Transfer

Table 3-9
Programmed I/O Transfer Control Signals,
Major Register Gating

Type of Transfer	C0 L	C1 L	C2 L	AC → BUS L	DATA T/F	EN0,1,2 L	AC LOAD L	PC LOAD L
OUTPUT, AC UNCHANGED	HI	HI	HI	LO	LO/HI	HI	LO	HI
OUTPUT, AC CLEARED	LO	HI	HI	LO	HI/LO	HI	LO	HI
INPUT, AC ORed with INPUT DATA	HI	LO	HI	LO	LO/HI	HI	LO	HI
JAM INPUT	LO	LO	HI	HI	LO/HI	HI	LO	HI
INPUT, DATA ADDED TO PC	LO	HI	LO	HI	LO/HI	LO	HI	LO
INPUT DATA TO PC	LO	LO	LO	HI	LO/HI	HI	HI	LO

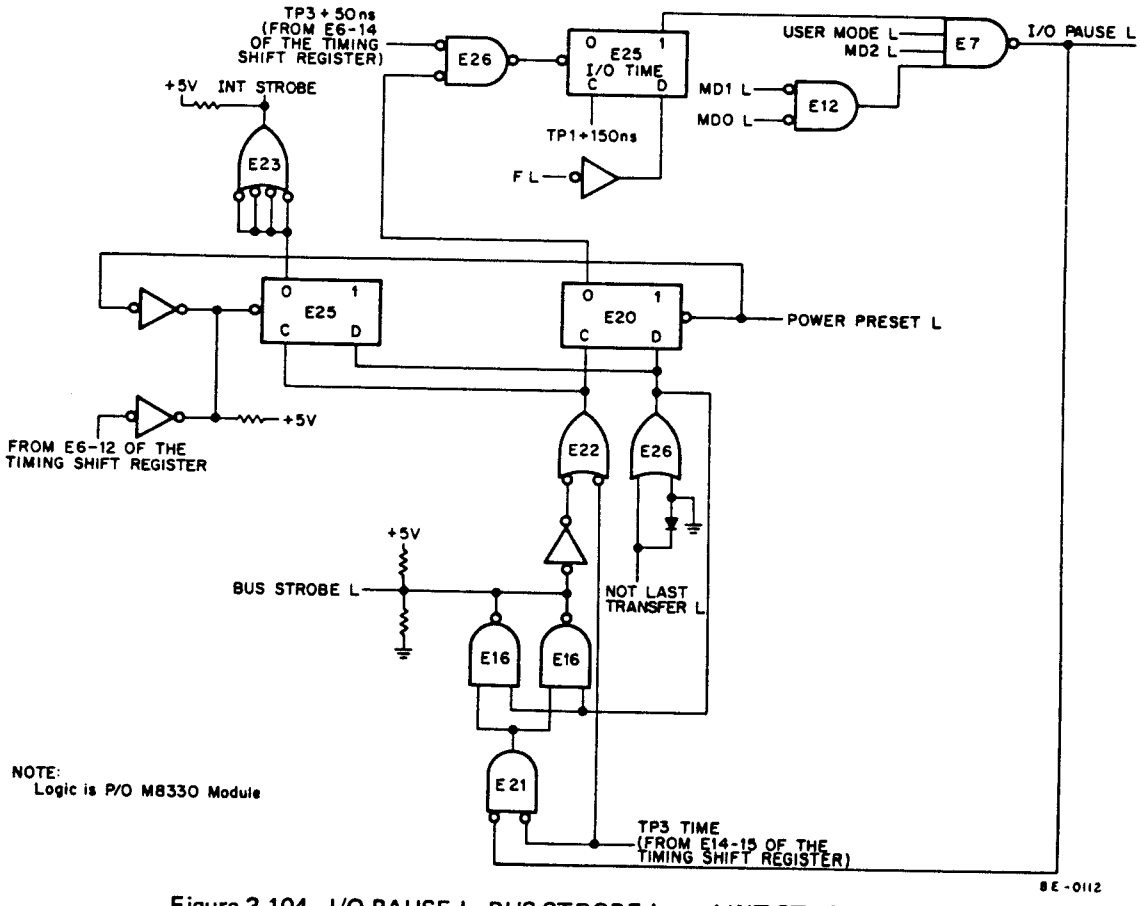


Figure 3-104 I/O PAUSE L, BUS STROBE L, and INT STROBE Logic

If the I/O transfer can be accomplished in a normal fast cycle (1.2 μs), the NOT LAST TRANSFER L signal remains high throughout the transfer (Paragraph 3.21 discusses the NOT LAST TRANSFER L signal and how the signal is used to interrupt normal processor timing). The BUS STROBE L signal is asserted at TP3 time by NAND gates E16 (the signal identified as TP3 TIME is a 100 ns negative pulse coinciding with the TP3 pulse). At the same time, flip-flop E20 is set. NAND gate E26 is enabled 50 ns later, clearing the I/O TIME flip-flop and causing the I/O PAUSE L signal to be negated. Note that the I/O TIME flip-flop is set during each FETCH cycle, even if the I/O PAUSE L signal is not asserted. Thus, the flip-flop must be cleared during each cycle; this is why the TP3 TIME signal is allowed to clock E20 via NOR gate E22.

If the option asserts the NOT LAST TRANSFER L signal, normal processor timing is interrupted. The timing of the logic in Figure 3-104 is also interrupted. The NOT LAST TRANSFER L signal prevents flip-flop E20 from being set by the TP3 TIME signal and ensures that NAND gates E16 do not assert BUS STROBE L. The option assumes responsibility for generating BUS STROBE L and clearing the I/O TIME flip-flop.

After the option negates the NOT LAST TRANSFER L signal, it issues a BUS STROBE L signal that sets flip-flop E20. The processor timing resumes, NAND gate E26 is enabled, and the I/O TIME flip-flop is cleared.

The INT STROBE signal synchronizes the program interrupt logic (Paragraph 3.42) and is used by data break devices to synchronize device and processor timing. It is normally asserted at TP3 time, when flip-flop E25 is set. When normal timing is interrupted for an I/O transfer, the INT STROBE signal is not generated until the peripheral negates the NOT LAST TRANSFER L signal and then issues BUS STROBE L. The INT STROBE signal remains high until E25 is cleared by a signal from E6, pin 12, of the Timing Shift Register (this results in an INT STROBE pulse of 100 to 150 ns duration).

3.41.3 IOT Skip Logic

The IOT Skip logic is shown in Figure 3-105. An option can assert the OMNIBUS Skip line in response to directions contained in the IOT instruction. When this occurs, flip-flop E17 is set at TP3 time. At TP4 of the FETCH cycle (B F SET L and (MS, IR DISABLE L) H ensure that this occurs only during FETCH), CARRY IN L is asserted and PC + 1 → CPMA is carried out by the processor (refer to Paragraph 3.34 for the discussion concerning the F SET L and IOT L signals).

3.41.4 Processor IOT Logic

A number of internal IOT instructions are designated processor IOT instructions. They are represented as $600X_8$ and are concerned, in general, with program interrupts. This paragraph discusses only the assertion of signals representing the processor IOTs, except in the case of the CAF instruction.

The appropriate logic is shown in Figure 3-106. NAND gate E15 is enabled whenever an internal IOT instruction is brought from memory (the signal at pin 2 of E49 represents an instruction of the form $600X_8$). The INTERNAL I/O L signal is asserted so that peripherals interfaced to the OMNIBUS via the KA8-E option (Positive I/O Bus Interface) will ignore the IOT instruction. The eight possible combinations of bits MD9—11 are decoded by E49 to provide the eight processor IOTs shown. These IOTs, with the exception of CAF, are used with program interrupt and with the KE8-E (EAE) and/or the KM8-E (Memory Extension) options. The IOTs that deal with program interrupts are discussed in Paragraph 3.42; refer to Volume 2 for information concerning those instructions relating to the KE8-E and KM8-E. The CAF IOT instruction causes the one-shot, E27, to be triggered at TP3 time. Transistor Q1 is turned on, asserting the OMNIBUS INITIALIZE signal. The one-shot is cleared at TP2 time of the next timing cycle; if the processor timing ends after the CAF instruction (the operator is single-stepping, for example), the one-shot times out after 3 μs, ensuring that the INITIALIZE signal is not asserted for an inordinate amount of time.

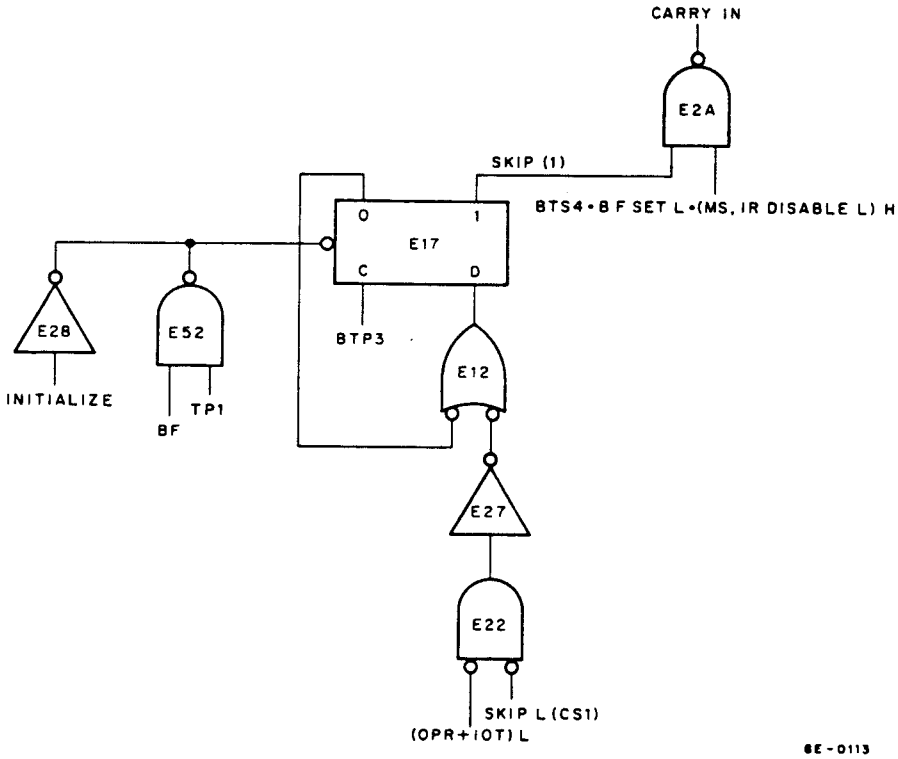
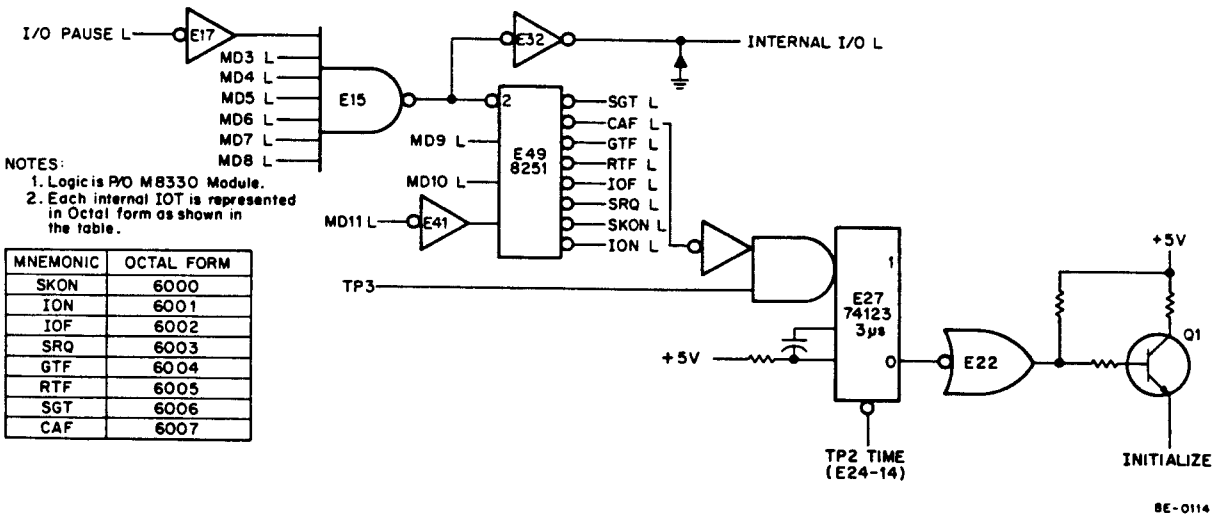


Figure 3-105 IOT Skip Logic (M8310)



NOTES:
1. Logic is P/O M8330 Module.
2. Each internal IOT is represented in Octal form as shown in the table.

MNEMONIC	OCTAL FORM
SKON	6000
ION	6001
IOF	6002
SRQ	6003
GTF	6004
RTF	6005
SGT	6006
CAF	6007

Figure 3-106 Processor IOT Logic

3.42 PROGRAM INTERRUPT LOGIC

3.42.1 Interrupt On/Off Logic

Program interrupt data transfers are more efficient than programmed I/O transfers. In the program interrupt transfer mode, the program is interrupted only when an option demands attention by asserting the OMNIBUS INT RQST L signal. The interrupt system monitors this INT RQST L signal. If the system is turned on when this signal is asserted, the processor executes a hardware-generated JMS to location 0. Simultaneously, it turns off the interrupt system; thus, further interrupts can occur only when the present one has been serviced. A program subroutine is entered to determine the identity of the requesting option. When this identity has been established, a servicing subroutine allows the option to take part in a programmed I/O dialogue with the processor.

The Interrupt On/Off logic is shown in Figure 3-107. The system can be turned on by the ION instruction. Note that the D input of the INT ENA flip-flop is high when the ION instruction (6001) is on the MD lines. Thus, the flip-flop is set at TP3 time of the instruction. Conversely, it is cleared at TP3 time of the IOF instruction (6002). When the 1 output goes high, the low is removed from the clear line of the INT DELAY flip-flop. At TP2 time of the next Fetch (F L) cycle, this flip-flop is set and its 1 output, now high, allows the next INT STROBE signal to clock the NO INT flip-flop. If the instruction being performed at the time that the flip-flop is clocked is in its concluding cycle, i.e., the F SET L signal is low; if the processor is not in the DMA state, i.e., the MS, IR DISABLE L signal is high; and, if an option has asserted the INT RQST L signal, NAND gate E7 is enabled. The two provisions ensure that an instruction that has been broken into by a data break device is completed when the device relinquishes control of the processor.

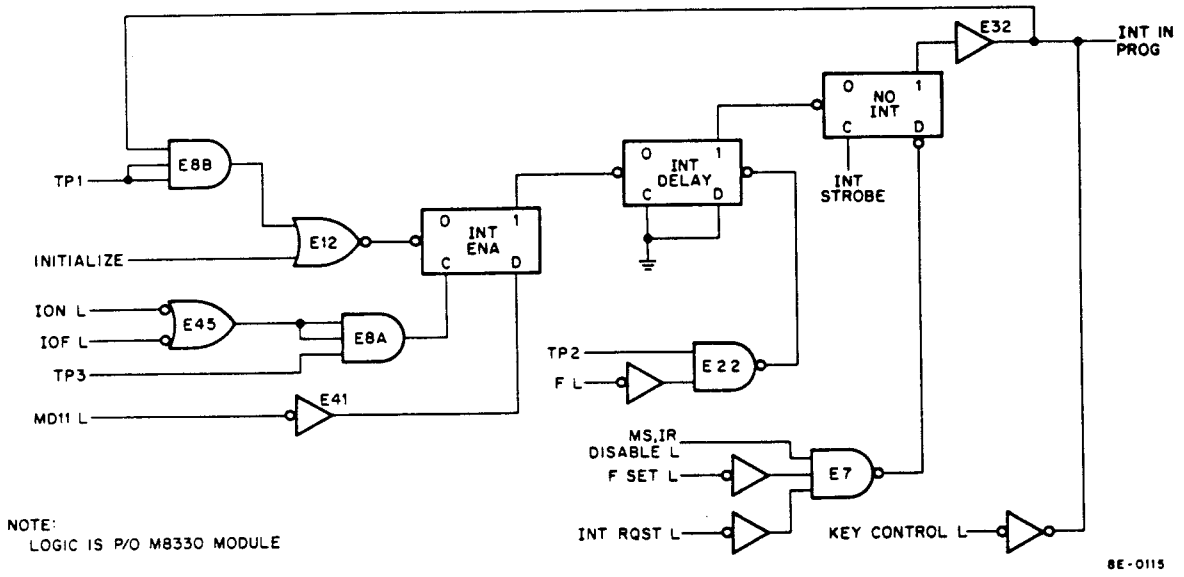


Figure 3-107 Interrupt On/Off Logic

If NAND gate E7 is enabled, the INT STROBE signal sets the NO INT flip-flop, causing the INT IN PROG signal to be asserted. At TP4 time, this signal forces the IR to JMS and forces the EXECUTE state flip-flop in the Major State Register to be set. At TP1 time of this EXECUTE cycle the INT ENA flip-flop is cleared via NAND gate E8B (note that this flip-flop is also cleared by INITIALIZE). The INT DELAY and the NO INT flip-flops are cleared in turn. The processor executes the appropriate JMS operations and then proceeds to the interrupt servicing routine. When the option has been serviced, an IOT instruction again turns the interrupt system on. Note that there is a delay of at least one complete cycle between the setting of the INT ENA flip-flop and the clocking of the NO INT flip-flop (there can be many more than one cycle if data break devices break into the normal timing). This delay allows the processor to obtain the return address from location 0, where it was stored, before a new interrupt can occur.

3.42.2 Interrupt Skip Logic

The SKON instruction (6000) grounds the OMNIBUS Skip line if the interrupt system is turned on (INT ENA flip-flop is set). At TP3 time of this instruction, the system is turned off, as shown in Figure 3-108 (the MD11 L signal is high). If the SKIP L signal is asserted by SKON, TP3 also sets the SKIP flip-flop (Figure 3-105). Then, during TS4, the Carry In line is asserted, the PC Register is incremented, and the instruction following SKON is skipped.

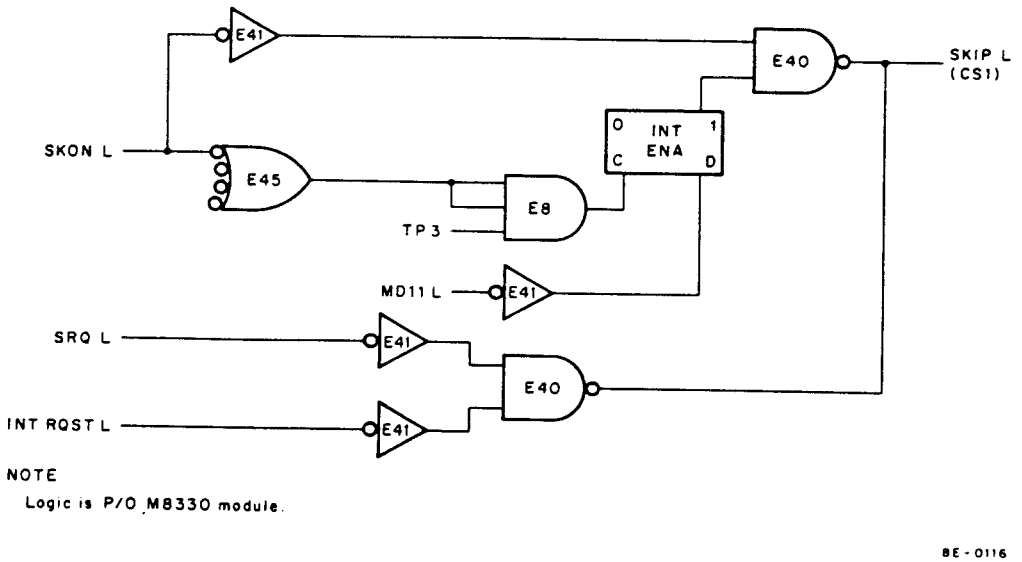


Figure 3-108 Interrupt Skip Logic

The SRQ instruction can also ground the Skip line, but only if the INT RQST L signal has been asserted by an option. Then the Skip logic and the Carry In logic cause the next instruction to be skipped.

3.42.3 Get/Return Interrupt Flags Logic

The GTF and RTF instructions have been discussed in relation to their use with the Link (Paragraph 3.39). The logic covered in Paragraph 3.39 is also presented in Figure 3-109. However, it is repeated only for continuity and is not discussed in detail.

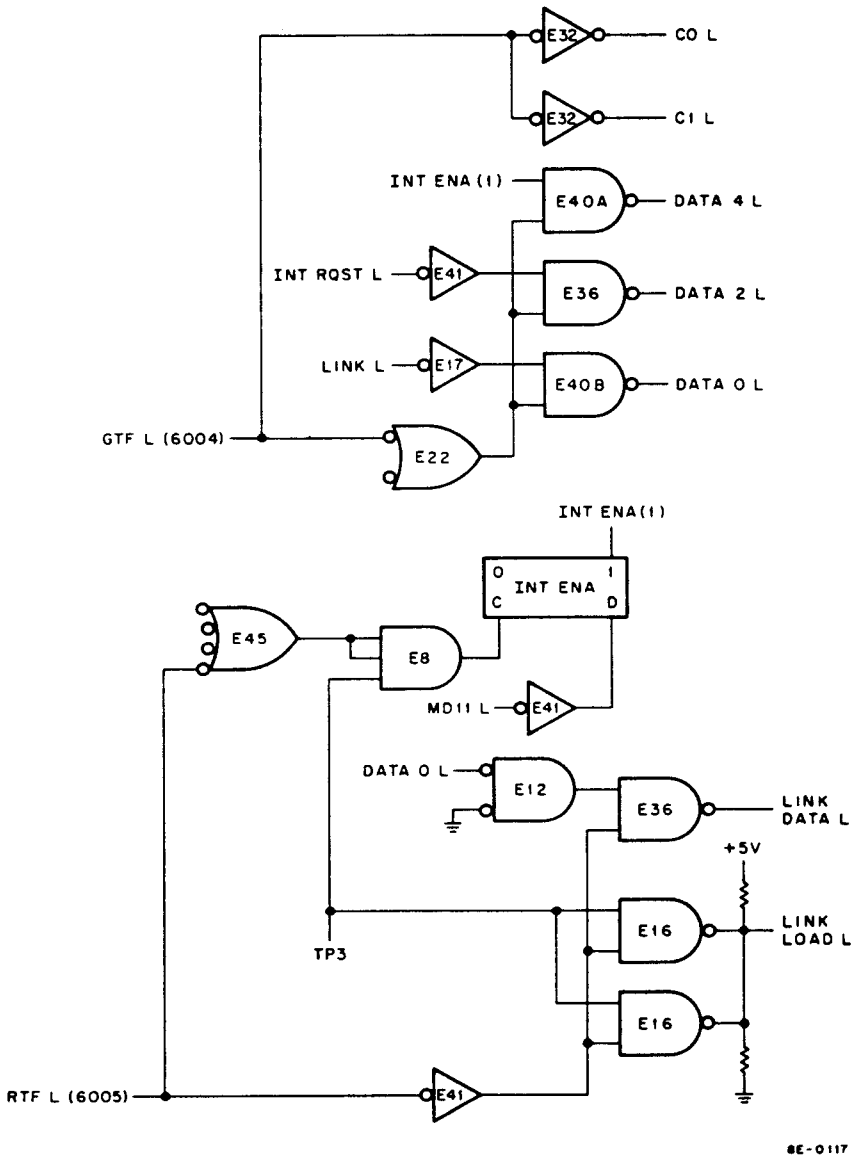


Figure 3-109 Get/Return Interrupt Flags Logic

When GTF L is generated, the states of the OMNIBUS LINK and INT RQST lines are gated to the DATA 0 and DATA 2 lines, respectively. The state of the INT ENA flip-flop is gated onto the DATA 4 line. Simultaneously, the C0 L and C1 L signals are asserted by gates E32, while the C2 L signal is left negated. The result of this gating is a jam input of data to the AC (Table 3-9 and Figure 3-103). The nine remaining AC bits – 1, 3, and 5 through 11 – are provided by the KM8-E and/or KE8-E options.

The RTF instruction, in addition to restoring a previous state of the Link line, also sets the INT ENA flip-flop at TP3 time. Both this instruction and GTF have minor significance within the processor itself, performing only housekeeping functions with the interrupt system. Their full potential is realized only when the KM8-E (Memory Extension) and/or KE8-E (EAE) options are used.

3.43 DATA BREAK TRANSFER CONTROL LOGIC

Data transfers that occur between memory and a data break peripheral are carried out in the Direct Memory Access (DMA) state of the processor (Paragraph 3.34.1). This state provides direct communication between the peripheral and memory by allowing the peripheral to assume control of major register gating. The peripheral accomplishes this by asserting a number of OMNIBUS signals when it is ready to make a data transfer.

The major OMNIBUS signal is the MS, IR DISABLE L signal. Paragraphs 3.34.1 and 3.34.2 describe how this signal is asserted by the peripheral at TP4 of a processor FETCH, DEFER, or EXECUTE state; the asserted signal forces the processor to enter the DMA state and disables the IR Register. At the same time (TP4), the output of the CPMA Register is removed from the OMNIBUS MA lines, because the peripheral asserted CPMA DIS L at INT STROBE H time. The peripheral, which monitors the MA lines through its control module, then specifies the memory location that is to receive or send a data word. Now the peripheral indicates the direction of transfer and provides the necessary gating signals.

The major register gating for bit 0, reduced to essential details, and the logic that controls the gating during the data break operation are shown in Figure 3-110. Note that MS, IR DISABLE L is also used in the control logic to assert the source control signals. Table 3-10 shows the state of the OMNIBUS signals and the source control signals for each of the three basic data break transfers. The following paragraph discusses the first entry in this table and presents some points that may not be apparent from Figure 3-110 or Table 3-10.

Table 3-10
OMNIBUS and Source Control Signal States,
Data Break Transfer Gating

Type of Transfer	MS IR DISABLE L	CPMA DISABLE L	MD DIR L	BREAK DATA CONT L	MA, MS LOAD CONT L	DATA T/F	EN0/1/2
ADM INPUT	LO	LO	HI	LO	LO	LO/HI	LO/LO/HI
INPUT	LO	LO	HI	HI	LO	LO/HI	HI/HI/HI
OUTPUT	LO	LO	LO	HI	LO	LO/HI	HI/HI/HI

The first entry is an input data transfer called Add To Memory (ADM). The peripheral asserts CPMA DIS L at TP3 and MS, IR DISABLE L at TP4 of the state that precedes entry to the DMA state. The MS, IR DISABLE L signal ensures that the next cycle is carried out in the DMA state, while CPMA DISABLE L allows TP4 to reset flip-flop E17. This flip-flop causes NAND gate E1 to be disabled, thereby removing CPMA0 from the MA0 line. At this time, the peripheral places the memory address of the transfer on the MA0 line; the DMA state is then entered at TS1. At TP1, MA, MS LOAD CONTROL L is asserted, preventing CPMA0 from being clocked during the data break operation (refer to Paragraph 3.34.1 for details concerning MA, MS LOAD CONTROL L). During TS2 time, the MS, IR DISABLE L signal asserts the DATAT/F signals and, with BREAK DATA CONT L, the EN signals. BREAK DATA CONT L, which must be asserted by the peripheral before TS2, causes the MD0 line to be gated to adder 0. DATA T/F causes the DATA 0 line to be gated to the adder. At the beginning of TS2,

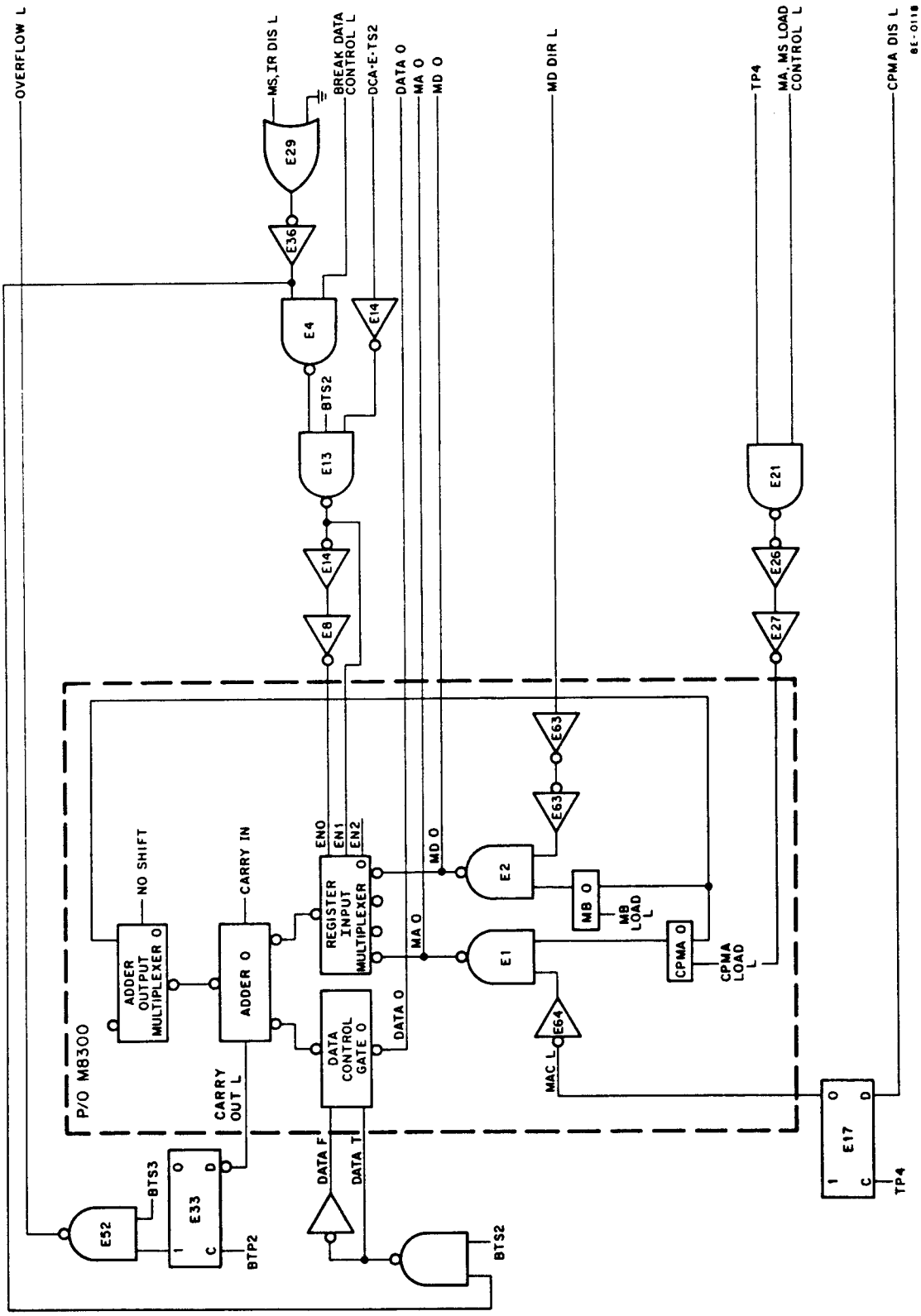


Figure 3-110 Processor, Data Break Transfer Control

the peripheral places the data to be transferred on the DATA lines. Thus, DATA0 + MD0 is placed on the MAJOR REGISTERS BUS during TS2 (disregard the Carry In line at this point). At TP2 time, MB0 is loaded with DATA0 + MD0, while, at the same time, the Timing Generator negates MD DIR L. Thus, the data is transferred to the addressed memory location.

Now consider the Carry In/Out lines of adder 0. Carries may be produced as a result of the ADM operation. The previously discussed method of sensing a Carry Out from adder 0 (LINK, Paragraph 3.39) is not available during the data break operation. Therefore, the Overflow flip-flop, E33, is provided to notify the peripheral when such a carry out occurs. E33 is set at TP2, if CAR OUT L is asserted. During TS3, the OMNIBUS OVERFLOW L signal is asserted; the peripheral uses this signal as determined by the peripheral programming.

The OVERFLOW L signal is commonly used with a 3-cycle data break device. In this application, the OVERFLOW L signal is gated with the word count major state within the data break control to indicate that the last word of a block is about to be transferred by the following Break (B) cycle.

SECTION 7 – CONSOLE TELETYPE CONTROL, KL8-E

3.44 TELETYPE CONTROL, GENERAL DESCRIPTION

The Teletype Control contains logic to transfer data between the Central Processor and the Teletype keyboard, reader, printer, and punch. The transmitter converts parallel information provided by the computer to serial information for the Teletype at the Teletype rate of speed. The 33 ASR Teletype requires one character every 9.09 ms; therefore, the purpose of the transmitter is to transmit (to the Teletype) one bit every 9.09 ms and to transfer a START bit, 8 DATA bits and 2 STOP bits (the format of the character) in 99.99 ms or 100 ms.

The transmitter services the teleprinter and punch. There are two operating modes for the transmit portion of the Teletype Control: punch and print. The punch mode can be disabled by the punch ON/OFF switch at the Teletype. When the punch is OFF, the data path is from the AC Register to the TTO Buffer to the printer. When the punch is ON, the data path includes both the punch and the printer.

A third combination includes the keyboard. The data path starts from the keyboard, continues to the TTI Buffer and to the AC Register. From the AC Register the data path continues to the punch buffer, to the printer, or to the printer and punch.

The receiver services the reader or keyboard. The receiver takes the serial information from the Teletype and converts it to parallel data for the computer. The operation is reversed for the transmitter. The receiver receives a bit every 9.09 ms and, when the character is fully assembled in the receiver (100 ms), the data is applied in parallel to the computer.

Two functions are provided by the receiver portion of the Teletype Control: the reader and the keyboard. The keyboard is always enabled. When a key on the keyboard is depressed, it automatically sends data from the Teletype to the receiver. This data is then transferred to the AC Register when the processor samples the Teletype receiver (TTI) buffer with an IOT instruction. The reader portion of the Teletype applies to the paper-tape reader, which can be disabled. If the processor instructs the reader to read, the reader buffer then receives information; or, the reader portion of the Teletype can be disabled at the Teletype unit, but the keyboard portion cannot be disabled. The assertion of the keyboard key automatically sends the corresponding character to the reader buffer.

3.45 TELETYPE CONTROL, FUNCTIONAL DESCRIPTION

A block diagram of the Console Teletype Control is illustrated in Figure 3-111. The primary logic/functions are illustrated by blocks. The address decoder provides selection logic to ensure that the processor is communicating with the Teletype rather than some other device. It receives bits MD3 L to MD8 L, decodes them, and signals the Operations Decoder that this IOT is addressed to Teletype. The I/O PAUSE L signal is used as a gating input to ensure that the instruction is an IOT instruction. If MD3 L to MD8 L equal 03_8, the receiver function is addressed and 04_8 addresses the transmit function.

The Operations Decoder begins to function when the address decoder signals that Teletype has been addressed. The Operations Decoder then looks at bits MD9 L to MD11 L and decodes the type of instruction to be performed. The Operations Decoder is divided into two sections: receiver functions and transmitter functions. Only one of these can be turned on during any one IOT. The Operations Decoder then enables all of the other functional blocks. If the Teletype Control is to read some information, the Operations Decoder enables the TTI Control. The TTI Control ensures that the correct information is read from the Teletype. Striking a key on the

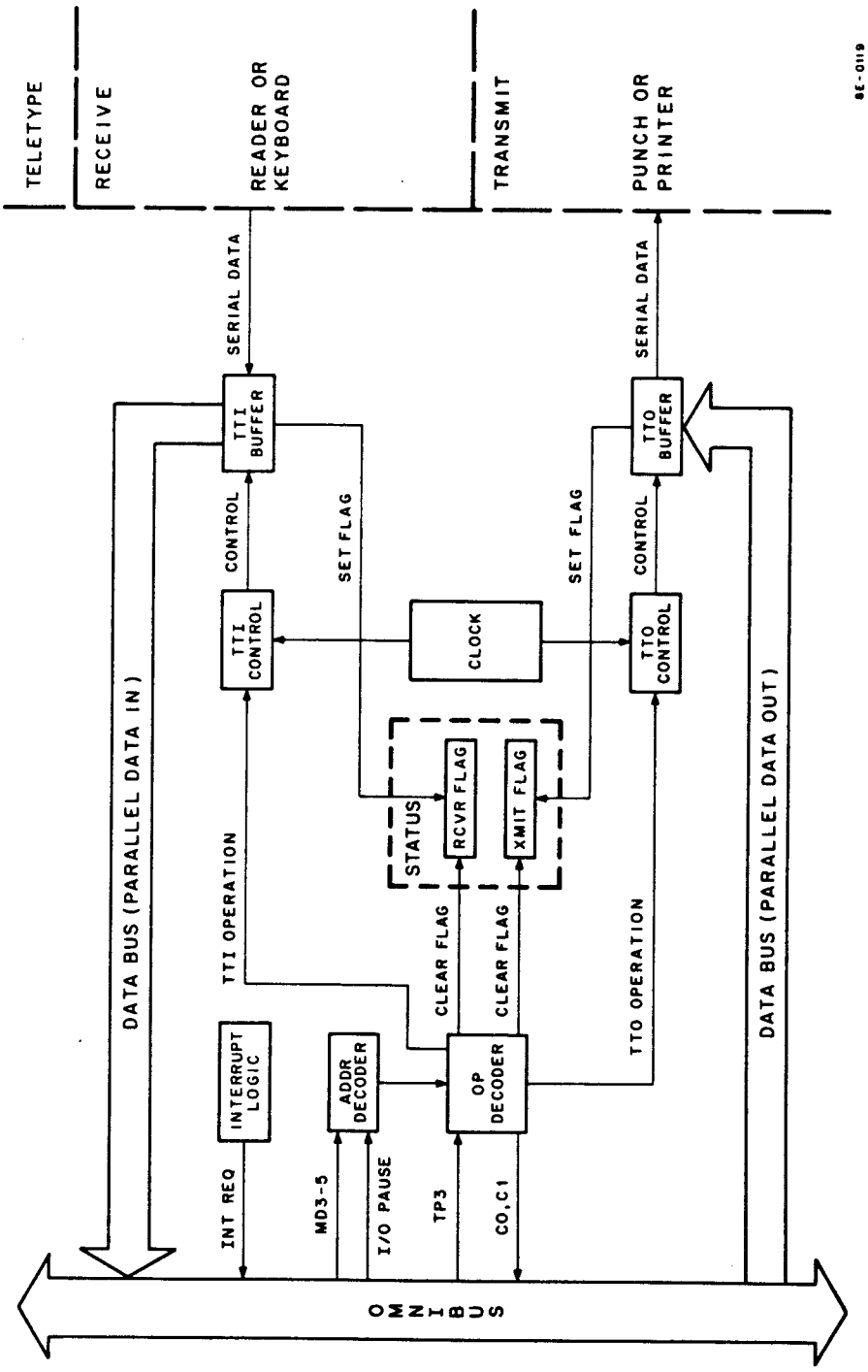


Figure 3-111 Teletype Control, Functional Block Diagram

keyboard brings data into the receiver portion of the Teletype Control. This automatically sends serial information into the receiver buffer. Of the 11 bits received from the Teletype, the first bit is the START bit. The TTI Control looks at the leading edge of the START bit. If the START bit is still present after 4.55 ms, the TTI Control is assured that it is the true START bit and allows the TTI buffer to serially shift in ten more bits from the Teletype. Otherwise, the buffer is inhibited.

After the TTI Control has determined that all of the information is in the TTI buffer, the TTI buffer sets the Reader flag to indicate that the buffer is full. The flag immediately activates the interrupt logic, if the interrupt control logic is enabled. If the Operations Decoder receives a KSF command, SKIP L is generated when the Reader flag is set.

The processor then sends a new IOT to the Teletype Control. The Operations Decoder receives this new instruction, which instructs it to read the buffer. A gating signal is developed in the Operations Decoder and applied to the buffer output gates. The eight data bits are then gated onto the DATA BUS (DATA 4–11) and loaded into the AC.

When the punch or printer is to be activated, the TTO Control logic is used. Another portion of the Operations Decoder directs the operations applying to the punch. Control signals developed in the Operations Decoder logic are applied to the TTO Control.

The TTO Control then clocks the TTO buffer input gates so that parallel data can be loaded from the OMNIBUS into the TTO buffer. The data transfer path is between the AC Register in the Central Processor to the TTO Buffer via the DATA BUS. The TTO Control then begins to clock the TTO Shift Register to enable a serial shift (one bit at a time) of the data to the Teletype at Teletype speed (9.09 ms per bit). In addition, the TTO buffer sends out a START bit, 2 STOP bits, and the 8 data bits to comprise an 11-bit word (one character). When the TTO buffer is empty, the TTO Control sets the Transmitter Flag. The set condition enables the interrupt logic. When the processor checks the flag, SKIP L is generated.

The clock is used to give a standard rate of 9.09 ms as a reference. This controls all events between the Teletype Control and the Teletype to ensure that the Teletype speed is always maintained. The events between the Teletype Control and the Central Processor are controlled by TP3 generated in the Timing Generator.

3.46 TELETYPE CONTROL, DETAILED LOGIC

3.46.1 Address Selection Logic

The Address Selection logic (Figure 3-112) decodes MD3 L through MD11 L. The device address is contained in bits 3 through 8, and the operation code is contained in bits 9 through 11. If the middle six bits are decoded as 03 (octal), the Operations Decoder is directed to the receiver functions. If the middle six bits are decoded as 04 (octal), the Operations Decoder is directed to the transmitter functions. The I/O PAUSE L signal is developed in the Timing Generator whenever bits MD0 through MD2 are decoded as a 6 (octal). I/O PAUSE L gates the address bits through the decoder. The output of the Address Selection logic provides an enabling signal to the receiver portion of the Operations Decoder or an enabling signal to the transmitter portion of the Operations Decoder. It also generates the INTERNAL I/O L signal, which is used in the positive I/O Bus Interface to prevent the generation of IOPs.

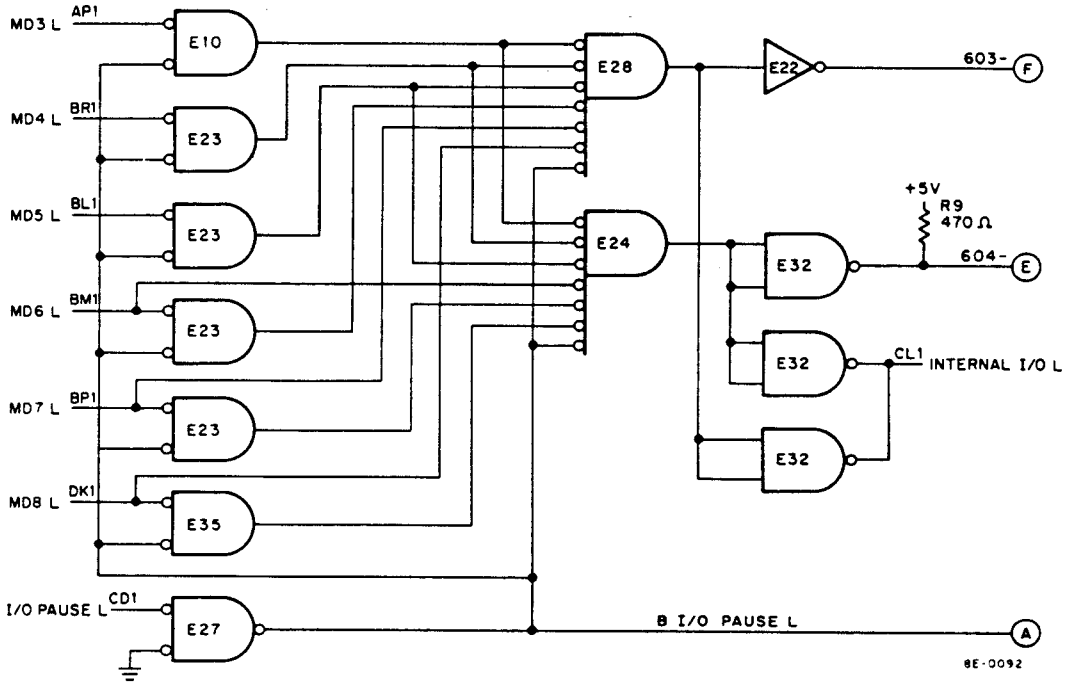


Figure 3-112 Address Selection, Logic Diagram

3.46.2 Interrupt Control Logic

The Interrupt Control logic functions to either enable or disable interrupts from the Teletype (Figure 1-113). Every time a flag is set (either receiver or transmitter), the INT RQST L signal is asserted, if the INTER ENABLE flip-flop is set (the 1 side is high). By modifying the state of the INTER ENABLE flip-flop, the INT RQST L signal can be either asserted or inhibited. To inhibit INT RQST L, a 0 is first placed into AC11L and applied to the DATA BUS. DATA 11 L is gated into the data input of the INTER ENABLE flip-flop by the I/O PAUSE L signal. The data input is clocked in by IOT instruction 6035, which is decoded by the Operations Decoder and applied to the INTER ENABLE flip-flop. This causes the 1 output to go low and negate gate E33 to inhibit the INT RQST L signal. The other qualifying input to E33 is the state of the receiver or transmitter flag. If AC11 L is a 1, the INT RQST L signal is then applied to the Interrupt Control logic via the OMNIBUS as the interface signal that starts the interrupt system sequence of events. The INTER ENABLE flip-flop is also set by CAF and by operation of the CLEAR key.

3.46.3 Operations Decoder Logic

The Operations Decoder logic (Figure 3-113) receives either 03 or 04 (octal) from the address selection logic to enable the receive or transmit logic. The operation to be performed is determined by the last three MD bits, MD9 L through MD11 L, to provide instructions such as SKIP, CLEAR FLAG, SET BUFFER, etc. These functions are given in the timing diagram in Figure 3-114, with respect to signals I/O PAUSE L, TS2 L, TP3, and TS4 L. The Teletype is in control as soon as the I/O PAUSE L signal is asserted low, which occurs 100 ns after the beginning of TS2. Signal SKIP L is generated when the MD9 L through MD11 L bits are decoded as 1 or 5 (octal). The clear operations are accomplished when MD9 L to MD11 L results in a 2 (octal). Table 3-11 illustrates the decoding functions in terms of the receive or transmit function and the last three MD bits.

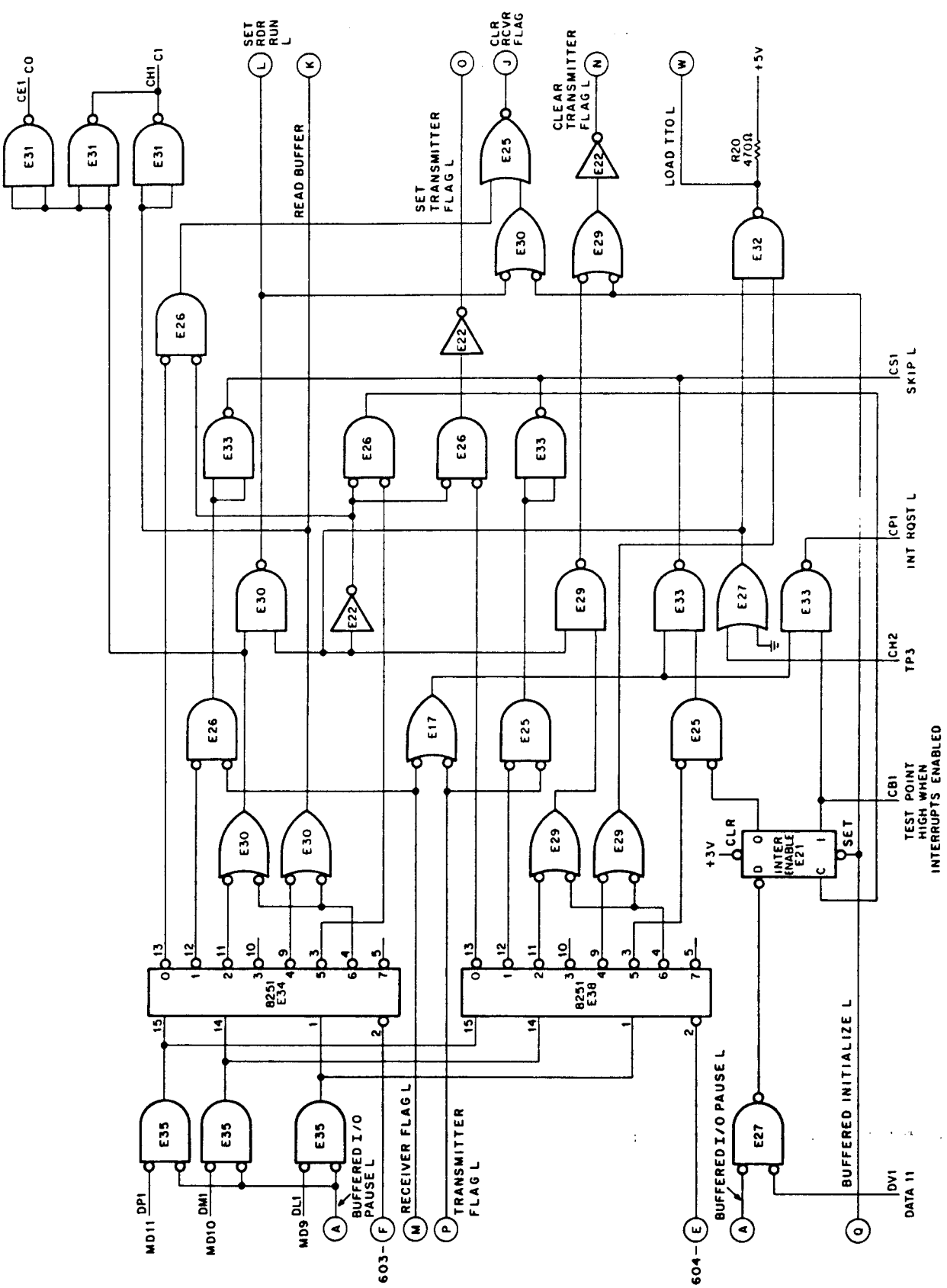


Figure 3-113 Operations Decoder, Logic Diagram

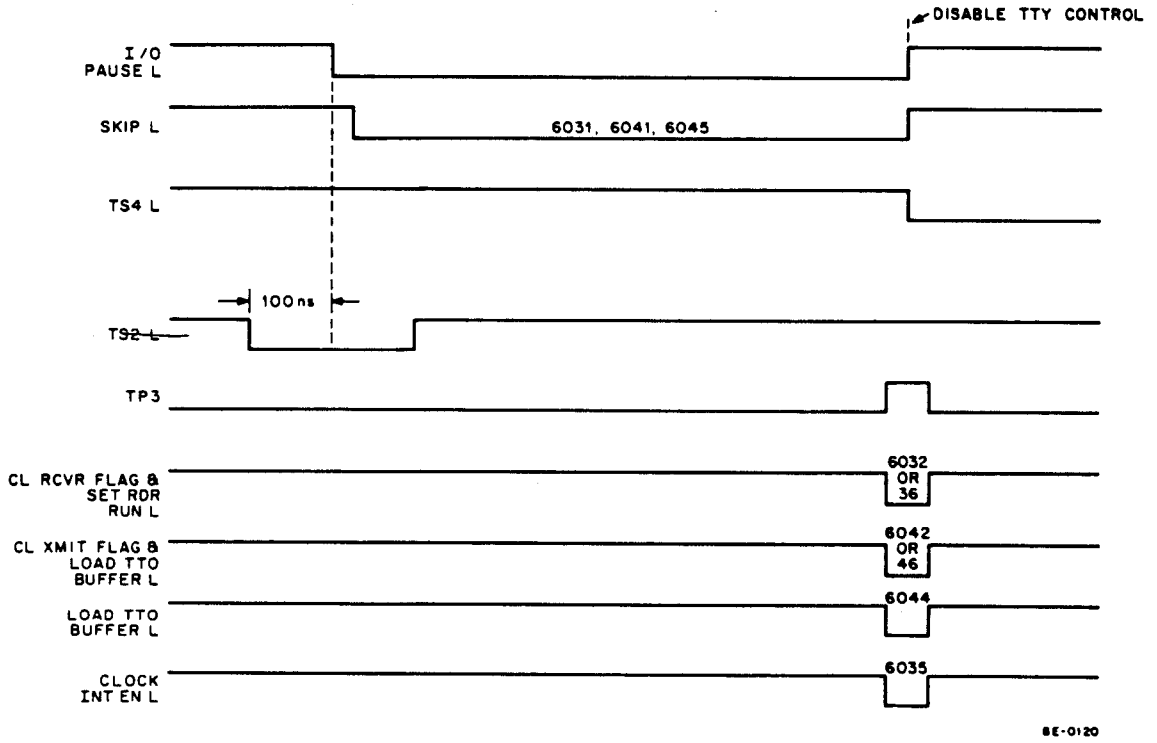


Figure 3-114 Operations Decoder, Timing Diagram

Some of the resulting control signals developed in the Operations Decoder are illustrated in the timing diagram. All are enabled by TP3 except the SKIP instructions; these occur when I/O PAUSE L is asserted. Other important control signals developed in the Operations Decoder include C0 L and C1 L, which are used to determine the type of data transfer between the processor and the Teletype Control. When the data in the TTI buffer is transferred to the DATA BUS, the 6036 instruction asserts C0 L and C1 L and takes the contents of the buffer and places it in the AC Register.

3.46.4 RCVR Flag

The RCVR flag (Figure 3-115) is connected to the end of the TTI buffer (Figure 3-117). The flag is set when the START bit is shifted out of the last stage of the TTI Shift Register. The START bit, therefore, sets the flag after all eight bits have been loaded into the TTI buffer. The 0 side of the RCVR flag flip-flop goes from a high to a low. At reference point (M), this low level signal is applied to the Operations Decoder (which looks at both the RCVR and Transmitter flags). The signal is then applied to the Interrupt Control logic (Figure 3-113), where this signal and the 1 side of the INTER ENABLE flip-flop is used to qualify the INT RQST gate and, thus, assert INT RQST L. The programmer clears the flag with the 6032 instruction. The flag indicates that the buffer is full, and that there is information that can be transferred to the AC Register.

Table 3-11
Operations Decoder Functions

REC 03	XMIT 04	MD9 L – MD11 L (octal)	Basic Operation	Off Page Reference	Controlled Logic
x		0	Clear Receiver Flag – Do not start Reader	J	RCVR Flag – Figure 3-115
x		1	Generate SKIP if Receiver Flag is set	M	Operations Decoder – Figure 3-105 RCVR Flag – Figure 3-115
x		2 (C0 L & C1 L = L)	Clear Receiver Flag, Clear AC, Set Reader Run	J	RCVR Flag – Figure 3-115
x		4 (C1 L = L)	Read TTI Buffer (parallel transfer of TTI Buffer to DATA BUS)		TTI Buffer – Figure 3-117
x		5	Sets or clears the INTER ENABLE flip-flop depending on AC11 being a 0 or 1	D	Interrupt Control Logic – Figure 3-113
x		6 (C0 L & C1 L = L)	Clear AC and Flag. Transfer TTI Buffer to DATA BUS	J K	RCVR Flag – Figure 3-115 TTI Buffer – Figure 3-117
	x	0	Sets the Transmitter Flag to ready the logic for another character	O	XMIT Flag – Figure 3-116
	x	1	Generate signal SKIP L if Transmitter Flag is set	P	Operations Decoder – Figure 3-113
	x	2	Clear the Transmitter Flag	N	XMIT Flag – Figure 3-116
	x	4	Enable transfer of TTO Buffer data to teleprinter or punch	W	TTO Buffer – Figure 3-120
	x	5	Generate signal SKIP L if the Transmitter Flag is set and the INTER ENABLE flip-flop is set	B	Operations Decoder – Figure 3-113 Interrupt Control – Figure 3-113
	x	6	Clear the Transmitter Flag and enable transfer of TTO Buffer to printer or punch	N W	XMIT Flag – Figure 3-116 TTO Buffer – Figure 3-120

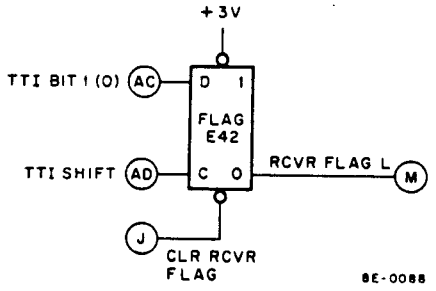


Figure 3-115 RCVR Flag

3.46.5 Transmitter Flag

The Transmitter flag (Figure 3-116) works in an opposite manner to the RCVR flag. The TTO buffer (Figure 3-120) shifts in 0s from the ENABLE flip-flop all the way up to the LINE flip-flop. When all 0s are placed in the TTO buffer, the TTO = 0 (S) signal is brought high and applied to the data input of the Transmitter flag flip-flop, which will again be brought low and applied (reference point P) to the Operations Decoder. At this point, it is applied to the Interrupt Control logic where the INT RQST L signal is generated. The processor is now informed that the Teletype transmitter is ready for another character. If the programmer checks the flag with a 6041 instruction, the SKIP L signal will be generated, which instructs the processor to skip the next instruction. A 6042 instruction will clear the flag. The programmer always clears the flag prior to a new punch operation.

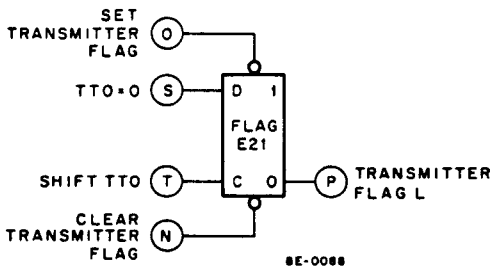


Figure 3-116 Transmitter Flag

3.46.6 TTI Buffer and Control

The TTI Buffer and Control circuits are illustrated in Figure 3-117. The TTI buffer receives an 11-bit code from the Teletype. The first bit turns the reader control circuitry on and presets the TTI buffer to all 1s. The control circuitry makes certain that a true START bit exists. To be a true START bit, the duration must be at least 4.55 ms. If a true START bit exists, the TTI Buffer Control circuit takes the incoming START bit and places it (as a 0) into the least significant bit in the TTI buffer. The output of the Clock (Figure 3-119) clocks the buffer at a 9.09 ms rate or 110 Hz. The original START bit, which was applied to bit 8 of the buffer, shifts one position to the right on each clock input. Following this, the START bit will be 8 bits of ASCII code. Two STOP bits are included at the end; thus, a total of nine clock pulses are applied to the buffer. The first clock pulse shifts in the START bit. The ninth clock pulse shifts the START bit from bit 1 to the RCVR flag and the flag is then set. The

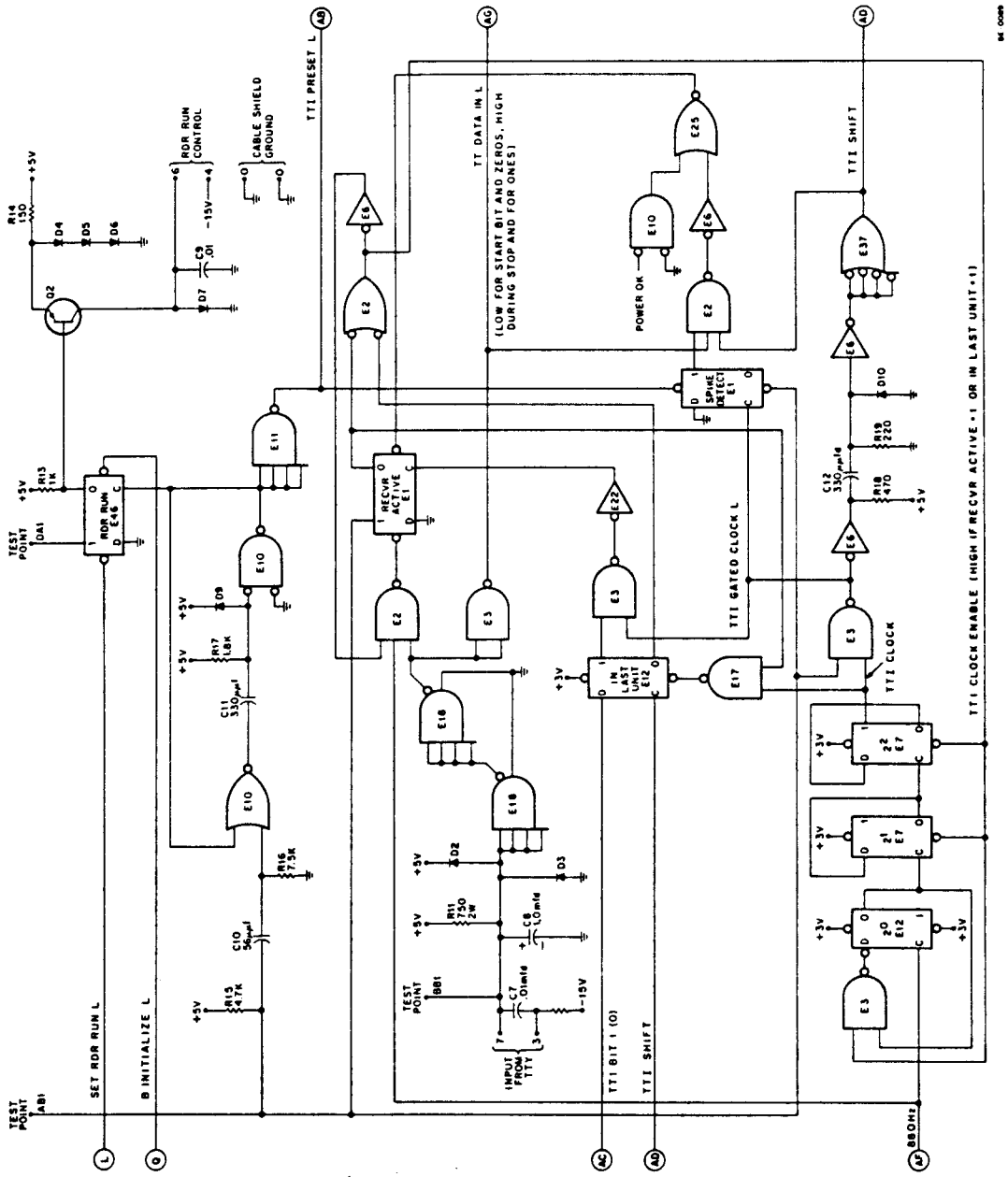


Figure 3-117 TTI Buffer and Control, Logic Diagram (Sheet 1 of 2)

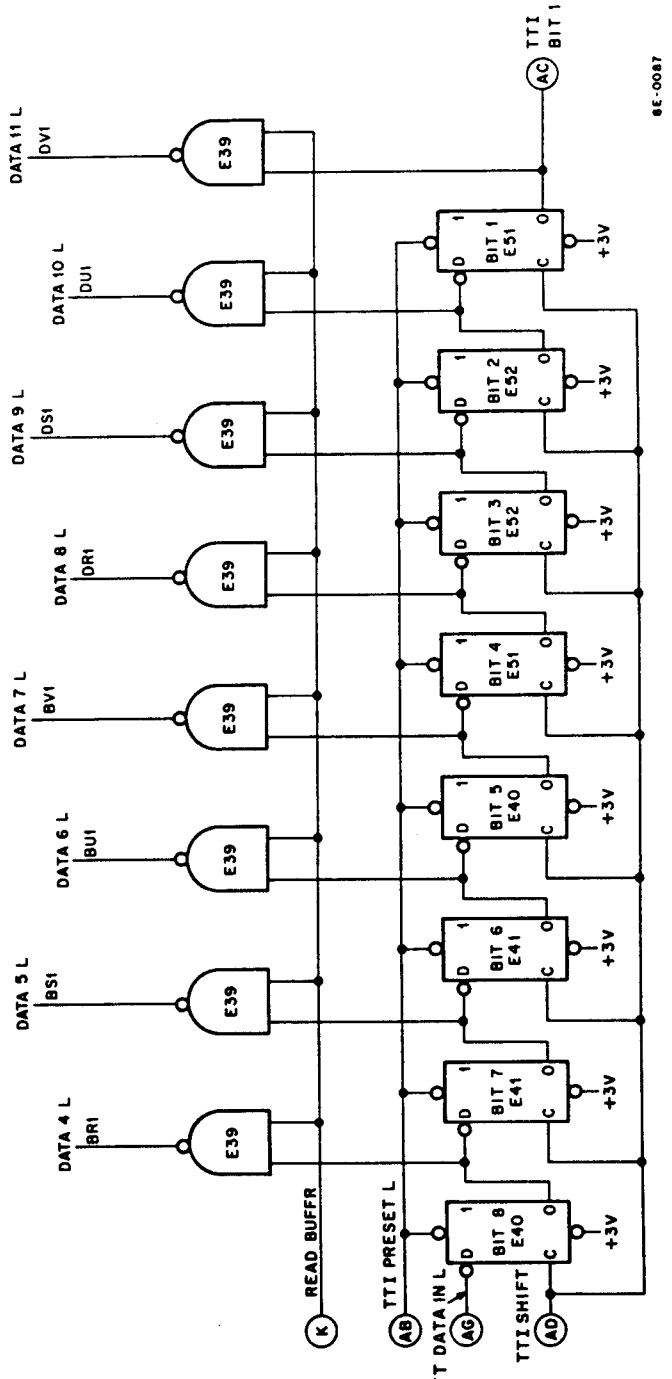


Figure 3-117 TTI Buffer and Control, Logic Diagram (Sheet 2 of 2)

START bit is also shifted into the IN LAST UNIT flip-flop in the control circuitry. The 1 side of this flip-flop is brought high and used to enable TTI GATED CLOCK, which clocks the RCVR ACTIVE flip-flop. The 0 side of the RCVR ACTIVE flip-flop is brought high when TTI GATED CLOCK goes high again (approximately the end of the last data element). At that time, the IN LAST UNIT flip-flop is cleared and the TTI CLOCK ENABLE L signal goes low. The TTI receiver is now ready to receive a new character. The RCVR ACTIVE flip-flop is the key to the entire operation; it prevents the clocking of the buffer when the buffer contains eight bits of information. The STOP bits are ignored for the receiver functions.

The SPIKE DETECTOR monitors the START pulse and ensures that it is at least 4.55 ms. If it is not, the SPIKE DETECT flip-flop causes the TTI SHIFT PULSE to clear the RCVR ACTIVE flip-flop.

The TTI timing is illustrated in Figure 3-118. Each bit lasts a duration of 9.09 ms, with a total of approximately 100 ms for all 11 bits. The serial line input is compared to the state of the flag and illustrates that the flag is set on the ninth clock pulse.

There are two methods of transferring data from the TTI Buffer to the AC Register: by a 6034 instruction or a 6036 instruction. A 6034 instruction causes the content of the buffer to be ORed with the AC and places the results in the AC. A 6036 instruction causes the content of the buffer to be placed on the DATA BUS, the AC is cleared, and the DATA BUS is loaded into the AC Register. Normally, a 6036 instruction is used. C0 L and C1 L will be asserted low by the Operations Decoder. This is used to transfer the buffer into the AC. The READ BUFFER L signal places the data in the TTI Buffer onto the DATA BUS. This qualifies the buffer output gates; whatever is in the TTI Buffer is gated out to the DATA BUS.

3.46.7 Teletype Control Clock

The clock is controlled by a 14.418 MHz crystal (Figure 3-119). The output is sent to a divide by 8 network (E9) and four divide by 16 networks (E5, E4, E8, and E13). Two outputs are available on E13. One is a 220 Hz output that is applied (reference AE) to the TTO Buffer Control where it is divided by 2 to provide 110 Hz for the TTO Buffer Control. The second output, 880 Hz, is applied (reference AF) to the TTI Buffer Control, where a 3-stage dividing network provides synchronized 110 Hz to the TTI Buffer Control Logic. The INITIALIZE signal is used to clear the dividing network when power is turned on.

3.46.8 TTO Buffer and Control

The TTO Buffer and Control logic is illustrated in Figure 3-120. The TTO Buffer functions to receive information from the AC via the DATA BUS, in parallel form, and send the information out to the Teletype in serial form. When the TTO is not transmitting a character, the LINE flip-flop is set. A START bit must first be sent by the LINE flip-flop. The XMITR ACTIVE flip-flop is originally cleared with the STOP 1 and STOP 2 flip-flops, due to the initialize state of the TTO control circuitry. Thus, when the ENABLE flip-flop (E42) clears, the XMITR ACTIVE flip-flop will be unconditionally direct set. Setting the XMITR ACTIVE flip-flop clears the LINE flip-flop (E36), which, in turn, generates the START bit for the Teletype. Because the contents of the DATA BUS (bits 4 through 11) are gated into the TTO buffer with instruction 6044 or 6046, the ENABLE flip-flop (E42) is unconditionally set and the FREQ DIV flip-flop (E20) is allowed to toggle. Each time the FREQ DIV flip-flop clears, the contents of the TTO buffer are shifted toward the LINE flip-flop. Because the ENABLE flip-flop is unconditionally set, its 1 side is transferred to bit 8 on the first 110 Hz clock input from the FREQ DIV flip-flop. This clock input also moves all other bits in the buffer up one position. Bit 1 shifts into the LINE flip-flop, and the first data bit is applied to the Teletype. This process continues at a 110 Hz clock rate until the bit that was in the ENABLE flip-flop is in the Bit 1 position of the buffer. Because all 0s are clocked in

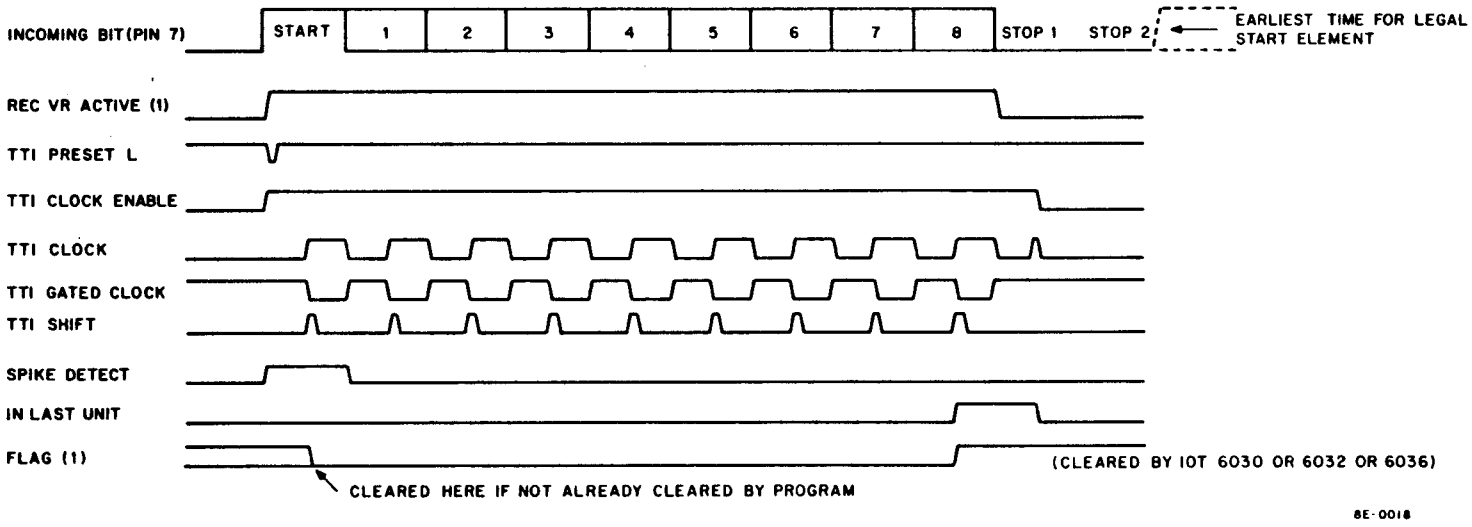


Figure 3-118 TTI Timing

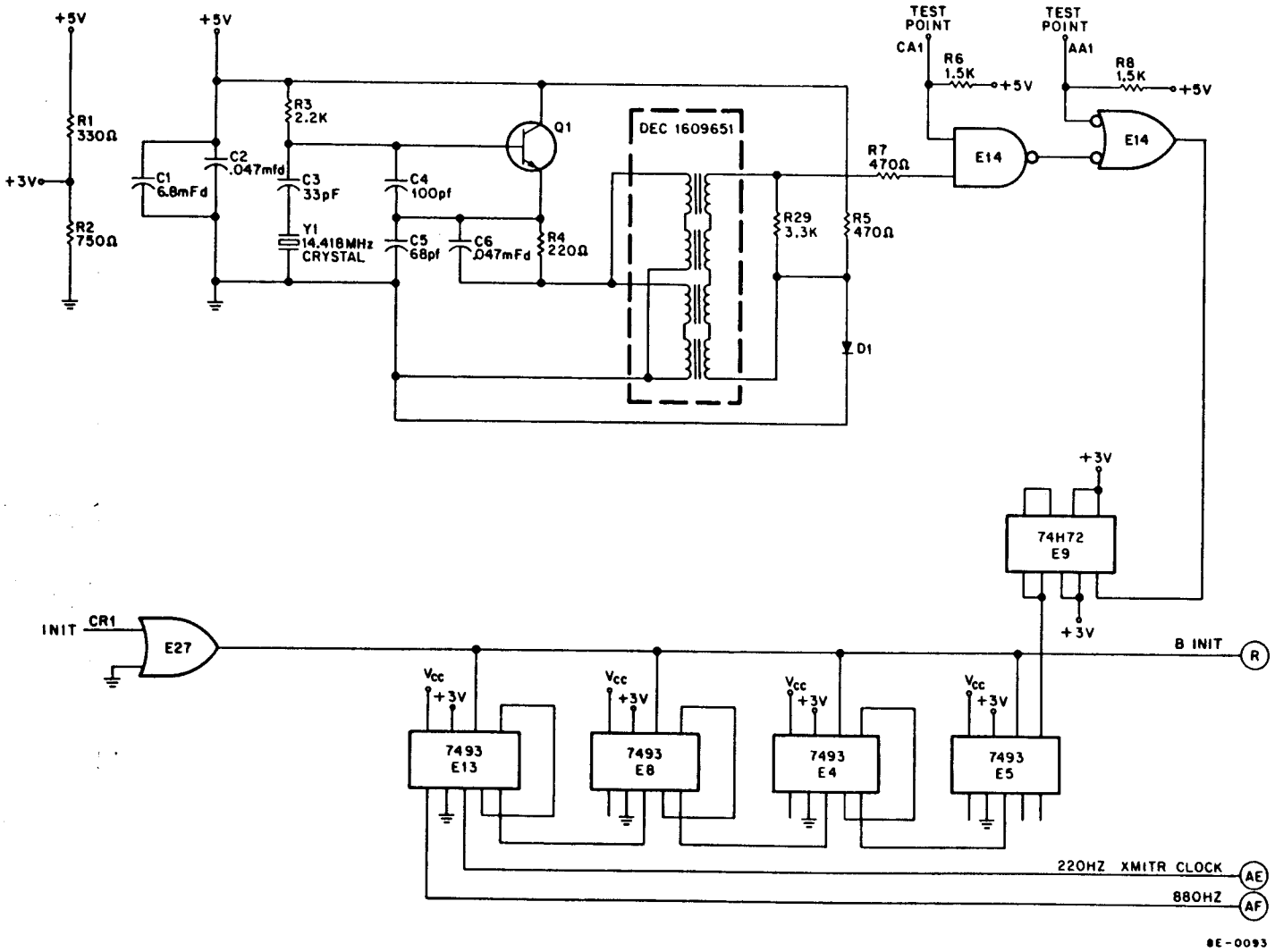


Figure 3-119 Teletype Control Clock, Logic Diagram

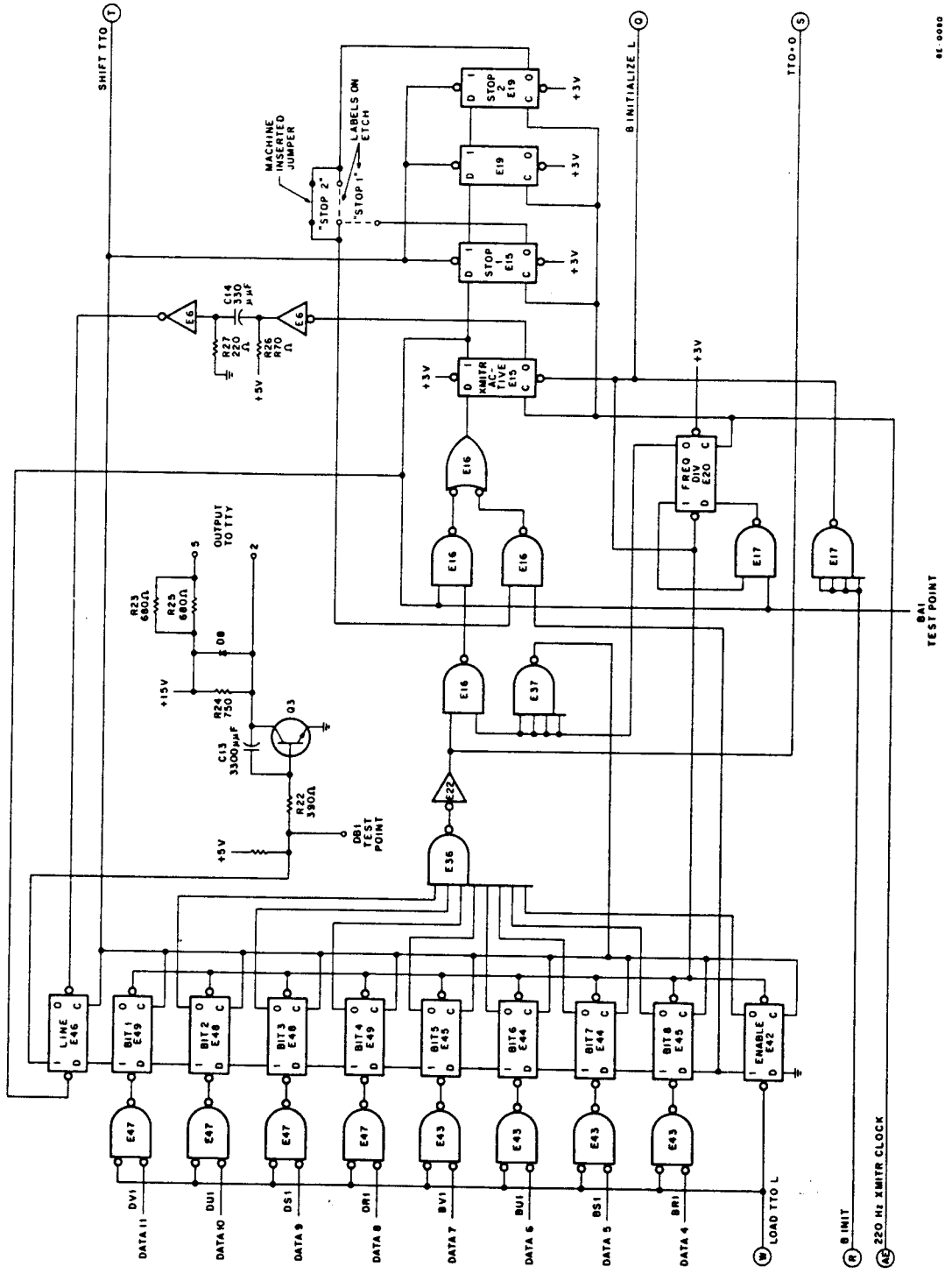


Figure 3-120 TTO Buffer and Control

behind the data bits, the 0 side of each flip-flop is high. When NAND gate E36 receives all eight qualifying inputs, the resulting output is applied to clear the XMITR ACTIVE flip-flop and set the Transmitter flag. The 1 side of XMITR ACTIVE is applied to the Stop flip-flops which, in turn, provide a delay equivalent to 2 bits (18.18 ms) before XMITR ACTIVE can be set again. (The STOP bits are set while the START and first data element were being transmitted to the Teletype.) On the 220 Hz pulse after XMITR ACTIVE clears, STOP 1 clears. On the next clock pulse, the middle flip-flop clears, its 0 side goes high again. This enabling level is then applied to the data input of the XMITR ACTIVE flip-flop.

Before a new character can be transmitted from the TTO to the Teletype, STOP 2 must clear. STOP 2 is cleared 18.18 ms after the last bit is shifted out to the Teletype. The machine inserted jumper provides 2 STOP bits. Without the jumper only one STOP bit is provided, and only STOP 1 need be cleared.

The TTO Buffer and Control Timing is illustrated in Figure 3-121. The data example (DATA = 125 (octal)) is used for illustrative purposes only. It illustrates that while the TTO buffer is shifting data out to the Teletype, the flag is cleared. When the last bit has been sent out, the flag is set. Notice that the TTO buffer can then be immediately loaded, but that transmission of the second character will be delayed until the two STOP bits have been sent.

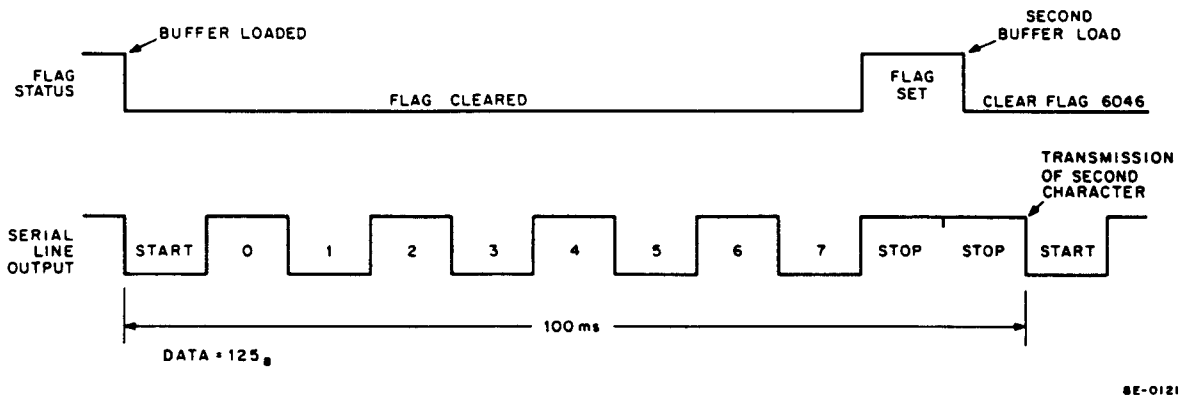


Figure 3-121 TTO Buffer and Control Timing

SECTION 8 – POWER SUPPLIES

The PDP-8/E computer uses a DEC H724 or H724A (the latter for 230 Vac lines) Power Supply that provides three regulated dc voltages, one non-regulated dc voltage, and one center-tapped ac winding that delivers 28 Vac. The PDP-8/F and the PDP-8/M use a DEC H740 Power Supply that provides three regulated dc voltages and 28 Vac, center-tapped.

Each type of power supply features dc-voltage monitoring, protection against overvoltage and thermal overload, and fusing of all dc power supplies. The PDP-8/E power supply is detailed in Paragraph 3.47; the PDP-8/F and PDP-8/M supply is described in Paragraph 3.48.

3.47 PDP-8/E POWER SUPPLY

3.47.1 Primary Network

The primary network of power transformer T1 is shown in Figure 3-122. The input ac voltage is controlled by relay K1. When the key switch on the front panel is turned to the POWER position, and the interlocks are connected, the relay closes, applying the input ac to the power transformer.

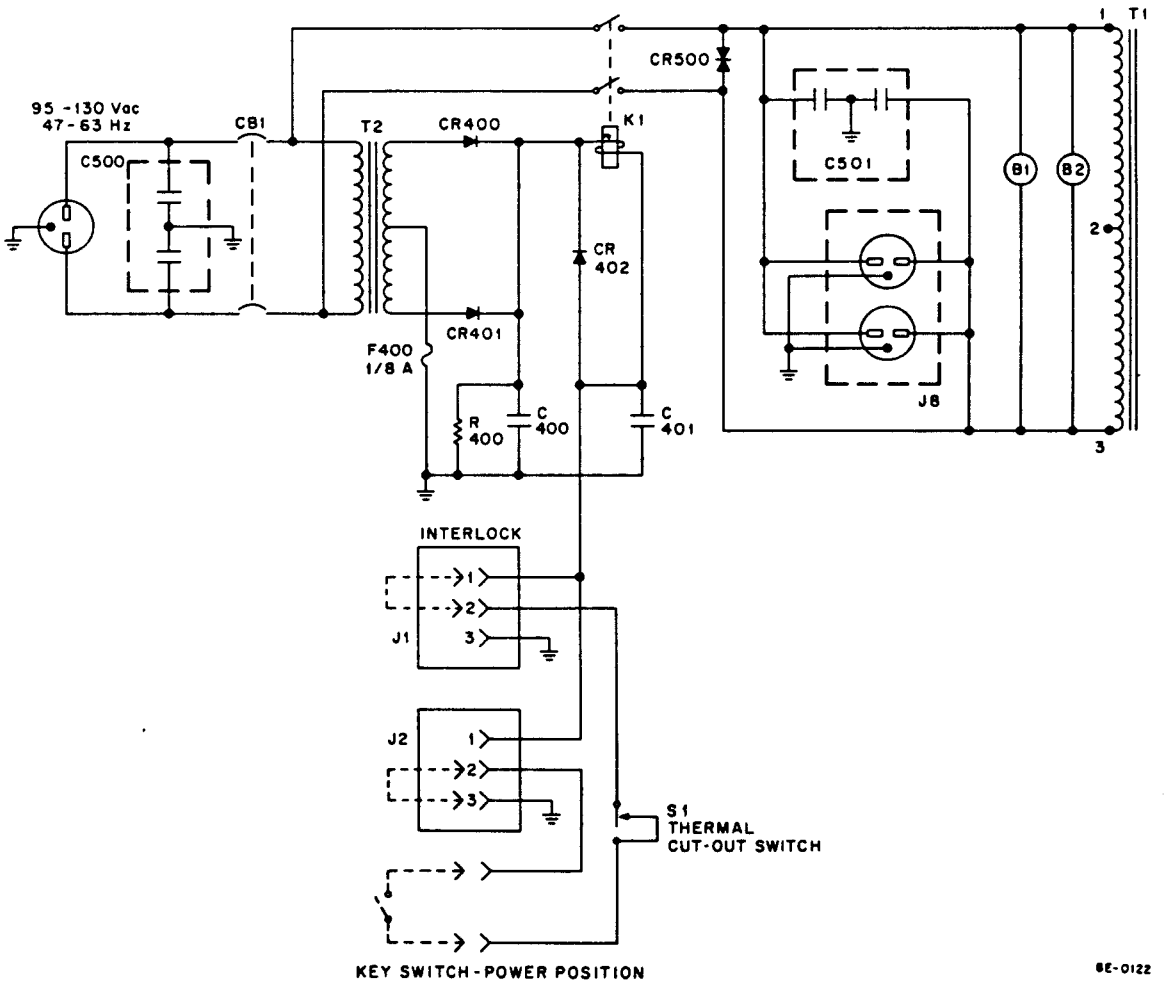


Figure 3-122 H729 Power Supply Primary Network

Switch S1 monitors the ambient temperature and opens when the temperature reaches 90°C, ±5°C. Relay K1 is, in turn, opened and removes the input ac. S1 must be reset by hand if it has been tripped.

3.47.2 +8 Vdc Power Supply

Figure 3-123 shows the +8 Vdc power supply. The +8V is obtained from a full-wave, center-tapped rectifier providing 2A, rated load. Because the +8 Vdc is used only as the supply for front panel indicators, filtering and regulation are unnecessary.

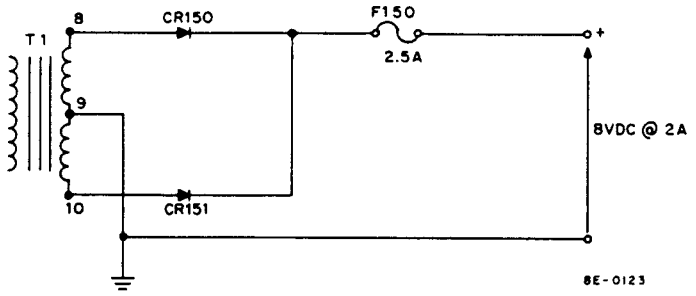


Figure 3-123 +8 Vdc Power Supply

3.47.3 +15 Vdc Power Supply

The +15 Vdc power supply is shown in Figure 3-124. The rectified dc voltage is filtered and applied to the series pass transistor, Q100, which provides +15V at its emitter. Load changes at the emitter are transferred by the R6/R7 voltage divider to the regulating difference amplifier, Q2/Q3. This amplifier, which becomes unbalanced when the base of Q3 changes from 0V, provides an error signal for emitter-follower Q1. The error signal is passed on to Q100, which then acts to correct the condition that produced the error. For example, an increase in the load on the output terminals requires that more current be supplied by Q100; therefore, the +15V output tends to decrease. The voltage divider produces a more negative level at the base of Q3. Q3 provides a positive error signal that is passed on to the base of Q100. The increase in forward-bias causes Q100 to oppose the tendency of the emitter voltage to decrease. The operating point of Q100 has been shifted; thus, the demand for more current is met, even though the collector-emitter voltage remains constant.

The +15 Vdc supply is also regulated against static and dynamic line voltage variation. Changes in the rectified dc voltage cause the collector-emitter voltage of Q100 to change in the same direction. The +15V output also changes, but in such small proportion that it essentially remains constant over the allowable ac input range.

Note that this power supply has no adjustment. The -15 Vdc regulated supply voltage works with the +15V output to develop error voltages at the R6/R7 voltage divider. In addition, the -15 Vdc voltage controls the total emitter current of the difference amplifier. Thus, static changes in the -15 Vdc output can be passed on to the +15 Vdc regulator. An adjustment potentiometer is included in the -15 Vdc supply; consequently, both supplies can be adjusted at the same time, the +15V output tracking the -15V output.

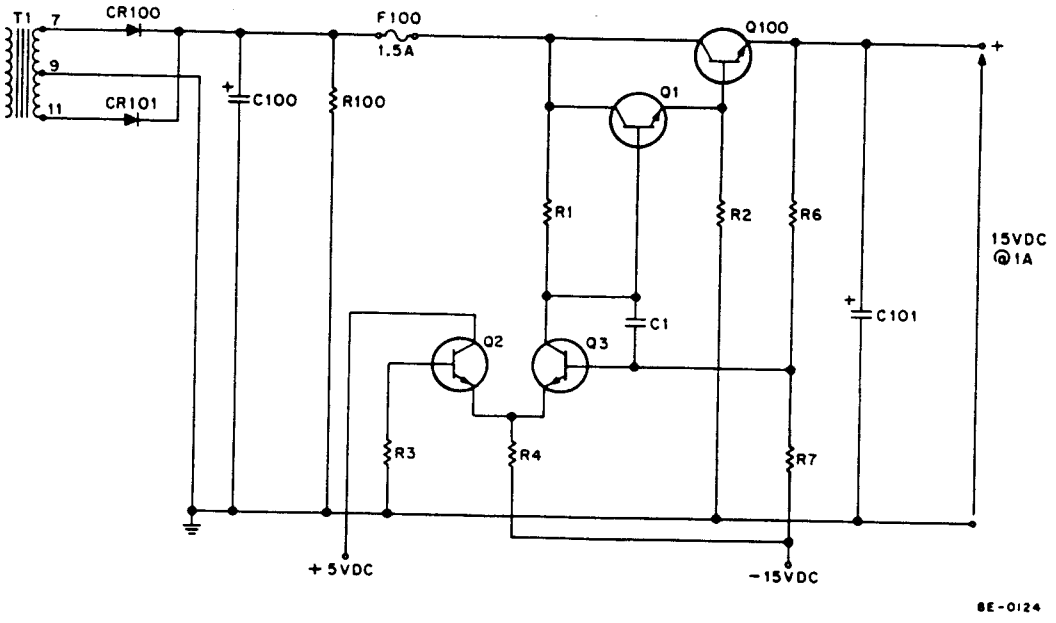


Figure 3-124 +15 Vdc Power Supply

3.47.4 -15 Vdc Power Supply

The -15 Vdc power supply is shown in Figure 3-125. This power supply is regulated in a manner similar to that described in the last paragraph. However, the procedure is carried out more precisely. The regulating amplifier is a precision voltage regulator IC, VR1. This IC contains a temperature-compensated reference amplifier, an error amplifier, and a series power-pass transistor. Pin 4 is the output of the reference amplifier. The reference voltage for the error amplifier is taken from the wiper arm of potentiometer R5, the -15 Vdc adjustment. This reference is compared with a sample of the -15 Vdc output, which is applied to the error amplifier at pin 2. The error signal is amplified and transferred, via pin 6, to Q300, which provides a change in base drive for Q301–304. Thus, static and dynamic load and line changes are regulated as in the +15 Vdc regulator.

Note that the series pass transistor consists of four transistors, Q301 through Q304, in parallel. Each transistor has a 0.25Ω, 3W resistor connected to its emitter lead. The 0.25Ω resistors encourage equal division of the regulated load current through the pass transistors. In the event of an overload that is not sufficiently large to burn out F300, there is less likelihood of damage to a pass transistor. In addition, the parallel arrangement reduces the possibility of a pass transistor burning out before F300 if the output terminals are shorted.

3.47.5 +5 Vdc Power Supply

The +5 Vdc power supply is shown in Figure 3-126. Again, precise regulation is possible through the use of a voltage regulator IC, VR5. The reference voltage is taken from potentiometer R21, the +5V adjustment, and compared to the +5V output. The error signal controls the parallel pass transistors.

3.47.6 dc Voltage Monitor Circuit

The dc voltage monitor circuit is shown in Figure 3-127. This circuit negates the POWER OK H signal whenever a regulated dc voltage is less than an established absolute value. Such a condition occurs when the power is turned on or off, or when a failure occurs within the power supply. These absolute values are 4.3V for +5 Vdc voltage, 12V for +15 Vdc voltage, and 13.5V for the -15 Vdc voltage.

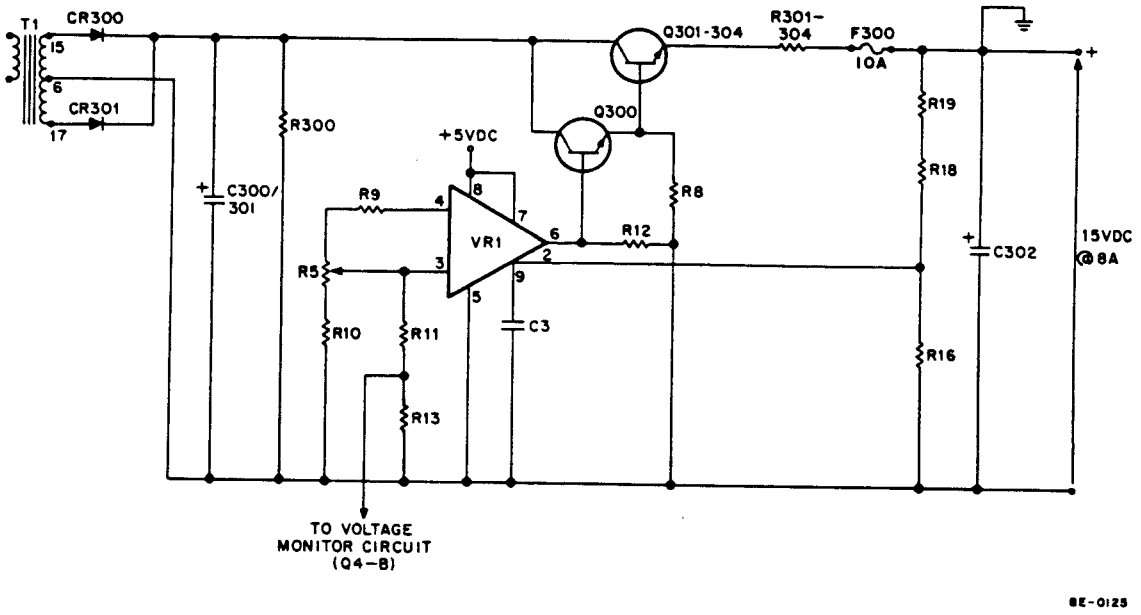


Figure 3-125 -15 Vdc Power Supply

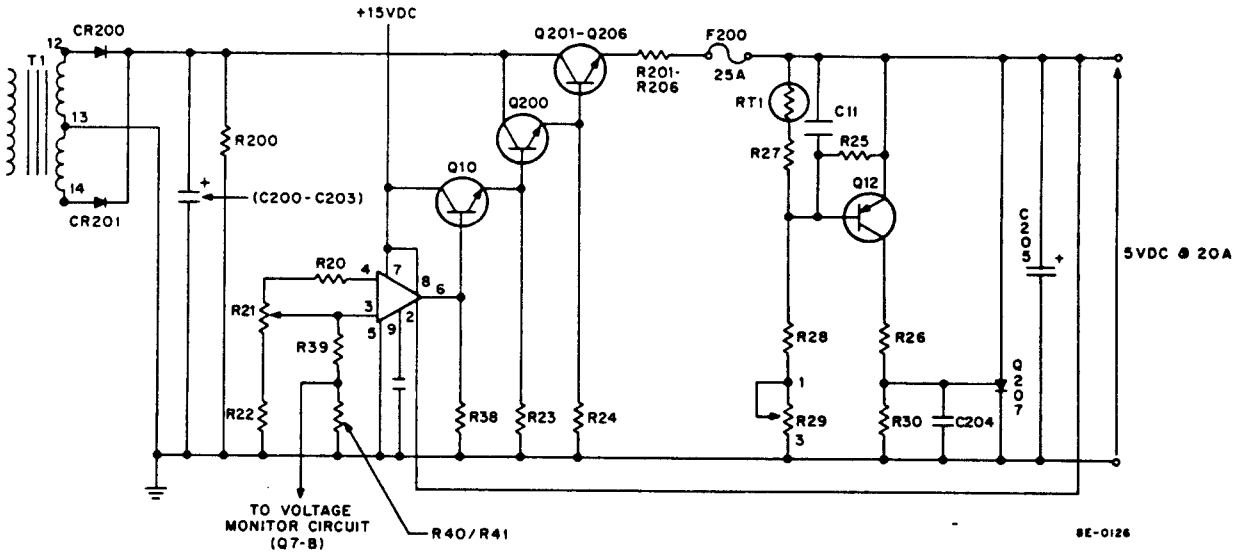


Figure 3-126 +5 Vdc Power Supply

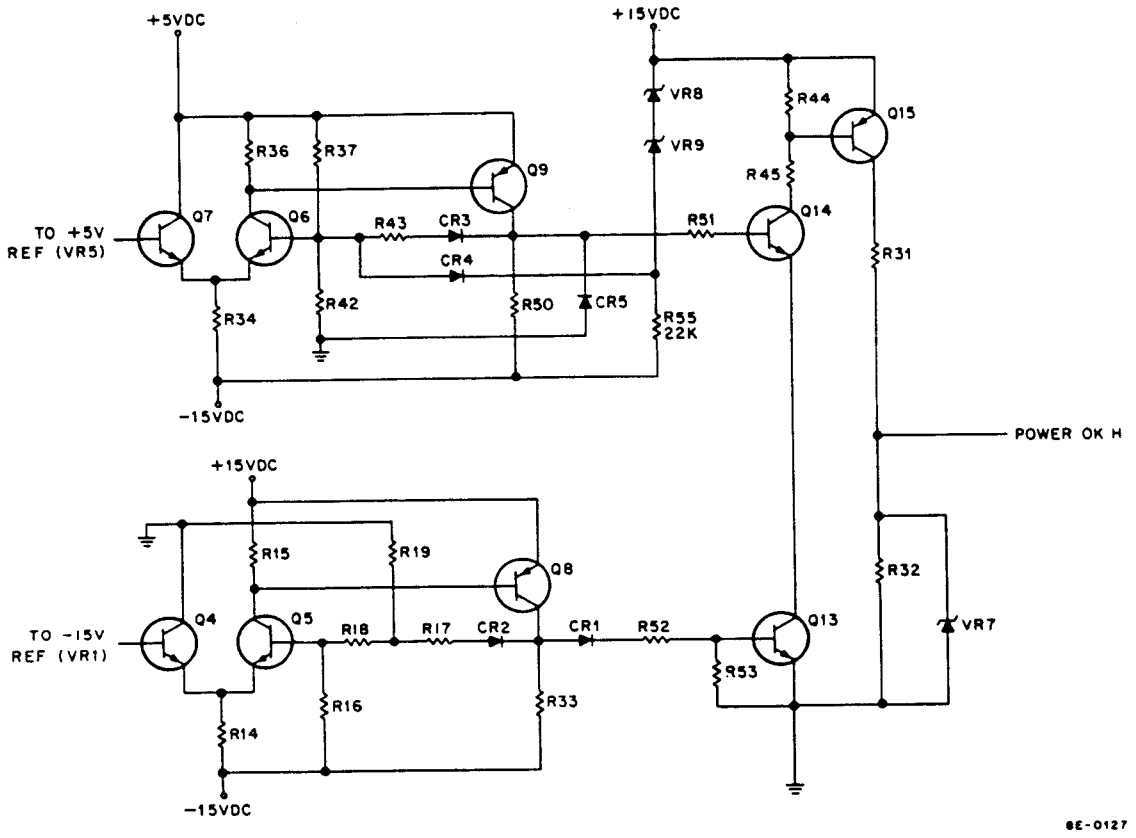


Figure 3-127 dc Voltage Monitor

If all three regulated voltages are above the limit, Q13, Q14, and Q15 are switched on. Zener diode VR7 maintains the POWER OK line at approximately 4.3V. If the -15 Vdc voltage goes more positive than -13.5V, Q13 is switched off, leaving the emitter of Q14 floating. Q15 turns off and the POWER OK line goes to ground. If the +15 Vdc voltage goes more negative than +12V, or if the +5 Vdc voltage goes more negative than +4.3V, Q14 is switched off, accomplishing the same result as before.

The circuits that control the switching of Q13 and Q14 are nearly identical. The circuit that includes Q6, Q7, and Q9 monitors both +5 Vdc and +15 Vdc. The Q6/Q7 difference amplifier is balanced when +5 Vdc is satisfactory. If +5 Vdc goes negative, Q6 is turned off, removing the emitter-base voltage of Q9, which turns off. The cathode of CR3 goes to ground, causing Q14 to turn off. If, instead of +5 Vdc, +15 Vdc goes more negative than its limit, CR4 is allowed to conduct. This action causes Q6 to turn off and, in turn, Q9 and Q14.

The circuit that includes Q4, Q5, and Q8 monitors the -15 Vdc voltage and works almost identically to the upper circuit. In this case, Q5 is turned off when -15 Vdc goes more positive. This action removes the base current from Q8. This removes base current from Q13, which then turns off.

3.47.7 Overvoltage Protection

The +5V power supply is provided with overvoltage protection, in the form of an SCR trigger circuit (Figure 3-126). If the +5V output rises to 6.5V, Q12 provides a triggering voltage for the gate of the SCR, Q207. The SCR conducts, and the resulting short circuit on the output terminals causes fuse F200 to burn out.

3.48 PDP-8/F and PDP-8/M POWER SUPPLY

3.48.1 Input Circuit

The H740 Power Supply input circuit is shown in Figure 3-128. The ac input is shown for a 130 Vac line only. When the turn-key switch is placed in the POWER position, line voltage is applied to the transformer and the fans. The thermostat on the regulator board opens if the ambient temperature reaches 100°C; the switch closes automatically when the temperature returns to approximately 64°C.

The line voltage is transformed to 28 Vac, center-tapped, and applied to the regulator board assembly via the secondary harness. All dc voltages are derived from this 28 Vac output. Connector J1 of the secondary harness is to be used for connecting either a KP8-E option (Power Fail Detect and Auto-Restart) or a DK8-EA option (Real-Time Clock Line Frequency).

3.48.2 +15 Vdc Power Supply

The +15 Vdc power supply (Figure 3-129) is series-regulated. The pass transistor, Q1, is a high-gain power Darlington and is mounted on the heat sink. Base drive current is supplied to Q1 via R38. Q3 limits the value of this current by shunting it away from the base of Q1. Q4, the voltage detector amplifier, biases Q3 and, thus, limits current in Q1. The +15 Vdc output is sampled by the voltage divider of R34, R35, and R36 and compared to the voltage across reference diode D8. If the output tends to change from the regulated value, Q4 generates an error signal. The error signal is returned to Q1, via Q3, and causes Q1 to correct the condition that produced the error.

Static and dynamic line voltage variations are also controlled by the power supply. Changes in the rectified dc voltage cause the output voltage to change, but in such small proportion that the output remains essentially constant over the allowable ac input range.

Transistor Q2 is an overload detector. When the output current reaches 1.5A the voltage across R33 is large enough to cause Q2 to conduct; thus, the base drive is removed from Q1, thereby limiting the output current.

3.48.3 +5 Vdc Power Supply

The +5 Vdc power supply is shown in Figure 3-130. This supply is similar to the +15 Vdc supply; i.e., the output voltage is sampled and compared to the voltage across a reference diode, and an error signal is developed that causes the pass transistor to rectify the error.

However, regulation of the output voltage is more efficient than in the +15 Vdc supply. The +5 Vdc regulator operates in a switching mode; the entire circuit is a power Schmitt trigger that is either on or off depending on the output voltage level. When Q6 is on, it supplies current through the filter choke, L1, to the output smoothing capacitor, C7, and the load. When Q6 is off, the L1 current decays through commutating diode D10, which becomes forward biased by the back EMF of L1. The waveform across D10 is a 30V rectangular pulse train. The filtered output across C7 is +5 Vdc with a 200 mV, peak-to-peak, 10 kHz sawtooth ripple. At the crest of the ripple Q6 turns off; at the valley of the ripple Q6 turns on. This switching mode of operation limits the dissipation in the circuit to the saturated forward losses of Q6 and D10, and the switching losses of Q6. Therefore, the heat sink can be smaller than in the +15 Vdc supply, and the number of power semiconductors can be fewer.

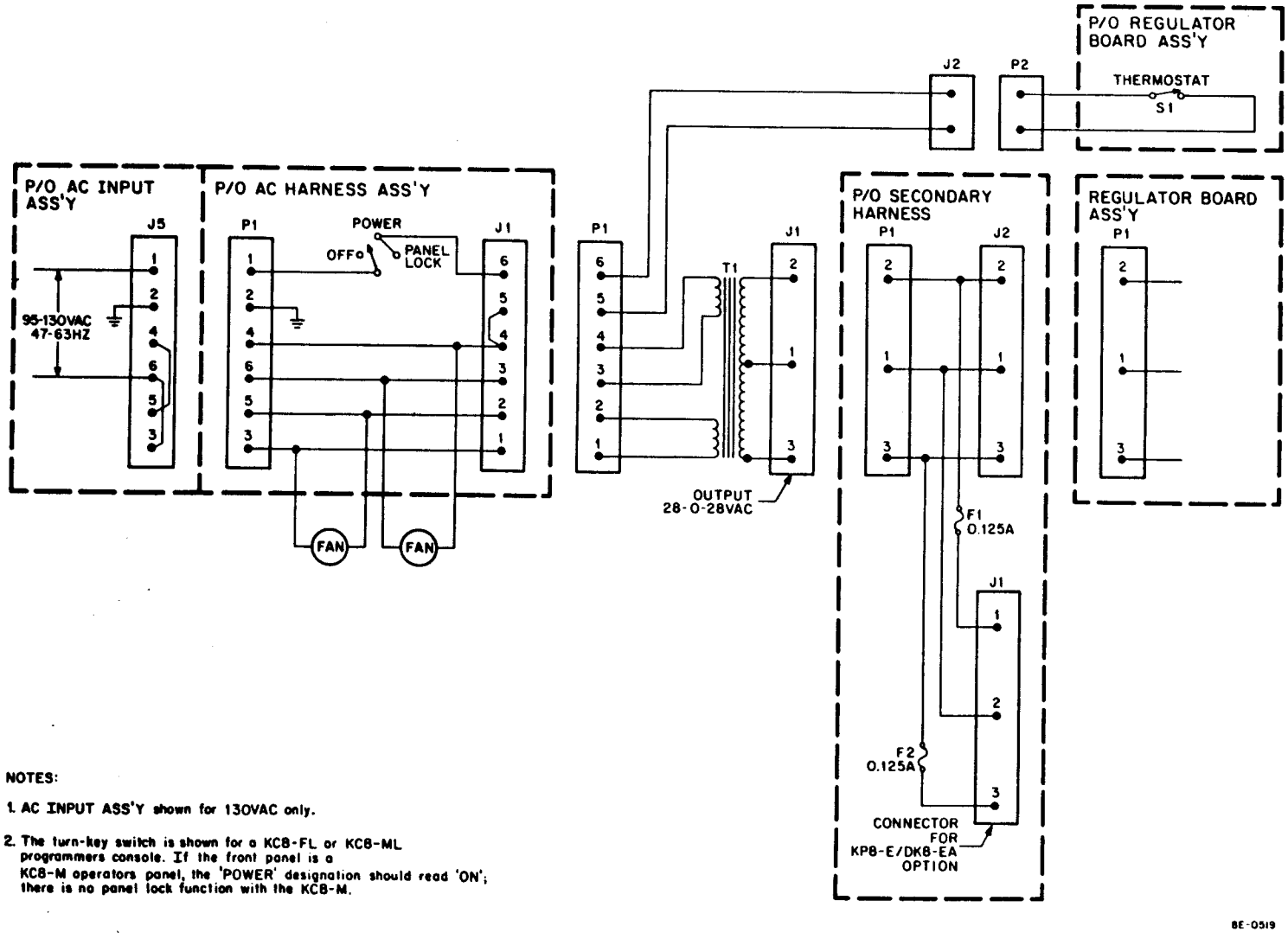


Figure 3-128 H740 Power Supply Input Circuit

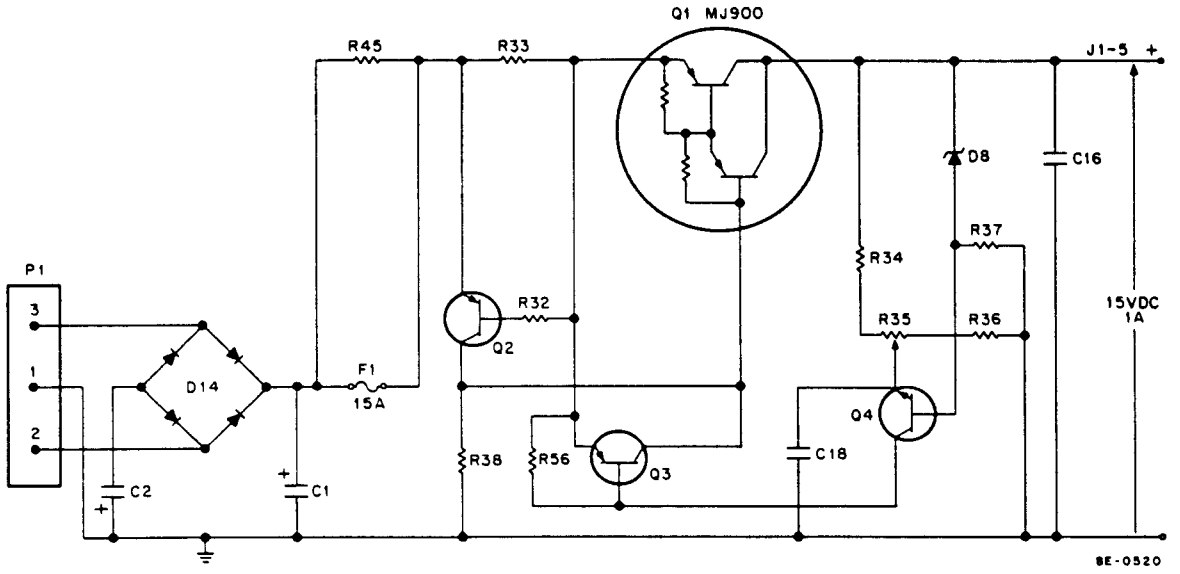


Figure 3-129 +15 Vdc Power Supply

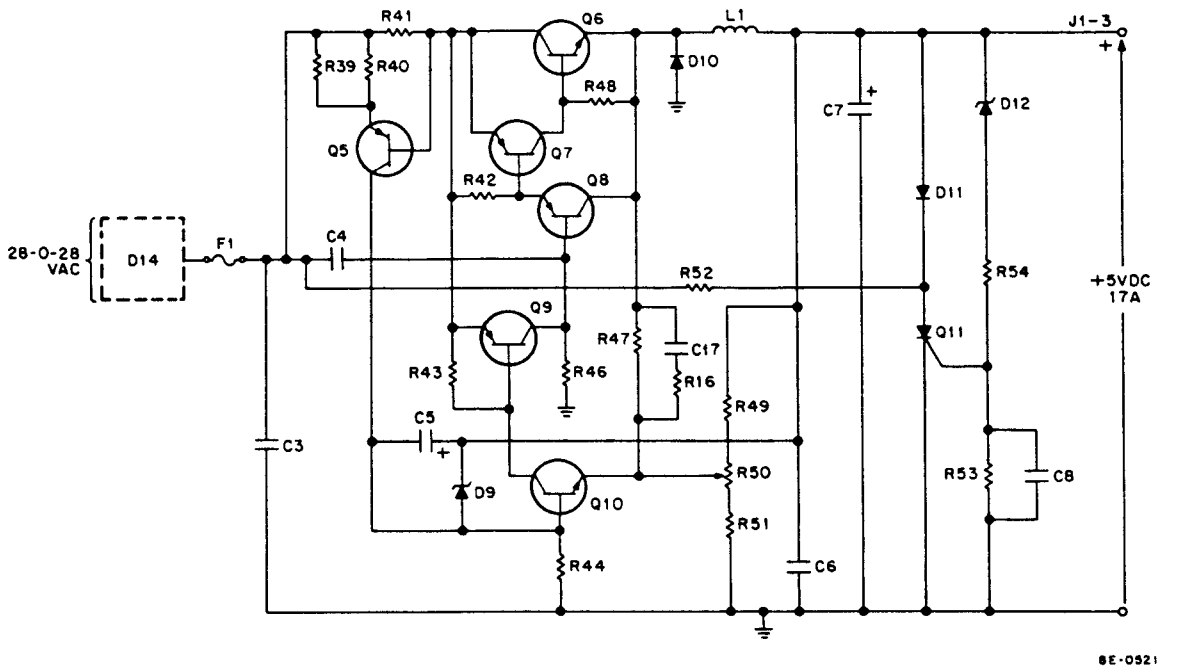


Figure 3-130 +5 Vdc Power Supply

Transistor Q5 detects an over-current condition when the voltage across R41 is sufficiently high. Output current is limited to a safe value since conduction of Q5 makes the reference voltage across D9 decrease to zero. Thus, Q10 conducts, shutting down the regulator.

If a fault causes the output voltage to increase beyond 6.3V, diode D12 conducts. The increased voltage on the gate of the SCR, Q11, fires the SCR, forcing the output voltage to a low value through D11 and causing F1 to burn out.

3.48.4 -15 Vdc Power Supply

The -15 Vdc power supply is shown in Figure 3-131. The circuit is the complement of that of the +5 Vdc supply. Minor differences in component types and values are necessary, but the switching mode of regulation is identical to that described in the preceding paragraph.

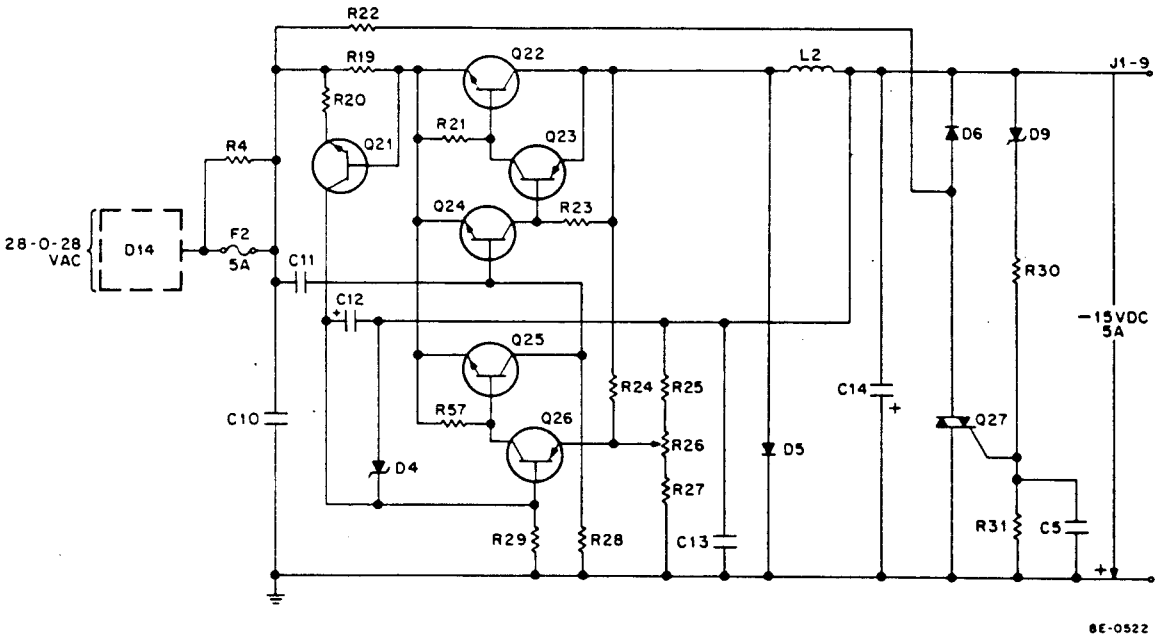


Figure 3-131 -15 Vdc Power Supply

3.48.5 Voltage Monitor Circuit

The voltage monitor circuit is shown in Figure 3-132. The circuit monitors the 28 Vac output of transformer T1 and the output of the +5 Vdc and +15 Vdc regulators.

The 28 Vac is rectified and filtered by diodes D1 and D2 and by capacitor C9, respectively. As C9 charges after power-on, transistor Q16, which is off initially, begins to conduct, while Q15 starts to turn off. A few milliseconds after power-on, the feedback provided by Zener diode D3 turns Q16 on heavily. The collector voltage of Q16 is positive enough to turn off both Q14 and Q13. The base of Q12 goes toward +5 Vdc and the POWER OK signal goes high.

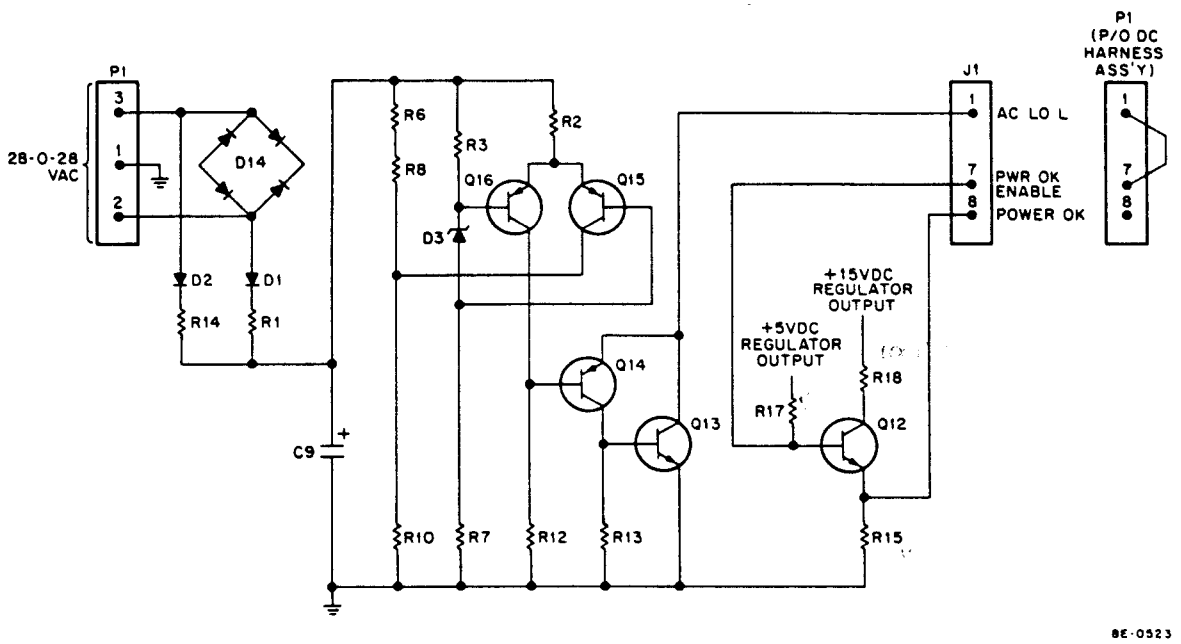


Figure 3-132 Voltage Monitor Circuit

Circuit parameters are chosen so that the regulated dc voltages are stable before Q14 and Q13 are turned off. If either the +15 Vdc output or the +5 Vdc output is missing or drops during normal operation, POWER OK is negated and the timing generator is halted.