SECTION 6
REAL-TIME PROGRAMMABLE CLOCK (DK8-EP)

4.6 INTRODUCTION

The DK8-EP Real-Time Programmable Clock consists of two PDP-8/E quad modules that plug into the OMNIBUS and are interconnected by an H851 Edge Connector. The M860 module, Real-Time Clock Control, contains a 12-bit binary counter, a 12-bit buffer register, and logic. The logic controls the counter/register operation and the companion quad module operation. This second quad module is the M518, containing Input logic and Schmitt triggers. Included on this module is a 12-bit storage register, logic that derives five different clock frequencies from a crystal-controlled oscillator, and three Schmitt trigger circuits that enable the user to control certain clock operations from external sources.

The DK8-EP provides a programmable time base that allows the user to control and/or record the occurrence of events both internal and external to the PDP-8/E. The clock can be used to count the number of events in a given amount of time, to measure the amount of time between two given events, or to initiate repetitive operations at specified intervals of time.

To perform these and similar operations, the major logic components listed must interact. The fundamental component is the 12-bit storage register, called the Clock Enable Register. This register can be loaded, bit by bit, under program control. Different bits are used to control different functional sections of the DK8-EP. For example: bit 7 of the register enables internally-generated clock pulses to be applied to the Clock Counter Register; bit 9, by enabling one of the Schmitt trigger circuits, allows an external source to control some aspects of clock operation. The Clock Enable Register can be set or cleared under program control and its contents can be transferred to the CPU AC Register at any time by a program instruction.

The Clock Counter Register counts clock pulses in a way that is predetermined by the state of Clock Enable Register bits 1 and 2. For example, the Clock Counter Register can count from 0 to 4096 repetitively, generating a signal (OVERFLOW L) each time it overflows to 0. In other circumstances, the programmer might wish the OVERFLOW L signal to be generated each 2000 clock pulses. The Clock Counter Register can be preset, to a count of 2096 in this example, in order to produce the desired result. The contents of the Clock Counter Register can be transferred to the AC Register at any time by a program instruction; however, to accomplish this transfer, the logic makes use of a 12-bit Clock Buffer Register.

The Clock Buffer Register is essential to the DK8-EP operation. Data is transferred between the AC Register and the Clock Counter Register via the Clock Buffer Register. To preset the Clock Counter Register, as in the example given, the program causes the Clock Buffer Register to be loaded from the AC Register with the preset count. Bits 1 and 2 of the Clock Enable Register can then be set so that the preset count, 2096, is loaded into the Clock Counter Register each time the OVERFLOW L signal is generated. Thus, the Clock Counter Register counts repetitively from 2096 to 4096. The Clock Buffer Register can be controlled by the program, and to a certain extent, by an external source. For example, Clock Enable Register bits 9–11 allow an external source to cause the Clock Buffer Register to be loaded; however, transfers between the AC and Clock Buffer Registers are limited to program control.

4.6.1 Block Diagram Discussion

The functional block diagram shown in Figure 4-51 illustrates the relationship between the registers and the remainder of the clock logic. Each functional block is discussed in detail in the following paragraphs and the applicable logic can be found in the indicated figures.

The IOT Decoder logic provides signals in response to the eight IOT instructions listed in Table 4-9. These IOT signals are distributed to the various functional sections as shown in Figure 4-52. The enabling signals from the Clock Enable Register are shown on the block diagram and are listed and explained in detail in Table 4-10.

For operations not involving external sources, the signal flow description can begin with the Clock Rate Select logic. The desired frequency of clock pulses is selected by the Rate Enable bits, while the Count Enable bit gates the clock pulses to the Load Buffer Register logic. This logic converts the clock pulses to B COUNT pulses that are counted by the Clock Counter Register. The logic also generates the LOAD BUFFER signal in response to program instructions and ensures that the B COUNT and LOAD BUFFER signals do not occur at the same moment (the significance of this is detailed in Paragraph 4.6.2.4).

The LOAD BUFFER signal causes the Clock Buffer Register to be loaded with data from either the AC Register, via the DATA 0—11 lines, or from the Clock Counter Register. The B COUNT pulses are counted by the Clock Counter Register in a way that is determined by signals from the Buffer/Counter Control Signal logic. These signals are asserted by combinations of IOT signals and Mode Enable bits 1 and 2.

The OVERFLOW L signal that is generated by the Clock Counter Register is applied to the OVERFLOW flip-flop logic. This logic asserts two signals that can be used by devices (e.g., analog-to-digital converters) to initiate some operation. OVERFLOW (0) L is suitable for external devices, while EXT PULSE L is designed to be used by OMNIBUS devices (A/D START). The status of the OVERFLOW flip-flop can be checked under program control and transferred to bit 0 of the AC Register via the DATA 0 line.

The logic also provides the OVERFLOW (1) signal that is applied to the Interrupt/Skip logic. Thus, each time the Clock Counter Register overflows: internal and external devices can begin some operation; the DK8-EP can request a program Interrupt; a program instruction can be skipped; or any combination of these events can occur, depending on the state of Clock Enable bits 0, 6, and 8.

Except for transfers between the Clock Buffer Register and the AC Register, the clock operates in much the same manner when an operation is initiated by external sources. Signals applied to the external input channels can activate one or more of the Schmitt triggers. The resulting signal, SCHMITT IN 1 L (or 2, or 3) is applied to the Input/Overflow Status logic. Here the status of each input channel can be checked under program control and forwarded to the AC Register. (Event Enable bit 9 enables the status of Schmitt trigger input 1 to be transferred to AC9 via DATA 9.)

The Input/Overflow Status logic allows the external event to generate the appropriate INT RQST (1, 2, or 3) L signal. This signal is applied to the Interrupt/Skip logic and can cause a program Interrupt. The INPUT 1+2+3 L signal is also asserted by the Input/Overflow Status logic and causes the Clock Rate Select logic to produce clock pulses. The Load Buffer Register logic and the Buffer/Counter logic then operate as already outlined.

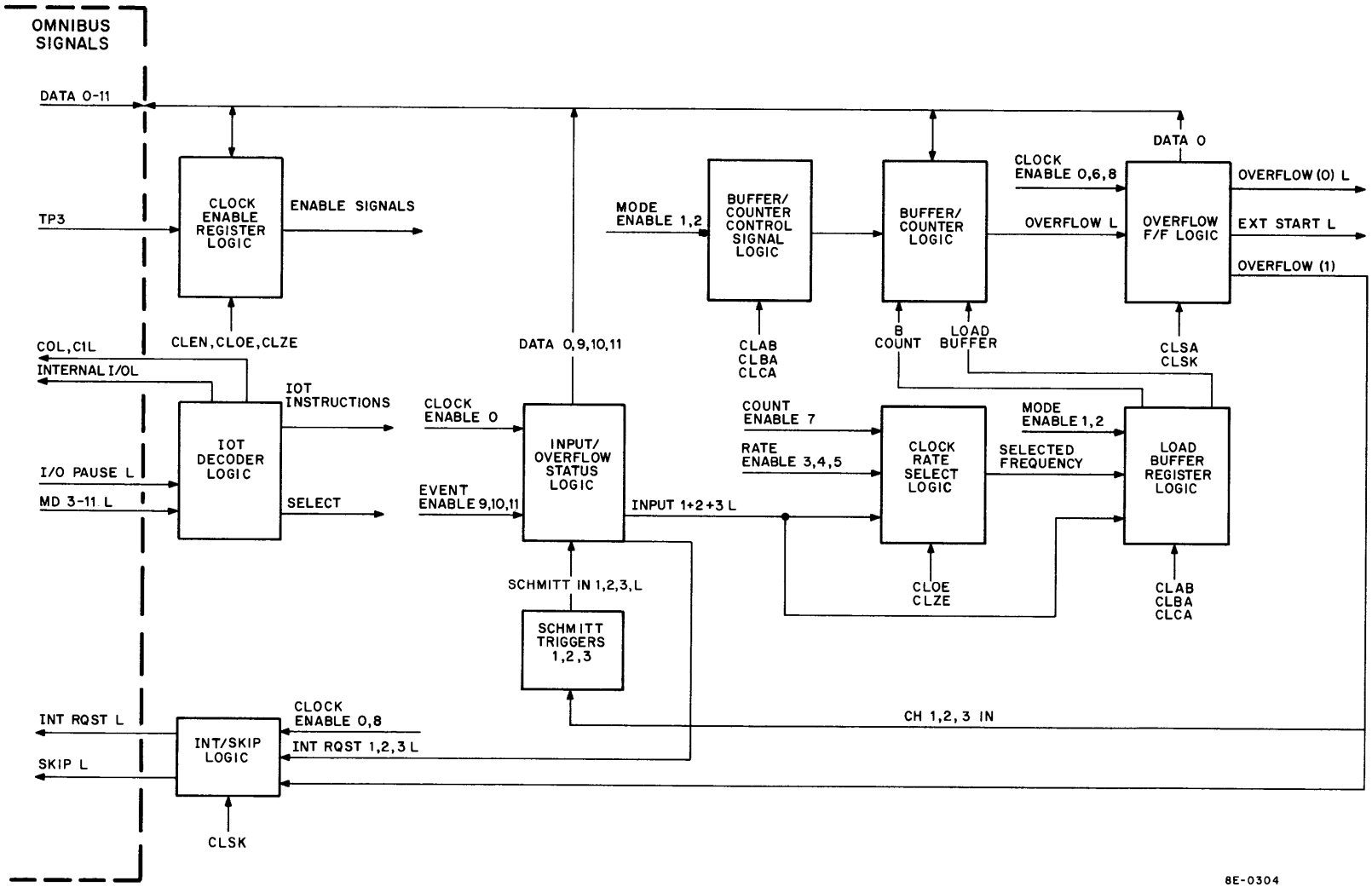


Figure 4-51 DK8-EP Functional Block Diagram

Table 4-9
DK8-EP IOT Instructions

Octal Code	Mnemonic	Function
6130	CLZE	Clear Clock Enable Register per AC Register. Each bit in the Clock Enable Register is cleared if the corresponding bit in the AC Register is set. The AC Register is unchanged.
6131	CLSK	Skip on a CLOCK flag. The next program instruction is skipped if either of the following skip conditions exists: *a.* An enabled Schmitt trigger has fired *b.* The Clock Counter Register has overflowed
6132	CLOE	Set Clock Enable Register per AC Register. Each bit in the Clock Enable Register is set if the corresponding bit in the AC Register is set. The AC Register is unchanged.
6133	CLAB	AC Register to Clock Counter Register. The contents of the AC Register are transferred to the clock and loaded into both the Clock Buffer and the Clock Counter Registers. The AC Register is unchanged.
6134	CLEN	Clock Enable Register to AC Register. The contents of the Clock Enable Register are transferred to the computer and JAMed into the AC Register. The Clock Enable Register is unchanged.
6135	CLSA	Status to AC Register. The state of the OVERFLOW flip-flop and of the three Schmitt input circuits is transferred to the computer and JAMed into the AC Register. Only the following AC bits are affected: **AC Bit — Status Condition** 0 — Enabled OVERFLOW flip-flop 9 — Enabled Schmitt input 3 10 — Enabled Schmitt input 2 11 — Enabled Schmitt input 1
6136	CLBA	Clock Buffer Register to AC Register. The contents of the Clock Buffer Register are transferred to the computer and JAMed into the AC Register. The Clock Buffer Register is unchanged.
6137	CLCA	Clock Counter Register to AC Register. The contents of the Clock Counter Register are transferred, via the Clock Buffer Register, to the computer and JAMed into the AC Register. The Clock Counter Register is unchanged.

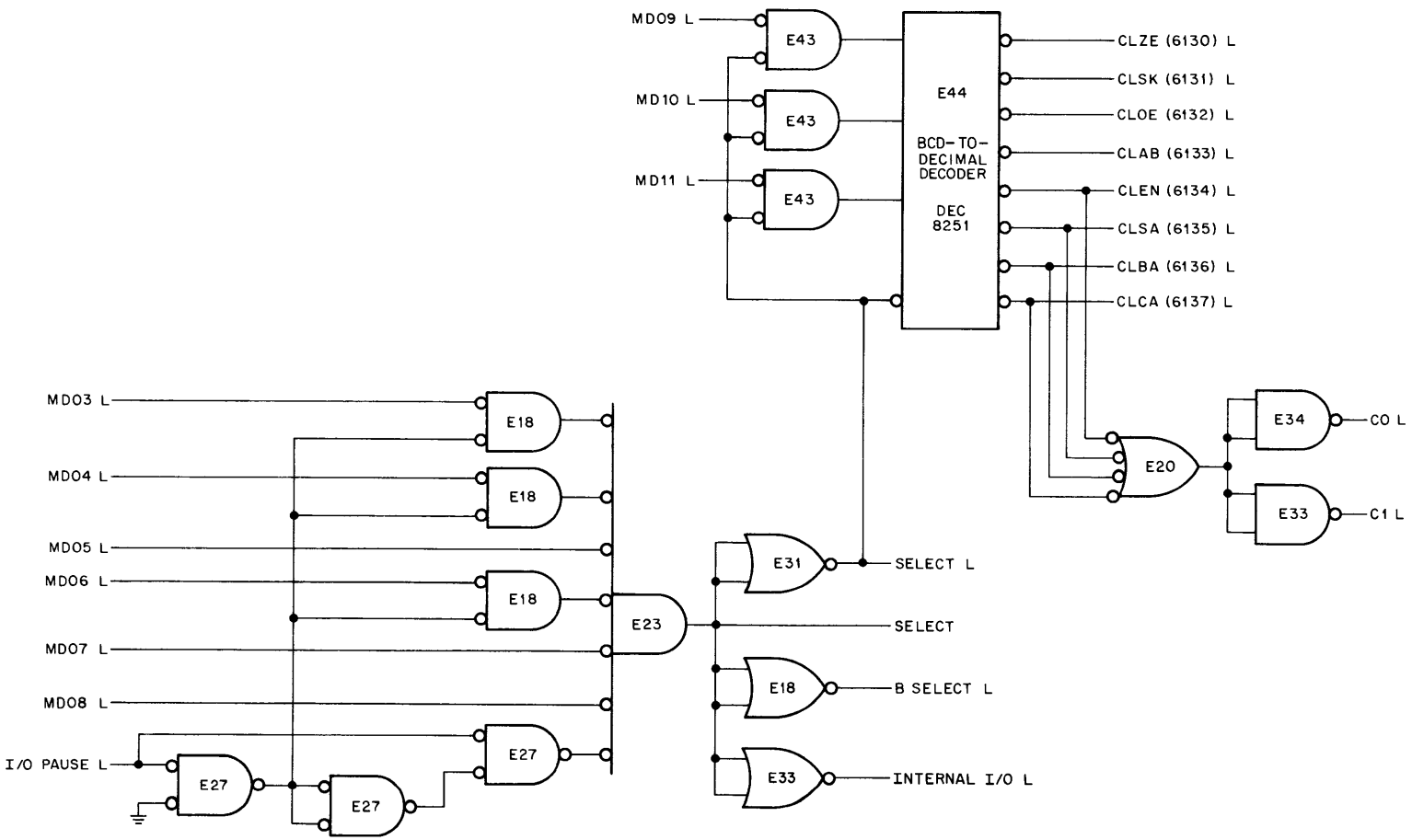


Figure 4-52 IOT Decoder Logic

Table 4-10
Clock Enable Register Enable Signals

Register Bit	Enable Signal Name	Function
0	CLOCK ENABLE 0	Enables a status check of the OVERFLOW flip-flop (CLSA), an instruction Skip on an overflow condition (CLSK), and a possible Interrupt request on an overflow condition.
1–2	MODE ENABLE 1 MODE ENABLE 2	Determine the Clock Counter Register mode of counting. The four possible modes are: 00 Register counts at the selected rate with overflow occurring every 4096 counts (see Table 4-11 for rates). 01 Register counts at the selected rate. At each overflow a preset count is loaded into the register from the Clock Buffer Register. Thus, overflow occurs every 4096 (preset) counts. 10 Register counts at the selected rate. An external event can sample the register at any time, causing the sample count to be transferred to the Clock Buffer Register. The Clock Counter Register continues counting. 11 Register counts at the selected rate. An external event can sample the register at any time. The sample count is transferred to the Clock Buffer Register and the Clock Counter Register is cleared before it resumes counting.
3–5	RATE ENABLE 3 RATE ENABLE 4 RATE ENABLE 5	Select the frequency of the internally-generated clock pulses (see Table 4-9 and Paragraph 4.6.2.3).
6	CLOCK ENABLE 6	Enables each Clock Counter Register overflow to generate the EXT PULSE L pulse that can be used by other OMNIBUS-connected devices (A/D START).
7	COUNT ENABLE 7	Inhibits clock pulses from being applied to the Clock Counter Register. (This bit can be cleared by firing an enabled Schmitt trigger.)
8	CLOCK ENABLE 8	Connects the Clock Interrupt logic to the computer interrupt system, enabling the Clock Interrupt conditions to assert the OMNIBUS INT RQST L signal.
9–11	EVENT ENABLE 9 EVENT ENABLE 10 EVENT ENABLE 11	Enable Schmitt trigger firings to turn on the clock, to cause a program interrupt, and to sample the Clock Counter Register (as set by bits 1, 2, 7, and 8).

Table 4-11
Frequencies Selected by Rate Enable Bits 3–5

Contents of Bits 3–5	Selected Multiplexer Output
000	No output
001	External frequency
010	100 Hz
011	1 kHz
100	10 kHz
101	100 kHz
110	1 MHz
111	No output

4.6.2 Detailed Logic

4.6.2.1 IOT Decoder Logic – The IOT Decoder logic is shown in Figure 4-52. The SELECT signal is asserted by NAND gate E23 when a 613X instruction is decoded. The SELECT signal, in turn, asserts the B SELECT L signal; the INTERNAL I/O L signal, which causes the positive I/O Bus interface to ignore the IOT instruction; and the SELECT L signal, which is used to gate MD bits 9–11 to the BCD-to-Decimal Decoder. The DEC 8251 IC decoder provides instructions 6130 through 6137, as illustrated in Figure 4-52.

Note that instructions 6134 through 6137 cause the OMNIBUS C0 L and C1 L signals to be asserted. Each of these instructions call for a transfer of data to the CPU AC Register. With C0 L and C1 L both asserted, the data is JAM-transferred into the AC Register, instead of being ORed with data already in the register.

4.6.2.2 Clock Enable Register Logic – The Clock Enable Register logic is shown in Figure 4-53. The register consists of 12 DEC 7474 flip-flops, only 2 of which are shown in the logic. Each flip-flop is given a descriptive title that characterizes the function of the flip-flop. For example: the MODE ENABLE 1 flip-flop is used to select, along with MODE ENABLE 2, the particular mode of operation of the Clock Counter Register; the COUNT ENABLE 7 flip-flop inhibits the Frequency Multiplexer from producing clock pulses at the selected frequency; the EVENT ENABLE 9 flip-flop enables an external event to control clock operations. (See Table 4-10 for a complete listing of the enable signals and their functions.)

Three program instructions deal exclusively with the Clock Enable Register. The CLEN (6134) instruction gates the register contents onto the DATA 0–11 lines. The information on the DATA lines is gated through the CPU Major Register and loaded into the AC Register. The CLZE (6130) and CLOE (6132) instructions clear and set, respectively, those Clock Enable Register flip-flops that correspond to set AC Register bits. For example, if bit 0 of the AC Register is logic 1, the CLOCK ENABLE 0 flip-flop can be set by the CLOE instruction or cleared by the CLZE instruction.

The Clock Enable Register can be cleared by the CLEAR signal. This signal is generated by the INITIALIZE signal that is produced when power is turned on, when the CLEAR key is depressed, or when the CAF instruction is issued.

4.6.2.3 Clock Rate Select Logic – The Clock Rate Select logic is shown in Figure 4-54. The basic 20-MHz clock frequency is provided by a crystal-controlled oscillator (see logic drawing E-CS-M518). This frequency is divided by the J-K flip-flop, E33. (When both the J and K inputs are high, the 1 output is changed with each positive transition at the C input.) The 10-MHz clock frequency is applied to a chain of DEC 7490 decade counters, each counter being wired to divide by ten. The output of each counter, which can be monitored at a test point, is applied to the E10 Frequency Multiplexer, a DEC 74151 IC. An external pulse source of any frequency can also be applied to the multiplexer, via pin RR on J1 of the M518 module. The desired frequency of clock pulses is obtained by selectively setting or clearing RATE ENABLE flip-flops 3, 4, and 5 with the CLOE or CLZE IOT instructions. Table 4-11 shows the relationship between the 1 outputs of these flip-flops and the multiplexer-output frequency.

Note that COUNT ENABLE 7 must be cleared if output pulses are to be obtained from the multiplexer. This particular Clock Enable Register flip-flop is illustrated because it differs from the remaining register flip-flops in an important way. Bit 7 can be cleared by an external event as well as by program instructions. Any of the three Schmitt-trigger input circuits can cause the INPUT 1+2+3 L signal to be asserted, thereby clearing the COUNT ENABLE 7 flip-flop and turning on the clock.

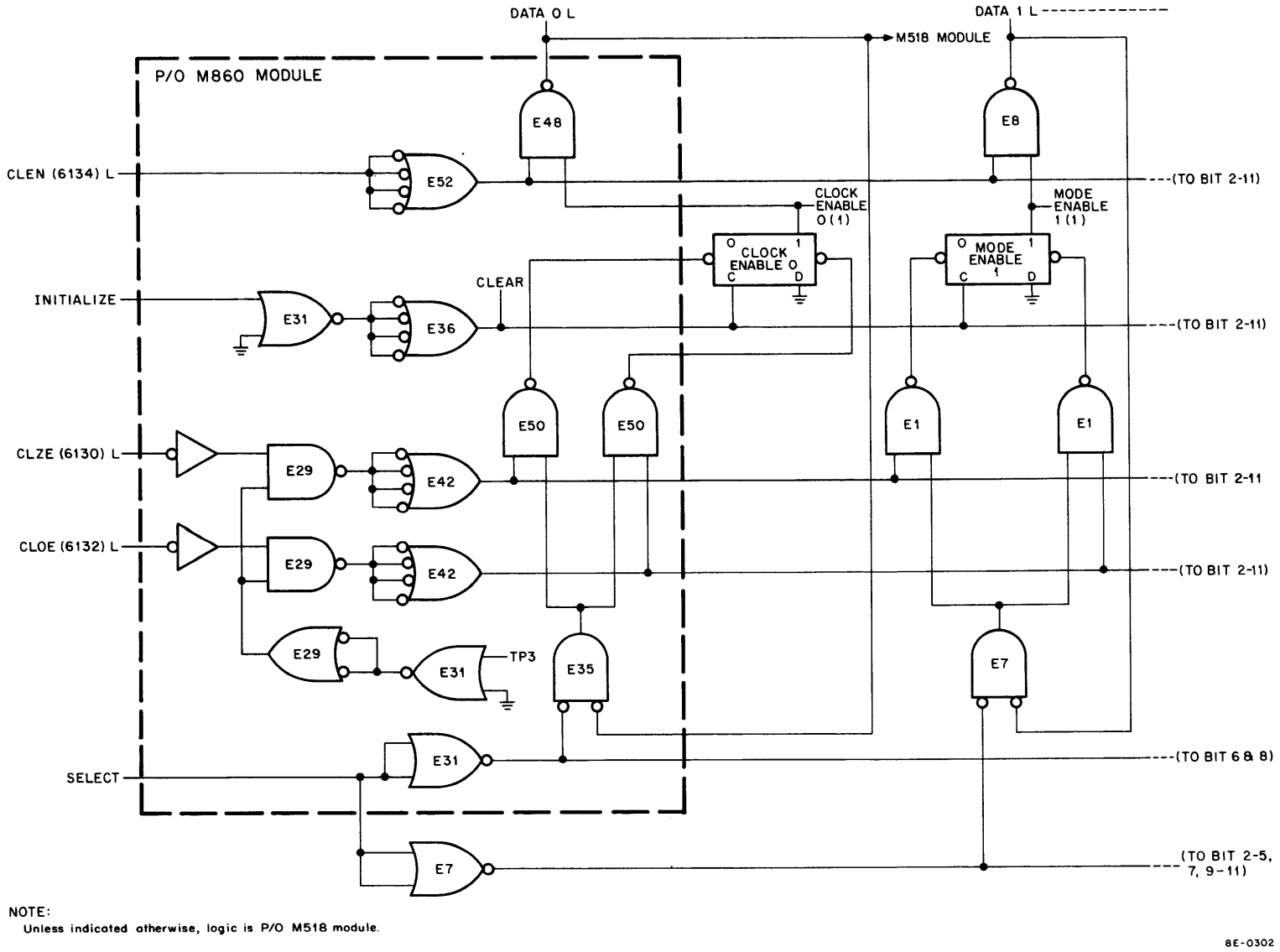
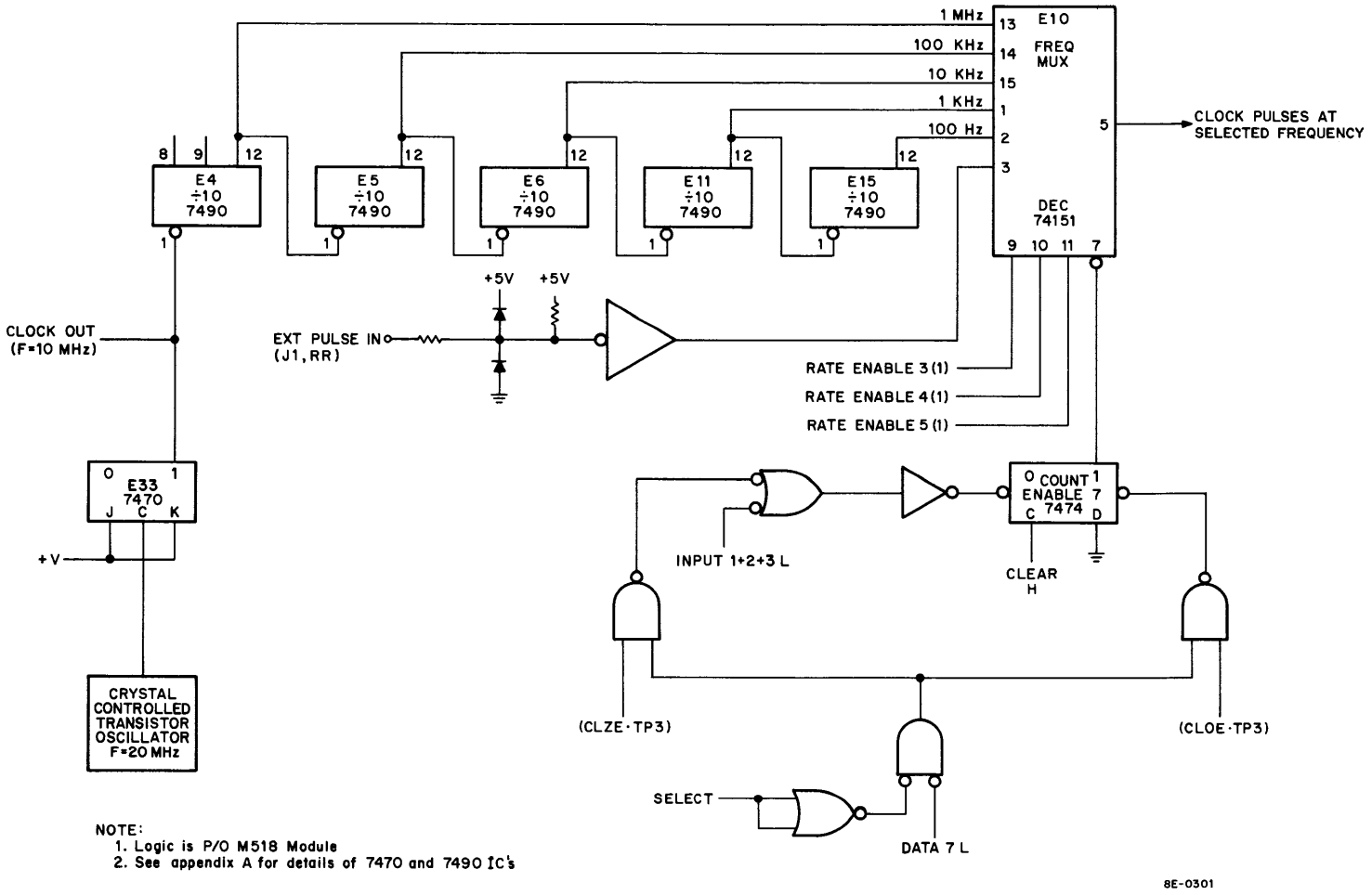


Figure 4-53 Clock Enable Register Logic



NOTE:
1. Logic is P/O M518 Module
2. See appendix A for details of 7470 and 7490 IC's

Figure 4-54 Clock Rate Select Logic

4.6.2.4 Load Buffer Register Logic – The Load Buffer Register logic is shown in Figure 4-55. The logic generates the LOAD BUFFER signal that loads the Clock Buffer Register and the B COUNT L pulses that are counted by the Clock Counter Register. The LOAD BUFFER signal can be asserted in response to either the 6133 IOT instruction, or the 6137 IOT instruction. The signal can also be asserted in response to an external event.

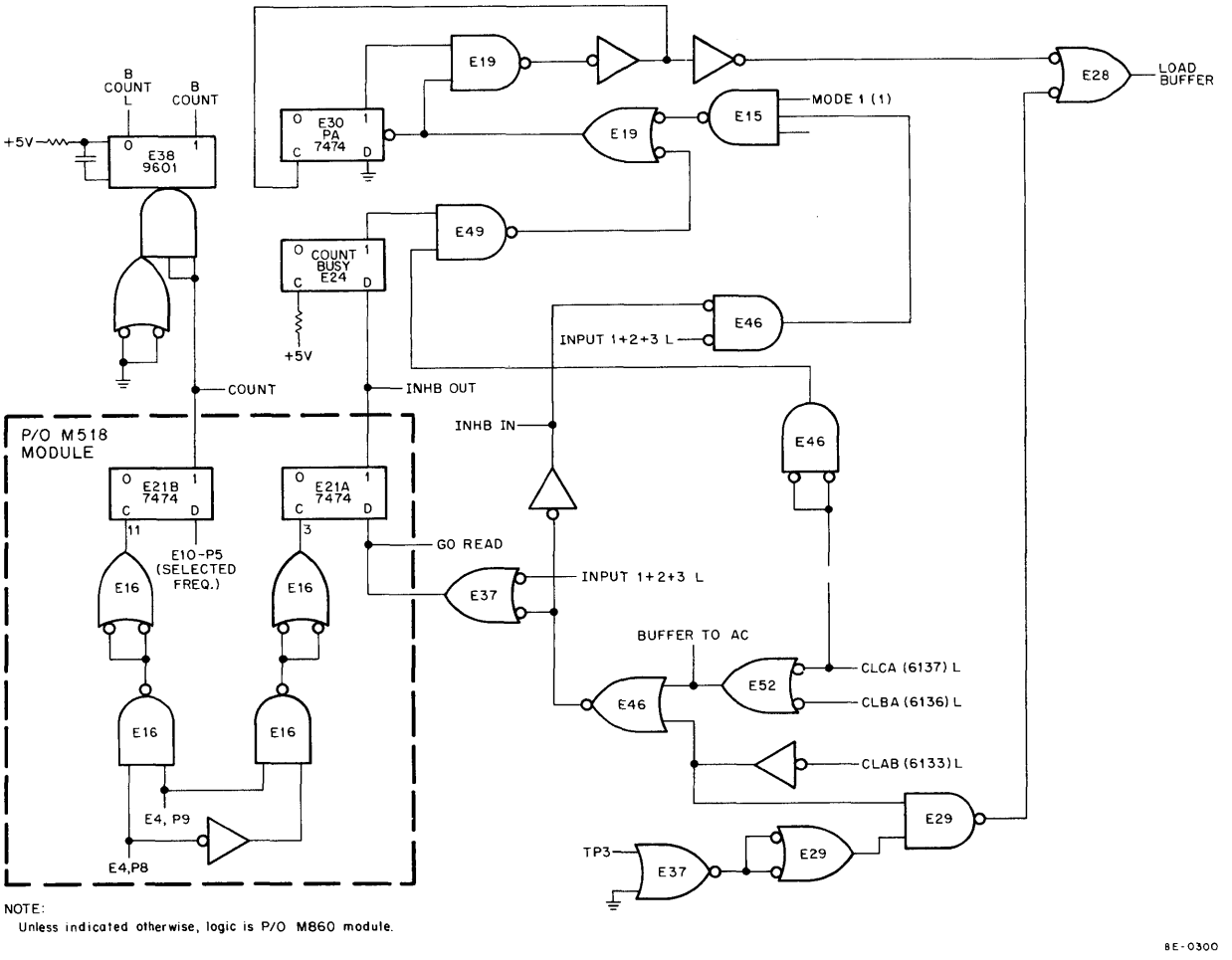


Figure 4-55 Load Buffer Register Logic

If the 6133 instruction is decoded, NAND gate E29 is enabled at TP3 time and, in turn, enables NOR gate E28 to assert the LOAD BUFFER signal. If an external event generates the INPUT 1+2+3 L signal, E28 again asserts the LOAD BUFFER signal. However, the enabling path is more complex than that of the 6133 instruction, for two reasons. First, the external event is allowed to generate the LOAD BUFFER signal only when Mode 10 or Mode 11 has been selected by Mode Enable bits 1 and 2 (NAND gate E15 can be enabled in either case). Second, an external event must not be allowed to assert the LOAD BUFFER signal when an IOT instruction is being carried out (NAND gate E46 is inhibited by the INHB IN signal during an IOT instruction). To illustrate the need for this prohibition, consider the 6136 and 6137 instructions. If the 6136 instruction is issued, the BUF TO AC signal is asserted. This signal gates the Clock Buffer Register output to the DATA lines. If the INHB IN signal were not asserted, an external event could generate the LOAD BUFFER signal during the IOT. Thus, the result would be identical to that achieved by a 6137 instruction.

The majority of the logic is designed to make the 6137 instruction operative. While the 6133 instruction loads the Clock Buffer Register with data from the CPU AC Register, the 6137 instruction loads the Clock Buffer with

the contents of the Clock Counter Register. Precautions have been taken to ensure that, if the 6137 instruction is issued, the LOAD BUFFER signal is not asserted at the moment the Clock Counter is changing its count. This prevents false counts that would result from reading the Clock Counter as one or more bits are in transition.

The precautionary logic includes flip-flop E21A and latch E24 shown in Figure 4-55. These two components and related gates generate signals that are illustrated in the timing diagram of Figure 4-56. Also shown in the timing diagram are signals generated by flip-flop E21B and one-shot E38. Both figures are referred to during the following discussion.

Flip-flops E21A and E21B are clocked by signals derived from the first decode counter in the Clock Rate Select logic (see Figure 4-54). The D input of E21B is controlled by the selected frequency output of the Frequency Multiplexer. In the timing diagram, this frequency is chosen as 1 MHz. When E21B is set, its 1 output triggers one-shot E38; E38 produces the B COUNT L signal for approximately 300 ns. The Clock Counter Register then counts the B COUNT L pulses which occur at the selected frequency.

When the 6137 instruction is issued, the CLCA L signal produced by the IOT Decoder logic asserts the GO READ signal. Because the clock is free-running, the GO READ signal can occur at any time with respect to the signal at the C input of E21A. If it occurs at the instant shown in the timing diagram, just before the pulse marked (A), E21A is set 200 ns before the B COUNT L signal is asserted. Because each bit of the Clock Counter Register consists of a master/slave flip-flop, the register bits are in transition on the leading edge of the B COUNT L pulse. Thus, the LOAD BUFFER signal causes the Clock Counter to be loaded into the Clock Buffer approximately 100 ns before the Clock Counter can change in response to the next B COUNT L pulse. (A delay of approximately 100 ns is introduced by the gating between E21A and NOR gate E28.) If the GO READ signal is asserted just after the leading edge of pulse (A), E21A is set by pulse (B), 300 ns after the B COUNT L signal is asserted. Thus, the Clock Buffer is loaded approximately 200 ns after the Clock Counter changes in response to the B COUNT pulse; enough time for the data to settle in the Clock Counter.

The amount of time between pulse (A) and pulse (B) is always 500 ns; therefore, the GO READ signal must be asserted for a longer amount of time to ensure that either (A) or (B) can set flip-flop E21A. This requirement is not met by normal CPU timing. In addition, under normal CPU timing it would be possible to lose the data that is being transferred from the Clock Counter Register. This could happen if the CLCA L signal were asserted just after the (A) pulse, for example. Under normal CPU timing for an internal IOT instruction (6137 is such an instruction), the AC LOAD L signal is asserted at TP3 time. In the present example, TP3 could occur before the count has been placed on the CPU Major Register Bus; thus, the count would be lost. Both these difficulties may be overcome by increasing the period of time during which the CLCA L signal is asserted. The logic shown in Figure 4-57 is designed to increase the available time to approximately 650 ns.

When the BUF TO AC signal is asserted, one-shot E47 is triggered and remains set for approximately 650 ns. The OMNIBUS NOT LAST TRANSFER L signal is asserted for the same amount of time. Because NOT LAST TRANSFER L is low when TP3 of the IOT instruction (6137 in the present example) occurs, the CPU timing is interrupted and stalled in TS3. Until normal timing is resumed, the OMNIBUS I/O PAUSE L signal remains low, keeping CLCA L low and, thus, GO READ high. When one-shot E47 times out, NOT LAST TRANSFER L is negated. Perhaps 40-ns later, time enough for the NOT LAST TRANSFER line to settle, latch E24 causes the OMNIBUS BUS STROBE L signal to be asserted. This signal causes normal CPU timing to begin from the point of interruption. I/O PAUSE L is negated, in turn negating the CLCA L signal and, finally, the GO READ signal.

4.6.2.5 Clock Buffer/Clock Counter Logic – The Clock Buffer/Clock Counter logic is shown in Figure 4-58. The logic for bits 1–10 is identical to that for bit 11. The Clock Buffer Register is a storage register composed of DEC7474 IC flip-flops. The Clock Counter Register is a presettable binary counter composed of DEC 74197 ICs (see Appendix A for a detailed description). Each bit can be preset at the S input and cleared at the R input.

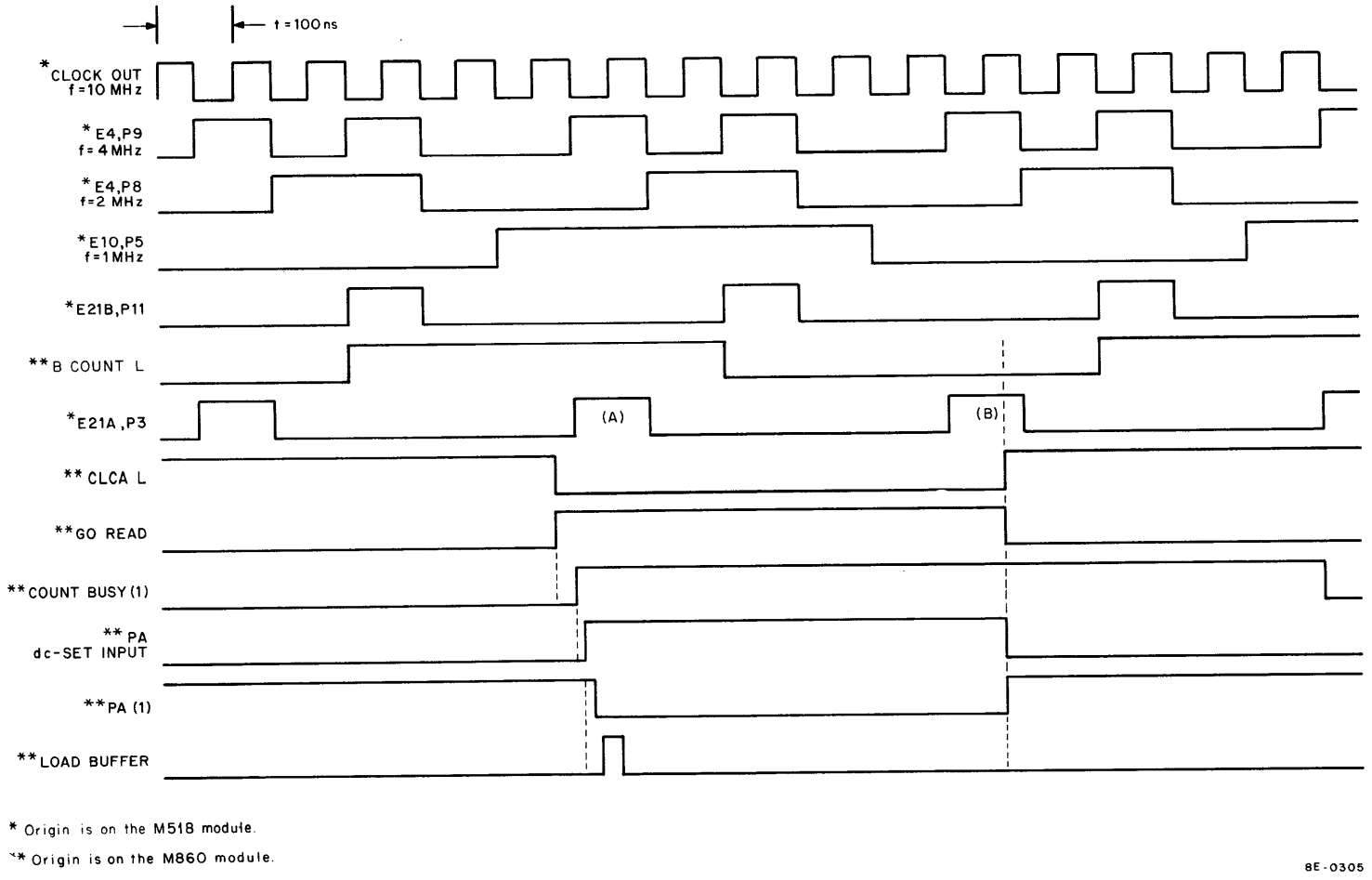


Figure 4-56 Load Buffer Register Timing

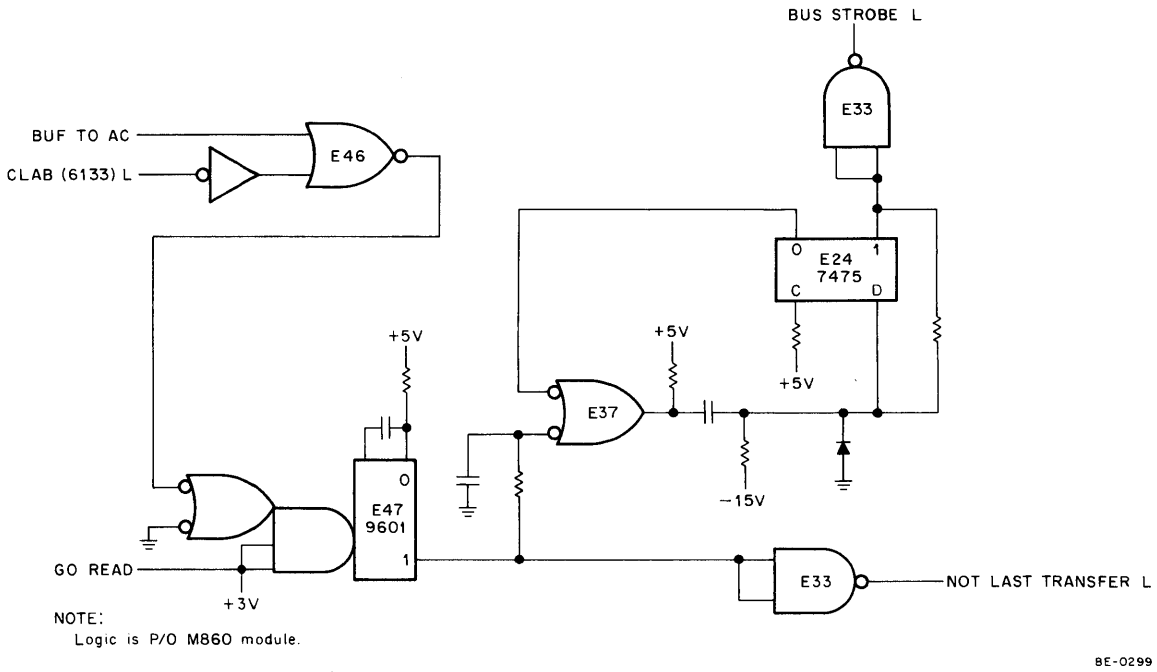


Figure 4-57 NOT LAST TRANSFER Logic

The B COUNT pulses are applied to the bit 11 toggle (T) input; the negative transition of each B COUNT L pulse causes the 1 output to change. The 1 output of each bit is connected to the T input of the following bit and the Buffer Multiplexer, a DEC 8266 IC.

The logic enables the exchange of data between the Clock Buffer Register and the Clock Counter Register, and between these two registers and the CPU AC Register. Data transfers between the Clock Buffer and the Clock Counter can be controlled by external events; however, transfers between these two registers and the AC can be accomplished only under program control. The first type of transfer, between Clock Buffer and Clock Counter, can be considered an internal transfer (internal to the clock), while the second can be considered as external.

Three different transfers can be grouped under each type, as shown in column 1 of Table 4-12. Column 2 of this table shows both the source register and the destination register of each transfer. Column 3 shows the control signals that must be asserted for each transfer. For example, if the DK8-EP is operating in mode 10, an external event can cause a transfer from the Clock Counter to the Clock Buffer by asserting the CTR ENABLE and the LOAD BUFFER signals. The output of the Clock Counter Register is applied to the Buffer Multiplexer. Because CTR ENABLE is asserted (this signal is asserted for all transfers except CLAB), the Clock Counter data is gated through the multiplexer to the Clock Buffer, which is then loaded by the LOAD BUFFER signal. Note that Mode 11 transfers are similar, but that the CLR CTR L signal clears the Clock Counter after data is transferred to the Clock Buffer.

The data placed in the Clock Buffer by any of these internal transfers can be passed to the AC only by an external-type transfer. The CLBA instruction asserts the BUF TO AC signal that gates the Clock Buffer output onto the OMNIBUS DATA lines. The information on the DATA lines is then gated to the AC Register and loaded at BUS STROBE L time.

Data can be transferred from the AC to the Clock Counter by the CLAB instruction. In this case, information in the AC Register is placed on the DATA lines and gated to the Buffer Multiplexer (B SELECT L is asserted by the IOT Decoder logic). The AC ENABLE signal is asserted, gating the data to the Clock Buffer. The LOAD BUFFER signal loads the register and the BUF TO CLR L signal gates the Clock Buffer output to the S input of the Clock Counter.

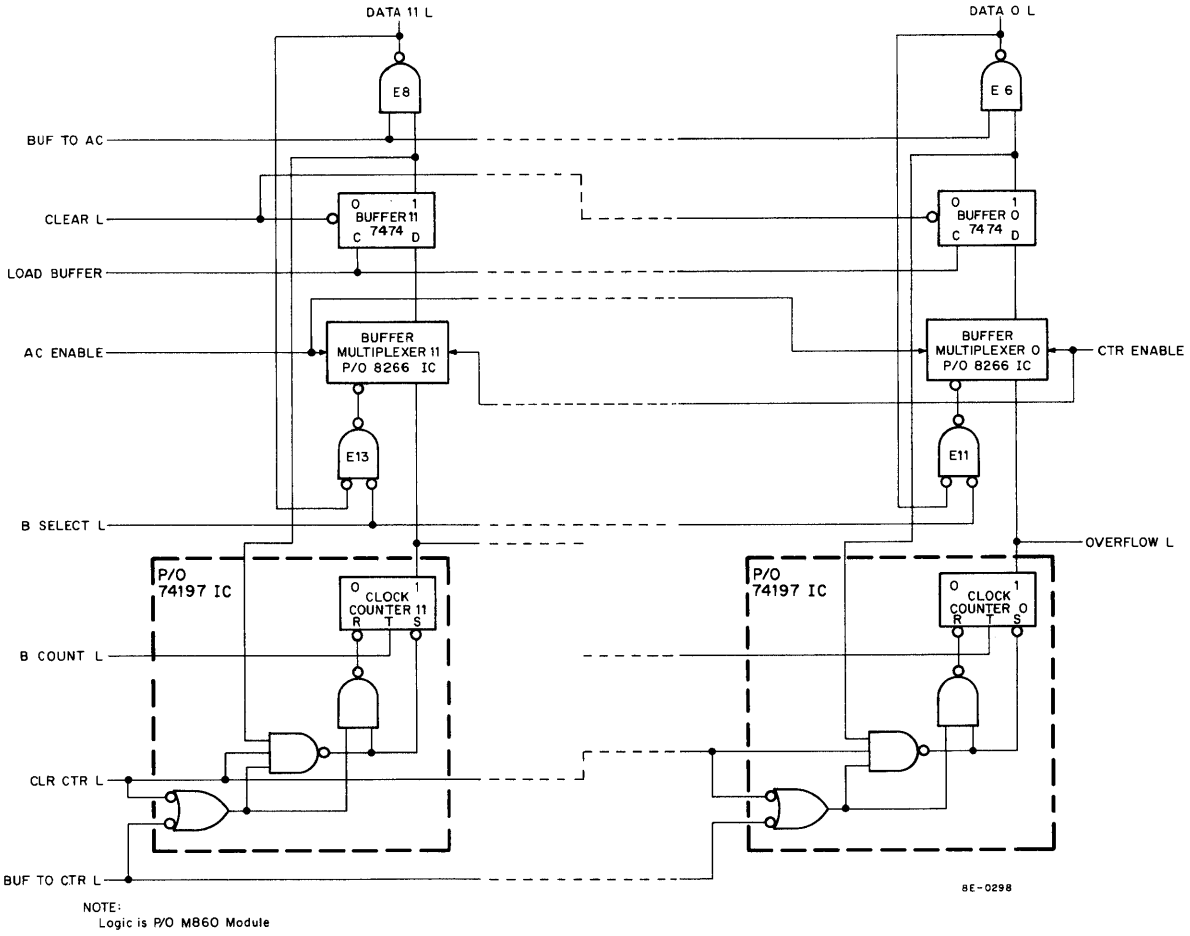


Figure 4-58 Clock Buffer/Clock Counter Logic

Table 4-12
Control Signals for Clock External and Internal Transfers

Transfer Type	From/To	Control Signal Asserted
Internal		
Mode 01	Clock Buffer/Clock Counter	BUF TO CTR L, CTR ENABLE
Mode 10	Clock Counter/Clock Buffer	CTR ENABLE, LOAD BUFFER
Mode 11	Clock Counter/Clock Buffer	CTR ENABLE, LOAD BUFFER, CLR CTR L
External		
CLAB	AC Register/Clock Counter	AC ENABLE, LOAD BUFFER, BUF TO CTR L
CLBA	Clock Buffer/AC Register	BUF TO AC, CTR ENABLE
CLCA	Clock Counter/AC Register	CTR ENABLE, LOAD BUFFER, BUF TO AC

The control signals are asserted by the logic illustrated in Figure 4-59. BUF TO AC and AC ENABLE are used only during external transfers and, thus, are asserted only by program instructions. (Note that the AC ENABLE and CTR ENABLE signals are mutually exclusive.) However, because the Clock Buffer is loaded during both

external and internal transfers, external events, as well as program instructions, can generate the LOAD BUFFER signal. The BUF TO CTR L signal is also used for both internal and external transfers (the Mode 01 transfer can be accomplished only when the MSB of the Clock Counter Register asserts the OVERFLOW L signal).

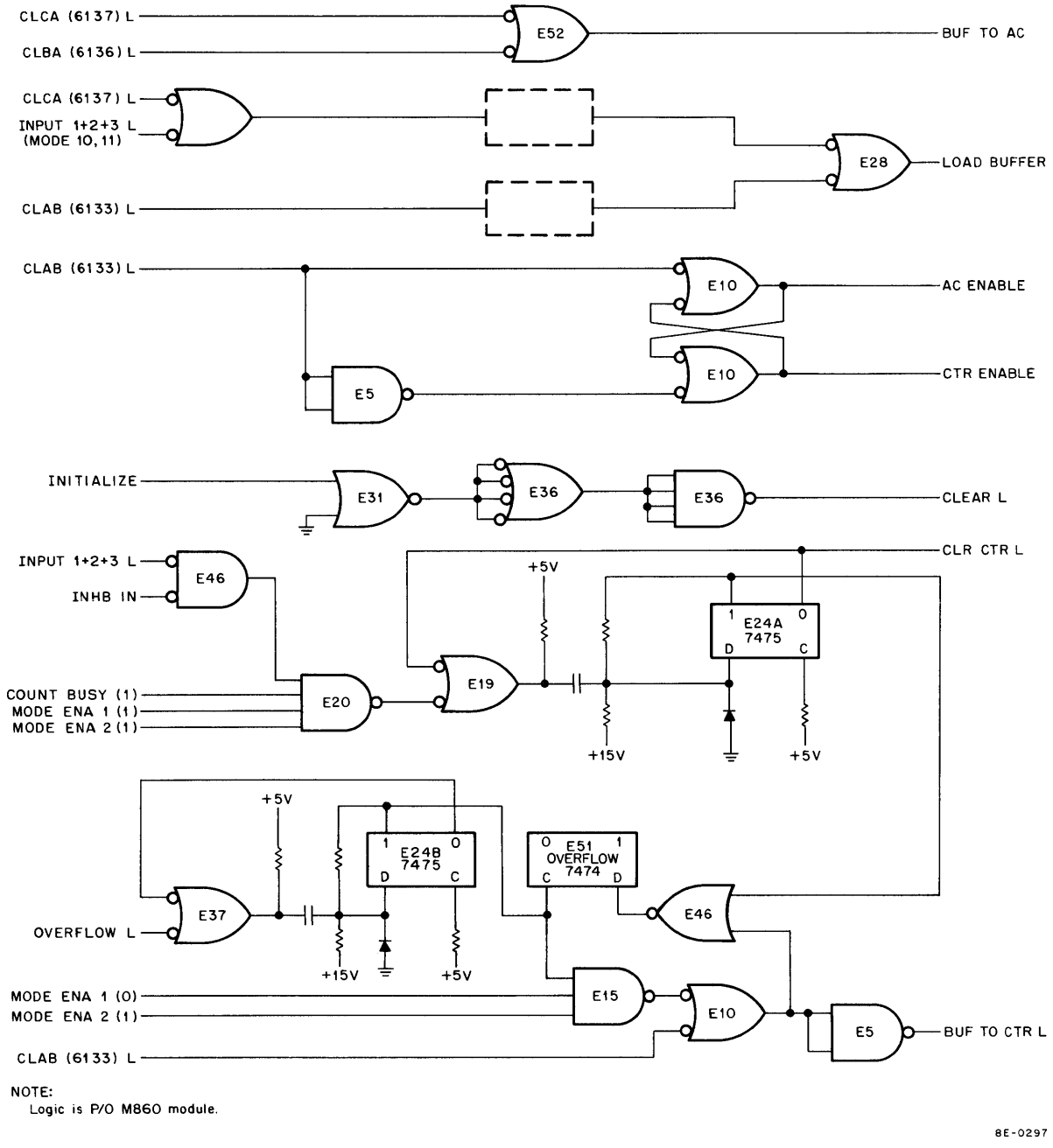


Figure 4-59 Clock Buffer/Clock Counter Control Signals

The CLR CTR L signal is asserted only when a Mode 11 transfer is initiated by an external event. Bi-stable latch E24A (one-shot) asserts the CLR CTR L signal. Note that the 1 output of the latch is applied to the E51 OVERFLOW flip-flop via NOR gate E46. The MSB of the Clock Counter might go from a high to a low when CLR CTR L is asserted. Such a transition asserts the OVERFLOW L signal, thereby enabling NOR gate E37. The 1 output of latch E24B goes high (one-shot). The OVERFLOW flip-flop would be set unintentionally if E46 was

not enabled. The same problem could arise when the CLAB instruction is issued; thus, NOR gate E10 is also applied to E46.

Although the CLR CTR L signal can be asserted only by an external event, the Clock Counter can be cleared under program control. The Clock Buffer must first be cleared. This can be done either by issuing a CAF (Clear All Flags) instruction, or by depressing the CLEAR key on the programmer's console. The INITIALIZE signal is asserted by either method (and by power turn-on). The CLEAR L signal then clears the Clock Buffer Register and the contents of the Clock Buffer can be transferred to the Clock Counter.

4.6.2.6 OVERFLOW Flip-Flop Logic – The OVERFLOW flip-flop logic is shown in Figure 4-60. The logic monitors the MSB of the Clock Counter Register via the OVERFLOW L signal. When the register asserts the OVERFLOW L signal, the OVERFLOW flip-flop, E51, is normally set. (NOR gate E46 prevents the flip-flop from being set unintentionally; see Paragraph 4.6.2.5.) The set state of the flip-flop enables the logic to assert various signals selectively. The selectivity depends on the state of certain Clock Enable Register flip-flops and on the particular program instruction, if any, that is issued.

The OVERFLOW (0) L signal and the EXT PULSE L signal have similar purposes; i.e., to initiate some operation, such as analog-to-digital conversion, when overflow of the Clock Counter Register occurs. When OVERFLOW L is asserted, latch E24 enables NAND gate E5, provided that the CLOCK ENABLE 6 flip-flop was set by a previous CLOE instruction, and sets the E51 OVERFLOW flip-flop. The 1 output of E51 enables NAND gate E19, thereby asserting the EXT PULSE L signal. E24 remains latched for approximately 100 ns. (The duration of the latch is determined by the RC time constant of the D input; see Appendix A for details.) Thus, the EXT PULSE L signal is a pulse that can be generated each time an overflow occurs.

The EXT PULSE L pulse is designed to be used with an A/D converter that plugs into the OMNIBUS. (The signal can be taken from the M860 module, conveniently, only via an H851 Edge Connector.) In such an application, the EXT PULSE L pulse sets the rate at which an analog signal is sampled by the A/D converter.

The 0 output of E51 is buffered before it asserts the OVERFLOW (0) L signal. This signal, once asserted, remains so until E51 is cleared under program control. The OVERFLOW (0) L signal is made available at a 40-pin connector on the M518 module. Thus, it can be used by both external and internal devices in a variety of applications.

No matter what the application, the OVERFLOW (0) L signal is useful, generally, only when it can be continuously asserted. Therefore, flip-flop E51 must be repeatedly cleared. The flip-flop can be cleared, not only by the INITIALIZE signal (generated at power turn-on, by the CLEAR key, or by the CAF instruction), but also by the 6135 instruction, STATUS TO AC. The 6135 instruction can clear E51 only if the CLOCK ENABLE 0 flip-flop has been previously set by a CLOE instruction. When E51 is set, NAND gate E50 is enabled and it, in turn, enables NOR gate E49. When CLSA is issued, the STATUS TO AC signal enables NAND gate E48, asserting the DATA 0 L signal. At TP3 time, E51 is cleared and can be set by the next Clock Counter Register overflow.

The status of E51 can be checked either by the CLSK instruction alone, or via the Interrupt system and the CLSK instruction. The Interrupt system can be used only if the CLOCK ENABLE 8 flip-flop has been set. E51 can then assert the INT RQST L signal. The CLSK instruction in the Interrupt-servicing routine will cause a Skip to the DK8-EP subroutine, where CLSA causes the status of E51 to be transferred via the DATA 0 line to AC0.

4.6.2.7 Interrupt/Skip Logic – The Interrupt/Skip logic is shown in Figure 4-61. As indicated in previous discussions, program Interrupts can be caused by external events and by overflow from the Clock Counter Register. The DK8-EP can be logically connected to the computer Interrupt system if the CLOCK ENABLE 8 flip-flop is set by a CLOE instruction. When this flip-flop alone is set, an external event can cause a program Interrupt. The event causes the INT REQ 1, 2, or 3 L signal to be generated by the Input/Overflow Status logic. This signal first enables NOR gate E15, and finally causes NAND gate E34 to assert the OMNIBUS INT RQST L signal.

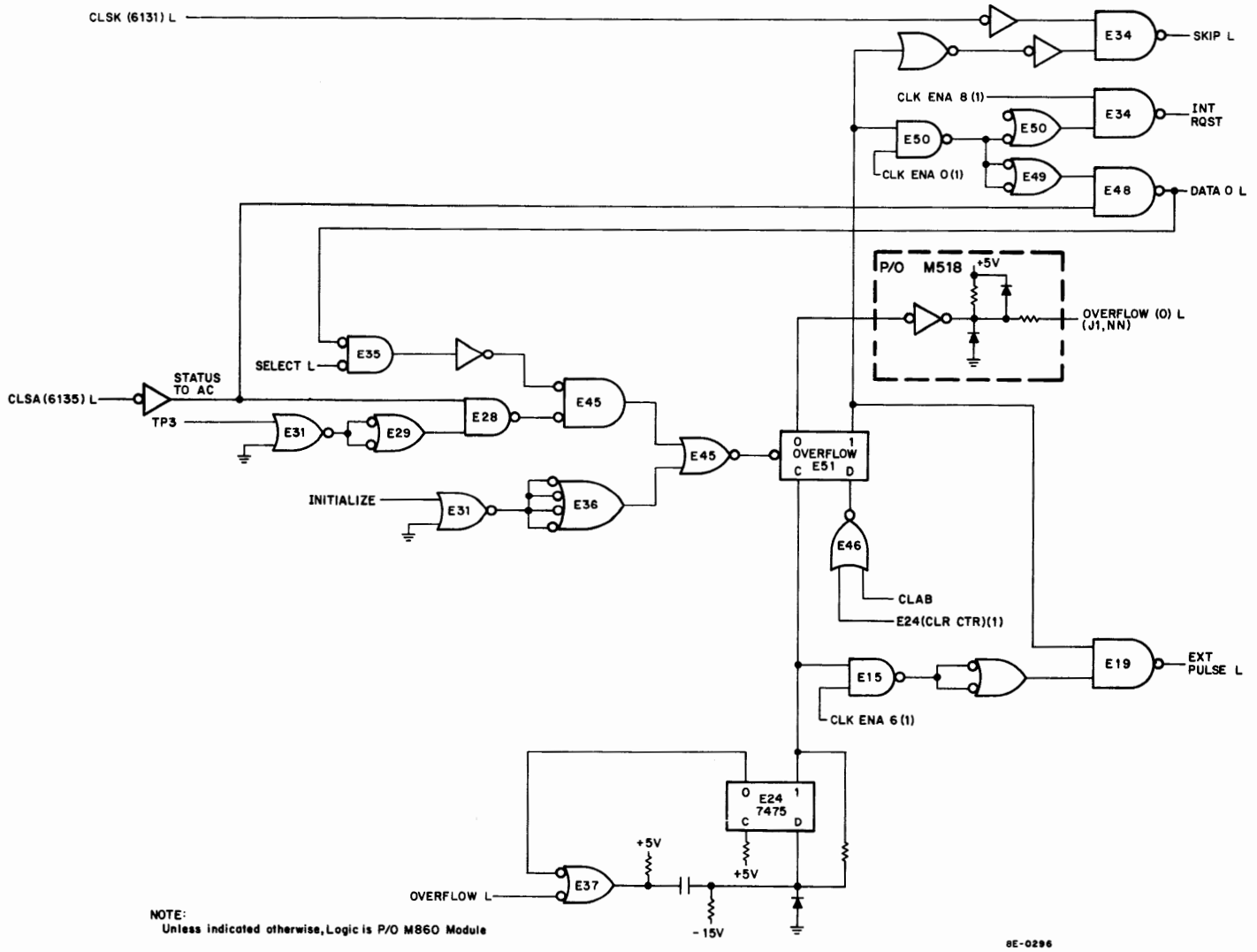


Figure 4-60 OVERFLOW Flip-Flop Logic

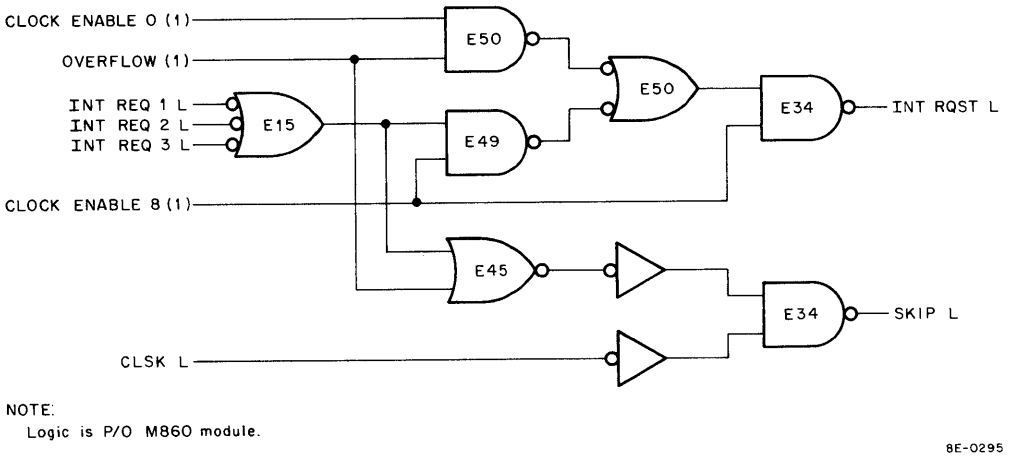


Figure 4-61 Interrupt/Skip Logic

If the OVERFLOW (1) signal from the OVERFLOW flip-flop logic is to generate an Interrupt request, the CLOCK ENABLE 0 flip-flop must also be set. This signal can then cause the INT RQST L signal to be asserted by NAND gate E34.

When the computer enters the Interrupt-servicing routine in response to the DK8-EP Interrupt request, the CLSK instruction causes NAND gate E34 to assert the OMNIBUS SKIP L signal. Thus, the computer can be directed to the particular subroutine that services the request. The CLSK instruction may still be used when the clock is not connected to the computer Interrupt system. The program can enter a waiting loop which checks the status of the two conditions (overflow and external event). When either condition is met, the appropriate subroutine is entered.

4.6.2.8 Input/Overflow Status Logic – The Input/Overflow Status logic, shown in Figure 4-62, enables the programmer to check the status of the OVERFLOW flip-flop and the Schmitt trigger input channels. Input channel 3 is illustrated fully in Figure 4-62; the logic functions are as follows. An external event can cause the SCHMITT IN 3 L signal to be asserted. If the EVENT ENABLE 09 flip-flop has been set by a previous CLOE IOT instruction, the SCHMITT IN 3 L signal sets the STATUS 3 flip-flop and asserts the INPUT 1+2+3 L signal. The latter signal can either actuate the clock or cause the contents of the Clock Counter Register to be transferred to the Clock Buffer Register. When the STATUS 3 flip-flop is set, it asserts the INT RQST 3 L signal. This signal can result in a program Interrupt request, if the CLOCK ENABLE 8 flip-flop has been set by some previous CLOE instruction.

Assume that the external event causes a program Interrupt. The OMNIBUS INT RQST L signal is asserted and the CPU enters the Interrupt-servicing routine. The CLSK instruction in the servicing routine causes the program to proceed to the DK8-EP subroutine. The CLSA IOT instruction determines how often the event occurs in a given amount of time and/or which input channel caused the program Interrupt. When this instruction is issued, the Select logic generates the CLSA L signal that, in turn, generates the STATUS TO AC signal (see Figure 4-62). The leading edge of the STATUS TO AC signal sets the SYNC 3 flip-flop (the D input of the flip-flop is high because the STATUS 3 flip-flop is set). NAND gate E3 and NAND gate E8 are enabled, the latter gate asserting the DATA 09 L signal. The CLSA L signal causes the OMNIBUS C0 L and C1 L signals to be asserted, resulting in a JAM-transfer to the AC Register of the information on the DATA 09 line.

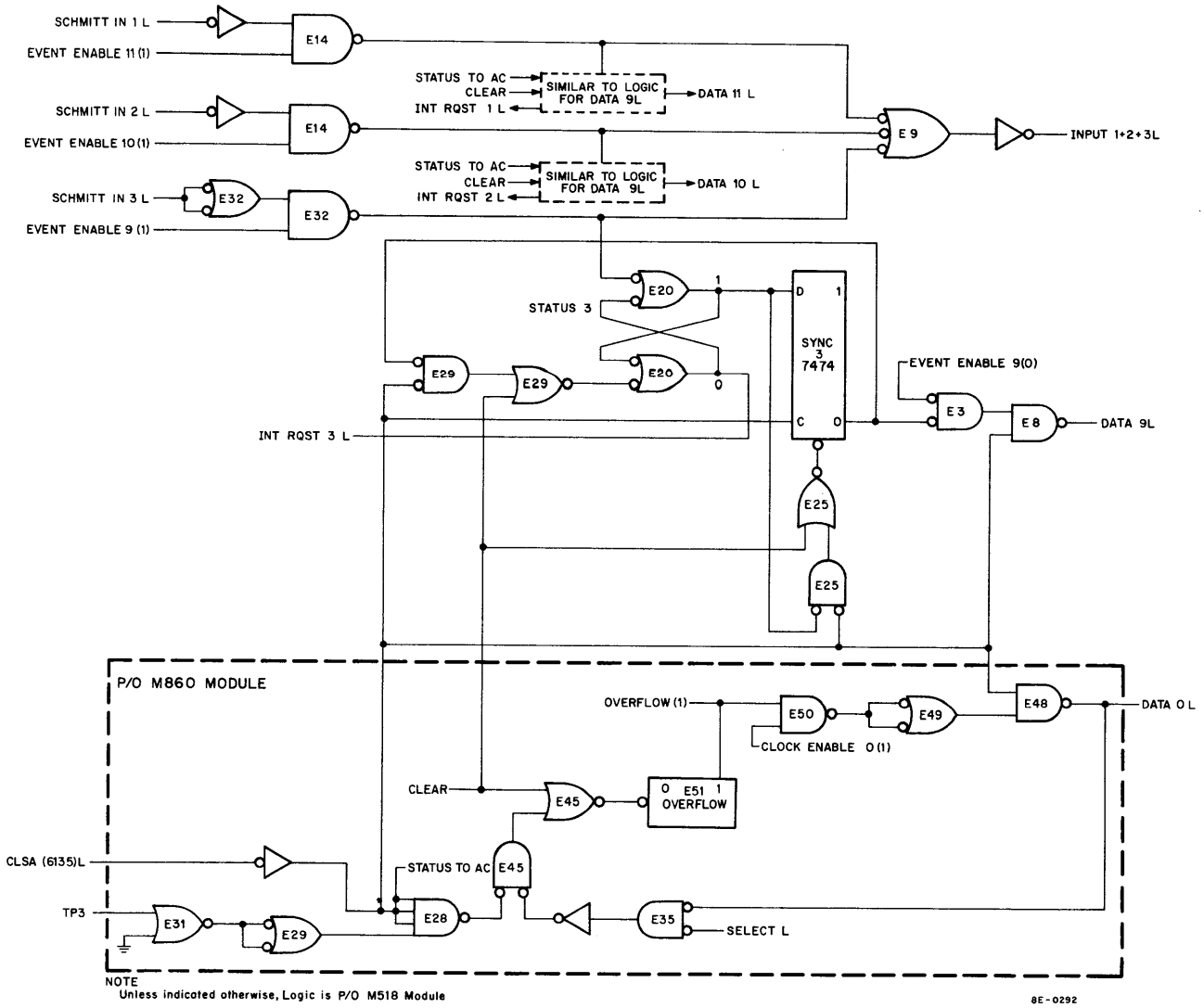


Figure 4-62 Input/Overflow Status Logic

At approximately the same time that the DATA 09 L signal is asserted, the STATUS 3 flip-flop is cleared via NAND gate E29 (the SYNC 3 flip-flop 0 output enables E29). Approximately 500 ns later (a function of the CPU Timing Generator), the CLSA L signal is negated. The SYNC 3 flip-flop is then cleared via NAND gate E25. The two flip-flops are cleared so that only one occurrence of an event is transferred for each interrogation, and so that an event is indicated only when the event actually does occur.

The OVERFLOW flip-flop also can cause a program Interrupt (see Figure 4-61 for the Interrupt logic). The CLSA instruction allows the programmer to differentiate between external-event-generated Interrupts and overflow-generated Interrupts. For example, if the OVERFLOW flip-flop is set, and if the CLOCK ENABLE 0 flip-flop has been set by a CLOE instruction, the CLSA L signal causes the DATA 0 L signal to be asserted. At the same time, the C0 L and C1 L signals are asserted and the data is JAMed into the AC Register. At TP3 time, the OVERFLOW flip-flop is cleared via NAND gate E45; the flip-flop can now be set by a new overflow from the Clock Counter Register.

4.6.2.9 Schmitt Trigger Logic – The Schmitt trigger logic for channel 1 is shown in Figure 4-63. The logic for channels 2 and 3 is identical to that shown for channel 1. The logic consists of a Schmitt trigger circuit, the components to the left of inverter E41, and the 2-output pulse-shaping circuits to the right of E41.

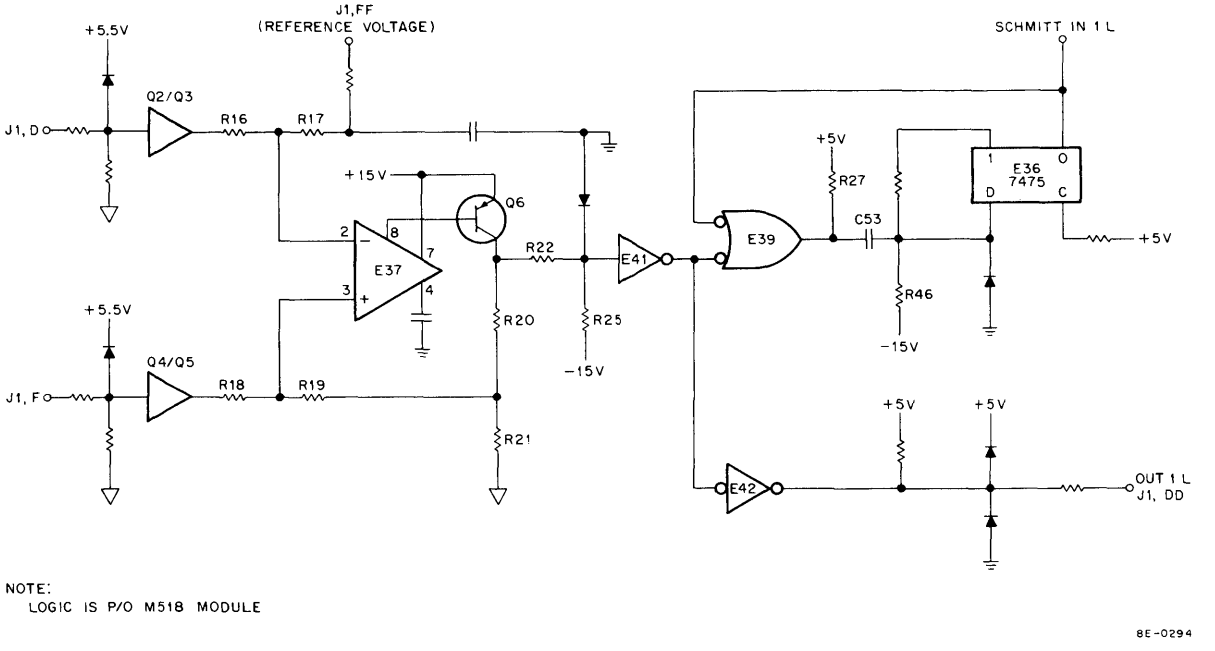


Figure 4-63 Schmitt Trigger Logic

The Schmitt trigger circuit is built around a DEC 1709C IC, E37 (see Appendix A for details). This IC is an operational amplifier used as a comparator in the Schmitt trigger. In this application, only half of the IC internal circuitry is being used; the output of E37 is taken from pin 8 rather than pin 6. The output at pin 8 has the same relationship to the input at pins 2 and 3, as does the output at pin 6; i.e., pin 8 inverts the input at pin 2, but not the input at pin 3. For clarity, many of the components comprising the Schmitt trigger inputs have been represented by the logic inverters designated Q2/Q3 and Q4/Q5 (see logic drawing E-CS-M518-0-1 for the actual circuits). The differential input to the Schmitt trigger is applied between J1D and J1F. The firing threshold voltage, which can be varied between ±5V, is applied at J1F. The hysteresis voltage, 0.3V, is determined by the value of resistors R20, R21, and R22.

Assume a threshold voltage of +4V. Until this voltage is exceeded by the differential input, the non-inverting input of E37 (pin 3) is positive with respect to the inverting input (pin 2), and pin 8 is at approximately +15V. Thus, transistor Q6 is in the non-conducting state and the input to inverter E41 is near ground. Both the SCHMITT IN 1 L and the OUT 1 L signals are negated. When the differential input crosses the threshold, going positive, the inverting input goes positive with respect to the non-inverting input. The voltage at pin 8 drops to near ground and Q6 switches on. The positive-going edge at the input of E41 triggers the latch circuit and E36 generates a pulse (duration determined by C53/R46 time constant), the SCHMITT IN 1 L signal, which is applied to the Input/Overflow Status logic. The OUT 1 L signal goes low when E41 is enabled, remaining low until the Schmitt trigger is reset. The trigger remains in the fired state until the differential input falls below +3.7V (threshold voltage minus hysteresis voltage). When this occurs, Q6 is turned off and remains off until the threshold is again exceeded.

The Schmitt trigger logic is part of the M518 module, Input logic, and Schmitt triggers. The DK8-EP user must provide cabling for the Schmitt trigger differential inputs and outputs, as well as for ground connections and threshold voltages; without these signals, the Schmitt triggers are inoperable. The connections to the M518 module are made via J1 on the module.

The DK8-ES user is provided with a DK8-EF front panel that not only simplifies the interconnection of signal sources and Schmitt triggers, but also includes potentiometers that allow the user to control the threshold voltages. The circuit schematic of this front panel is illustrated, in part, in Figure 4-64. The schematic is shown for channel 1 only; the circuits for channels 2 and 3 are identical.

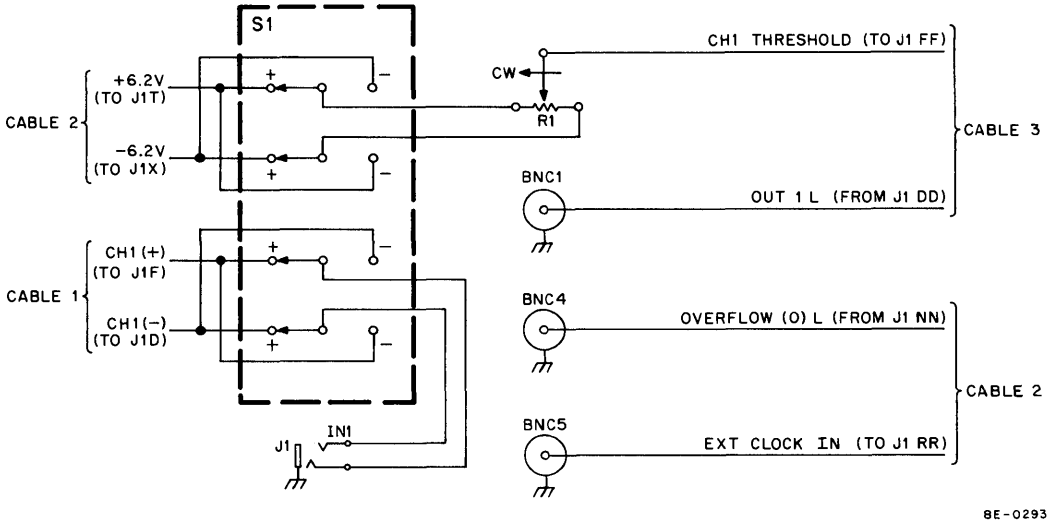


Figure 4-64 DK8-EF Front Panel Schematic

The source signal, designated IN1, is applied at phone jack J1 on the front of the DK8-EF. The SLOPE switch allows the user to select a positive- or negative-going slope on which to fire the Schmitt trigger. Note that a positive slope is accompanied by a positive threshold voltage; when a negative slope is selected, the threshold voltage is switched negative. The input signal is switched through S1 and connected by cable 1 to J1 on the M518 module. The reference voltage is taken from the wiper arm of R1 and connected by cable 3 to J1. The OUT 1 L signal from the M518 module is also connected by cable 3 to the BNC1 connector on the front panel. Channels 2 and 3 are wired in similar fashion. Note that cables 1, 2, and 3 are wired directly to the BNC connectors, the switches, and the potentiometers, with no intervening jacks or connectors. The three cables are joined in one 40-pin Berg connector that mates with J1.

Also note that two signals, not directly associated with the Schmitt triggers, are accessible at the front panel connectors. One of these signals is the OVERFLOW (0) L signal from the OVERFLOW flip-flop logic. The other is the EXT CLOCK IN signal (input) which is applied to the Clock Rate Select logic.