SECTION 8 - INPUT/OUTPUT SECTION

8.1. GENERAL DESCRIPTION

8.1-1. OBJECTIVES

To present the general description of the I/O Section.

8.1-2. INTRODUCTION

The I/O section is responsible for all communication in the form of control signals and word transfers with peripheral equipment.

8.1-3. REFERENCES

UNIVAC 1219 Technical Manual, Volume I, Paragraphs 1-5d, 3-6c(2), and 4-3a through 4-3e.

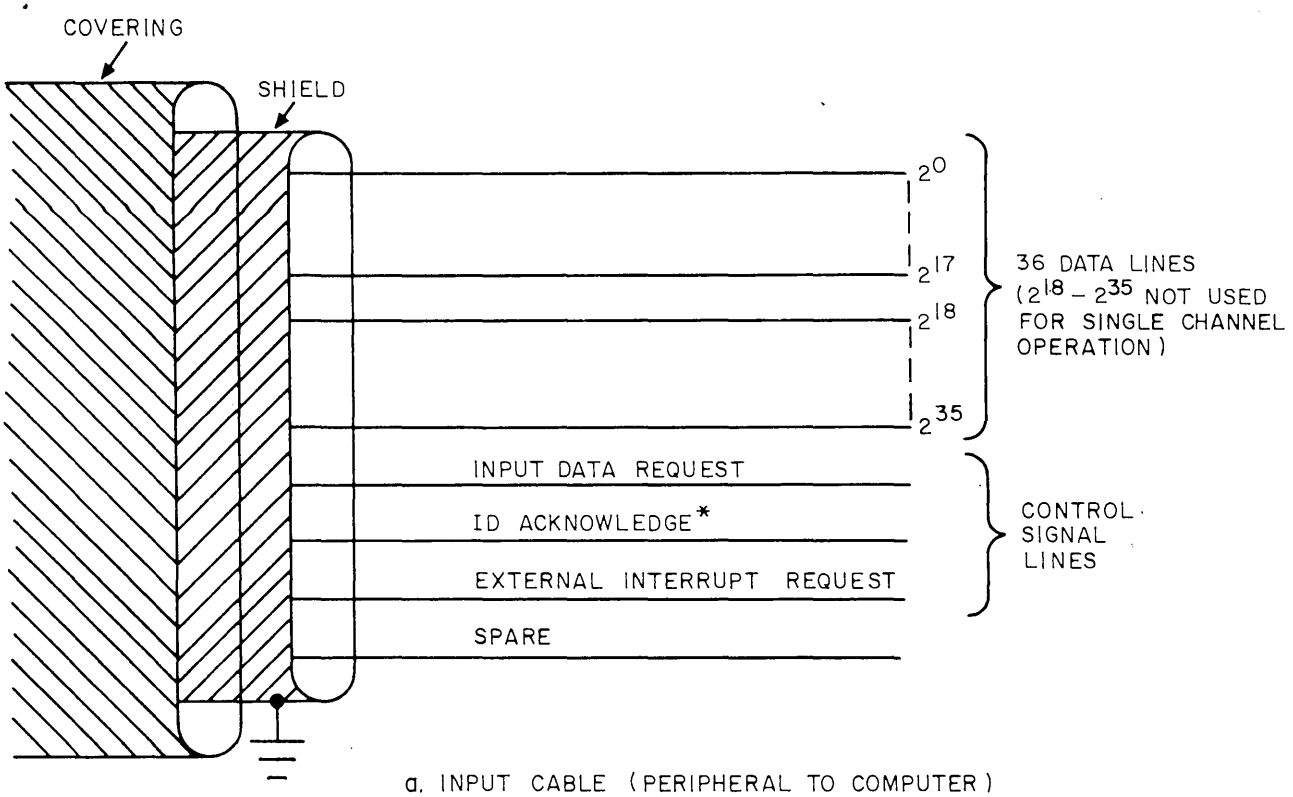
8.1-4. INFORMATION

a. Channels, I/O Cable, and Chassis Assignment. The 1219 Computer is able to communicate with a maximum of 16 peripheral devices. Each communication path is referred to as a channel. The channels are numbered $0\text{-}17_8$. Each data word transfer involves a maximum of 18 bits or 36 bits in parallel for single or duel channel operation, respectively. Each channel has an associated input cable and output cable. The shielded cables consist of twisted pairs of data or signal lines and their individual ground return lines. Refer to figure 8.1-1 and UNIVAC 1219 Technical Manual, Volume I, Section 2, table 2-2 for the cable description. The I/O control signals are discussed later in this sheet.

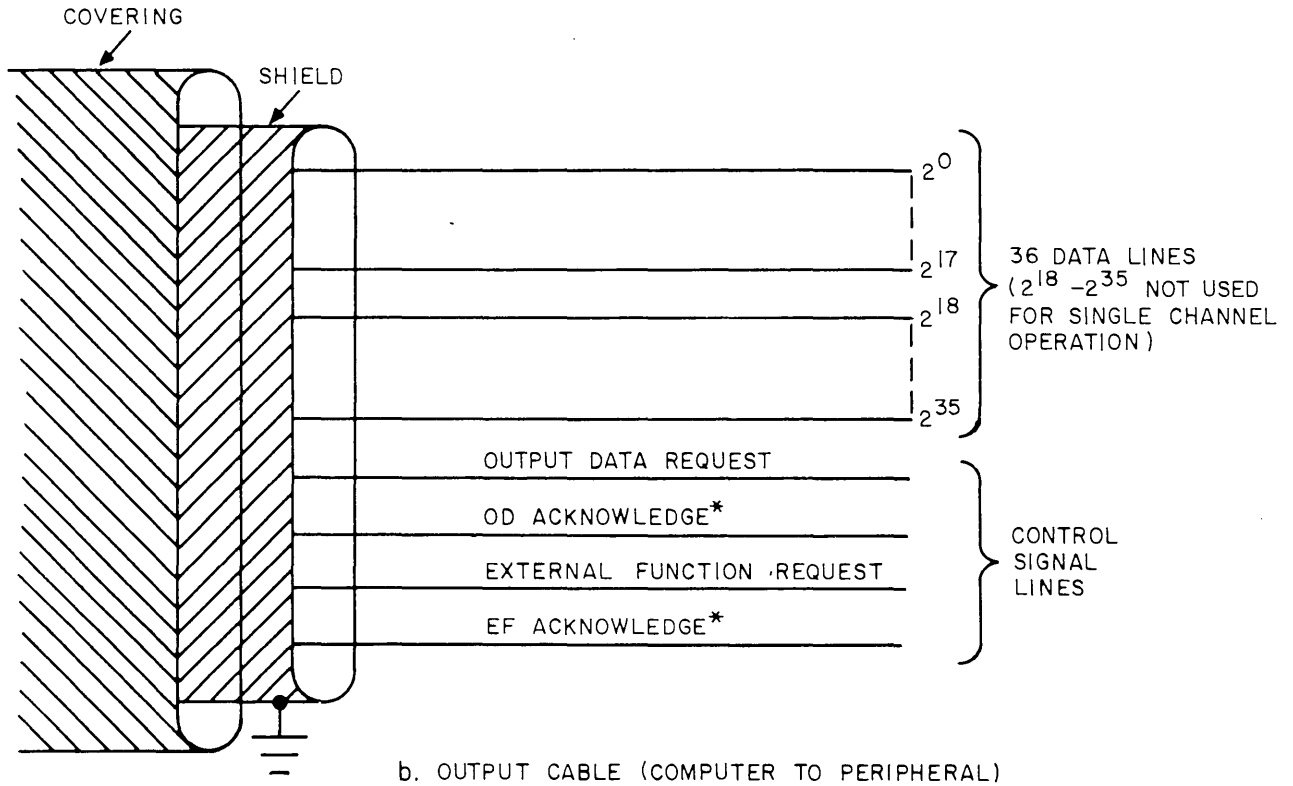
The logic for the 16 channels is distributed among the four I/O chassis, with each chassis controlling four channels. Depending upon the number of channels required, all four I/O chassis may not be present. The number of channels may be any multiple of four up to sixteen. Refer to table 8.1-1 for the channel/chassis assignment.

TABLE 8.1-1. I/O CHANNEL/CHASSIS ASSIGNMENT

CHASSIS	CHANNELS
2	0, 2, 4, 6
1	1, 3, 5, 7
10	10, 12, 14, 16
9	11, 13, 15, 17



a. INPUT CABLE (PERIPHERAL TO COMPUTER)



b. OUTPUT CABLE (COMPUTER TO PERIPHERAL)

NOTES: EACH LINE IS TWISTED WITH A GROUND RETURN LINE (NOT SHOWN).
*ACKNOWLEDGE SIGNALS ARE SENT FROM COMPUTER TO PERIPHERAL. ALL OTHER SIGNALS ARE SENT TO COMPUTER FROM PERIPHERAL.

Figure 8.1-1. I/O Cables

b. Interface. This term refers to the characteristics involved in the mating together of two pieces of equipment so as to enable data or signal exchange between them. All I/O factors such as voltage levels, signal duration, signal rate, and line impedance can be considered under the topic of interface. There are two types of interface employed in this computer. Refer to table 8.1-2 for the basic interface characteristics.

TABLE 8.1-2. I/O INTERFACE CHARACTERISTICS

Interface Type	Line Voltage Levels	ID Acknowledge Signal	OD/EF Acknowledge Signal
Slow	H = 0v (grd.); L = -15v	14.725 ms	14.375 ms
Fast	H = 0v (grd.); L = -3v	2.725 ms	2.375 ms

The acknowledges are signals sent from the computer to the peripheral device indicating the completion of a data word exchange. The signals are discussed later in this sheet.

The type of interface is selected by plug-in jumper cards. All four channels on a common chassis have the same type of interface. Of these four channels, the interface for data input to the computer need not be the same type as for output data operations. The type of interface used is primarily determined by the requirements of the peripheral equipment and cable length.

c. Data Exchange Rate. The time necessary to complete one data word exchange operation for single channel is four microseconds. The maximum I/O rate is one word exchange every two microseconds because of a two microsecond overlap of consecutive word exchange operations.

The type of interface affects the I/O rate because of the acknowledge timing. For example, the ID Acknowledge signal timing is common to the four channels of a chassis and prevents any input data operation on these channels while the signal is being generated. Likewise, the OD/EF Acknowledge signal timing will disable any output data operation on the four channels of a chassis. If dual channel operation (36-bit word transfer) is employed, the acknowledge timing will prevent another data exchange operation of the same type on any channel of the entire drawer (2 chassis). Dual channel operation is discussed in more detail later in this sheet. Refer to table 8.1-3 for the I/O rates.

d. I/O Signals.

1. Data Request Signals.

a) Definition. Request signals are those which are sent from a peripheral device to the computer and cause the transfer of one data word. The request honoring operation requires the use of memory to output or input a word. Therefore, it disables the instruction sequences to stop the program during its 2 microsecond use of memory. The request operations use the I/O1 and I/O2-sequences.

TABLE 8.1-3. DATA TRANSFER RATES (MAXIMUM)*

OTHER CONDITIONS	MODE (WORD SIZE)	FAST INTERFACE			SLOW INTERFACE		
		ONE CHASSIS	TWO CHASSIS	FOUR CHASSIS	ONE CHASSIS	TWO CHASSIS	FOUR CHASSIS
Alternate Inputs and Outputs on Alternate Channels	Single Channel	500	500	500	83.3	167	334
	Dual Channel	NA	250	250	NA	76.9	154
Inputs Only or Outputs Only on Alternate Channels	Single Channel	250	500	500	41.6	83.3	167
	Dual Channel	NA	167	250	NA	38.4	76.9
Inputs Only or Outputs Only on Individual Channel	Single Channel	167	NA	NA	41.6	NA	NA
	Dual Channel	NA	125	NA	NA	38.4	NA
Alternate Inputs and Outputs on Individual Channel	Single Channel	334	NA	NA	83.3	NA	NA
	Dual Channel	NA	250	NA	NA	76.9	NA

*All rates are in thousands of words per second.

b) Types.

1) Input Data Request (IDR). This signal is sent to the computer with a data word. The IDR requests the computer to accept the word.

2) Output Data Request (ODR). This signal requests the computer to output a data word to the peripheral device.

3) External Function Request (EFR). This signal requests the computer to output an external function word to the peripheral device. Usually, the external function word is used to control the peripheral device according to its bit configuration.

The EFR operation could also be used as a technique of outputting data to the peripheral device.

4) External Interrupt Request (EIR). This signal is sent to the computer possibly with a status word. Usually, the status word is used to inform the computer of the condition of the peripheral device as indicated by its bit configuration. For example, it may indicate timing errors, data word content errors, data word content errors, data ready condition, etc. The EIR requests the computer to accept the status word and inform the program of its occurrence by means of a program jump using the EIM interrupt signal. The EIM interrupt is discussed later in this sheet. The status word is stored in memory at the address 00101_8 + 2x channel if the channel is 0-7 or address 00301_8 + 2x channel if the channel is $10\text{-}17_8$.

Some equipments do not send status words but generate the EIR merely as means of interrupting the program. The EIR status word transfer operation could also be used as a technique of inputting data to the computer.

2. Real Time Clock Request (RTC Request). The RTC Request is generated by an internal, accurate oscillator operating at a frequency of 1024 pps. This oscillator is used to maintain a constantly updated 18-bit clock count in the control memory address 00015_8. The RTC request causes the computer to increment by +1 the content of this address. The addition is performed in an open-ended adder manner such that the incrementing of the 777777_8 count results in the 000000_8 count. This clock count can be preset to any value by the program and periodically referenced so as to time certain program events. The honoring of the RTC request can be prevented only by the RTC DISCONNECT switch.

Like the data request signals, the RTC request uses the I/O1-sequence to perform the updating operation.

3. Interrupt Signals.

a) Definition. Interrupt signals are those which cause an interruption to the program in the form of a jump. Each type of interrupt has a special jump address assigned to it. The jump address is also referred to as the interrupt entrance register. The jump is made without disturbing P, such that a return jump instruction could be used at the jump address to retain P for later return to the program at the point of interruption as well as execute a second jump to an interrupt routine.

With the exception of the external sync interrupt, all interrupts are internally generated.

b) Types.

1) Monitor Interrupts.

a. Input Data Monitor Interrupt (ID Mon). This signal can occur on any of the I/O channels. It indicates that the ID buffer set up by the program on its channel has been completed and that the program had desired the interrupt at the ID buffer termination. The ID buffer refers to the specified number of data words to be inputted on the particular channel.

This ID Mon interrupt causes a program jump to address 00160_8 + 2x channel if the channel is 0-7 or address 00360_8 + 2x channel if the channel is $10\text{-}17_8$.

b. External Function/Output Data Monitor Interrupt (EF/OD Mon). This signal can occur on any of the I/O channels. It indicates that either the EF or OD buffer set up by the program on its channel has been completed and that the program had desired the interrupt at the buffer termination. The EF and OD buffers refer to the specified number of external function word transfers and output data transfers, respectively, to occur on the particular channel. Since the EF and OD buffers share common logic, only one of these buffers can be "active" at a time for a particular channel. Therefore, the program must be aware of the buffer type which was active in order to correctly interpret this monitor interrupt.

The EF/OD Mon interrupt causes a program jump to address 00140_8 + 2x channel if the channel is 0-7 or address 00340_8 + 2x channel if the channel is $10\text{-}17_8$.

c. External Interrupt Monitor Interrupt (EI Mon). This signal can occur on any of the I/O channels. It occurs as a result of the honoring of the external interrupt request on its channel. It simply indicates that the EIR was honored and its associated status word (if there is one) has been stored in memory.

The EI Mon interrupt causes a program jump to address 00100_8 + 2x channel if the channel is 0-7 or address 00300_8 + 2x channel if the channel is $10\text{-}17_8$.

2) Special Interrupts.

a. RTC Overflow Interrupt. This signal is generated when the RTC request is honored and increments the clock count from 777777_8 to 000000_8. With the RTC oscillator operating at a frequency of 1024 pps, this interrupt occurs every 256 seconds. It can be prevented by the RTC DISCONNECT switch.

The RTC overflow interrupt causes a program jump to the control memory address 00013_8.

b. RTC Monitor Interrupt. This signal is generated during the honoring of the RTC request if the clock count in address 00015_8 (before incrementation) equals the content of the control memory address 00014_8, and the interrupt was desired by the program. The comparison count in address 00014_8 must be set up by the program. The program must also execute the RTC instruction (f = 50:14) to enable the interrupt when the clock reaches this count. This instruction must be executed after each interrupt if another RTC monitor interrupt is desired.

The RTC monitor interrupt causes a program jump to the control memory address 00012_8.

c. External Sync Interrupt. This is the only interrupt which is directly generated externally. It is a non-channel type and is carried by a separate I/O cable. It can be used as a timing input from an external device or allow external control over the program.

The external sync interrupt causes a program jump to the control memory address 00016_8.

d. Instruction Fault Interrupt. This is the only interrupt which is directly generated by the program. It occurs when the program attempts to execute a format 1 instruction with the function code of 00, 01, or 77_8. These are considered illegal or fault instructions.

The instruction-fault interrupt causes a program jump to the control memory address 00000_8 if the AUTO RECOVERY switch is in the down position or address 00500_8 (bootstrap) if the switch is in the up position. The interrupt also activates the PROGRAM FAULT indicator which can only be extinguished by master clear.

e. Resume Fault Interrupt (Inter-Computer Time-Out Fault). This signal occurs only during output data transfers to another computer. It indicates that the receiving computer did not accept a data word within a specified period of time. It is timed by the RTC operations; therefore, the RTC request must not be disconnected. The amount of time allowed for the acceptance of each data word is determined by the wiring configuration to examine a particular bit position of the RTC clock count. The time period allowed involves two changes of the bit tested.

When the interrupt occurs, it is necessary for the program to execute the SRSM instruction (f = 50:20). Otherwise, all EF and OD operations on the particular chassis involved will be disabled.

The resume-fault interrupt causes a program jump to the control memory address 00011_8.

e. Buffers and Address Control Words. Buffer refers to the number of words set up by the program to be transferred. ID, OD, and EF buffers are established by the instructions IN (f = 50:11), OUT (f = 50:12), and EXF (f = 50:13), respectively. The two addresses immediately following these instructions in the program are reserved as the buffer limits. Buffer limits are the initial and final addresses for the data to be transferred.

The terminal address control word (TACW) immediately follows the buffer instruction in the program. Among other things, it contains the 16-bit final data address. For example, if an OD buffer is being established the TACW specifies the address from where the last data word of the buffer is to be obtained and outputted. The TACW also may request the continuous data mode (CDM) at buffer termination. CDM is discussed later in this sheet. Refer to figure 8.1-2 for the TACW format.

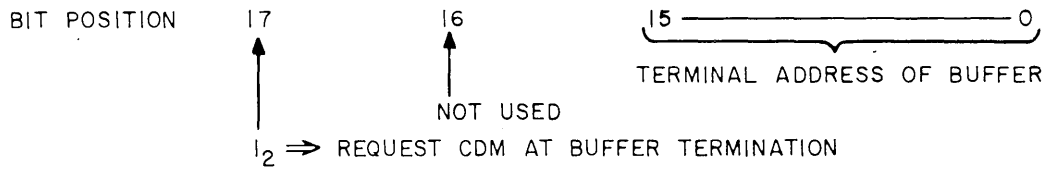


Figure 8.1-2. Terminal Address Control Word Format

The initial address control word (IACW) immediately follows the TACW in the program. Among other things, it contains the 16-bit initial data address. For example, if an ID buffer is being established the IACW specifies the address where the first input data word of the buffer is to be stored. The IACW also may request the monitor interrupt to be generated at buffer termination. That is, after the last word of the buffer is transferred, the ID Mon, OD Mon, or EF Mon interrupt, depending upon the buffer type, is generated on the particular channel.

The IACW also specifies the direction of the buffer. A forward buffer is one which provides sequentially increasing addresses for the data. A backward buffer provides sequentially decreasing data addresses. In both cases, the TACW and IACW specify the terminal and initial buffer addresses, respectively. Refer to figure 8.1-3 for the IACW format.

Once the buffer is established (made active), the IACW is more appropriately referred to as the current address control word (CACW). The TACW and IACW are transferred into special control memory addresses. See Table 8.1-4. In control memory, the IACW is updated by ± 1, depending upon the buffer direction, after each word transfer. It, therefore, always specifies the address for the next word transfer. Thus, the IACW becomes the CACW.

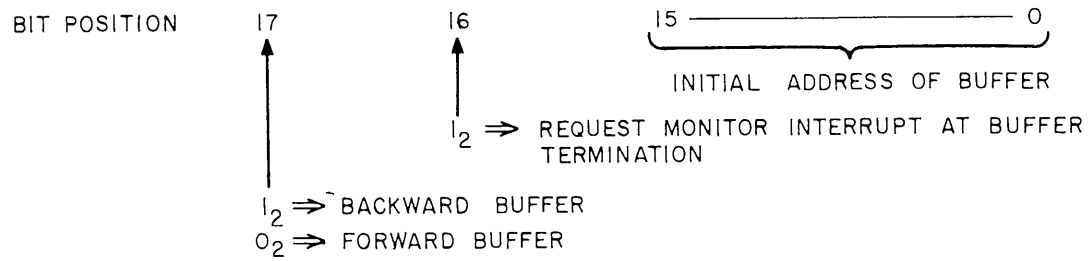


Figure 8.1-3. Initial Address Control Word Format

TABLE 8.1-4. ASSIGNED ADDRESSES FOR ADDRESS CONTROL WORDS

ADDRESS CONTROL WORD	SPECIAL ADDRESSES	
	ID	EF/OD
TACW	00060_8 + 2x chan	00040_8 + 2x chan
IACW (CACW)	00061_8 + 2x chan	00041_8 + 2x chan

All of the I/O instructions are format 2 type. The k designator (6 least significant bits) specifies the channel number. Refer to figure 8.1-4 for programing examples of buffer initiation.

f. Special I/O Logic.

1. Active Flip-Flops. Each channel has an active flip-flop for ID and EF/OD. This flip-flop in the set state is the only requirement for the computer to detect and honor the associated request signal from a peripheral device. The active flip-flop is set by the buffer instruction (f = 50:11 - 50:13). There is an EF Mode flip-flop which allows the computer to distinguish between an EFR and ODR. This is necessary because the EF/OD Active flip-flop is shared by both functions.

At buffer termination, the active flip-flop is cleared to prevent the honoring of more requests than had been determined by the address control words.

2. Priority. Since it is possible to have more than one I/O signal requesting I/O services at the same time, a priority system exists to select one of these signals to be honored. Priority is set up on a channel number and function basis. The channel priority can be altered by plug-in cards. The priority scheme is discussed in a later sheet.

3. I/O Translator. After a signal is selected by the priority logic, the information is placed in the I/O translator. This logic translates the signal type (function) and its channel number and assists in developing the control signals necessary to execute the proper I/O service.

g. Block Diagram Description of Data Request Operations. The I/O1-sequence is used to honor an IDR, ODR, EFR, or EIR. Each of these signals requires a word transfer between the computer and the requesting peripheral device. They may also generate a monitor interrupt which is a separate signal to be considered

PROGRAM ADDRESS		
01000	501202	OUT INSTRUCTION, CHANNEL 2
01001	003000	TACW
01002	20200	IACW

NOTES: 1. IACW SPECIFIES:

a. FIRST OUTPUT DATA WORD WILL BE OBTAINED FROM ADDRESS 02000_8.
b. OD MON INTERRUPT ON CHANNEL 2 AT BUFFER TERMINATION.
c. FORWARD BUFFER.

2. TACW SPECIFIES:

a. LAST OUTPUT DATA WORD WILL BE OBTAINED FROM ADDRESS 03000_8.
b. NO CDM AT BUFFER TERMINATION.

3. IACW AND TACW TOGETHER ESTABLISH A 01001_8 -WORD BUFFER.

a. OD BUFFER EXAMPLE

PROGRAM ADDRESS		
04000	501104	IN INSTRUCTION, CHANNEL 4
04001	006000	TACW
04002	407500	IACW

NOTES: 1. IACW SPECIFIES:

a. FIRST INPUT DATA WORD WILL BE STORED AT ADDRESS 07500_8.
b. NO ID MON INTERRUPT AT BUFFER TERMINATION.
c. BACKWARD BUFFER.

2. TACW SPECIFIES:

a. LAST INPUT DATA WORD WILL BE STORED AT ADDRESS 06000_8.
b. NO CDM AT BUFFER TERMINATION.

3. IACW AND TACW TOGETHER ESTABLISH A 01501_8 -WORD BUFFER.

b. ID BUFFER EXAMPLE

Figure 8.1-4. Buffer Program Examples

by priority before it can be honored to cause the program jump. Refer to figures 8.1-5, 8.1-6, and 8.1-7 for block diagram descriptions.

h. Dual Channel Operations. Dual channel operation involves the use of an odd numbered channel with the next lower channel which is even numbered to effect a 36-bit word transfer. The request signal must occur on the odd channel and the CHANNEL FUNCTION switch (one for each odd channel) must be in the dual position. If the request occurs on the even channel, single channel mode is in effect regardless of the switch.

The 18-bit word for each channel is treated normally as single channel. The only difference is that the acknowledge signal is not sent until after both halves of the 36-bit word have been transferred.

i. Continuous Data Mode (CDM). CDM is initiated at buffer termination if requested by the TACW ($TACW_{17} = 1$). It keeps the channel active by preventing the normal clearing of the associated active flip-flop, and it establishes a new set of address control words. It obtains the new TACW from the control memory address 00020_8 + 2x channel if the channel is 0-7 or address 00220_8 + 2x channel if the channel is $10\text{-}17_8$. These words are transferred to the normal special control memory addresses which hold the words during the buffer operation.

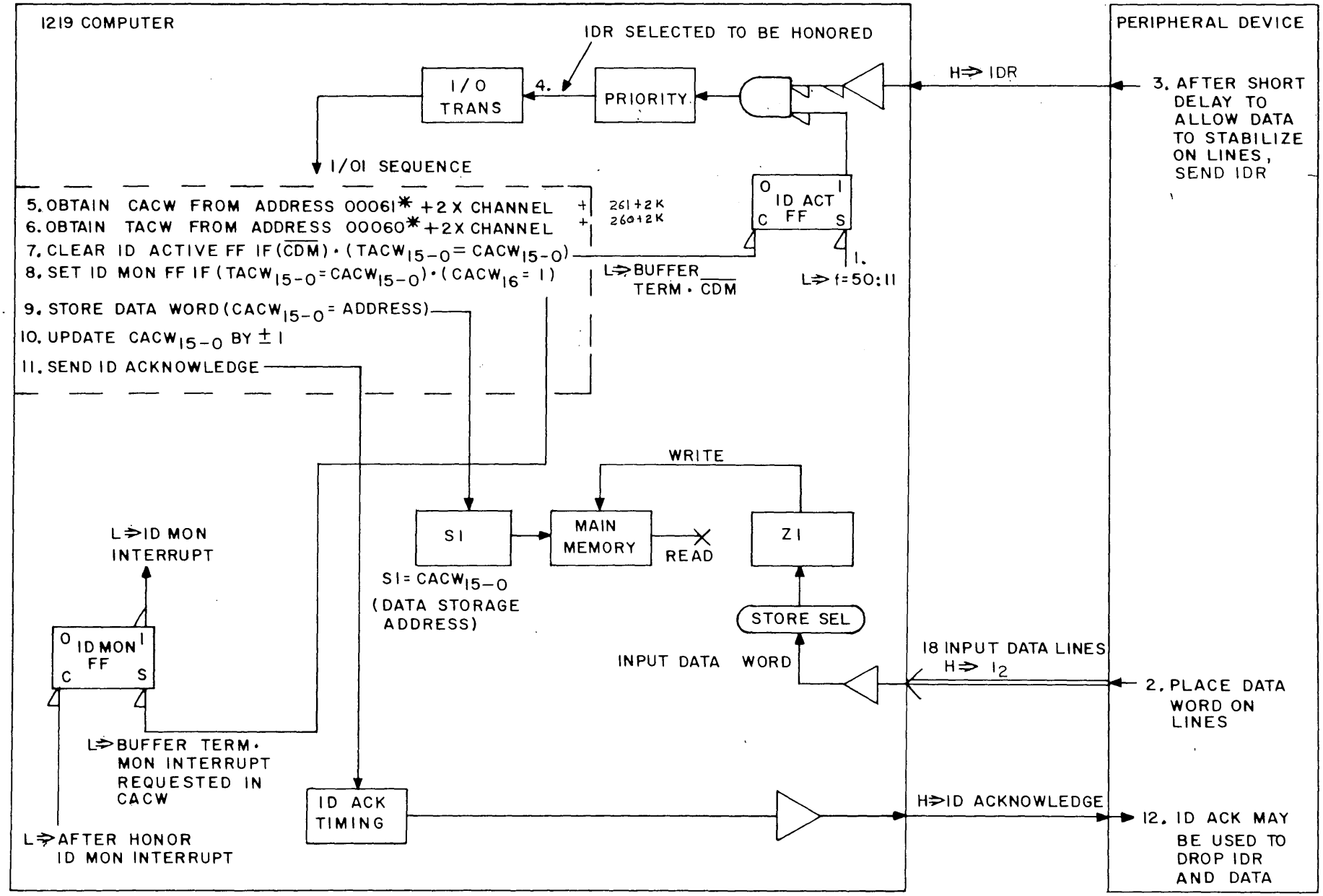
Prior to buffer termination, the program must have stored the "reload" TACW and CACW in preparation for CDM. If requested by the CACW ($CACW_{16} = 1$), the normal monitor interrupt can occur at the termination of the old buffer after the new buffer has automatically been established. Each new set of address control words can determine its buffer direction, requirement for monitor interrupt, and requirement for CDM again.

Usually the monitor interrupt technique is used with CDM. The interrupt is the indication to the program that the address control words have been reloaded and, if CDM reload is to be performed again, the program must prepare the next TACW and CACW. Realize that for a particular channel, all three functions (ID, OD, and EF) use the same addresses to obtain the reload TACW and CACW. Therefore, if both an ID buffer and EF or OD buffer are active at the same time on the same channel, probably the program would not request CDM for both.

j. Externally Specified Index (ESI). ESI operation involves the use of an odd numbered channel with the next lower channel which is even numbered. The normal special addresses used to obtain the address control words are not used. The odd channel carries the control memory address to obtain the TACW. The CACW is extracted from the next consecutively high address. ESI allows the peripheral device to select its own address control words for each word transferred. These address control words must be set up by the program. The word transfer is effected over the even channel.

To enable ESI operation, the request signal must occur on the odd channel and the CHANNEL FUNCTION switch must be in the ESI position. If the request occurs on the even channel, single channel mode is in effect regardless of the switch.

ESI mode with an EIR forces dual channel operation (36-bit status word).



NOTES: ONCE ID ACTIVE FF IS SET, STEPS 2 THROUGH 12 ARE REPEATED FOR EACH INPUT DATA WORD TRANSFER.
*ADD 00200_8 TO ADDRESSES OF CACW AND TACW IF CHANNEL IS 10-178.

Figure 8.1-5. Input Data Operations, Single Channel Without ESI and ESA

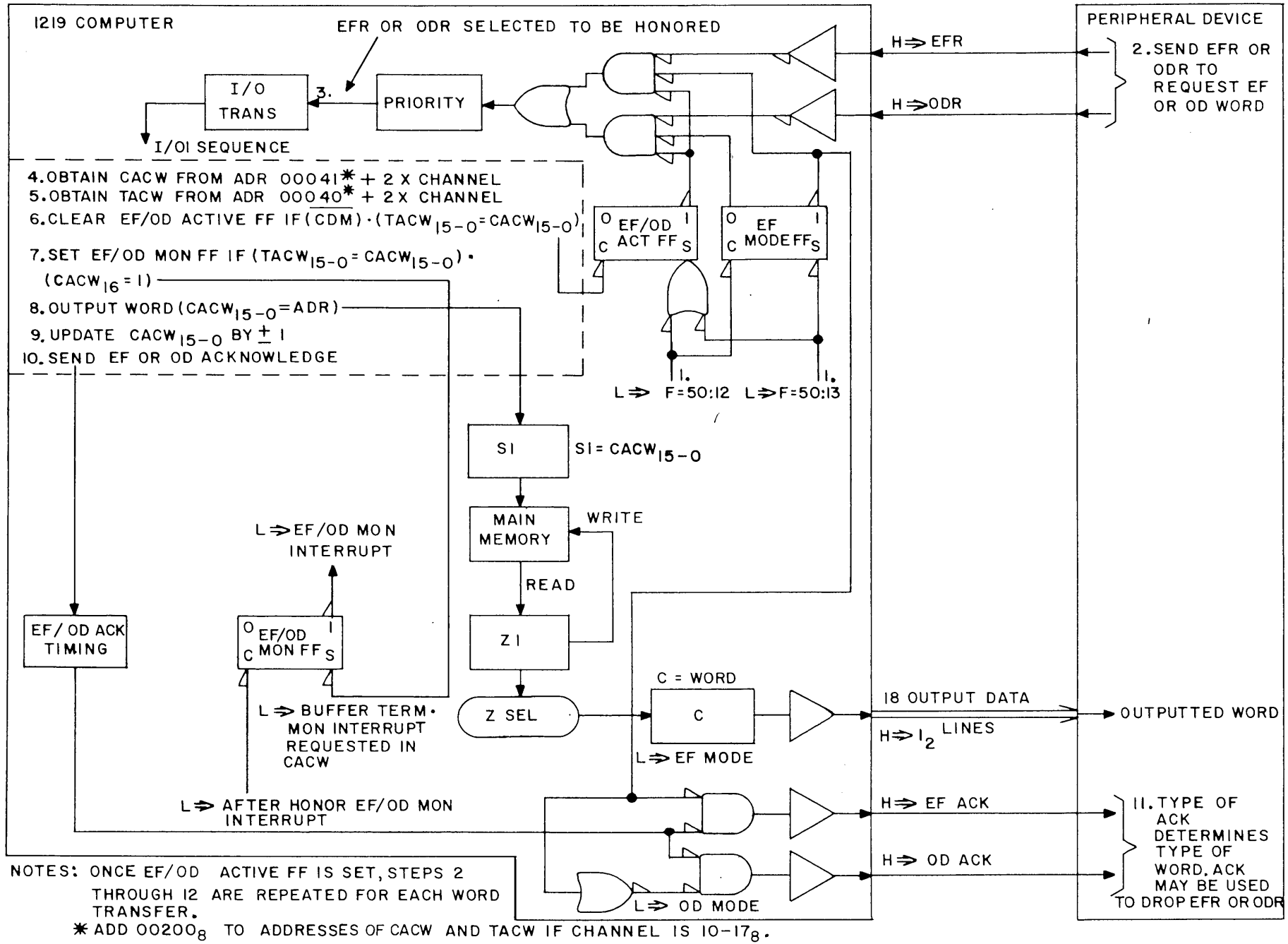
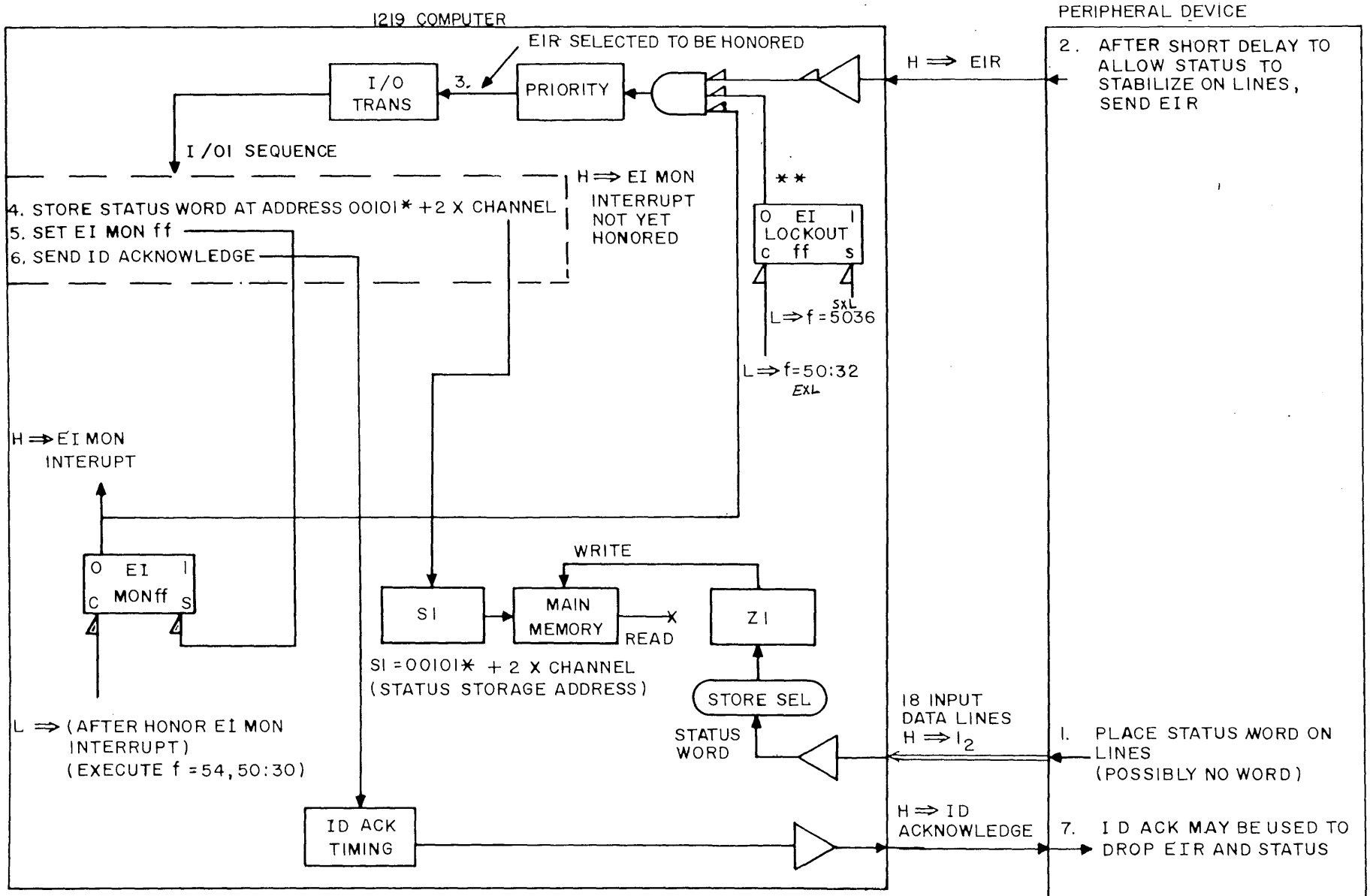


Figure 8.1-6. External Function/Output Data Operations, Single Channel Without ESI and ESA



NOTES: * ADD 00200_8 TO STATUS STORAGE ADDRESS IF CHANNEL IS $10-17_8$

** EI LOCKOUT ff ALLOWS PROGRAM TO ENABLE AND DISABLE HONORING OF ALL EIR'S

Figure 8.1-7. External Interrupt Operations, Single Channel

k. Externally Specified Address (ESA). ESA operation involves the use of an odd numbered channel with the next lower channel which is even numbered. With this operation in effect, there are no address control words. The peripheral device directly specifies over the odd channel, the address for the word transfer requested. The word transfer is effected over the even channel.

To enable ESA operation, the request signal must occur on the odd channel and the CHANNEL FUNCTION switch must be in the ESA position. If the request occurs on the even channel, single channel mode is in effect regardless of the switch.

ESA mode with an EIR forces dual channel operation (36-bit status word).

8.1-5. SUMMARY

Read the UNIVAC 1219 Technical Manual, Volume I, Paragraph 3-6c(2) for a description of the I/O instructions. More detailed descriptions of these are presented in later sheets. Review the general description of the I/O section by reading UNIVAC 1219 Technical Manual, Volume I, Paragraphs 1-5d and 4-3a through 4-3e.

SECTION 8 - INPUT/OUTPUT SECTION

8.2. INSTRUCTION EXECUTION OF SIN, SOUT, SEXF, INSTP, OUTSTP, EXFSTP

8.2-1. OBJECTIVES

To present the detailed theory of operation involved in the execution of instructions with f = 50:01 - 50:03, 50:15 - 50:17.

8.2-2. INTRODUCTION

These instructions set one of the active flip-flops or terminate ID, OD, or EF operations on a selected channel.

8.2-3. REFERENCES

a. UNIVAC 1219 Technical Manual, Volume I, Paragraphs 4-3d and 4-7, table 4-11.

b. UNIVAC 1219 Technical Manual, Volume II, Section 9 (logic diagrams).

8.2-4. INFORMATION

a. General Description.

1. Instruction Interpretation.

a) SIN, f = 50:01. This instruction sets the ID Active flip-flop for the channel specified by the four least significant bits (k) of the instruction word. Input data operations are enabled on this channel.

b) SOUT, f = 50:02. This instruction sets the EF/OD Active flip-flop and clears the EF Mode flip-flop for channel k. Output data operations are enabled on this channel.

c) SEXF, f = 50:03. This instruction sets the EF/OD Active flip-flop and sets the EF Mode flip-flop for channel k. External function operations are enabled on this channel.

d) INSTP, f = 50:15. This instruction terminates input data operations on channel k by clearing its ID Active flip-flop. Its ID Monitor flip-flop is also cleared to prevent the occurrence of an ID monitor interrupt.

e) OUTSTP, f = 50:16 and EXFSTP, f = 50:17. Since the OD and EF buffer operations for the same channel share common logic, these instructions are actually the same. They terminate output operations (OD and EF) on channel k by clearing the associated EF/OD Active flip-flop. The EF/OD Monitor flip-flop for the channel is also cleared to prevent the occurrence of an EF/OD monitor interrupt.

Both instructions are included in the repertoire simply to provide compatibility with the UNIVAC 1218 Computer programs.

2. Execution Sequence (I). All operations are performed within the I-sequence. Only the one memory reference to obtain the instruction is necessary.

b. Detailed Analysis.

1. Data Flow Block Diagram. Refer to figure 8.2-1 for a block diagram description of the execution of f = 50:01 - 50:03, 50:15 - 50:17.

Most of the I-sequence operations are as previously described. If necessary, refer to study guide sheet number 5.4 for a detailed description.

K0 is set to the six least significant bit positions of the instruction word from Z-select. Only the four lower bits of this value are used. At T3.2 time, the I/O translator register is set to $K0_{3-0}$ which is the channel number. This register is also affected by the function code which indicates the type of buffer (ID, OD, or EF).

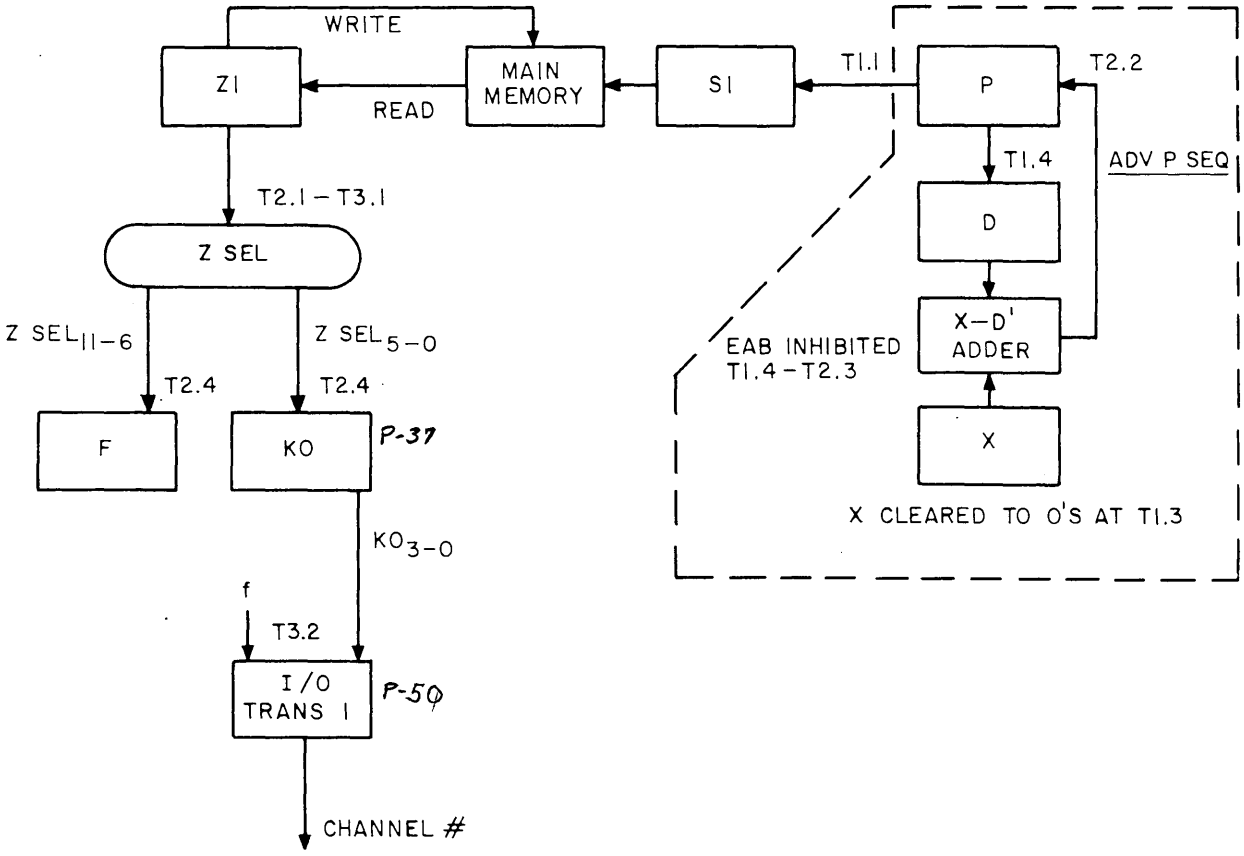
As discussed in a prior sheet, the instruction could be obtained from bootstrap or control memory.

2. Essential Commands. Refer to table 8.2-1 for a sequential list of essential I-sequence events. Develop these commands by referring to the proper enable pages in the logic diagrams.

8.2-5. SUMMARY

The f = 50:01 - 50:03, 50:15 - 50:17 instructions are format 2 and use the value k. The k value is available in K0 after T2.4 time. Only the I-sequence is required to complete the execution of these instructions.

9-49 I/O TRANSLATOR # SCAN TIMING
9-53 I/O COMMANDS
9-57 " " & CONTROL II
9-58 " " CONTROL III
9-59 TRANSLATOR I
9-60 TRANSLATOR II



T4.3 IF f = 50:01; SET ID ACT FF p. 61 → 64
IF f = 50:02; SET EF/OD ACT FF & CLEAR EF MODE FF
IF f = 50:03; SET EF/OD ACT FF & SET EF MODE FF
IF f = 50:15; CLEAR ID ACT FF & CLEAR ID MON FF
IF f = 50:16; 50:17; CLEAR EF/OD ACT FF & CLEAR EF/OD MON FF

Figure 8.2-1. I-Sequence Data Flow for f = 50:01 - 50:03, 50:15 - 50:17

TABLE 8.2-1. I SEQUENCE ESSENTIAL COMMANDS FOR
f = 50:01 - 50:03, 50:15 - 50:17

TIME NOTATION	COMMANDS
T4.4	Clear S1
T1.1	P → S1, Init Memory, *set Incr P ff
T1.3	*Clear D, *clear X, clear Z1, clear F, *set OXL11 ff
T1.4	$*P_L \rightarrow D_{L'}$, $*P_U \rightarrow D_{U'}$, clear K0, *set Inhib EAB ff
T2.1	*Clear P, Z1 → Z Sel, *clear Incr P ff
T2.2	*Adder → P
T2.3	*Clear OXL11 ff, *clear Inhib EAB ff
T2.4	$Z\ Sel_{11-6} \rightarrow F$, set OXF06 ff, $Z\ Sel_{5-0} \rightarrow K0$
T3.1	Clear I/O Trans 1, drop Z1 → Z Sel
T3.2	f & K0 → I/O Trans 1
T4.3	**Set Act ff, **clear Act & Mon ff's

*These events are concerned with or are controlled by the advance-P subsequence.

** if f = 50:01; set ID Act ff
if f = 50:02; set EF/OD Act ff & clear EF Mode ff
if f = 50:03; set EF/OD Act ff & set EF Mode ff
if f = 50:15; clear ID Act ff & clear ID Mon ff
if f = 50:16, 50:17; clear EF/OD Act ff & clear EF/OD Mon ff
I/O Translator 1 selects the channel number.

SECTION 8 - INPUT/OUTPUT SECTION

8.3. INSTRUCTION EXECUTION OF IN, OUT, EXF

8.3-1. OBJECTIVES

To present the detailed theory of operation involved in the execution of instructions with f = 50:11 - 50:13.

8.3-2. INTRODUCTION

These instructions set-up ID, OD, or EF buffer operations on a particular channel with specified buffer limits.

8.3-3. REFERENCES

a. UNIVAC 1219 Technical Manual, Volume I, Paragraphs 4-3d and 4-7, tables 4-11, 4-15, and 4-16.

b. UNIVAC 1219 Technical Manual, Volume II, Section 9 (logic diagrams).

8.3-4. INFORMATION

a. General Description.

1. Instruction Interpretation.

a) IN, f = 50:11. This instruction enables input data operations on channel k by setting its ID Active flip-flop. The terminal address control word (TACW) is taken from the next consecutive address after this instruction. The initial address control word (IACW) is taken from the next consecutive address after the TACW. These control words specify the area of memory involved in the data exchange operation.

b) OUT, f = 50:12. This instruction enables output data operations on channel k by setting its EF/OD Active flip-flop and clearing its EF Mode flip-flop. The origins of TACW and IACW are the same for f = 50:11.

c) EXF, f = 50:13. This instruction enables external function operations on channel k by setting its EF/OD Active flip-flop and setting its EF Mode flip-flop. The origins of TACW and IACW are the same as for f = 50:11.

2. Execution Sequences.

a) I-Sequence. During the I-sequence which obtains the instruction from memory, the channel number is placed in K0.

b) IB1-Sequence. The B1 and I-sequences run in parallel to obtain the TACW from memory.

c) IB2-Sequence. The B2 and I-sequences run in parallel to store the TACW in control memory, obtain the IACW from memory, and set the proper active flip-flop.

d) Next I-Sequence. The initial portion of the I-sequence for the next instruction is used to store the IACW in control memory.

b. Detailed Analysis.

1. I-Sequence. The I-sequence operations are as previously described. If necessary, refer to study guide sheet number 5.4 for a detailed description. At the end of the I-sequence, KO contains the channel number. Only the four least significant bits of KO are interpreted.

2. Data Flow Block Diagram. Refer to figure 8.3-1 for a block diagram description of the execution of f = 50:11 - 50:13.

The IB1-sequence uses a memory reference to obtain the TACW from the address contained in P. P was advanced to the next consecutive address after this instruction by the advance-P subsequence during the I-sequence which obtained this instruction. The advance-P subsequence is used during this IB1-sequence to increment P to address of the IACW.

The TACW is applied to one side of the adder from D. X is set to all 1's and applies this -0 value to the other side. The adder, therefore, outputs the TACW unmodified which is placed in Z1. The I/O translator register is set according to the channel number from KO and the type of active flip-flop to be set from the function code translator.

The IB2-sequence uses a control memory reference to store the TACW. The storage address is dependent upon the function code and channel number as supplied by the I/O translator register. The gating of control memory to ZO is disabled during the read portion of the memory cycle which destroys the original memory content.

Another memory cycle is used to obtain the IACW from the address contained in P. The IACW passes through the X-D' adder unmodified just as did the TACW and is placed in Z1. The advance-P subsequence is used to increment P to the address of the next instruction in the program.

The initial portion of the I-sequence for the next instruction initiates a control memory reference to store the IACW. The storage address for the IACW is one greater than that for the TACW. The IACW storage address is formulated by setting SO from the I/O translator register and also setting SO_{00} to 1_2.

3. Essential Command. Refer to table 8.3-1 for a sequential list of essential IB1, IB2, and next I-sequence events. Develop these commands by referring to the proper enable pages in the logic diagrams.

8.3-5. SUMMARY

The f = 50:11 - 50:13 instructions are format 2 and use the value k. The k value is available in KO after T2.4 time of the I-sequence. The IB1 and IB2-sequences are required to complete the executions of these instructions.

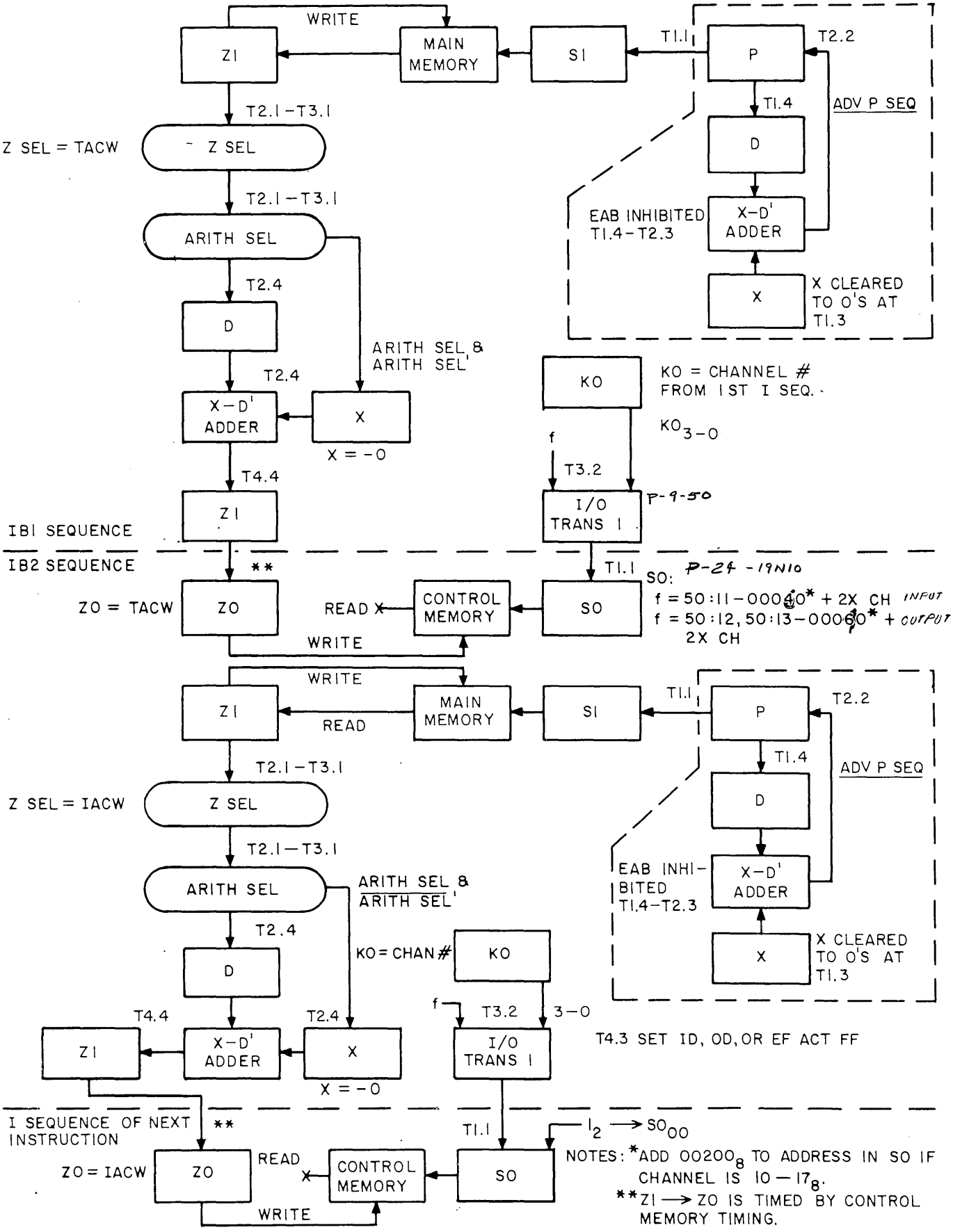


Figure 8.3-1. IB1 and IB2-Sequence Data Flow for f = 50:11 - 50:13

TABLE 8.3-1. IB1, IB2, AND NEXT I SEQUENCE ESSENTIAL COMMANDS
FOR f = 50:11 - 50:13

TIME NOTATION	COMMANDS
	IB1 SEQUENCE
T4.4	Clear S1
T1.1	P→S1, Init Memory, *set Incr P ff
T1.3	*Clear D, *clear X, clear Z1, *set 0XL11 ff
T1.4	*P_L→D_L, *PU→DU, *set Inhib EAB ff
T2.1	*Clear P, Z1→Z Sel, Z Sel→Arith Sel, *clear Incr P ff
T2.2	*Adder→P
T2.3	*Clear 0XL11 ff, *clear Inhib EAB ff, clear D, clear X
T2.4	Arith Sel→D, Arith Sel→X & Arith Sel'→X (set X = 1's)
T3.1	Clear I/O Trans 1, drop Z1→Z Sel, drop Z Sel→Arith Sel
T3.2	f & K0→I/O Trans 1
T4.3	Clear Z1
T4.4	Adder→Z1, **disable CM→ZO, clear S1
	IB2 SEQUENCE
T1.1	I/O Trans 1→SO, Init CM, P→S1, Init Memory, * set Incr P
T1.3	*Clear D, *clear X, clear Z1, *set 0XL11 ff
T1.4	*P_L→ D_L, *P_U→D_U, *set Inhib EAB ff, drop disable CM→ZO
T2.1	*Clear P, Z1→Z Sel, Z Sel→Arith Sel, *clear Incr P ff
T2.2	*Adder→P
T2.3	*Clear 0XL11 ff, *clear Inhib EAB ff, clear D, clear X
T2.4	Arith Sel→D, Arith Sel→X & Arith Sel'→X (set X = 1's)
T3.1	Clear I/O Trans 1, drop Z1→Z Sel, drop Z Sel→Arith Sel
T3.2	f & K0→I/O Trans 1
T4.3	Clear Z1, ***set ID, OD, or EF Act ff
T4.4	Adder→Z1, **disable CM→ZO
	I SEQUENCE OF NEXT INSTRUCTION
T1.1	I/O Trans 1→SO, 1_2 →SO_{00}, Init CM
T1.4	Drop disable CM→ZO

*These events are concerned with or are controlled by the advance P subsequence.

**Z1→ZO is timed by control memory timing.

***if f = 50:11, set ID Act ff
if f = 50:12, set EF/OD Act ff & clear EF Mode ff
if f = 50:13, set EF/OD Act ff & set EF Mode ff
I/O Translator 1 selects the channel number.

SECTION 8 - INPUT/OUTPUT SECTION

8.4. SIGNAL DETECTION AND SELECTION

8.4-1. OBJECTIVES

To present the detailed theory of operation involved in input/output signal selection.

8.4-2. INTRODUCTION

The signal selection logic determines which one of the present input/output signals will be honored first.

8.4-3. REFERENCES

a. UNIVAC 1219 Technical Manual, Volume I, Paragraph 4-3.

b. UNIVAC 1219 Technical Manual, Volume II, Section 9 (logic diagrams).

8.4-4. INFORMATION

a. General Description.

1. Signal Types. The input/output section of the computer handles signals generated externally to the computer as well as some originating internally. The types of external signals are external interrupt request (EIR), external function request (EFR), output data request (ODR), input data request (IDR), and external sync. The first four of these listed are channel type. That is, each of the 16 channels can generate these requests. All other signals are generated from within the computer. Among these are the monitor interrupts (EI Mon, EF/OD Mon, and ID Mon). These interrupt signals are also channel type.

The computer can handle only one signal at a time and others present must wait. A priority scheme is established by the logic to service the signals in a specific sequence. The priority set-up actually has two levels. One is for channel priority, as required by the channel type signals; and the other is for function priority to select the type of signal to be honored (serviced). Refer to table 8.4-1 for the signal types listed in their order of function priority.

If a data request or monitor interrupt type is to be honored, channel priority is considered first. Then, the type of signal is selected from those of the function group which are present on the selected channel. The priority scheme is discussed in more detail later in this sheet. Refer to figure 8.4-1 for a block diagram showing all of the signals handled by the input/output logic.

The priority logic for the data requests and monitor interrupts is discussed later in this sheet. The priority logic for the special interrupts is analyzed in a later sheet.

TABLE 8.4-1. SIGNALS PRESENTED TO INPUT/OUTPUT LOGIC

(non-channel)	RTC Request	
Data Requests (channel type)	EIR EFR or ODR* IDR*	disabled by External Interrupt Lockout flip-flop set
Special Interrupts (non-channel)	Instruction Fault Resume Fault RTC Mon Ext Sync RTC Overflow	disabled by All Interrupt Lockout flip-flop set
Monitor Interrupts (channel type)	EI Mon EF/OD Mon* ID Mon*	

EFR and ODR are listed together because only one of these buffer types can be active at one time on one channel. Signals are listed in their order of function priority.

* Priority of these signals can reverse.

2. Data Request and Monitor Interrupt Priority. Priority logic is used to perform the actual selection of one of the signals presented to it during one of the two scan periods. The scan periods are described later in this sheet.

Channel logic is distributed among four chassis. Each chassis is connected to four channels. The channels are divided into two groups. Channels 0 - 7 are considered the lower group. Channels 10 - 17 are considered the upper group. Of each group, channel priority selects the highest numbered channel which has a signal present. The signals of the selected channel enter function priority which determines the type of signal of the selected channel to be honored.

Group priority then determines that a signal selected from the upper group (channels 10 - 17) will be honored before a signal from the lower group (channels 0 - 7). The overall priority setup is such that the highest numbered channel which carries a signal is selected and one signal type on that channel is selected.

3. Data Request Scan. A certain period of the main timing cycle is used to scan the computer input signals for data requests. Data requests include external interrupt, external function, input data, and output data requests. These are all signals from external devices which require some computer service.

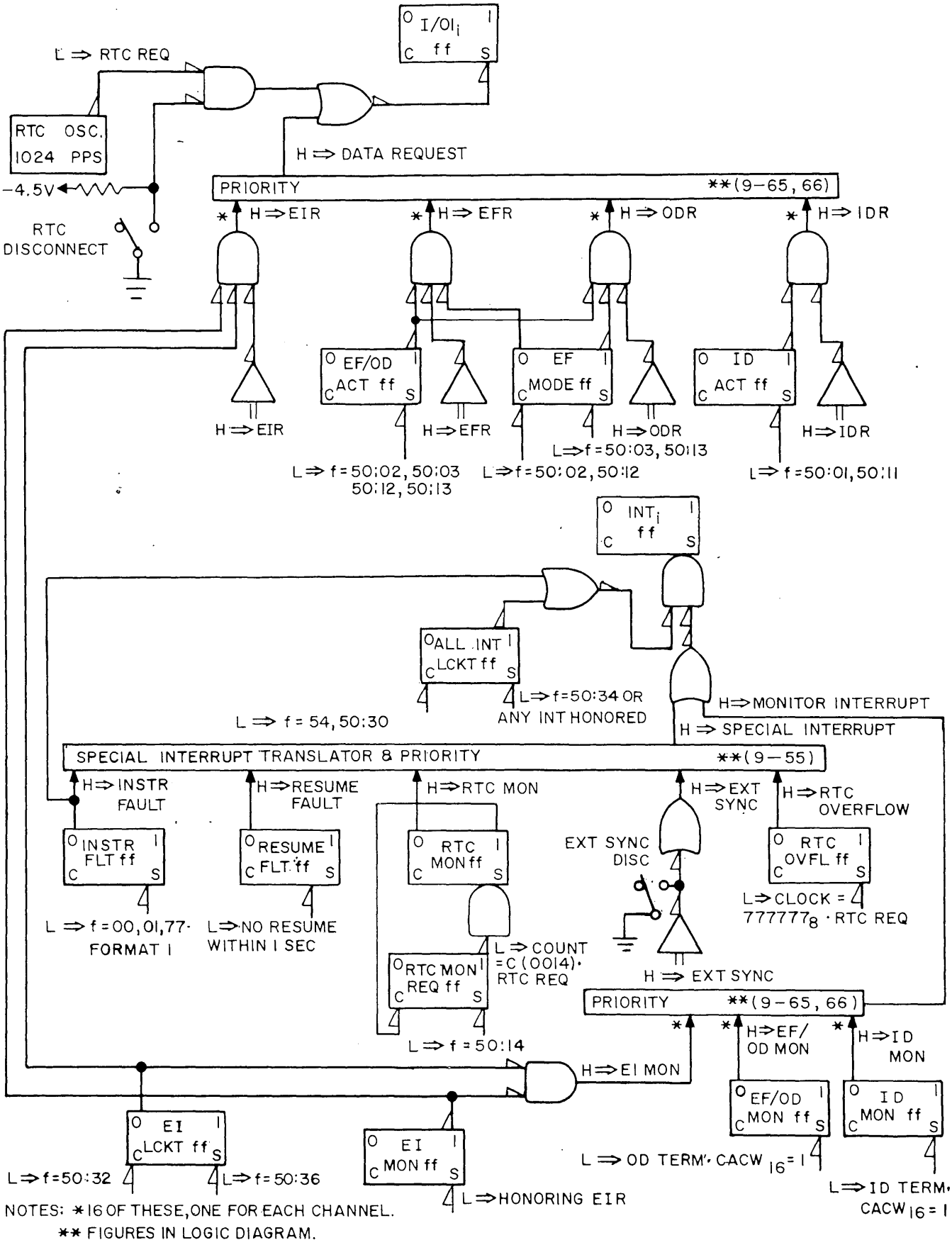


Figure 8.4-1. Simplified Signal Input Logic/Block Diagram

The described priority scheme is altered if an IDR or EIR occurs immediately after an ODR or EFR is honored. In this case, the IDR has higher priority than an ODR. The priority of ODR's and IDR's is therefore alternated if they continuously occur.

Real time clock (RTC) operations are discussed in a later sheet.

4. Interrupt Request Scan. If no data request was detected during the scan period described above, another scan occurs for monitor interrupts. The same priority logic is used to select a particular signal to be honored. Refer to table 8.4-1 for the monitor interrupt types. If either an ODR or EFR was previously honored and an IDR or EIR was not since honored, an ID Mon has higher priority than an OD Mon.

Special interrupt operations are discussed in a later sheet.

b. Detailed Analysis.

1. Request Signal Entrance Logic. The logic which detects the input/output signals is constructed in a one-shot manner. This arrangement prevents the signal from being honored more than once. Any signal which is selected and honored must drop and reappear before it can again be detected. Refer to logic diagrams, figure 9-61 for the channel 0, 1 signals input logic.

This logic is either for channel 0, 10_8, 1, or 11_8 depending upon the particular chassis. Each type of signal has an associated flip-flop. The IDR signal detection is analyzed as an example.

The ID Request flip-flop 0XRg0 provides the one-shot function for the IDR. A low logic level from 05Yg0 represents an IDR being applied to the computer. If the IDR is absent, the ID Request flip-flop is set during Ø2. A resulting low level enable is applied to gate 05Rg0 on pin 8. The low level enable on pin 7 is present only if the ID Active flip-flop is set. This flip-flop must have been set by a previously executed f = 50:01, 50:11 instruction.

The low level enable on pin 6 of 05Rg0 is present if the input circuitry is available. Actually this line is at a low level if the input acknowledge signal is not currently being generated.

The low level enable on pin 5 of 05Rg0 is present during the signal scan period. 05Rg0 is fully enabled if the IDR is present which applies a low level on its pin 9. The resulting high level output of 05Rg0 applies this IDR to the priority logic on this particular chassis. The 80Rg0 output is to channel priority. The 16Rg4 output is to function priority.

If this IDR is selected by the priority logic and honored, the ID Request flip-flop 0XRg0 is cleared to provide the signal one-shot function. Pin 10 of 0XRg0 becomes a low level during the honoring operation. Pin 9 is a low level indicating that this channel is being serviced. The clearing of this flip-flop presents a high level disable to gate 05Rg0, pin 8. Therefore, the recognition of the IDR signal on this channel is prevented until flip-flop 0XRg0 is set. The setting of 0XRg0 can only occur if the external equipment drops the IDR signal. The clearing is timed by Ø2.

EIR, EFR, and ODR signals pass through similar entrance logic. EF and OD buffers cannot be active at the same time on the same channel. They have a common active

flip-flop which, when set, enables the recognition of either the EFR or ODR. The status of the EF Mode flip-flop determines which type of buffer is active. If set, it allows the EFR to be sent to priority. If cleared, it enables the ODR.

2. Data Flow Block Diagram.

a) Data Request Scan. Refer to figure 8.4-2 for a block diagram description of the input/output signal selection.

All signals are presented to the one-shot entrance gates. The Chan Req ⟶ Chan Pri (T4.2-T1.2) command forces channel priority on each chassis to enable all channels. The resulting enables to the one-shot gates allow all input signals to be detected. However, only the outputs of the gates for EIR, EFR, ODR, and IDR signals are allowed to enter channel priority at T1.1 time. These data request signals are enabled by the Data Req ⟶ Chan Pri command. The monitor interrupt signals do not enter channel priority at this time because of the absence of the Int Req ⟶ Chan Pri command.

As soon as the channel priority on each chassis receives the request signals, channel selection is made. The channel priority register for channels 0, 2, 4, 6 is connected to the channel priority logic for channels 1, 3, 5, 7 and vice versa. This interconnection causes the channel priority logic for these two chassis to act as one priority system. This system then selects the highest numbered channel of 0-7 which carries an input signal. The channel priority logic for channels 10-17 is likewise interconnected. It then selects one channel of 10-17.

The Chan Req ⟶ Chan Pri command drops at T1.2 time. By this time, channel priority has performed the selection. Since channel priority is no longer forced to indicate all channels, only the selected channel one-shot gates are enabled (one gate of channels 0-7 if there is an input signal present and one gate of channels 10-17 if there is an input signal present).

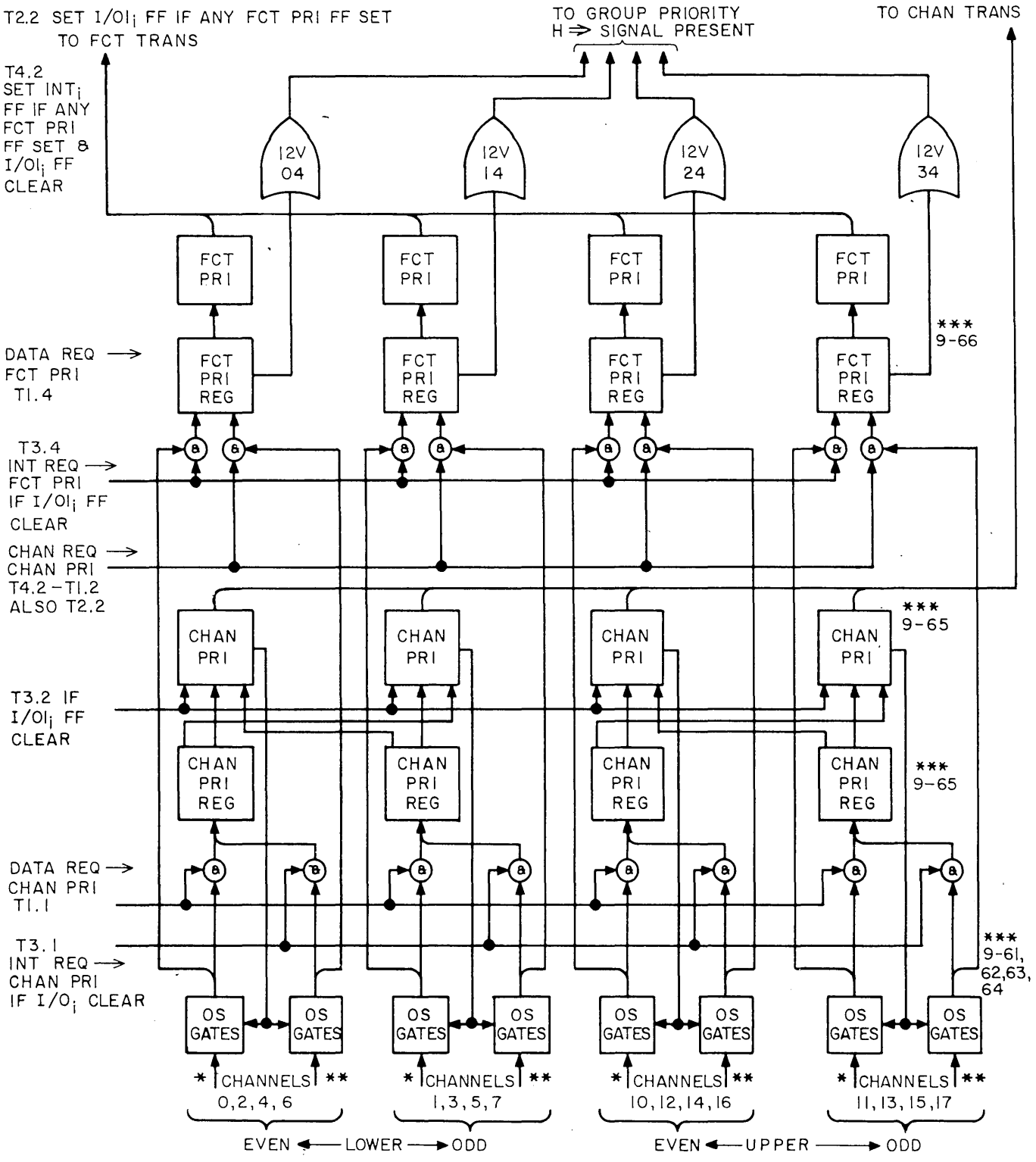
The Data Req ⟶ Fct Pri command enters the function priority register with the signals from the enabled one-shot gates. Only the signals on the selected channels are affected.

Function priority presents a high logic level to group priority if it contains a signal. Group priority selects the upper group (channels 10-17) if there is an associated input signal; otherwise, the lower group (channels 0-7) is selected. One particular channel is chosen after group priority and function priority selects the type of signal on this channel.

If a signal was processed during the above events, the I/O 1_i Sequence flip-flop is set at T2.2 time. At this time, the selected signal information is passed to the function, channel, and I/O translators. The translators outputs are used to control the computer operations to effect the honoring of the selected signal.

b) Interrupt Request Scan. Refer to figure 8.4-2.

If the I/O1_i flip-flop was not set during the data request scan, no EIR, EFR, ODR nor IDR signal was detected. A scan is then made for monitor interrupts. The Chan Req ⟶ Chan Pri command occurs again at T2.2-T3.2 time to allow channel priority to enable all one-shot entrance gates. During this scan period, however, only the entrance gates for the monitor interrupts are allowed to pass their signals to channel priority because of the presence of the Int Req ⟶ Chan Pri command.



NOTES: *ODR, IDR, EFR, & EIR SIGNALS.
**OD MON, ID MON, EF MON, & EI MON SIGNALS (INTERNALLY GENERATED).
***FIGURES IN LOGIC DIAGRAMS.

Figure 8.4-2. Input/Output Signal Selection Flow

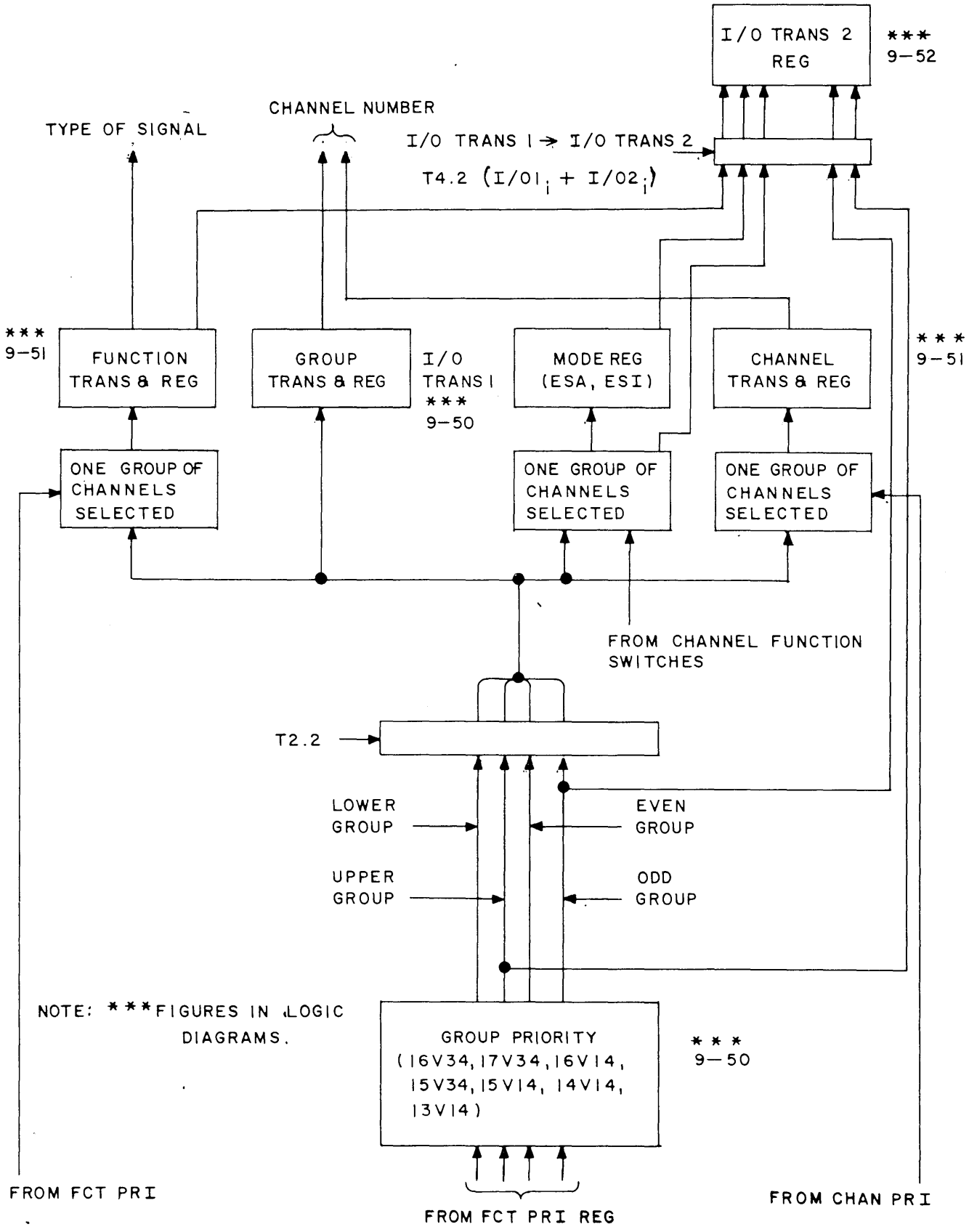


Figure 8.4-2. Input/Output Signal Selection Flow (Cont.)

Channel priority functions as described for the data requests. During the Int Req⟶Fct Pri command, the signals of the selected channels enter function priority which selects one type of signal. Group priority selects one particular channel. At the next T2.2 time, the signal information is passed to the translators.

If a monitor interrupt signal was detected during the above events, the Int_i Sequence flip-flop is set at T4.2 time. The setting of this flip-flop is prevented by the I/Ol_i Sequence flip-flop if a data request signal was detected during the first scan.

3. Essential Commands. Refer to table 8.4-2 for a sequential list of essential events concerning signal selection. After the commands shown, events follow to honor the selected signal. These events and RTC operations are discussed in later sheets. Develop the commands shown by referring to the proper enable pages in the logic diagrams.

4. Channel Priority. Refer to logic diagrams, figure 9-65.

Each I/O chassis has channel priority logic. Request signals enter channel priority by way of the 80Rg- gates during the data request scan period. If no request is detected, the monitor interrupt signals enter by way of the 80Mg- gates during the interrupt request scan period.

The channel flip-flops are set if any signal is present on their channels, regardless of signal type. The 11Vg- gates provide the channel enable which "opens" all of the signal entrance gates to allow all data requests or monitor interrupts to enter channel priority. The 11Vg- gates have a common "Chan Req⟶Chan Pri" input which forces these gates to enable all channels. When this signal drops, all signals have been inputted to channel priority and one channel has been selected. The 10Vg- inputs to the 11Vg- gates enable one channel. Thus, when the signals enter function priority, only the data requests or monitor interrupts of the selected channel are involved.

The channel priority logic on the two chassis of the same drawer are interconnected. These are the dotted-line inputs and outputs shown. These lines are connecting a chassis with odd channels to the chassis with even channels and vice-versa. It is this interconnection which causes the highest channel number with a signal present to be selected among the eight channels of the particular drawer. Thus, only one channel priority of each drawer can indicate a selected channel.

The 10Vg- gates perform the actual channel selection. Refer to table 8.4-3 for the conditions to enable these gates. Realize that these gates are fed by the channel priority flip-flops on this chassis as well as on the other chassis of the same drawer.

5. Function Priority. Refer to logic diagrams, figure 9-66.

Each I/O chassis has function priority logic. Only those signals which are present on the channel selected by channel priority are enabled to enter function priority. There is a separate flip-flop for each function type. Request signals enter this logic by way of the 16Rg- gates during the data request scan period. If no request is detected, the monitor interrupt signals enter by way of the 16Mg- gates during the interrupt request scan period.

TABLE 8.4-2. SCAN SEQUENCE ESSENTIAL COMMANDS

TIME NOTATION	COMMANDS
T4.2	Chan Req ⟶ Chan Pri P-65
T4.4	Clear Chan & Fct Pri
T1.1	Data Req ⟶ Chan Pri
T1.2	Drop Chan Req ⟶ Chan Pri
T1.4	Data Req ⟶ Fct Pri
T2.1	Clear Fct XLATOR, Chan XLATOR, & I/O Trans 1 P-50 E 51
T2.2	Set $I/O1_i$ ff if any Fct Pri ff set
	Pri ⟶ Fct, Chan, & I/O Trans 1 if any Fct Pri ff set
	Chan Req ⟶ Chan Pri if $I/O1_i$ ff clear
T2.4	Clear Chan & Fct Pri if $I/O1_i$ ff clear
T3.1	Int Req ⟶ Chan Pri if $I/O1_i$ ff clear
T3.2	Drop Chan Req ⟶ Chan Pri ✓
T3.4	Int Req ⟶ Fct Pri if $I/O1_i$ ff clear, clear Chan & Fct Pri if $I/O1_i$ ff clear
T4.1	Clear I/O Trans 2, clear Fct, Chan, & I/O Trans 1 if $I/O1_i$ ff clear
T4.2	Set $I/O1_f$ ff if $I/O1_i$ ff set ✓
	Set Int_i ff if (any Fct Pri ff set) · ($I/O1_i$ ff clear)
	I/O Trans 1 ⟶ I/O Trans 2 if $I/O1_i$ ff set ✓
	Pri ⟶ Fct, Chan, & I/O Trans 1 if $I/O1_i$ ff clear

TABLE 8.4-3. CHANNEL PRIORITY GATES, CHASSIS 1 AND 2

	Gates	Conditions to Output High Level
Chassis 1	10V13	Chan 7
	10V12	Chan 5 · $\overline{\text{chan 7}}$ · $\overline{\text{chan 6}}$
	10V11	Chan 3 · $\overline{\text{chan 7}}$ · $\overline{\text{chan 6}}$ · $\overline{\text{chan 5}}$ · $\overline{\text{chan 4}}$
	10V10	Chan 1 · $\overline{\text{chan 7}}$ · $\overline{\text{chan 6}}$ · $\overline{\text{chan 5}}$ · $\overline{\text{chan 4}}$ · $\overline{\text{chan 3}}$ · $\overline{\text{chan 2}}$
	12V11	Chan 5 or 7 selected
	12V10	Chan 3 or 7 selected
Chassis 2	10V03	Chan 6 · $\overline{\text{chan 7}}$
	10V02	Chan 4 · $\overline{\text{chan 7}}$ · $\overline{\text{chan 6}}$ · $\overline{\text{chan 5}}$
	10V01	Chan 2 · $\overline{\text{chan 7}}$ · $\overline{\text{chan 6}}$ · $\overline{\text{chan 5}}$ · $\overline{\text{chan 4}}$ · $\overline{\text{chan 3}}$
	10V00	Chan 0 · $\overline{\text{chan 7}}$ · $\overline{\text{chan 6}}$ · $\overline{\text{chan 5}}$ · $\overline{\text{chan 4}}$ · $\overline{\text{chan 3}}$ · $\overline{\text{chan 2}}$ · $\overline{\text{chan 1}}$
	12V01	Chan 4 or 6 selected
	12V00	Chan 2 or 6 selected
Conditions for chassis 9 and 10 are similar. Add 10_8 to channel numbers above.		

Even though each I/O chassis has function priority logic like that shown, only one function priority on each drawer will be presented with signals. This is because the two channel priorities on each drawer are inter-connected such that only one channel is selected between them. Then, only the function priority on the chassis of the selected channel will receive signals.

11Vg4 outputting a low level indicates that at least one of the function priority flip-flops is set or that a signal has been selected on its chassis. The two outputs of this logic to figure 9-51 will specify the type of selected signal. If 01Vg7 (EI flip-flop) outputs a high level, the selected type is either EF/OD or ID; since there is no EI type, and it has the highest function priority.

The high level output of 12Vg6 is prevented by its input from the EI flip-flop if an EI type is present. If the EI flip-flop is clear, 12Vg6 can output a high level if an ID type is selected. This selection is determined by 11Vg6. 11Vg6 outputs a low level if no EF/OD type is present or if an ID type is present and output was last. The output last indication from 51Vg4 actually means the Priority Alternator

flip-flop is set. This flip-flop is set whenever an OD or EF Acknowledge signal is sent and is cleared whenever an ID Acknowledge signal is sent. Both of these acknowledges are on this same chassis. This flip-flop then allows the function priority of EF/OD and ID type signals to be reversed depending upon the type of I/O operation performed last.

The outputs of the EI flip-flop, 12Vg6, and 12Vg4 then indicate the presence of a signal on this chassis and the type of the selected signal. Refer to table 8.4-4 for the signal type translation indication from this logic.

TABLE 8.4-4. FUNCTION PRIORITY SELECTION INDICATIONS

SELECTED SIGNAL TYPE	PRIORITY OUTPUTS	
	01Vg7 (EI ff)	12Vg6
EI	L	L
EF/OD	H	L
ID	H	H

6. Group Priority. Refer to logic diagrams, figure 9-50.

Group priority logic is comprised of gates 16V34, 17V34, 16V14, 15V34, 15V14, 14V14, and 13V14. This logic receives indications from each of the four function priorities, by way of the 12Vg4 inputs, as to whether or not these priorities contain a signal. A high level from a 12Vg4 gate indicates that a signal has been selected on its chassis. From this information, group priority simply selects the signal of higher channel number; which means it selects the I/O chassis that has the selected signal. The effect of chassis selection is to allow the outputs of the channel priority and function priority on the selected chassis to enter the function and channel translators. Refer to table 8.4-5 for the group priority selection indications.

The group priority outputs are combined by the I/O translator 1 to select one chassis.

7. Group Translator (I/O Translator 1). Refer to logic diagrams, figure 9-50.

The group translator logic is comprised of flip-flops 0XG07, 0XG08, and 0XG00 and their output gates. These flip-flops are set according to the selection made by group priority. Flip-flop 0XG00 is set by the odd channel gate signal. Refer to logic diagrams, figure 9-49 to see that this signal is dependent upon timing and the fact that group priority has selected an odd channel chassis.

Thus, 0XG00 being set indicates an odd chassis signal has been selected. Its clear state indicates an even chassis signal.

TABLE 8.4-5. GROUP PRIORITY SELECTION INDICATIONS

Group Priority Gates	Function Priority Gate Outputs upper-group chass 9 12V34	chass 10 12V24	lower group chass 1 12V14	chass 2 12V04	Meaning of Group Priority Gate Outputs
13V14 = L			H		Chassis 1 signal
14V14 = H		(L and	H)		(Chass 1 signal) · (no chass 10 signal)
15V14 = L	(H) or	(L and	H)		(Chass 9 signal) + (chass 1 signal · no chass 10 signal) - same as "odd group"
16V14 = L	(L) and	(H or	L)		(Chass 10 signal · no chass 9 signal) + (no chass 9 signal · no chass 1 signal). - same as (chass 10 signal) + (chass 2 signal)* - same as "even group"
15V34 = L	(H or	H)			(Chass 9 signal) + (chass 10 signal)
& 17V34 = L					- same as "upper group"
16V34 = L	(L or	L)			(No chass 9 signal) · (no chass 10 signal) - same as (chass 1 signal) + (chass 2 signal)* - same as "lower group"

*This chart assumes a signal to be present on at least one chassis. Therefore, if a gate is conditioned by the absence of signals on a chassis, it indicates the presence of a signal on another chassis.

The setting of flip-flop 0XG07 is dependent upon timing from the request gate signal and also the fact that group priority has selected an upper chassis (chassis 9 or 10). Thus, 0XG07 being set indicates an upper chassis signal has been selected. Its clear state indicates a lower chassis signal.

Flip-flop 0XG08 is set simply to indicate that a signal has been selected by priority and is present in the translator. The setting of this flip-flop is enabled by the same timing input as is 0XG07. The other enable to set 0XG08 is from 13V04. Notice that 13V04 is fed by the 12Vg4 gates of function priority. If any signal is present in any function priority, 13V04 outputs a low level. Thus, 0XG08 being set indicates a signal is present and enables the group translator gates (20G10, 20G11, 20G12, and 20G13). Refer to table 8.4-6 for the group translator indications.

TABLE 8.4-6. GROUP TRANSLATOR INDICATIONS

TRANSLATOR FLIP-FLOPS		INDICATIONS
0XG07	0XG00	
Clear	Clear	20G10 = H, chassis 2 selected (lower-given group)
Clear	Set	20G11 = H, chassis 1 selected (lower-odd group)
Set	Clear	20G12 = H, chassis 10 selected (upper-even group)
Set	Set	20G13 = H, chassis 9 selected (upper-odd group)

A signal is assumed to be present (flip-flop 0XG08 set).

8. Mode Translator (I/O Translator 1). Refer to logic diagrams, figure 9-50.

If the selected channel is odd, the position of the CHANNEL FUNCTION switch for that chassis is sensed. If the switch position is either dual, ESI, or ESA, the associated flip-flop is set. The Dual flip-flop is actually part of I/O Translator 2 and its setting is performed at a later time if the output of 11G09 is a low level. If the selected channel is even, single channel mode is enforced.

9. Channel Translator. Refer to logic diagrams, figure 9-51.

The channel translator is comprised of flip-flops 0XG02 and 0XG01 and their output gates. These flip-flops are set according to the selection made by channel priority on the chassis selected by group priority. The channel priority inputs are from the 12Vg- gates. Two of these inputs enter the channel translator as selected by the 16V34, 17V34, 13E00, and 14E00 inputs from group priority. Refer back to table 8.4-3 for the translation of the 12Vg- inputs.

The Channel Translator flip-flops actually indicate the selected channel of the selected chassis. When channel translation is combined with the Group Translator outputs, the complete channel number is described. Channel translation is performed by gates 20G00, 20G01, 20G02, and 20G03. Refer to table 8.4-7 for the channel translation indications.

10. Function Translation. Refer to logic diagrams, figure 9-51.

The function translator is comprised of flip-flops 0XG05 and 0XG04 and their output gates. These flip-flops are according to the selection made by function priority on the chassis selected by group priority. The function priority inputs are from the 12Vg6 gates and the EI function flip-flops (01Vg7 inputs). Two of these inputs enter the function translator as selected by the 16V34, 17V34, 13E00, and 14E00 inputs from group priority. Refer back to table 8.4-4 for the translation of the inputs from function priority.

Function translation is performed by gates 20G04, 20G06, and 20G07. Refer to table 8.4-8 for the function translation indications.

TABLE 8.4-7. CHANNEL TRANSLATOR INDICATIONS

TRANSLATOR FLIP-FLOPS		
0XG02	0XG01	INDICATIONS
Clear	Clear	20G00 = H; channel 0, 1, 10, or 11 selected
Clear	Set	20G01 = H; channel 2, 3, 12, or 13 selected
Set	Clear	20G02 = H; channel 4, 5, 14, or 15 selected
Set	Set	20G03 = H; channel 6, 7, 16, or 17 selected

TABLE 8.4-8. FUNCTION TRANSLATOR INDICATIONS

TRANSLATOR FLIP-FLOPS		
0XG05	0XG04	INDICATIONS
Clear	Clear	20G07 = H, EI function selected
Set	Clear	20G06 = H, EF/OD function selected
Set	Set	20G04 = H, ID function selected

The condition of 0XG05 clear and 0XG04 set is not used.

The function translator outputs are used to dictate the basic operation to be performed by the I/O logic in honoring the selected signal.

11. Data Request Sequence Timing. The honoring of a data request involves the use of memory under control of the I/O sequences. Single channel operation uses only the I/O1-sequence, one main memory cycle. In this case, the program sequences are disabled for 2 microseconds. Dual channel operation requires two main memory cycles to complete the transfers of both halves of a 36-bit word. The I/O1 and I/O2 sequences are used, and the program sequences are disabled for four microseconds.

The scan sequence is always running to detect and select I/O signals to be honored. If a signal is detected, it is gated into the priority logic and then into the I/O translator. If the scan sequence selects a data request signal to be honored, as the signal information is placed in the I/O translator the $I/O1_i$ flip-flop is set. The next 2 microsecond period beginning with T1.1 time is then considered to be the I/O1-sequence, and the requested I/O service is performed.

There is no scan flip-flop; rather, every main timing cycle is considered to be a scan sequence which is testing all signal entrance logic for I/O requests. Therefore during each sequence of each instruction, the possibility of I/O operation is being tested by the priority logic. The detection of a data request activates the I/O1-sequence during which the program is effectively halted. The amount of delay to the program is determined by whether the I/O service involves single or dual

channel operation. This delay period will be extended by consecutive I/O-sequences if data requests are continually selected by consecutive scan sequences. Since the scan sequence also runs during the I/O1-sequence, it is conceivable that each main timing cycle could be under control of an I/O sequence and the program would not progress. In a practical situation, however, there would not be such an extended period of continual I/O activity because of the limited speed of peripheral devices and the high speed of the computer in handling an I/O request.

The scan sequence can be effectively disabled under certain conditions by preventing the setting of the $I/O1_i$ flip-flop. A direct disable to this flip-flop is present if f = 57 and the R1-sequence is in control or if f = 20-23 and the first R1-sequence is in control (without the R2-sequence). These conditions also prevent the setting of the $I/O2_i$ flip-flop. Refer to logic diagrams, figure 9-13 for this disable.

An indirect disable to the setting of the $I/O1_i$ flip-flop is the prevention of the Data Req Function Priority command. This disable can exist if the $I/O1_f$ flip-flop is set and the I/O service being performed involves dual channel operation or ESI/ESA which has a terminate condition. Both of these conditions require the I/O2-sequence to follow the I/O1-sequence. This indirect disable to the I/O1-sequence also exists if the CDM-sequence is to be run. Refer to logic diagrams, figure 9-49 for this disable.

The disable to the program sequences by the I/O-sequences occurs between the initial and final portions, beginning at T2.2 time. During the set time of either the $I/O1_i$ or $I/O2_i$ flip-flop, the outputs of the I_f, $R1_f$, $R2_f$, and W_f flip-flops are disabled; and the setting of the I_i, $R1_i$, $R2_i$, W_i, $B1_i$, and $B2_i$ flip-flops is inhibited. During the set time of either the $I/O1_f$ or $I/O2_f$ flip-flops, the advancing of the sequence final flip-flops is prevented. That is, the setting and clearing of the I_f, $R1_f$, $R2_f$, W_f, $B1_f$, and $B2_f$ flip-flops is inhibited.

These disable conditions are such that any sequence of any instruction may effectively be halted after the operations controlled by the sequence initial flip-flop are executed and those operations caused by the sequence final flip-flop will not be performed until after the interrupting I/O service is performed. The halting point can be considered to be at T2.2 time, the setting time for the program sequence final flip-flop as well as the setting time for the $I/O1_i$ and $I/O2_i$ flip-flops.

Refer to figures 8.4-3, 8.4-4, and 8.4-5 for timing diagrams illustrating possible relationships of the program and I/O-sequences. These diagrams show the I/O operations interrupting the I-sequence. Realize that any sequence can be disturbed in the same manner.

8.4-5. SUMMARY

Channel, function, and group priorities select the highest channel number which has a signal present and then one type of signal on that channel to be honored. The translators have flip-flops to hold that selected signal information and translation gates to inform the rest of the I/O logic as to the type of operation to be performed and the channel involved.

Operations concerning the real time clock and special interrupts are discussed in later sheets.

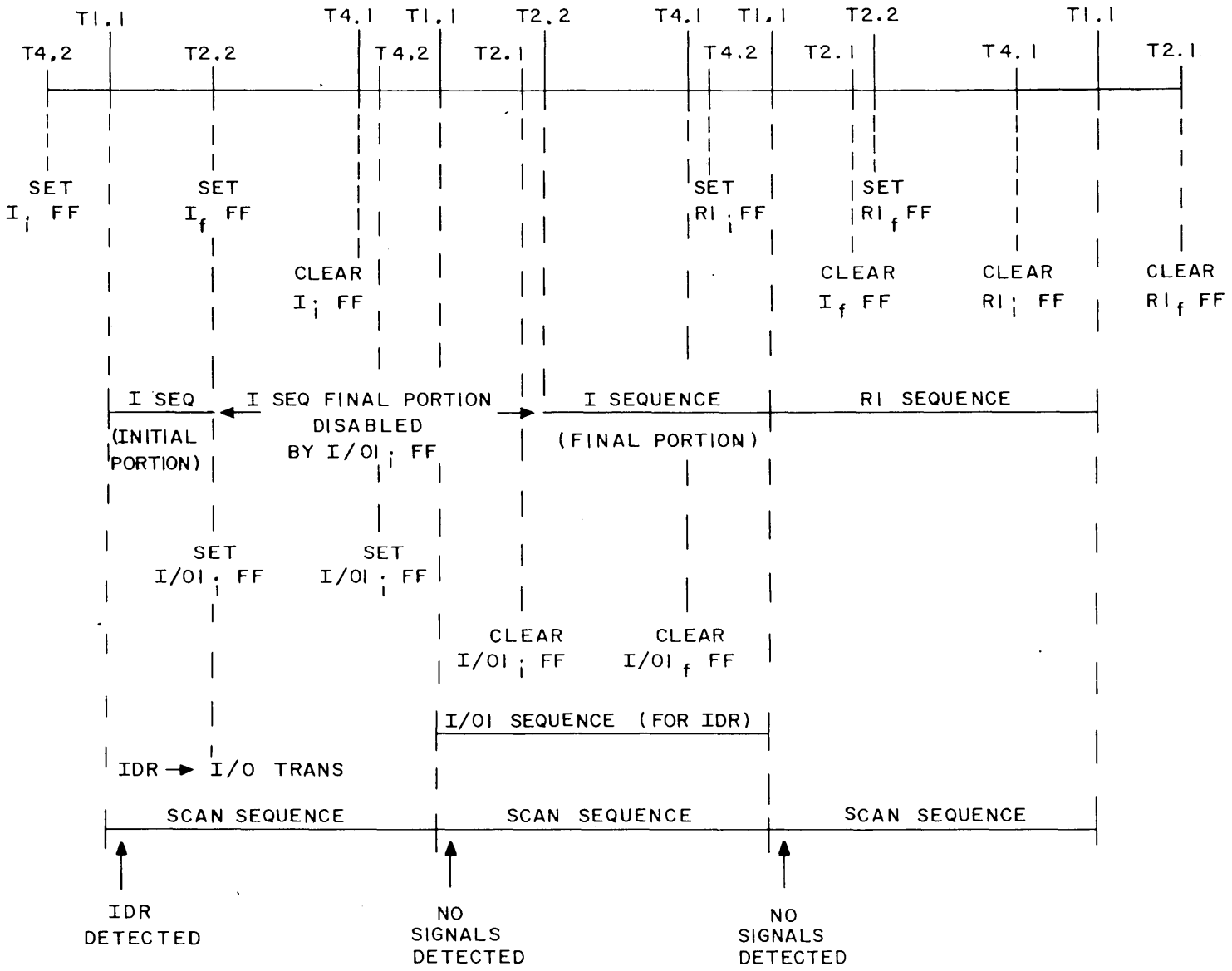


Figure 8.4-3. Single Data Request, Single Channel, Timing Example

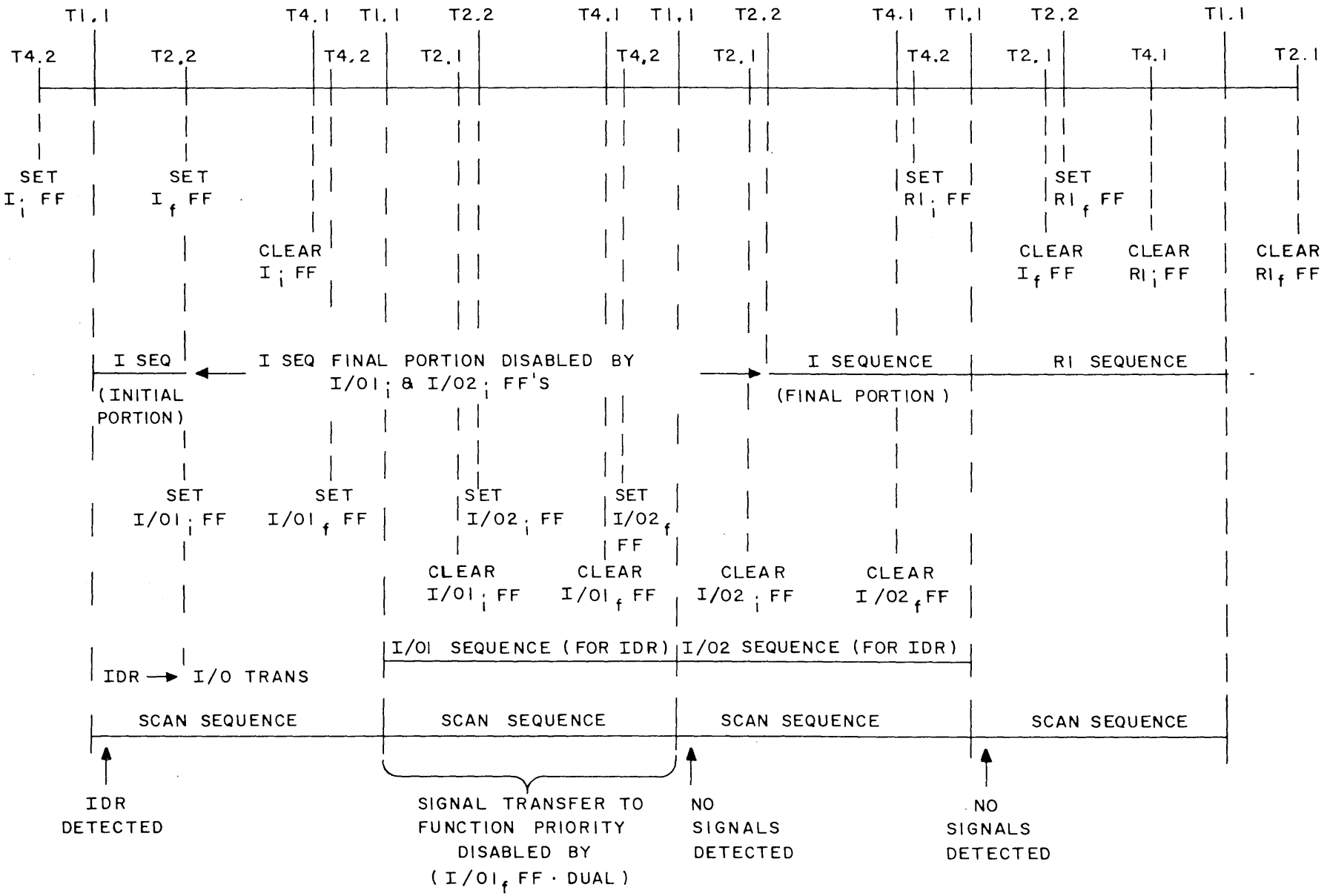


Figure 8.4-4. Single Data Request, Dual Channel, Timing Example

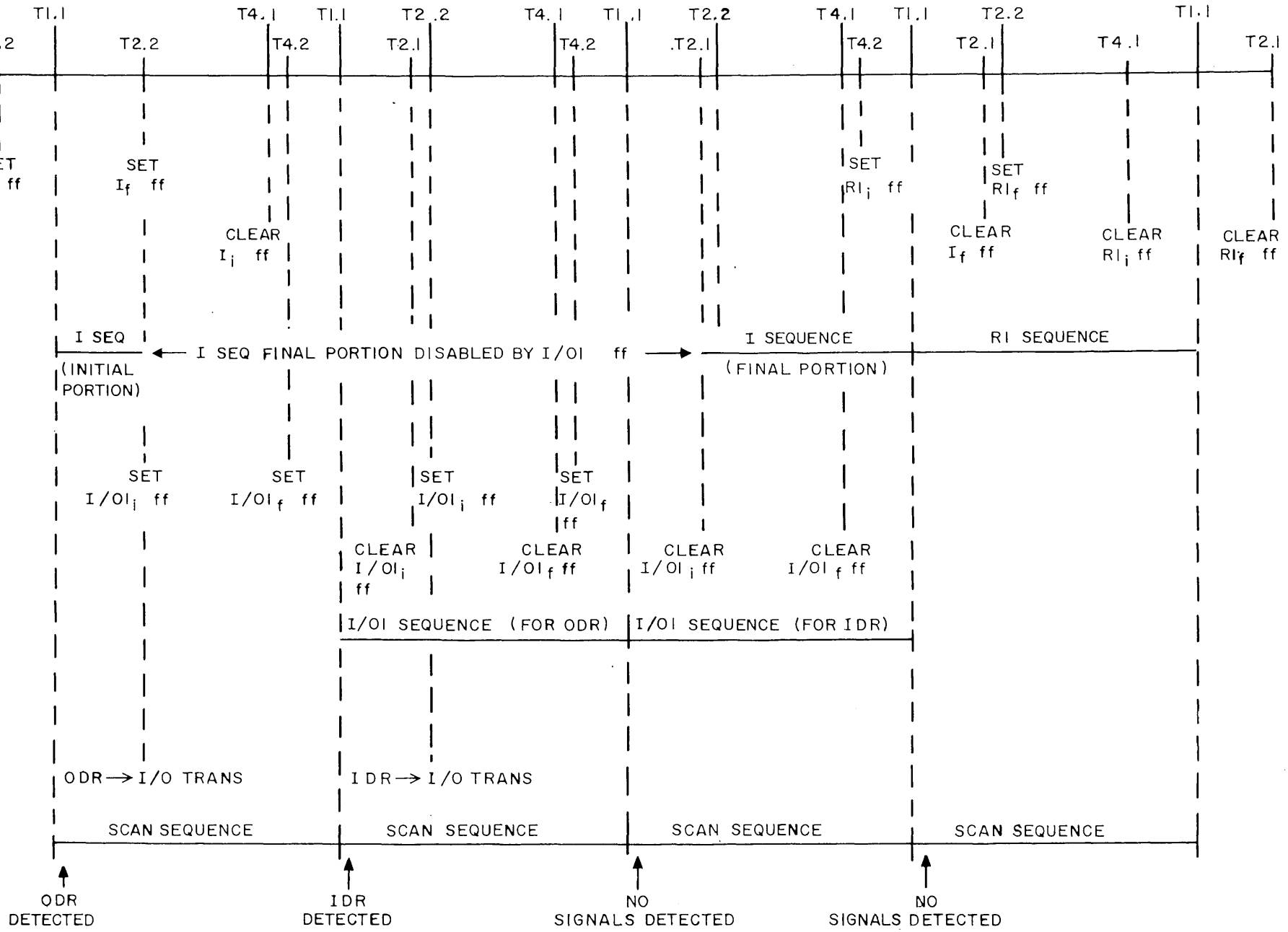


Figure 8.4-5. Multiple Data Request, Single Channel, Timing Example

NAME: _____

8.4-6. STUDY QUESTIONS

a. Refer to logic diagrams, figures 9-50 and 9-51.

Indicate below the states of flip-flops 0XG08, 0XG07, 0XG05, 0XG04, 0XG00, 0XG02, and 0XG01 assuming that an ODR signal on channel 13_8 was selected.

0XG08	0XG07	0XG02	0XG01	0XG00	0XG05	0XG04
_____	_____	_____	_____	_____	_____	_____

b. Refer to logic diagrams, figure 9-50.

Indicate below the logic level outputs of the group priority gates assuming that the function priorities on chassis 10 and 2 contain signals.

13V14 = _____ 15V14 = _____ 15V34 = _____ 16V34 = _____

14V14 = _____ 16V14 = _____ 17V34 = _____

c. Given: 11V10 grounded output (logic diagrams, figure 9-65).

Describe the effect that this malfunction would have upon I/O operations.

d. Given: 11G02 grounded output (logic diagrams, figure 9-51).

Describe the effect that this malfunction would have upon I/O operations if the channel selected by channel and group priorities were number 7.

SECTION 8 - INPUT/OUTPUT SECTION

8.5. INPUT DATA REQUEST OPERATIONS

8.5-1. OBJECTIVES

To present the detailed theory of operation involved in input data request honoring.

8.5-2. INTRODUCTION

The input data request is a signal from an external device which requests that the computer accept an inputted data word.

8.5-3. REFERENCES

a. UNIVAC 1219 Technical Manual, Volume I, Paragraph 4-3.

b. UNIVAC 1219 Technical Manual, Volume II, Section 9 (logic diagrams).

8.5-4. INFORMATION

a. Input Data Operations Without ESI or ESA.

1. General Description.

a) Single Channel Operation. The input data request (IDR) from an external device is honored by the computer only if the particular channel is input active. The Input Active (ID) flip-flop is set during the execution of f = 50:01, 50:11.

If the priority logic selects an IDR signal, the current address control word (CACW) is obtained from the control memory address 00061_8 + 2 x channel # if the channel is 0-7 or 00261_8 + 2 x channel # if the channel is $10\text{-}17_8$. The terminal address control word (TACW) is obtained from the control memory address 00060_8 + * 2 x channel # if the channel is $10\text{-}17_8$. The 18-bit input data word is stored at the address specified by the $CACW_{15\text{-}0}$.

The $CACW_{15\text{-}0}$ is modified by 1 as determined by the value in bit position 17 of the CACW and is restored in control memory. $CACW_{17}$ determines whether the buffer is to be forward or backward. In a forward buffer ($CACW_{17} = 0_2$), the data words are stored at consecutively incremented addresses as the IDR signals occur. In this case, the $CACW_{15\text{-}0}$ is modified by +1 after each data word is stored. In a backward buffer ($CACW_{17} = 1_2$), the data words are stored at consecutively decremented addresses as the IDR signals occur. In this case, the $CACW_{15\text{-}0}$ is modified by -1 after each data word is stored.

Refer to tables 8.5-1 and 8.5-2 for buffer examples.

TABLE 8.5-1. SINGLE CHANNEL FORWARD ID BUFFER
IACW = 005000, TACW = 005003

DATA STORAGE ADDRESS ($CACW_{15-0}$)	WORD INPUT SEQUENCE
05000	1
05001	2
05002	3
05003	4

TABLE 8.5-2. SINGLE CHANNEL BACKWARD ID BUFFER
IACW = 405003, TACW = 405000

DATA STORAGE ADDRESS ($CACW_{15-0}$)	WORD INPUT SEQUENCE
05000	4
05001	3
05002	2
05003	1

The TACW is not modified but is compared with the CACW. If their 16 least significant bits are equal before the modification of the CACW, the current input data word being processed completes the buffer. To terminate the buffer, the ID Active flip-flop for the particular channel is cleared which disables the recognition of future IDR signals. At this buffer termination time, the value in bit position 16 of the CACW is sensed. If $CACW_{16} = 1_2$, an input monitor interrupt signal is to be generated to indicate to the program the termination; thus, the ID Monitor flip-flop for the particular channel is set. The resulting ID Monitor signal is handled later by the computer as a separate request for input/output service and is discussed in a later sheet.

The ID-acknowledge signal is sent on the same channel which carried the request.

If $TACW_{17} = 1_2$, continuous data mode is specified. This bit is sensed when the buffer is terminated and allows the special continuous data mode (CDM) sequence to be activated. This sequence prevents the clearing of the ID Active flip-flop and reloads the TACW and CACW locations in control memory with new buffer control words. CDM operations are presented in a later sheet.

b). Dual Channel Operation. The dual channel mode is instated for a particular channel by positioning the channel function switch on its drawer to the dual position. If dual channel has been selected, the IDR signal is actually requsting that two separate 18-bit data words be accepted, one on the IDR channel which must be odd for dual channel operation, and the other word on the next lower channel which is even numbered. The input operations for the second data word are identical to those for the first word. The second word, therefore, is stored in memory at the address specified by the $CACW_{15-0}$ which has been modified by 1 during the storage of the first word.

The type of buffer (forward or backward) determines which of the two words is stored first. The 18-bit data word from the odd channel (IDR channel) is stored first if the buffer is forward. The even channel word is stored first if the buffer is backward. Refer to tables 8.5-3 and 8.5-4 for buffer examples assuming the IDR signals occur on channel 3.

TABLE 8.5-3. DUAL CHANNEL FORWARD ID BUFFER ON CHANNELS 2, 3
IACW = 005000, TACW = 005003

	DATA STORAGE ADDRESS ($CACW_{15-0}$)	INPUTTING CHANNEL	WORD INPUT SEQUENCE	IDR SEQUENCE
storage of 1st 36-bit word	05000	3	1	1st IDR
	05001	2	2	
storage of 2nd 36-bit word	05002	3	3	2nd IDR
	05003	2	4	

TABLE 8.5-4. DUAL CHANNEL BACKWARD ID BUFFER ON CHANNEL 2, 3
IACW = 405003, TACW = 405000

	DATA STORAGE ADDRESS ($CACW_{15-0}$)	INPUTTING CHANNEL	WORD INPUT SEQUENCE	IDR SEQUENCE
storage of 2nd 36-bit word	05000	3	4	2nd IDR
	05001	2	3	
storage of 1st 36-bit word	05002	3	2	1st IDR
	05003	2	1	

The TACW and CACW are compared during each 18-bit word input operation. Therefore, the clearing of the ID Active flip-flop and the setting of the ID Monitor flip-flop can occur during either the first or second word storage operation. The TACW and IACW should be chosen to provide a buffer for an exact multiple of two 18-bit words such that the buffer will always terminate as a result of the second word storage operation from a particular IDR.

The ID-acknowledge signal is sent on the odd channel which carried the IDR. The acknowledge occurs after the second word of the IDR operations has been accepted.

2. Detailed Analysis.

a) Data Flow Block Diagram.

1) Single Channel Operation. Refer to figure 8.5-1 for a block diagram description of the single channel ID operation.

The scan sequence is the main timing cycle during which the $I/O1_i$ flip-flop is set upon detection and selection of an IDR signal. During this sequence, the CACW is obtained from control memory at the address 00061_8 + 2 channel # or 00261_8 + 2 x channel # depending upon the channel. The address of either 00060_8 + 2 x channel # or 00260_8 + 2 x channel # is formulated in the I/O translator which is set according to the priority selection. SO_{00} is set to 1_2 to make the address odd.

Another control memory reference is used to obtain the TACW. The TACW and CACW are compared from ZO and B. The result of the comparison is recorded by the Terminate flip-flop. If $TACW_{15-0} = CACW_{15-0}$, this IDR will be the last to be honored.

The B-network is used to modify the $CACW_{15-0}$ as determined by $CACW_{17}$. The value of this bit is recorded by the B $\pm$ 1 flip-flop.

The I/O1-sequence is used to store the 18-bit input data word. The memory reference uses the address specified by the unmodified $CACW_{15-0}$ from B. A control memory reference is used at T1.1 time to store the modified CACW value for use during the next input operation.

The remaining I/O1-sequence operations are used only if the operation is dual channel.

2) Dual Channel Operation. Refer to figures 8.5-1 and 8.5-2.

The scan and I/O1-sequence are used as for single channel operation. The last portion of the I/O1-sequence and the I/O2-sequence are used to perform the same operations executed during the last portion of the scan sequence and the first portion of the I/O1-sequence. These operations input the second word and store it at the first word storage address $\pm$ 1.

b) Essential Commands. Refer to table 8.5-5 for a sequential list of essential scan, I/O1, and I/O2-sequence events. Develop the commands shown by referring to the proper enable pages in the logic diagrams.

The events concerning priority selection are as discussed previously and are not shown. The events concerning the continuous data mode sequence, which is initiated as a result of the setting of the Continuous Data Request flip-flop, are analyzed in a later sheet. The ID acknowledge timing is discussed later in this sheet.

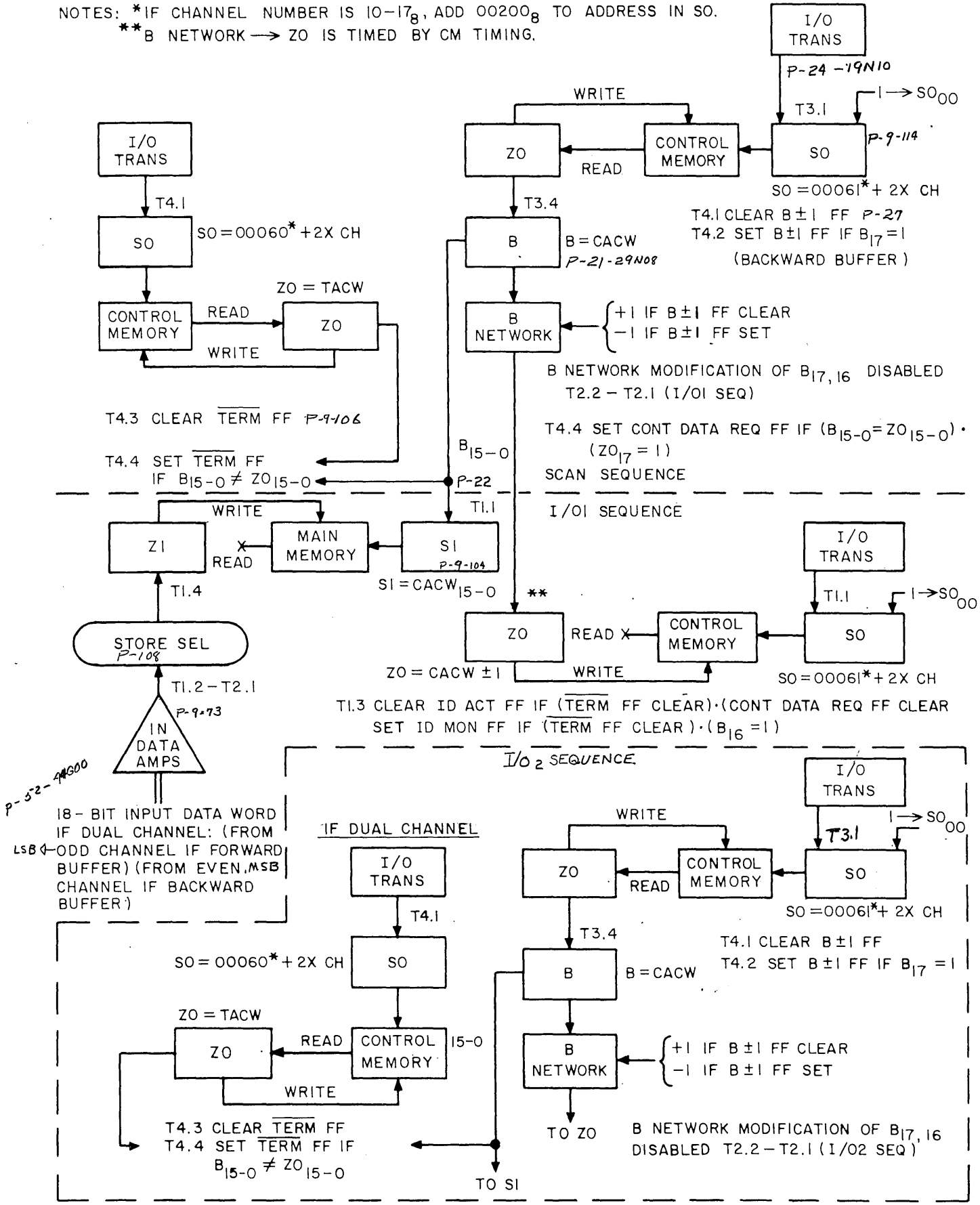
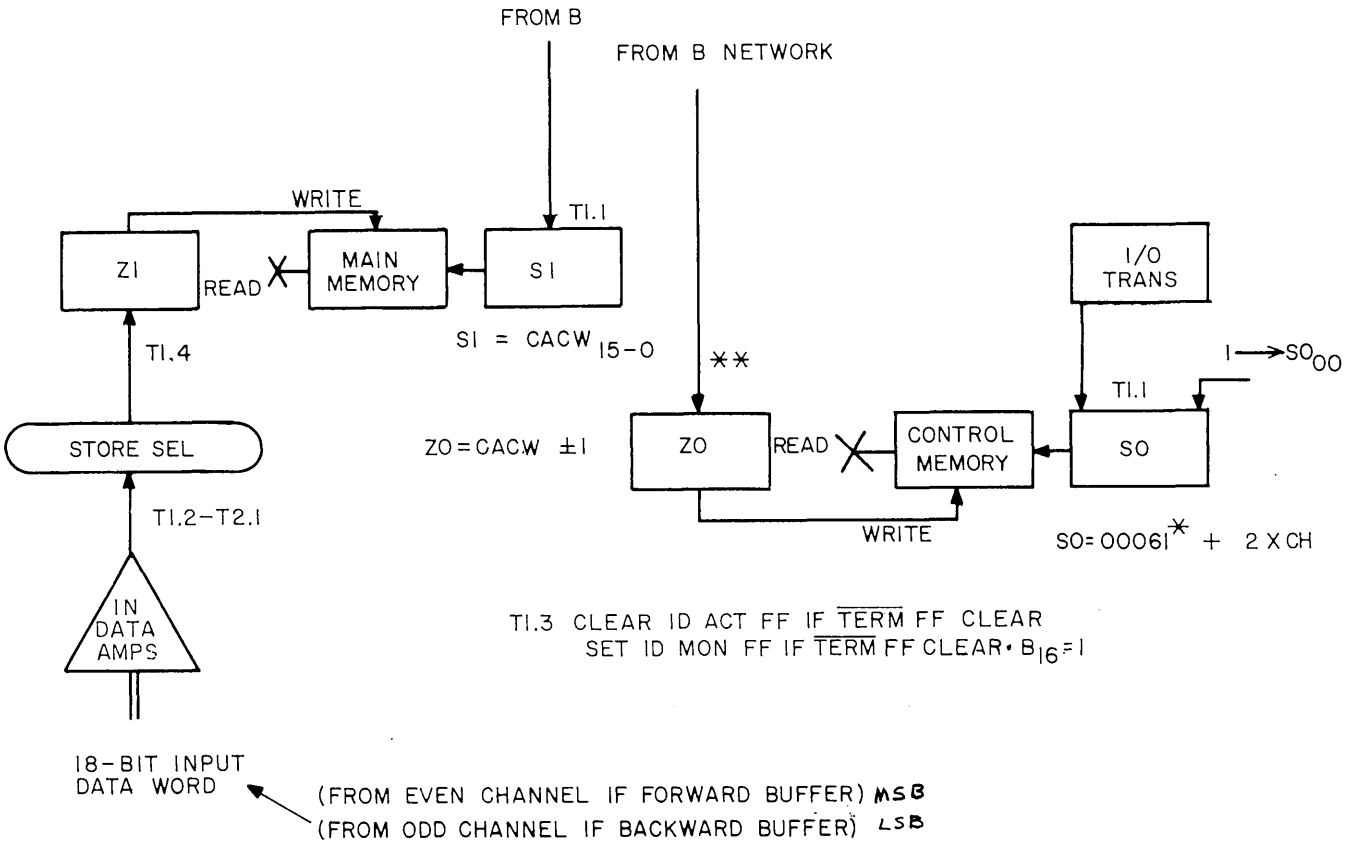


Figure 8.5-1. Scan and I/O1-Sequence Data Flow for Single or Dual Channel ID Operations



NOTE: * IF CHANNEL IS $10-17_8$, ADD 00200_8 TO ADDRESS IN SO

** B NETWORK ⟶ ZO IS TIMED BY CM TIMING

Figure 8.5-2. I/O2-Sequence Data Flow for Dual Channel ID Operations

c) ID-Acknowledge Timing. The ID-acknowledge signal is sent after the data word storage operation. In the case of dual channel operation, the acknowledge is sent after both words have been inputted. The purpose of this signal is to acknowledge the receipt of data. The communicating external device usually uses the acknowledge to time the dropping of its IDR signal. The ID-acknowledge signal is sent for a time duration of 2.75 microseconds and 14.75 microseconds for fast and slow interface, respectively. The ID-Acknowledge Generation flip-flop determines the acknowledge duration of its set time.

Refer to tables 8.5-6 and 8.5-7 for the timing analysis of the acknowledge for fast and slow interface, respectively. Develop these tables by referring to the proper enable pages in the logic diagrams.

TABLE 8.5-5. SCAN, I/01, AND I/02 SEQUENCE ESSENTIAL COMMANDS FOR SINGLE OR DUAL CHANNEL ID OPERATIONS

TIME NOTATION	COMMANDS
	SCAN SEQUENCE
T2.1	Clear Fct, Chan, & I/O Trans 1
T2.2	Set $I/01_i$ ff, Pri→Fct, Chan, & I/O Trans 1, disable B $Network_{17,16}$
T3.1	I/O Trans→SO, 1→SO_{00}, Init CM, clear ZO
T3.2	Clear B
T3.4	ZO→B*
T4.1	I/O Trans→SO, Init CM, clear ZO, clear B ± 1 ff (+1→B Network) clear I/O Trans 2
T4.2	Set $I/01_f$ ff, set B ± 1 ff (-1→B Network) if B_{17} = 1
	I/O Trans 1→I/O Trans 2 (set Dual ff if dual chan selected), clear all ID Ack Reg ff's
T4.3	Clear Pri Alternator ff, set ID Ack Reg ff, set $6XL_g0$ ff clear $\overline{Term}$ ff, clear ID Req ff
T4.4	Set $\overline{Term}$ ff if $B_{15-0} \neq ZO_{15-0}$, disable CM→ZO
	Clear S1, disable Mem→Z1, set Cont Data Req ff if $(B_{15-0} = ZO_{15-0})\cdot(ZO_{17} = 1)$
	I/01 SEQUENCE
T1.1	I/O Trans→SO, 1→SO_{00}, Init CM, B→S1, Init Memory
T1.2	Input Data Amps→Store Sel**
T1.3	Clear Z1, clear ID Act ff if ($\overline{Term}$ ff clear)·(Cont Data Req ff clear)
	Set ID Mon ff if ($\overline{Term}$ ff clear)·(B_{16} = 1)
T1.4	Store Sel→Z1, set ID Ack Gener ff if Dual ff clear
T2.1	Clear $I/01_i$ ff, drop Input Data Amps→Store Sel, drop disable B $Network_{17,16}$
T2.2	Set $I/02_i$ ff if Dual ff set, disable B $Network_{17,16}$ if $I/02_i$ ff set
T2.4	Drop disable Mem→Z1, drop disable CM→ZO
The following occurs if $I/02_i$ ff set (dual channel) excepting, "clear $I/01_f$ ff"	
T3.1	I/O Trans→SO, 1→SO_{00}, Init CM, clear ZO
T3.2	Clear B
T3.4	ZO→B*
T4.1	Clear $I/01_f$ ff, I/O Trans→SO, Init CM, clear ZO
	Clear B ± 1 ff (+1→B Network), clear I/O Trans 2
T4.2	Set $I/02_f$ ff, set B ± 1 ff (-1→B Network) if B_{17} = 1
	I/O Trans 1→I/O Trans 2 (set Dual ff), clear all ID Ack Reg ff's
T4.3	Set ID Ack Reg ff, set 6XLg0 ff, clear $\overline{Term}$ ff
T4.4	Set $\overline{Term}$ ff if $B_{15-0} \neq ZO_{15-0}$, disable CM→ZO
	Clear S1, disable Mem→Z1, set Cont Data Req ff if $(B_{15-0} = ZO_{15-0})\cdot(ZO_{17} = 1)$
	I/02 SEQUENCE (if dual channel)
T1.1	I/O Trans→SO, 1→SO_{00}, Init CM, Clear ZO
	B→S1, Init Memory
T1.2	Input Data Amps→Store Sel**
T1.3	Clear Z1, clear ID Act ff if ($\overline{Term}$ ff clear)·(Cont Data Req ff clear)·
	Set ID Mon ff if $\overline{Term}$ ff clear · B_{16} = 1
T1.4	Store Sel→Z1, set ID Ack Gener ff
T2.1	Clear $I/02_i$ ff, drop Input Data Amps→Store Sel
	Drop disable B $Network_{17,16}$
T2.4	Drop disable Mem→Z1, drop disable CM→ZO
T4.1	Clear $I/02_f$ ff

*B Network →ZO is timed by CM timing.

**If operation is in dual channel mode, the channel inputted to Store Sel is as follows.

	I/01 Seq.	I/02 Seq.
B ± 1 ff clear (forward buffer)	odd	even
B ± 1 ff set (backward buffer)	even	odd

TABLE 8.5-6. ID-ACKNOWLEDGE TIMING FOR FAST INTERFACE

TIME NOTATION	ACTION
T4.2	Clear all ID Ack Reg ff's P-9-68
T4.3	Set ID Ack Reg ff, set 6XLg0 ff P-9-70
T1.4	Set ID Ack Gener ff, send ID-acknowledge signal
.85 µs	
2.725 µs T3.3	Clear 6XLg0 ff
1.875 µs	
T3.2	Clear ID Ack Gener ff, drop ID-acknowledge signal

TABLE 8.5-7. ID-ACKNOWLEDGE TIMING FOR SLOW INTERFACE

TIME NOTATION	ACTION
T4.2	Clear all ID Ack Reg ff's
T4.3	Set ID Ack Reg ff, set 6XLg0 ff
T1.4	Set ID Ack Gener ff, send ID-acknowledge signal
T1.4	
T1.4	
12 µs delay of 61Lg2 T1.4	
T1.4	
14.725 µs T1.4	
T1.4	Set 6XLg3 ff
.85 µs	
T3.3	Clear 6XLg0 ff
1.875 µs	
T3.2	Clear ID Ack Gener ff, drop ID-acknowledge signal
4 µs delay of 60Lg2 T3.2	
T3.2	Clear 6XL3 ff

b. Input Data Operations With ESI.

1. General Description. The externally specified index mode (ESI) is instated for a particular channel by positioning the Channel Function switch on its drawer to the ESI position. When in effect, it alters the operations which honor an IDR on its channel. Two channels are used together, the one on which the IDR is sent which must be odd numbered and the next lower even numbered channel. Instead of the TACW and CACW being obtained from their normal addresses in control memory, their control memory origins are specified by the requesting external device. Refer to figure 8.5-3 for the input word format.

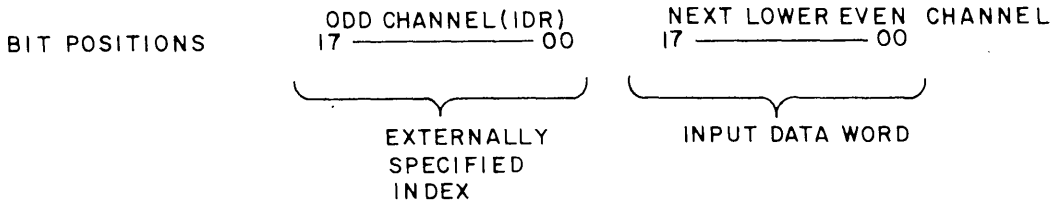


Figure 8.5-3. ESI Input Word Format for IDR Operations

The IDR signal must appear on an odd channel. The input word on that channel is accepted as the control memory address of the TACW. The CACW is obtained from the next consecutively higher address.

The $CACW_{15-0}$ is modified by ± 1 depending upon its bit position 17. The data word on the even channel is stored at the address specified by the unmodified $CACW_{15-0}$. If the $TACW_{15-0} = CACW_{15-0}$ (unmodified), the buffer is terminated and the input monitor interrupt signal is generated if $CACW_{16} = 1$. The ID-acknowledge signal is sent on the odd channel.

The ESI mode of operation is useful if more than one piece of equipment is connected to a common channel by means of a multiplexing device. Each piece of equipment can indirectly specify its data storage locations in memory by means of the externally specified index which could be generated in the multiplexer as it handled each data word. Each equipment would have a unique index value. The multiplexer must occupy an odd and the next lower even channels.

Upon buffer termination, the externally specified index is also stored in main memory at the address $0016l_8$ + 2 x channel # if the channel is 0-7 or $0036l_8$ + 2 x channel # if the channel is $10\text{-}17_8$. By inspecting the content of this address, the program could determine which of the equipment on that channel made the last input and made the channel inactive.

2. Detailed Analysis.

a) Data Flow Block Diagram.

1) Not Buffer Termination. Refer to figure 8.5-5 for a block diagram description of the ID operations with ESI.

Data flow is similar to that for normal input operations. The exceptions are that the control memory addresses for the CACW and TACW are from the odd numbered channel. The externally specified index is used in SO at T4.1 time to obtain the TACW. The transfer into SO involves a one place left shift. Realize also that SO has no bit position 06. The inputted bits 05 and 06 are therefore placed in SO_{07} and SO_{08}, respectively. Nothing is placed in SO_{00} such that the address of the TACW is always even. Refer to figure 8.5-4 for a description of the ESI transfer to SO.

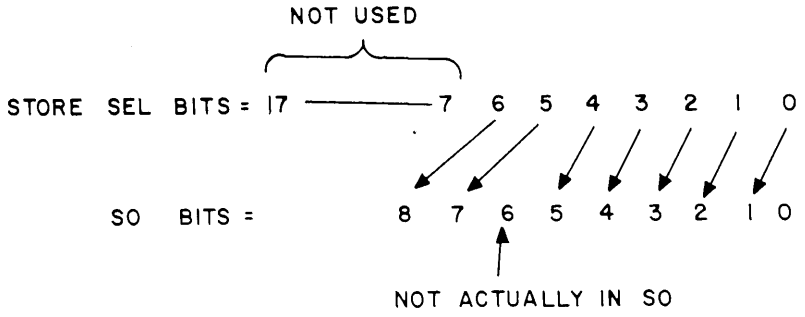


Figure 8.5-4. Store Select L1 $\longrightarrow$SO Transfer for ESI

The address of the CACW is formulated by setting SO_{00} to 1_2 in addition to the left shift input. The CACW is therefore obtained from the next consecutive higher address. All other I/O1-sequence operations are the same as those for normal input operations. The CACW is modified by ± 1 and is restored by another control memory reference. The input data word on the even channel is stored at the address specified by the unmodified $CACW_{15-0}$.

The termination and monitor interrupt capabilities are the same as for normal input operations. The Continuous Data Mode Sequence is initiated upon buffer completion if the $TACW_{17} = 1_2$. However, it is of no use in reloading the address control words, since this sequence always reloads the normal control memory address of the CACW and TACW.

That is, it will place new control words in control memory addresses 00060_8 + 2 x channel # and 00061_8 + 2 x channel # or 00260_8 + 2 x channel # and 00261_8 + 2 x channel #, depending upon the channel number. These addresses are not used since the external device specifies the origins of the CACW and TACW. However, if the Continuous Data Request flip-flop is set, it will prevent deactivating the buffer (the ID Active flip-flop will not be cleared).

2) Buffer Termination. Refer to figure 8.5-6 for the I/O2-sequence operations executed upon buffer termination.

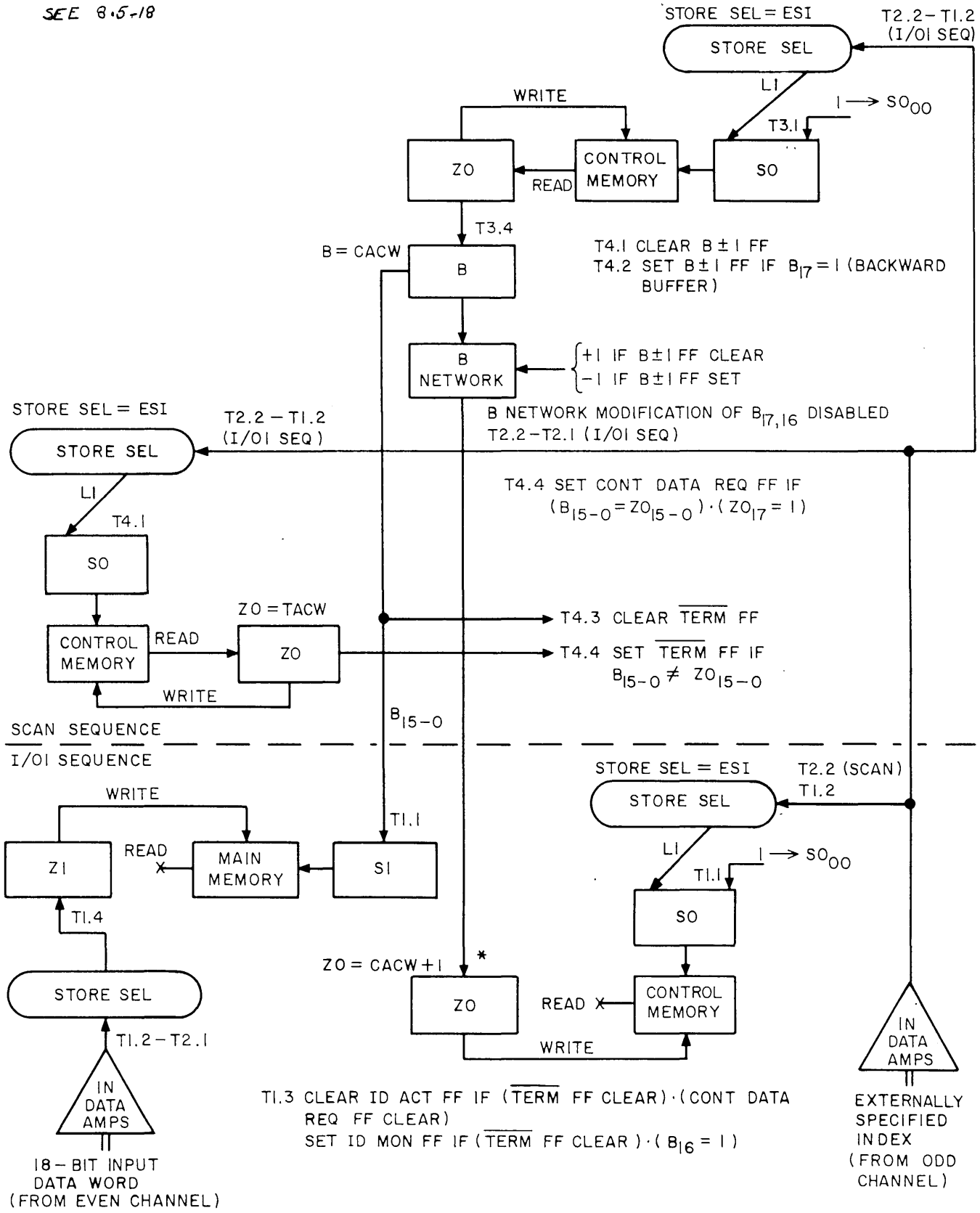
The I/O2-sequence is initiated if the TACW equals the unmodified CACW. It uses a memory reference to store the externally specified index in memory for use by the program.

b) Essential Commands. Refer to table 8.5-8 for a sequential list of scan, I/O1 and I/O2-sequence events. Develop the commands shown by referring to the proper enable pages in the logic diagrams.

TABLE 8.5-8. SCAN, I/10, AND I/02 SEQUENCE ESSENTIAL COMMANDS FOR ID OPERATIONS WITH ESI

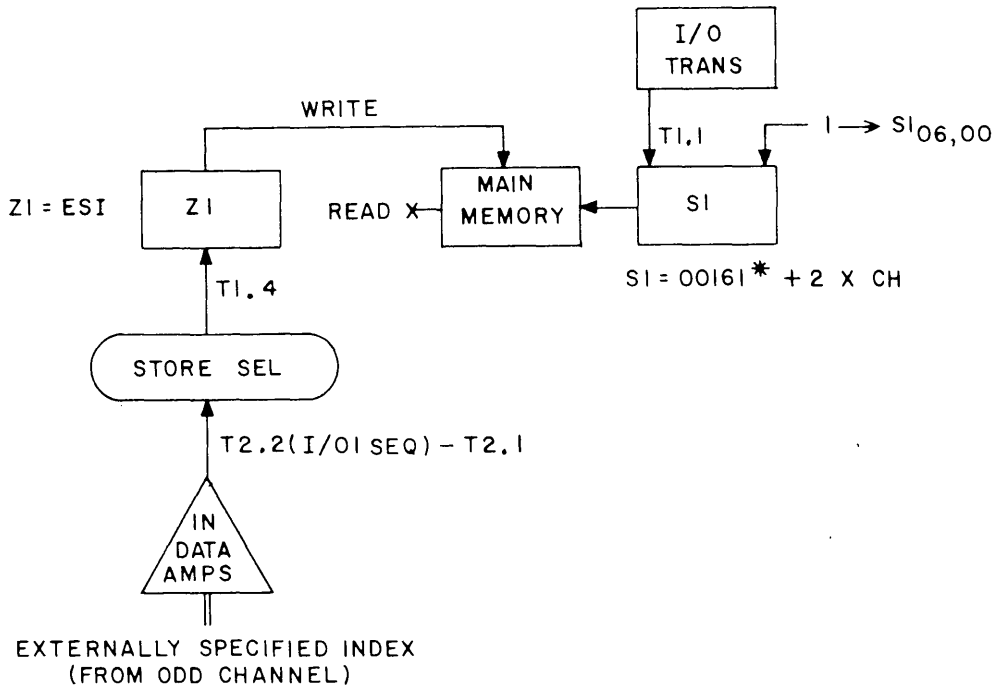
TIME NOTATION	COMMANDS
	SCAN SEQUENCE
T2.1	Clear Fct, Chan, & I/O Trans 1
T2.2	Set $I/01_i$ ff, Pri→Fct, Chan, & I/O Trans 1 (set ESI ff), disable B $Network_{17,16}$
	Odd chan Input Data Amps→Store Sel
T3.1	Store Sel L1→SO (ESI), 1→SO_{00}, Init CM, clear ZO
T3.2	Clear B
T3.4	ZO→B*
T4.1	Store Sel L1→SO (ESI), Init CM, clear ZO, clear I/O Trans 2
	Clear B ± 1 ff (+1→B Network)
T4.2	Set $I/01_f$ ff, set B ± 1 ff (-1→B Network) if $B_{17} = 1$
	I/O Trans 1→I/O Trans 2, clear all ID Ack Reg ff's
T4.3	Clear ID Req ff, set ID Ack Reg ff, set 6XLg0 ff, clear Pri Alternator ff, clear $\overline{\text{Term}}$ ff
T4.4	Set $\overline{\text{Term}}$ ff if $B_{15-0} \neq ZO_{15-0}$, disable CM→ZO, disable Mem→Zl, clear S1
	Set Cont Data Req ff if $(B_{15-0} = ZO_{15-0})\cdot(ZO_{17} = 1)$
	I/01 SEQUENCE
T1.1	Store Sel L1→ SO (ESI), 1→SO_{00}, Init CM, clear ZO
	B→S1, Init Memory
T1.2	Drop odd chan Input Data Amps→Store Sel, even chan Input Data Amps→Store Sel
T1.3	Clear Z1, clear ID Act ff if ($\overline{\text{Term}}$ ff clear)·(Cont Data Req ff clear)
	Set ID Mon ff if ($\overline{\text{Term}}$ ff clear)·$(B_{16} = 1)$
T1.4	Store Sel→Z1, set ID Ack Gener ff if $\overline{\text{Term}}$ ff set
T2.1	Clear $I/01_i$ ff, drop even chan Input Data Amps→Store Sel
	Drop disable B $Network_{17,16}$
T2.2	Set $I/02_i$ ff if $\overline{\text{Term}}$ ff clear, odd chan Input Data Amps→Store Sel if $I/02_i$ ff set
T2.4	Drop disable Mem→Z1, drop disable CM→ZO
T4.1	Clear $I/01_f$ ff
The following occurs if $I/02_i$ ff set (terminate)	
T4.2	Set $I/02_f$ ff, clear all ID Ack Reg ff's
T4.3	Set ID Ack Reg ff, set 6XLg0 ff
T4.4	Disable Mem→Z1, Clear S1
	I/02 SEQUENCE (if terminate)
T1.1	I/O Trans→S1, 1→$S1_{00}$, 1→$S1_{06}$, Init Memory
T1.3	Clear Z1
T1.4	Store Sel→Z1, set ID Ack Gener ff
T2.1	Clear $I/02_i$ ff, drop odd chan Input Data Amps→Store Sel
T2.4	Drop disable Mem→Z1
T4.1	Clear $I/02_f$ ff

*B Network→ZO is timed by CM timing.



NOTE: *B NETWORK ⟹ ZO IS TIMED BY CM TIMING.

Figure 8.5-5. Scan and I/O1-Sequence Data Flow for ID Operations with ESI



NOTE: * IF CHANNEL IS $10-17_8$, ADD 00200_8 TO ADDRESS IN SI.

Figure 8.5-6. I/O2-Sequence Data Flow for Terminate ID Operations With ESI

The events concerning priority selection are as previously discussed and are not shown. The events concerning the continuous data mode sequence are analyzed in a later sheet. The ID-acknowledge timing is the same as for normal input operations.

c. Input Data Operations With ESA.

1. General Description. The externally specified address mode (ESA) is instated for a particular channel by positioning the Channel Function switch on its drawer to the ESA position. When in effect, it alters the operations which honor an IDR on its channel. Two channels are used together, the one on which the IDR is sent which must be odd numbered and the next lower even numbered channel. Instead of the storage address for the input data being determined by the CACW, the external device directly specifies the address. Refer to figure 8.5-7 for the input word format.

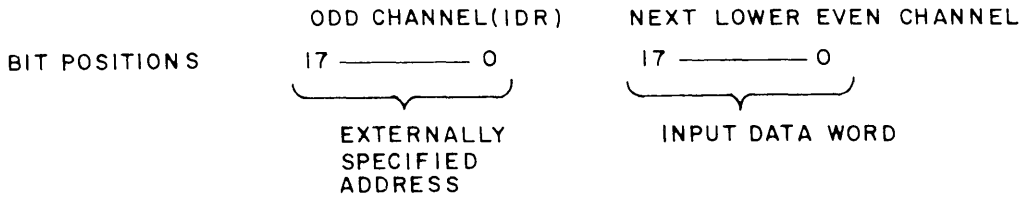


Figure 8.5-7. ESA Input Word Format for IDR Operations

The IDR signal must come from an odd channel. The input word on that channel is accepted as the data storage address. The word on the even channel is stored at this address. There are no buffer limits (address control words).

The only way that the external device can terminate the input buffer is for the 16 least significant bits inputted as the address to be all 0's. Upon termination, the input monitor interrupt signal is generated if bit $16 = 1_2$ of the odd channel. Also at termination, the externally specified address is stored at the address $0016_8 + 2$ x channel # if the channel is 0-7 or $0036_8 + 2$ x channel # if the channel is $10\text{-}17_8$.

2. Detailed Analysis.

a) Data Flow Block Diagram. Refer to figure 8.5-8 for a block diagram description of the ID operations with ESA.

During the last portion of the scan sequence, the externally specified address is inputted and placed in B. The circuitry which is normally used to compare the TACW with the CACW is still effective in examining ZO_{15-0} and B_{15-0}. If $B_{15-0} = 0$, the input operations will terminate after this current data storage.

The I/O1-sequence performs the data storage from the even channel using the externally specified address in B. As normal, the input monitor interrupt signal can be generated upon termination if selected by bit 16 of this address word.

The I/O2-sequence is used only if the input operation is terminated. It performs the storage of the externally specified address.

b) Essential Commands. Refer to table 8.5-9 for a sequential list of scan, I/O1, and I/O2-sequence events. Develop these commands by referring to the proper enable pages in the logic diagrams.

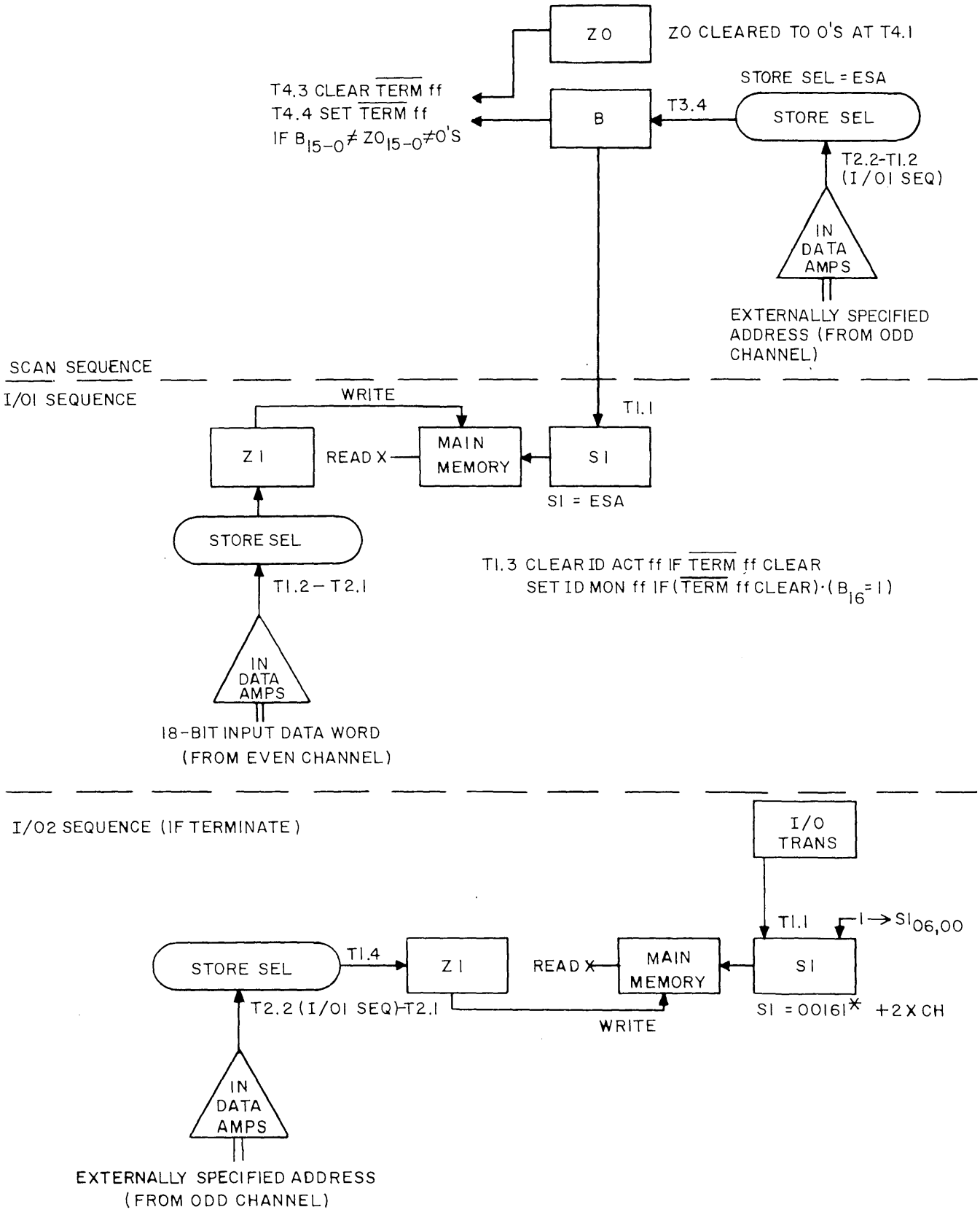
The events concerning priority selection are as previously discussed and are not shown. The events concerning the continuous data mode sequence are analyzed in a later sheet. The ID-acknowledge timing is the same as for normal input operations.

8.5-5. SUMMARY

A data word sent to the computer is accompanied by the IDR signal which requests the computer to accepts the word. The ID-acknowledge signal is sent back to the data sending peripheral device to indicate the completion of the input operation. One of four modes can be selected with the CHANNEL FUNCTION switch (single channel, dual channel, ESI, or ESA). Mode can only be selected if the IDR occurs on an odd channel; otherwise, single channel mode is in effect.

Single channel operation involves one 18-bit word inputted on the requesting channel. Dual channel operation causes one 18-bit word to be inputted on the requesting channel, which must be odd, and another on the next lower even channel.

ESI and ESA allow the external device to either indirectly or directly specify the storage address for the inputted data word.



NOTE: IF CHANNEL IS $10-17_8$ ADD 00200_8 TO ADDRESS IN S1

Figure 8.5-8. Scan, I/01, and I/02-Sequence Data Flow for ID Operations with ESA

TABLE 8.5-9. SCAN, I/01, AND I/02 SEQUENCE ESSENTIAL COMMANDS FOR ID OPERATIONS WITH ESA

TIME NOTATION	COMMANDS
	SCAN SEQUENCE
T2.1	Clear Fct, Chan, & I/O Trans 1
T2.2	Set $I/01_i$ ff, Pri→Fct, Chan, & I/O Trans 1 (set ESA ff)
	Odd chan Input Data Amps→Store Sel
T3.2	Clear B
T3.4	Store Sel→B
T4.1	Clear I/O Trans 2, clear ZO
T4.2	Set $I/01_f$ ff, I/O Trans 1→I/O Trans 2, clear all ID Ack Reg ff's
T4.3	Clear ID Req ff, set ID Ack Reg ff, set 6XLg0 ff
	Clear $\overline{\text{Term}}$ ff, clear Pri Alternator ff
T4.4	Set $\overline{\text{Term}}$ ff if $B_{15-0} \neq ZO_{15-0}$, clear S1, disable Mem→Z1
	I/01 SEQUENCE
T1.1	B→S1, Init Memory
T1.2	Drop odd chan Input Data Amps→Store Sel
	Even chan Input Data Amps→Store Sel
T1.3	Clear Z1, clear ID Act ff if $\overline{\text{Term}}$ ff clear
	Set ID Monitor ff if ($\overline{\text{Term}}$ ff clear)·($B_{16} = 1$)
T1.4	Store Sel→Z1, set ID Ack Gener ff if $\overline{\text{Term}}$ ff clear
T2.1	Clear $I/01_i$ ff, drop even chan Input Data Amps→Store Sel
T2.2	Set $I/02_i$ ff if $\overline{\text{Term}}$ ff clear, odd chan Input Data Amps→Store Sel if $I/02_i$ ff set
T2.4	Drop disable Mem→Z1
T4.1	Clear $I/01_f$ ff
The following occurs if $I/02_i$ ff set (terminate)	
T4.2	Set $I/02_f$ ff, clear all ID Ack Reg ff's
T4.3	Set ID Ack Reg ff, set 6XLg0 ff
T4.4	Disable Mem→Z1, clear S1
	I/02 SEQUENCE (if terminate)
T1.1	I/O Trans→S1, 1→$S1_{00}$, 1→$S1_{06}$, Init Memory
T1.3	Clear Z1
T1.4	Store Sel→Z1, set ID Ack Gener ff
T2.1	Clear $I/02_i$ ff, drop odd chan Input Data Amps→Store Sel
T2.4	Drop disable Mem→Z1
T4.1	Clear $I/02_f$ ff

NAME: _____

8.5-6. STUDY QUESTIONS

a. Given: pin 11 of 0XG19 ff grounded (logic diagrams, figure 9-106)

Describe the effect which this malfunction would have upon normal ID buffer operations.

b. Given: pin 8 of 5XV12 ff grounded (logic diagrams, figure 9-68)

1. Describe any effect which this malfunction would have during the honoring of an IDR on channel 1.

2. Describe any effect which this malfunction would have during the honoring of an IDR on channel 10_8.

c. Given: input word on channel 7 = 072656

ESI mode is in effect for channel 7. The given input word will be used to formulate the address to obtain the CACW and TACW. From what control memory addresses will these addresses control words be obtained?

address to obtain TACW = _____

address to obtain CACW = _____

ESI (DUAL)

STORE BUFFER CONTROL WORDS {EVEN - TACW / ODD - CACW

MULTIPLEXOR	STORE WORDS
00 - 07	00 - 17
10 - 17	20 - 37
20 - 27	40 - 57
30 - 37	60 - 77
40 - 47	200 - 217
50 - 57	220 - 237
60 - 67	240 - 257
70 - 77	260 - 277
100 - 107	400 - 417
110 - 117	420 - 437
120 - 127	440 - 457
130 - 137	460 - 477
140 - 147	600 - 617
150 - 157	620 - 637
160 - 167	640 - 657
170 - 177	660 - 677

SECTION 8 - INPUT/OUTPUT SECTION

8.6. OUTPUT, DATA AND EXTERNAL FUNCTION REQUESTS OPERATIONS

8.6-1. OBJECTIVES

To present the detailed theory of operation involved in output data and external function requests operations.

8.6-2. INTRODUCTION

The output data request is a signal from an external device which requests that the computer output a data word. The external function request is also sent from an external device; it requests an external function control word. The computer can have just one of these buffer types active at one time on a particular channel.

8.6-3. REFERENCES

a. UNIVAC 1219 Technical Manual, Volume I, Paragraph 4-3.

b. UNIVAC 1219 Technical Manual, Volume II, Section 9 (logic diagrams).

8.6-4. INFORMATION

a. OD/EF Operations Without ESI or ESA.

1. General Description.

a) Single Channel Operation. The output data request (ODR) or external function request (EFR) from an external device is honored by the computer only if the particular channel is output active or external function active, respectively. The External Function/Output Data (EF/OD) Active flip-flop is set during the execution of f = 50:02, 50:03, 50:12, 50:13.

If the priority logic selects an ODR or EFR signal, the current address control word (CACW) is obtained from the control memory address 00041_8 + 2x channel # if the channel is 0-7 or 000241_8 + 2 x channel # if the channel is $10\text{-}17_8$. The terminal address control word (TACW) is obtained from the control memory address 00040_8 + 2 x channel # if the channel is 0-7 or 00240_8 + 2 x channel # if the channel is $10\text{-}17_8$. The 18-bit word which is outputted is obtained from the address specified by the $CACW_{15\text{-}0}$.

The $CACW_{15\text{-}0}$ is modified by 1 as determined by the value in bit position 17 of the CACW and is restored in control memory. $CACW_{17}$ determines whether the buffer is to be forward or backward. In a forward buffer ($CACW_{17} = 0_2$), the outputted words are obtained from consecutively incremented addresses as the ODR or EFR signals occur. In this case, the $CACW_{15\text{-}0}$ is modified by +1 after each word is outputted. In a

backward buffer ($CACW_{17} = 1_2$), the outputted words are obtained from consecutively decremented addresses as the ODR or EFR signals occur. In this case, the $CACW_{15-0}$ is modified by -1 after each word is outputted.

Refer to tables 8.6-1 and 8.6-2 for buffer examples.

TABLE 8.6-1. SINGLE CHANNEL FORWARD OD/EF BUFFER
IACW = 005000, TACW = 005003

ADDRESS OF OUTPUTTED WORD ($CACW_{15-0}$)	WORD OUTPUT SEQUENCE
05000	1
05001	2
05002	3
05003	4

TABLE 8.6-2. SINGLE CHANNEL BACKWARD OD/EF BUFFER
IACW = 405003, TACW = 405000

ADDRESS OF OUTPUTTED WORD ($CACW_{15-0}$)	WORD OUTPUT SEQUENCE
05000	4
05001	3
05002	2
05003	1

The TACW is not modified but is compared with the CACW. If their 16 least significant bits are equal before the modification of the CACW, the current output word being processed completes the buffer. To terminate the buffer, the EF/OD Active flip-flop for the particular channel is cleared which disables the recognition of future ODR and EFR signals. At this buffer termination time, the value in bit position 16 of the CACW is sensed. If $CACW_{16} = 1_2$, an external function/output data monitor signal is to be generated to indicate to the program the termination; thus, the EF/OD Monitor flip-flop for the particular channel is set. The resulting EF/OD monitor signal is handled later by the computer as a separate request for input/output service and is discussed in a later sheet.

The OD-acknowledge or EF-acknowledge signal is sent on the same channel which carried the request. The type of acknowledge signal which is sent indicates to the receiving device that the outputted word is a data word or a control word.

If $TACW_{17} = 1_2$, continuous data mode is specified. This bit is sensed when the buffer is terminated and allows the special continuous data mode (CDM) sequence to be activated. This sequence prevents the clearing of the EF/OD Active flip-flop and reloads the TACW and CACW locations in control memory with new buffer control words. CDM operations are presented in a later sheet.

b) Dual Channel Operation. The dual channel mode is instated for a particular channel by positioning the CHANNEL FUNCTION switch on its drawer to the DUAL position. If dual channel has been selected, the ODR or EFR signal is actually requesting that the computer output two sepearate 18-bit words, one on the request channel which must be odd for dual channel operation and the other word on the next lower channel which is even numbered. The output operations for the second word are identical to those for the first word. The second word, therefore, is obtained from memory at the address specified by the $CACW_{15-0}$ which has been modified by 1 during the outputting of the first word.

The type of buffer (forward or backward) determines which of the two channels is serviced first. The odd channel, which must carry the ODR or EFR, receives the first 18-bit word if the buffer is forward. The even channel is serviced first if the buffer is backward. Refer to tables 8.6-3 and 8.6-4 for buffer examples assuming the ODR or EFR signals occur on channel 3.

TABLE 8.6-3. DUAL CHANNEL FORWARD OD/EF BUFFER ON CHANNELS 2, 3
IACW = 005000, TACW = 005003

	ADDRESS OF OUTPUTTED WORD ($CACW_{15-0}$)	RECEIVING CHANNEL	WORD OUTPUT SEQUENCE	ODR OR EFR SEQUENCE
1st 36-bit output word	05000	3	1	1st ODR/EFR
	05001	2	2	
2nd 36-bit output word	05002	3	3	2nd ODR/EFR
	05003	2	4	

The 16 least significant bits of the TACW and CACW are compared during each 18-bit word output operation. Therefore, the clearing of the EF/OD Active flip-flop and the setting of the EF/OD Monitor flip-flop can occur during either the first or second word output operation. The TACW and IACW should be chosen to provide a buffer for an exact multiple of two 18-bit words such that the buffer will always terminate as a result of the second word output operation from a particular ODR or EFR.

The OD-acknowledge or EF-acknowledge signal is sent on the odd channel which carried the ODR or EFR. The acknowledge occurs after the second word has been outputted.

TABLE 8.6-4. DUAL CHANNEL BACKWARD OD/EF BUFFER ON CHANNEL 2, 3
IACW = 405003, TACW = 405000

	ADDRESS OF OUTPUTTED WORD ($CACW_{15-0}$)	RECEIVING CHANNEL	WORD OUTPUT SEQUENCE	ODR OR EFR SEQUENCE
2nd 36-bit output word	05000	3	4	2nd ODR/EFR
	05001	2	3	
1st 36-bit output word	05002	3	2	1st ODR/EFR
	05003	2	1	

2. Detailed Analysis.

a) Data Flow Block Diagram.

1) Single Channel Operation. Refer to figure 8.6-1 for a block diagram description of the single channel OD/EF operation.

The scan sequence is the main timing cycle during which the $I/O1_i$ flip-flop is set upon detection and selection of an ODR or EFR signal. During this sequence the CACW is obtained from control memory at the address 00041_8 + 2 x channel # or 000241_8 + 2 x channel # depending upon the channel. The address of either 00040_8 + 2 x channel # or 000240 + 2 x channel # is formulated in the I/O translator which is set according to the priority selection. SO_{00} is set to 1_2 to make the address odd.

Another control memory reference is used to obtain the TACW. The TACW and CACW are compared from ZO and B. The result of the comparison is recorded by the terminate flip-flop. If $TACW_{15-0} = CACW_{15-0}$, this ODR or EFR will be the last to be honored.

The B-netword is used to modify the $CACW_{15-0}$ as determined by $CACW_{17}$. The value of this bit is recorded by the B $\pm$ 1 flip-flop.

The I/O 1-sequence is used to obtain and output the 18-bit word. The memory reference uses the address specified by the unmodified $CACW_{15-0}$ from B. A control memory reference is used at T1.1 time to store the modified CACW value for use during the next output operation.

The remaining I/O1 sequence operations are used only if the operation is dual channel.

2) Dual Channel Operation. Refer to figures 8.6-1 and 8.6-2.

The scan and I/O1-sequences are used as for single channel operations. The last portion of the I/O1-sequence and the I/O2-sequence are used to perfrom the same operations executed during the last portion of the scan sequence and the first portion of the I/O1-sequence. These operations obtain the second word from the first word address $\pm$1.

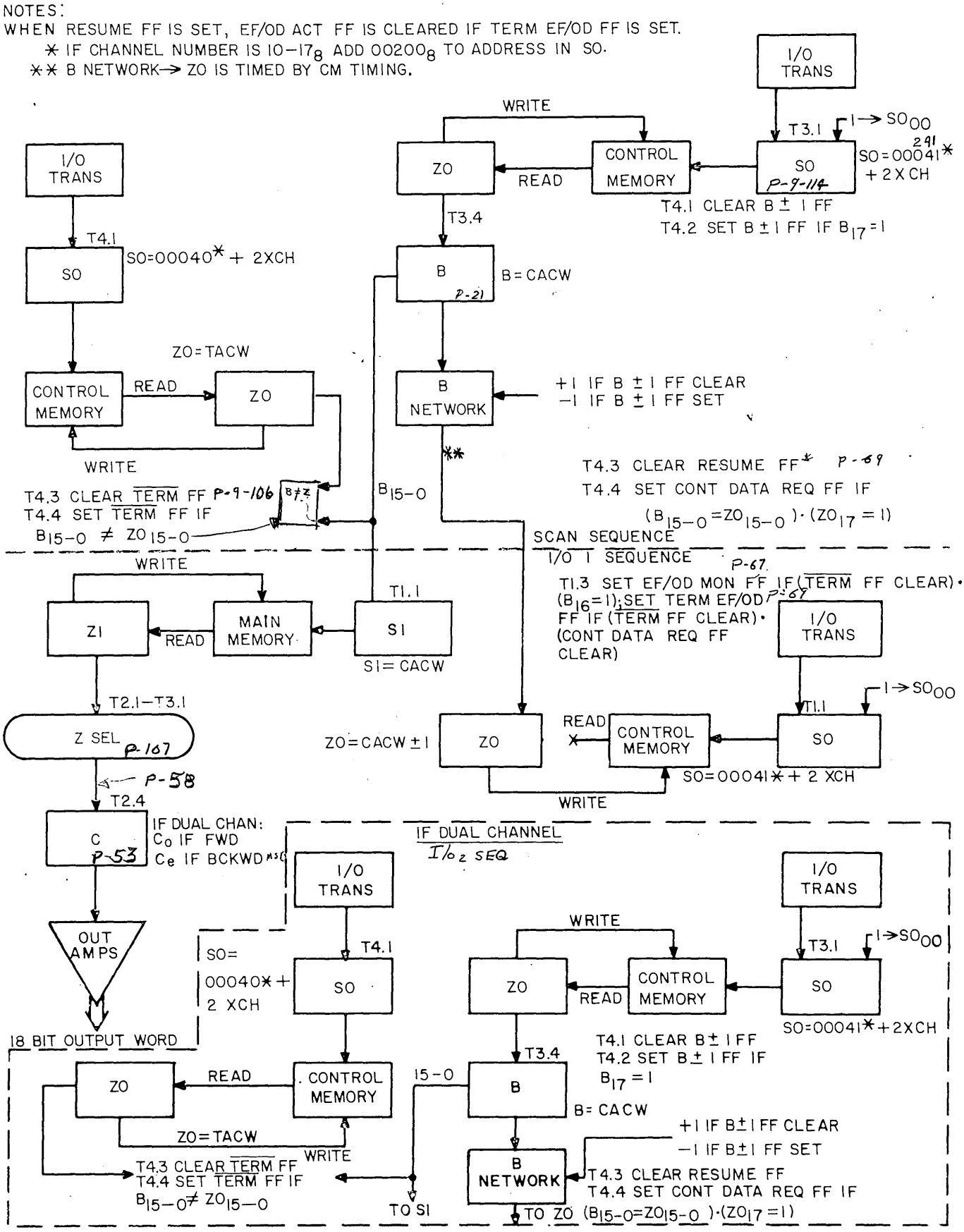
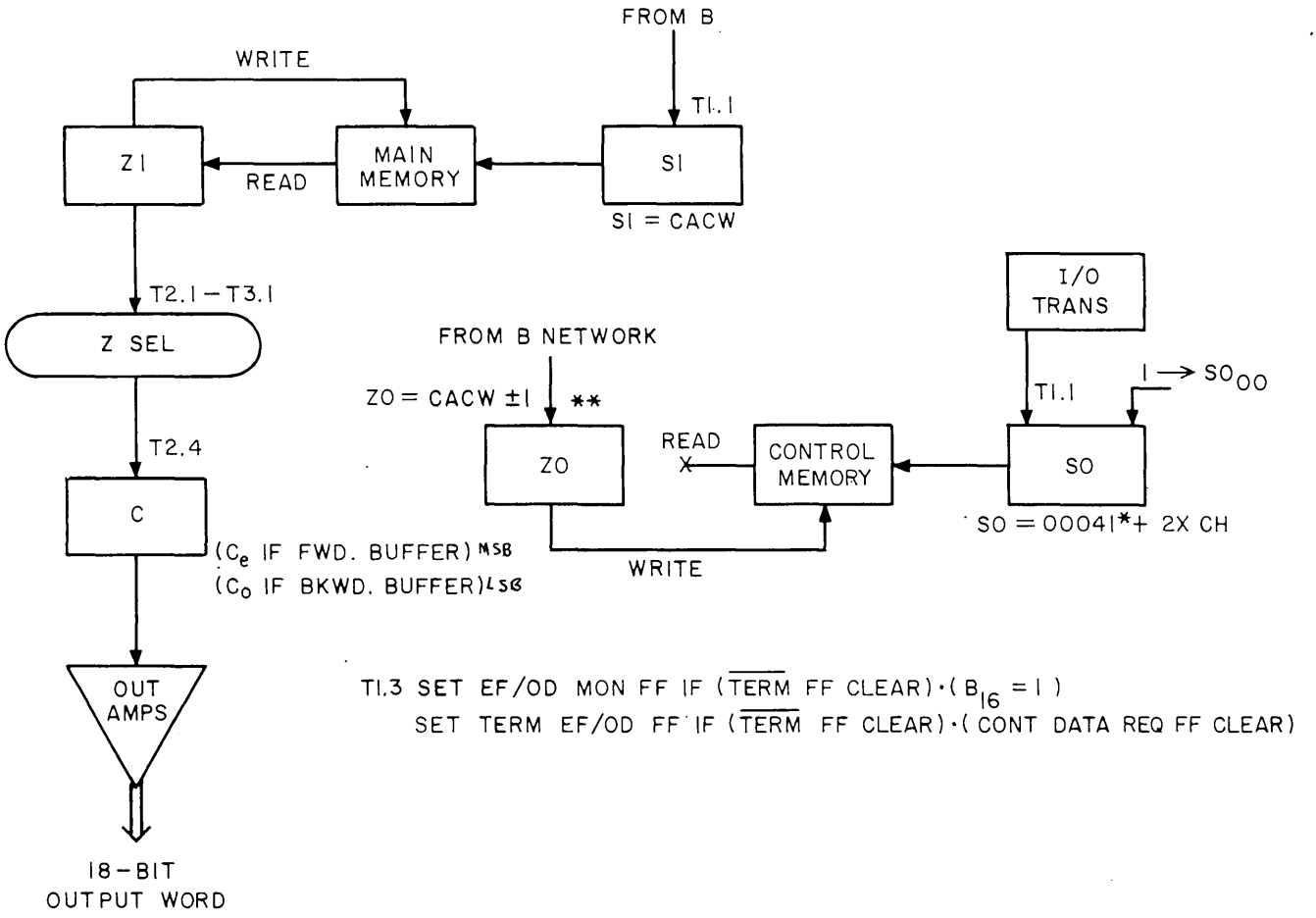


Figure 8.6-1. Scan and I/O1-Sequence Data Flor for Single or Dual Channel OD/EF Operations



NOTES: WHEN RESUME FF IS SET, EF/OD ACT FF IS CLEARED IF TERM EF/OD FF IS SET.

*IF CHANNEL NUMBER IS $10-17_8$, ADD 00200_8 TO ADDRESS IN SO.

**B NETWORK ⟶ Z0 IS TIMED BY CM TIMING.

Figure 8.6-2. I/O2-Sequence Data Flow for Dual Channel OD/EF Operations

b) Essential Commands. Refer to table 8.6-5 for a sequential list of essential Scan, I/O1, and I/O2-sequence events. Develop the commands shown by referring to the proper enable pages in the logic diagrams.

The events concerning priority selection are as discussed previously and are not shown. The events concerning the continuous data mode sequence, which is initiated as a result of the setting of the Continuous Data Request flip-flop, are analyzed in a later sheet. The EF/OD-acknowledge timing is discussed later in this sheet.

b. EF/OD-Acknowledge Timing.

1. General Description. The EF/OD-acknowledge signal is sent after the word output operation. In the case of dual channel operation, the acknowledge is sent after both words have been outputted. The purpose of this signal is to indicate that a word is currently being outputted and is ready for receipt by the receiving device. The receiving device usually uses the acknowledge to time the dropping of its ODR or EFR signal. The acknowledge signal is sent for a time duration of 2.375 microseconds and 14.375 microseconds for fast and slow interface, respectively.

When operating in the inter-computer mode, the acknowledge timing assists in processing the next ODR from the receiving computer.

2. Detailed Analysis.

a) Not Inter-Computer Mode. Refer to figure 8.6-3 and logic diagrams, figures 9-61, 9-68, and 9-69.

The EF/OD-Acknowledge Generation flip-flop determines the acknowledge duration by its set time. The EF/OD Control flip-flop determines which type of acknowledge is sent, EF or OD. Notice that the Resume flip-flop is set from the acknowledge timing via 52Lg1 and the INTER-COMPUTER/NORMAL switch. When set, the Resume flip-flop clears the EF/OD Active flip-flop via 72Lg0 if the Terminate EF/OD flip-flop is set. Refer to tables 8.6-6 and 8.6-7 for the timing analysis of the acknowledge for fast and slow interface, respectively.

b) Inter-Computer Mode. Refer to figure 8.6-3 and logic diagrams, figures 9-61, 9-68, and 9-69.

Much of the timing is the same as for normal mode. The major difference involves the dropping of the acknowledge. This signal is held applied to the other computer until that computer sends an ID-acknowledge indicating that it has received the word just outputted. The ID-acknowledge is detected by this computer as an ODR. Also, until this acknowledge is received, the Resume flip-flop remains clear. Thus, the EF/OD Active flip-flop is not enabled to be cleared until the acknowledge is received which appears as an ODR. The delayed clearing of the active flip-flop allows one more ODR than is necessary to complete the buffer, to be recognized and set the Resume flip-flop which in turn can clear the active flip-flop.

Until the Resume flip-flop is set, output is not available on the particular chassis. Therefore, no other word can be outputted via the same C register and the current output word will be held applied to the receiving computer until it responds with the ID-acknowledge (ODR). As described above, this ODR signal sets the Resume

TABLE 8.6-5. SCAN, I/01, AND I/02 SEQUENCE ESSENTIAL COMMANDS FOR SINGLE OR DUAL CHANNEL OD/EF OPERATIONS

TIME NOTATION	COMMANDS
	SCAN SEQUENCE
T2.1	Clear Fct, Chan, & I/O Trans 1
T2.2	Set $I/01_i$ ff, Pri→Fct, Chan, & I/O Trans 1, disable B Network $_{17, 16}$
T3.1	I/O Trans→SO, 1→SO_{00}, Init CM, clear ZO
T3.2	Clear B
T3.4	ZO→B*
T4.1	I/O Trans→SO, Init CM, clear ZO, clear B±1 ff (+1→B Network), clear I/O Trans 2
T4.2	Set $I/01_f$ ff, set B+1 ff (-1→B Network) if B_{17}=1 I/O Trans 1→$\overline{I/O}$ Trans 2 (set Dual ff if dual chan selected)
T4.3	Clear $\overline{\text{Term}}$ ff, clear Resume ff
T4.4	Set $\overline{\text{Term}}$ ff if $B_{15-0} \neq ZO_{15-0}$, disable CM→ZO Clear S1, set Cont Data Req ff if $(B_{15-0}=ZO_{15-0})\cdot(ZO_{17}=1)$
	I/01 SEQUENCE
T1.1	I/O Trans→SO, 1→SO_{00}, Init CM, B→S1, Init Memory
T1.2	Clear all EF/OD Ack Reg ff's
T1.3	Clear EF Req if EFR, clear Z1, set EF/OD Control ff if EF mode clear EF/OD Control ff if OD mode clear OD Req ff if (inter-cmptr) + (inter-cmptr · ODR) set EF/OD Mon ff if ($\overline{\text{Term}}$ ff clear)·(B_{16}=1), set Pri Alternator ff set Term EF/OD ff if ($\overline{\text{Term}}$ ff clear)·(Cont Data Req ff clear) set Ack Delay ff if Dual ff clear, set EF/OD Ack Reg ff
T2.1	Clear $I/01_i$ ff, Z1→Z Sel, drop disable B Network $_{17, 16}$
T2.2	Set $I/02_i$ ff if Dual ff set, disable B Network $_{17, 16}$ if $I/02_i$ ff set
T2.3	Clear C**
T2.4	Z Sel→C**, drop disable CM ZO
The following occurs if $I/02_i$ ff set (dual channel) excepting "clear $I/01_f$ ff" and "drop Z1—Z Sel"	
T3.1	I/O Trans→SO, 1→SO_{00}, Init CM, Clear ZO, drop Z1→Z Sel
T3.2	Clear B
T3.4	ZO→B*
T4.1	Clear $I/01_f$ ff, I/O Trans→SO, Init CM, clear ZO clear B±1 ff (+1→B Network), clear I/O Trans 2
T4.2	Set $I/02_f$ ff, set B+1 ff (-1→B Network) if B_{17}=1 I/O Trans 1→$\overline{I/O}$ Trans 2 (set Dual ff)
T4.3	Clear $\overline{\text{Term}}$ ff, clear Resume ff
T4.4	Set $\overline{\text{Term}}$ ff if $B_{15-0} \neq ZO_{15-0}$, disable CM ZO Clear S1, set Cont Data Req ff if $(B_{15-0}=ZO_{15-0})\cdot(ZO_{17}=1)$
	I/02 SEQUENCE
T1.1	I/O Trans→SO, 1→SO_{00}, Init CM, clear ZO B→S1, Init Memory
T1.2	Clear all EF/OD Ack Reg ff's
T1.3	Clear Z1, set EF/OD Ack Reg ff set EF/OD Mon ff if ($\overline{\text{Term}}$ ff clear)·(B_{16}=1) set Term EF/OD ff if ($\overline{\text{Term}}$ ff clear)·(Cont Data Req ff clear) set Ack Delay ff
T2.1	Clear $I/02_i$ ff, Z1→Z Sel, drop disable B Network $_{17, 16}$
T2.3	Clear C**
T2.4	Z Sel→C**, drop disable CM ZO
T3.1	Drop Z1→Z Sel
T4.1	Clear $I/02_f$ ff

*B Network→ZO is timed by CM timing.

**If operation is in dual channel mode, the C register which is cleared and receives word from Z-select is as follows.

	I/01 Seq.	I/02 Seq.
B±1 ff clear (forward buffer)	Co	Ce
B±1 ff set (backward buffer)	Ce	Co

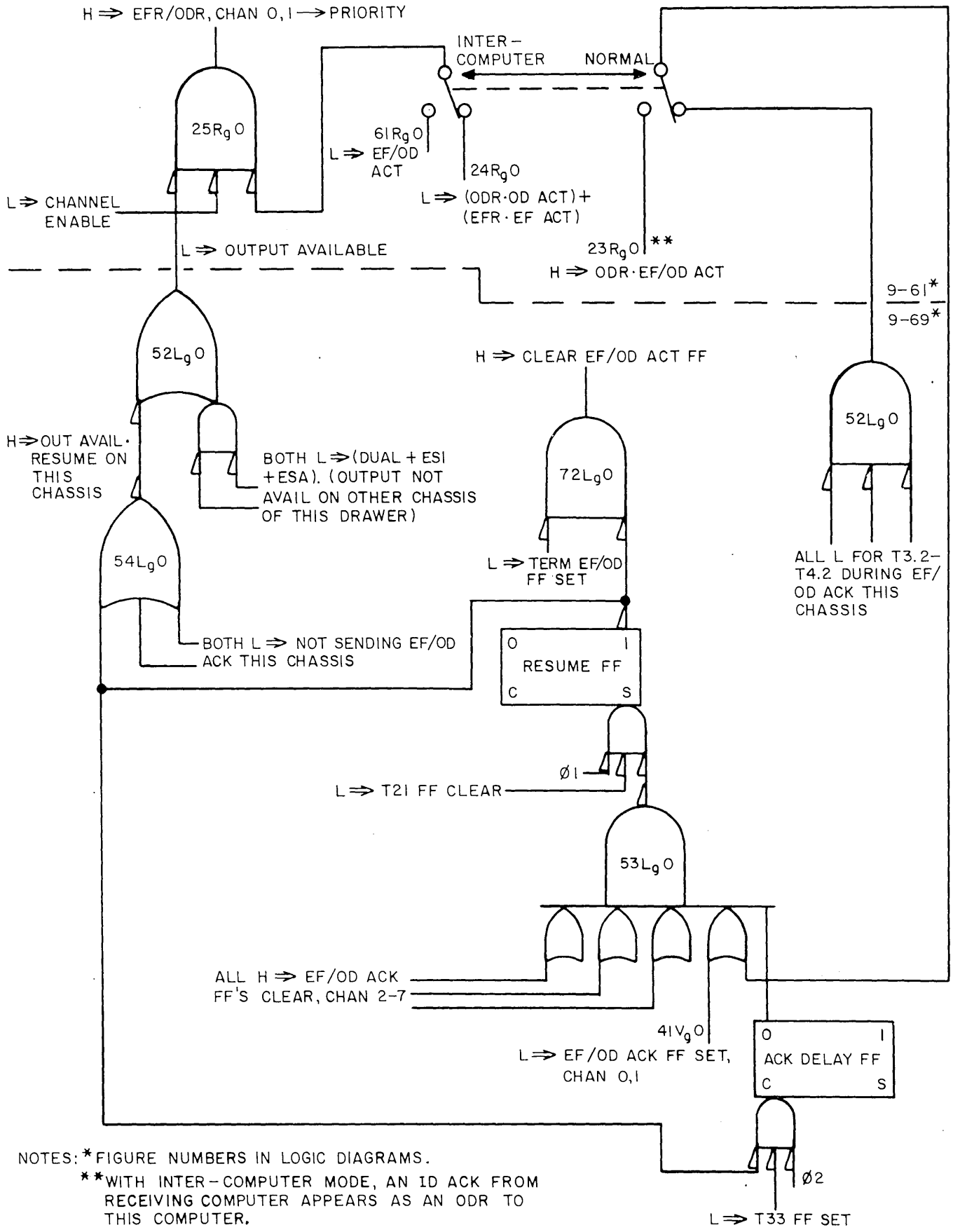


Figure 8.6-3. OD/EF-Acknowledge Simplified Logic

TABLE 8.6-6. EF/OD-ACKNOWLEDGE TIMING FOR FAST INTERFACE, NOT INTER-COMPUTER

TIME NOTATION	ACTION
T1.2	Clear all EF/OD Ack Reg ff's P-68
T1.3	Set EF/OD Ack Reg ff, set Ack Delay ff Set EF/OD Control ff if EF mode, clear EF/OD Control ff if OD mode SET PRIORITY ALTERNATOR
T4.1	Set Resume ff, clear EF/OD Act ff if Term EF/OD ff set
3.4 Time T4.4 (1.25 µs)	Clear Term EF/OD ff, set EF/OD Ack Gener ff, send EF/OD-Acknowledge signal
2.875 µs T3.2 (1.125 µs)	Clear Ack Delay ff
T1.3	Clear EF/OD Ack Gener ff, drop EF/OD Acknowledge signal

TABLE 8.6-7. EF/OD-ACKNOWLEDGE TIMING FOR SLOW INTERFACE, NOT INTER-COMPUTER

TIME NOTATION	ACTION
T1.2	Clear all EF/OD Ack Reg ff's
T1.3	Set EF/OD Ack Reg ff, set Ack Delay ff Set EF/OD Control ff if EF mode, clear EF/OD Control ff if OD mode
4 µs delay of 59Lg2 T1.3	
T1.3 (1.625 µs)	
T4.4	Set EF/OD Ack Gener ff, send EF/OD Acknowledge signal
T4.4	
10 µs delay of 59Lg3 T4.4	
T4.4	
T4.4	
14.375 µs	
T4.4 (1.625 µs)	Set 5XLg3 ff
T4.1 (1.625 µs)	Set Resume ff, clear EF/OD Act ff if Term EF/OD ff set
T4.4	Clear Term EF/OD ff
T3.2 (1.125 µs)	Clear Ack Delay ff
T1.3	Clear EF/OD Ack Gener ff, drop EF/OD-Acknowledge signal
T2.2	Clear 5XLg3 ff

flip-flop. If the buffer is not to terminate (EF/OD Active flip-flop is not cleared when the Resume flip-flop is set), the Resume flip-flop completely satisfies 25RgO which generates the next ODR signal to request the next word to be outputted. The receiving computer cannot send an EFR signal.

The status of the Resume flip-flop provides an indication of whether or not the receiving computer acknowledged receipt of the last outputted word. If it is not set within a specific time period, the Resume Fault special interrupt is generated to notify the program. This signal is discussed in a later sheet.

Refer to tables 8.6-8 and 8.6-9 for the timing analysis of the acknowledge for fast and slow interface, respectively.

Inter-computer operations and programing considerations are discussed in a later sheet.

TABLE 8.6-8. EF/OD-ACKNOWLEDGE TIMING FOR FAST INTERFACE, INTER-COMPUTER

TIME NOTATION	ACTION
T1.2	Clear all EF/OD Ack Reg ff's
T1.3	Set EF/OD Ack Reg ff, set Ack Delay ff Set EF/OD Control ff if EF mode, clear EF/OD Control ff if OD mode
T4.4	Set EF/OD Ack Gener ff, send EF/OD-Acknowledge signal
wait for next ODR	
receive ODR from other computer (ID-Acknowledge)	
Next T1.1, T3.1, or T4.1	Set Resume ff, internally generate next ODR, clear EF/OD Act ff if Term EF/OD ff set
Next Ø4	Clear Term EF/OD ff
T3.2	Clear Ack Delay ff
T1.3	Clear EF/OD Ack Gener ff, drop EF/OD-Acknowledge signal

c. OD/EF Operations With ESI.

1. General Description. The externally specified index mode (ESI) is instated for a particular channel by positioning the CHANNEL FUNCTION switch on its drawer to the ESI position. When in effect, it alters the operations which honor an ODR or EFR on its channel. Two channels are used together, the one on which the ODR or EFR is sent which must be odd numbered and the next lower even numbered channel. Instead of the TACW and CACW being obtained from their normal addresses in control memory, their control memory origins are specified by the requesting external device.

TABLE 8.6-9. EF/OD-ACKNOWLEDGE TIMING FOR SLOW INTERFACE, INTER-COMPUTER

TIME NOTATION	ACTION
T1.2	Clear all EF/OD Ack Reg ff's
T1.3 (4 µs delay of 59Lg2)	Set EF/OD Ack Reg ff, set Ack Delay ff Set EF/OD Control ff if EF mode, clear EF/OD Control ff if OD mode
T1.3 (4 µs delay of 59Lg2)	
T1.3 (1.625 µs)	
T4.4 (1.625 µs)	Set EF/OD Ack Gener ff, send EF/OD-Acknowledge signal
T4.4 (10 µs delay of 59Lg3)	
T4.4 (10 µs delay of 59Lg3)	
T4.4 (10 µs delay of 59Lg3)	
T4.4 (10 µs delay of 59Lg3)	
T4.4 (wait for next ODR)	Set 5XLg3 ff
receive ODR from other computer (ID-Acknowledge)	
next T1.1, T3.1, or T4.1	Set Resume ff, internally generate next ODR, clear EF/OD Act ff Term EF/OD ff set
next Ø4	Clear Term EF/OD ff
T3.2	Clear Ack Delay ff
T1.3	Clear EF/OD Ack Gener ff, drop EF/OD-Acknowledge signal
T2.2	Clear 5XLg3 ff

The ODR or EFR signal must appear on an odd channel. There must be an ESI address word accompanying the request on this same channel. This word is the control memory address of the TACW. The CACW is obtained from the next consecutively higher address. The content of the address specified by the $CACW_{15-0}$ is the requested output word. This word is outputted on both the requesting odd channel and the next lower even channel. Refer to figure 8.6-4 for the input and output word formats.

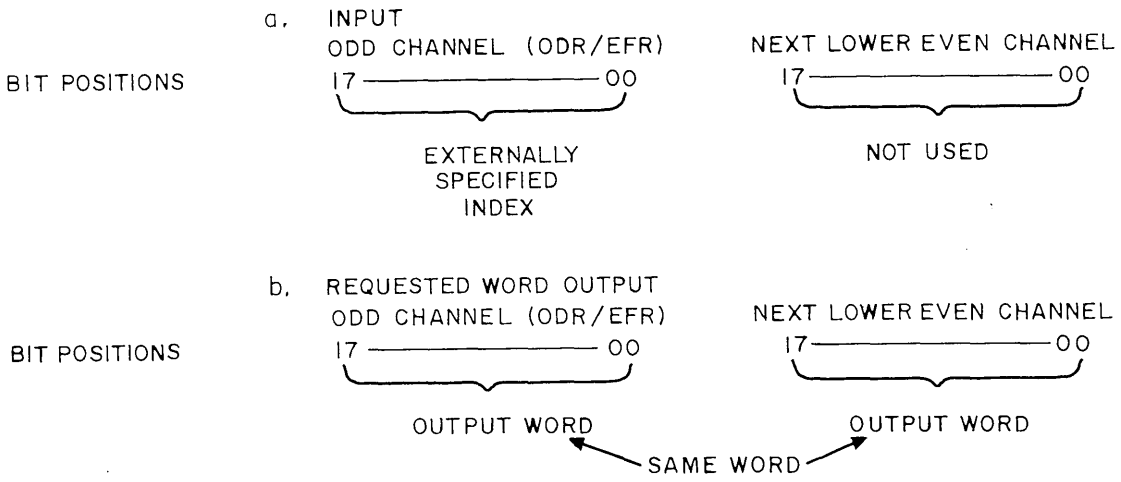


Figure 8.6-4. ESI Word Formats for ODR/EFR Operations

The $CACW_{15-0}$ is modified by ± 1 depending upon its bit position 17. The outputted word is obtained from the address specified by the unmodified $CACW_{15-0}$. If the $TACW_{15-0} = CACW_{15-0}$ (unmodified), the buffer is terminated and the output data/external function monitor interrupt signal is generated if $CACW_{16} = 1$. The OD or EF Acknowledge signal is sent on the odd channel.

The ESI mode of operation is useful if more than one piece of equipment is connected to a common channel by means of a multiplexing device. Each piece of equipment can indirectly specify the address of the outputted word which it requests by means of the externally specified index. This index could be generated in the multiplexer as it requested each word. Each equipment would have a unique index value. The multiplexer must occupy an odd and the next lower even channels.

Upon buffer termination, the externally specified index is also stored in main memory at the address 0014_{16} + 2x channel # if the channel is 0-7 or 0034_{16} + 2x channel # if the channel is $10\text{-}17_{8}$. By inspecting the content of this address, the program could determine which of the equipment on that channel received the last output word and made the channel inactive.

2. Detailed Analysis.

a) Data Flow Block Diagram.

1) Not Buffer Termination. Refer to figure 8.6-5 for a block diagram description of the OD/EF operations with ESI.

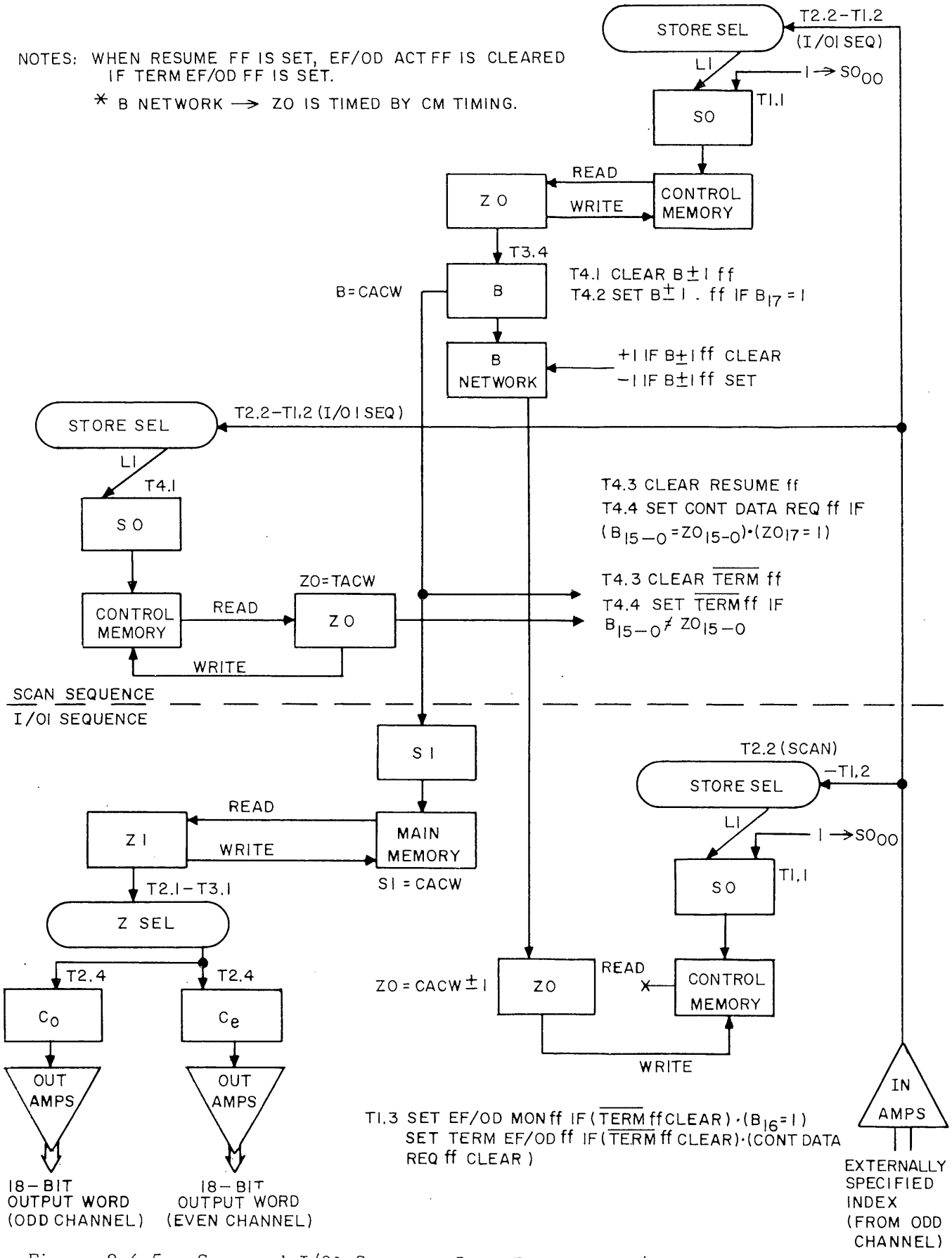


Figure 8.6-5. Scan and I/O1-Sequence Data Flow for OD/EF Operations with ESI

Data flow is similar to that for normal OD/EF operations. The exceptions are that the control memory addresses for the CACW and TACW are from the odd numbered channel. The externally specified index is used in SO at T4.1 time to obtain the TACW. The transfer into SO involves a one place left shift. Realize also that SO has no bit position 06. The inputted bits 05 and 06 are therefore placed in SO_{07} and SO_{08}, respectively. Nothing is placed in SO_{00} such that the address of the TACW is always even. Refer to figure 8.6-5 for a description of the ESI transfer to SO.

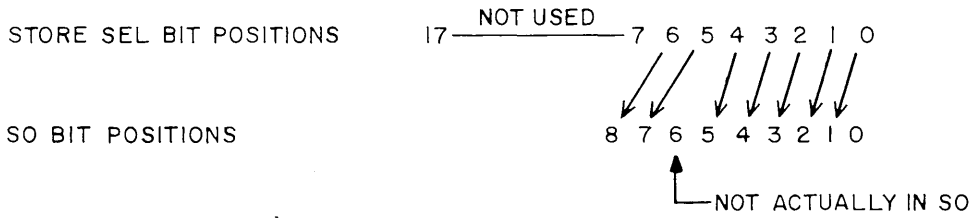


Figure 8.6-5. Store Select L1 ⟶ SO Transfer for ESI

The address of the CACW is formulated by setting SO_{00} to 1_2 in addition to the left shift input. The CACW is therefore obtained from the next consecutively higher address. All other I/O1-sequence operations are the same as those for normal output operations. The CACW is modified by ± 1 and is restored by another control memory reference. The output word is obtained from the address specified by the unmodified $CACW_{15-0}$.

The termination and monitor interrupt capabilities are the same as for normal output operations. The continuous data mode sequence can be initiated upon buffer completion if the $TACW_{17} = 1$, but is of no use in reloading the address control words; since this sequence always reloads the normal control memory addresses of the CACW and TACW. That is, it will place new control words in control memory addresses 00040_8 + 2x channel # and 00041_8 + 2x channel # or 00240_8 + 2x channel # and 00241_8 + 2x channel #, depending upon the channel number. These addresses are not used since the external device specifies the origins of the CACW and TACW. However, if the continuous data request flip-flop is set, it will prevent deactivating the buffer (the EF/OD Active flip-flop will not be cleared).

2) Buffer Termination. Refer to figure 8.6-7 for the I/O2-sequence operations executed upon buffer termination.

The I/O2-sequence is initiated if the $TACW_{15-0}$ equals the unmodified $CACW_{15-0}$. It uses a memory reference to store the externally specified index in memory for use by the program.

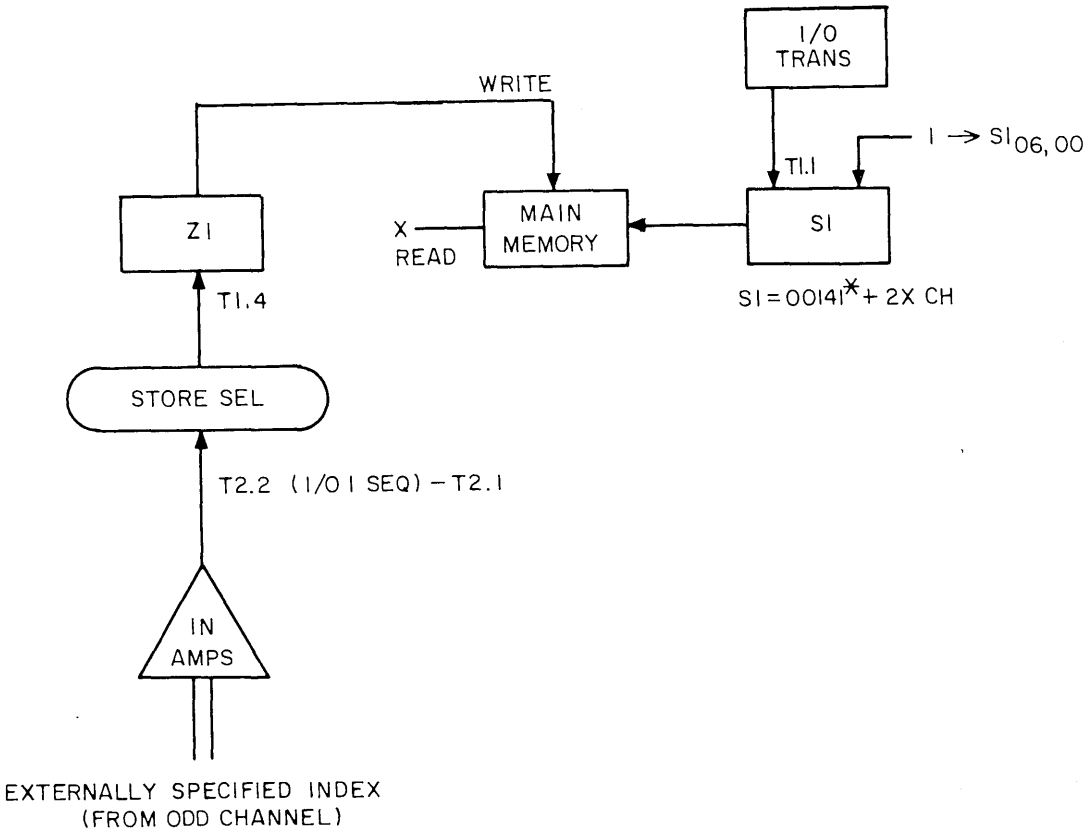
b) Essential Commands. Refer to table 8.6-10 for a sequential list of scan, I/O1, and I/O2-sequence events. Develop the commands shown by referring to the proper enable pages in the logic diagrams.

The events concerning priority selection are as previously discussed and are not shown. The events concerning the continuous data mode sequence are analyzed in a later sheet. The OD-acknowledge and EF-acknowledge timing is the same as for normal output operations.

TABLE 8.6-10. SCAN, I/01, AND I/02 SEQUENCE ESSENTIAL COMMANDS FOR OD/EF OPERATIONS WITH ESI

TIME NOTATION	COMMANDS
	SCAN SEQUENCE
T2.1	Clear Fct, Chan, & I/O Trans 1
T2.2	Set $I/01_i$ ff; Pri→Fct, Chan, & I/O Trans 1 (set ESI ff) disable B Network $_{17, 16}$, odd chan Input Amps→Store Sel
T3.1	Store Sel L1→S0 (ESI), 1→SO_{00}, Init CM, clear ZO
T3.2	Clear B
T3.4	Z0→B*
T4.1	Store Sel L1→S0 (ESI), Init CM, clear ZO, clear I/O Trans 2 clear B+1 ff (+1→B Network)
T4.2	Set $I/01_f$ ff, set B+1 ff (-1→B Network) if B_{17}=1 I/O Trans 1→$\overline{I/O}$ Trans 2
T4.3	Clear $\overline{\text{Term}}$ ff, clear Resume ff
T4.4	Set $\overline{\text{Term}}$ ff if $B_{15-0} \neq ZO_{15-0}$, disable CM→ZO, clear S1 Set Cont Data Req ff if $(B_{15-0}=ZO_{15-0})\cdot(ZO_{17}=1)$
	I/01 SEQUENCE
T1.1	Store Sel L1→S0 (ESI), 1→SO_{00}, Init CM, clear ZO B→S1, Init Memory
T1.2	Drop odd chan Input Amps→Store Sel, clear all EF/OD Ack Reg ff's
T1.3	Clear Z1, clear EF Req ff if EFR clear OD Req ff if (inter-cmptr)+($\overline{\text{inter-cmptr}}$·ODR) set EF/OD Control ff if EF mode, clear EF/OD Control ff if OD mode set Pri Alternator ff, set EF/OD Ack Reg ff set Ack Delay ff if $\overline{\text{Term}}$ ff set set Term EF/OD ff if ($\overline{\text{Term}}$ ff clear)·(Cont Data Req ff clear) set EF/OD Mon ff if ($\overline{\text{Term}}$ ff clear)·(B_{16}=1)
T2.1	Clear $I/01_i$ ff, Z1→Z Sel, drop disable B Network $_{17, 16}$
T2.2	Set $I/02_i$ ff if Term ff clear, odd chan Input Amps→Store Sel if $I/02_i$ ff set
T2.3	Clear C_o & C_e
T2.4	Z Sel→C_o & C_e, drop disable CM→ZO
T3.1	Drop Z1→Z Sel
T4.1	Clear $I/01_f$ ff
The following occurs if $I/02_i$ ff set (terminate)	
T4.2	Set $I/02_f$ ff
T4.4	Clear S1, disable Mem→Z1
	I/02 SEQUENCE (if terminate)
T1.1	I/O Trans→S1, 1→$S1_{00}$, 1→$S1_{06}$, Init Memory
T1.2	Clear all EF/OD Ack Reg ff's
T1.3	Clear Z1, set EF/OD Ack Reg ff, set Ack Delay ff
T1.4	Store Sel→Z1
T2.1	Clear $I/02_i$ ff, drop odd chan Input Amps—Store Sel
T2.4	Drop disable Mem→Z1
T4.1	Clear $I/02_f$ ff

*B Network→ZO is timed by CM timing.



NOTE: *IF CHANNEL IS $10\text{–}17_8$, ADD 00200_8 TO ADDRESS IN SI.

Figure 8.6-7. I/O2-Sequence Data Flow for Terminate OD/EF Operations with ESI

d. OD/EF Operations With ESA.

1. General Description. The externally specified address mode (ESA) is instated for a particular channel by positioning the CHANNEL FUNCTION switch on its drawer to the ESA position. When in effect, it alters the operations which honor an ODR or EFR on its channel. Two channels are used together, the one on which the ODR or EFR is sent which must be odd numbered and the next lower even numbered channel. Instead of the output word being obtained from the address determined by the CACW, the external device directly specifies the address.

The ODR or EFR signal must appear on an odd channel. There must be an ESA address word accompanying the request on this same channel. This word is the address of the requested output word. There are no buffer limits (address control words). The requested word is outputted on both the requesting odd channel and the next lower even channel. Refer to figure 8.6-8 for the input and output word formats.

The only way that the external device can terminate the OD or EF buffer is for the 16 least significant bits inputted as the address to be all 0's. Upon termination, the output data/external function monitor interrupt signal is generated if bit 16 = 1_2 of the odd channel. Also at termination, the externally specified address is

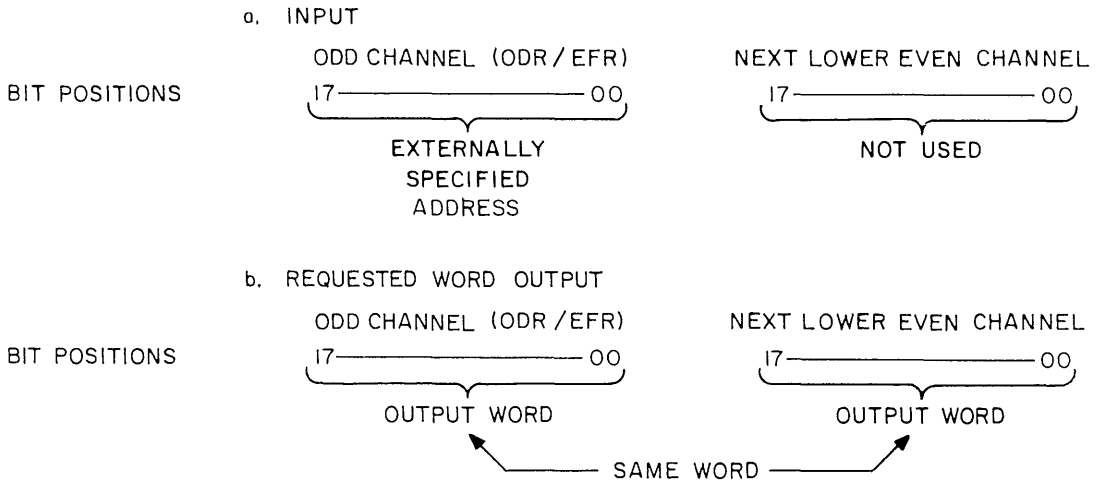


Figure 8.6-8. ESA Word Formats for ODR/EFR Operations

stored at the address 00141_8 + 2x channel # if the channel is 0-7 or 00341_8 + 2x channel # if the channel is $10\text{-}17_8$.

2. Detailed Analysis.

a) Data Flow Block Diagram. Refer to figure 8.6-9 for a block diagram description of the OD/EF operations with ESA.

During the last portion of the scan sequence, the externally specified address is inputted and placed in B. The circuitry which is normally used to compare the TACW with the CACW is still effective in examining $ZO_{15\text{-}0}$ and $B_{15\text{-}0}$. If $B_{15\text{-}0}$ = 0's, the output operations will terminate after this current request has been handled.

The I/O1-sequence obtains the requested word using the externally specified address in B. This word is outputted on the requesting odd channel and also on the next lower even channel. As normal, the output data/external function monitor interrupt signal can be generated upon termination if selected by bit 16 of this address word in B.

The continuous data mode sequence cannot be initiated. That is, the Continuous Data Request flip-flop is not set since the setting of this flip-flop requires $ZO_{17} = 1$.

The I/O2-sequence is used only if the output operation is terminated. It performs the storage of the externally specified address.

b) Essential Commands. Refer to table 8.6-11 for a sequential list of scan, I/O1, and I/O2-sequence events. Develop these commands by referring to the proper enable pages in the logic diagrams.

The events concerning priority selection are as previously discussed and are not shown. The OD-acknowledge and EF-acknowledge timing is the same as for normal output operations.

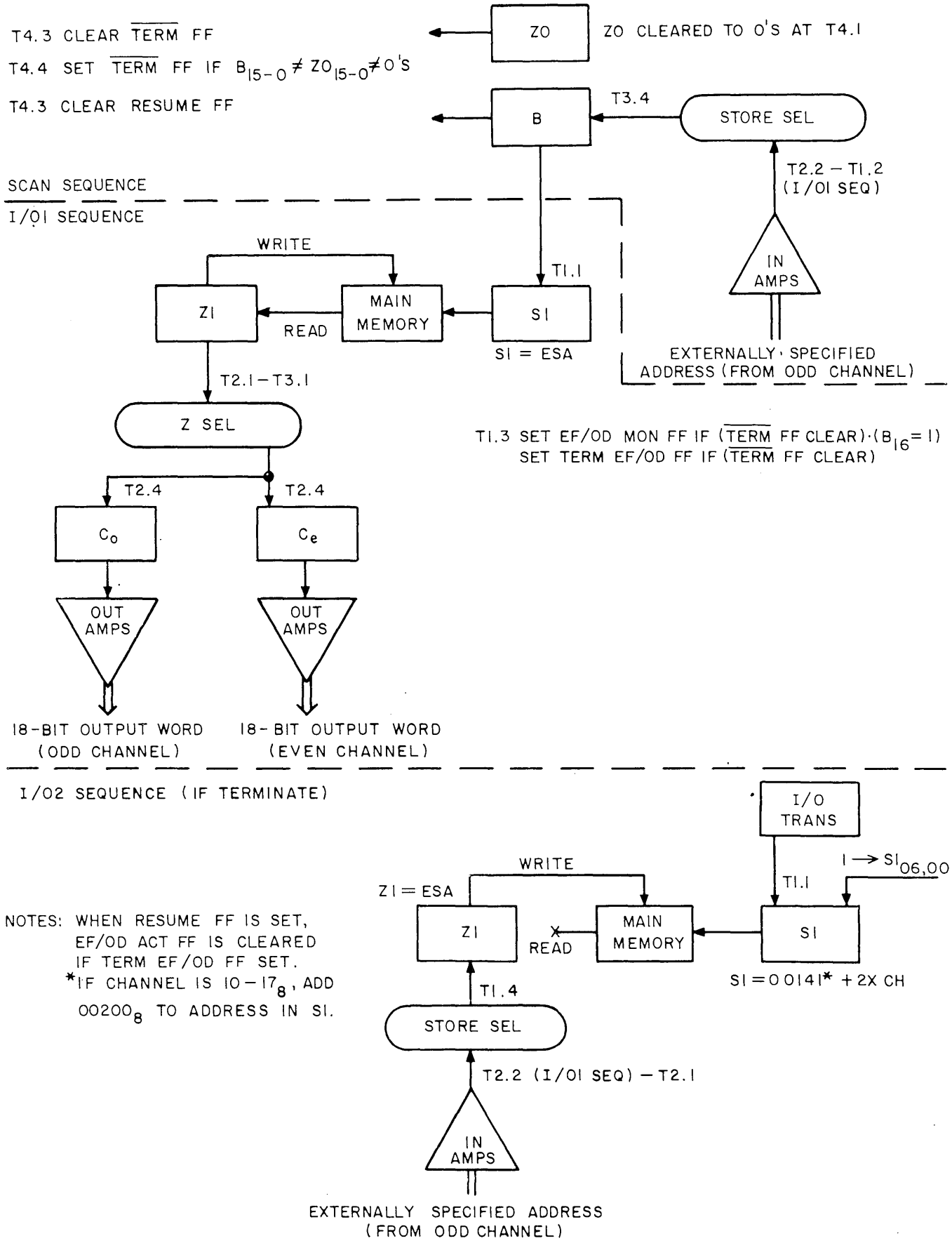


Figure 8.6-9. Scan, I/01, and I/02-Sequence Data Flow for OD/EF Operations with ESA

TABLE 8.6-11. SCAN, I/01, AND I/02 SEQUENCE ESSENTIAL COMMANDS FOR OD/EF OPERATIONS WITH ESA

TIME NOTATION	COMMANDS
	SCAN SEQUENCE
T2.1	Clear Fct, Chan, & I/O Trans 1
T2.2	Set $I/01_i$ ff; Pri→Fct, Chan, & I/O Trans 1 (set ESA ff)
	Odd chan Input Amps→Store Sel
T3.2	Clear B
T3.4	Store Sel→B
T4.1	Clear I/O Trans 2, clear ZO
T4.2	Set $I/01_f$ ff, I/O Trans 1→I/O Trans 2
T4.3	Clear $\overline{\text{Term}}$ ff, clear Resume ff
T4.4	Set $\overline{\text{Term}}$ ff if $B_{15-0} \neq ZO_{15-0}$, clear S1
	I/01 SEQUENCE
T1.1	B→S1, Init Memory
T1.2	Drop odd chan Input Amps→Store Sel, clear all EF/OD Ack Reg ff's
T1.3	Clear Z1, clear EF Req ff if EFR clear OD Req ff if (inter-cmptr)+(inter-cmptr·ODR) set EF/OD Control ff if EF mode, clear EF/OD Control ff if OD mode set Pri Alternator ff, set EF/OD Ack Reg ff set Ack Delay ff if $\overline{\text{Term}}$ ff set set Term EF/OD ff if $\overline{\text{Term}}$ ff clear set EF/OD Mon ff if ($\overline{\text{Term}}$ ff clear)·($B_{16=1}$)
T2.1	Clear $I/01_i$ ff, Z1→Z Sel
T2.2	Set $I/02_i$ ff if $\overline{\text{Term}}$ ff clear, odd chan Input Amps→Store Sel if $I/02_i$ ff set
T2.3	Clear C_o & C_e
T2.4	Z SEL→C_o & C_e
T3.1	Drop Z1→Z Sel
T4.1	Clear $I/01_f$ ff
The following occurs if $I/02_i$ ff set (terminate)	
T4.2	Set $I/02_f$ ff
T4.4	Clear S1, disable Mem→Z1
	I/02 SEQUENCE (if terminate)
T1.1	I/O Trans→S1, 1→$S1_{00}$, 1→$S1_{06}$, Init Memory
T1.2	Clear all EF/OD Ack Reg ff's
T1.3	Clear Z1, set EF/OD Ack Reg ff, set Ack Delay ff
T1.4	Store Sel→Z1
T2.1	Clear $I/02_i$ ff, drop odd chan Input Amps→Store Sel
T2.4	Drop disable Mem→Z1
T4.1	Clear $I/02_f$ ff

8.6-5. SUMMARY

The ODR and EFR signals are sent to the computer requesting a word to be outputted. According to the request type, a data word or control word is sent as indicated by the type of acknowledge signal accompanying the word. One of four modes can be selected on the CHANNEL FUNCTION switch (single channel, dual channel, ESI, or ESA). Mode can only be selected if the request occurs on an odd channel; otherwise, single channel mode is in effect.

Single channel operation involves one 18-bit word to be outputted on the requesting channel. Dual channel operation causes one 18-bit word to be sent on the requesting channel, which must be odd, and another on the next lower even channel.

ESI and ESA allow the external device to either indirectly or directly specify the address of the requested word. This word is sent on both the requesting odd channel as well as the next lower even channel. Since there is a common C register for the four odd channels on the same drawer and another C register common to the four even channels on the same drawer, the outputted word is actually sent over all eight channels of the particular drawer. However, the acknowledge signal is sent only to the requesting odd channel.

NAME: _____

8.6-6. STUDY QUESTIONS

a. Given: 74L00 constant low level output (logic diagrams, figure 9-69)

Describe the effect of this malfunction on OD operations. Indicate all channels affected.

b. Given: 4XV04 flip-flop, pin 12 grounded (logic diagrams, figure 9-68)

Describe the effect of this malfunction on OD operations for channel 6.

c. Given:

Address		Content
00043	=	406000
05777	=	123456
06000	=	701234
06001	=	567012

An ODR occurs on channel 5 and is honored. The CHANNEL FUNCTION switch is in the DUAL position.

1. Which are the two channels involved? _____ & _____

2. On which channel is the OD Acknowledge sent? _____

3. Give the data word sent on each of these two channels.

channel	data word
_____	_____
_____	_____

d. Given: Same as question c.

An ODR occurs on channel 5 and is honored. The CHANNEL FUNCTION switch is in the DUAL position. Consider the effect that each of the two following malfunctions would have on this operation. Indicate the channels involved and the data word sent on each of these two channels for each of these conditions. Assume that all C registers were cleared prior to this OD operation.

1. 44G00 grounded output (logic diagrams, figure 9-52)

channel	data word
_____	_____
_____	_____

2. 44G00 constant low level output

channel	data word
_____	_____
_____	_____

3. 43G00 constant low level output (logic diagrams, figure 9-52)

channel	data word
_____	_____
_____	_____

SECTION 8 - INPUT/OUTPUT SECTION

8.7. CONTINUOUS DATA MODE OPERATIONS

8.7-1. OBJECTIVES

To present the detailed theory of operation involved in continuous data mode operations.

8.7-2. INTRODUCTION

The continuous data mode sequence is initiated upon any buffer termination if requested by the $TACW_{17} = 1$. This sequence automatically reloads the two control memory addresses involved with a new TACW and CACW. The particular channel remains active such that operations can continue with a new buffer.

8.7-3. REFERENCES

a. UNIVAC 1219 Technical Manual, Volume I, Paragraph 4-3.

b. UNIVAC 1219 Technical Manual, Volume II, Section 9 (logic diagrams).

8.7-4. INFORMATION

a. General Description. The continuous data mode (CDM) is program selectable. If the $TACW_{17} = 1$, this CDM operation is initiated upon buffer termination. The purpose of CDM is to automatically, without program operation required, provide a new buffer when one has terminated without deactivating the channel. That is, a new predetermined TACW and CACW replace those which have terminated. The CDM operation effectively makes for a continuous ID, OD, or EF buffer as seen by the communicating external device.

The new reload TACW can request another CDM operation upon its termination by the $TACW_{17} = 1$. Even though the channel is not made inactive, the monitor interrupt can occur as normal upon each buffer completion if requested by the $CACW_{16} = 1$. This interrupt notifies the program that a buffer has been completed and that the new reload TACW and CACW are currently being used. If the new $TACW_{17}$ requests another CDM operation, the program probably will set up the next reload TACW and CACW. The size of the current buffer determines the amount of time that the program has in which to set up the next reload address control words. Also, each new reload TACW and CACW can specify the buffer direction (forward or backward) with $CACW_{17}$.

On a particular channel, the CDM operation should be used with only one type of buffer at one time. This restriction exists because there is only one pair of control memory addresses to hold the reload TACW and CACW for each channel. These reload words are used for the ID, OD, or EF operations on that channel depending upon which of these operations requests the CDM reload function.

The reload TACW is held in control memory at the address 00020_8 + 2x channel # if the channel is 0-7 or 00220_8 + 2x channel # if the channel is $10\text{-}17_8$. The reload CACW is held in control memory at the address 00021_8 + 2x channel # if the channel is 0-7 or 00221_8 + 2x channel # if the channel is $10\text{-}17_8$. During the CDM-sequence, these reload address words are placed in the normal control memory addresses reserved for the ID, OD, or EF operation on the particular channel which terminated.

b. Detailed Analysis.

1. Data Flow Block Diagram. Refer to figure 8.7-1 for a block diagram description of the CDM-sequence operations.

The CDM-sequence is initiated in parallel with the last portion of either the I/O1 or I/O2-sequence when control memory is available for use. The first control memory reference, at T3.1 time, obtains the reload CACW which is placed in ZO. A second memory reference stores this word from ZO. The storage address is the address which is reserved for the CACW as used by the ID, OD, or EF operation.

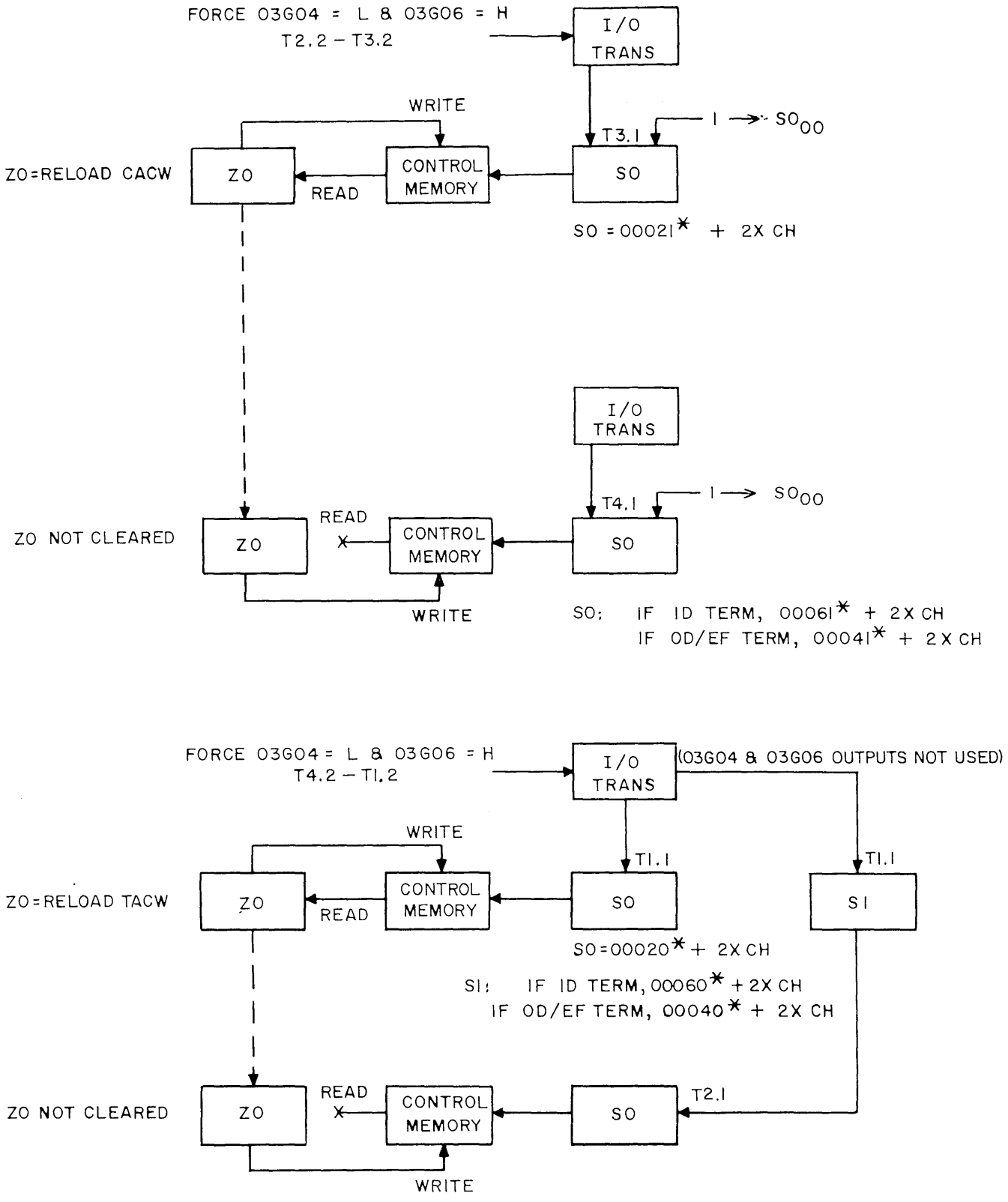
The address used to obtain the reload CACW is dependent upon the channel number which initiated the CDM-sequence. The channel number and type of I/O operation information is still contained in the I/O-translator. Thus, the I/O-translator is used to formulate the proper control memory addresses in SO. SO_{00} is set to 1_2 such that the reload CACW origin and destination addresses are odd.

The remainder of the CDM-sequence involves two more control memory references which obtain and store the reload TACW. The major difference from this operation and that handling the reload CACW is the addresses placed in SO. SO_{00} is not set, thus the reload TACW origin and destination addresses are even, in each case one address lower than that for the reload CACW. The address placed in SO at T2.1 time is transferred by way of S1 because there is no direct transfer into SO from the I/O-translator at this time.

2. Essential Commands. Refer to table 8.7-1 for a sequential list of essential CDM-sequence events. Develop the commands shown by referring to the proper enable pages in the logic diagrams.

8.7-5. SUMMARY

The CDM-sequence is initiated during either the I/O1 or I/O2-sequence if the buffer currently being handled has been completed ($TACW_{15-0} = CACW_{15-0}$) and the $TACW_{17} = 1$. When initiated, this operation prevents the clearing of the particular active flip-flop and provides a continuation of the same I/O operation on the same channel. The continued I/O operation, however, involves a new buffer area because of the new TACW and CACW which are set up by this sequence. The monitor interrupt operation may still occur upon each buffer completion as normal.



NOTE: IF CHANNEL IS $10-17_8$, ADD 00200_8 TO THE ADDRESS

Figure 8.7-1. Continuous Data Mode Sequence Data Flow

TABLE 8.7-1. CONTINUOUS DATA MODE SEQUENCE ESSENTIAL COMMANDS

TIME NOTATION	COMMANDS
T4.2	Set $I'O_f$ ff (1 or 2)
T4.4	Set Cont Data Req ff if $(B_{15-0} = ZO_{15-0}) \cdot (ZO_{17} = 1)$
(wait until can set Cont Data Seq ff)	
T2.2	Set Cont Data Seq ff if $(\overline{I/O1_f}$ ff set) · (Dual ff set + ESI · $\overline{\text{Term}}$ ff clear) Force 03G04 = L & 03G06 = H*
Following in parallel with I/O1-sequence if not dual channel and not ESI terminate or I/O2-sequence.	
T3.1	Set 2XG28 ff, I/O Trans → SO, 1 → SO_{00}, Init CM, Clear ZO
T3.2	Drop force 03G04 & 03G06*
T3.3	Disable CM → ZO
T4.1	I/O Trans → ZO, 1 → SO_{00}, Init CM
T4.2	Force 03G04 = L & 03G06 = H*
T4.3	Drop disable CM → ZO
T4.4	Clear S1
T1.1	I/O Trans → SO, Init CM, clear ZO, I/O Trans → S1
T1.2	Drop force 03G04 & 03G06*
T1.3	Disable CM → ZO
T2.1	Clear Cont Data Seq ff, clear Cont Data Req ff, S1 → SO Init CM
T2.3	Clear 2XG28 ff, drop disable CM → ZO

*03G04 and 03G06 (Function Translator) are gates involved in the I/O Trans → SO transfer. These gates do not affect the I/O Trans → S1 transfer.

NAME: _____

8.7-6. STUDY QUESTIONS

a. Given: 12G28 constant low level output (logic diagrams, figure 9-34)

Describe the effect that this malfunction would have upon the CDM-sequence which is initiated by an ID buffer termination.

b. Given: 2XG28 flip-flop, pin 13 grounded (logic diagrams, figure 9-34)

Describe the effect that this malfunction would have upon the CDM-sequence which is initiated by an OD buffer termination.

SECTION 8 - INPUT/OUTPUT SECTION

8.8. INSTRUCTION EXECUTION OF OUTOV, EXFOV

8.8-1. OBJECTIVES

To present the detailed theory of operation involved in the execution of instructions with f = 50:26, 50:27.

8.8-2. INTRODUCTION

These instructions force out either an OD or EF word depending on which of these buffer types was set up last.

8.8-3. REFERENCES

a. UNIVAC 1219 Technical Manual, Volume I, Paragraph 4-3.

b. UNIVAC 1219 Technical Manual, Volume II, Section 9 (logic diagrams).

8.8-4. INFORMATION

a. General Description.

1. Instruction Interpretation. Actually, these instructions are the same. They force out either an OD or EF word as indicated by the type of acknowledge signal sent with the word. The acknowledge type is determined by the state of the EF Mode flip-flop. This flip-flop is controlled by the last type (OD or EF) buffer set up by the program. The particular buffer does not need to be active.

The output operation is accomplished by simulating an ODR or EFR. As normal, the origin of the output word is determined by the CACW. Also, the $CACW_{15-0}$ is modified by ± 1 as specified by the $CACW_{17}$. Normal EF/OD Monitor interrupt operations and CDM operations can occur. As for normal ODR/EFR operations, channel mode can be selected by the CHANNEL FUNCTION switch if the channel is odd. The channel is specified by k of the instruction word. Single or dual channel mode can be used; but if the CHANNEL FUNCTION switch is in the ESI or ESA position, single mode is used.

2. Execution Sequences.

a. I-sequence. During the I-sequence which obtains the instruction from memory, the channel number and function type (EF/OD) information is placed in the I/O-translator as would normally be done by an ODR or EFR.

b) Remaining Sequences. If output is not available on the particular chassis, the sequences are disabled by the Hold flip-flops. This "hold" condition remains until output is available, after which the word is outputted with the use of the I/O1 and I/O2-sequences. I/O2-sequence is used for dual channel only.

If a resume-fault interrupt is present, the no-resume condition on the particular chassis will cause output to be not available. In this case, the Hold flip-flops are cleared to allow the sequences to continue and the resume-fault interrupt will be honored. The word is not outputted; however, P is decremented back to the address of the same f = 50:26, 50:27 instruction. Thus, when the program resumes after the interrupt is honored, this same instruction will again attempt to force out the same word.

b. Detailed Analysis.

1. First I-Sequence.

a) Data Flow Block Diagram. Refer to figure 8.8-1 for a block diagram description of the I-sequence.

Most of the I-sequence operations are as previously described. If necessary, refer to study guide sheet number 5.4 for a detailed description.

KO receives the channel number. This value with the EF/OD function information is placed in the I/O-translator 1 at T3.2 time.

As discussed in a prior sheet, the instruction could be obtained from bootstrap or control memory.

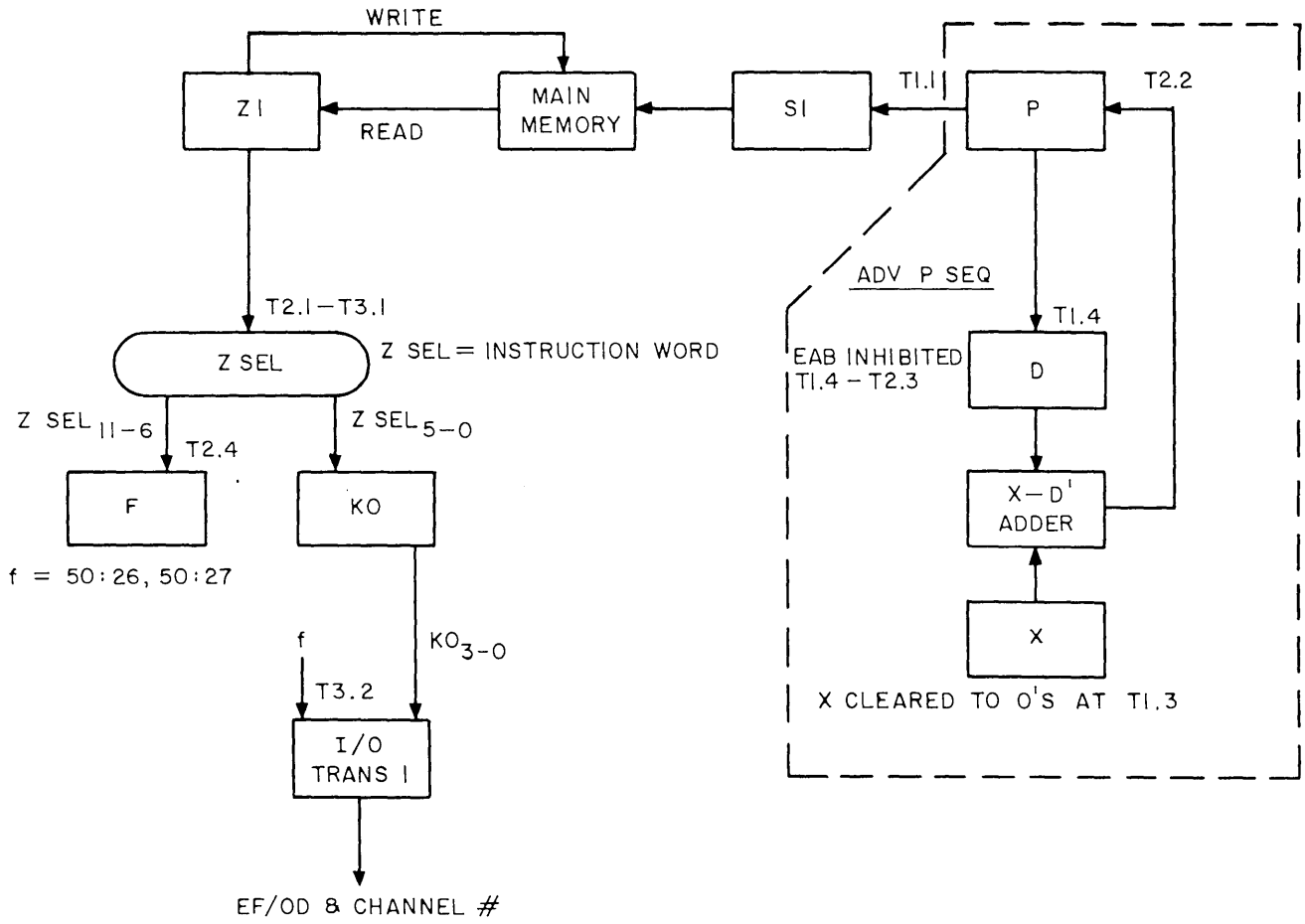
b) Essential Commands. Refer to table 8.8-1 for a sequential list of essential I-sequence events. Develop these commands by referring to the proper enable pages in the logic diagrams.

2. Remaining Sequences with Output Available and No-Resume Fault. With this condition, the word is outputted with the use of the I/O-sequences. Prior to the I/O1-sequence, the instruction sequences are held disabled for a main timing cycle. This allows time for the setting of the $I/O1_i$ flip-flop and performance of the I/O1 operations that would normally occur during the scan sequence.

Refer to table 8.8-2 for a sequential list of essential events. Develop these commands by referring to the proper enable pages in the logic diagrams.

3. Remaining Sequences with Output Not Available and No Resume Fault. With this condition, the instruction sequences must be held disabled until output becomes available. When output is available, the Hold flip-flops are cleared and the normal I/O-sequence operations occur to output the word.

Refer to table 8.8-3 for a sequential list of essential events. Develop these commands by referring to the proper enable pages in the logic diagrams.



T3.4 SET HOLD 1 FF
T4.2 SET INT$_i$ FF IF RESUME FAULT INTERRUPT TO BE HONORED
T4.3 SET HOLD 2 FF

Figure 8.8-1. I-Sequence Data Flow for f = 50:26, 50:27

TABLE 8.8-1. I-SEQUENCE ESSENTIAL COMMANDS FOR f = 50:26, 50:27

TIME NOTATION	COMMANDS
T4.4	Clear S1
T1.1	P $\rightarrow$ S1, Init Memory, *set Incr P ff
T1.3	*Clear D, *clear X, clear Z1, *set OXL11 ff, clear f
T1.4	*$P_L \rightarrow D_L$, *$P_U \rightarrow D_U$, *set Inhib EAB ff, clear K0
T2.1	*Clear P, Z1 $\rightarrow$ Z Sel, *clear Incr P ff
T2.2	*Adder $\rightarrow$ P
T2.3	*Clear OXL11 ff, *clear Inhib EAB ff
T2.4	Z $Sel_{11-6} \rightarrow$ F, set OXF06 ff, Z $Sel_{5-0} \rightarrow$ K0
T3.1	Clear I/O Trans 1, drop Z1 $\rightarrow$ Z Sel
T3.2	f & K0 $\rightarrow$ I/O Trans 1
T3.4	Set Hold 1 ff
T4.1	Clear $I_i$ff
T4.2	Set $I_i$ff, set Int_iff if resume-fault to be honored
T4.3	Set Hold 2 ff

NOTE: These events are concerned with or are controlled by the advance-P subsequence.

4. Remaining Sequences with Resume Fault.

a) Essential Commands. With this condition, the sequence hold function terminates and the next I-sequence is allowed to run. If no higher priority I/O request is present, the interrupt sequence will also run next to honor the interrupt. Realize that the next I-sequence which runs will be under control of the interrupt sequence to cause the interrupt program jump.

P is also decremented back to the address of this instruction to allow a program return to this same instruction. Thus, a later attempt can be made to force out the same word. The decrement operations are discussed later in this sheet.

Refer to table 8.8-4 for a sequential list of essential events. Develop these commands by referring to the proper enable pages in the logic diagrams.

TABLE 8.8-2. SEQUENCES AFTER FIRST I-SEQUENCE ESSENTIAL COMMANDS FOR f = 50:26, 50:27 WITH OUTPUT AVAILABLE AND NO RESUME FAULT

TIME NOTATION	COMMANDS
	HELD SEQUENCE FOLLOWING NORMAL I-SEQUENCE OF INSTRUCTION
T2.1	Clear I/O Trans 1
T2.2	Set $I/O1_i$ ff, set clear Hold ff, f & KO → I/O Trans 1
T4.2	Set $I/O1_i$ ff, clear Hold 1, clear Clear Hold ff
	I/O1-SEQUENCE
T1.3	Clear Hold 2 ff

NOTES: First sequence shown is not under instruction sequence control. Sequences are disabled by the Hold 1 and 2 flip-flops set during the first I-sequence. The I/O operations performed during this time simulate those which occur during the scan sequence that precedes I/O1.

The I/O1-sequence performs normal EF/OD operations. Normal events which occur after the setting of $I/O1_i$ flip-flop are not shown. Normal I/O2-sequence follows if dual channel operation is in effect.

b) Decrement P Data Flow Block Diagram. Refer to figure 8.8-2 for a block diagram description of the decrementing function of the advance-P subsequence.

Prior to this operation, P has been advanced to the next address. Since the output function has not been accomplished and a resume-fault interrupt will be honored, it is necessary to decrement back to the address of this same instruction to execute it again after the interrupt operations.

The decrementing operations are similar tothose for incrementation. The P value to be decremented is placed in D. X is set to all 1's (-0). Thus, the subtraction of X-D' will not require an end-around borrow and the adder output should be (-0)-D' or D. This borrow is inserted to cause the adder output to be D-1 which is placed in P.

8.8-5. SUMMARY

The OUTOV and EXFOV instructions cause a word to be forced out on the channel specified by k. The word type (OD or EF) is indicated by the acknowledge signal sent with the word. The type of acknowledge is selected by the EF Mode flip-flop which has been set or cleared by the last EF or OD buffer set up by the program. Actually, then, these instructions are the same. Either one forces a word, the type of which is determined by the last output buffer initiated (OD or EF). The buffer does not need to be active. Channel mode can be single or dual only.

TABLE 8.8-3. SEQUENCES AFTER FIRST I-SEQUENCE ESSENTIAL COMMANDS FOR f = 50:26, 50:27 WITH OUTPUT NOT AVAILABLE AND NO RESUME FAULT

TIME NOTATION	COMMANDS
	HELD SEQUENCES FOLLOWING NORMAL I-SEQUENCE OF INSTRUCTION
(wait for Output Available)	
Output Available	
next T2.1	Clear I/O Trans 1
T2.2	Set $I/O1_i$ ff, set Clear Hold ff, f & KO → I/O Trans 1
T4.2	Clear Hold 1 ff, set $I/O1_f$ ff, clear Clear Hold ff
	I/O1-SEQUENCE
T1.3	Clear Hold 2 ff

NOTES: Until output is available, no instruction sequence is in control. Sequences are disabled by the Hold 1 and 2 flip-flops set during the first I-sequence. The I/O operations performed after the setting of $I/O1_i$ flip-flop and prior to the I/O1-sequence simulate those which occur during the scan sequence that precedes I/O1.

The I/O1-sequence performs normal EF/OD operations. Normal events which occur after the setting of $I/O1_i$ flip-flop are not shown. Normal I/O2-sequence follow if dual channel operation is in effect.

TABLE 8.8-4. SEQUENCES AFTER FIRST I-SEQUENCE ESSENTIAL COMMANDS FOR f = 50:26, 50:27 WITH RESUME FAULT

TIME NOTATION	COMMANDS
	HELD SEQUENCE FOLLOWING NORMAL I-SEQUENCE OF INSTRUCTION
T1.4	*Set Decr P ff, *set Insert EAB ff
	*Arith Sel & Arith Sel' $\longrightarrow$ X (set X = all 1's)
T2.2	Set Clear Hold ff
T2.3	*Set 0XL11 ff, *clear Insert EAB ff, *clear D
T2.4	*Set Insert EAB ff, *$P_L \longrightarrow D_L$, *$P_U \longrightarrow D_U$
T3.1	*Clear P, *clear Decr P ff
T3.2	*Adder $\longrightarrow$ P
T3.3	*Clear 9XL11 ff
T4.2	Clear Hold 1 ff, clear Clear-Hold ff
T4.3	*Clear Insert EAB ff
	INTERRUPT AND I-SEQUENCE IN PARALLEL
T1.3	Clear Hold 2 ff
T2.1	Clear I_f ff
T2.2	Set I_f ff, set Int_f ff

*These events are concerned with or are controlled by the decrementing function of the advance-P subsequence.

First sequence shown is not under sequence control. Sequences are disabled by the Hold 1 and 2 flip-flops set during the first I-sequence.

Second sequence is a normal interrupt sequence to honor the resume-fault if no higher priority signal is present. All commands for this sequence are not shown.

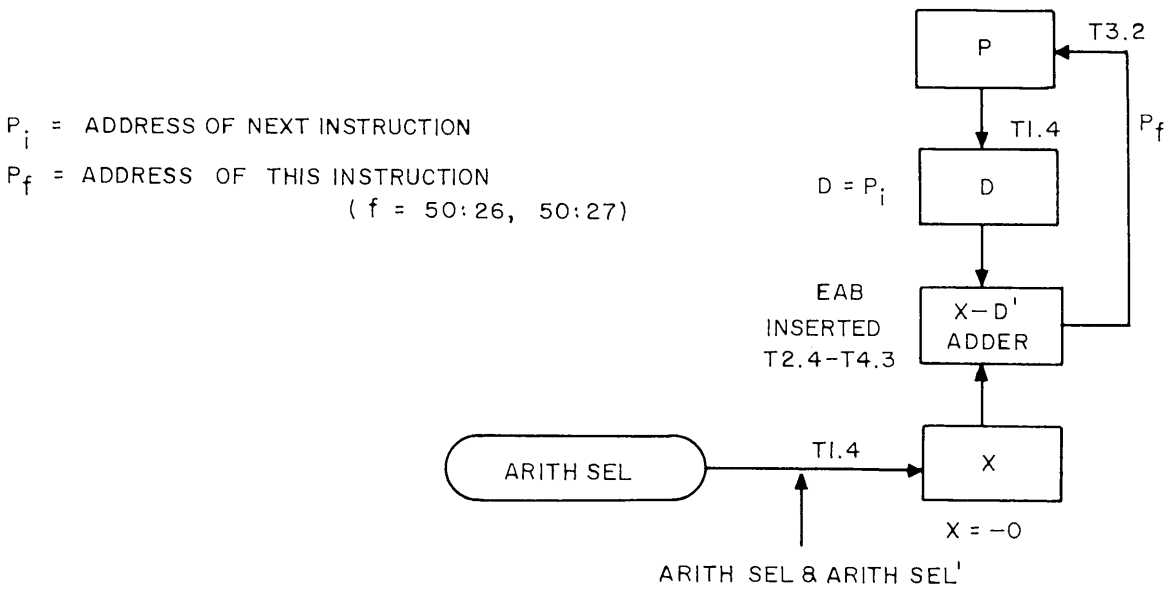


Figure 8.8-2. P Decrement for f = 50:26, 50:27 with Resume Fault

NAME: _____

8.8-6. STUDY QUESTIONS

a. Given: 22E00 grounded output (logic diagrams, figure 9-49)

Describe the effect that this malfunction would have on the execution of the f = 50:26 instruction.

SECTION 8 - INPUT/OUTPUT SECTION

8.9. INSTRUCTION EXECUTION OF SKPIIN, SKPOIN, SKPFIN, SKPNR, SRSM

8.9-1. OBJECTIVES

To present the detailed theory of operation involved in the execution of the SKPIIN, SKPOIN, SKPFIN, SKPNR, and SRSM instructions.

8.9-2. INTRODUCTION

These instructions provide means for checking the activeness of I/O buffers on any channel, checking the resume condition on any chassis, and forcing the resume condition on any chassis.

8.9-3. REFERENCES

a. UNIVAC 1219 Technical Manual, Volume I, Paragraph 4-3.

b. UNIVAC 1219 Technical Manual, Volume II, Section 9 (logic diagrams).

8.9-4. INFORMATION

a. General Description.

1. Instruction Interpretation.

a) SKPIIN, f = 50:21. This instruction senses the status of the ID Active flip-flop for channel k. If this flip-flop is clear, the next sequential instruction is skipped.

b) SKPOIN, f = 50:22. This instruction senses the status of the OD buffer for channel k. If the EF Mode flip-flop is clear and the EF/OD Active flip-flop is set, the next sequential instruction is skipped.

c) SKPFIN, f = 50:23. This instruction senses the status of the EF buffer for channel k. If the EF Mode flip-flop is set and the EF/OD Active flip-flop is set, the next sequential instruction is skipped.

d) SKPNR, f = 50:57. This instruction senses the status of the output available condition on the I/O chassis group which includes channel k. The next sequential instructionis skipped if output is not available. On any I/O group, output is available if an OD or EF-acknowledge signal is not currently being sent on a channel of that group and the Resume flip-flop on that chassis is set.

For non inter-computer operation, the condition of output not available only exists for the duration of the acknowledge since the Resume flip-flop is set by the acknowledge timing.

For inter-computer operation, the Resume flip-flop is initially set; however, as each word is outputted, it is cleared. The ID-acknowledge from the receiving computer, which appears to the sending computer as an ODR, sets the Resume flip-flop. The output not available or no resume condition, then, is an indication that the last word sent to the receiving computer has not been accepted (acknowledged).

e) SRSM, f = 50:20. This instruction sets the Resume flip-flop on the I/O chassis group which includes channel k. On this chassis output then becomes available if an OD or EF-acknowledge is not currently being sent.

This instruction probably would be executed after the occurrence of the resume-fault special interrupt. The resume-fault interrupt occurs if the Resume flip-flop remains clear for a specific period of time. This interrupt, then, occurs in the inter-computer mode and indicates that the receiving computer did not accept the last outputted word within the allowed time period. The Resume flip-flop in the clear state causes output to be not available on its chassis and prevents further OD and EF operations on that chassis. The SRSM instruction re-establishes the output available and resume condition.

2. Execution Sequence (I). All operations are performed within the I-sequence. Only the one memory reference to obtain the instruction is necessary.

b. Detailed Analysis.

1. Data Flow Block Diagram. Refer to figure 8.9-1 for a block diagram description of the execution of f = 50:20 - 50:23, 50:57.

Most of the I-sequence operations are as previously described. If necessary, refer to study guide sheet number 5.4 for a detailed description.

K0 and I/O translation 1 receive the channel and the I/O chassis selection. This information is specified by the four least significant bits of K.

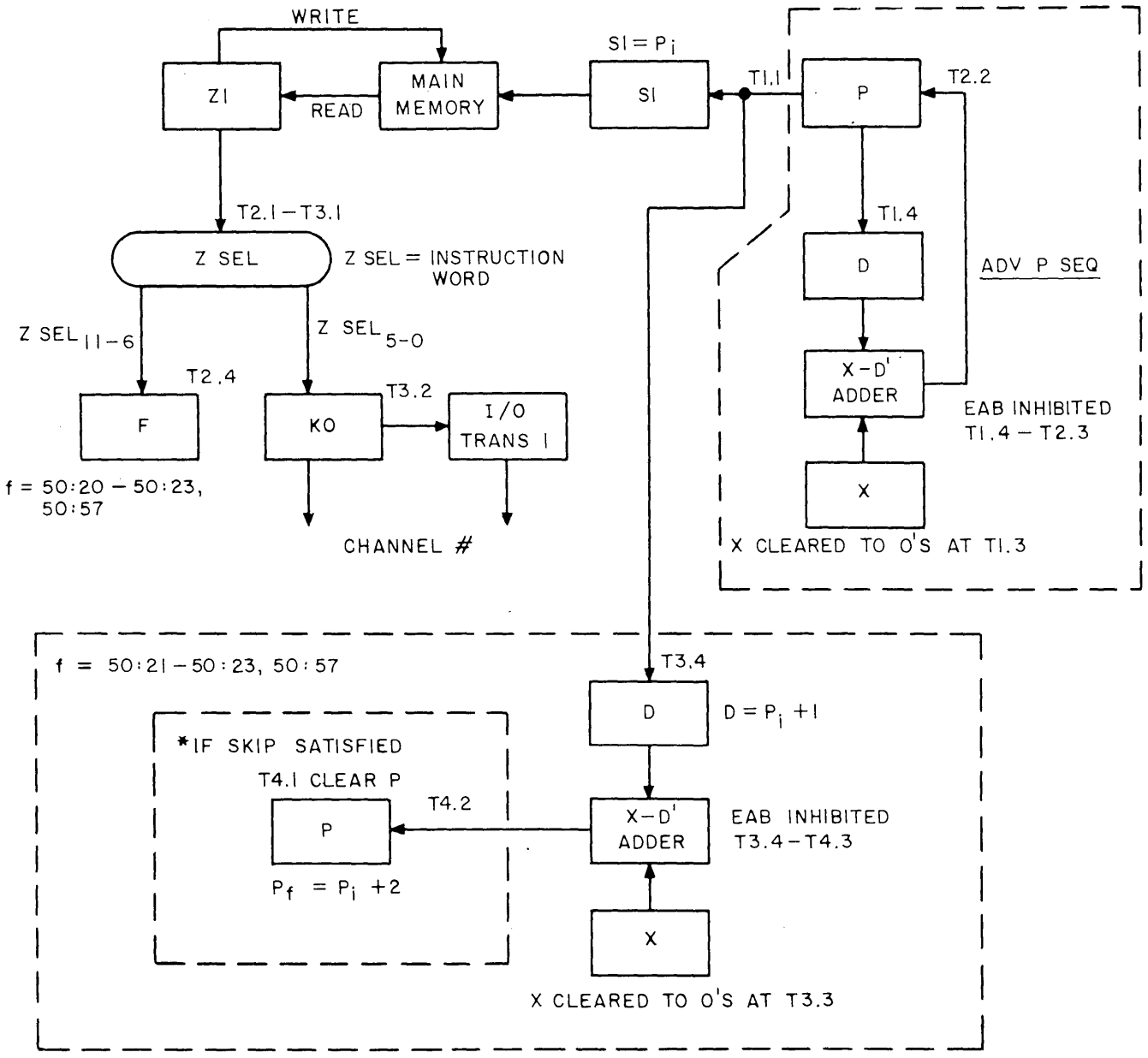
For f = 50:21 - 50:23, 50:57, the X-D' adder is used to increment P by +1 a second time just as is done by the advance-P subsequence. The second clearing of P and transfer into P from the adder is conditioned by the buffer activeness or output available condition as determined by the function code. If P receives the result of this second incrementation, the next sequential instruction will be skipped.

For f = 50:20, the Resume flip-flop on the chassis specified by k is set at T4.1 time. As discussed in a prior sheet, the instruction could be obtained from bootstrap or control memory.

2. Essential Commands. Refer to table 8.9-1 for a sequential list of essential I-sequence events. Develop these commands by referring to the proper enable pages in the logic diagrams.

8.9-5. SUMMARY

The SKPIIN, SKPOIN, SKPFIN, SKPNR instructions provide programed means of determining whether or not a channel is active (ID, OD, or EF) and if output is available (resume condition) on a particular I/O chassis group. The SRSM instruction forces the resume condition on the specified I/O chassis.



IF f = 50:20: T4.I SET RESUME FF

NOTE: *SKIP IF: (f = 50:21)·(ID ACT FF CLEAR)
(f = 50:22)·(EF/OD ACT FF SET)·(EF MODE FF CLEAR)
(f = 50:23)·(EF/OD ACT FF SET)·(EF MODE FF SET)
(f = 50:57)·(OUTPUT NOT AVAILABLE)

Figure 8.9-1. I-Sequence Data Flow for f = 50:20-50:23, 50:57

TABLE 8.9-1. I SEQUENCE DATA FLOW FOR f = 50:20-50:23, 50:57

TIME NOTATION	COMMANDS	f = 50:21 50:22 50:23 50:57	50:20
T4.4	Clear S1	X	X
T1.1	P → S1, Init Memory, *set Incr P ff	X	X
T1.3	*Clear D, *clear X, clear Z1, clear F, *set OXL11 ff	X	X
T1.4	$*P_L \rightarrow D_L$, $*P_U \rightarrow D_U$, *set Inhib EAB ff	X	X
T2.1	*Clear P, Z1 → Z Sel, *clear Incr P ff	X	X
T2.2	*Adder → P	X	X
T2.3	*Clear OXL11ff, *clear Inhib EAB ff	X	X
T2.4	$Z\ Sel_{11-6} \rightarrow F$, set OXF06 ff	X	X
T3.1	Drop Z1 → Z Sel, clear I/O Trans 1	X	X
T3.2	f & K0 → I/O Trans 1	X	X
T3.3	Clear D, clear X	X	
T3.4	$P_L \rightarrow D_L$, $P_U \rightarrow D_U$, set Inhib EAB ff	X	
T4.1	Set Resume ff		X
	Clear P if skip satisfied **	X	
T4.2	Adder → P if skip satisfied **	X	
T4.3	Clear Inhib EAB ff	X	

* These events are concerned with or are controlled by the advance-P subsequence.

** Skip condition is satisfied if:

(f = 50:21) · (ID Act ff clear)
(f = 50:22) · (EF/OD Act ff set) · (EF Mode ff clear)
(f = 50:23) · (EF/OD Act ff set) · (EF Mode ff set)
(f = 50:57) · (output not available)

NAME: _____

8.9-6. STUDY QUESTIONS

a. Given: 10J12 grounded output (logic diagrams, figure 9-7) instruction = 502103

Describe the effect that this malfunction would have upon the execution of the given instruction.

SECTION 8 - INPUT/OUTPUT SECTION

8.10. EXTERNAL INTERRUPT REQUEST OPERATIONS

8.10-1. OBJECTIVES

To present the detailed theory of operation involved in external interrupt request honoring.

8.10-2. INTRODUCTION

The external interrupt is a signal from an external device which requests a program jump to a special address and the acceptance of an inputted code word.

8.10-3. REFERENCES

a. UNIVAC 1219 Technical Manual, Volume I, Paragraph 4-3.

b. UNIVAC 1219 Technical Manual, Volume II, Section 9 (logic diagrams).

8.10-4. INFORMATION

a. General Description.

1. Single Channel Operation. The external interrupt signal (EIR) from an external device may be considered as a request. It requests that the computer accept the inputted 18-bit word from the input data lines which usually accompanies the EIR signal. This code word is sometimes referred to as a status word. Its bit configuration may indicate data handling errors, effect control over a certain program, etc. However, the EIR signal may, in some cases, be the desired indication itself without a code word. For example, it may signify the completion of a specific time period to synchronize the computer program operations with real time.

In addition to the storage of the code word, the EIR signal causes the setting of the EI Monitor flip-flop for the particular channel. This flip-flop generates the internal EI Mon signal, which is handled as a separate request for input/output services. This signal is detected during the interrupt scan period. When honored, the EI Mon signal causes a program jump to the address 00100_8 + 2 x channel if the channel is 0-7 or 00300_8 + 2 x channel if the channel is $10\text{-}17_8$.

As discussed in a later sheet, the External Interrupt Lockout flip-flop being set (by f = 50:36, 50:37) presents the detection of both the EIR and EI Mon on any channel. Therefore, the honoring of these signals can be delayed for as long as is desired after which this lockout flip-flop can be cleared by f = 50:32, 50:33. During the EIR honoring operations, the code word is stored and the EI Mon flip-flop is set. Once the EI Mon flip-flop is set, it prevents the detection of another EIR signal on the same channel. The EI Mon flip-flop is not cleared until its

signal is honored; thus, the EIR signal occurrence is recorded by this flip-flop until the program jump is effected by the honoring of the EI Mon signal.

During the honoring of the EIR signal, the code word is stored in memory at address $0010l_8$ + 2 x channel if the channel is 0-7 or $0030l_8$ + 2 x channel if the channel is $10\text{-}17_8$.

2. Dual Channel, ESI, or ESA Operation. If the CHANNEL FUNCTION switch is positioned so as to select either the dual channel, ESI, or ESA modes, dual channel operation is instated. The only difference in the resulting EIR signal honoring operations from those with single channel selected is the storage of a 36-bit inputted code word instead of an 18-bit word.

First, the EIR-carrying odd channel half of the code word is stored in memory at address $0010l_8$ + 2 x channel if the channel is 0-7 or $0030l_8$ + 2 x channel if the channel is $10\text{-}17_8$. Then, the next lower even channel half of the code word is stored in memory at address $0010l_8$ + 2 x even channel if the channel is 0-7 or $0030l_8$ + 2 x even channel if the channel is $10\text{-}17_8$. The storage address for the second half of the code word is dependent upon the even channel number.

b. Detailed Analysis.

1. Data Flow Block Diagram. Other than those events concerned with the selection of the EIR signal and setting the translators accordingly, no essential data flow occurs during the Scan Sequence. Refer to figure 8.10-1 for a block diagram description of the EIR operations.

If single channel operation is in effect, only the I/O1-sequence is used. A memory cycle is initiated to store the 18-bit code word from the data lines. If the Dual flip-flop is set (dual channel, ESI, or ESA), the half of the 36-bit code word from the odd channel is stored.

If the Dual flip-flop is set, the I/O2-sequence is used to store the second half of the 36-bit code word in memory.

2. Essential Commands. Refer to table 8.10-1 for a sequential list of essential scan, I/O1, and I/O2-sequence events. Develop these commands by referring to the proper enable pages in the logic diagrams.

The events concerning priority selection are as previously discussed and are not shown. The ID-acknowledge timing is the same as presented for ID honoring operations.

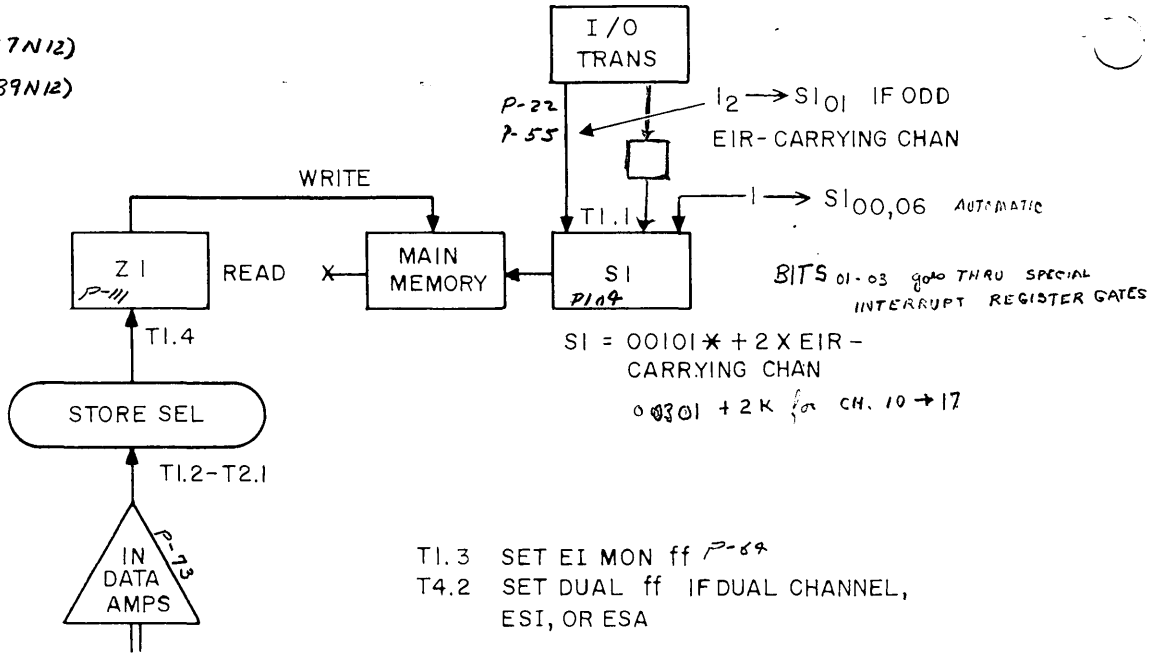
8.10-5. SUMMARY

When sent to the computer, the EIR signal causes the storage of the word on the associated input data lines. Depending upon the peripheral device, this may or may not have meaning. The ID-acknowledge signal is sent back to the peripheral device indicating the detection of the EIR and acceptance of the coded status word.

Also upon detection of an EIR signal, the EI Monitor flip-flop is set for the same channel. The resulting EI Mon interrupt signal is presented to priority and, when selected, causes a program jump which notifies the program of the occurrence of the EIR signal.

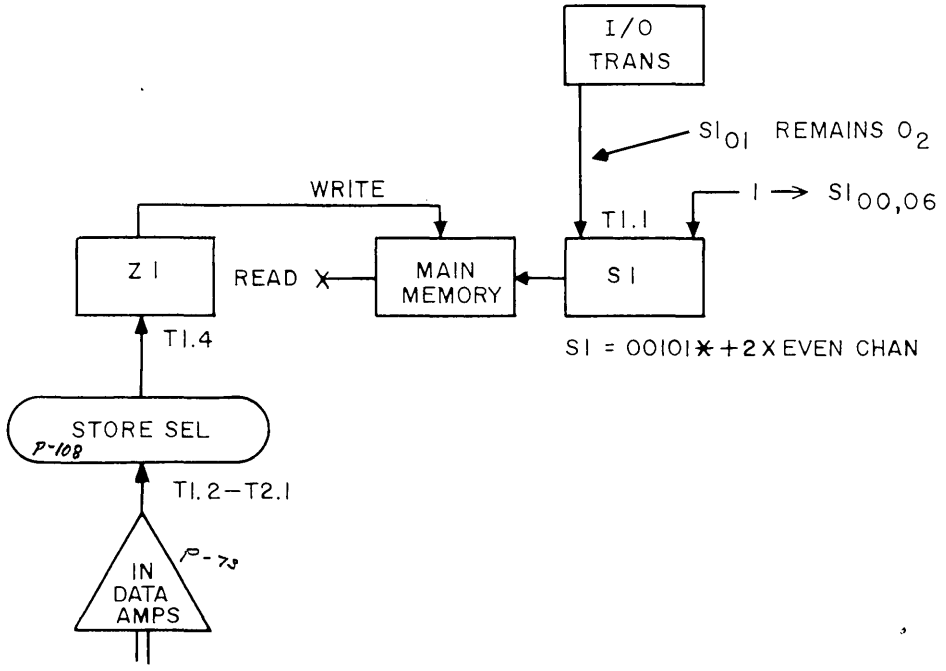
If the EIR signal occurs on an odd channel and the associated CHANNEL FUNCTION switch selects either dual channel, ESI, or ESA, dual channel operation is put into effect. Dual channel operation stores both input words from the odd channel and the next lower even channel. In this case, the ID-acknowledge signal is not generated until both words have been accepted.

P-22
TRANSLATOR → S1
XLATOR → S1 01-03 (37N12)
XLATOR → S1 04, 05, 07 (39N12)
78N12 outputs H
79N12 outputs L
SET S1 06 (35N12)
SET S1 00 (58N12)



I/01 SEQUENCE

I/02 SEQUENCE (ONLY IF DUAL CHANNEL, ESI, OR ESA)



NOTE: * IF CHANNEL IS $10-17_8$, ADD 00200_8 TO ADDRESS IN S1

Figure 8.10-1. I/01 and I/02-Sequence Data Flow for EI Operations

TABLE 8.10-1. SCAN, I/01, AND I/02 SEQUENCE ESSENTIAL COMMANDS FOR EI OPERATIONS

TIME NOTATION	COMMANDS
	SCAN SEQUENCE
T2.1	Clear Fct, Chan, & I/O Trans 1
T2.2	Set I/O_i ff, Pri→Fct, Chan, & I/O Trans 1
T3.1	Clear ZO
T3.2	Clear B
T3.4	ZO→B
T4.1	Clear I/O Trans 2, clear B ± 1 ff
T4.2	Set $I/01_f$ ff, clear all ID Ack Reg ff's, I/O Trans 1→I/O Trans 2
	Set Dual ff if dual, ESI, or ESA
T4.3 —	Clear EI Req ff, set ID Ack Reg ff, set 6XLg0 ff
	Clear Pri Alternator ff
T4.4	Clear S1, disable Mem→Z1
	I/01 SEQUENCE
T1.1	I/O Trans→S1**, 1→$S1_{00}$, 1→$S1_{06}$, Init Memory
T1.2	Input Data Amps→Store Sel*
T1.3	Set EI Mon ff, clear Z1
T1.4	Store Sel→Z1, set ID Ack Gener ff if Dual ff clear
T2.1	Clear $I/01_i$ ff, drop Input Data Amps→Store Sel
T2.2	Set $I/02_i$ ff if Dual ff set
T2.4	Drop disable Mem→Z1
The following occurs if $I/02_i$ ff set (dual channel) except "clear $I/01_f$ ff"	
T3.1	Clear ZO
T3.2	Clear B
T3.4	ZO→B
T4.1	Clear $I/01_f$ ff, clear I/O Trans 2, clear B ± 1 ff
T4.2	Set $I/02_f$ ff, I/O Trans 1→I/O Trans 2, clear all ID Ack Reg ff's
	Set Dual ff if dual, ESI or ESA
T4.3	Set ID Ack Reg ff, set 6XLg0 ff
T4.4	Clear S1, disable Mem →Z1
	I/02 SEQUENCE (if dual channel)
T1.1	I/O Trans→S1**, 1→$S1_{00}$, 1→$S1_{06}$, Init Memory
T1.2	Input Data Amps→Store Sel*
T1.3	Clear Z1
T1.4	Store Sel→Z1, set ID Ack Gener ff
T2.1	Clear $I/02_i$ ff, drop Input Data Amps→Store Sel
T2.4	Drop disable Mem→Z1
T4.1	Clear $I/02_i$ ff

*If operation is in dual channel, ESI or ESA mode, the channel inputted to Store Sel is as follows.

I/01 Seq.	I/02 Seq.
odd	even

**$S1_{01}$ remains 0_2 during I/02 Sequence but is set to 1_2 during I/01 sequence if EIR-carrying channel is odd.

SECTION 8 - INPUT/OUTPUT SECTION

8.11. REAL TIME CLOCK REQUEST OPERATIONS, GENERATION OF RTC MONITOR AND OVERFLOW INTERRUPTS, AND INSTRUCTION EXECUTION OF RTC

8.11-1. OBJECTIVES

To present the detailed theory of operation involved in real time clock operations.

8.11-2. INTRODUCTION

The real time clock is a 4 and 1/4 minutes clock that is updated at the rate of 1024 pps.

8.11-3. REFERENCES

a. UNIVAC 1219 Technical Manual, Volume I, Paragraphs 4-3 and 4-7, table 4-11.

b. UNIVAC 1219 Technical Manual, Volume II, Section 9 (logic diagrams).

8.11-4. INFORMATION

a. General Description.

1. Real Time Clock. Internal to the computer is an oscillator which is used to increment by +1 the content of control memory address 00015_8 at a constant rate. The oscillator is referred to as the real time clock. By periodic programed inspection of the contents of this address, theclock can be used to time program operations. This address can be entered by a store instruction to preset it to a specific count and later used as a real time reference. The oscillator frequency is 1024 cycles per second. Each cycle generates a real time clock request (RTC) signal which is honored by the input/output logic. The RTC honoring operations consist of incrementing by +1 the content of address 00015_8. Each bit position of the memory location represents twice the time value of the next less significant bit position. Refer to figure 8.11-1 for the bit position time weights.

The RTC signal is not presented to the input/output logic if the RTC Disconnect switch is in the "ON" position.

2. RTC, f = 50:14, Instruction and RTC Monitor Interrupt. A monitor of the clock count in control memory address 00015_8 is automatically performed if enabled by the RTC Monitor Request flip-flop being set. This flip-flop is set by the execution of f = 50:14.

The monitor function is effected by comparing the clock count each time that it is incremented with the content of control memory address 00014_8. A comparison count must be set in address 00014_8 by a store instruction. When the clock count is

incremented to equal the comparison count, the RTC Monitor flip-flop is set to generate the RTC monitor interrupt. This interrupt signal is handled by a separate input/output operation and causes a program jump to the control memory address 00012_8.

3. RTC Overflow Interrupt. When the clock count is 777777_8, the next RTC request will cause it to be incremented to 000000. This condition is referred to as real time clock overflow and results in the setting of the RTC Overflow flip-flop which generates the RTC overflow interrupt. This interrupt signal is handled by a separate input/output operation and causes a program jump to the control memory address 00013_8.

BIT POSITIONS	09	08	07	06	05	04	03	02	01	00
	1/2	1/4	1/8	1/16	1/32	1/64	1/128	1/256	1/512	1/1024
BIT POSITIONS	17	16	15	14	13	12	11	10		
	128	64	32	16	8	4	2	1		

Figure 8.11-1. RTC Address 00015_8 Bit Weights in Seconds

b. Detailed Analysis.

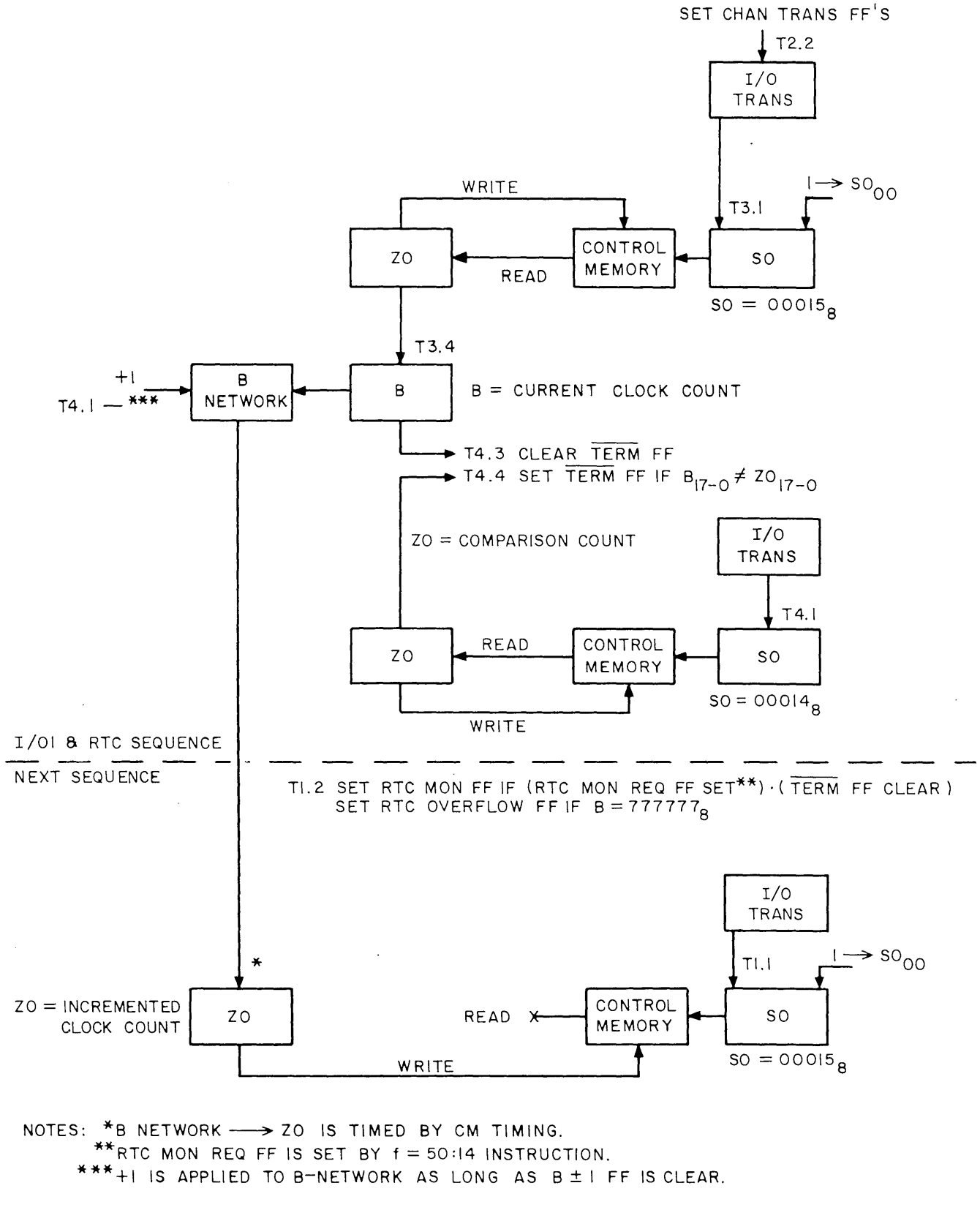
1. RTC Request Operations.

a) Data Flow Block Diagram. Refer to figure 8.11-2 for a block diagram description of RTC request operations.

The RTC request operations are performed during the I/O1-sequence with the RTC-sequence flip-flop set. At T3.1 time, a control memory reference is initiated to obtain the current clock count from address 00015_8. This address in SO is formulated by setting $SO_{3,2}$ from the two Channel Translator flip-flops. Both of these flip-flops are set at T2.2 time. SO_{00} is also set to complete the address 00015_8. The extracted current clock count is placed in B.

A second control memory reference is used to obtain the comparison count from address 00014_8. This address is formulated in SO in the same way as 00015_8 except that SO_{00} remains clear. From ZO and B, the current clock count is compared with the comparison count. If they are equal, the Terminate flip-flop remains clear. If the RTC Monitor Request flip-flop has not been set, the result of the comparison will have no effect.

The current clock count in B is incremented by +1 in the B-network and is restored in address 00015_8 by a third control memory cycle. This memory cycle actually occurs during the first portion of the next sequence. The incremented clock count is stored from ZO. The gating of control memory is disabled during the read portion of the control memory cycle which destroys the previous clock count.



NOTES: *B NETWORK ⟶ ZO IS TIMED BY CM TIMING.
**RTC MON REQ FF IS SET BY f = 50:14 INSTRUCTION.
***+1 IS APPLIED TO B-NETWORK AS LONG AS B ± 1 FF IS CLEAR.

Figure 8.11-2. I/O1 & RTC and Next Sequence Data Flow for RTC Operations

b) Essential Commands. Refer to table 8.11-1 for a sequential list of I/O1 & RTC and next sequence events for RTC request operations. Develop these commands by referring to the proper enable pages in the logic diagrams.

The setting of the RTC Overflow flip-flop is conditioned by the B-network. Refer to logic diagrams, figure 9-54. 10E17 must output a high level to set this flip-flop. Input 10B17 at a low level indicates a carry to B Network$_{17}$. Input 01B17 at a low level indicates $B_{17} = 1_2$. Therefore, a high level from 10E17 implies B = 777777_8. The B-network and the events concerning the honoring of the RTC monitor and RTC overflow interrupt signals are presented in later sheets.

Also shown in this figure are commands pertaining to the resume-fault interrupt signal which is timed by the real time clock. The fault indication is enabled by a carry signal from the B-network. The resume-fault interrupt is discussed in later sheets.

2. RTC, f = 50:14, Instruction. This instruction uses only the I-sequence. Refer to logic diagrams, figure 9-54.

The RTC Monitor Request flip-flop is set at T3.3 time by the f = 50:14 instruction. This flip-flop partially enables the setting of the RTC Monitor flip-flop. Notice that if the RTC Monitor flip-flop is set, the RTC Monitor Request flip-flop is cleared. Thus, for each RTC Monitor interrupt signal desired, a separate f = 50:14 instruction must be executed.

If necessary, refer to study guide sheet number 5.4 for a detailed description of the normal I-sequence operations.

8.11-5. SUMMARY

The RTC request is generated by an internal oscillator operating at a frequency of 1024 pps. The request causes the updating by +1 of the contents of the control memory address 00015_8. The associated RTC Monitor and RTC Overflow interrupts are used to notify the program of elapsed periods timed by the real time clock.

The clock operation, and therefore the clock interrupts, can be disabled with the RTC Disconnect switch.

TABLE 8.11-1. I/01 & RTC AND NEXT SEQUENCE ESSENTIAL COMMANDS FOR RTC OPERATIONS

TIME NOTATION	COMMANDS
RTC ocs. pulse	
The following occurs only if RTC Disconnect switch is off	
next Ø1	Set 0XG29 ff
RTC osc. pulse	
next Ø3	Set RTC Req ff
Ø4	
Ø1	Clear 0XG29 ff
	I/01 & RTC SEQUENCE
T2.2	Set $I/01_i$ ff, set RTC Seq ff, set Chan Trans ff's
T3.1	I/O Trans→SO, 1→SO_{00}, Init CM, set 3XG29 ff
	Clear RTC Req ff, clear ZO
T3.2	Clear B
T3.4	ZO→B
T4.1	I/O Trans→SO, Init CM, clear B ± 1 ff (+1→B Network*), clear ZO
T4.2	Set $I/01_f$ ff
T4.3	Clear $\overline{\text{Term}}$ ff
T4.4	Set $\overline{\text{Term}}$ ff if $B_{17-0} \neq ZO_{17-0}$, disable CM→ZO
	NEXT SEQUENCE
T1.1	I/O Trans→SO, 1→SO_{00}, Init CM, clear ZO
	Set Resume Fault ff if (carry→B $Network_{10}$)·(any 8XL00-8XL30 ff clear)
T1.2	Set RTC Mon ff if (RTC Mon Req ff set)·($\overline{\text{Term}}$ ff clear)
	Set RTC Overflow ff if B = 777777
	Clear all 8XL00-8XL30 ff's if carry→B $Network_{10}$
T1.3	Clear RTC Mon Req ff if RTC Mon ff set
T2.1	Clear $I/01_i$ ff, clear RTC Seq ff
T2.3	Clear 3XG29 ff
T4.1	Clear $I/01_f$ ff

*B Network→ZO is timed by CM timing.

SECTION 8 - INPUT/OUTPUT SECTION

8.12. INSTRUCTION EXECUTION OF RIL, EXL, SIL, SXL, WTFI AND SPECIAL AND MONITOR INTERRUPT OPERATIONS

8.12-1. OBJECTIVES

To present the detailed theory of operation involved in the honoring of special and monitor interrupt signals and the execution of certain associated instructions.

8.12-2. INTRODUCTION

Monitor interrupts indicate to the program OD/EF and ID buffer termination and the occurrence of an EI-request. Special interrupts indicate fault function codes, inter-computer error, special RTC conditions, and the occurrence of the external sync signal.

These instructions allow program control over the honoring of interrupts and a wait for an interrupt.

8.12-3. REFERENCES

a. UNIVAC 1219 Technical Manual, Volume I, Paragraph 4-3.

b. UNIVAC 1219 Technical Manual, Volume II, Section 9 (logic diagrams).

8.12-4. INFORMATION

a. RIL, EXL, SIL and SXL Instructions

1. General Description.

a) Instruction Interpretation.

1) RIL, f = 50:30. This instruction clears the All Interrupt Lockout flip-flop to enable the honoring of any special or monitor interrupt. EI-monitor interrupts may still be "locked-out" by the External Interrupt Lockout flip-flop. The k designator is not used. If an EI-monitor interrupt was the last interrupt to be honored and the All Interrupt Lockout flip-flop had not since been cleared, the particular EI Monitor flip-flop is cleared.

2) EXL, f = 50:32. This instruction clears the External Interrupt Lockout flip-flop to enable the honoring of any EI-request signal or EI-monitor interrupt. EI-monitor interrupts may still be "locked-out" by the All Interrupt Lockout flip-flop. The k designator is not used.

3) SIL, f = 50:34. This instruction sets the All Interrupt Lockout flip-flop, thereby preventing the honoring of all special and monitor interrupts. The k designator is not used. When honored, interrupts cause program jumps to special addresses. The SIL instruction could be used to prevent any interruptions should they not be desired.

4) SXL, f = 50:36. This instruction sets the External Interrupt Lockout flip-flop, thereby preventing the honoring of all EI-request signals and EI-monitor interrupts. The k designator is not used. The SXL instruction could be used to inhibit program interruptions from EI-monitor interrupts without disabling the effect of other interrupts.

5) WTFI, f = 50:24. This instruction initiates the wait sequence. The wait sequence continues to reinitiate itself, thus stopping the program. This sequence is only terminated by a special or monitor interrupt as it initiates the interrupt sequence. This instruction causes the program to wait for an interrupt.

b) Execution Sequences.

1) I-Sequence. During the I-sequence which obtains the instruction from memory, the lockout flip-flops are controlled by the RIL, EXL, SIL, and SXL instructions.

2) Wait Sequence. This sequence is initiated by the WTFI instruction and remains in control until the occurrence of an interrupt.

2. Detailed Analysis.

a) RIL, EXL, SIL, SXL Instructions. Refer to logic diagrams, figure 9-28.

Notice that the All Interrupt Lockout flip-flop is set and cleared by f = 50:34 and f = 50:30, respectively. This flip-flop can also be cleared by the f = 54 instruction which performs an indirect program jump. The "lock-out" function is accomplished by preventing the setting of the Int_i Sequence flip-flop via 03G70.

Notice that the External Interrupt Lockout flip-flop is set and cleared by f = 50:36 and f = 50:32, respectively. The "lock-out" function is accomplished by disabling all EI-request and EI-monitor interrupt gates which present these signals to the priority logic. The disable signal is transferred through 3gG71 in the logic diagrams, figure 9-58.

b) WTFI Instruction. Refer to logic diagrams, figure 9-12.

Notice that the $Wait_i$ flip-flop is set by f = 50:24 at T4.2 time. Until the interrupt sequence is initiated, this $Wait_i$ flip-flop is set at every T4.2 time.

b. Special and Monitor Interrupt Operations

1. General Description.

a) Special Interrupt Types.

1) Instruction Fault. This interrupt occurs any time the program attempts to execute an illegal function code (f = 00, 01, 77). In addition to the normal interrupt operations described later in this sheet, the Program Fault indicator is illuminated. This indicator will remain lighted until the MASTER CLEAR switch has been depressed. This switch can be manipulated during the run mode without effecting its register-clearing function.

2) Resume Fault (Inter-Computer Time-Out Fault). This interrupt can only occur if a channel is in the inter-computer mode. If the receiving computer does not accept an outputted word and reply with the ID-acknowledge signal within the allotted time period, this fault condition exists.

3) RTC Monitor. This interrupt occurs if enabled by the previously executed f = 50:14 instruction. During each RTC updating operation, the current clock count is compared with the constant contained in control memory address 00014_8. If they are equal, this interrupt is generated.

4) External Sync. This non-channel interrupt is sent from an external device. This signal could be used for a timing input to synchronize the real time clock or as a control function to cause a program switch to perform some particular operation.

5) RTC Overflow. This interrupt occurs whenever the RTC updating operation causes the clock count to advance from 777777_8 to 000000.

b) Monitor Interrupt Types.

1) External Interrupt Monitor (EI Mon). This interrupt occurs whenever an EI-request occurs. As is the EI-request, the EI Mon is "locked-out" if the External Interrupt Lockout flip-flop is set. This flip-flop is program set and cleared. Each of the 16 channels can generate an EI Mon interrupt.

2) External Function/Output Data Monitor (EF/OD Mon). This interrupt occurs if requested when an EF/OD buffer terminates; i.e., all words have been outputted as specified by the TACW and CACW. The buffer monitor function and interrupt is enabled by the $CACW_{16} = 1$. Each of the 16 channels can generate an EF/OD Mon interrrupt.

3) Input Data Monitor (ID Mon). This interrupt occurs if an ID buffer terminates. The interrupt must be requested by the $CACW_{16} = 1$. Each of the 16 channels can generate an ID Mon interrupt.

c) Honoring Operations. The honoring of one of these interrupts simply involves the execution of a program jump to a special address determined by the interrupt type and, in the case of a monitor interrupt, channel number. Refer to table 8.12-1 for the jump addresses.

The jump is effected by placing in S1 the jump address at the beginning of the next I-sequence. The normal transfer of P $\rightarrow$ S1 is disabled. Also, the normal program counting of P by +1 is prevented. Therefore, the instruction from the jump address is executed rather than the next sequential instruction. When executed due to an interrupt, the instruction at the interrupt address can only use the U portion of its word for an address. That is, it cannot formulate U_P or U_{SR}. XU can be formulated. Usually this instruction performs a jump to an interrupt routine. The interrupt routines, then, would be located in bank 0.

TABLE 8.12-1. SPECIAL AND MONITOR INTERRUPT JUMP ADDRESSES

	INTERRUPT TYPE	JUMP ADDRESS
Special Interrupts (non-channel type)	Instruction Fault	00000 (00500_8 if Auto Recovery)
	Resume Fault	00011_8
	RTC Monitor	00012_8
	External Sync	00016_8
	RTC Overflow	00013_8
Monitor Interrupts (channel type)	EI Monitor	00100_8* + 2x chan
	EF/OD Monitor **	00140_8* + 2x chan
	ID Monitor **	00160_8* + 2x chan

* Add 00200_8 to jump address if channel is $10\text{-}17_8$. Interrupts are listed according to priority.

** Priority of EF/OD and ID-monitor interrupts may be reversed according to Priority Alternator flip-flop.

If the instruction at the jump address does not alter P (not a jump instruction), the computer will return to the normal program sequence at the interrupted point and continue. These interrupt operations are controlled by the interrupt sequence which is initiated so as to run in parallel with the I-sequence.

Except in the case of the instruction fault the interrupt will be ignored (interrupt sequence will not be initiated) if the All Interrupt Lockout flip-flop is set. This flip-flop can be program set and cleared so as to allow the program to determine when it is to be interrupted. This flip-flop is also set during the honoring operations for an interrupt. Thus, after each interrupt is honored, the Lockout flip-flop must be program cleared if another interrupt is to be detected.

Except in the case of EI Mon interrupts during the interrupt/I-sequence which honors the interrupt, the flip-flop which was set to generate the interrupt is cleared. The EI Mon flip-flop has a dual function. Not only does it generate the EI-monitor interrupt signal; but, until it is cleared, it also disables the recognition of any future EI-request on its channel. This prevents another possible status word being brought into memory and destroying the current status word. The program is not aware of the existence of the status word in memory until the EI-monitor interrupt occurs. Until the EI-monitor flip-flop is cleared, the status word will be retained and the program can operate on this word at its leisure. This flip-flop is cleared at the same time that the program clears the All Interrupt Lockout flip-flop.

2. Detailed Analysis.

a) Special Interrupt Translator. Refer to logic diagrams, figure 9-55.

This translator not only selects which special interrupt is to be honored according to a hard-wired priority, but also formulates the jump address to be placed in S1. Notice that if the instruction-fault interrupt is being honored, nothing is placed in the translator register. Thus, the jump address is 00000. Refer to table 8.12-2 for the translator conditions as affected by the special interrupt type.

TABLE 8.12-2. SPECIAL INTERRUPT TRANSLATOR CONDITIONS

SPECIAL INT. TYPE	TRANSLATOR REGISTER ff's				JUMP ADDRESS	OTHER CONDITIONS
	2XG18	2XG17	2XG16	2XG15		
Instr Fault	clear	clear	clear	clear	00000*	
Resume Fault	set	clear	clear	set	00011_8	23G18 = L
RTC Monitor	set	clear	set	clear	00012_8	23G15 = L
Ext. Sync	set	set	set	clear	00016_8	23G16 = L
RTC Overflow	set	clear	set	set	00013_8	23G17 = L
	$S1_{03}$	$S1_{02}$	$S1_{01}$	$S1_{00}$		
	Corresponding S1 Bits					

* Instruction fault jump address is 00500_8 if AUTOMATIC RECOVERY switch is on. A separate source supplies S1 with this address.

A "set" translator flip-flop inserts a 1_2 in the jump address. Interrupts are listed according to priority.

The gates listed under "Other Conditions" in this table enable the correct interrupt generation flip-flop to be cleared when it is honored.

b) Data Flow Block Diagram. Refer to figure 8.12-1 for a block diagram description of the honoring operations for special and monitor interrupts.

The translators are set according to the type of signal and its channel number (monitor interrupts only) during the scan sequence. The instruction for the I-sequence is obtained from the address placed in S1. P is not advanced and therefore retains the address of the instruction which would normally have been executed if the interrupt had not occurred.

Execution of the instruction is normal except that the values U_P and U_{SR} cannot be formulated. If this instruction does not alter P (program jump), the interrupted program will continue after completion of this instruction. Usually the

instruction at the interrupt address performs a return jump to an interrupt routine and retains P, so that a later return to the interrupted point in the program can be effected.

c) Essential Commands. Refer to table 8.12-3 for a sequential list of essential scan and interrupt/I-sequence events. Develop the commands shown by referring to the proper enable pages in the logic diagrams. Only those commands which differ from the normal I-sequence operations are listed.

Refer to logic diagrams, figure 9-54 for the external-sync logic. The external-sync logic is similar to that for a data request. Flip-flop 0XG16 provides a one-shot function for the signal to allow it to set the external sync flip-flop once. This interrupt can be disabled by the DISCONNECT EXTERNAL SYNC switch.

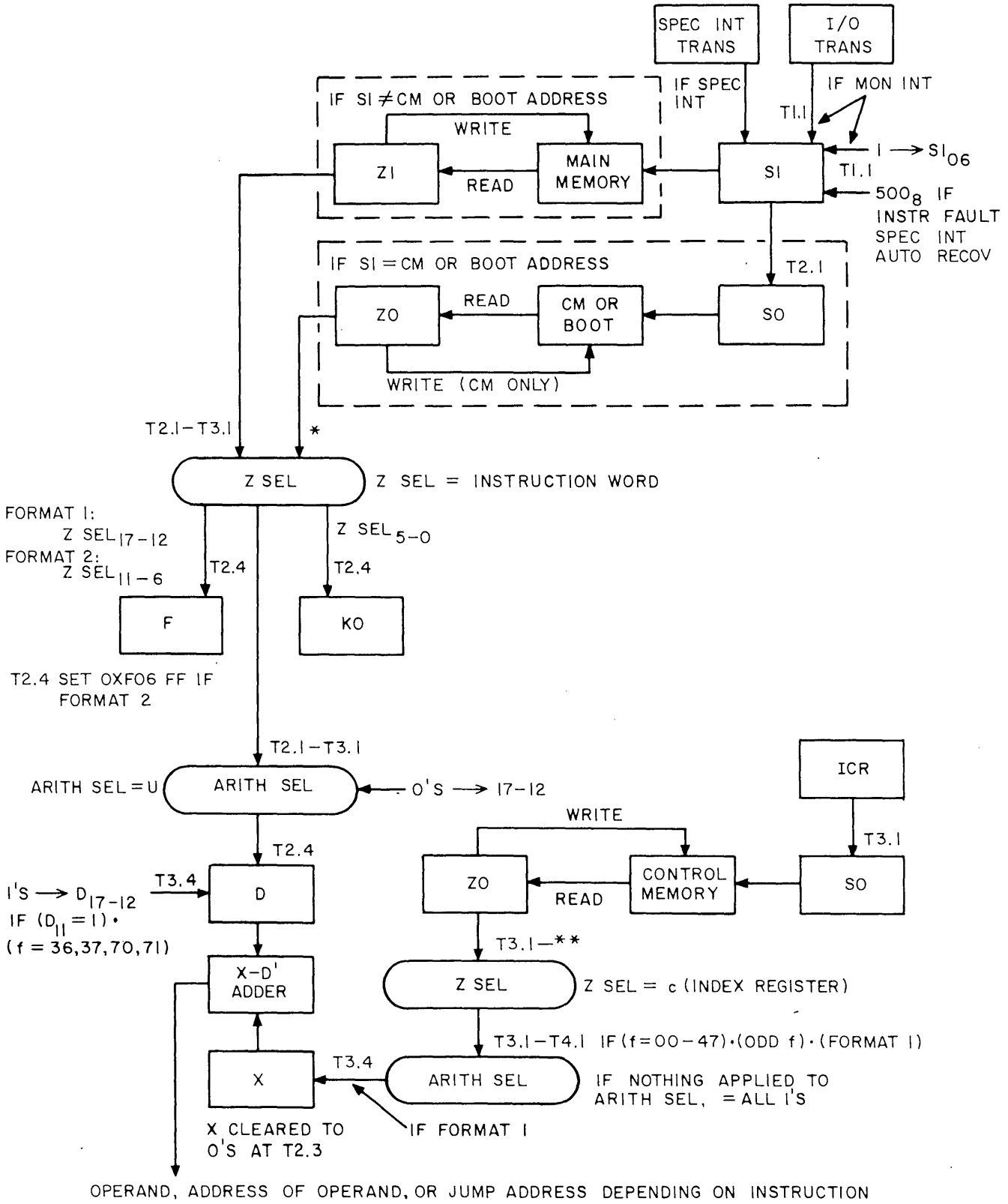
d) EI Monitor Operations. EI monitor operations differ from those for the other interrupts. The clearing of the EI-monitor flip-flop is not performed until the program clears the All Interrupt Lockout flip-flop. Also, the EI-monitor flip-flop disables detection of the EI-request signal on its channel until it is cleared. Both the EI monitor and EI request can be "locked-out" by the External Interrupt Lockout flip-flop.

Refer to figure 8.12-2 for a simplified diagram of the EI-request and monitor logic on one channel. Operations are initiated by the occurrence of the EI-request signal.

8.12-5. SUMMARY

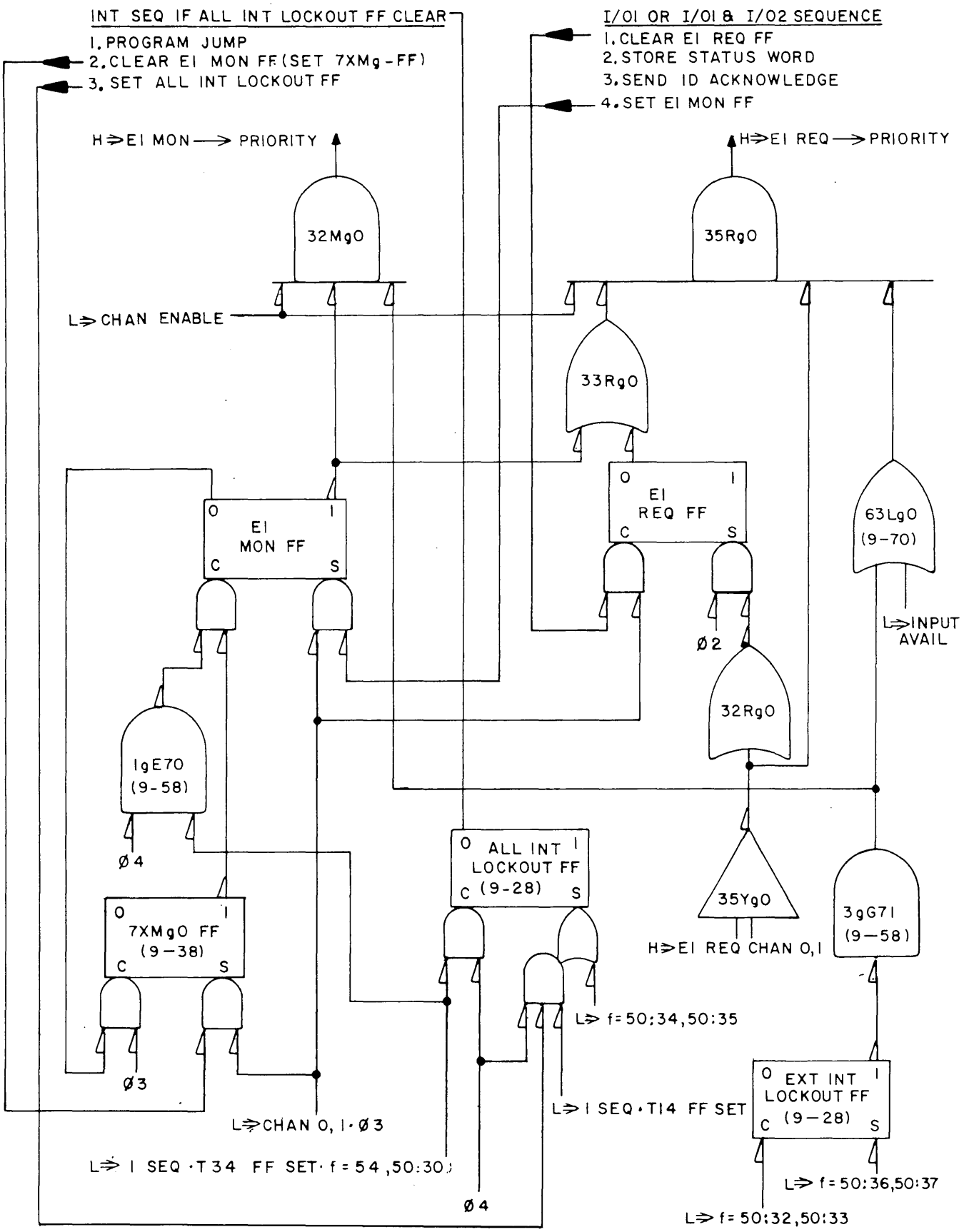
The special and monitor interrupt signals provide indications to the program by interrupting its normal sequence of instructions and causing a program jump to a special address. According to the interrupt type, these interrupts occur from buffer terminations (if requested by the associated $CACW_{16} = 1$, illegal function codes, EI-request, external sync, inter-computer fault, and certain RTC conditions.

Once an interrupt is honored, the All Interrupt Lockout flip-flop must be program cleared to honor another interrupt. This flip-flop, as well as the External Interrupt Lockout flip-flop, can be program set and cleared to allow control of interrupts. Also, the program can cause itself to wait for an interrupt.



NOTES: ONLY THOSE EVENTS COMMON TO MOST INSTRUCTIONS ARE SHOWN.
*ZO → Z SEL OCCURS AS LONG AS SI = CM ADDRESS.
**ZO → Z SEL OCCURS FOR THE DURATION OF THE T24 FF CLEAR.

Figure 8.12-1. Interrupt/I-Sequence Data Flow



NOTE: UNLESS OTHERWISE NOTED, CIRCUITS SHOWN ARE IN LOGIC DIAGRAMS, FIG. 9-61

Figure 8.12-2. EIR and EI Mon Interaction, Simplified Logic Diagram

TABLE 8.12-3. SCAN AND INTERRUPT IN PARALLEL WITH I SEQUENCE ESSENTIAL COMMANDS FOR SPECIAL AND MONITOR INTERRUPTS

TIME NOTATION	COMMANDS	Spec Int	Mon Int
	SCAN SEQUENCE		
T2.1	Clear Spec Int Trans Reg	X	
T2.2	Chan Req→Chan Pri		X
T2.4	Clear Chan Pri, Clear Fct Pri		X
T3.1	Int Req→Chan Pri		X
T3.2	Drop Chan Req→Chan Pri		X
T3.4	Int Req→Fct Pri		X
	Set Instr Fault ff & Prog Fault ff if f=00, 01, 77	X	
	Init Load Mode & disable Run Mode if (Auto Recov on)·(Instr Fault ff set)	X	
T4.1	Clear Fct, Chan, & I/O Trans 1		X
	Clear Spec Int Trans Reg	X	
T4.2	Pri→Fct, Chan, & I/O Trans 1*		X
	Set Int_i ff if (All Int Lockout ff clear + Instr Fault spec int)·(I Seq next)	X	X
	Set I_i ff if I Seq next	X	X
T4.3	Spec Int Req→Spec Int Trans Reg if $\overline{\text{Instr Fault spec int}}$	X	
T4.4	Clear S1	X	X
The following occurs if Int_i ff set; otherwise, normal I-Seq follows			
	INTERRUPT AND I SEQUENCE IN PARALLEL		
T1.1	I/O Trans→S1*, 1→$S1_{06}$*, Init Memory		X
	Spec Int Trans→S1, Init Memory	X	
	00500_8→S1 if Load Mode (Instr Fault · Auto Recov)	X	
	Disable P→S1 & Adv P Subsequence	X	X
T1.2	S1→P if Load Mode (Instr Fault · Auto Recov)	X	
T1.3	Clear Instr Fault ff, clear Spec Int ff just honored	X	
	Clear EF/OD or ID Mon ff just honored		X
	Set 7XMg-ff if EI Mon** just honored		X
T1.4	Set All Int Lockout ff	X	X
T2.1	S1→SO & Init CM/Boot if S1 = CM or Boot address	X	X
	Set Bootstrap ff if S1 = $00500\text{-}00537_8$		X
T2.2	Set Int_f ff, set I_f ff	X	X
T3.4	Disable SR→$D_{15\text{-}12}$ & $P_{15\text{-}12}$ $D_{15\text{-}12}$	X	X
T4.1	Clear Int_i ff, clear I_i ff	X	X
T2.1	Clear Int_f ff, clear I_i ff	X	X

*These commands occur if there is no special interrupt; thus, special interrupts have higher priority than monitor interrupts.

**If honored, EI Mon ff not cleared until program clears All Interrupt Lockout ff. EI Mon ff to be cleared is selected by 7XMg-ff's. EI Mon is not detected unless the EI Lockout ff is clear. Except advance-P subsequence, P→S1, SR→$D_{15\text{-}12}$, and $P_{15\text{-}12}$→$D_{15\text{-}12}$, all normal I-Seq events occur (not shown).

NAME:_____

8.12-6. STUDY QUESTIONS

a. Refer to logic diagrams, figure 9-54.

03N41 is the gate which clears the special interrupt flip-flop after it is honored. Notice that with the exception of the Instruction Fault flip-flop, the clearing of the special interrupt flip-flop is enabled by 03N41 and the 23G1- inputs from the translator which select the flip-flop which has just been honored. The Instruction Fault flip-flop is cleared unconditionally. Explain why there is no input to this flip-flop from the translator to select it to be cleared.

b. The program is frequently being jumped to address 00011_8. Some of these jumps should not normally occur. RTC is running. Determine whether each of the following malfunctions, occurring individually, could cause the erroneous jumps. Indicate your answers by writing "yes" or "no" beside each malfunction condition.

_____ 1. 1XG18 ff, pin 11 grounded (logic diagrams, figure 9-54)

_____ 2. 1XG18 ff, pin 12 grounded

_____ 3. 8XL30 ff, pin 13 grounded (logic diagrams, figure 9-34)

_____ 4. 8XL30 ff, pin 15 grounded

_____ 5. 5XL10 ff, pin 11 grounded (logic diagrams, figure 9-69)

_____ 6. 5XL10 ff, pin 11 constant low level output

c. The program currently running uses RTC-monitor and RTC-overflow interrupts. It is not prepared to handle any other interrupt types. At any time, the occurrence or absence of these two interrupts should not fault the computer. It is found that the program periodically faults (instruction fault). Determine whether each of the following malfunctions, occurring individually, could cause the faults. Indicate your answers by writing "yes" or "no" beside each malfunction condition.

_____ 1. 13G17 grounded output (logic diagrams, figure 9-55)

_____ 2. 13G17 constant low level output

_____ 3. 13G15 grounded output (logic diagrams, figure 9-55)

c. (Continued)

_____ 4. 13G15 constant low level output

_____ 5. 11E15 grounded output (logic diagrams, figure 9-54)

_____ 6. 00N42 grounded output (logic diagrams, figure 9-53)

_____ 7. 78N12 constant low level output (logic diagrams, figure 9-22)

SECTION 8 - INPUT/OUTPUT SECTION

8.14. B NETWORK ADDER

8.14-1. OBJECTIVES

To present the detailed theory of operation involved in the B-network.

8.14-2. INTRODUCTION

The B-network modifies by $\pm$ 1 the content of B. It is used to update the $CACW_{15-0}$ during ID, OD, and EF operations and to provide the counting action for the real time clock. The BSK and BJP instructions also use this adder.

8.14-3. REFERENCES

UNIVAC 1219 Technical Manual, Volume II, Section 9 (logic diagrams).

8.14-4. INFORMATION

a. General Description. The B-network is an 18-bit open-ended adder. It is hard-wired to the output of the B register to provide B $\pm$ 1. The modification performed is controlled by the B $\pm$ 1 flip-flop. If this flip-flop is cleared, it causes the network to add B + 1. With the flip-flop set, the subtraction of B - 1 is performed.

One use of the network is to increment the clock count for RTC operations. Also, during ID, OD, and EF operations, it modifies the $CACW_{15-0}$ by $\pm$ 1 depending on the direction of the buffer. During the updating of the CACW, stages 17 and 16 of the network are disabled. That is, the $CACW_{17,16}$ will pass through the network without modification.

The BSK (f = 56) and BJP (f = 73) instructions also use the B-network to provide the index register counting action.

The B-network output can only be taken to ZO.

b. Detailed Analysis.

1. Effect of Borrow Request and Carry. If a carry or a request for a borrow is applied to a bit position of B, the result of this subtraction produces the complement of the bit. Refer to figures 8.14-1 and 8.14-2 for examples.

1	0	B BIT
+1	+1	CARRY
0	1	RESULT (1'S COMPLEMENT OF B)

Figure 8.14-1. B Network Carry Examples (B + 1)

1	0	B BIT
−1	−1	BORROW
0	1	RESULT (1'S COMPLEMENT OF B)

Figure 8.14-2. B Network Borrow Examples (B - 1)

2. B Network Stage 00. Refer to logic diagrams, figure 9-118.

The addition or subtraction of B ± 1 always affects B_{00} to produce is complement. This complemented value is placed in ZO_{00} from the "0" side output of the B_{00} flip-flop. A low level to ZO_{00} represents 1_2. A low level from 00B00 represents $B_{00} = 0_2$. Thus, ZO_{00} receives B_{00} during the B Network ZO transfer.

3. Generation of Carries and Borrow Requests. If a carry or borrow request is applied to a particular bit position, the complement of that bit is the result. If that same B bit position cannot absorb the carry or supply the borrow, the carry/borrow request is propagated to the next highter bit. If a B bit can absorb a carry, it contains 0_2. That is, the carry can be added to the bit without producing another carry. If a B bit can supply a borrow, it contains 1_2. A bit of B is said to contain an "enable" if it can satisfy a carry/borrow request and prevent it from propagating further.

All carries/borrow requests originate from the addition/subtraction of $B_{00} + 1_2$. A carry is applied to a particular bit position if all of the less significant B bit positions contain 1's. A borrow request is applied to a particular bit position if all of the less significant B bit positions contain 0's. Refer to table 8.14-1 for the conditions necessary to apply carries/borrow requests to the B bits.

TABLE 8.14-1. B NETWORK CARRY/BORROW REQUEST CONDITIONS

CARRY/BORROW REQUEST	CONDITIONS
Carry/brw → 01	No enable in 00
Carry/brw → 02	(No enable in 01)·(carry/brw → 01)
Carry/brw → 03	(No enable in 02)·(carry/brw → 02)
Carry/brw → 04	(No enable in 03)·(carry/brw → 03)
Carry/brw → 05	(No enable in 04)·(carry/brw → 04)
Carry/brw → 06	(No enable in 05)·(carry/brw → 05)
etc.	etc.

NOTES: "Carry/brw → XX" means a carry or borrow request is being applied to bit position XX.

"No enable in XX" means $B_{XX} = 1_2$ if B + 1

0_2 if B - 1

SECTION 8 - INPUT/OUTPUT SECTION

8.13. INTER-COMPUTER OPERATIONS

8.13-1. OBJECTIVES

To present the detailed theory of operation and programing considerations involved in inter-computer operations.

8.13-2. INTRODUCTION

The input/output data and control transfer technique is somewhat altered for inter-computer operations. The Resume Fault interrupt, output available condition, special programing, and the ODR/EFR logic configuration should be considered.

8.13-3. REFERENCES

UNIVAC 1219 Technical Manual, Volume I, Paragraph 4-3b(1)(a)2.

8.13-4. INFORMATION

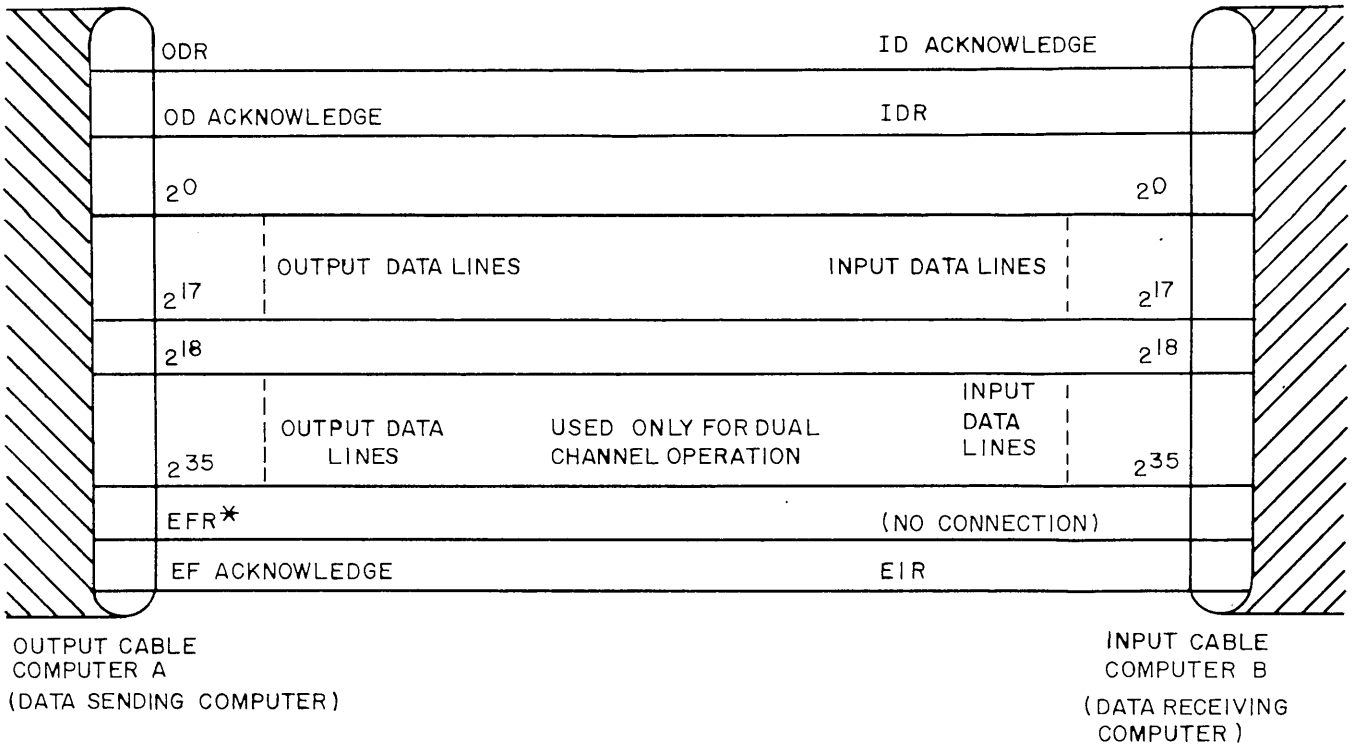
a. General Description.

1. Inter-Computer Channels. Inter-computer communications can be set up on any of the 16 channels. Except for ODR operations, the I/O functions are normal. To set up the special inter-computer ODR logic configuration, all that is necessary is to place the CHANNEL INTER-COMPUTER switch for the particular channel in the up position.

The I/O cable connections must be reversed from normal. The input cable from one computer must be connected to the output cable terminals of the other computer and vice-versa. This cross-cabling results in the output data lines from one computer appearing as input data lines to the other and some of the acknowledge signals being considered as request signals by the receiving computer. Refer to figure 8.13-1 for the inter-computer cable description.

2. External Function/External Interrupt Exchange. Refer to figure 8.13-1. An EF word and EF-acknowledge from computer A appear to computer B as a status word and EIR. There is no EFR signal. The EFR line of computer A does not have a logic connection at computer B. Also, the special inter-computer ODR logic of computer A cannot recognize an EFR.

Using the external function/external interrupt exchange, one computer could execute control over the program of the other by means of the EF/status word bit configuration. This coded word could cause the receiving computer to set up an ID buffer to accept data and specify how to process this data.



NOTE: *WITH INTER-COMPUTER LOGIC IN CONTROL, EFR IS NOT USED.

Figure 8.13-1. Inter-Computer I/O Cabling

3. Data Exchange. Refer to figure 8.13-1. During the inter-computer data exchange operations, an output data word and OD-acknowledge from computer A appear to computer B as an input data word and IDR. As computer B accepts each word, it responds with the ID-acknowledge which appears to computer A as the ODR. This ODR actually indicates that computer B has accepted the last data word and is a request for the next data word.

The transfer process is initiated in the special inter-computer ODR logic of computer A. The setting-up of the OD buffer by the OUT instruction (f = 50:12) sets the EF/OD Active flip-flop for the particular channel. This active flip-flop simulates the first ODR. Thus, the buffer instruction actually forces out the first data word for inter-computer operation. Each word after the first is not transferred until computer A receives the ID-acknowledge (ODR) from the acceptance of the previous word.

If computer B does not respond with the ID-acknowledge within a specified length of time after a data word has been sent to it, computer A will notify its own program by generating the resume-fault interrupt.

b. Detailed Analysis.

1. ODR Logic. Refer to study guide sheet number 8.6 for the inter-computer ODR logic description.

The setting of the EF/OD Active flip-flop at the buffer initiation simulates the first ODR to start the data transfer process. During the sending of the associated OD-acknowledge, the Resume flip-flop is cleared. It is not set until the

ID-acknowledge (ODR) from the data-receiving computer is received. The setting of the Resume flip-flop again satisfies the ODR gate so as to present this signal to priority to be honored for the next word. This process continues until the OD buffer terminates.

As each word is outputted, the corresponding OD-acknowledge is not dropped until the ID-acknowledge is received to set the Resume flip-flop. Therefore, while a word is being sent to the data-receiving computer waiting for the acknowledge, output is not available for the particular chassis.

Realize that when the last word of the OD buffer is outputted, the EF/OD Active flip-flop is not immediately cleared as for normal channel operation. It remains set until the corresponding last ID-acknowledge (ODR) is received. It is necessary that the Active flip-flop remain set so as to allow the last ODR to be recognized and set the Resume flip-flop. The setting of this flip-flop provides the enable to clear the EF/OD Active flip-flop.

2. Resume Fault Interrupt. Refer to study guide sheet number 8.12 for the resume fault description.

The Resume Fault flip-flop is set if the Resume flip-flop on any of the four I/O chassis is in the clear state for longer than a specified length of time. For normal channel operation, this flip-flop is set for approximately the duration of the EF or OD-acknowledge. Therefore, inter-computer operation is the only case whereby the flip-flop could remain clear for a relatively long period of time. The resume-fault condition exists if the data-receiving computer does not clear the Resume flip-flop in time with its ID-acknowledge (ODR).

Upon detection of a resume-fault interrupt, the program should set the Resume flip-flop so as to provide the output available condition for the particular chassis. The absence of this condition prevents any OD or EF activity for any channel on the chassis. The Resume flip-flop is set by the SRSM instruction (f = 50:20).

The interrupt is timed by the real time clock. Thus, it is necessary that the clock not be disabled by the RTC DISCONNECT switch. Otherwise, the program would not be notified of the fault condition and output would remain not available for the chassis.

8.13. SUMMARY

The most obvious difference of the inter-computer communication technique as compared with normal channel operation is the cross-cabling. Acknowledge signals sent by one computer are considered as requests by the other. With the exception of the ODR/EFR logic, the logic operations are the same as for the normal channel mode.

SECTION 8 - INPUT/OUTPUT SECTION

8.14. B NETWORK ADDER

8.14-1. OBJECTIVES

To present the detailed theory of operation involved in the B-network.

8.14-2. INTRODUCTION

The B-network modifies by $\pm$ 1 the content of B. It is used to update the $CACW_{15-0}$ during ID, OD, and EF operations and to provide the counting action for the real time clock. The BSK and BJP instructions also use this adder.

8.14-3. REFERENCES

UNIVAC 1219 Technical Manual, Volume II, Section 9 (logic diagrams).

8.14-4. INFORMATION

a. General Description. The B-network is an 18-bit open-ended adder. It is hard-wired to the output of the B register to provide B $\pm$ 1. The modification performed is controlled by the B $\pm$ 1 flip-flop. If this flip-flop is cleared, it causes the network to add B + 1. With the flip-flop set, the subtraction of B - 1 is performed.

One use of the network is to increment the clock count for RTC operations. Also, during ID, OD, and EF operations, it modifies the $CACW_{15-0}$ by $\pm$ 1 depending on the direction of the buffer. During the updating of the CACW, stages 17 and 16 of the network are disabled. That is, the $CACW_{17,16}$ will pass through the network without modification.

The BSK (f = 56) and BJP (f = 73) instructions also use the B-network to provide the index register counting action.

The B-network output can only be taken to ZO.

b. Detailed Analysis.

1. Effect of Borrow Request and Carry. If a carry or a request for a borrow is applied to a bit position of B, the result of this subtraction produces the complement of the bit. Refer to figures 8.14-1 and 8.14-2 for examples.

1	0	B BIT
+1	+1	CARRY
0	1	RESULT (1'S COMPLEMENT OF B)

Figure 8.14-1. B Network Carry Examples (B + 1)

1	0	B BIT
−1	−1	BORROW
0	1	RESULT (1'S COMPLEMENT OF B)

Figure 8.14-2. B Network Borrow Examples (B - 1)

2. B Network Stage 00. Refer to logic diagrams, figure 9-118.

The addition or subtraction of B ± 1 always affects B_{00} to produce is complement. This complemented value is placed in ZO_{00} from the "0" side output of the B_{00} flip-flop. A low level to ZO_{00} represents 1_2. A low level from 00B00 represents $B_{00} = 0_2$. Thus, ZO_{00} receives B_{00} during the B Network ZO transfer.

3. Generation of Carries and Borrow Requests. If a carry or borrow request is applied to a particular bit position, the complement of that bit is the result. If that same B bit position cannot absorb the carry or supply the borrow, the carry/borrow request is propagated to the next highter bit. If a B bit can absorb a carry, it contains 0_2. That is, the carry can be added to the bit without producing another carry. If a B bit can supply a borrow, it contains 1_2. A bit of B is said to contain an "enable" if it can satisfy a carry/borrow request and prevent it from propagating further.

All carries/borrow requests originate from the addition/subtraction of $B_{00} + 1_2$. A carry is applied to a particular bit position if all of the less significant B bit positions contain 1's. A borrow request is applied to a particular bit position if all of the less significant B bit positions contain 0's. Refer to table 8.14-1 for the conditions necessary to apply carries/borrow requests to the B bits.

TABLE 8.14-1. B NETWORK CARRY/BORROW REQUEST CONDITIONS

CARRY/BORROW REQUEST	CONDITIONS
Carry/brw → 01	No enable in 00
Carry/brw → 02	(No enable in 01)·(carry/brw → 01)
Carry/brw → 03	(No enable in 02)·(carry/brw → 02)
Carry/brw → 04	(No enable in 03)·(carry/brw → 03)
Carry/brw → 05	(No enable in 04)·(carry/brw → 04)
Carry/brw → 06	(No enable in 05)·(carry/brw → 05)
etc.	etc.

NOTES: "Carry/brw → XX" means a carry or borrow request is being applied to bit position XX.

"No enable in XX" means $B_{XX} = 1_2$ if B + 1

0_2 if B - 1

The B-network logic for bits 17 through 01 have gates which test all of the less significant B bits for the configurations previously described. If a carry/borrow request is applied, the complement of the B bit is the network output as was shown for bit 00. Refer to figure 8.14-3 for the B-network carry/borrow request logic. This is a portion of that shown in the logic diagrams, figures 9-118 through 9-120. The logic for the other bit positions follows the same pattern.

4. Timing Signal for Resume Fault Indication. Refer to logic diagrams, figure 9-119.

The output of 11B10 to 11E18 is used to time the resume-fault special interrupt. The high level from 11B10 applies a carry/borrow request to bit position 10 (1-second RTC bit). The fault indication is used during RTC operations. Two consecutive carry indications from 11B10 mean that a 1-second time period has elapsed. This is the period during which the inter-computer resume condition must be established; otherwise, the resume-fault interrupt is generated.

5. B-Network$_{17,16}$ Disable During ID, OD, EF Operations. During ID, OD, or EF operations, the B-network is used to modify the CACW$_{15-0}$ by $\pm$ 1. To prevent modification of the CACW$_{17,16}$ which are the buffer direction and monitor interrupt bits, respectively, carries and borrow requests to these bits are disabled.

Refer to logic diagrams, figure 9-120.

The high level from 11B16 applies a carry/borrow request to bit position 16. This gate is disabled by its input from 06G25 during the I/O-sequences if an RTC update operation is not occurring. Disabling of this gate not only prevents a carry/borrow request from being applied to bit 16, but also prevents modification of bit 17. Thus, the B-network simply transfers these bits from B to Z0.

8.14-5. SUMMARY

The input to the B-network is hard-wired to the output of the B register. This adder constantly outputs the value of B $\pm$ 1 as controlled by the B $\pm$ 1 flip-flop. The network is used during several input/output operations as well as the executions of BSK and BJP instructions.

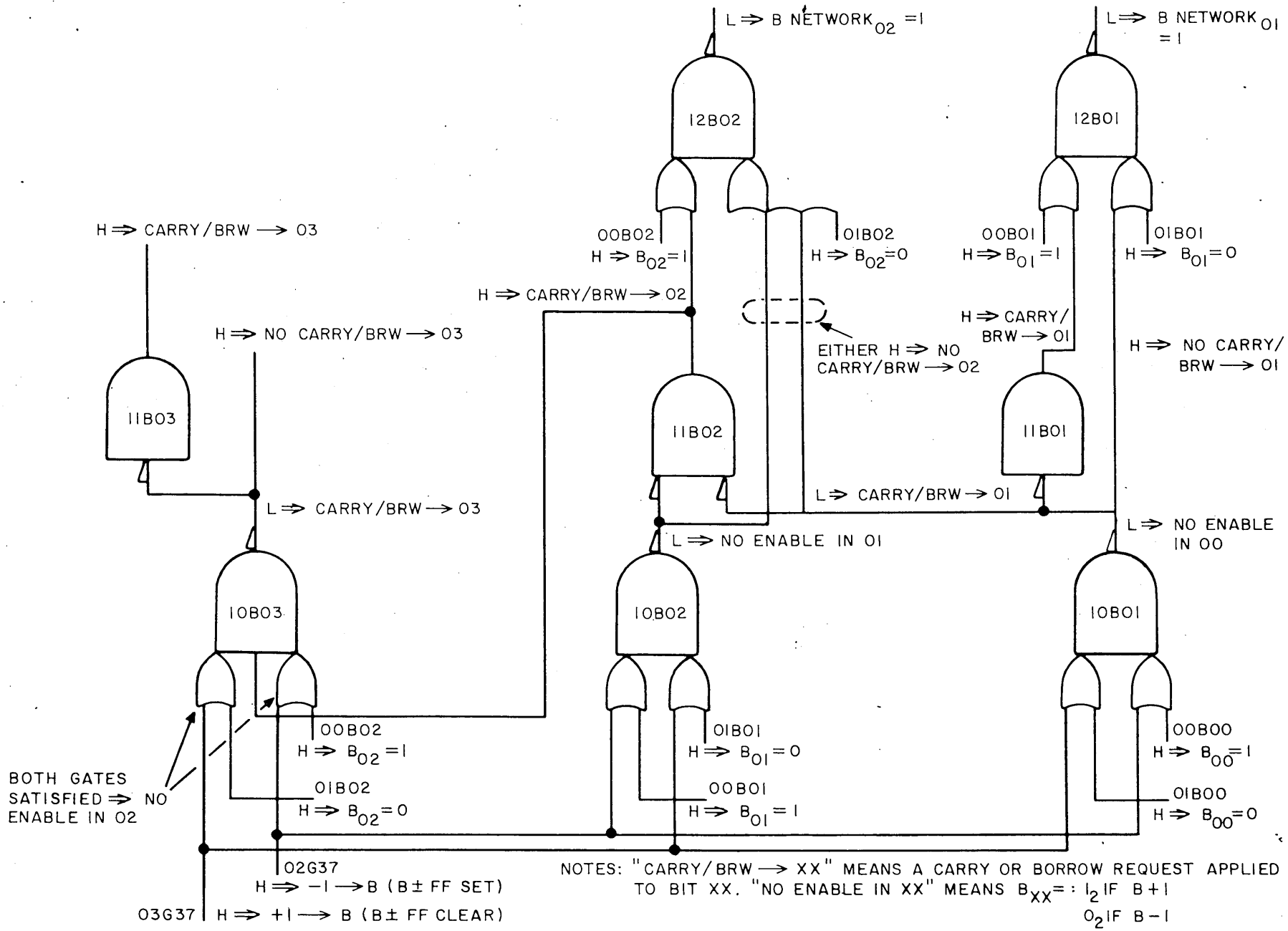


Figure 8.14-3. B-Network Simplified Logic Portion