Systems Engineering Laboratories

SEL 810A
Interface Design Manual

Systems Engineering Laboratories

Reference Manual

SEL 810A Interface Design

February 15, 1967

SEL 95003A
Price: $15.00

6901 West Sunrise Blvd., Fort Lauderdale, Florida 33310
Area Code 305 587-2900

PREFACE

The purpose of this manual is to provide information concerning the Unit Input/Output Configuration designed by Systems Engineering Laboratories, Incorporated. The information is intended to be an aid to those engineers designing or using peripheral units which communicate with the SEL 810A general purpose computer.

CONTENTS

Paragraph Page

Paragraph Page

ILLUSTRATIONS

Figure Page

TABLES

1.0 Computer Organization

1.1 General Description

The SEL 810A computer is a fast, parallel operation, 16-bit binary general purpose machine. The major characteristics are as follows:

All silicon monolithic integrated logic circuits
4,096 word memory, expandable to 32,768 words
1.75 microsecond full cycle time
Computation time including access and indexing
  Add, Subtract 3.5 microseconds
  Multiply 7 microseconds
Double-length accumulator (32 bits)
Hardware index register (lower accumulator)
I/O structure capable of handling 64 units or controllers
  (Drivers and terminators for 16 units supplied
  with basic computer)
Two separate levels of priority interrupts (Basic Computer)
Four sense switches
Switch-addressable program halt
ASR-33 typewriter with paper tape reader and punch
Size - 24" wide, 62" high, 26" deep
  (41" deep including desk top)
Temperature environment - 50° to 95° F

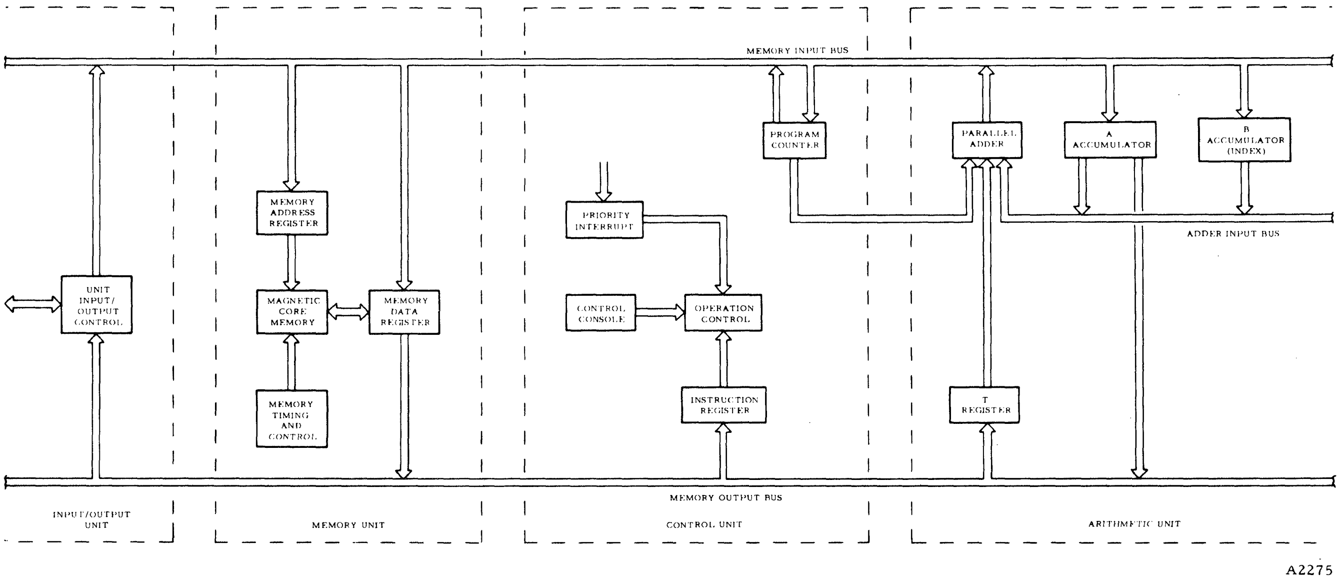


Figure 1.1-1 SEL 810A Computer Block Diagram

The 810A computer is formed by four major units - memory, control, arithmetic and input/output. The memory stores the instruction words which define the operation of the computer and the data words on which the computer operates. The control unit calls up the instruction words, decodes them and issues commands to operate the computer. The arithmetic unit performs computation with data words supplied by the input/output unit and the memory unit under the direction of the control unit. The input/output unit transmits data words, commands and status reports between the computer and peripheral equipment. The computer operates on, and from, 16-bit binary words which are transferred in parallel between the computer units. Arithmetic operations are performed using two's complement binary arithmetic with negative words stored in the two's complement form.

1.2 Memory Unit

The memory unit is composed of one, two, three or four separate modules. Each module has either 4,096 (4K module) or 8,192 (8K module) addressable storage locations. Each location consists of one 16-bit data or instruction word. The total number of storage locations may range from 4,096 provided by the basic 4K module to 32,768 available with four 8K modules. The 16-bit capacity of each location may be expanded to 18 bits with optional memory parity bit and/or program protect.

Individual modules are composed of these four elements:

(1) 4K or 8K MAGNETIC CORE MEMORY (16, or 18 bits)

(2) 12 bit (4K) or 13 bit (8K) MEMORY ADDRESS REGISTER

(3) 16 (or 18) bit DATA REGISTER

(4) self-contained TIMING AND CONTROL

Instruction words and data words are loaded into specific addresses prior to the program execution. Loading may be performed manually through the panel controls or automatically from peripheral units through the use of the supplied loader program. Each input word is transferred to the memory data register and the accompanying storage address is transferred to the memory address register. When both registers have been loaded, a "write" command is issued by the program control unit and the bits in the memory data register are written into the magnetic cores addressed by the memory address register.

When the entire group of instruction words forming a program is loaded and execution is started, addresses selected by the control unit are sent to the memory address register and a "read" command is issued. The status of each core at that address is sensed and transferred to the memory output register. The sensing of the cores sets them all to the same state, so the word now in the memory data register is immediately rewritten into its original memory location so as to be available for later use. The word is also transferred to the control unit to be decoded or to the arithmetic unit for computation. The memory read and write cycles are completely automatic so that only the memory address and source or destination must be supplied by the program.

1.3 Control Unit

The control unit contains a 15-bit PROGRAM REGISTER capable of directly addressing 32,768 memory locations. This register supplies the addresses of the instruction words from which the computer operates. The register is initially set to the address of the first instruction of a program when the computer is started. It is then automatically increased by one by each instruction until a Halt, Branch or Conditional Skip instruction is read from memory. The Halt instruction stops the computer while the Branch instructions change the contents of the program register to the operand address contained in the instruction. The Skip instructions cause the program register to be increased by either one or two counts, depending on the value of the Skip condition specified by the instructions.

The instruction words are read from memory into the INSTRUCTION REGISTER and automatically restored in memory. The binary digits forming the instruction word are then applied to the OPERATION CONTROL circuits. The unique codes assigned to each instruction are then decoded and used to provide timing and gating signals to the remainder of the machine. The signals from switches on the CONTROL CONSOLE are also connected to the OPERATION CONTROL circuits. External PRIORITY INTERRUPTS will cause the control circuits to switch the program register to programs designed to process the external demand.

The memory cycle during which instruction words are read and decoded is referred to as the "Instruction Cycle". Some instructions, called memory reference instructions, contain a memory address which specifies the location of an "operand" which is to be operated on by the computer. The word format for a memory reference instruction is shown in Figure 1.3-1. For these instructions, one or more additional memory cycles, called "Execution Cycles", are required. During the instruction cycle, the memory address is supplied in part by the "operand address" contained in the instruction word and by the program register. The operand is read from memory and operated upon according to signals provided by the operation code. Most memory reference instructions are accessed and executed in a total of two cycles. However, instructions such as multiply and divide require more than one execution cycle.

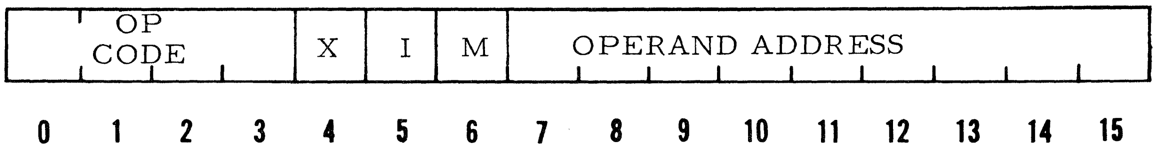


Figure 1.3-1 Memory Reference Instruction Word Format

Many instruction words require no operand from memory and are executed completely within the instruction cycle. Others, while requiring no operand from memory, do require one or more execution cycles for completion. Chief among this latter group are the shift instructions. For these instructions, a group of bits within the instruction word (see Figure 1.3-2) defines the number of shifts to be performed while the operation code of the word defines the type of shifting to be done. Other instructions, notably the input/output control instructions, are composed of two instruction words; one defining the type of operation and the unit and the other defining the actual operand or the operand memory location. The words forming these input/output instructions are automatically unloaded from memory in the proper sequence.

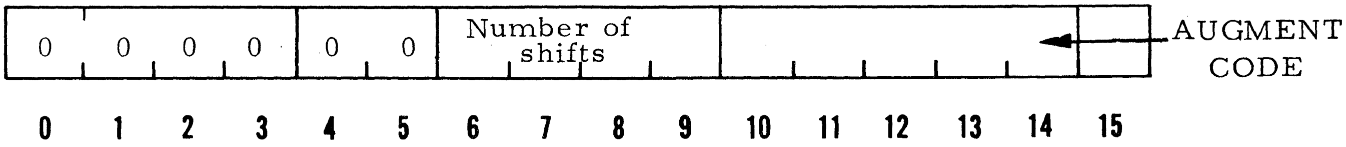


Figure 1.3-2 Augmented 00_8 Instruction Word

1.4 Arithmetic Unit

The arithmetic unit consists of a 16-bit adder and several accessory storage registers. Two of these registers, the A ACCUMULATOR and the B ACCUMULATOR, may be loaded and unloaded by program control. The A ACCUMULATOR is the primary arithmetic register and derives its name from its function of accumulating results of the arithmetic operations. Because only one word may be taken from the memory and input/output units by each instruction, the second operand in add and subtract operations must be loaded in a register prior to the add and subtract instructions. The A ACCUMULATOR fulfills this function and also provides temporary storage for the result of the arithmetic operation. The B ACCUMULATOR holds the multiplier during multiply operations and stores the least significant bits of the product. In addition to these strictly arithmetic functions, the two accumulators provide a convenient storage area for rearranging data words through shifting and logical operations.

A third register connected to the adder is the T REGISTER which holds the operand unloaded from the memory. This 16-bit register plus the 16-bit A and B ACCUMULATORS supply inputs to the 16-bit binary ADDER. When an add instruction is performed, the data words are simply added according to the rules of binary arithmetic.

The basic data format of the SEL 810A computer is a 16-bit binary single-precision fixed point word (shown in Figure 1.4-1). This format contains the sign bit in bit position 0, with bit position 1 holding the most significant data bit and bit position 15 holding the least significant bit. This format is defined as an integer with an imaginary binary point located to the right of bit position 15. The 810A set of library integer subroutines assumes this representation. The programmer may, of course, scale single-precision words in any desired manner and utilize the extensive shift and test instruction repetoire to maintain the binary point location.

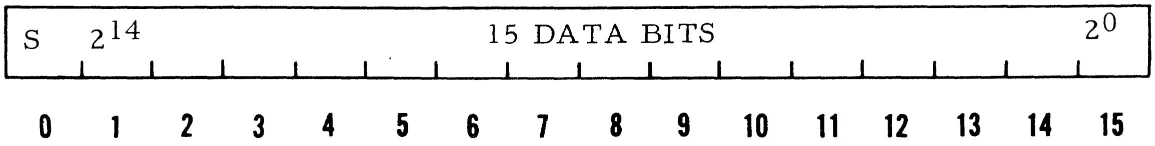


Figure 1.4-1 Single-Precision Fixed Point Data

1.5 Input/Output

The basic SEL 810A is supplied with a "party line" input/output unit called the Unit Input/Output Control (UI/OC). The UI/OC is capable of communicating with up to 64 peripheral units or unit controllers. Unit controlllers, such as magnetic tape control units, may communicate with several units each. Therefore, the number of individual units which can communicate with the computer is virtually unlimited.

Data transfer instructions are provided which enable word transfers directly between computer memory and peripheral units as well as between the A accumulator and peripheral units. In addition, external unit command and test instructions are provided.

The I/O instruction set is particularily powerful because each instruction causes several functions to be performed. First, execution of each I/O instruction causes a unit to be connected to the computer. The unit number is contained in each I/O instruction. Second, an automatic test is made of the unit which determines if the unit can execute the instruction. Third, the data or command transfer is made if the unit is "ready". Fourth, the unit is disconnected. If the unit is not ready when tested, the computer will either wait until the unit is ready and then transfer or it will disconnect the unit and advance the program register to a "reject" location. A "Wait Flag" is provided in each I/O instruction (except the test instruction) to enable the programmer to specify the "Wait" or "Skip" mode of execution. The normal time required to perform the complete connect, test, transfer, and disconnect operation is only three machine cycles (5.25 microseconds).

In addition to the basic I/O structure, which meets the requirements of many systems, up to eight fully buffered channels can be added to the computer. These channels permit transfer of blocks of data up to 32,768 words in length between the computer and peripheral units. Block transfer is made under hardware control at rates up to 572,000 words per second. A single cycle is stolen per word transfer. An automatic reinitialization feature is provided which enables chaining of block transfers.

A priority interrupt system is provided which enables the computer to have up to 96 individual levels of priority interrupts. Interrupts can be selectively enabled and disabled under program control. A unique memory location is assigned to each level.

An ASR-33 typewriter, paper tape punch and reader are supplied with the basic computer. The reader (input unit) operates at 20 characters per second and the punch and printer (output unit) operates at 10 characters per second. The typewriter can be operated either on-line or off-line. When operating on-line, the input and output units operate independently, which enables, for example, a paper tape to be read and a separate set of characters to be printed at the same time.

1.6 Computer Data Flow

Most data transfers within the SEL 810A Computer move all 16 bits of a computer word at once. This type of transfer is termed "parallel" as opposed to "serial" transfers where 16 separate bit transfers would be required to transfer a computer word. The exceptions to this general rule are the lateral shifts within and between the accumulators, transfers of flag bits between registers and transfers of 6 or 8 bit characters to and from peripheral devices.

The parallel transfers are accomplished by timing signals derived from the 10 MHz computer clock and directed by the decoding circuits of the Instruction Register (See paragraph 1.7 - Computer Timing). The major data paths are shown in Figure 1.6-1.

Each major data path (bus) consists of 16 (18 if parity and program protect is included) parallel signal lines between the source and destination registers or units. Multiple-input OR gates are used on those busses which transfer data from more than one source. AND gates permit specific registers or units to be selected as the source and destination of the data. The memory modules contain input and output AND circuits, address registers and data registers not shown in Figure 1.6-1. Also not shown are input AND circuits on the Instruction Register and the T Register, as these are intrinsic circuits on the flip-flops in the registers.

An instruction word read from a memory module is brought into that module's data register, gated through the 17-input OR circuit, and a 17-input AND gate to the Memory Output Bus (MOB). From the MOB it is then applied through intrinsic AND gates into the Instruction and T Registers. If the instruction is a memory reference instruction, the address portion of the word is applied to the 9 low order adder circuits. If a MAP bit is contained in the instruction, the 6 high-order bits of the program register are gated through 6 AND circuits to the bit 1 - 6 positions of the Adder (bit 0 is the sign bit; not used in an address). If the index flags in the instruction, the 16 bit (including sign) of the B Accumulator (hardware index register) is gated onto the Adder Input Bus to be summed with the address. If either or both the MAP and Index Flags are absent, the gates from the source register (s) remain closed, the output address of the adder is applied through 12

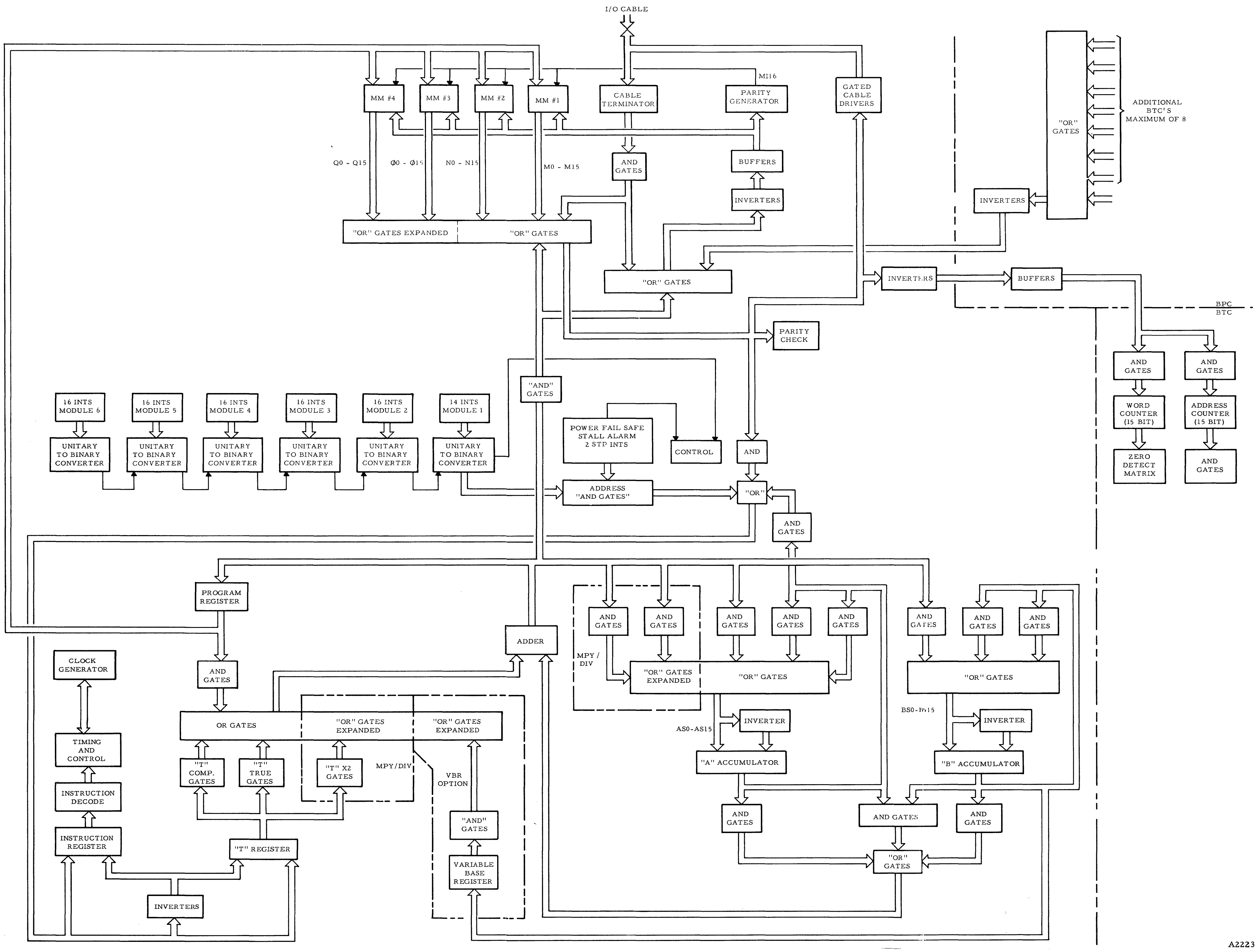


Figure 1.6-1 Detailed Computer Block Diagram

AND gates and input OR circuits to the memory addressing control (when the module is selected from the two high-order address bits) and to the memory modules where it is gated into the selected module's address register.

A data word read from memory is applied through to the Memory Output Bus and loaded in the T Register. The data word may then be applied through the adder to the input AND gates of the A or B accumulators. If an arithmetic operation is to be performed, the outputs of either the A or B Accumulators may be AND gated onto the Adder Input Bus to be summed with the memory word. The result (or the unchanged data word, depending on the instruction) may be AND gated into either the A or B Accumulator.

When a data word is to be transferred from an accumulator register to memory, the AND gates of the desired source are opened to the Adder Input Bus. The output of the Adder is then AND and OR gated to the selected memory module and the Parity Generator. The word is entered into the module's data register along with the parity check bit.

Branch instructions change the instruction addresses in the Program Register. The path for these changes are from the T Register through the Adder into the Program Register. These Instructions contain the low-order 9 bit address within the instruction word itself. To this may be appended (if the MAP flag is present) the upper 6 bits of the Program Register. In this case, those 6 bits are AND gated into bit positions 1 - 6 of the Adder, and the remaining low-order 9 bits are gated from the T Register to the Adder.

1.7 Computer Timing

The computer retrieves and executes instructions during specified intervals determined by the timing generator in the computer. The intervals are machine cycles named the Instruction (I) cycle and the Execution (E) cycles. The E cycles are subscripted 1 and 2 to differentiate between them.

Every instruction is obtained from memory and decoded during an I cycle. The I cycle will be followed by one or more E cycles or an I cycle, depending on the timing required by the particular instruction decoded. At the completion of the last cycle required by an instruction, the I cycle for the next instruction is generated, thus an I cycle may be followed by an E_1 cycle, an E_1 and E_2 cycle, or an I cycle.

Each I and E cycle is composed of 16 shorter time units designated "T" times, e.g., T1, T2, T3, etc., these 16 basic pulses are gated to produce supplementary timing pulses, generally of longer durations, to perform the various functions required to decode and execute an instruction. The basic timing signals are shown in Figure 1.7-1.

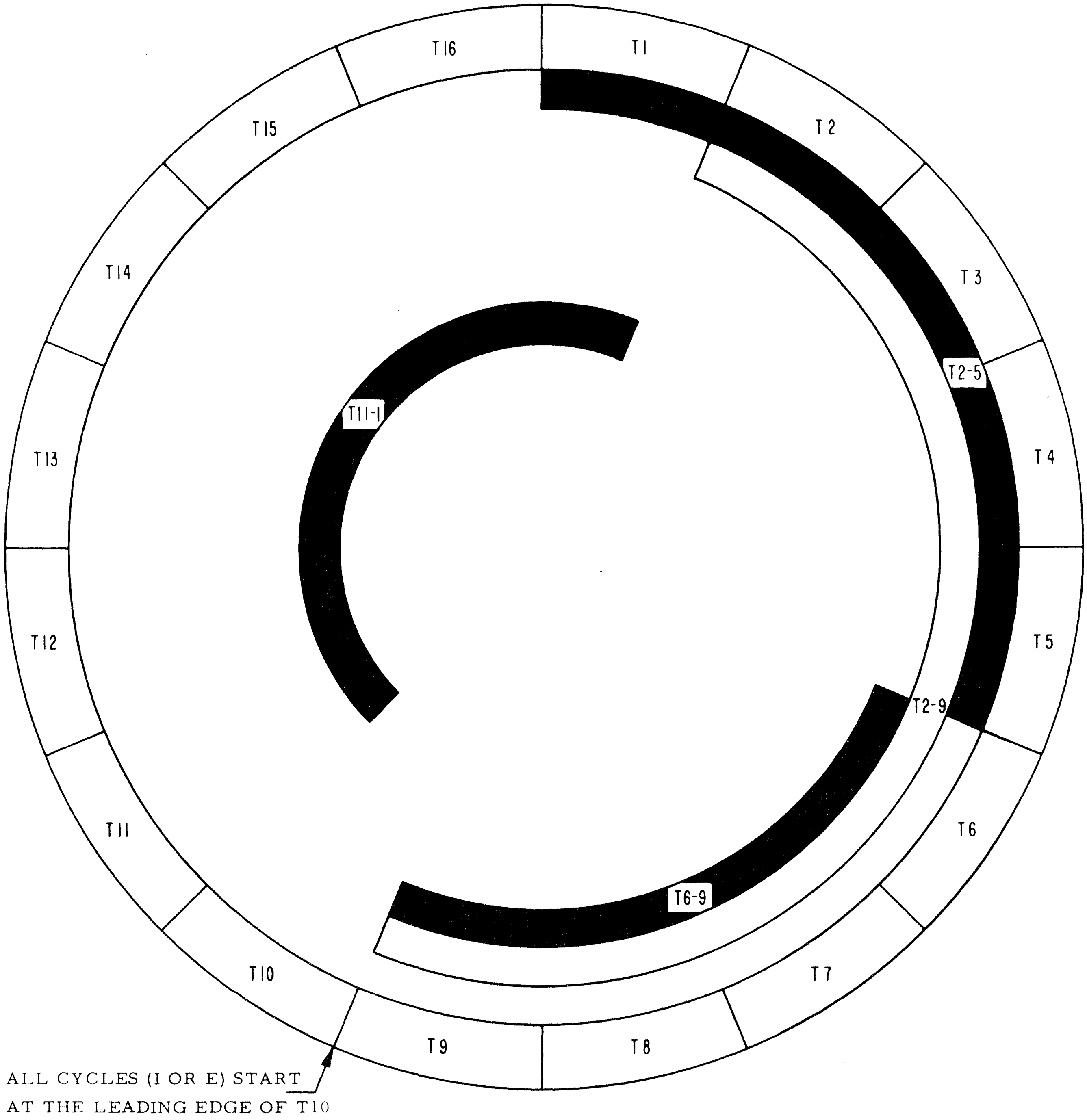


Figure 1.7-1 Basic Computer Timing

The sequence of the timing signals for all instructions revolves around the read or write memory sequences. The T1 timing pulse occurs when a memory word has been read from the memory and is available for use by the control and arithmetic units. The computer word is then generally held in active registers (i. e., Instruction, Transfer and Accumulator registers) until the next T1 time when it may be replaced by the next memory word.

The memory requires approximately 600 nanoseconds to obtain a word from the addressed magnetic cores. In order for this word to be available by T1 time, the memory "read" command is given at T9 time of the preceeding cycle. The memory output word is then present on the Memory Output Bus from the beginning of T14 through the end of T1. During this period, the memory word is applied to the steering inputs of the Instruction Register and T Register flip-flops and to the cable driver circuits, source of the I/O output bus. The memory word is also held in the Memory Data Register until replaced by the next word from memory, at the next T14 time.

During I cycles, the memory word is treated as an instruction word and is loaded into both the Instruction and T Registers at T1 time. By T2 time the operation code has been decoded. If it is amemory reference instruction, the T Register supplies the nine least significant bits of the operand address. If the instruction is not a memory reference instruction, the contents of the T Register are ignored. The T register is used for data transfers during the execution of many augmented instructions.

A "1" in the Indirect flag (bit 5) position of a memory reference instruction word does not affect the base or indexed address in any manner but does force a memory read command to the memory. At the same time, the address output of the adder is gated to the Memory Address Register and the timing control is forced to remain in the I cycle (termed the "Indirect I Cycle"). The word subsequently read from memory and put in the T Register by the next T1 timing signal is treated as a 14-bit address with the address bits located in the 14 low-order positions of the register. The two high-order bits of this word are loaded into bit positions 4 and 5 of the Instruction Register replacing the previous contents of those positions. The two bits are the Index and Indirect flags respectively.

The address bits in the T Register are gated to the 14 low-order positions of the Adder. If the MAP flag is a "1",the one most significant bit (bit 1) of the Program Register is gated to the bit 1 position of the Adder. If the MAP flag is a "0", a "0" is gated to the bit 1 position of the Adder. At the same time, the "new" Index flag is tested and, if a "1", the contents of the B Accumulator are summed with the base address. If the "new" Indirect flag is a "1" the address is gated to the Memory Address Register, a memory "read" command is generated and the computer is again held in the I cycle.

If the "new" or the original Indirect flag is a "0" the computer timing is allowed to advance to the E_1 cycle and the address output of the Adder, either direct or indirect, is gated to the Memory Address Register. Either a memory "read" or a memory "write" signal may be generated depending on the operation code contained in the current instruction.

For those instructions which are to read a word from memory, the word will be loaded into the T Register at T1 time. From the T Register the word may then be gated to an arithmetic register, or to the Program Register depending on the operation code in the Instruction Register. When words are to be written into the memory, they may come from an accumulator register, the program register or an input/output device. Regardless of the source, the word must be gated to the Memory Data Register during the period T12 through T1. This timing is necessary due to the memory timing that starts the write operation at T2 time. The period from T9, when the write command was given, until T2 time is used to erase the previous word from the addressed memory location.

2.0 810A Input/Output Description

This section of the Interface Manual describes in detail the Input/Output data flow and timing, the SEL Data Terminal, the instructions available for I/O control and I/O parity mechanics exclusive of Block Transfer Channels which are discussed in Section 3.

2.1 Input/Output Instructions

Six instructions are provided to perform data input/output and external unit control. The basic operation of these instructions is described in this paragraph. A more detailed description of the I/O hardware operation is contained in paragraph 2.5.

Two instructions, A Input (AIP) and A Output (AOP) are provided to enable words or characters to be transferred betwen the A accumulator and peripheral units. These instructions provide a convenient character assembly/disassembly capability. Each of these instructions occupies a single memory location. The two instructions, Memory Input (MIP) and Memory Output (MOP), enable words or characters to be transferred directly between specified memory locations and peripheral units. The instruction Command External Unit (CEU) enables all system units connected to the computer to be controlled by the program. The CEU instruction is used to initiate Block Transfer Control channels as well as to control computer peripheral units and special system units. The Test External Unit (TEU) instruction is provided to enable system units to be tested by the computer. The test result causes the instruction following the TEU to be either executed or skipped. Two memory locations are required to store the MIP, MOP, CEU and TEU instructions.

The basic, automatic execution sequence for all I/O instructions consists of three steps:

(1) Connect the unit specified by the instruction to the I/O bus.

(2) Execute the transfer directly between the unit and the A accumulator or memory.

(3) Disconnect the unit from the I/O bus.

Three very significant features of this execution sequence are:

(1) The unit is always specified by the I/O instruction,

(2) the unit is always connected to and disconnected from the computer by the execution of the instruction, and

(3) data transfers are always made directly between the specified unit and the computer with no intermediate buffering.

The result of these three features is that the computer I/O structure is always available for use without testing. It is never "busy", except during the times that I/O instructions are being executed. No channel testing or selection is ever required. In addition, no unit selection instructions are required, since each I/O instruction causes the unit specified by the instruction to be selected for transfer.

Data or command word transfer instructions may be executed in either of two modes - Wait Mode or Skip Mode.

Wait Mode

In this mode, the transfer is not made until the unit sends a "Ready" signal to the computer. The computer continues to test for the Ready signal each machine cycle and then executes the transfer during the first cycle following the recognition of the Ready signal. After the transfer, the unit is disconnected and the next instruction in sequence is executed. The specific meaning of the Ready signal is defined in each I/O instruction description.

Skip Mode

In this mode, the Ready signal is tested only once. If the Ready signal is present, the transfer is executed. The Program Register is then advanced by two, which causes the next instruction to be skipped. If the unit indicates "Not Ready", the unit is disconnected and the Program Register is advanced by one. This conditional skip feature enables all I/O instructions (except TEU) to perform the total function of "Connect Unit, Test for Ready, Transfer if Ready, and Disconnect Unit".

Execution of the TEU instruction requires no Ready test. An on-line unit is always "Ready" to be tested. The test word is always transferred to the unit and a Test Return signal is tested. The result of the test is a conditional skip of the next instruction.

In addition to providing selectable execution modes, the two-word I/O instruction (MIP, MOP, CEU and TEU) provide two selectable operand addressing modes - Immediate Mode and Address Mode.

Immediate Mode

In this mode, the second instruction word is treated as the operand. In executing MOP, CEU and TEU instructions, the contents of the second instruction location are transferred to the specified unit. MIP execution consists of transferring a word or character from a specified unit into the second instruction location.

Address Mode

In this mode, the second instruction word is interpreted as the operand address. The indirect address format is used in the second instruction word. Therefore, indexing and indirect chaining may be used in addressing the operand.

Address Mode (continued)

The addressing mode is specified in the first instruction word by the value of the Indirect Address Flag (I). If I is a ONE, the Address Mode is executed.

The format of the I/O instruction words is show below. The specific coding used in each instruction is defined in the individual instruction descriptions.

FIRST WORD

C_1	C_2	C_3	C_4	R	I	M	A_1	A_2	W	U_1	U_2	U_3	U_4	U_5	U_6
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

SECOND WORD, ADDRESS MODE

X	I														
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

SECOND WORD, IMMEDIATE MODE

16 BIT OPERAND															
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

DEFINITIONS

SYMBOL	DEFINITION	CONTAINED IN
C	Command Code	All I/O Instructions
R	Character Merge Flag	AIP
I	Indirect Address Flag	MIP, MOP, CEU, TEU
M	Map Bit	MIP, MOP, CEU, TEU
A	Augmented Command Code	All I/O Instructions
W	Wait Flag	All except TEU
U	Unit Number (00 - 77_8)	All I/O Instructions
X	Index Flag	MIP, MOP, CEU, TEU

CEU

Command External Unit n

13.01M.0
13.01M.1 (WAIT)

Word Format:

Word 1

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	0	1	1	0	I	M	0	0	W	UNIT NO. (n)					

Word 2 (Address Mode)

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
X	I	14-BIT ADDRESS													

Description: Transfers the command (up to 16 bits) contained in the specified memory location to unit n.

Operand Address Modes: Immediate (I = 0), Address (I = 1) (Word 1)

Note: To reference a 15-bit address, M is required (M = 1) to append the most significant 15th bit to the 14-bit address.

Execution Modes: Skip (W = 0), Wait (W = 1)

Transfer Criterion: A unit answers "Ready" to a CEU test if the unit can immediately start execution of any new function command.

Note: The bits in most unit command codes are microprogrammed. Hence, either one or several function commands may be transferred to a unit by execution of a single CEU instruction.

Timing: 3 cycles + wait

Registers Affected: None

Indicators: I/O WAIT

TEU 13.01M.2

Test External Unit n

Word Format:

Word 1

1	0	1	1	0	I	M	0	1	0	UNIT NO. (n)					
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Word 2 (Address Mode)

X	I	14-BIT ADDRESS													
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Description: Transfers the test code (up to 16 bits) contained in the specified memory location to unit n. A return signal from the unit is then tested, and the program register is advanced accordingly. If the return signal indicates a "Ready" or "Go" condition, the next instruction in sequence is skipped. A return signal indicating a "Not Ready" or abnormal condition causes the next instruction to be executed.

Operand Address Modes: Immediate (I = 0), Address (I = 1) (Word 1)

Note: To reference a 15-bit address, M is required (M = 1) to append the most significant 15th bit to the 14-bit address.

Execution Modes: This instruction is always executed in the same mode. An on-line unit is always "Ready" to accept a test code. Therefore, the code is always transferred and the return is always tested. The Wait Flag is not used.

Timing: 3 cycles

Registers Affected: None

Indicators: None

AOP

Accumulator Word Output to Unit n

1700
1701 (WAIT)

Word Format:

1	1	1	1	0	0	0	0	0	W	UNIT NO. (n)					
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Description: Transfers a word from the A accumulator to unit n. Character oriented units accept only bits A_0 - A_7.

Execution Modes: Skip (W = 0), Wait (W = 1)

Transfer Criterion: A unit answers "Ready" to an AOP test if the unit can immediately receive a new word or character.

Timing: 3 cycles + wait

Registers Affected: None

Indicators: I/O WAIT

AIP

NO MERGE **1702**
1703 (WAIT)
MERGE **1742**
1743 (WAIT)

Accumulator Word Input From Unit n

Word Format:

1	1	1	1	R	0	0	0	1	W	UNIT NO. (n)					
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Description: Transfers a word or character from unit n into the A accumulator. Character oriented units transfer characters into bits A_8 - A_{15}.

Note: This instruction contains a convenient provision for character assembly in the A accumulator. If the optional Merge Flag (R) is a ONE, bits A_0 - A_7 remain unchanged by execution of an AIP instruction. Therefore, an 8-bit character can be read and merged with one previously read and shifted left eight bit positions. If R is ZERO, the A accumulator is cleared prior to the input of a character or word.

Execution Modes: Skip (W = 0), Wait (W'= 1)

Transfer Criterion: A unit answers "Ready" to an AIP test if the unit has a word or character ready for immediate transfer.

Timing: 3 cycles + wait

Registers Affected: A accumulator

Indicators: I/O WAIT

MOP

Memory Word Output to Unit n

17.01M.4
17.01M.5 (WAIT)

Word Format:

Word 1

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1	1	1	0	I	M	1	0	W	UNIT NO. (n)					

Word 2
(Address Mode)

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
X	I	14-BIT ADDRESS													

Description: Transfers a word from the specified memory location to unit n. Character oriented units accept only the 7 most significant bits of the specified memory location.

Operand Address Modes: Immediate (I = 0), Address (I = 1) (Word 1)

Note: To reference a 15-bit address, M is required (M = 1) to append the most significant 15th bit to the 14-bit address.

Execution Modes: Skip (W = 0), Wait (W = 1)

Transfer Criterion: A unit answers "Ready" to an MOP test if the unit can immediately receive a new word or character.

Timing: 3 cycles + wait

Registers Affected: None

Indicators: I/O WAIT

MIP

17.01M.6
17.01M.7 (WAIT)

Memory Word Input from Unit n

Word Format:

1	1	1	1	0	I	M	1	1	W	UNIT NO. (n)					
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Word 1

X	I	14-BIT ADDRESS													
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Word 2
(Address Mode)

Description: Transfers a word or character from unit n to the specified memory location. Character oriented units transfer characters into the 7 least significant bits of the specified memory location.

Operand
Address Modes: Immediate (I = 0), Address (I = 1) (Word 1)

Note: To reference a 15-bit address, M is required (M = 1) to append the most significant 15th bit to the 14-bit address.

Execution Modes: Skip (W = 0), Wait (W = 1)

Transfer
Criterion: A unit answers "Ready" to an MIP test if the unit has a word or character ready for immediate transfer.

Timing: 3 cycles + wait

Registers
Affected: None

Indicators: I/O WAIT

2.2 I/O - Computer Data Flow

The logic which is used to perform input and output data transfers is shown in Figure 2.2-1. Only one I/O device is shown, but all others will connect in the same manner. Note that each connecting cable is also applied to a plug to which another device may be connected in parallel with the first. This "daisy-chaining" of devices (units) allows all units to appear identical from the mainframe.

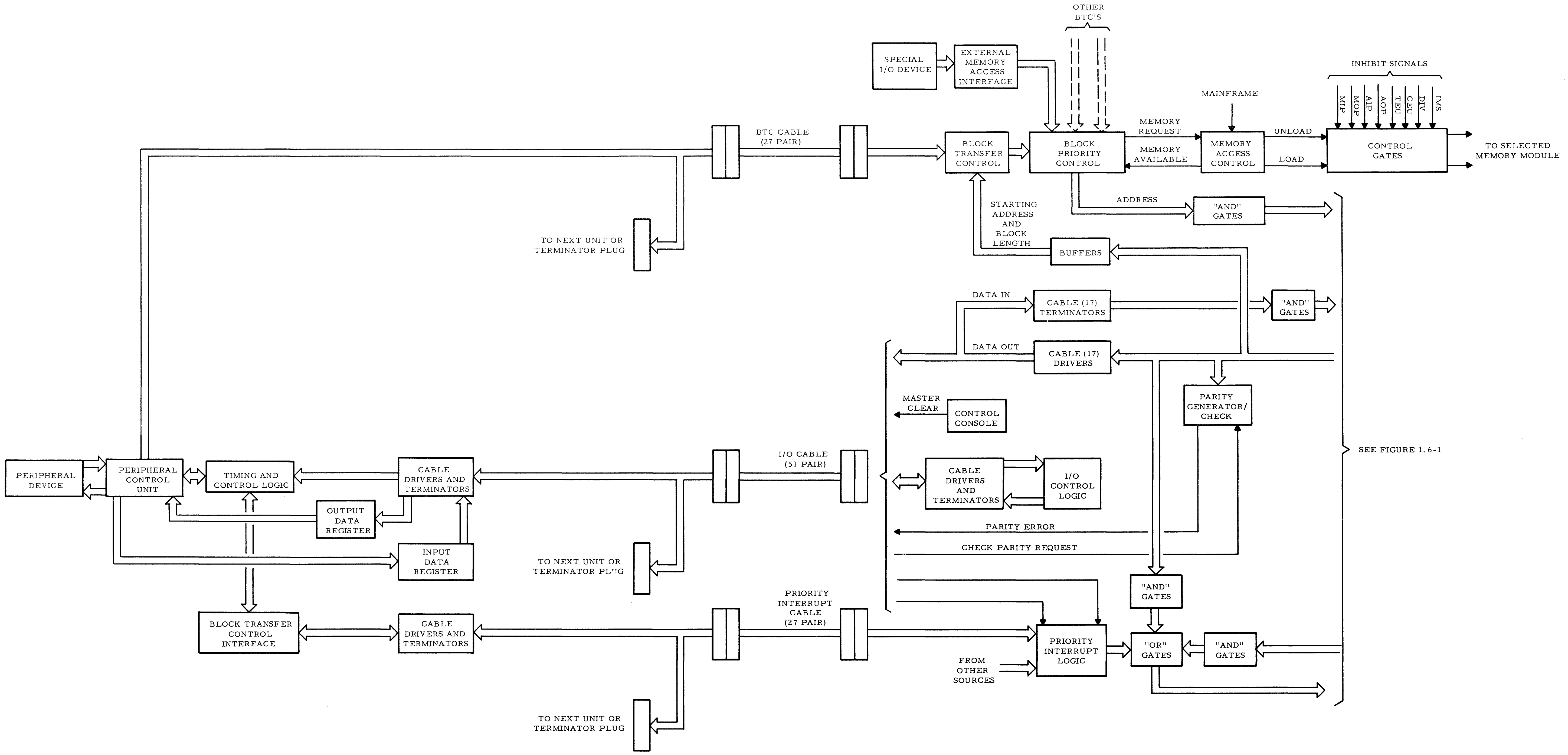
The CEU, TEU and MOP instructions all call the second word containing the command, test and data bits from memory. This second word is read from the magnetic cores into the Memory Data Register. The outputs of this register connect through the Memory Output OR Gates to the Parity Check circuits, the Memory Output Bus AND gates and the I/O Cable Drivers. The word remains in the Memory Data Register for an entire memory cycle (one word time) and, for the CEU, TEU and MOP instructions, is gated through the Cable Drivers to the I/O units.

The control signals necessary to transfer the memory word and to select the proper unit are supplied from the I/O control circuits. The memory word and control signals are supplied to all units in parallel. Only the one unit selected by the unique unit number code will be able to accept the memory word and use the supplied control signals.

The memory word and control signals are gated through cable terminators in the Data Terminal by the unit number code. The function and test bits of the CEU and TEU second words are gated by control signals to set control latches or to jointly test for various unit conditions. The data word supplied by the MOP instruction is loaded into an Output Data Register. The Data Terminal returns signals to the mainframe upon the acceptance of the command, test or data bits. The TEU instruction also causes a "high" or "low" level to be generated on a command "sense return" line to indicate the status of the condition tested.

The AOP instruction causes the data word in the A Accumulator to be transferred to the Output Data Register in the selected unit. The path data is from the A Accumulator through the Adder, the Memory Output OR Gates and the I/O Cable Drivers to the Output Data Register.

The MIP instruction gates the input word from the unit to the Memory Data Register. The data path followed by the input word is from the unit Input Data Register through the unit cable drivers, the "Data In" cable terminators, the Memory Input AND Gates, Memory Input OR Gates to the Data Register of the selected Memory Module.



A2303

Figure 2.2-1 Detailed Block Diagram - Computer I/O Interface Diagram

The AIP instruction which inputs a data word into the A Accumulator, follows the same path up to the Memory Input OR Gates. Here, instead of being gated to a Memory Module, the data word is applied through the Memory Output OR Gates. The word is then gated through the intrinsic AND circuits into the T Register. Later in the execution cycle, the word is transferred from the T Register through the Adder to the A Accumulator.

The two standard I/O priority interrupts (described in paragraph 2.4.4) provide specific memory addresses to the mainframe when they occur. These addresses are forced through hardware circuits onto the Memory Output Bus and then to the T Register. At the same time, a hardware Store Place and Branch (SPB) Indirect instruction code is forced onto the Memory Output Bus and then to the Instruction Register. Each memory location addressed by the interrupt is reserved for the starting address of the priority interrupt servicing program for that specific interrupt.

The block transfer control (described in Section 3) utilizes specific memory locations to hold the block count and starting memory address. The data path for these memory words is from the Data Register in the selected Memory Module, through the Memory Output OR Gates, the Block Transfer Control buffers into the Block Transfer Control. The return path shown on the diagram is for the "current" memory address which can be stored in memory, if requested by the program in a CEU instruction.

2.3 I/O - Computer Timing

Input/Output instructions are all augmented instructions containing operation codes of 13_8 or 17_8 and all, except AIP and AOP, are formed by two memory words. The first word contains the operation code, three to five augment code bits and Indirect, MAP and Wait flags. The single-word AIP and AOP instructions contain the same data plus a Merge flag in the AIP instruction. (The function and use of these flags are explained in detail in paragraph 2.5 of this manual and chapters 2 and 4 of the SEL 810A Reference Manual.)

The first word (and only word for AIP and AOP instructions) of the I/O instructions is loaded into the Instruction Register during the I cycle. This word remains in the instruction register for the duration of the instruction execution period. Bit positions 10 through 15 of the instruction register hold the unit number code bits during the entire instruction period. The outputs of the six unit number flip-flops are connected through cable drivers to the computer I/O Bus. The unit specified by the unique unit number code is the only unit connected to the data and command buses during the current instruction.

The I, E_1, and E_2 cycles are produced by a group of mainframe circuits termed the Control Cycle Generator. The Generator advances from I cycle to E_1 cycle and back to I cycle for "direct" memory reference instruction. It advances into the E_2 cycle for the Increment Memory and Skip (IMS) and Compare Memory and the "A" Accumulator (CMA) instruction. Most augmented instructions other than I/O instructions require only the I cycle for unloading the instruction AND executing it. In the case of shift instructions, where four lateral shifts of the accumulators are performed during each cycle, the Control Cycle Generator is held in the current I cycle until the specified number of shifts are completed.

During I/O instructions, the Control Cycle Generator is held in the I cycle for at least one additional memory cycle time. This is to allow the mainframe to test the unit to determine that the unit is ready to receive a command of data. This one-cycle delay is essentially a propagation delay so that time is allowed for the unit to respond to the test. If the unit is located a long distance (i. e., several hundred feet or more) away from the mainframe, the propagation delay of the query and response signals may be significant.

The return signal consits of two parts: (1) a return synchronizing pulse that causes the Control Cycle Generator to advance to the E_1 cycle; (2) a return test signal that is either "high" to indicate the unit is ready or "low" to indicate that the unit is not ready. If the instruction is programmed for the Skip Mode, a "high" test return signal causes the completion of the instruction during the E_1 cycle and advances the program register twice to skip the next sequential instruction. If the test return is "low", the E_1 cycle occurs but no signals are generated to execute the instruction and the program register is advanced once to execute the next sequential instruction.

In the Wait Mode of operation, the second I cycleis repeated indefinitely until the selected unit is ready. Both the return synchronizing pulse and the "high" test return are generated when the ready condition occurs. The Control Cycle Generator then advances the E_1 cycle, the instruction execution signals are generated and the program register is advanced once to call the next sequential instruction.

The preceeding paragraphs describe the general computer timing sequences when an I/O instruction is in the Immediate Address Mode (i. e., the data or command word is in the second word of the instruction or in the A accumulator). When the instruction is programmed for the Direct Address Mode, the sequence is slightly different.

The presence of a "1" in the bit 5 (Indirect flag) position of the Instruction Register forces the Control Cycle Generator into an "Indirect I" cycle. During this cycle, the second I cycle specified in the preceeding text, another memory "read" is accomplished. This second word (stored in indirect address format) is taken from the memory location following the first word of the instruction. The 14-bit address portion of the word is then loaded into the 14 least significant bit positions of the T register. The Indirect and Index flags possible contained in bit positions 0 and 1 of the address word are loaded into bit positions 4 and 5 of the Instruction Register.

The Indirect flag contained in the address word replaces the Indirect flag contained in the first word and is stored in the Instruction Register. If the new flag is a "0", the computer now waits until the return synchronizing signal is received from the unit. If the "new" flag is a "1", another "Indirect I" cycle is accomplished and the Indirect flag in that address word replaces the previous flag in the instruction register. The "Indirect I" cycles are repeated until a "0" is loaded into bit 5 of the Instruction Register with each direct address specifying the memory location of the next address. If the return synchronizing signal occurs during one of these "Indirect I" cycles, it is stored until a "0" finally appears in bit 5 of the Instruction Register.

When the final indirect address has been called into the T register and the return synchronizing signal is received, the address held in the T register specifies the source or destination memory cell for the data, command or test word used in the E_1 cycle.

For command, test and output data transfer instructions, this address specifies the source. In the Immediate Address Mode the source is always the second word of the instruction or the A Accumulator. The execution of the CEU, TEU and MOP instruction requires that the command, test or data word be read from memory and be placed on the I/O Bus. This is accomplished by reading the memory output word into the Memory Data Register during the E_1 cycle (when the unit is ready) and gating the outputs of this register to the I/O Bus. During an AOP instruction the outputs of the A Accumulator are gated through the Adder and Memory Output OR Gates to the I/O Bus.

The input data transfer instructions, MIP and AIP, require a slightly different timing arrangement during the E_1 cycle. The MIP instructions requires that the data word supplied by the unit be present at the inputs of the Memory Data Register during T14 through T1 times. This is because the memory "write" cycle clears the addressed cores and the data register to "0's" during the first half of the memory cycle from T9 through T14 times. The memory timing circuits then generate a "Load Data Register" signal at T1 time to load the new word into the Data Register. This word is then subsequently copied into the addressed magnetic cores.

The AIP instruction requires that the input word be on the Memory Output Bus by T1 time when it is loaded into the T Register. The contents of the T Register are then gated through the Adder to the inputs of the A Accumulator. The word is finally loaded into the A Accumulator at T9 time.

2.4 Data Terminal

2.4.1 General

To insure uniform timing in data communications between the mainframe and standard or non-standard peripheral units, Systems Engineering Laboratories developed a standard Data Terminal. This data terminal provides a uniform interface between the mainframe and various types of input/output devices. The Data Terminal consists of two functional cards plus other assorted circuits (Figure 2.4-1). One functional card contains a patchable NOR matrix which is used to provide standard level output signals when the one (of 64) unit code for which it is wired is decoded. The second card contains all of the circuits necessary to answer and return standard timing signals that occur for the various input/output instructions.

2.4.2 Signal Definitions

The computer I/O signals are shown in the upper right hand corner of Figure 2.4-1. These signals are:

Data Transfer Instruction (310) - signifies to the data terminal that an AIP, AOP, MIP, MOP instruction is being executed. The signal is present for the duration of the instruction.

Test Instruction (311) - signifies that a test external unit (TEU) instruction is being executed and is present for the duration of the instruction. The second word of the instruction contains the test code.

Command Instruction (312) - signifies that a Command External Unit (CEU) instruction is being executed and is present for the duration of the instruction. The second word of the instruction contains the function (operation) bits.

Instruction Sync (313) - The instruction sync is used to allow the unit to interrogate its unit code lines, and in turn answer via the unit sync return line. The instruction sync signal is present until the unit answers.

Input/Output (509) - signifies the direction of transfer in conjunction with a Data Transfer Instruction. The signal is present for the duration of the instruction.

Wait Flag (510) - signifies that the instruction being executed contains the computer wait flag bit. The wait flag causes the unit to inhibit answering via the unit sync return until it is ready to perform the desired instruction. The wait flag is not used during a Test Instruction. The flag is present for the duration of the instruction.

Computer Data Here (512) - signifies that the computer has recognized the unit sync return line and the test return - if applicable - and has proceeded into the execution cycle of the instruction. The Computer Data Here is used to notify the unit that the data bits are present on the I/O Bus and can be interrogated or loaded into the Output Register. The signal remains until the unit answers on its Unit Data Accepted line.

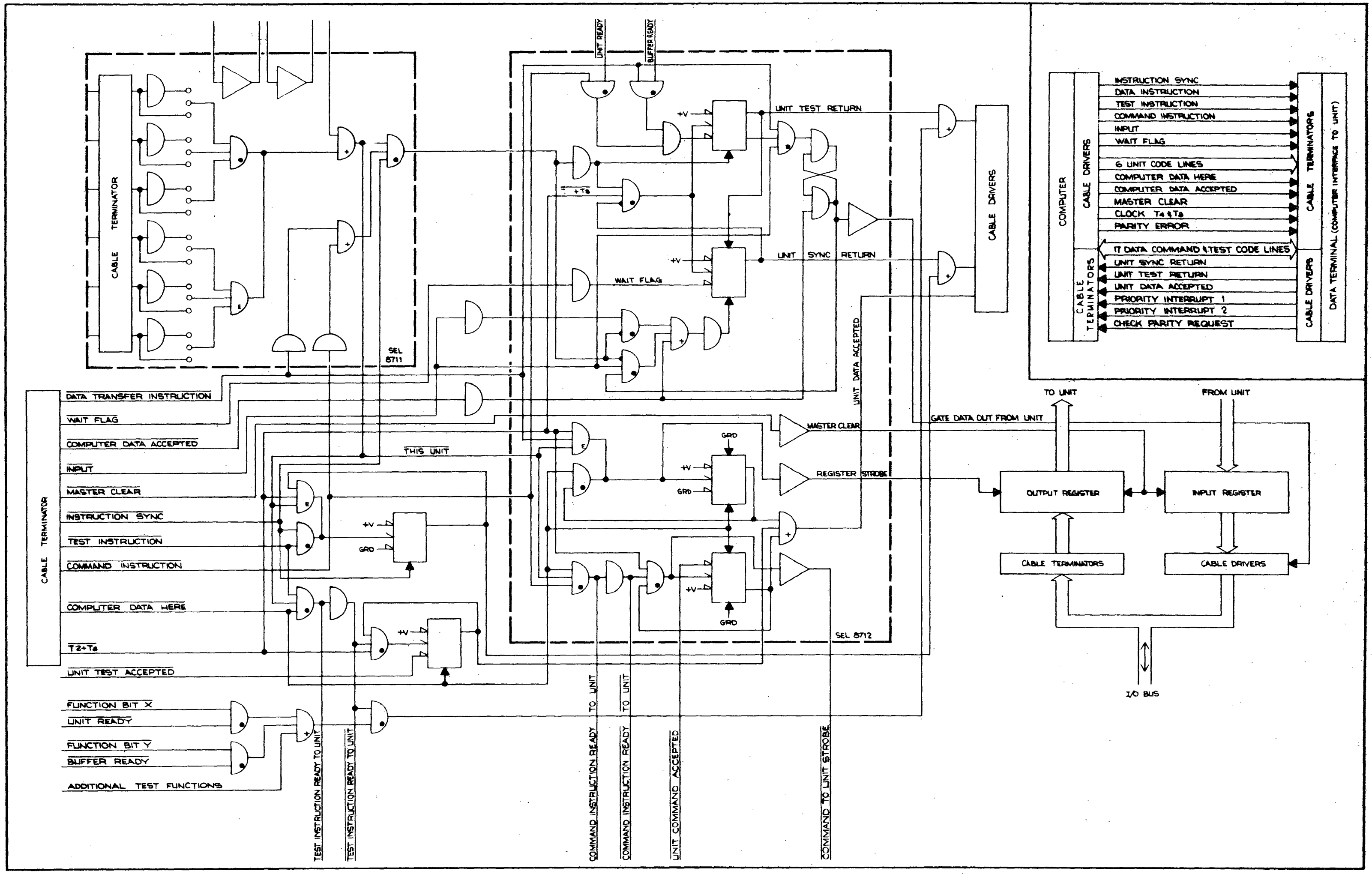


Figure 2.4-1 SEL Data Terminal - Block Diagram

A2295

Computer Data Accepted (513) - signifies that the computer has accepted data from the unit as the result of an AIP, MIP instruction. The signal remains until the unit removes the Unit Data Here signal which caused it (See Unit Data Accepted below).

Unit Test Return (710) - signifies the status of the unit afer receipt of the Instruction Sync from the computer. If the unit is capable of performing the required instruction, the Test Return line is enabled thus allowing the computer program a "skip" of the next instruction. The Test Return signal remains valid until the Instruction Sync signal is removed.

Unit Sync Return (711) - signifies that the unit has recognized the Instruction Sync signal. The signal remains until the computer removes the Instruction Sync signal.

Unit Data Accepted (712) - signifies that the unit has accepted the data word present on the I/O Bus. In the case of the test instruction it notifies the computer that the test return line is valid and can be interrogated. The signal remains until the Computer Data Here signal is removed.

Master Clear (709) - The master clear line is activated by the CLEAR switch on the computer control panel and also by the ICB condition when power is turned on. In the case of the CLEAR switch, the line is a DC level and remains activated as long as the switch is depressed. The ICB condition enables the Master Clear line for approximately 1 second.

$T_2 + T_8$ (511) - The clock line is formed by OR gating the two designated timing pulses together from the master computer clock. Each pulse is 110 nanoseconds in duration.

Parity Error to Unit (505) - signifies that the word transferred to the computer contained a parity error. This line is valid only if the equipment contains the optional I/O parity check capability.

Parity Request from Unit (506) - This line also operates in conjunction with the optional I/O parity check and is enabled during any input data transfer instruction. The line notifies the computer to perform a parity check on the word being transferred.

Parity Bit to Unit (507) - All data words presented to the I/O Bus include a parity bit to allow the unit to perform a parity check if desired.

Parity Bit to Unit (508) - If the I/O Parity option is included, the unit must provide a parity bit for any input data transfer in order that the computer can perform a parity check on the word.

The "Instruction sync" signal from the computer is accompanied by one of the following:

(1) Data Instruction for MOP, MIP, AOP and AIP instructions or

(2) Test Instruction for the TEU instruction or

(3) Command instruction for the CEU instruction.

The input signal is high for the AIP or MIP instruction or low for the AOP or MOP instruction. The six Unit Code lines are always present to select a particular unit. The Computer Data Here signal is present for CEU, TEU, MOP and AOP instructions. The Computer Data Accepted signal is generated near the conclusion of the MIP and AIP instructions. The T2 and T8 clock pulses are the only computer timing pulses used by the data terminal. The "Parity Error" signal is generated when an input parity error has occured and the Check Parity Request signal came from the I/O unit.

The Unit Sync Return is generated by the Instruction Sync signal from the computer immediately when no wait flag is present and after either the unit is ready or the buffer is ready depending whether this is a Command or Data Transfer Instruction if the wait flag is present in the original instruction. The Unit Test Return signal is generated by the Instruction Sync signal if the unit is ready for a Command Instruction or the buffer is ready for a Data Transfer instruction. The Unit Data Accepted signal is generated as a result of an Output (from the computer) Data Transfer which can occur with CEU, MOP or AOP instructions. The Priority Interrupt signals can occur if the standard priority interrupts are enabled and the interrupt condition is present.

2.4.3 Data Terminal Sharing

The Data Terminal as described in the previous sections normally operates with one peripheral device utilizing one unit number. In the case of special-purpose units it is possible to share the unit sync logic and utilize more than one unit number. The particular application usually dictates the most optimum logic configuration, or in some cases where it is required to have more than 64 units connected to the computer I/O structure, it is necessary for many units to share one unit number. Also, each of the I/O cables are only capable of driving 16 loads where each cable terminator presents one load; a condition sometimes requiring sharing of a Data Terminal to conserve circuits.

Figure 2.4-2 is a block diagram of three units sharing one Data Terminal where it is used for command functions to the individual units. It can be seen that each of the various functions (command instruction ready to unit, command instruction strobe to unit, unit ready, command accepted, etc.) must be gated against

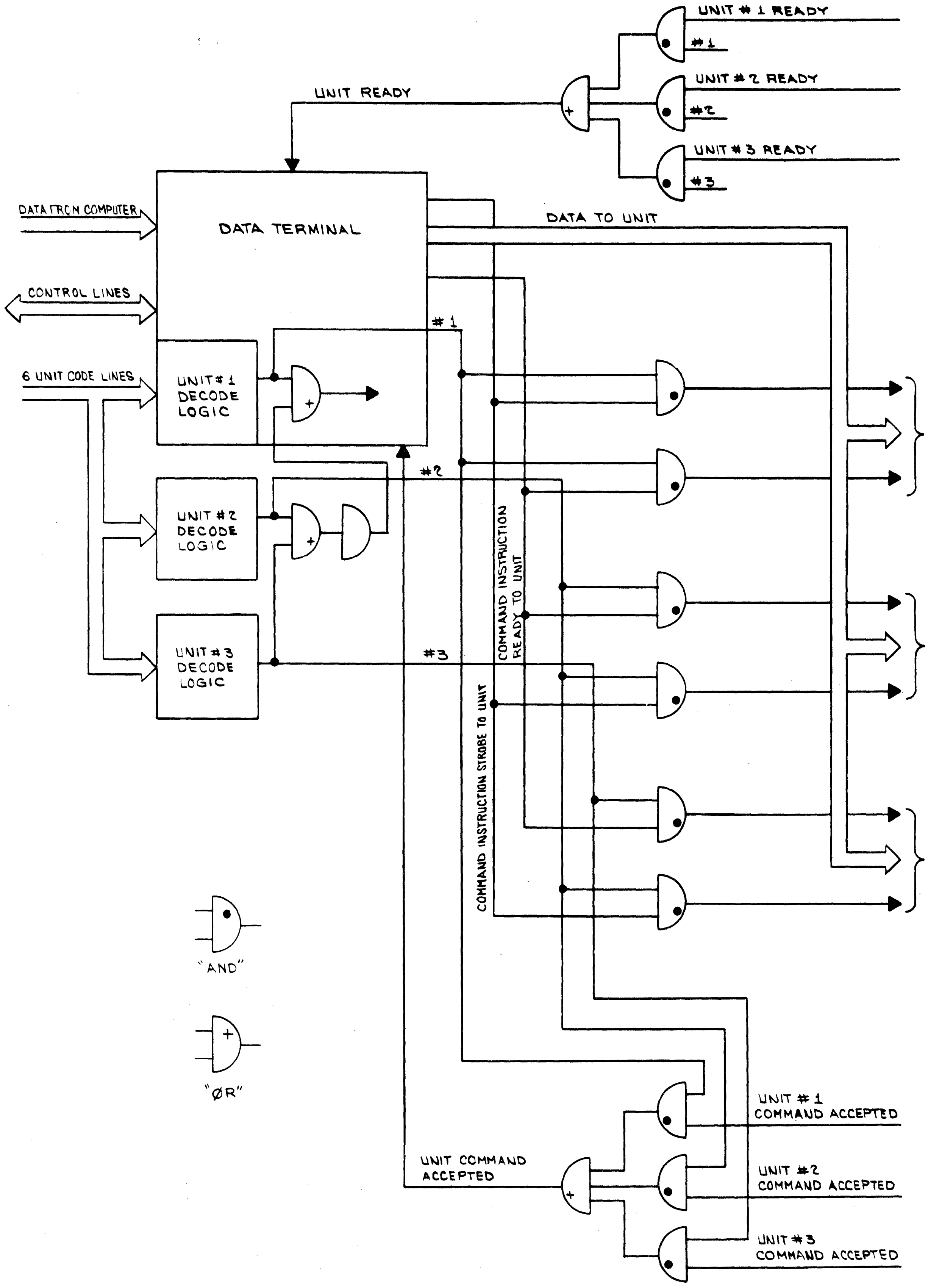


Figure 2.4-2 Shared Data Terminal - Command Sharing

their respective unit decode logic in order to insure that only the addressed unit is commanded. This same approach can be extended as shown in Figure 2.4-3. This block diagram shows an example where three output registers, each having separate unit numbers, are sharing a common Data Terminal. The MOP/AOP instructions are used to output data to the registers and the only additional gating required is to steer the register strobe to the proper unit. The unit ready is also gated with the unit number to enable the skip feature of the instructions to be executed by the computer.

Various schemes can be devised to conserve unit numbers. The most applicable uses of Data Terminal sharing are in the case where a large number of input or output registers (displays, for example) must be connected to the computer I/O Bus. If the registers do not utilize the full computer word, addressing can be done in the spare bits of the computer word such that only one unit number is used. For example, lf 50 eight-bit digital-to-analog converters were to be connected to a Data Terminal, the individual converter address can be coded in the extra bits of the output word.

Data word of MOP or AOP	D/A ADDRESS	D/A WORD

Due to the large number of possible uses, it is not possible to set down any hard or fast rules that apply to Data Terminal sharing. The examples and block diagrams presented are for guidelines and merely illustrate possible applications.

2.4.4 Standard I/O Priority Interrupts

Each standard Data Terminal contains a pair of priority interrupt circuits (See SEL 810A Priority Interrupt, Section 4.0) formed on one functional circuit card. A logic diagram of this card is shown in Figure 2.4-4.

The Computer Data Here signal, used to transfer the function bits of the second word of the CEU instruction, is also used to disable or enable either or both of these two standard priority interrupts. The presence of a "1" in function bit 1 of the second word of the CEU instruction causes either or both interrupts to be enabled. The presence of a "0" in function bit 1 of the second word causes either or both standard interrupt to be disabled. The selection of the interrupt to be enabled or disabled is determined by the presence of "1's" in function bits 2 and 3. If function bit 2 is a "1", the input interrupt will be either enabled or disabled depending on the state of function bit 1. If function bit 3 is a "1", the output interrupt will be enabled or disabled depending on the state of function bit 1.

The enable/disable logic utilizes a function of circuit design of the micrologic flip-flop. If both the J and K inputs to the flip-flop are at +V, any strobe trigger will be ignored. If, however, +V exists on the set input and 0V on the reset steering input, the flip-flop will be set by any trigger strobe that is applied.

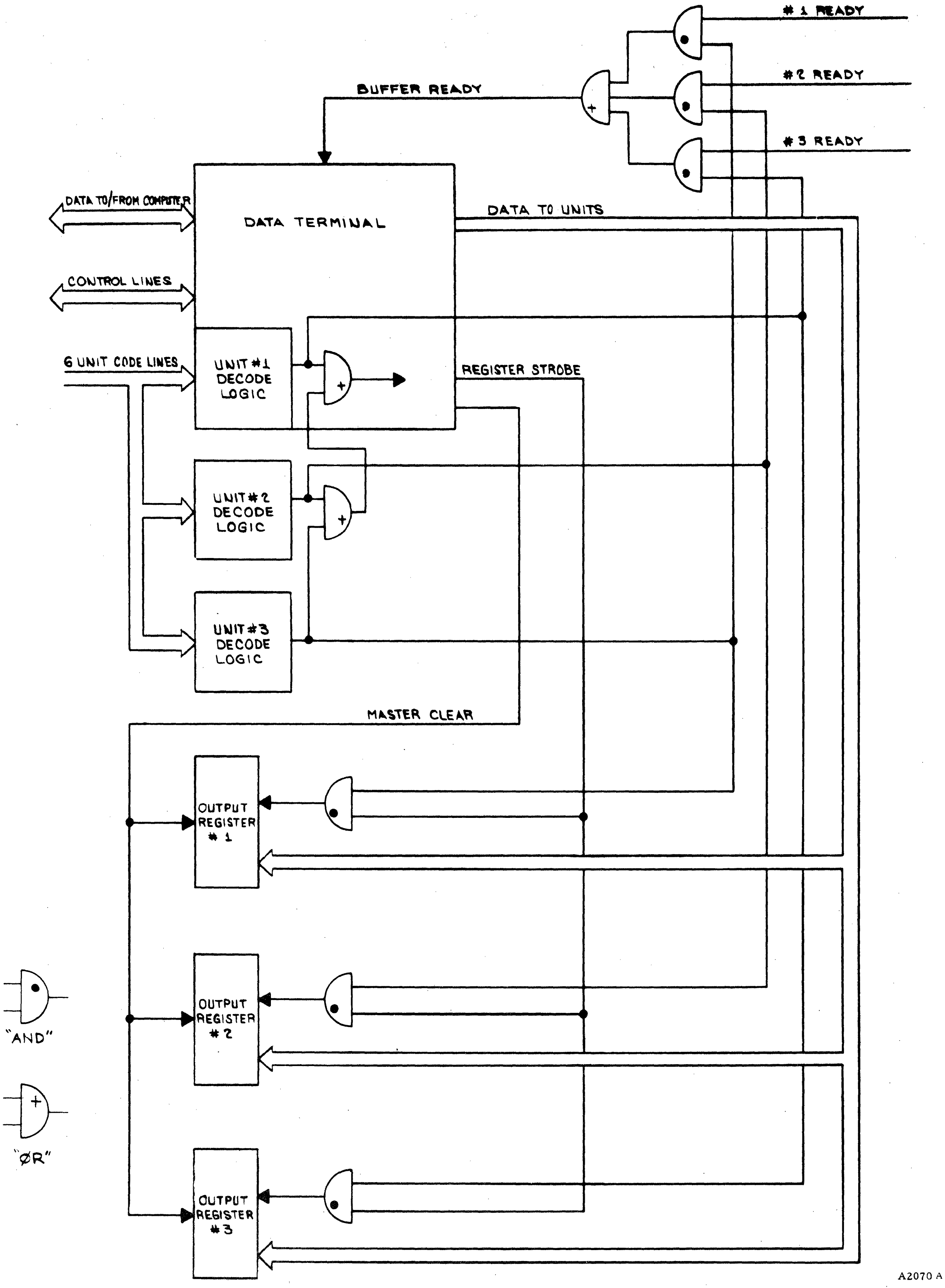
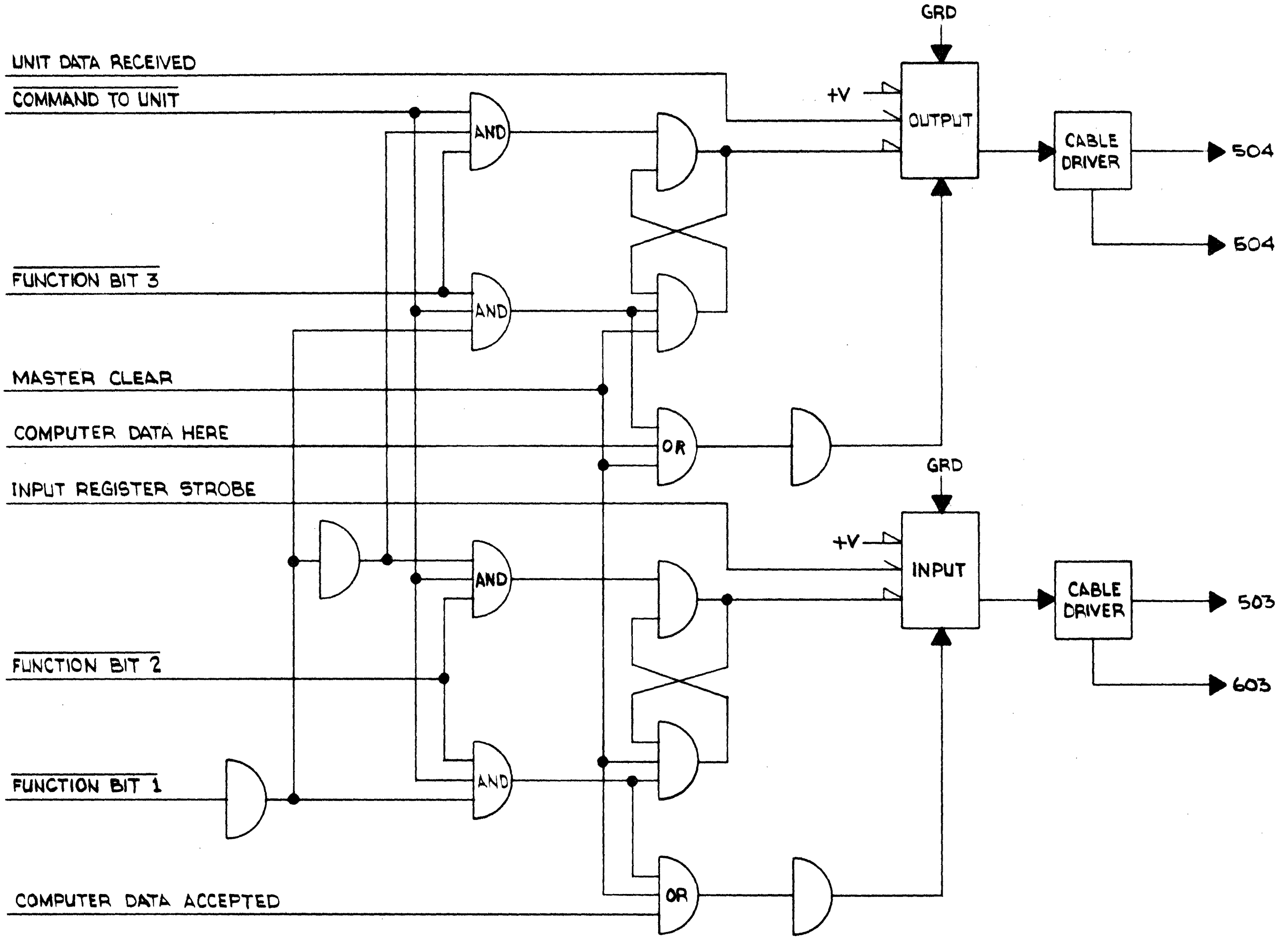


Figure 2.4-3 Shared Data Terminal - Register Sharing



A2077A

Figure 2.4-4 Standard I/O Interrupt - Logic Diagram

When either interrupt channel is enabled, the NOR latch connected to the reset steering input goes to 0 Volts so that the flip-flop may be set. When the channel is reset, this same NOR latch is reset so that +V is applied to the reset steering input thus making the flip-flops impervious to any trigger strobe. The Computer Data Here signal produces a Command to Unit signal which is applied to four AND circuits so that either the input or output interrupt or both may be either enabled or disabled by this command. Note that it is not possible to enable one interrupt while disabling the other; however, the arrangement does allow these interrupts to be selectively enabled or disabled by the same command that causes that unit to perform some operation.

2.5 I/O Instruction Sequences, Timing and Data Flow Descriptions

2.5.1 Command External Unit

2.5.1.1 Function of the Command External Unit Instruction

The Command External Unit Instruction (CEU) is used to prepare an external unit for some subsequent operation. The instruction may:

(1) Set the mode of operation of the unit;
(2) Set the direction of operation of the unit;
(3) Select a sub-unit of the addressed unit for operation;
(4) Enable or disable the standard input and/or output priority interrupts;
(5) Initialize a block transfer operation;
(6) Extract the current memory address of a block transfer operation;
(7) Set the number of characters per word for a character assembly buffer.

The CEU instruction is a two-word instruction which is stored in two sequential memory locations. The instruction may operate in either the Wait or Skip Modes of operation and in either Immediate or Address Modes. The first word of this instruction sets the modes of operation by flags contained in bit positions five and nine.

Bit nine contains the Wait Flag which if set to a "1", causes the Unit's Data Terminal logic, not to answer "Unit Sync Return" until the unit is "Ready", therefore causing the computer to wait. The presence of a "0" in bit 9 of the first word will cause the computer to operate in a Skip Mode. The Skip Mode causes the computer to access the next sequential instruction following the CEU instruction if the unit is ready but to skip the next sequential instruction and obtain the subsequent instruction if the unit is not ready. When the computer is in a wait cycle, it cannot be interrupted by any priority interrupt except for the optional power fail safe/ restore or stall alarm interrupts. If however, the Skip Mode is used, the computer may be programmed through use of a BRU instruction to cycle until the unit is ready before proceeding. This allows the servicing of priority interrupts that might occur between the CEU instruction and the subsequent BRU instruction.

The indirect flag located in bit five position of the first word of the CEU instruction defines whether the instruction will operate in the Immediate Mode or the Address Mode. If the indirect flag is a "0" (Immediate Mode) the second word of the instruction becomes the operand. The Address Mode, indicated by a "1" in bit position five of the first word of the instruction, utilizes the indirect address word format. The second word of the instruction becomes an indirect address of 14 bits plus an index and indirect flag. The indirect flag (bit 1) may be used for indirect chaining, and the index flag (bit 0) for an index operation. If the MAP bit (bit 6) of the first word of the instruction is a "1", the most significant bit of the Program Register is appended to the 14 bit address. The inclusion of the MAP bit modifier allows the operand to be called anywhere in the entire 32K of memory. If the MAP bit contains a "0", the 15th or most significant bit of the address will always be zero.

The final word obtained from memory at the end of an indirect chain is identical in format to the second word called from memory in the Immediate Mode. (Specific bit assignments are shown in Appendix A).

This second word may contain up to 16 function bits. These function bits specify:

(1) In bit 0, BTC initialize;

(2) Bit 1, Priority interrupt enable/disable;

(3) In bits 2 and 3, the 2 standard I/O interrupts to be either enabled or disabled depending on the state of bit 1;

(4) In bits 4 through 9, function code bits which specify the operation, mode, etc., to which the unit is to respond;

(5) In bits 10 through 12, the tape unit no. if the addressed unit is a tape control unit;

(6) In bit 13, a command to load the current address of a block transfer control unit a specified memory location;

(7) In bits 14 and 15 the number of characters into which a computer word is to be disassembled for output transfer if the address unit contains a character assembly buffer.

2.5.1.2 CEU Peripheral-Computer Timing Relationships

The basic computer timing, described in more detail in paragraph 1.7, consists of a continuous series of 1.75 microsecond cycles. This cycle time is the time required for the memory to perform a complete read/write cycle. The cycle during which the instruction is unloaded from memory is referred to as the I (instruction) cycle; the memory cycles during which data is unloaded from or loaded into memory are referred to as E (execution) cycles.

Each CEU instruction consists of at least two I cycles and one E cycle. The first I cycle is the cycle during which the first word of the instruction is unloaded from memory. The second I cycle is the time during which the unit is tested to see whether it is ready to accept a command from the computer. The second I cycle may be repeated indefinitely if the instruction is used in the Wait Mode. Additional I cycles are also required if the instruction is used in the Address Mode requiring more than one indirect cycles. The first indirect cycle is performed while the unit is being tested. When the second word containing the data (function code bits, etc.) is unloaded from memory at the end of an indirect chain or immediately if the Immediate Mode is used, and the unit has responded that it is ready for a transfer of instructions from the computer, then the computer, is allowed to go into an execution cycle referred to as the E_1 cycle.

2.5.1.3 CEU Execution Sequence

Refer to Figure 2.5-1.

The basic instruction, used in the Skip and Immediate Address modes, requires three complete memory cycle times as shown in the timing diagram. Note that during the first I cycle the command instruction is shown as being available from the beginning of T1 time of the first I cycle continuously until the beginning of T1 time of the I cycle of the next subsequent instruction. During this entire period the CEU instruction remains in the Instruction Register in the mainframe and thus provides the CEU enables to both the mainframe circuits and to the selected unit. The Instruction Register also holds the wait flag and the unit number for this period.

At T4 time of the first I cycle, the computer circuits produce an Instruction Sync level which is gated to the selected units circuits by the unit code. If the unit is ready, the INSTRUCTION SYNC signal gates a T2 or T8 pulse to set a latch in the peripheral unit which provides simultaneous Unit Test Return and Unit Sync Return signals back to the mainframe. The Instruction Sync signal which initiated this action in the unit also set a latch in the mainframe which holds the mainframe to the I cycle. When the Unit Sync Return signal comes back to the mainframe it resets this latch and allows the computer to advance into the E_1 cycle during which the second word is unloaded from memory. The Unit Test Return signal is used to perform the skip test. If the Unit Test Return signal is received, the program register will be advanced twice so as to skip the next subsequent instruction. If the Unit Test Return signal does not come back, no Computer Data Here signal will be generated, and the program register will be advanced once and will call the next sequential instruction.

The Unit Test Return signal also provides a Computer Data Here signal from the computer to the unit coincident with the time that the second word of the instruction is available on the I/O Bus. The Computer Data Here signal sets a latch in the unit, which provides a signal back to the computer that states Unit Data Accepted. The philosophical point is that if the unit was ready to return the Unit Sync Return and Unit Test Return signals, then it obviously will be ready to accept the data from the second word when strobed by the Computer Data Here signal.

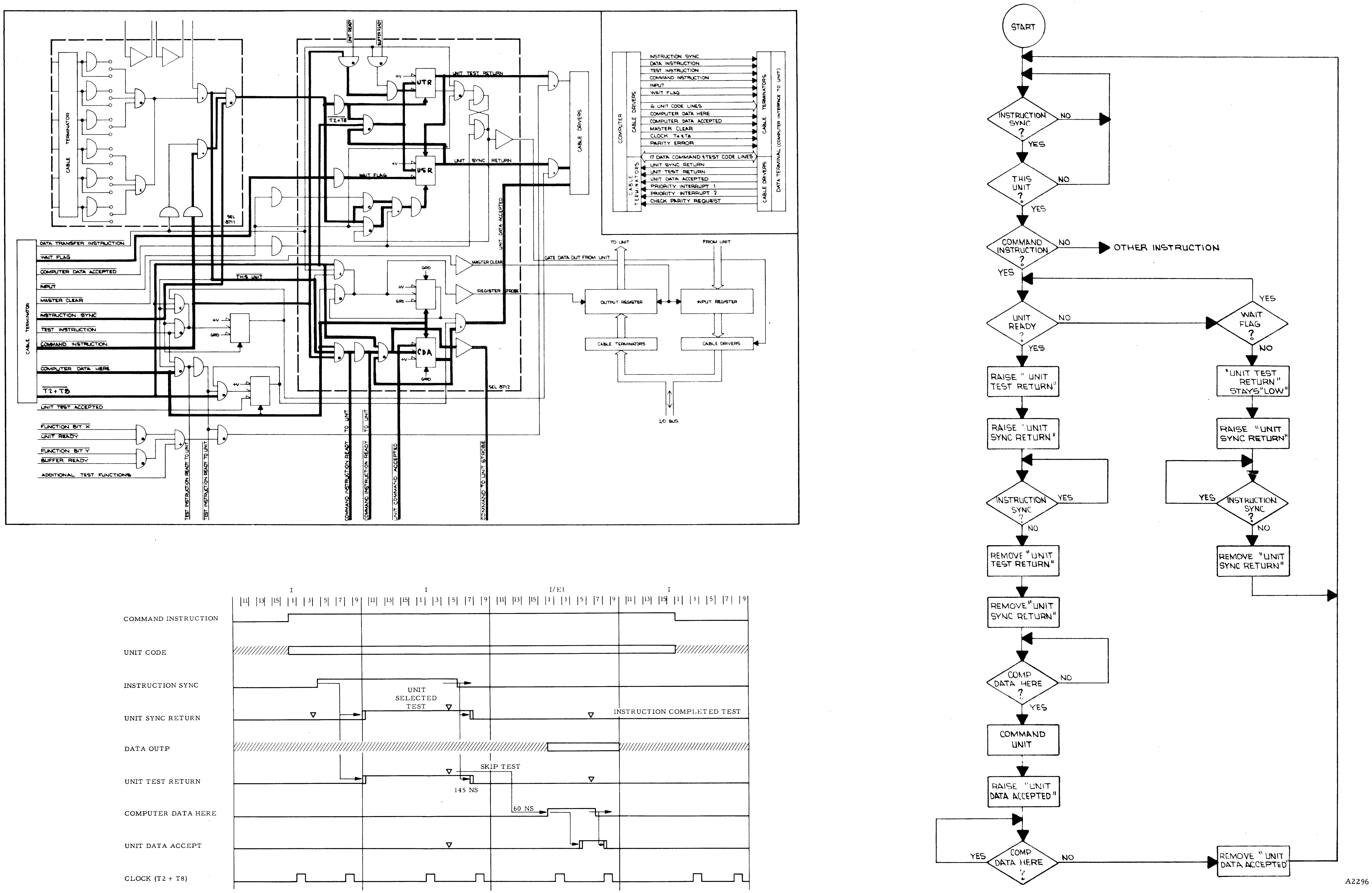


Figure 2.5-1 CEU Flow, Timing and Data Terminal Diagrams

The entire sequence of the CEU instruction as it pertains to the unit is shown in the CEU Flow Diagram (Figure 2.5-1). The first signal produced by the mainframe to the unit is the Instruction Sync level. When that signal occurs, and the unit is selected by the unit code, it will produce the Unit Test Return level at the next T2 or T8 pulse, if the unit is ready, which in turn produces the Unit Sync Return level. Note, that if the unit is not ready when the Instruction Sync level is generated by the mainframe, a test is made to determine if the wait flag is present in the original instruction. If the wait flag is present, the unit will continually test for the unit ready condition, and the mainframe will remain in the I cycle. If the wait flag is not present, then the Unit Test Return signal remains low but the Unit Sync Return signal is raised.

The removal of the INSTRUCTION SYNC level, provides for removing the Unit Sync signal. If the unit is ready, both the Unit Test Return and Unit Sync Return levels are produced. Again, the removal of the Instruction Sync level provides the means of resetting or removing the Unit Test Return and Unit Sync Return signals. The subsequent occurrence of the Computer Data Here signal will command the unit to perform its function by gating the function bits into the proper circuits in the peripheral unit. At the same time it will raise the Unit Data Accepted line. The Computer Data Here signal remains for a long enough period for the unit to answer Data Accepted. The removal of Computer Data Here causes the unit to remove the Unit Data Accepted signal.

2.5.1.4 Detailed Operation of SEL Data Terminal for CEU Instructions

The presence of the CEU Instruction in the instruction register causes the COMMAND INSTRUCTION line at the output of the cable terminator to go to 0 volts (See Figure 2.5-1). This 0 volt level is applied through an inverter and an OR gate on the functional Unit Select card. The output of the OR gate is ANDed with the 0 volt unit select level from the upper OR gate.

The COMMAND INSTRUCTION 0 volt level is also applied to partially enable two other AND gates, one shown on the drawing at the top of functional card 8712 and the other at the bottom of that card. The first gate is fully enabled when the UNIT READY is at 0 volts. The output of the fully enabled AND gate is applied through an OR gate to disable the reset steering input of the UTR flip-flop thus effectively enabling the set steering input of that flip-flop. The other AND gate partially enabled by the COMMAND INSTRUCTION level is shown at the bottom of the drawing of the 8712 functional card and is one input to a gate which is completely enabled by THIS UNIT signal produced by the unit select card and the COMPUTER DATA HERE signal.

The first timing signal to be sent to the data terminal for the CEU Instruction is the INSTRUCTION SYNC signal. This line goes to 0 volts at approximately T4 time and causes the output of the AND gate on the unit select card to go to +V. This "1" level is applied through an inverter to remove the DC reset signal from the UTR flip-flop and to partially enable an AND gate queried by T2 or T8 pulses. The output of the unit select AND gate is also applied through two AND gates on an OR gate to remove the DC reset from the USR flip-flop. The next T2 or T8 pulse to query the AND gate enabled by the inverter output will be gated through to query both the UTR and USR flip-flops. This gated timing pulse will set the UTR flip-flop only if the unit is ready due to the steering arrangement from the UNIT READY and COMMAND INSTRUCTION AND gate connected to the reset steering input. The same timing pulse however will set the USR flip-flop to produce a UNIT SYNC RETURN signal if the instruction is programed for the skip mode. Note that the WAIT FLAG level will be at +V if the wait flag was not in the word format, the CEU instruction. This +V level is inverted and applied as a zero volt level to the reset steering input of the USR flip-flop, thus effectively enabling the set steering input of that flip-flop. If the wait flag was present in the CEU instruction, then the output of the inverter and the input to the reset steering input will be +V. The presence of +V on both the set and reset steering inputs to the flip-flop will make its state impervious to any trigger strobe. Therefore the only way that the UNIT SYNC RETURN signal can be produced is by the DC set input to the USR flip-flop. This will occur only if the UTR flip-flop is set. This will occur only when the unit becomes ready and the reset steering input of the UTR flip-flop is disabled. The steering inputs of the UTR flip-flop are constantly tested by T2 and T8 pulses. Therefore, when the unit UTR flip-flop is set it produces a UNIT TEST RETURN signal which in turn immediately produces the UNIT SYNC RETURN signal by setting the USR flip-flop.

The generation of the UNIT TEST RETURN signal when in the skip mode will cause the program register to advance twice after calling out the second word of the CEU instruction. If the UNIT TEST RETURN signal is not produced, i. e., if the unit test return signal remains low, then the program register will advance to the next instruction following the second word of the CEU instruction. The UNIT SYNC RETURN signal, always produced by this instruction, will release the I/O hold latch in the mainframe to allow the computer to read the second word of the CEU instruction from memory. When this word is available on the I/O bus and on the inputs to the unit, the computer causes the COMPUTER DATA HERE line to go to 0 volts if the unit test return line wasn't true. The 0 volts level removes the DC set input to the CDA flip-flop and the DC reset signal from the UDA flip-flop. The same 0 volt level also completes the enabled to the bottom AND gate on the SEL 8712 card thus producing a +V COMMAND INSTRUCTION READY TO UNIT signal and through an inverter a 0 volt COMMAND INSTRUCTION READY TO UNIT signal. The 0 volts reset output of the CDA flip-flop combines with the COMMAND INSTRUCTION READY TO UNIT to partially enable an AND gate on the trigger input line of the CDA flip-flop. The next T2 or T8 timing pulse will be gated through this AND gate to query the trigger input and to produce a COMMAND TO UNIT STROBE signal through an inverting buffer. The CDA flip-flop will be reset by the first T2 or T8 pulse to occur following the UNIT COMMAND ACCEPTED signal going to 0 volts and thus disabling the set steering input of that flip-flop. The UNIT COMMAND ACCEPTED signal will go to 0 volts as a result of the COMMAND INSTRUCTION READY TO UNIT signal and the subsequent use of this signal to strobe the function bits, etc., of the second word of the CEU instruction into the proper registers, latches etc. The time between the generation of the COMMAND INSTRUCTION READY TO UNIT signal and the changing of the UNIT COMMAND ACCEPTED level from +V to 0 volts is entirely a function of the particular unit and its electronic and circuit configuration. When the CDA flip-flop is reset, the reset output goes to +V and provides a 0 volt level from the OR gate directly above the flip-flop. This OR gate then produces a UNIT DATA ACCEPTED signal back to the mainframe to reset the mainframe control circuits to their initial configuration.

2.5.2 Test External Unit

2.5.2.1 Function of the Test External Unit Instruction

The Test External Unit instruction (TEU) is used to test an external unit for some particular function. The instruction may test for: (1) unit busy; (2) parity error; (3) bottom of form (for printer); (4) unit inoperative; (5) not busy; (6) end-of-file, end-of-record, load point, write ring in, etc., (8) specific functions for specific equipment as designated by the designer of that equipment.

The TEU is a two-word instruction which is stored in two sequential memory locations. The instruction operates only in the Skip Mode and is always presumed "ready" for a test. If the result of the test is true, then the program register in the mainframe is advanced twice following the unloading of the second word of the TEU instruction. If the tested-for condition is absent then the program register is advanced to the next sequential instruction following the second word of the TEU instruction. This instruction does NOT contain the wait flag option.

The TEU instruction may be used in either the Immediate or the Address Mode. The programmed mode is determined by the presence of a "1" or "0" in the Indirect flag position, bit 5, in the first word of the TEU instruction. If the Indirect flag position contains a "0" (Immediate Mode), the second word from memory will contain the operand. The Address Mode utilizes the indirect address word format to contain the direct address. This format provides the direct address plus an Indirect flag bit and an Index flag bit. The address portion contains the 14 least significant bits of the memory address. The second most signficant bit (bit 1) will be a 0 if bit 6 of the first word of the TEU instruction, the MAP bit is a "0". If the MAP bit contains a "1", the most significant bit of the program register becomes the most significant bit of the address. Inasmuch as the most significant bit of the program register may be either a "0" or a "1", the inclusion of the MAP bit modifier allows the programmer to operate anywhere in the entire 32K memory. The indirect address format is used because it allows the utilization of the Indirect flag for chaining and the Index flag for accessing a table of data words.

The final word obtained from memory at the end of an indirect chain is identical in format to the second word called from memory in the Immediate Mode. The operand contains 16 bits which are coded to specify the function to be tested.

2.5.2.2 TEU Peripheral - Computer Timing Relationships

The basic computer timing, described in detail in paragraph 1.7, consists of a continuous series of 1.75 microseconds cycles. This cycle time is the time required for the memory to perform a complete read/write cycle, The cycle during which the instruction is unloaded from memory is referred to as the I (instruction) cycle; the memory cycles during which data is unloaded from or loaded into memory are described as E (Execution) cycles.

Each TEU instruction consists of one or more I cycles and at least two E cycles. The first I cycle is a cycle during which the first word of the instruction is unloaded from memory. During this I cycle and the second I cycle, the unit is prepared for the subsequent function bits used in the test instruction. Additional I cycles are also required if the instruction is used in the Address Mode requiring more than one indirect cycle to obtain the second word from memory. When the second word containing the data (function code bits) is unloaded from memory at the end of the indirect chain or immediately if the Immediate Mode is used, the computer is allowed to go into an execution cycle referred to as the E_1 cycle.

2.5.2.3 TEU Execution Sequence

The basic instruction, if used in the Immediate Mode, requires three complete memory cycle times as shown in the timing diagram (See Figure 2.5-2). Note that during the first I cycle the command instruction is shown as being available from the beginning of T1 time of the first I cycle continuously until the beginning of T1 time of the I cycle of the next subsequent instruction. During this entire period the TEU instruction remains in the Instruction Register in the mainframe and thus provides the TEU enables to both the mainframe and, through the mainframe circuits, to the selected unit. The Instruction Register also holds the unit number for this period.

At T4 time of the first I cycle the computer circuits produce an INSTRUCTION SYNC level which is gated to the selected unit circuits by the unit code. If the unit is connected, the INSTRUCTION SYNC signal gates a T2 pulse or a T8 pulse to set a latch in the peripheral unit which provides UNIT SYNC RETURN signal back to the mainframe. The INSTRUCTION SYNC signal which initiated this action in the unit also sets a latch in the mainframe which holds the mainframe to the I cycle. When the UNIT SYNC RETURN signal comes back to the mainframe it resets this latch and allows the computer to advance into the E_1 cycle during which the second word is unloaded from memory.

The computer provides a Computer Data Here signal to the unit coincident with the time that the second word of the instruction is available on the I/O Bus. The Computer Data Here signal causes the unit to test the function specified by the second word, and raise the Unit Test Return signal if the test was true. The unit raises the Unit Data Accepted signal to signify to the computer that the unit test line is ready to be tested. If the Unit Test Return is true, the program register

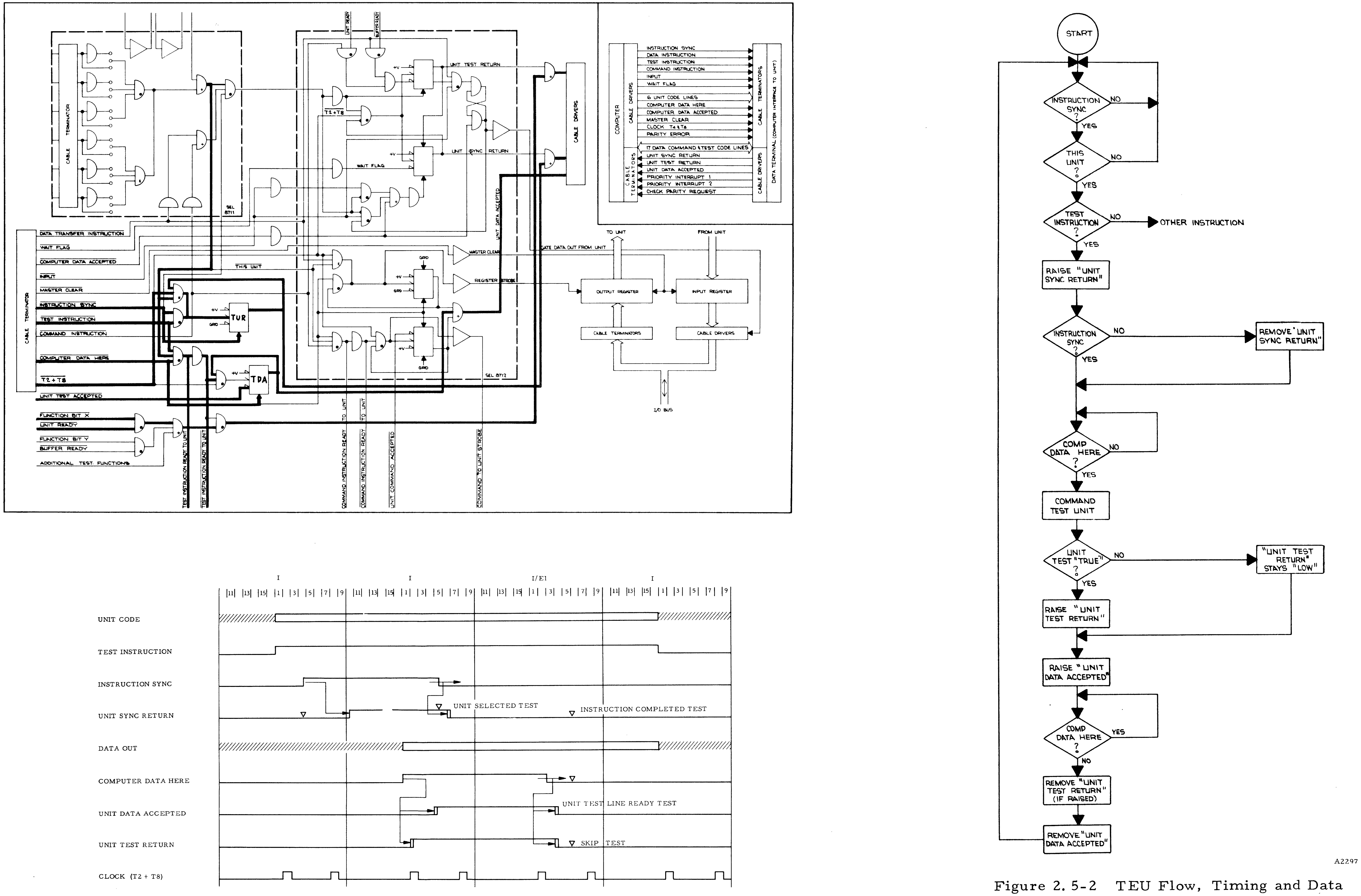


Figure 2. 5-2 TEU Flow, Timing and Data Terminal Diagrams

will be advanced twice so as to skip the next subsequent instruction. If the Unit Test Return signal is not received by the computer, the next sequential instruction will be executed. The computer removes the Computer Data Here signal after the Unit Test Return signal has been tested, which causes the unit to remove both Unit Test Return and Unit Data Accepted signals.

The input sequence of the TEU instruction as it pertains to the unit is shown in the TEU flow diagram (Figure 2.5-2). The first signal produced by the mainframe to the unit is the INSTRUCTION SYNC level. When that signal occurs, and the unit is selected by the unit code, it will produce a UNIT SYNC RETURN level.

The next signal produced by the mainframe is the COMPUTER DATA HERE signal. When that signal arrives it is gated to test the various function bits as compared to the functions which they specify. If the tested-for condition is true, then the UNIT TEST RETURN signal is produced and in turn produces the UNIT DATA ACCEPTED signal. The COMPUTER DATA HERE signal also removes the UNIT TEST RETURN and the UNIT DATA ACCEPTED levels.

2.5.2.4 Detailed Operation of the SEL Data Terminal for TEU Instructions

The presence of the TEU instruction in the Instruction Register causes the TEST INSTRUCTION signal to go to 0 volts. This 0 volt level partially enables two AND circuits. One of these is composed of a 2-input AND gate connected with a 3-input expander and providing the trigger input to the UTR flip-flop. This AND gate is completely enabled when the UTR flip-flop is reset and the unit is selected and the INSTRUCTION SYNC signal goes to + 0 volts. The occurrence of the first computer-produced timing signal, the INSTRUCTION SYNC level, gates the next T2 or T8 pulse through the AND circuit to set the UTR flip-flop. The resulting +V level from the set output of this flip-flop inhibits the trigger input AND gate and also provides a UNIT SYNC RETURN signal back to the mainframe.

The next signal produced by the mainframe is the COMPUTER DATA HERE which goes to 0 volts as when the second word of the TEU instruction becomes available on the I/O Bus. This level AND's with the TEST INSTRUCTION 0 volt level and THIS UNIT 0 volt level produced by the unit select card to provide a TEST INSTRUCTION READY TO UNIT and through an inverter a TEST INSTRUCTION READY TO UNIT signal. The raw COMPUTER DATA HERE signal also removes the DC reset signal from the TDA flip-flop. The TEST INSTRUCTION READY TO UNIT signal gates the next T2 or T8 pulse as a trigger strobe to the TDA flip-flop. The TEST INSTRUCTION READY TO UNIT also partially enables another AND gate which will be fully enabled if the function bit and the tested for condition are found to coincide. If this condition is found to be true then the two inputs to the AND gate go to 0 volts and the output is of the AND gate is applied through an OR gate as a UNIT TEST RETURN level. Sometime following this signal the UNIT TEST ACCEPTED signal goes to 0 volts to allow the next T2 or T8 pulse to trigger the TDA flip-flop and set it. The resulting +V output of the TDA flip-flop is applied through the NOR gate as a UNIT DATA ACCEPTED signal which goes back to the mainframe and returns the COMPUTER DATA HERE signal to +V. When this occurs the TDA flip-flop is DC reset and the unit and the Data Terminal as well as the mainframe returns to their initial configurations.

2.5.3 Output Data Transfer Instructions (MOP and AOP)

There are two output data transfer instructions available on the SEL 810A computer. These are the Accumulator Out to Peripheral (AOP) and the Memory Out to Peripheral (MOP) instructions. The AOP instruction is a single word instruction which transfer data from the A accumulator to the selected unit. The MOP instruction transfers data from a specified memory cell to the peripheral unit. The MOP and the AOP instructions require a minimum of three memory cycle times to be executed. Both instructions may operate in either the Skip or the Wait Mode of operation; but only the MOP instruction may be operated in the Immediate or Address Mode.

2.5.3.1 Accumulator Out to Peripheral Instruction

In addition to the function code, the AOP instruction word contains a wait flag in bit position 9 and the unit number to which the data is to be transferred in bit positions 10 through 15. If the wait flag position contains a "1" this causes the computer to operate in Wait Mode. In this mode the unit will not answer with unit sync return until it is ready to accept the data, therefore causing the computer to wait. The presence of the "0" in the wait flag position will cause the computer to operate in a Skip Mode. The Skip Mode causes the computer to access the next sequential instruction following the AOP instruction if the unit is not ready; but to skip the next sequential instruction and obtain the subsequent instruction if the unit is ready. When the computer is in a wait cycle, it cannot be interrupted by any priority interrupt except for the optional power fail safe/restore or stall alarm interrupts. If however the skip mode is used, the computer may be programmed through use of BRU instruction to cycle until the unit is ready before proceeding. This allows the servicing of priority interrupts that might occur between the AOP instruction and the BRU instruction.

2.5.3.2 Memory Out to Peripheral Instruction

The MOP is a two-word instruction which is stored in two sequential memory locations. The instruction may operate in either the Wait or Skip Modes and either Immediate or Address Modes of operation. The first word of this instruction sets the modes of operation by flags contained in bit positions 5 and 9. Bit 9 contains the wait flag which is set to a "1" to cause the computer to operate in the Wait Mode. This mode will cause the unit not to answer Unit Sync Return until it is ready to receive the data word, causing the computer to wait. The presence of a "0" in the wait flag position of the first word will cause the computer to operate in the Skip Mode. The Skip Mode causes the computer toe xit to the next sequential instruction following the second word of the MOP instruction if the unit is not ready; but to skip the next sequential instruction and obtain the subsequent instruction if the unit is ready. When the computer is in a wait cycle it cannot be interrupted by any priority interrupt except for the optional power fail safe/restore or stall alarm interrupts. If the Skip Mode is used, the computer may be programmed through the use of a BRU instruction to cycle until the unit is ready before proceeding. This allows servicing of priority interrupts that might occur between the MOP instruction and the BRU instruction.

The Indirect flag located in bit position 5 of the first word of the MOP instruction defines whether the instruction will operate in the Immediate Mode or the Address Mode. If the Indirect flag position contains a "0" the second word unloaded from memory becomes the operand. The Address Mode (bit 5 contains a "1") utilizes the indirect address word format to contain the direct address. This format provides a direct address plus indirect and index flags. The address portion contains the 14 least significant bits of the memory address. The 15th bit will be a zero if bit 6 of the first word, the MAP bit, is a "0". If the MAP bit contains a "1", the most significant bit of the program register becomes the most significant bit of the direct address. Inasmuch as the most significant bit of the program register may be either a 0 or 1, inclusion of the MAP bit modifier allows the programmer to operate anywhere in the entire 32K of memory. The indirect address format is used because it allows the utilization of the Indirect flag for chaining and the Index flag for accessing tables of words.

The word obtained from memory at the end of an indirect chain is identical in format to the second word called from memory inthe Immediate Mode. This second word contains the data word that is to be transferred to the peripheral unit.

2.5.3.3 AOP Peripheral - Computer Timing Relationships

The basic computer timing, described in detail in paragraph 1.7, consits of a continuous series of 1.75 microseconds cycles. This cycle time is a time required for a memory to perform a complete read/write cycle. The cycle during which the instruction is unloaded from memory is referred to as the I (instruction) cycle.

Each AOP instruction consists of at least two I cycles. The first I cycle is a cycle during which the AOP instruction word is unloaded from memory. The second I cycle is the time during which the unit is tested to see whether it is ready to accept a data transfer from the computer. The second I cycle may be repeated indefinitely if the instruction is used in the Wait Mode. When the unit has responded that it is ready for a transfer of data from the computer then the computer is allowed to go into E1 cycle during which the data from the A Accumulator is transferred to the unit.

2.5.3.4 MOP Peripheral - Computer Timing Relationships

Each MOP instruction consists of at least two I cycles and one E cycle. The first I cycle is a cycle during which the first word of the instruction is unloaded from memory. The second I cycle is a time which the unit is tested to see whether it is ready to accept a data transfer from the computer. The second I cycle may be repeated indifinitely if the instruction is operated in the Wait Mode. Additional I cycles are also required if the instruction is used in the Address Mode requiring more than one indirect cycles. When the second word containing the data is unloaded from memory at the end of an indirect chain or immediately if the Immediate Mode is used, and the unit has responded that it is ready for a transfer of data from the computer, then the computer is allowed to go into an execution cycle referred to as E_1 cycle.

2.5.3.5 AOP/MOP Execution Sequence

The execution sequence, insofar as a "ready" peripheral unit is concerned, is identical for both the AOP and MOP instructions. The basic instructions used in the Skip and Immediate Address Modes requires a minimum of 3 complete memory cycle times as shown in the timing diagram. Note that during the first I cycle the DATA TRANSFER INSTRUCTION enable is shown as being available from the beginning of T1 time of the first I cycle continuously until the beginning of T1 time of the I cycle of the next subsequent instruction. During this entire period the MOP or AOP instruction remains in the Instruction Register in the mainframe and thus provides the output transfer enables to both mainframe and, through mainframe circuits, to the selected unit. The Instruction Register also holds the wait flag and the unit number for this period.

At T4 time of the first I cycle the computer circuits produce an INSTRUCTION SYNC level which is gated through the selected unit circuits by the unit code. If the unit is ready, the INSTRUCTION SYNC signal gates a T2 or a T8 pulse to set a latch in the peripheral unit which provides simultaneously a UNIT TEST RETURN and UNIT SYNC RETURN back to the mainframe. The INSTRUCTION SYNC signal which initiated this action in the unit also set a latch in the mainframe which holds the mainframe to the I cycle. When the UNIT SYNC RETURN signal comes back to the mainframe, it resets this latch and allows the computer to advance into the E_1 cycle during which the second word is unloaded from memory (or transferred from the A Accumulator). The UNIT TEST RETURN signal is used to perform the skip test. If the UNIT TEST RETURN signal is received, the program register will be advanced twice so as to skip the next subsequent instruction. If the UNIT TEST RETURN signal does not come back, then the program register will be advanced once and call the next sequential instruction. The UNIT TEST RETURN signal also provides a COMPUTER DATA HERE signal from the computer to the unit coincident with the time that the data word provided by the instruction is available on the I/O Bus. The COMPUTER DATA HERE signal sets a latch which provides a signal back to the computer termed UNIT DATA ACCEPTED. The UNIT DATA ACCEPTED signal is produced at the same time that an OUTPUT REGISTER

STROBE signal is generated in the peripheral unit. The OUTPUT REGISTER STROBE loads the data word provided by the memory or the A Accumulator into the Output Register of the unit.

Output transfer is controlled by AOP and MOP instructions. The transfer sequence is illustrated in Figure 2.5-3. The first signal produced by the mainframe to the unit is the INSTRUCTION SYNC level. When that signal occurs, if the Output Buffer is ready for a new word to be strobed in and, if the unit is selected by the unit code, the unit will produce a UNIT TEST RETURN signal. Note that if the buffer is not ready for a transfer to be performed when the INSTRUCTION SYNC level is generated by the mainframe, a test is made to determine if the wait flag is present in the original instruction. If the wait flag is present, the unit will continually test for the BUFFER READY condition and the mainframe will remain in the I cycle. If the wait flag is not present, then the UNIT TEST RETURN signal remains low but the UNIT SYNC RETURN signal is returned.

The removal of the INSTRUCTION SYNC level, also provides for the removing of the UNIT SYNC RETURN signal. If the buffer is ready, both the UNIT TEST RETURN and the UNIT SYNC RETURN levels are produced. Again, the removal of the INSTRUCTION SYNC level provides the means of resetting or removing the UNIT TEST RETURN and UNIT SYNC RETURN signals. The subsequent occurrence of the COMPUTER DATA HERE signal will provide the signal to load the Output Register and produce the UNIT DATA ACCEPTED signal. The COMPUTER DATA HERE signal then remains until the computer received the UDA signal. The removal of computer data here resets the unit data accepted flip-flop.

2.5.3.6 Detailed Operation of the SEL Data Terminal for AOP/MOP Instructions

Refer to Figure 2.5-3.

The presense of either an AOP or a MOP instruction in the instruction register will cause the $\overline{\text{DATA TRANSFER INSTRUCTION}}$ level to go to 0 volts. This 0 volt level is applied through an inverter and an OR gate to enable the output AND gate on the unit select functional card. The $\overline{\text{DATA TRANSFER INSTRUCTION}}$ 0 volt level also enables an AND gate shown at the top of SEL 8712 card drawing. The second input to this AND gate will be at 0 volts when the buffer is ready. The output of this AND gate is applied through an OR gate to disable the reset steering input of the UTR flip-flop when the buffer is ready. The disabling of the reset steering input effectively enables the set steering input. The $\overline{\text{DATA TRANSFER INSTRUCTION}}$ 0 volt level also partially enables a 4-input AND gate which provides the triggering input to the UDA flip-flop. The other enabling inputs to the 4-input AND gate are $\overline{\text{THIS UNIT}}$ 0 volts level and the normally 0 volt set output of the UDA flip-flop itself.

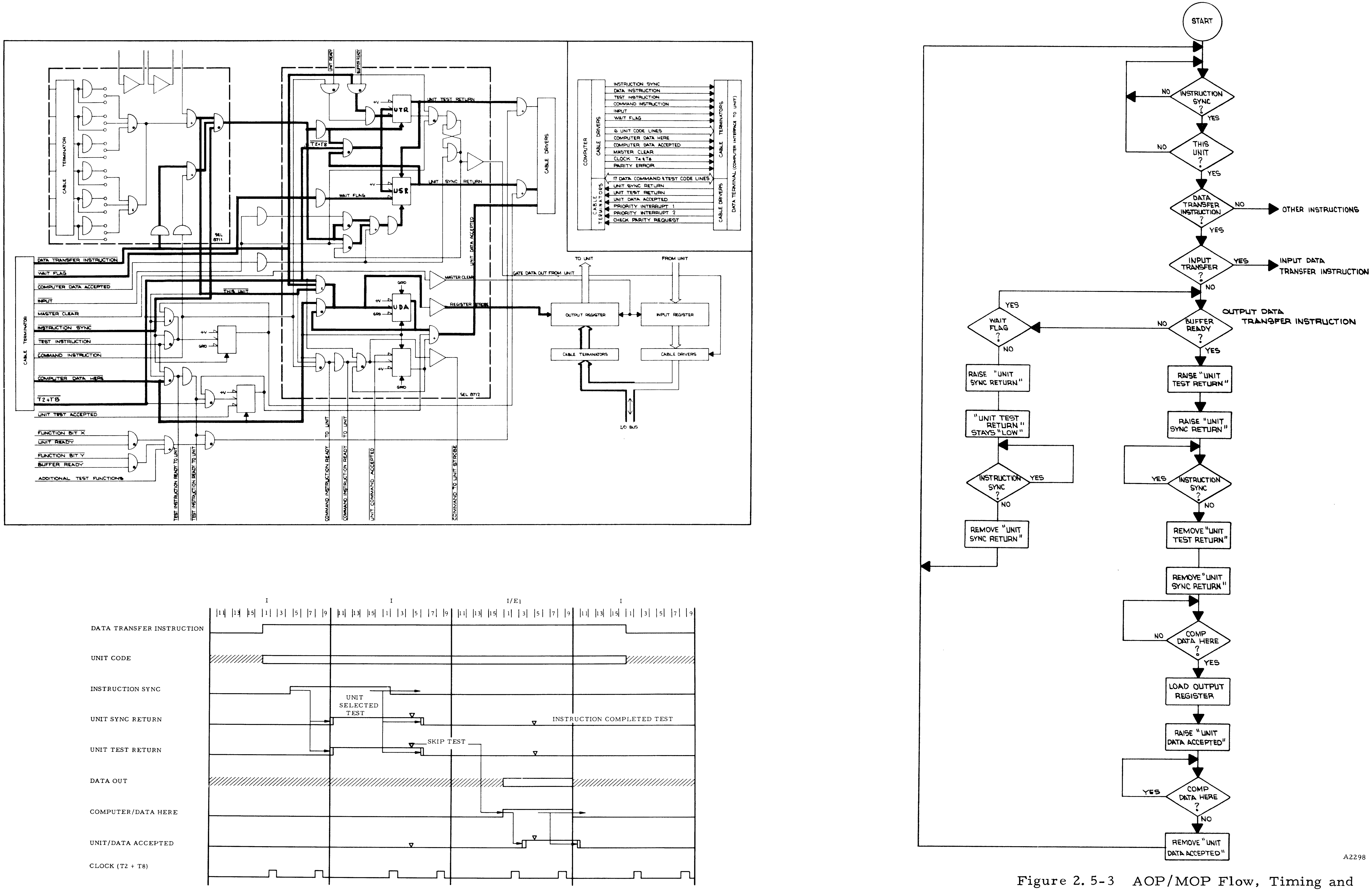


Figure 2.5-3 AOP/MOP Flow, Timing and Data Terminal Diagrams

The first signal produced by the computer to the unit is the INSTRUCTION SYNC signal which goes to 0 volts at the output of the cable terminator. This 0 volt level completes the enables to the unit select AND gate. The output of this circuit removes the DC reset to the UTR flip-flop through an inverter and enables an AND gate which connects to the trigger inputs of both the UTR and USR flip-flops. This gate allows T2 and T8 timing pulses to query the trigger input of both flip-flops. If the transfer instruction contains a wait flag the reset input of the USR flip-flop will be at +V. The set steering input is permanently held at +V so the T2 or T8 trigger strobes have no affect. The UTR flip-flop will be set only when the BUFFER READY level goes to 0 volts to effectively enable the set steering input of that flip-flop. When the flip-flop is set the UNIT TEST RETURN signal will be produced and sent to the computer. At the same time, the signal sets the USR flip-flop to provide the UNIT SYNC RETURN signal. In the Skip Mode the USR flip-flop will be set by the first T2 or T8 timing pulse to be gated by the INSTRUCTION SYNC and DATA TRANSFER INSTRUCTION enables. The UNIT TEST RETURN line will reflect the condition of buffer ready or not ready.

The next timing signal to be gated to the unit by the mainframe is the COMPUTER DATA HERE signal if the unit is ready. This 0 volt level completes the enable on the 4-input AND gate controlling the trigger input of the UDA flip-flop. The resulting output to the AND gate will set the UDA flip-flop to produce a UNIT DATA ACCEPTED signal and will also provide a strobe to load the output data word into the Output Register. The return of the COMPUTER DATA HERE signal to +V at the end of the AOP or MOP instruction DC resets the UDA flip-flop to remove the UNIT DATA ACCEPTED signal.

2.5.4 Input Data Transfer Instructions (MIP and AIP)

There are two input data transfer instructions available on the SEL 810A computer. These are the Accumulator in from Peripheral (AIP) and the Memory in from Peripheral (MIP) instructions. The AIP instruction is a single word instruction which transfers data from the selected unit to the A Accumulator. The MIP instruction transfers data from the selected unit to a specified memory cell. The MIP and AIP instructions require a minimum of three memory cycle times to be executed. Both instructions may operate in either the Skip or the Wait modes, but only the MIP instruction may be operated with the Immediate or Address mode.

2.5.4.1 Accumulator In to Peripheral Instruction

In addition to the function code, the AIP instruction word contains a wait flag in bit position 9 and the unit number to which the data is to be transferred in bit positions 10 through 15. If the wait flag position contains a "1" this causes the computer to operate in Wait Mode. In this mode the unit will not answer unit sync return until it is ready to transfer data. The presence of the "0" in the wait flag position will cause the computer to operate in a Skip Mode. The Skip Mode causes the computer to access the next sequential instruction following the AIP instruction if the unit is not ready; but to skip the next sequential instruction and obtain the subsequent instruction if the unit is ready. When the computer is in a wait cycle, it cannot be interrupted by any priority interrupt except for theoptional power fail safe/restore or stall alarm interrupts. If however, the Skip Mode is used, the computer may be programmed through use of BRU instruction to cycle until the unit is ready before proceeding. This allows the servicing of priority interrupts that might occur between the AIP instruction and the BRU instruction.

The AIP instruction format also includes a "Merge" flag in bit position 4 for programming convenience. If this flag is a "0", the input data word is loaded into all 16-bit positions of the A Accumulator. If this word is an 8-bit character, the character is loaded into bit position 8 through 15 and zeros are loaded into bit positions 0 through 7. If the "Merge" flag is a "1", only the eight least significant bits (a character) are loaded into bit positions 8 through 15 and the contents of bit positions 0 through 7 (a prior input character) are undisturbed. NOTE: If bits 8 through 15 contain data when the "Merge" operation occurs, the "new" character will be "ORed" with the previous data.

2.5.4.2 Memory In From Peripheral Instruction

The MIP instruction is a two-word instruction which is stored in two sequential memory locations. The instruction may operate in either the Wait or Skip Modes and either Immediate or Address Modes of operation. The first word of this instruction sets the modes of operation by flags contained in bit positions 5 and 9. Bit 9 contains the wait flag which is set to a "1" to cause the computer to operate in the Wait Mode. This mode will cause the unit not to answer unit sync return until it is ready to transfer data. The presence of a "0" in the wait flag position of the first word will cause the computer to operate in the Skip Mode. The Skip Mode causes the computer to exit to the next sequential instruction following the second word of the MOP instruction if the unit is not ready; but to skip the next sequential instruction and obtain the subsequent instruction if the unit is ready. When the computer is in a wait cycle it cannot be interrupted by any priority interrupt except for the otpional power fail safe/restore or stall alarm interrupts. If the Skip Mode is used, the computer may be programmed through the use of a BRU instruction to cycle until the unit is ready before proceeding. This allows servicing of priority interrupts that might occur between the MIP instruction and the BRU instruction.

The Indirect flag located in bit position 5 of the first word of the MIP instruction defines whether the instruction will operate in the Immediate Mode orthe Address Mode. If the Indirect flag position contains a "0", the second word becomes the location in which the data word is to be loaded. The Address Mode (the Indirect flag is a "1") utilizes the indirect address word format to contain the direct address. This format provides a direct address plus indirect and index flags. The address portion contains the 14 least significant bits of the memory address. The 15th bit will be a zero if bit 6 of the first word, the MAP bit, is a "0". If the MAP bit contains a "1", the most significant bit of the program register becomes the most significant bit of the direct address. Inasmuch as the most significant bit of the program register may be either a 0 or 1, inclusion of the MAP bit modifier allows the programmer to operate anywhere in the entire 32K of memory. The indirect address format is used because it allows the utilization of the Indirect flag for chaining and the Index flag for accessing tables of words.

2.5.4.3 AIP Peripheral - Computer Timing Relationships

The basic computer timing, described in detail in paragraph 1.7, consists of a continuous series of 1.75 microseconds cycles. This cycle time is a time required for a memory to perform a complete read/write cycle. The cycle during which the instruction is unloaded from memory is referred to as the I (instruction) cycle.

Each AIP instruction consists of at least two I cycles and one E cycle. The first I cycle is a cycle during which the AIP instruction word is unloaded from memory. The second I cycle is the time during which the unit is tested to see whether it is ready to provide a data transfer to the computer. The second I cycle may be repeated indefinitely if the instruction is used in the Wait Mode. When the unit has responded that it is ready to provide a transfer of data to the computer then the computer is allowed to go into the E_1 cycle during which the data from the unit is transferred to the A Accumulator.

2.5.4.4 MIP Peripheral - Computer Timing Relationships

Each MIP instruction consists of at least two I cycles and one E cycle. The first I cycle is a cycle during which the first word of the instruction is unloaded from memory. The second I cycle is a time which the unit is tested to see whether it is ready to provide a data transfer to the computer. The second I cycle may be repeated indefinitely if the instruction is operated in the Wait Mode. Additional I cycles are also required if the instruction is used in the Address Mode requiring more than one indirect cycles. When the address of the word to receive the data is read from memory at the end of an indirect chain or immediately if the Immediate Mode is used, and the unit has responded that it is ready to transfer data to the computer, then the computer is allowed to go into an execution cycle referred to as E_1 cycle.

2.5.4.5 AIP/MIP Execution Sequence

Refer to Figure 2.5-4

The execution sequence, insofar as a "ready" peripheral unit is concerned, is identical for both the AIP and MIP instructions. The basic instructions used in the Skip and Immediate Modes requires a minimum of 3 complete memory cycle times as shown in the timing diagram. Note that during the first I cycle the DATA TRANSFER INSTRUCTION enable is shown as being available from the beginning of T1 time of the first I cycle continuously until the beginning of T1 time of the I cycle of the next subsequent instruction. During this entire period the MIP

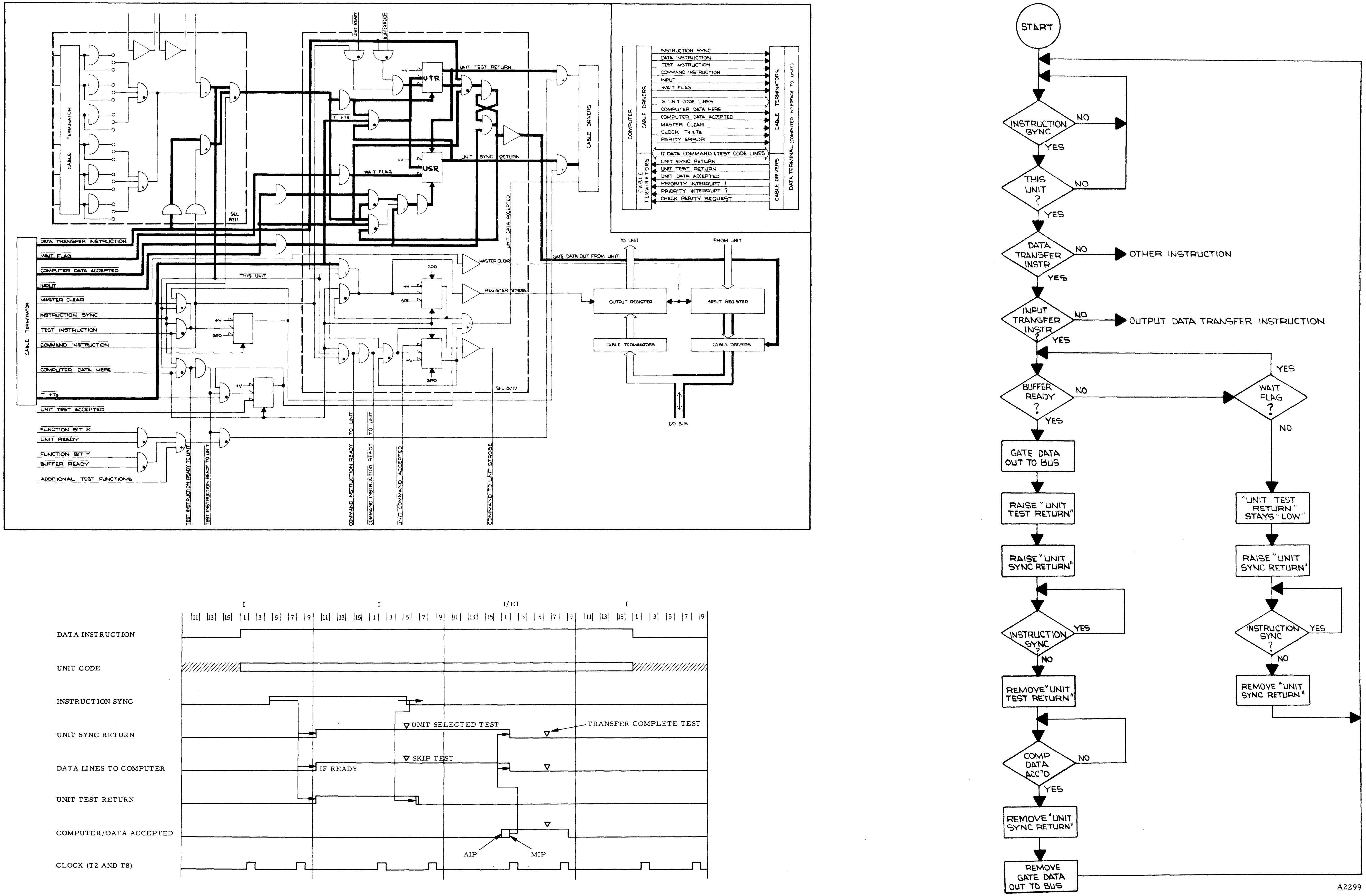


Figure 2.5-4 AIP/MIP Flow, Timing and Data Terminal Diagrams

or AIP instruction remains in the instruction register in the mainframe and thus provides the output transfer enables to both mainframe and, through mainframe circuits, to the selected unit. The Instruction Register also holds the wait flag and the unit number for this period.

At T4 time of the first I cycle the computer circuits produce an INSTRUCTION SYNC level which is gated through the selected unit circuits by the unit code. If the unit is ready, the INSTRUCTION SYNC signal gates a T2 or T8 pulse to set a latch in the peripheral unit which provides simultaneously a UNIT TEST RETURN and a UNIT SYNC RETURN back to the mainframe. The INSTRUCTION SYNC signal which initiated this action in the unit also sets a latch in the mainframe which holds the mainframe to the I cycle. When the UNIT SYNC RETURN signal comes back to the mainframe it resets this latch and allows the computer to advance into the E_1 cycle during which the second word is loaded into memory (or transferred to the A Accumulator). The UNIT TEST RETURN signal is used to perform the skip test. If the UNIT TEST RETURN signal is received, the program register will be advanced twice so as to skip the next subsequent instruction. If the UNIT TEST RETURN signal does not come back, then the program register will be advanced once and call the next sequential instruction. The UNIT TEST RETURN signal also provides a GATE DATA OUT TO COMPUTER signal to the Input Register in the unit. The computer then provides a COMPUTER DATA ACCEPTED signal that resets the GATE DATA OUT TO COMPUTER and UNIT SYNC RETURN levels.

Input transfer is controlled by AIP and MIP instructions. The transfer sequence is illustrated in Figure 2.5-4. The first signal produced by the mainframe to the unit is the INSTRUCTION SYNC level. When that signal occurs and the unit is ready for a new word to be strobed in, and if the unit is selected by the unit code, the unit will produce a UNIT TEST RETURN signal. Note that if the buffer is not ready for a transfer to be performed when the INSTRUCTION SYNC level is generated by the mainframe a test is made to determine if the wait flag is present in the original instruction. If the wait flag is present, the unit will continually test for the BUFFER READY condition and the mainframe will remain in the I cycle. If the wait flag is not present, then the UNIT TEST RETURN signal remains low but the UNIT SYNC RETURN signal is returned.

The removal of the INSTRUCTION SYNC level, provides for the removing of the UNIT SYNC RETURN. If the buffer is ready, both the UNIT TEST RETURN and the UNIT SYNC RETURN levels are produced. Again, the removal of the INSTRUCTION SYNC level provides the means of resetting or removing the UNIT TEST RETURN and UNIT SYNC RETURN signals. The subsequent occurrence of the COMPUTER DATA ACCEPTED signal will provide the means to gate the input word to the computer. The COMPUTER DATA ACCEPTED signal removes the UNIT SYNC RETURN and the GATE DATA OUT TO BUS signals.

2.5.4.6 Detailed Operation of the SEL Data Terminal for AIP/MIP Instructions

The presence of either a AIP or an MIP instruction in the instruction register will cause the DATA TRANSFER INSTRUCTION and INPUT levels to go to 0 volts. The DATA TRANSFER INSTRUCTION 0 volt level is applied through an inverter and an OR gate to enable the output AND gate on the unit select functional card. The DATA TRANSFER INSTRUCTION 0 volt level also enables an AND gate shown at the top of SEL 8712 card drawing. The second input to this AND gate will be at 0 volts when the buffer is ready. The output of this AND gate is applied through an OR gate to disable the reset steering input of the UTR flip-flop when the buffer is ready. The disabling of the reset steering input effectively enables the set steering input.

The INPUT 0 volt level enables an AND gate on the output of the UTR flip-flop to set a NOR latch when the UTR flip-flop is set. The latch output provides the GATE DATA OUT TO COMPUTER signal to the Input Register. The first signal produced by the computer to the unit is the INSTRUCTION SYNC signal which goes to 0 volts. This 0 volt level completes the enables to the unit select AND gate The output of this circuit removes the DC reset to the UTR flip-flop through an inverter and enables an AND gate which connects to the trigger inputs of both the UTR and USR flip-flops. This gate allows T2 and T8 timing pulses to query the trigger input of both flip-flops. If the transfer instruction contains a wait flag,the reset input of the USR flip-flop will be at +V. The set steering input is permanently held at +V so the T2 or T8 trigger strobes have no affect. The UTR flip-flop will be set only when the BUFFER READY level goes to 0 volts to effectively enable the set steering input of that flip-flop. When the flip-flop is set the UNIT TEST RETURN signal will be produced and sent to the computer. At the same time the signal sets the USR flip-flop to provide the UNIT SYNC RETURN signal. In the Skip Mode the USR flip-flop will be set by the first T2 or T8 timing pulse to be gated by the INSTRUCTION SYNC and DATA TRANSFER INSTRUCTION enables. The UNIT TEST RETURN line will reflect the condition of buffer ready or not ready.

The next timing signal to be gated to the unit by the mainframe is the COMPUTER DATA ACCEPTED signal which goes to 0 volt if the unit was ready. This 0 volt level is inverted to reset the USR flip-flop and the GATE DATA OUT FROM UNIT NOR latch.

2.6 I/O Parity

The Input/Output parity option provides a parity check on data transferred between the computer and peripheral units. A priority interrupt is normally used in conjunction with I/O parity in order to process the parity error.

All data transferred to a peripheral unit contains an additional parity bit allowing the unit to perform a parity check; however, the standard units for the computer do not have parity check capability unless they are purchased as such. In order to utilize the I/O parity check option, the unit must be prepared to check parity on an output data transfer instruction during the time the computer data here is present. If a parity error is detected during this interval, a priority interrupt in the unit is set.

The computer uses the memory parity generator to perform a parity check on input data from a peripheral unit. During an input data transfer instruction the unit must enable the parity request line to the computer. This line should be enabled during the gate data out time. The gate data out latch can be connected directly to the cable driver that provides the parity request. The unit also must contain a parity generator to provide a parity bit to the computer. During the time that the data word from the peripheral unit is being loaded into either the computer memory or the A Accumulator, a parity check is performed. If an error is recognized during an input data transfer the error line to the peripheral unit is enabled and in turn the unit can enable its priority interrupt so that the error can be processed.

Figure 2.6-1 illustrates the logic and signal lines required to have I/O parity capability. The lines which interface the peripheral unit and the computer refer to the I/O Bus which is described in detail in the Appendix.

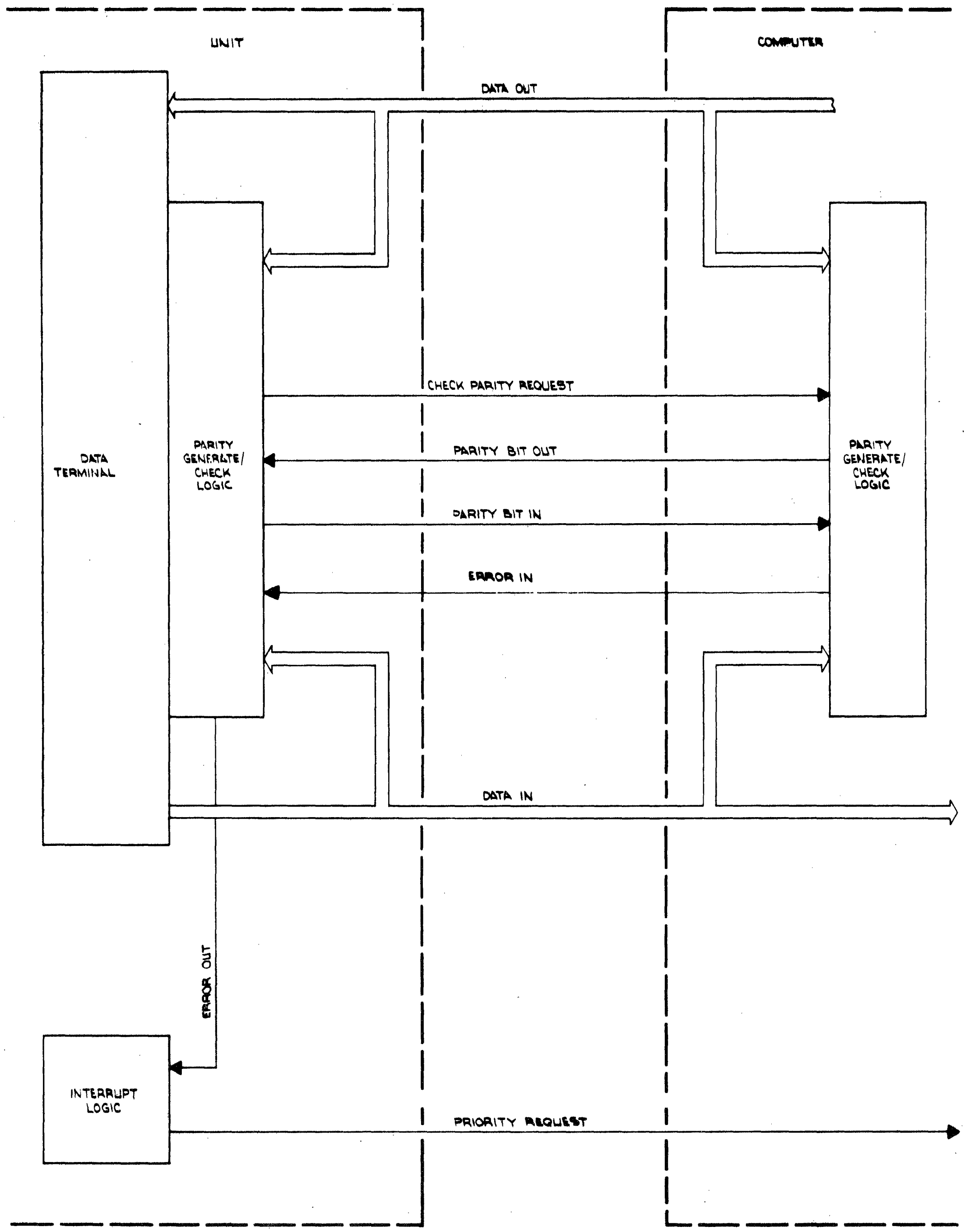


Figure 2.6-1 I/O Parity Block Diagram

A2075

3.0 Block Transfer Control Unit

3.1 General Capabilities

The SEL Block Transfer Control (BTC) is an optional computer input/output control unit which enables fully-buffered transfer of data between peripheral units and computer memory. The salient features of this unit are listed below:

Bits per Transfer	Full Computer Word
Maximum Words per Block	32, 768
Maximum Transfer Rate	572, 000 words per second
Memory Cycles Stolen per Transfer	1 (Fast Unit)
Block Transfer Re-initialization	Automatic
Maximum Number of BTC's per Computer*	8
Maximum Number of Peripheral Units per BTC	16

3.1.1 BTC Operation

A block diagram of the Block Transfer Control Unit and Block Priority Control is shown in Figure 3. 1-1. The BTC contains two binary counters plus transfer initialization and synchronization logic. One of the counters stores the current word address (CWA) and the second stores the word count (WC). CWA defines the storage location for each word transferred to/from memory and WC defines the number of words to be transferred. The initial values for CWA and WC are obtained from two fixed locations in computer memory by the BTC each time a new block transfer is initiated (see Table 4. 1 for memory location assignments). Each time a word is transferred between memory and the selected peripheral unit, CWA is incremented and WC is decremented. The block transfer is completed, and an interrupt is generated, when WC = 0. After a block transfer is completed, the BTC automatically initiates a new block transfer by obtaining a new initial set of CWA and WC values from the two dedicated memory locations. The block transfer sequence is ended by placing a terminate code in the WC word. The terminate code is a ONE in bit 0 (sign bit).

*See Section 3. 3, External Memory Access

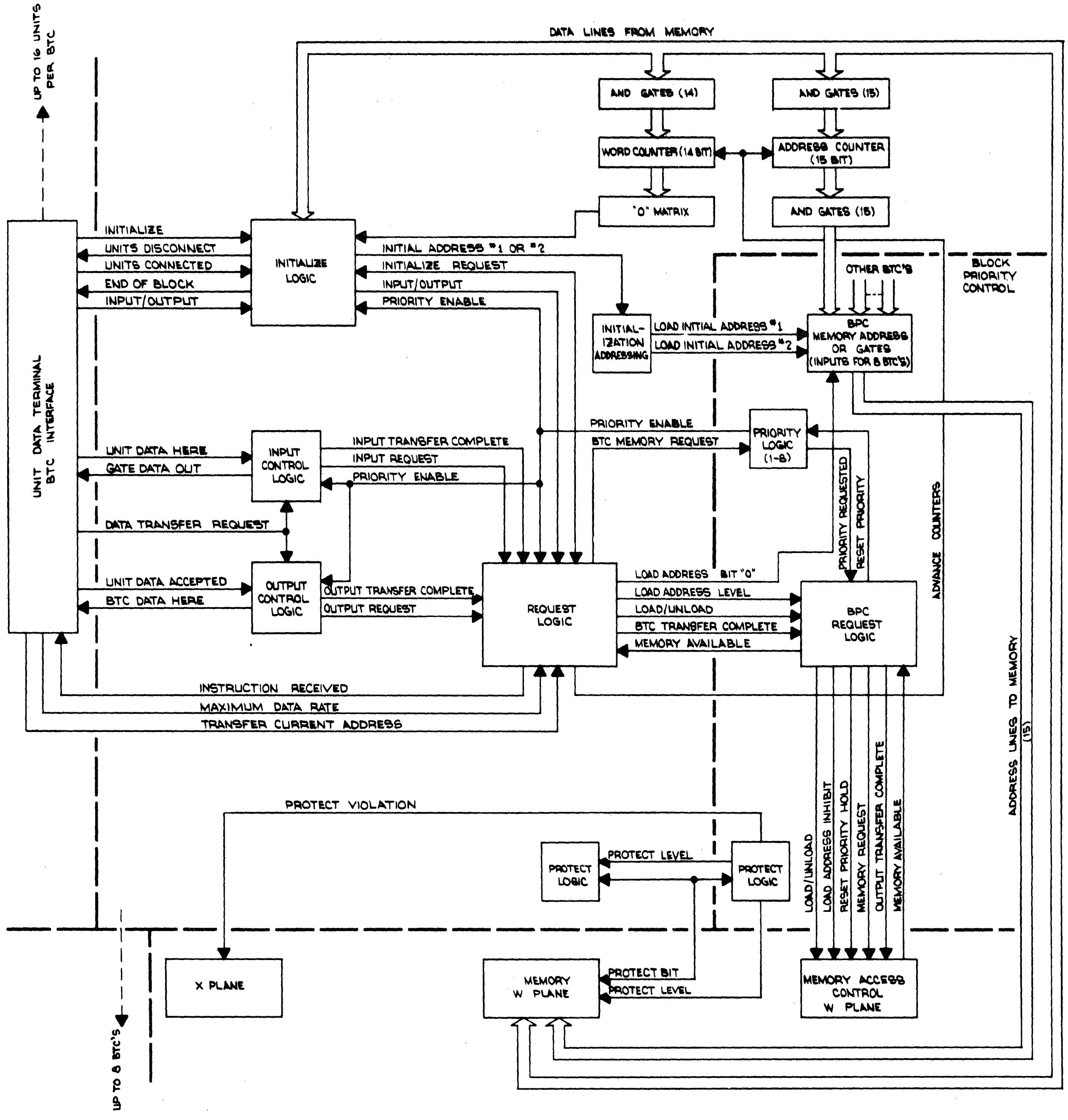


Figure 3.1-1 Block Transfer Control and Priority Block Diagram A2074

Octal Memory Location	BTC Assignment
1060	BTC 1 Current Word Address
1061	BTC 1 Block Length
1062	BTC 2 CWA
1063	BTC 2 BL
1064	BTC 3 CWA
1065	BTC 3 BL
1066	BTC 4 CWA
1067	BTC 4 BL
1070	BTC 5 CWA
1071	BTC 5 BL
1072	BTC 6 CWA
1073	BTC 6 BL
1074	BTC 7 CWA
1075	BTC 7 BL
1076	BTC 8 CWA
1077	BTC 8 BL

The CWA value may be transferred from the BTC to the dedicated memory location by the execution of a CEU command to the unit currently operating with the BTC by inserting a ONE in bit 13 of the command code. If the BTC currently has a memory cycle request in to the Block Priority Control the transfer CWA command is stored by the BTC and the CEU instruction is allowed to terminate, in order not to tie up the computer. Immediately following the memory cycle for the BTC the CWA is transferred to the dedicated memory location.

3.1.2 BTC Initialization and Data Flow

The BTC is initialized through the peripheral unit to/from which the block transfer is to be made. Figure 3.1-2 shows the data and control paths involved. The figure shows two peripheral units connected to one BTC and a third peripheral unit connected to a second BTC. These units, as previously described, may communicate with the computer through execution of any of the I/O instructions. In addition, they may transfer data under BTC control, rather than under single-word program control.

Execution of the proper Command External Unit (CEU) instruction causes the unit specified by the instruction to send an Initialize signal to the BTC to which it is cabled. In may peripheral units, this instruction also causes the unit to initiate action to produce/accept data. When the BTC receives the Initialize signal from the unit, it requests a memory cycle through the Block Priority Control (BPC). It also generates the address of the CWA memory location assigned to it. When the memory cycle is granted, the CWA value is transferred from the memory to the CWA counter in the BTC.

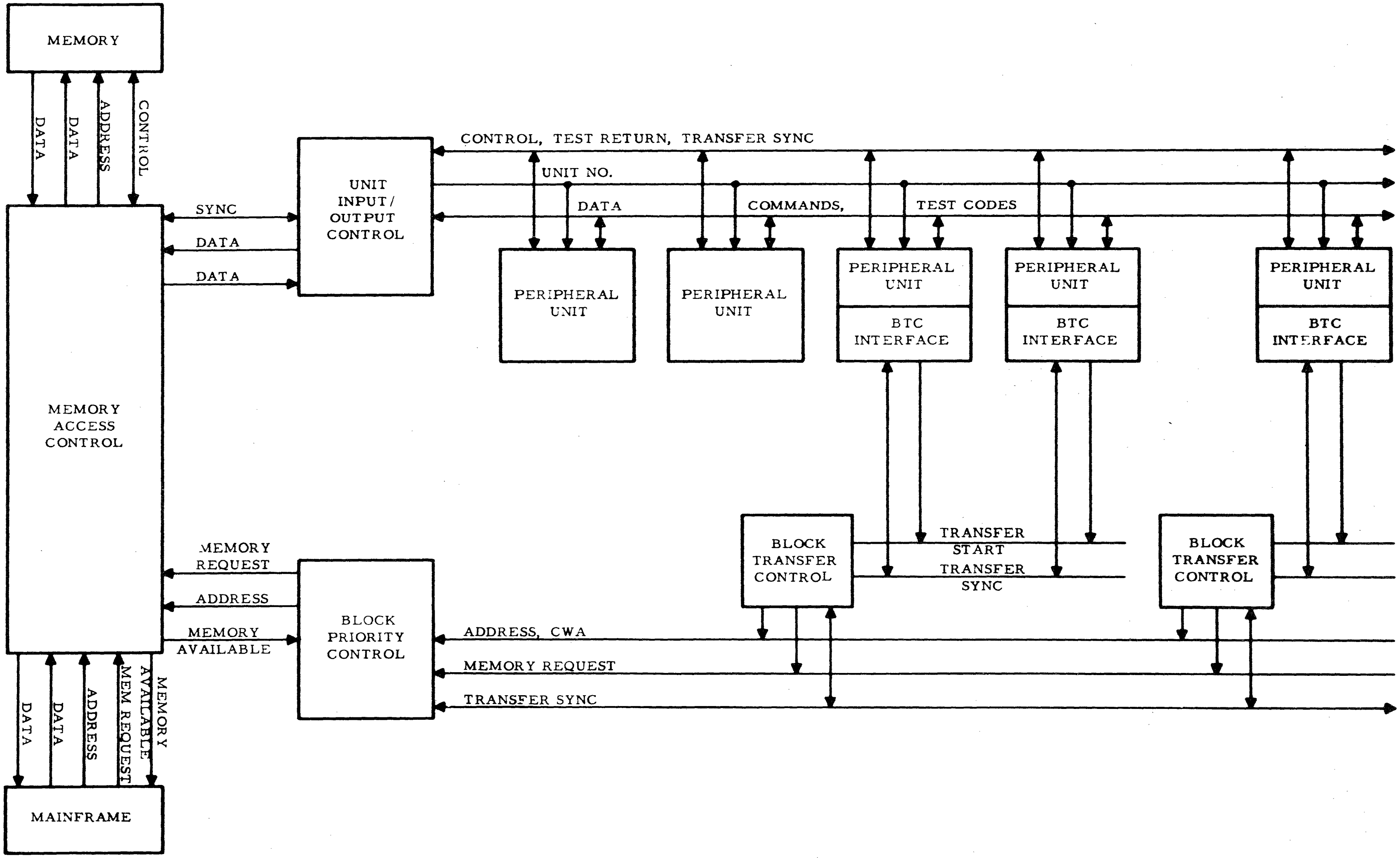


Figure 3.1-2 I/O Configuration and Computer Interface

A1850

A request for a second cycle is then made by the BTC and the address of the memory location containing WC is placed on the address lines by the BTC. When the second cycle occurs, WC is transferred from memory to the WC counter in the BTC. The terminate bit (bit 0) contained in the WC word is also tested and a latch is set if the terminate bit is a ONE, which signifies that no more block transfers are to be made after completion of the one being initialized. The maximum time for the entire initialization is four cycles for the CEU execution, plus 2 cycles for the CWA and WC transfers, which occur immediately following CEU execution.

After BTC initialization, words are transferred between the selected peripheral unit and memory over the Unit I/O Bus under the joint control of the BTC, the Block Priority Control (BPC) and the Memory Access Control (MAC). All word transfers are initiated by the unit which sends a Data Transfer Request to the BTC. The Data Transfer Request causes the BTC to request a memory cycle through the BPC to the MAC. When it is determined that a memory cycle can be granted, a Memory Available signal is sent to the BTC. The BTC, in turn, sends a signal to the peripheral unit which causes it to connect to the Unit I/O data lines, execute the data transfer, and then disconnect from the data lines. After completion of a word transfer, the CWA value is incremented and the WC value is decremented in the BTC counters. All words are transferred by repetition of this cycle, which is always initiated by the peripheral unit.

When the value of WC is decremented to zero, the block transfer is terminated. If the terminate latch in the BTC had not been set by the terminate bit in the last WC word acquired from memory, a new block transfer is automatically initiated by the BTC. Re-initialization consists of acquiring new CWA and WC values from the memory locations assigned to the BTC. After re-initialization, an interrupt is generated which signifies that the transfer of the last block is completed and a new block transfer is initialized. The interrupt processing routine can then store in the dedicated locations the CWA and WC values for the next block transfer anytime prior to the completion of the current block transfer. This re-initialization technique reduces the problem of re-initialization block transfers under program control between the times of occurrence of two successive words in a continuous data stream.

If the terminate latch in the BTC had been set by the terminate bit in the last WC word acquired from memory, an interrupt is generated when the value of WC is decremented to zero and no new transfer is initialized by the BTC. In addition, the Data Transfer Request line from the peripheral unit is disconnected until a new Initialize signal is received. Hence, the BTC disconnects from the peripheral unit.

Figure 3.1-3 shows a flowchart of the Block Transfer Control. The operations described in this section are illustrated by the flowchart.

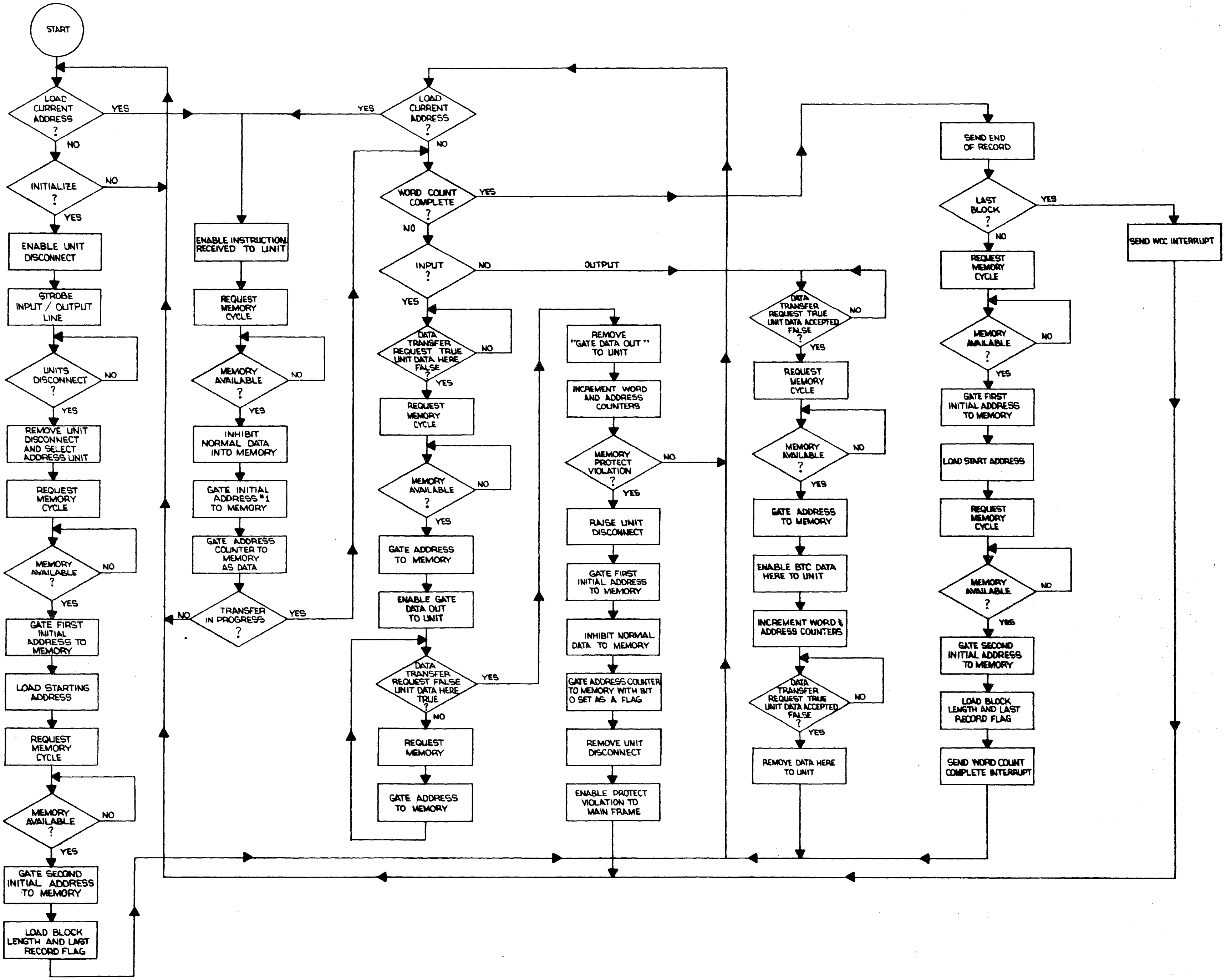

START
LOAD CURRENT ADDRESS ?
YES
NO
INITIALIZE ?
NO
YES
ENABLE UNIT DISCONNECT
STROBE INPUT / OUTPUT LINE
UNITS DISCONNECT ?
NO
YES
REMOVE UNIT DISCONNECT AND SELECT ADDRESS UNIT
REQUEST MEMORY CYCLE
MEMORY AVAILABLE ?
NO
YES
GATE FIRST INITIAL ADDRESS TO MEMORY
LOAD STARTING ADDRESS
REQUEST MEMORY CYCLE
MEMORY AVAILABLE ?
NO
YES
GATE SECOND INITIAL ADDRESS TO MEMORY
LOAD BLOCK LENGTH AND LAST RECORD FLAG
ENABLE INSTRUCTION RECEIVED TO UNIT
REQUEST MEMORY CYCLE
MEMORY AVAILABLE ?
NO
YES
INHIBIT NORMAL DATA INTO MEMORY
GATE INITIAL ADDRESS #1 TO MEMORY
GATE ADDRESS COUNTER TO MEMORY AS DATA
TRANSFER IN PROGRESS ?
NO
YES
LOAD CURRENT ADDRESS ?
YES
NO
WORD COUNT COMPLETE ?
YES
NO
INPUT ?
NO
OUTPUT
YES
DATA TRANSFER REQUEST TRUE UNIT DATA HERE FALSE ?
NO
YES
REQUEST MEMORY CYCLE
MEMORY AVAILABLE ?
NO
YES
GATE ADDRESS TO MEMORY
ENABLE GATE DATA OUT TO UNIT
DATA TRANSFER REQUEST FALSE UNIT DATA HERE TRUE ?
YES
NO
REQUEST MEMORY
GATE ADDRESS TO MEMORY
REMOVE "GATE DATA OUT" TO UNIT
INCREMENT WORD AND ADDRESS COUNTERS
MEMORY PROTECT VIOLATION ?
NO
YES
RAISE UNIT DISCONNECT
GATE FIRST INITIAL ADDRESS TO MEMORY
INHIBIT NORMAL DATA TO MEMORY
GATE ADDRESS COUNTER TO MEMORY WITH BIT 0 SET AS A FLAG
REMOVE UNIT DISCONNECT
ENABLE PROTECT VIOLATION TO MAIN FRAME
DATA TRANSFER REQUEST TRUE UNIT DATA ACCEPTED FALSE ?
NO
YES
REQUEST MEMORY CYCLE
MEMORY AVAILABLE ?
NO
YES
GATE ADDRESS TO MEMORY
ENABLE BTC DATA HERE TO UNIT
INCREMENT WORD & ADDRESS COUNTERS
DATA TRANSFER REQUEST TRUE UNIT DATA ACCEPTED FALSE ?
NO
YES
REMOVE DATA HERE TO UNIT
SEND END OF RECORD
LAST BLOCK ?
YES
NO
SEND WCC INTERRUPT
REQUEST MEMORY CYCLE
MEMORY AVAILABLE ?
NO
YES
GATE FIRST INITIAL ADDRESS TO MEMORY
LOAD START ADDRESS
REQUEST MEMORY CYCLE
MEMORY AVAILABLE ?
NO
YES
GATE SECOND INITIAL ADDRESS TO MEMORY
LOAD BLOCK LENGTH AND LAST RECORD FLAG
SEND WORD COUNT COMPLETE INTERRUPT
A2084

Figure 3.1-3 Block Transfer Control Flowchart

3.2 Block Priority Control

The Block Priority Control (BPC) processes memory cycle requests from the individual BTC's and allocates them on a priority basis. Each BTC is assigned a particular priority. The BPC priority logic is similar to that of the priority interrupt logic in the computer, insuring that higher priority BTC's are always serviced before lower priority units. All BTC's may request a memory cycle simultaneously but the highest priority BTC will be granted the next memory cycle. Figure 3.2-1 is a flowchart of the Memory Access Control Sequence which processes BPC requests.

The BPC operates in conjunction with the mainframe Memory Access Control Logic. The BPC has priority over the mainframe requests so that the mainframe is "locked out" during the time a word transfer involving a BTC takes place. There are certain instructions and certain cycles of other instructions where the BPC is inhibited from stealing cycles from the computer. All of the Input/Output instructions (MIP, MOP, AIP, AOP, TEU, CEU) inhibit the BPC. The last cycle of the DIV instruction and the E_1 cycle of IMS also inhibits the BPC from having access to memory.

A succession of Input/Output instructions however, does not inhibit cycle stealing by the BTC's since the BPC may request memory in between any sequence of instructions. The maximum time that the highest priority BTC would have to wait for a memory cycle would depend on an I/O instruction. All I/O instructions require a minimum of three cycles depending on the length of cable between the computer and the peripheral unit. The Wait flag also causes the cycle to be extended.

The maximum collective transfer rate for a BTC (or group of BTC's) is 572,000 words per second. As mentioned before, cycle stealing from the main program is automatic and each word transferred requires one or more cycles. The peripheral unit must be able to respond in 300 nanoseconds in order not to require an additional computer cycle. The use of the Maximum Transfer Rate signal from the peripheral unit to the BTC is required in order to attain a transfer rate of 572 KC for any one BTC, otherwise, a rate of 286 KC is realized. The maximum transfer rate line is controlled by the peripheral unit. This signal must be enabled during the first Data Transfer Request to the BTC and remain enabled as long as the unit desires to continue at a maximum rate. Upon completion of the block, the unit must disable this control line in case another unit operating on the same BTC should desire to connect for data transfer.

It must be remembered that the computer is inhibited from every obtaining a cycle as long as a BTC is operating at this maximum transfer rate. Another point of importance is that the BTC must be set up to terminate at the finish of the particular block, otherwise, the computer is "locked out" since there is no way for the program to modify the dedicated memory location containing the block length.

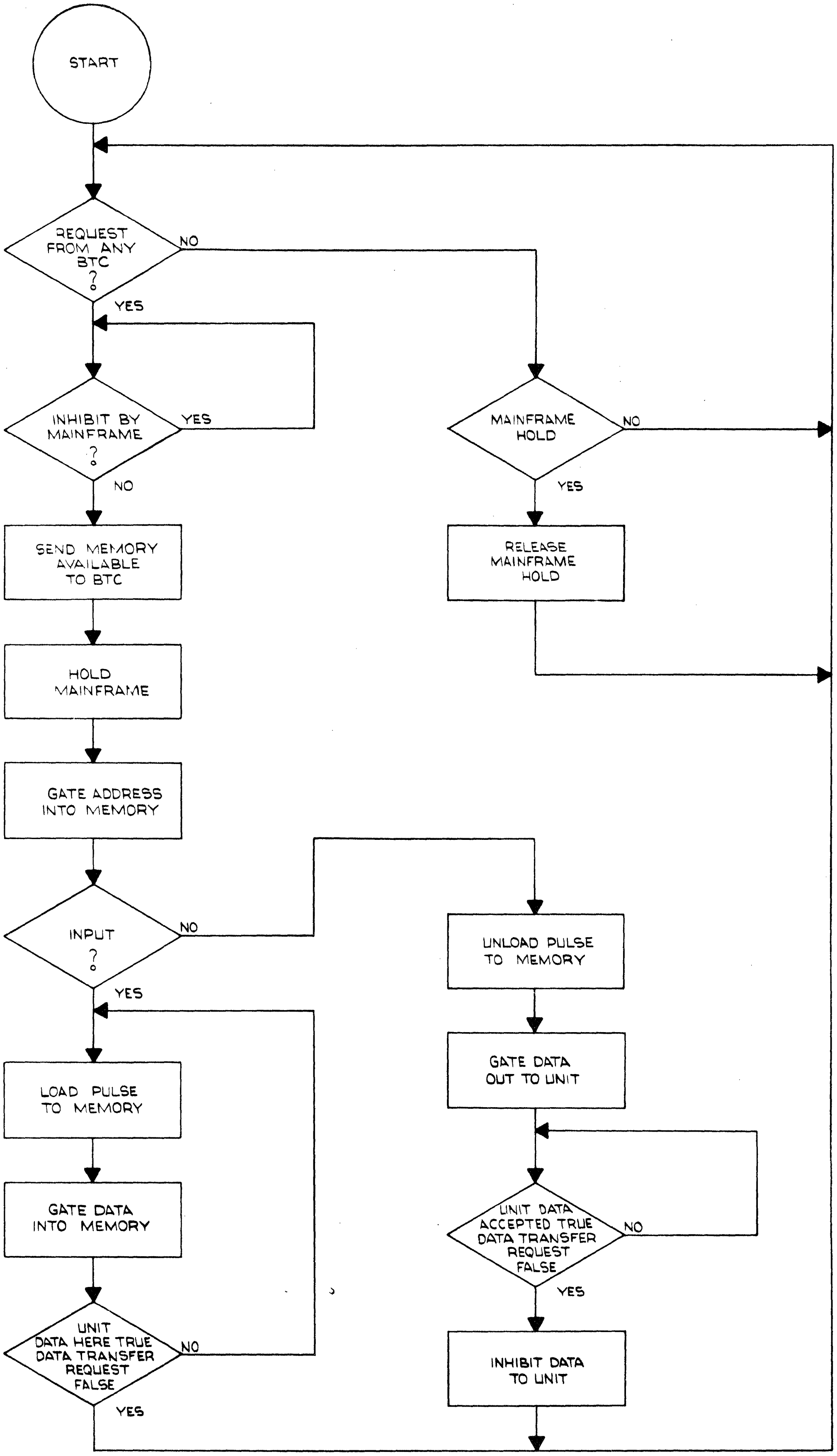


Figure 3.2-1 Memory Access Control Flowchart

A2072

A problem arises when operating at the maximum transfer rate. If the unit, through some failure or programming error, does not input the required number of words to the BTC the computer remains "hung up" since the BPC is requesting memory awaiting the last data word to be transferred. The only way out of this situation is to have the Stall Alarm option which waits a certain period of time and then generates an interrupt which overrides all other conditions in the computer. These conditions mentioned above are all failure conditions or programming errors and may be given consideration as such.

3.3 External BTC (External Memory Access)

The external memory access allows a special peripheral unit to have direct random access to the computer memory. This access is completely asynchronous with the computer timing and operates in the Block Priority Control (BPC) as do the Block Transfer Control (BTC) units. This unit connects in place of the eighth BTC thus leaving a maximum capability of 7 BTC's.

Data communication between the computer and the special device is accomplished over the I/O bus in the same manner as all other communications to peripheral devices. One special cable is required over which address information, memory cycle requests, load and unload requests, and other control signals are carried. A block diagram is shown in Figure 3.3-1 depicting the logical connection and signal interface with the BPC.

Since the memory priority is established by the BPC it is possible for this special unit to have highest priority in memory assuming that all BTC's were given lower priority. The BPC request to the Memory Access Control has priority over the computer mainframe.

A great many advantages can be seen from having this option available. No initialization is required, as in a BTC, thus access to memory can be as short as 1.75 usec. When using the BTC it is necessary to issue a CEU instruction to initialize the BTC and then, in turn, the BTC must request two memory cycles to obtain its starting address and block length (total of 6 cycles). If only one data word (block length of one) is to be transferred, a minimum of 7 computer cycles are required. Another important feature is the absence of program intervention or overhead. The external unit generates all of the necessary signals to load or unload a desired location in memory.

When requesting a memory cycle from the external unit, a line called Memory Request is enabled. The minimum pulse width that can be presented is 2 usec. and the maximum width must be less than the duration of the load or unload cycle following the memory available signal from the BPC logic. In other words, if the external unit does not desire more than one sequential memory cycle the request line should remain enabled for 2 usec. It is possible to enable the request line for a longer period of time and have each successive cycle granted to the external unit providing there are not BTC requests having higher memory priority.

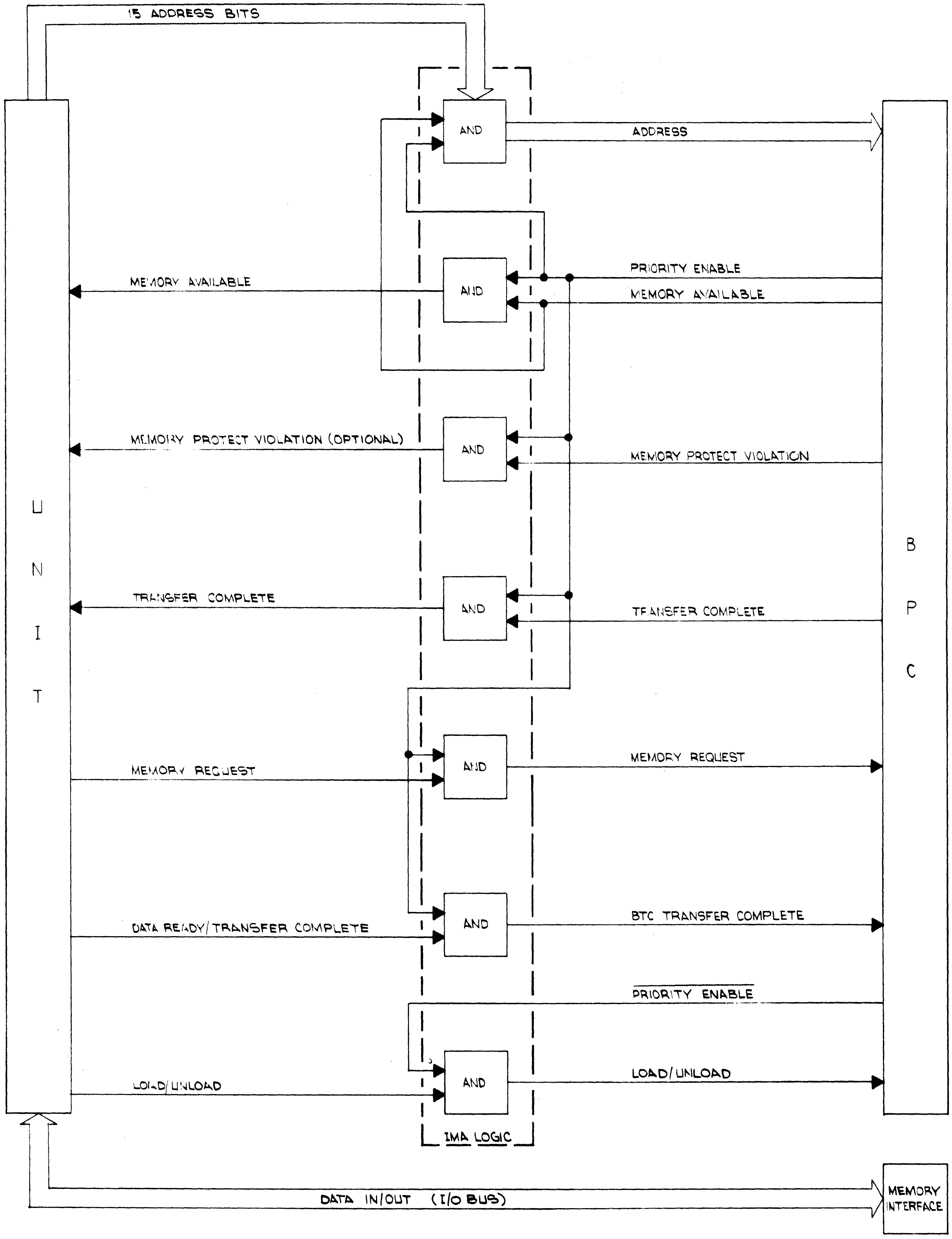


Figure 3.3-1 External Memory Access Block Diagram

A2076

When operating in the output mode the external unit must enable the transfer complete line signaling the BPC that it has received the data word from memory so the BPC can release the memory hold. Data from memory remains on the I/O Bus until this signal is received. When operating in the input mod e the external unit must enable its data lines upon receipt of a memory available signal from the BPC. After the data word has been loaded into memory the BPC then issues a transfer complete signal at which time the unit must remove its data from the I/O Bus.

In either mode of operation the address lines, signifing the memory location desired, must be enabled coincident with the memory request signal. The desired direction of transfer must also be enabled at the same time.

The Memory Protect option is available for use with a unit capable of external memory access. Control of the protect violation is accomplished by the external unit.

3.4 Block Transfer Control Unit Interface

The BTC interface logic is an integral part of the data terminal used with a unit operating with a BTC. As shown in Figure 3.1-2 the units data terminal connects both to the standard I/O Bus and to the Block Transfer Control.

Figure 3.4-1 shows a logic block diagram of the logic required to provide the BTC Interface. Signal lines applied through cable terminators and cable drivers interface with the BTC through the BTC cable. All other signals are derived from either the data terminal or the unit control logic.

The Data Transfer Request line is the primary signal to the BTC used to initiate data transfer. The line is enabled by one of two gates depending on the direction of transfer. When the BTC is operating in the input mode the ready condition is determined by the fact that the input buffer register has been loaded. Conversely, when the BTC is operating in the output mode the ready condition is determined by the fact that the output register has been unloaded and ready to receive another data word. In conjunction with the data transfer request line are the unit data accepted and unit data here lines to the BTC. The labels applied to these lines are self-explanatory.

Figure 3.4-2 shows the timing diagram of the Block Transfer Control referenced to the BTC Unit Interface. The diagram shows an example of three data words transferred between the peripheral unit and the memory including the initialization cycle where the CEU instruction is utilized. Both input and output data transfers are shown. The logic shown in Figure 3.4-2 is representative of a general purpose interface, and depending on the nature of the peripheral unit, could become much more complex.

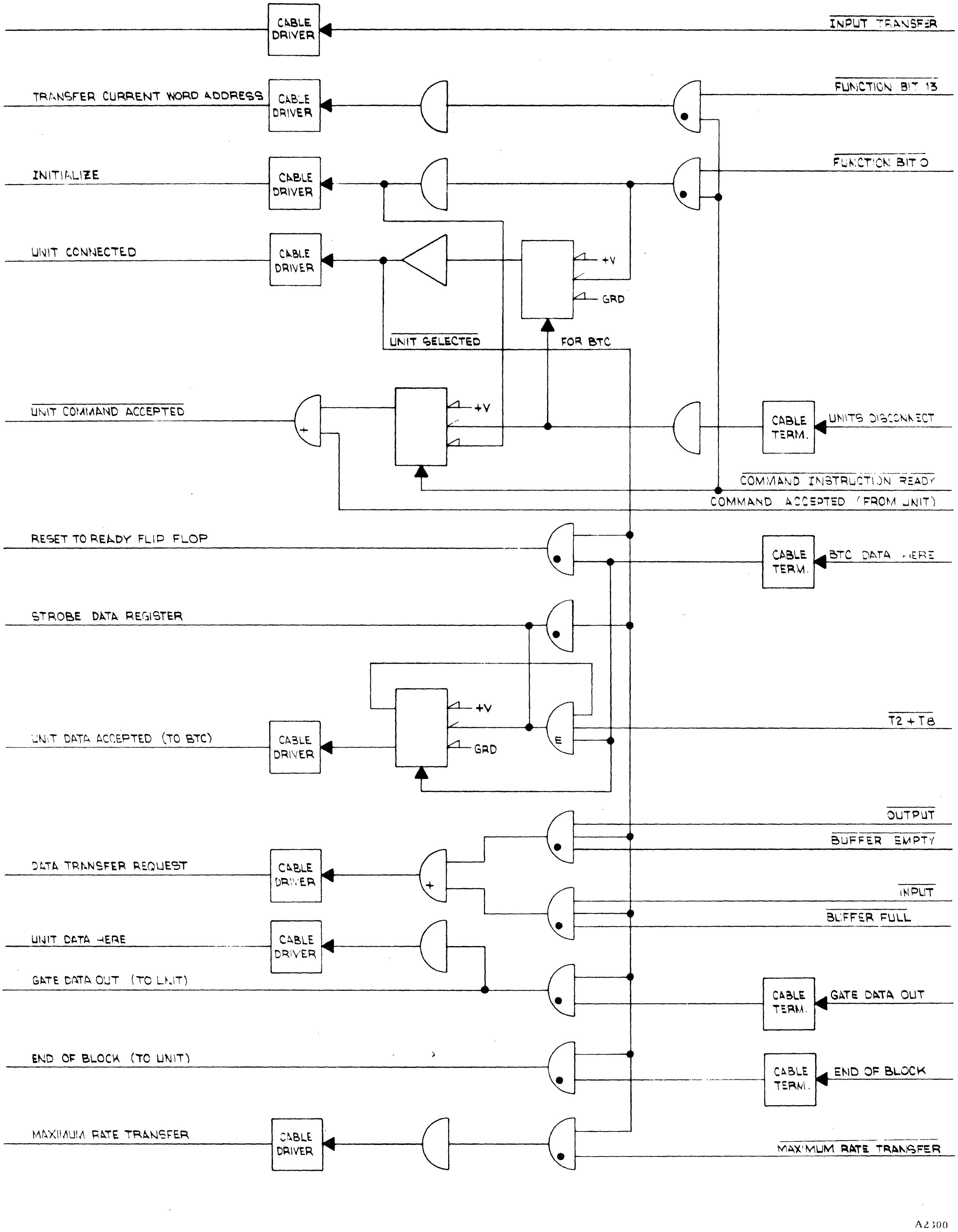


Figure 3.4-1 Block Transfer Control Interface Logic Diagram

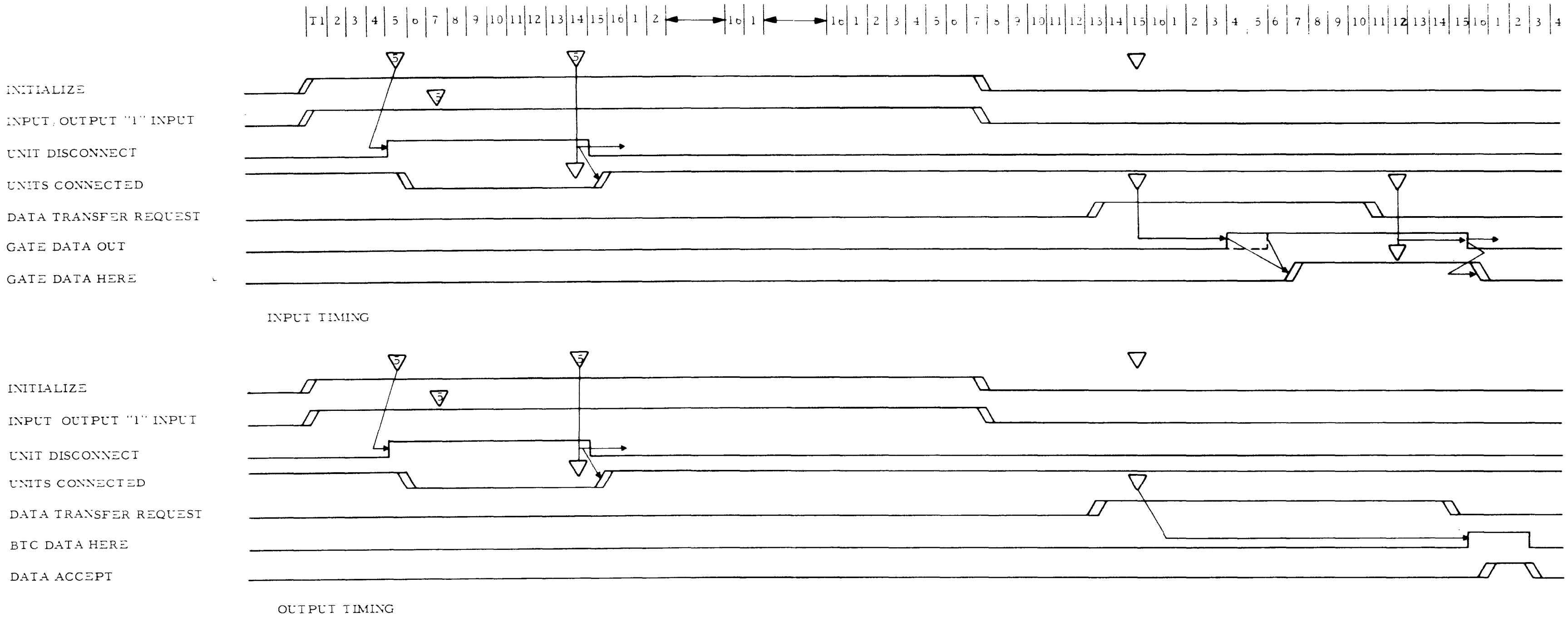


Figure 3.4-2 Block Transfer Control Interface Timing Diagram

A2504

4.0 Priority Interrupt System

4.1 General Description

The SEL 810A computer can have up to 96 individual levels of priority interrupts. Ninety-four of these levels can be selectively enabled and disabled under program control. Two levels are supplied with the basic computer. Additional levels are available in groups of 16 each, except that the first group contains 14 additional levels.

Assignment of interrupts is highly flexible. Internal signals such as Overflow and Memory Parity can be connected to interrupt levels. BTC, End of Block signals and external signals from peripheral units and custom system components are connected to the levels which best fit the operation of each system.

Two special interrupts, Power Fail Safe/Restore and Stall Alarm, are optionally available. These levels, when present, are always enabled and carry the highest priorities. Interrupts signals at these levels override all other computer functions, including Halt, I/O Wait and indirect address chaining.

A unique location in memory is assigned to each interrupt level. These locations are assigned in MAP 1 to keep the entire MAP 0 available for program usage. Location $1,002_8$ is assigned to the highest priority level, location 1003_8 to the second highest level, etc. Table 4.1 shows the assignment of interrupt locations as well as BTC locations.

When an interrupt signal is recognized by the mainframe, a wired-in instruction SPB*L (Store Place and Branch Indirect) is executed, where L is the address of the memory location assigned to the interrupt level. By storing the starting location of the interrupt processing routine in L, a linkage is provided to any point in memory. Since the address of the next instruction to have been executed in the interrupt program is stored in the interrupt routine entry point by the SPB instruction, a means for returning to the point of interrupt is provided.

The mainframe may be interrupted by a particular interrupt level provided that:

(a) the level has been previously enabled and
(b) no higher level interrupt is active.

If a higher level interrupt is active when an interrupt signal occurs, the interrupt will be stored until the completion of execution of the higher level interrupt processing routine. The lower level routine will then be initiated. It will continue until completed or until interrupted by a higher level interrupt signal. In this case, the lower level routine will be completed after completion of the higher level routine. Program control will then be returned to the original point of interrupt. The priority logic enables any number of interrupt levels to be requested at the same time. Routine execution is always performed in the order of priority of the requested interrupts.

TABLE 4. 1 PRIORITY INTERRUPT AND BTC MEMORY LOCATION ASSIGNMENTS

MEMORY LOCATION (Octal)	INTERRUPT ASSIGNMENT		MEMORY LOCATION (Octal)	INTERRUPT ASSIGNMENT	
	GROUP	LEVEL		GROUP	LEVEL
1000	PFS/R		1051	3	4
1001	SA		1052	3	5
1002	0	1	1053	3	6
1003	0	2	1054	3	7
1004	0	3	1055	3	8
1005	0	4	1056	3	9
1006	0	5	1057	3	10
1007	0	6	1060	BTC 1 FWA	
1010	0	7	1061	BTC 1 WD CNT	
1011	0	8	1062	BTC 2 FWA	
1012	0	9	1063	BTC 2 WD CNT	
1013	0	10	1064	BTC 3 FWA	
1014	0	11	1065	BTC 3 WD CNT	
1015	0	12	1066	BTC 4 FWA	
1016	1	1 (Std.)	1067	BTC 4 WD CNT	
1017	1	2 (Std.)	1070	BTC 5 FWA	
1020	1	3	1071	BTC 5 WD CNT	
1021	1	4	1072	BTC 6 FWA	
1022	1	5	1073	BTC 6 WD CNT	
1023	1	6	1074	BTC 7 FWA	
1024	1	7	1075	BTC 7 WD CNT	
1025	1	8	1076	BTC 8 FWA	
1026	1	9	1077	BTC 8 WD CNT	
1027	1	10	1100	3	11
1030	1	11	1101	3	12
1031	1	12	1102	4	1
1032	2	1	1103	4	2
1033	2	2	.	.	.
1034	2	3	.	.	.
1035	2	4	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
1050	3	3	1157	7	10

PFS/R = POWER FAIL SAFE/RESTORE
SA = STALL ALARM
FWA = FIRST WORD ADDRESS
WD CNT = WORD COUNT

4.2 Priority Interrupt Detailed Logic Description

The detailed logic circuits for one priority interrupt level are shown in Figure 4.2-1. Two priority interrupt levels are contained on each 8242 priority interrupt circuit card. A priority interrupt system consists of up to 96 levels, input control gates, an output OR gate and a unitary to binary converter.

The flip-flop is designated as the "in-service" latch, the latch associated with input pins 3 and 14 is designated as the "request" latch, and the latch associated with input pins 9, 10 and 19 is designated as the "enable" latch. When the INTERRUPT switch on the control panel is positioned to ON, a "hold at reset" signal is removed from the "in-service" and "enable" latches. When a PIE instruction is performed, a strobe pulse from the timing and control logic is gated to query the AND gates associated with the enable latches in each circuit. If the level has been selected, the AND gate provides a signal to set the enable latch. A level select gate performs a similar function to reset the latch when a PID instruction is performed. The set output of the latch enables or inhibits a 4-input AND gate.

The second "Interrupt Source" line connects the external signal to the priority interrupt logic. This signal is gated to set the "request" latch at any time except between T12 and T2 time by the timing and control logic. If an interrupt signal has been generated, the 2-input AND gate provides a signal to set the "request" latch. If the "enable" latch has been set and a higher priority interrupt routine is not being performed, the 4-input AND gate provides an interrupt request to the unitary to binary converter and through the output OR gate to the timing and control logic. When the priority interrupt request is honored, a signal is received from the timing and control logic to strobe the unitary to binary converter to generate a Store Place and Branch instruction which is placed on the memory output bus. The output of the "request" latch is also presented through an AND gate and OR gate to inhibit the output of the "request" latches in the lower level priority interrupt channels.

If the request is honored by the mainframe, the timing and control logic provides a signal which permits the in-service control latch to set the "in-service" flip-flop. The positive output of the "in-service" flip-flop terminates the priority interrupt request at the 4-input AND gate and inhibits the lower level priority interrupt channels.

Upon completion of the priority interrupt routine, a "reset in-service flip-flop" command is received from the control logic to query the AND gate associated with pin 16. If the "in-service" latch of a higher level circuit has not been set, indicating that a higher priority interrupt level interrupted this lower level interrupt's program, the AND gate provides a signal to reset the "in-service" and "request" latches. A detailed timing diagram and flow chart of the interrupt process is shown in Figure 4.2-2.

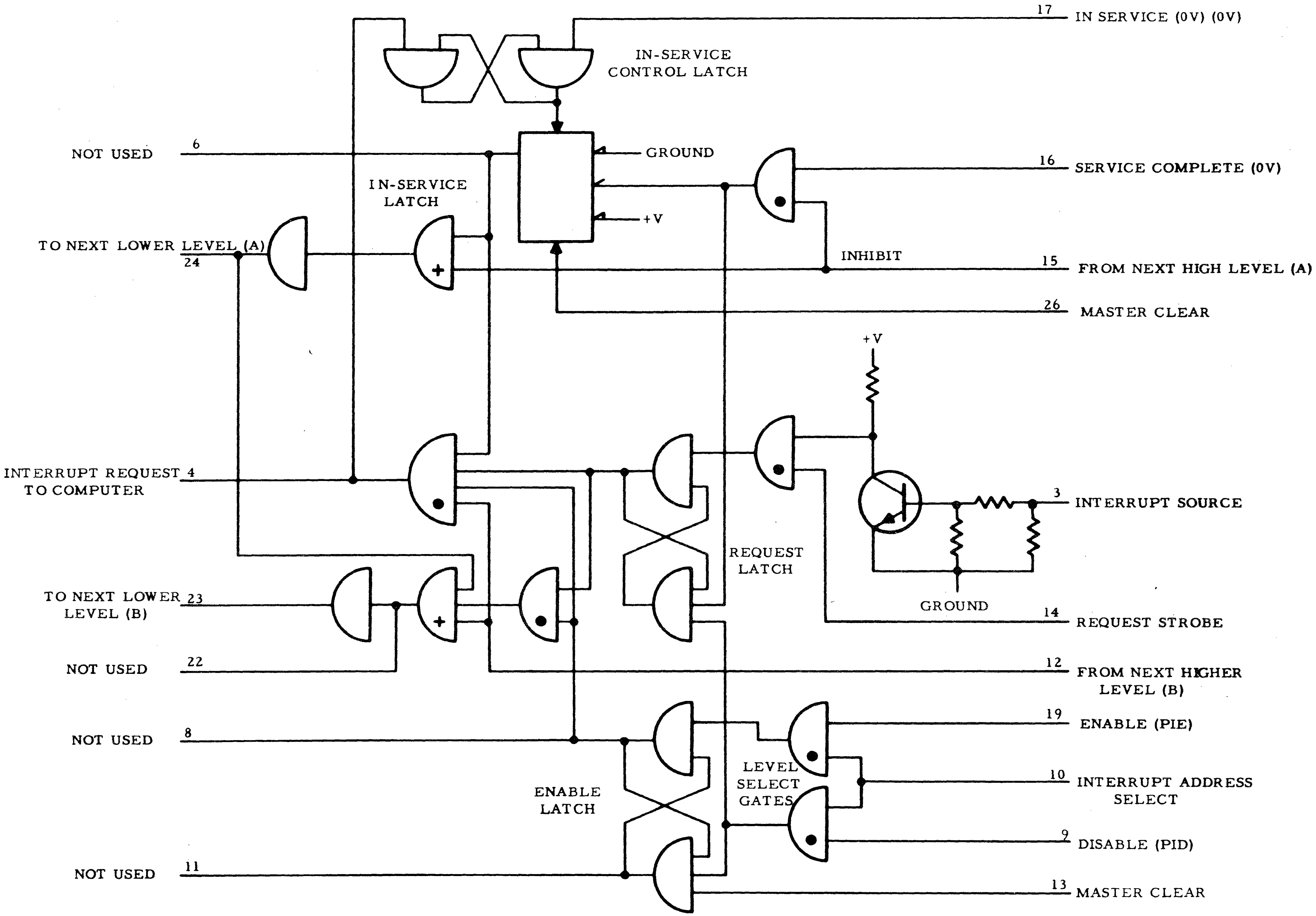
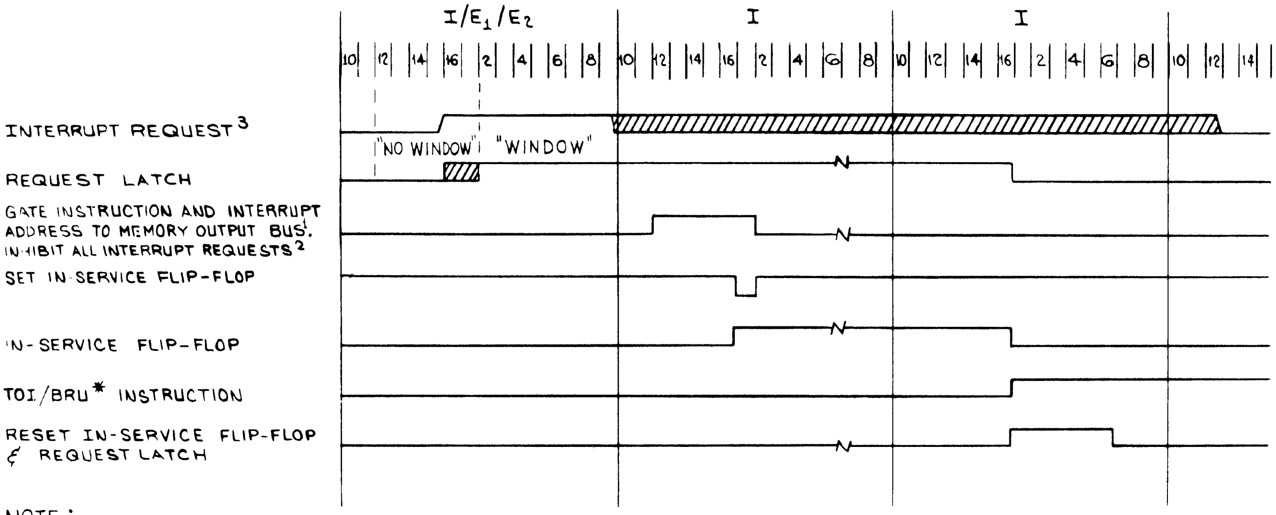


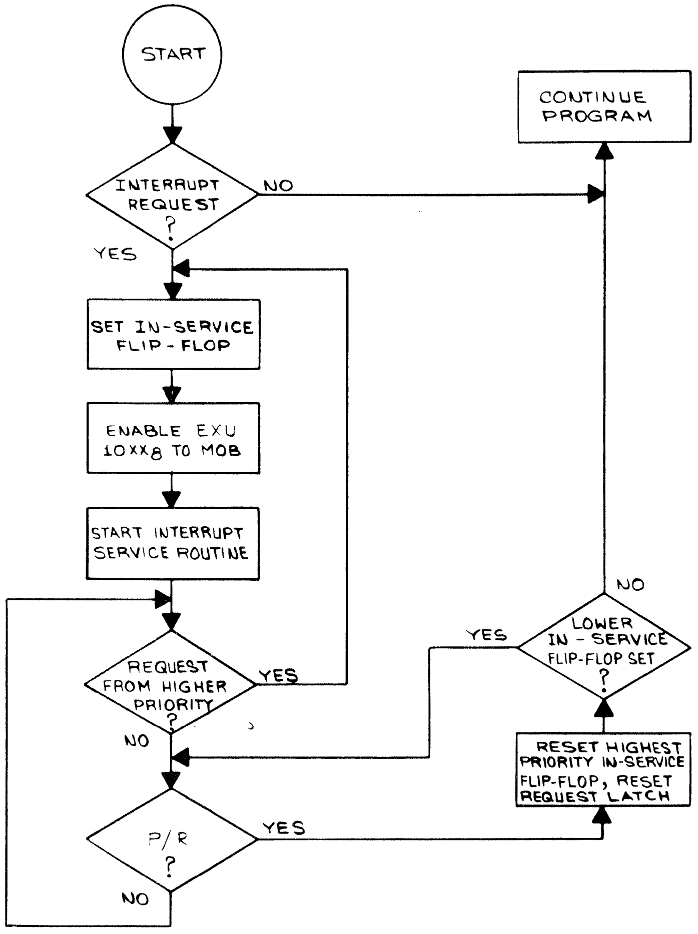
Figure 4.2-1 Priority Interrupt Logic Diagram

A2301



NOTE :

1. 810A-SPB* TO ADDRESS $10XX_8$
   840A-EXU TO ADDRESS $1XX_8$
2. PREVENTS ANTICOINCIDENCE
3. MINIMUM REQUEST PULSE WIDTH = 1 μsec
   MAXIMUM REQUEST PULSE WIDTH = 5 μsec


START
INTERRUPT REQUEST ?
NO
YES
CONTINUE PROGRAM
SET IN-SERVICE FLIP-FLOP
ENABLE EXU 10XX8 TO MOB
START INTERRUPT SERVICE ROUTINE
REQUEST FROM HIGHER PRIORITY ?
YES
NO
P/R ?
YES
NO
RESET HIGHEST PRIORITY IN-SERVICE FLIP-FLOP, RESET REQUEST LATCH
LOWER IN-SERVICE FLIP-FLOP SET ?
YES
NO

A2154

Figure 4.2-2 Priority Interrupt Timing Diagram

Figure A1 SEL 810A Peripheral Unit Command and Test Code Formats

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MAGNETIC TAPE FORMAT 0		CONNECT = 1 DISCONN = 0	WORD TRANSFER READY INTERRUPT	END OF RECORD INTERRUPT	(FORMAT) 0	REWIND	ERASE 4 INCHES OF TAPE	BCD = 1 BINARY = 0	DENSITY		TAPE TRANSPORT			CURRENT WORD ADDRESS IN	CHARACTERS PER WORD	
MAGNETIC TAPE FORMAT 1	BTC INITIALIZE		WORD TRANSFER READY INTERRUPT	END OF RECORD INTERRUPT	(FORMAT) 1	WRITE RECORD	WRITE END OF FILE	READ RECORD	ADVANCE RECORD	ADVANCE/ END OF FILE	BACKSPACE RECORD	BACKSPACE END OF FILE	CORRECT CRC ERROR (9 TRACK OPTION)	CURRENT WORD ADDRESS IN	CAB	
ASR-33/35			IN	OUT	READER MODE	KEY MODE	CLEAR									
PAPER TAPE READER AND PUNCH			IN	OUT	PUNCH POWER ON	PUNCH POWER OFF	READER ENABLE	READER DISABLE								
NCR CARD READER			IN		BINARY FEED CARD	BCD FEED 1 CARD		REAL STACK OFFSET	EJECT CARD (PUNCH)	PUNCH STACKER OFFSET						
CALCOMP			END OF EX.		PEN DOWN	PEN UP	DRUM DOWN	DRUM UP	CAR. LEFT	CAR. RIGHT						
LINE PRINTER			END OF PRINT	BUFF NOT BUSY	ADVANCE FORMAT N	ADVANCE 1 LINE	TOP OF DRUM	PRINT	CLEAR BUFFER	FILL BUFFER						
DISC SEEK			SEEK ERROR	SEEK COMPLETE		NUMBER OF TRACKS TO BE MOVED 64	32	16	8	4	2	1	1		*FWD	*REV
DISC DATA			SEEK ERROR	SEEK COMPLETE	SECTOR NUMBER 8	4	2	1	HEAD NUMBER 8	4	2	1	0		WRITE	READ

CEU Second Word Format

*TO SEEK TRACK 00 BOTH BITS MUST BE ZERO

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
CARD READER AND PUNCH										SKIP NO PUNCH ERROR						
DISC					SKIP SEEK COMP	SKIP NO SEEK ERROR	SKIP ON BOD	SKIP ON BOS	SKIP PACK ON LINE	SKIP NO READ OVERFLOW	SKIP NO WRITE OVERFLOW	SKIP NO CHECK SUM ERROR	SKIP NO FILE UNSAFE	SKIP DCU READY	SKIP NOT BUSY	
MAGNETIC TAPE	SKIP ON NOT BUSY	SKIP ON NO END OF FILE	SKIP ON NO OVERFLOW	SKIP ON LOAD POINT	SKIP ON END OF RECORD INTERRUPT	SKIP ON NO PARITY ERROR	SKIP ON WRITE RING IN	SKIP ON NO END OF TAPE	SKIP ON REWINDING	SKIP ON NO CRC ERROR (9 TRACK ONLY)						
LINE PRINTER					SKIP NOT BUSY		SKIP NO PARITY ERROR		SKIP NO BOTTOM OF FORM	SKIP IF PRINTER OP.						

TEU Second Word Format

A2082

APPENDIX B

810A - INPUT/OUTPUT CONNECTOR DESCRIPTION

This appendix contains illustrations and tables which provide information required to interface external equipment with the SEL 810A computer, including peripheral units, block transfer controls, and priority interrupts. The conventions of fabrication as applied to connector keying and signal definition are stated and will be used as a standard.

The I/O connector panel assembly, a representation shown in Figure B1, is located on the lower center of the computer cabinet. All equipment external to the main computer cabinet is connected through this panel. The configuration shown will accommodate 64 peripheral units (J1, J4, J5 and J6), 8 Block Transfer Controls (J3, J14-20), and 62 individual priority interrupts (J2 and J7-J13).

If the system does not require in excess of 16 peripheral units, only J1 is provided on the I/O connector panel (1R3). If additional jacks are required due to more than 16 units, wiring information is provided upon request. J1 is also referred to as the computer I/O Bus, since it is common to all peripheral units supplying both data and control functions.

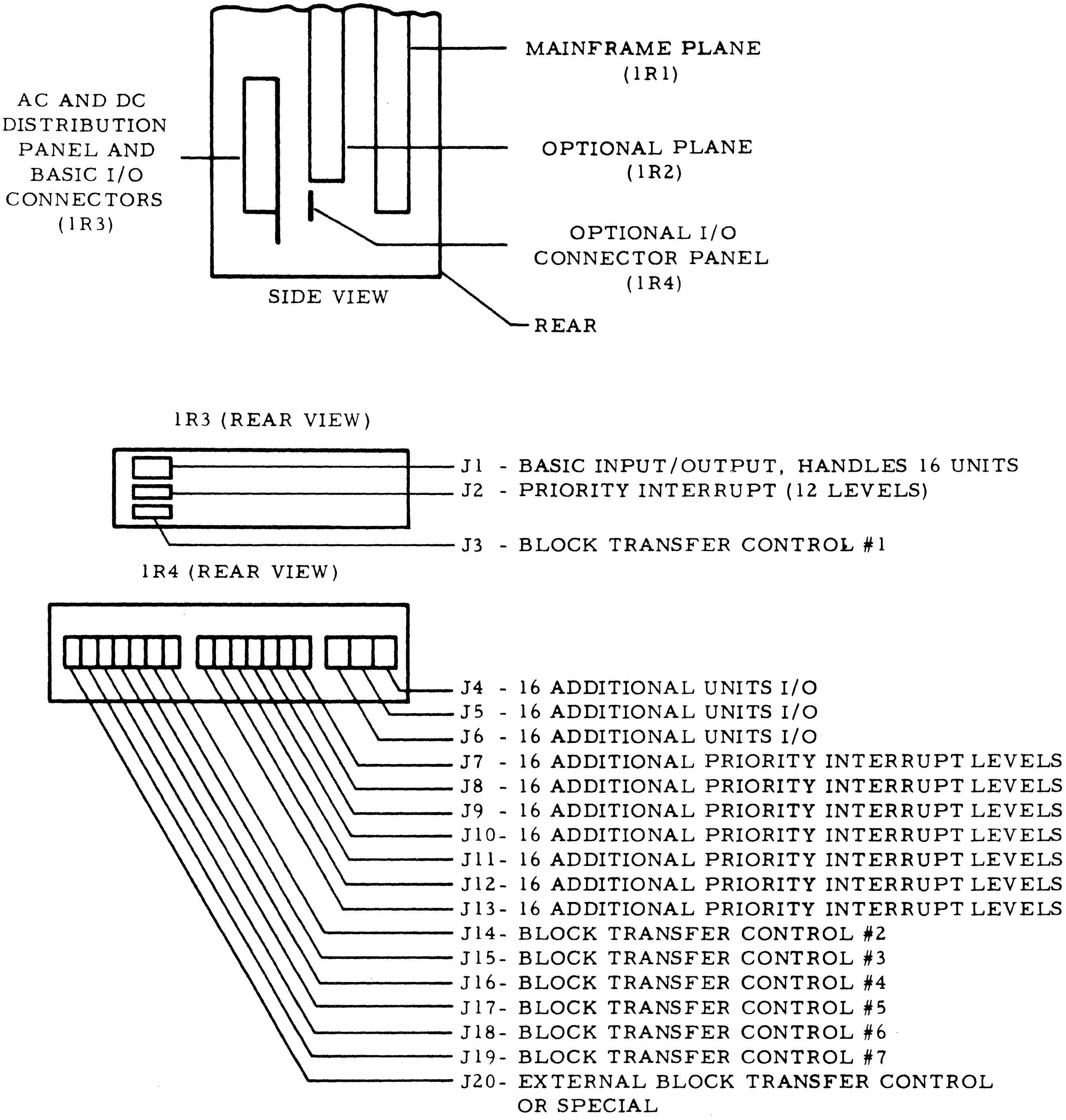
Figure B2 is a functional diagram showing a typical connection of a peripheral unit to the computer I/O Bus. All units are wired with two connectors in parallel in order to facilitate additional peripheral units as desired. The end of the cable, i. e., the last unit in the chain, is terminated with a terminator plug consisting of 100-ohm resistors across each twisted pair. Without this terminator plug, reflections will be set up on the open transmission line which will cause malfunctions.

Tables B1 and Figure B3 define the signal lines contained in the 51 pair cable and the keying associated with the cable connectors and the fixed connectors which are mounted in the unit and in the computer I/O connector assembly. The numbers specifying keying hardware refer to the items listed in Table B2, which is a list of the connector hardware and the associated AMP part number. Table B1 wiring information applies only to J1.

Table B3 and Figure B4 define the signal lines and connector keying applying to the first Block Transfer Control. The connectors defined as J3, J14 - J20 are used to connect units operating in conjunction with BTC. Up to 16 units can be connected to one BTC, thus similar chaining of units as in the I/O Bus is required. Each unit is connected to the computer I/O Bus regardless of whether it operates on a BTC since data communication and command functions are performed via the I/O Bus.

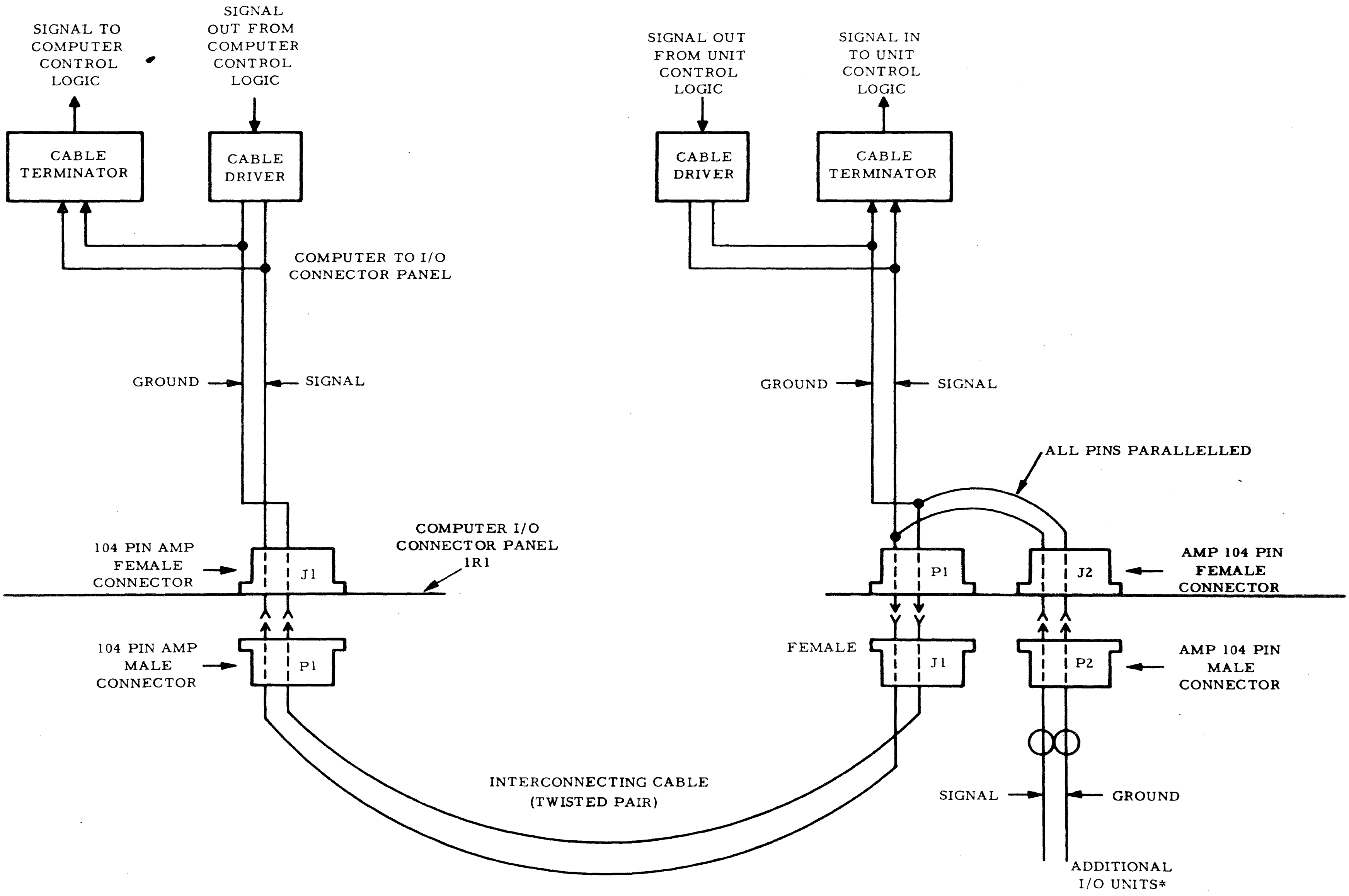
Tables B4 and B5 defines the priority interrupts, their associated connector, module number and assigned octal memory location. The two standard interrupts are shown in the table which are the first two interrupts in group 2. Each interrupt module contains 16 priority interrupts. The first two interrupts are assigned to Power Fail Safe and Stall Alarm (which are purchased as an optional item). These two interrupt levels have the highest priority, but are not controllable in respect to enabling or disabling under program control. The highest interrupt available to the programmer then becomes interrupt number 3, which is in the first group of the first module. Priority decreases as group and interrupt numbers increase.

Figure B5 shows functionally the connection of a priority interrupt and also defines the connector keying. The external signal necessary to assure recognition by the computer is also shown.


MAINFRAME PLANE
(1R1)
AC AND DC
DISTRIBUTION
PANEL AND
BASIC I/O
CONNECTORS
(1R3)
OPTIONAL PLANE
(1R2)
OPTIONAL I/O
CONNECTOR PANEL
(1R4)
SIDE VIEW
REAR
1R3 (REAR VIEW)
J1 - BASIC INPUT/OUTPUT, HANDLES 16 UNITS
J2 - PRIORITY INTERRUPT (12 LEVELS)
J3 - BLOCK TRANSFER CONTROL #1
1R4 (REAR VIEW)
J4 - 16 ADDITIONAL UNITS I/O
J5 - 16 ADDITIONAL UNITS I/O
J6 - 16 ADDITIONAL UNITS I/O
J7 - 16 ADDITIONAL PRIORITY INTERRUPT LEVELS
J8 - 16 ADDITIONAL PRIORITY INTERRUPT LEVELS
J9 - 16 ADDITIONAL PRIORITY INTERRUPT LEVELS
J10- 16 ADDITIONAL PRIORITY INTERRUPT LEVELS
J11- 16 ADDITIONAL PRIORITY INTERRUPT LEVELS
J12- 16 ADDITIONAL PRIORITY INTERRUPT LEVELS
J13- 16 ADDITIONAL PRIORITY INTERRUPT LEVELS
J14- BLOCK TRANSFER CONTROL #2
J15- BLOCK TRANSFER CONTROL #3
J16- BLOCK TRANSFER CONTROL #4
J17- BLOCK TRANSFER CONTROL #5
J18- BLOCK TRANSFER CONTROL #6
J19- BLOCK TRANSFER CONTROL #7
J20- EXTERNAL BLOCK TRANSFER CONTROL
OR SPECIAL

A2302

Figure B1 I/O Connector Panels


SIGNAL TO COMPUTER CONTROL LOGIC
SIGNAL OUT FROM COMPUTER CONTROL LOGIC
CABLE TERMINATOR
CABLE DRIVER
COMPUTER TO I/O CONNECTOR PANEL
GROUND
SIGNAL
104 PIN AMP FEMALE CONNECTOR
J1
COMPUTER I/O CONNECTOR PANEL 1R1
104 PIN AMP MALE CONNECTOR
P1
INTERCONNECTING CABLE (TWISTED PAIR)
SIGNAL OUT FROM UNIT CONTROL LOGIC
SIGNAL IN TO UNIT CONTROL LOGIC
CABLE DRIVER
CABLE TERMINATOR
GROUND
SIGNAL
ALL PINS PARALLELLED
P1
J2
AMP 104 PIN FEMALE CONNECTOR
FEMALE
J1
P2
AMP 104 PIN MALE CONNECTOR
SIGNAL
GROUND
ADDITIONAL I/O UNITS*

* TERMINATOR PLUG USED WHEN NO UNIT CONNECTED

Figure B2 Typical I/O Unit Cabling

P1706

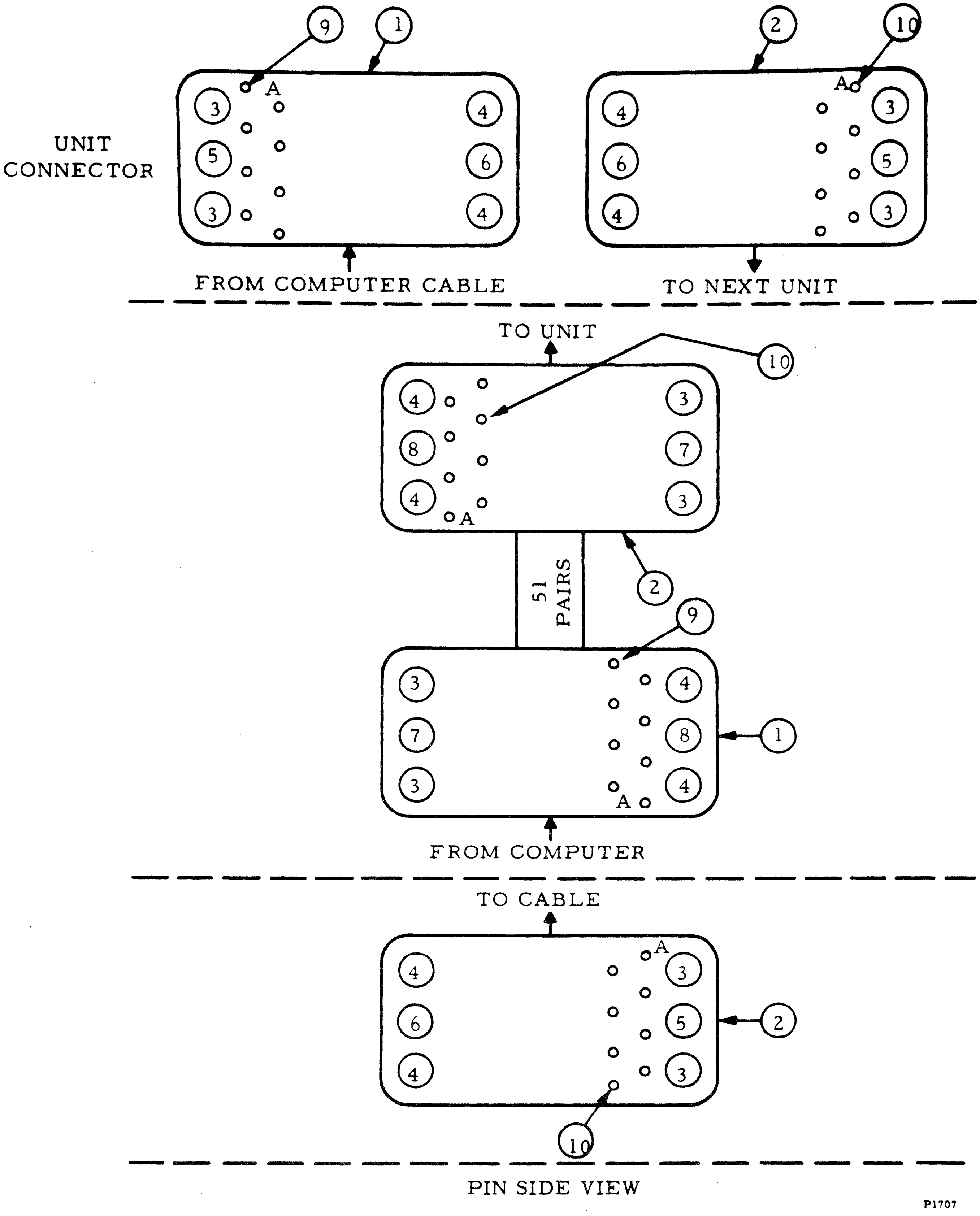
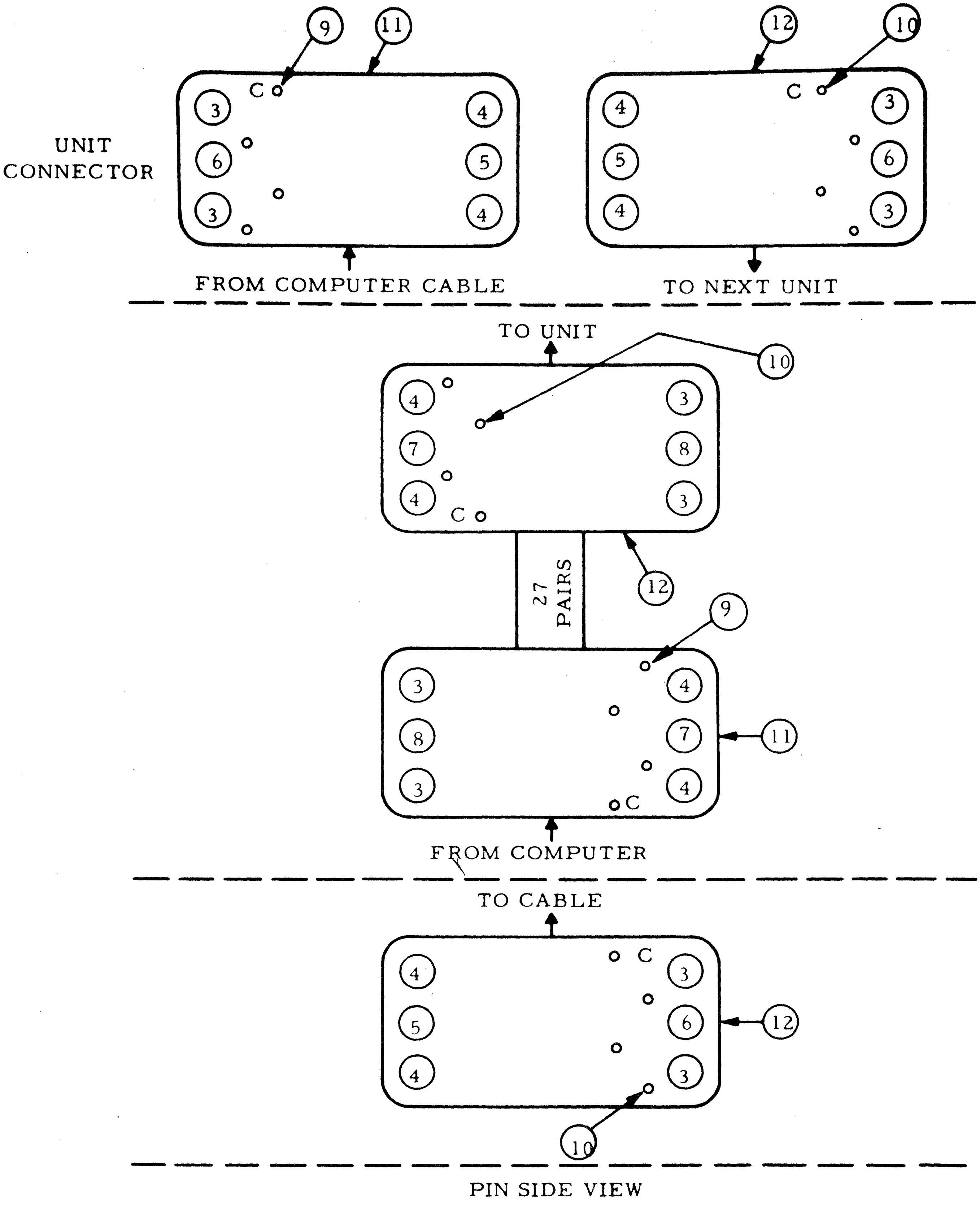
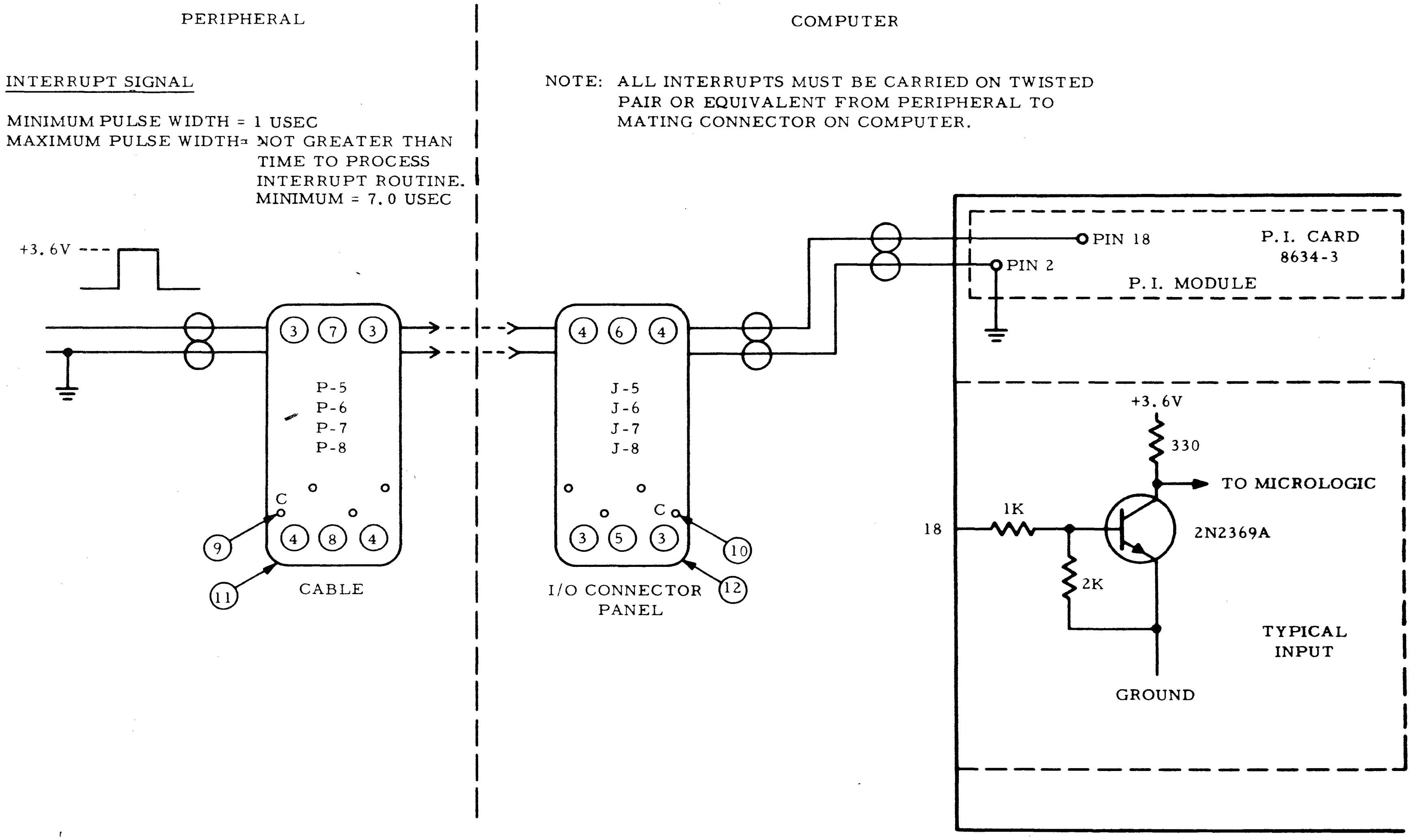


Figure B3 Connector Conventions - I/O Bus



P1714

Figure B4 Connector Conventions - Block Transfer Control



INTERRUPT SIGNAL

MINIMUM PULSE WIDTH = 1 USEC
MAXIMUM PULSE WIDTH= NOT GREATER THAN TIME TO PROCESS INTERRUPT ROUTINE. MINIMUM = 7.0 USEC

NOTE: ALL INTERRUPTS MUST BE CARRIED ON TWISTED PAIR OR EQUIVALENT FROM PERIPHERAL TO MATING CONNECTOR ON COMPUTER.

Figure B5 Typical Priority Interrupt Connection - SEL 810A

P1709

TABLE B1 - I/O BUS CONNECTOR

SIGNAL WIRE			GROUND WIRE	
PIN NO.	WIRE COLOR	FUNCTION	PIN NO.	WIRE COLOR
101	Brown	Data or Function Bit 0	201	Black
102	Red	Data or Function Bit 1	202	Black
103	Orange	Data or Function Bit 2	203	Black
104	Yellow	Data or Function Bit 3	204	Black
105	Green	Data or Function Bit 4	205	Black
106	Blue	Data or Function Bit 5	206	Black
107	Violet	Data or Function Bit 6	207	Black
108	Gray	Data or Function Bit 7	208	Black
109	White	Unit Select Bit 1 from Computer	209	Black
110	Brown	Unit Select Bit 2 from Computer	210	Blk-Wht
111	Red	Unit Select Bit 4 from Computer	211	Blk-Wht
112	Orange	Unit Select Bit 8 from Computer	212	Blk-Wht
113	Yellow	Unit Select Bit 16 from Computer	213	Blk-Wht
301	Brown	Data Bit 8	401	White
302	Red	Data Bit 9	402	White
303	Orange	Data Bit 10	403	White
304	Yellow	Data Bit 11	404	White
305	Green	Data Bit 12	405	White
306	Blue	Data Bit 13	406	White
307	Violet	Data Bit 14	407	White
308	Gray	Data Bit 15	408	White
309	Green	Unit Select Bit 32 from Computer	409	Blk-Wht
310	Blue	Transfer Instruction from Computer	410	Blk-Wht
311	Violet	Test Instruction from Computer	411	Blk-Wht
312	Brown	Command Instruction from Computer	412	Blue
313	Red	Instruction Sync from Computer	413	Blue
501	Brown		601	Gray
502	Red		602	Gray
503	Orange	Input Interrupt to Computer	603	Gray
504	Yellow	Output Interrupt to Computer	604	Gray
505	Green	Parity Error from Computer	605	Gray
506	Blue	Parity Check Request to Computer	606	Gray
507	Violet	Parity Bit from Computer	607	Gray
508	Orange	Parity Bit to Computer	608	Blue
509	Yellow	Input/Output from Computer	609	Blue
510	Green	Wait Bit from Computer	610	Blue
511	Brown	Clock (T2 or T8) from Computer	611	Green
512	Red	Data Here from Computer	612	Green
513	Orange	Data Accepted from Computer	613	Green

Continued on next page.

TABLE B1 - I/O BUS CONNECTOR CONT'D

SIGNAL WIRE			GROUND WIRE	
PIN NO.	WIRE COLOR	FUNCTION	PIN NO.	WIRE COLOR
701	Brown		801	Violet
702	Red		802	Violet
703	Orange		803	Violet
704	Yellow		804	Violet
705	Green		805	Violet
706	Blue		806	Violet
707	Yellow		807	Green
708	Brown		808	Yellow
709	Red	Master Clear from Computer	809	Yellow
710	Brown	Test Return to Computer	810	Orange
711	Red	Sync Return to Computer	811	Orange
712	Brown	Data Accepted to Computer	812	Red
712		Shield	813	

TABLE B2 - I/O CABLE AND CONNECTOR PARTS LIST

PART DESCRIPTION	AMP PART NUMBER
(1) 104 Pin Contact Male Block	201036-1
(2) 104 Pin Contact Female Block	201037-1
(3) Corner Guide Socket for 104 Contact Block	201047-7
(4) Corner Guide Pin for 104 Contact Block	201046-1
(5) Fixed Jack Screw, Male	200873-8
(6) Fixed Jack Screw, Female	200872-8
(7) Turnable Jack Screw, Male	200871-1
(8) Turnable Jack Screw, Female	200867-1
(9) #20 Pin (#24-20 Wire)	20034-1-ST
(10) #20 Socket (#24-20 Wire)	20031-1-ST
(11) 50 Pin Contact Male Block	200276-2
(12) 50 Pin Contact Female Block	200277-2
(13) Corner Guide Socket for 50 Contact Block	200835-5
(14) Corner Guide Pin for 50 Contact Block	200833-5
(15) 104 Pin Hood	201364-4
(16) 104 Pin Shield Assembly	201131-1
(17) Die Set for #20 Pin and Socket	69158
(18) Hand Tool for #20 Pin and Socket	45099
(19) Extraction Tool for all Contacts	305183
(20) Insertion Tool for all Contacts	380431-2

I/O Cables* - Constructed of 22 Ga. color coded wires (19 strands/wire). Cables are available with 51 or 27 twisted pairs of wires. Wire bundle is wrapped in Foil-Alum-Mylar shield and covered with vinyl jacket. Shield is provided with #22 gauge stranded drain wire. Standard lengths in stock: 10 ft. and 30 ft.

*Manufacturer - Columbia Wire and Cable Company

TABLE B3 - BLOCK TRANSFER CONTROL CONNECTOR

SIGNAL WIRE			GROUND WIRE	
PIN NO.	WIRE COLOR	FUNCTION	PIN NO.	WIRE COLOR
101	Brown	Initialize from Computer	201	Black
102	Red	Unit Connected	202	Black
103	Orange	Load/Unload	203	Black
104	Yellow	Data Here to Computer	204	Black
105	Green	Data Accepted to Computer	205	Black
106	Blue	Data Transfer Request	206	Black
107	Violet	Load Current Address	207	Black
108	Gray	Maximum Transfer Rate	208	Black
109	White		209	Black
110	Brown		210	Blk-Wht
111	Red		211	Blk-Wht
112	Orange		212	Blk-Wht
			213	Blk-Wht
301	Brown	BTC Data Here from Computer	401	White
302	Red	Gate Data Out	402	White
303	Orange	Unit Disconnected	403	White
304	Yellow	Instruction Received	404	White
305	Green	End of Record	405	White
306	Blue		406	White
307	Violet		407	White
308	Gray		408	White
309	Green		409	Blk-Wht
310	Blue		410	Blk-Wht
311	Violet		411	Blk-Wht
312	Brown		412	Blue
			413	Blue

TABLE B4 - PRIORITY INTERRUPT ASSIGNMENTS - SEL 810A

Module	Group	Level	Jack	Sig. Pin	Grd. Pin	Octal Memory Location	Remarks
1	--	--	N/A	N/A	N/A	1000	Power Fail Safe (optional)
1	--	--	N/A	N/A	N/A	1001	Stall Alarm (optional)
1	0	1	J2	103	203	1002	
1	0	2	J2	104	204	1003	
1	0	3	J2	105	205	1004	
1	0	4	J2	106	206	1005	
1	0	5	J2	107	207	1006	
1	0	6	J2	108	208	1007	
1	0	7	J2	109	209	1010	
1	0	8	J2	110	210	1011	
1	0	9	J2	111	211	1012	
1	0	10	J2	112	212	1013	
1	0	11	J2	301	401	1014	
1	0	12	J2	302	402	1015	
1	1	1	J1	503	603	1016	Standard Interrupt
1	1	2	J1	504	604	1017	Standard Interrupt
2	1	3	J7	101	201	1020	
2	1	4	J7	102	202	1021	
2	1	5	J7	103	203	1022	
2	1	6	J7	104	204	1023	
2	1	7	J7	105	205	1024	
2	1	8	J7	106	206	1025	
2	1	9	J7	107	207	1026	
2	1	10	J7	108	208	1027	
2	1	11	J7	109	209	1030	
2	1	12	J7	110	210	1031	
2	2	1	J7	111	211	1032	
2	2	2	J7	112	212	1033	
2	2	3	J7	301	401	1034	
2	2	4	J7	302	402	1035	
2	2	5	J7	303	403	1036	
2	2	6	J7	304	404	1037	
3	2	7	J8	101	201	1040	
3	2	8	J8	102	202	1041	
3	2	9	J8	103	203	1042	
3	2	10	J8	104	204	1043	
3	2	11	J8	105	205	1044	
3	2	12	J8	106	206	1045	

NOTE: Continued on next page.

TABLE B4 - PRIORITY INTERRUPT ASSIGNMENTS - SEL 810A CONT'D

Module	Group	Level	Jack	Sig. Pin	Grd. Pin	Octal Memory Location	Remarks
3	3	1	J8	107	207	1046	
3	3	2	J8	108	208	1047	
3	3	3	J8	109	209	1050	
3	3	4	J8	110	210	1051	
3	3	5	J8	111	211	1052	
3	3	6	J8	112	212	1053	
3	3	7	J8	301	401	1054	
3	3	8	J8	302	402	1055	
3	3	9	J8	303	403	1056	
3	3	10	J8	304	404	1057	
4	3	11	J9	101	201	1100	
4	3	12	J9	102	202	1101	
4	4	1	J9	103	203	1102	
4	4	2	J9	104	204	1103	
4	4	3	J9	105	205	1104	
4	4	4	J9	106	206	1105	
4	4	5	J9	107	207	1106	
4	4	6	J9	108	208	1107	
4	4	7	J9	109	209	1110	
4	4	8	J9	110	210	1111	
4	4	9	J9	111	211	1112	
4	4	10	J9	112	212	1113	
4	4	11	J9	301	401	1114	
4	4	12	J9	302	402	1115	
4	5	1	J9	303	403	1116	
4	5	2	J9	304	404	1117	

TABLE B5 - INTERRUPT CONNECTOR

SIGNAL WIRE			GROUND WIRE	
PIN NO.	WIRE COLOR	INTERRUPT LEVEL	PIN NO.	WIRE COLOR
101	Brown	1 (2nd, 3rd and 4th Connectors only)	201	Black
102	Red	2 (2nd, 3rd and 4th Connectors only)	202	Black
103	Orange	3	203	Black
104	Yellow	4	204	Black
105	Green	5	205	Black
106	Blue	6	206	Black
107	Violet	7	207	Black
108	Gray	8	208	Black
109	White	9	209	Black
110	Brown	10	210	Blk-Wht
111	Red	11	211	Blk-Wht
112	Orange	12	212	Blk-Wht
			213	Blk-Wht
301	Brown	13	401	White
302	Red	14	402	White
303	Orange	15	403	White
304	Yellow	16	404	White
305	Green		405	White
306	Blue		406	White
307	Violet		407	White
308	Gray		408	White
309	Green		409	Blk-Wht
310	Blue		410	Blk-Wht
311	Violet		411	Blk-Wht
312	Brown		412	Blue
			413	Blue