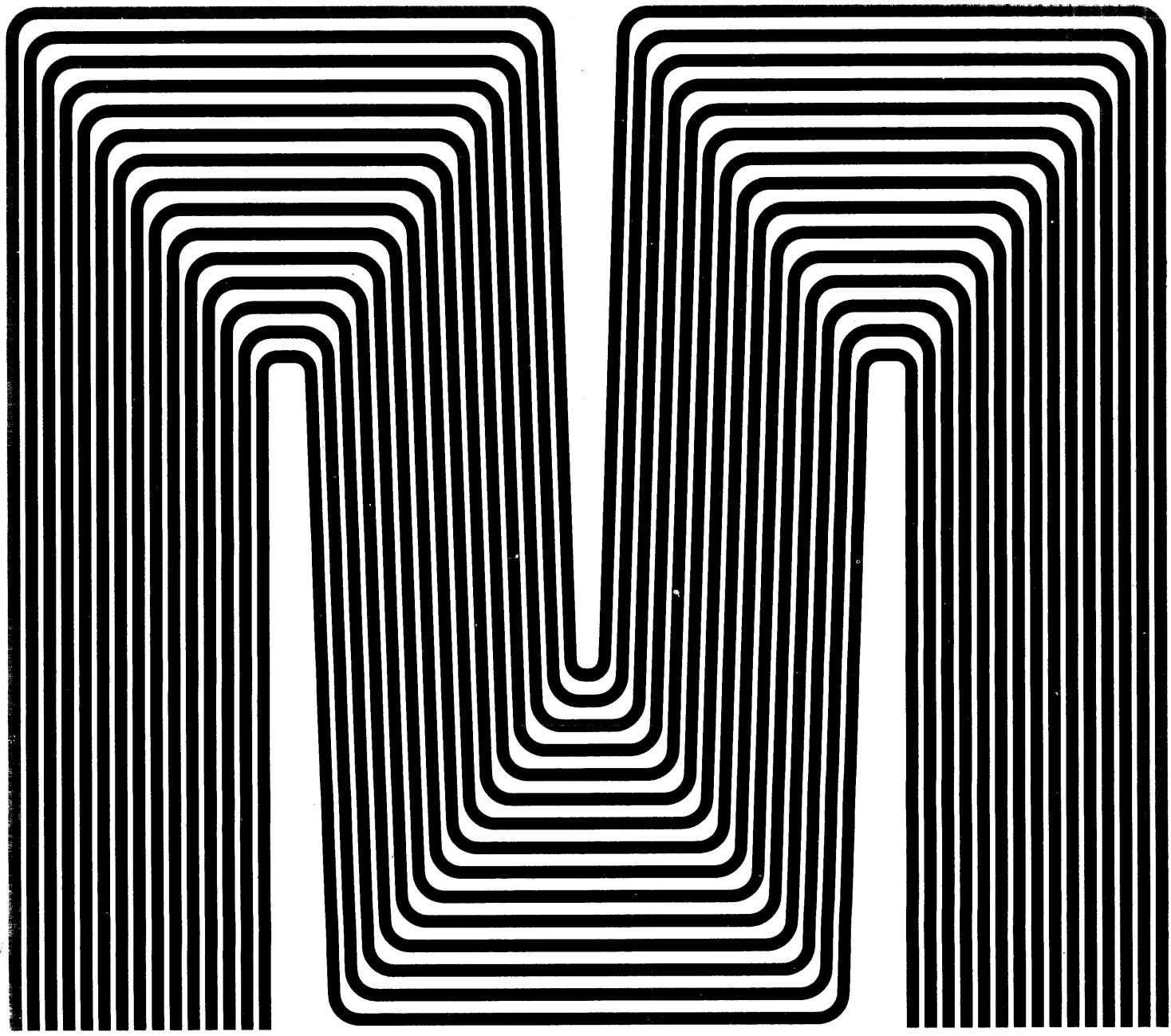
Microdata

Micro 400 Computer **Reference Manual**



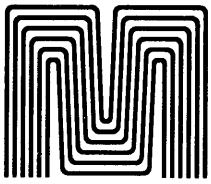
MICRO 400 COMPUTER

REFERENCE MANUAL

71 - 2 - 0400 - 001

March 1971

Microdata



Microdata Corporation
644 East Young Street
Santa Ana, California 92705

Printed in USA.

TABLE OF CONTENTS

TABLE OF CONTENTS (continued)

INTRODUCTION

1.0 The MICRO 400 is a very low cost, high performance general purpose processor designed to fill the gap between special purpose digital hardware and the mini computer. Basic design considerations were given to optimizing the processor for application to such tasks as peripheral control, data communications, instrumentation, process control and automatic test. With powerful computing ability, easy interfacing and modular design, the MICRO 400 is truly a systems component. The central processor is designed on one board 12-1/2 inches by 18 inches and is interconnected to all other systems components via the unique MICRObus concept using an easily-handled, easily-constructed flat cable rather than a back plane.

The MICRO 400 offers a number of additional features not available on other computers. The MICRObus interconnects all systems components including central processor, memory and special peripheral controllers by means of a simple flat wire cable. The standard Peripheral Data Bus is designed to take full advantage of a wide range of peripheral

device controllers. Base Register addressing is an optional feature not available on any other mini computer. This feature allows offset addressing (operand plus offset) to 65,536 words of memory and provides total program relocatability on both the object level as well as the absolute level.

Memory is expandable from a basic 1,024 words of core memory to 65,536 words of core memory, Also, read only memory is available as an option and directly replaces core memory for implementation.

The ability of the MICRO 400 to adapt to the host system is an excellent criteria for determining true systems compatibility. The design philosophy behind the MICRO 400 is not only to provide the capabilities needed in a mini computer but to allow the user to choose the particular capability needed for his application. If a feature is needed initially it can be provided as a low cost factory option. If the need is uncertain the same feature can easily be added in the field as an option.

1.1 SYSTEM FEATURES

Some of the standard features available with the MICRO 400 central processor include:

- Multi-Accumulator Arithmetic
- 105 Program Instructions
- Up to 32 Peripheral Devices
- Direct and Index Addressing
- Memory Directly Addressable in 4096 pages
- Two's Complement Arithmetic
- Local Operator's Control Panel (removable)
- Maintenance Programs
- Utility Debug Programs
- Full Line of Peripherals and Systems Interfaces
- Microbus Interconnection
- Programmed I/O bus
- Priority Interrupts

Some of the optional features available with the MICRO 400 central processor include:

- Base Register Addressing to 65K Words
- Automatic I/O
- Power Fail Protection from Run or Halt

2.0 CENTRAL PROCESSOR

The central processor is the control unit for the entire system. It governs all peripheral input/output equipment, performs all the arithmetic, logical and data handling operations, and sequences the program. It is connected to the memory by the MICRObus and to the peripheral equipment by either the MICRObus or the Peripheral Data bus. The processor handles words of 8-bits which are stored in memory with a maximum capacity of 65,536 words in memory. Words in memory are used by the central processor as instructions in a program, as addresses, or as operands, i,e., data for the program. The processor performs a program by executing instructions retrieved from consecutive memory locations as counted by the 12-bit program counter. At the end of each memory cycle the program counter is incremented by one so that the next byte is normally taken from the next consecutive location in memory. Sequential program flow is altered by changing the contents of the program counter, either by incrementing two extra times in a skip instructions, or by replacing its contents with the values specified by a jump instruction. The other internal registers in the

central processor include the two accumulators, (A register and the B register). Data can be moved in either direction between any memory location and either accumulator. All arithmetic and logical operations are performed on the operands contained in the accumulators with the result appearing in one of those accumulators. Associated with the accumulators is the overflow flip-flop which indicates when an arithmetic overflow occurs out of the most significant bit during an arithmetic instruction. Also associated with the accumulators is the Link flip-flop which indicates that a carry occurs out of the most significant register position. Transfers to and from peripheral devices are also made through the accumulators.

On the MICRO 400 front panel is a set of data switches through which the operator or maintenance man can supply words and addresses to the program. The front panel also has a number of control switches that allow the operator to start and stop the central processor, to deposit the contents of the data switches in any memory location or accumulator, and to display the contents of any location or accumulator in the data indicators. The 12 indicators on the front panel can display the contents of the

program counter, the accumulators, or the memory data register. The remaining lights display internal control conditions that are useful in program debugging. A complete description of the operator's control panel is contained in Section V.

2.1 INSTRUCTIONS

Instructions that move data to and from memory or the peripherals, use one of the accumulators as a source or destination while addressing a memory location or an input/output device as the other destination or source. The arithmetic and logical instructions do not have to reference memory, they simply address the two accumulators, either or both of which may supply operands, and one of which may receive the result. Therefore the memory is used for storage of the program and permanent data while all calculations are carried out in the accumulators. This reduces considerably the amount of data movement as compared with a single accumulator system, and saves instructions.

The input/output hardware allows addressing of up to 32 peripheral devices. A single instruction can transfer a word between an accumulator and a device and at the same time control the device operation. Included in the I/O system are facilities for program interrupt and for Direct Memory high speed data transfers. The program interrupt facilitates processor control of the peripheral equipment by allowing any device to interrupt the normal program execution and receive service on a priority basis. The processor acknowledges an interrupt request by responding to an address in memory that is provided by the interrupting device. This address in memory will

contain a Jump and Mark command that sequences the program to the appropriate interrupt service routine. The Jump and Mark command also will store the contents of the Program Counter in the interrupt service routine for exit back to the interrupted program.

Devices requiring high speed data flow, such as magnetic tape or discs, can gain direct access to memory through the Direct Memory Access Channel without requiring the execution of any program instructions or processor control. The program simply pauses while the memory access is being made and then resumes normal program execution.

Throughout this manual all numbers representing instruction words, register contents, memory addresses and any numbers appearing in program examples are hexadecimal unless otherwise specified. Computer words are represented by two hexadecimal digits. The use of numbers in the text to specify quantities of objects such as words or locations, to count steps in an operation, bit positions etc. employs standard decimal notation. Where necessary for clarity, base suffixes will be used as in the example shown.

$175_{10} = AF_{16}$

$1AB = Hex 1AB

As described in the introduction, the MICRO 400 functions by executing instructions retrieved from consecutive memory locations as counted by the 12-bit program counter. The instructions to be executed have many variations, but can generally be classified as follows:

INSTRUCTION TYPE	ACTION
Load/Store Accumulators	Data is transferred from one memory location or one accumulator. Either may be the source of the data, the other is the destination.
Modify Memory	One memory location is affected. The contents of this memory location are either shifted, rotated/not rotated, and interogated for various skip conditions. During the shift, new data may be shifted into the memory location from various sources.
Jump, Jump and Mark	The contents of the program counter are altered to affect a change in program sequencing. The Jump and Mark accomplishes the same thing however it writes the old contents of the program counter into the locations specified by the altered program counter.

INSTRUCTION TYPE	ACTION
Arithmetic, Logical	The two accumulators are affected. One or both may contain operands. For arithmetic, both are operand sources and one is destination. For logical, only one accumulator is specified.
Input/Output	Similar to Transfer Data however memory is not involved. One accumulator and one I/O device are specified. Data is transferred in either direction, and device control can be initiated, or device status can be tested for selected conditions.
Control	Similar to arithmetic/logical except no accumulator is involved.

Instructions in the first three classes are considered Memory Reference instructions. These instructions must address memory to fetch or store operands, or store the contents of the program counter.

The load/store accumulator commands contain the complete 12-bit operand address necessary to load or store accumulator data anywhere in a 4096 word page. Indexing is also available to modify the effective operand address. In the latter case, the instruction contains an 8-bit literal bias or displacement which is added to the

address value in the 16-bit index register to extend the operand addressing to 65,536 words of memory.

The Modify Memory Commands contain an 8-bit address to directly address the first 256 words of the effective 4096 word page.

The Jump and Jump and Mark Commands are divided into 4 categories: Jump in Current Page (JPC), Jump in Extended Page (JPE), Jump and Mark in Current Page (JMC) and Jump and Mark in Extended Page (JME). The page size is 4096 words, page 0 is the low order 4096 words of memory and page 1 is the second 4096 words of an 8192 word memory system. The Jump Current commands are used to alter the contents of the program counter without altering the page pointer, while the Jump Extended commands will switch from page 0 to page 1 or from page 1 to page 0.

The Jump and Mark command stores the contents of the program counter in the jump address and jump address plus one. In addition, the upper 4 bits of the two words of data stored as the Mark address are stored as a Jump Command so the return address becomes a stand-

alone subroutine exit. Either the JPC or JPE op codes will be stored depending on whether the Jump and Mark command executed was a JMC or JME.

2.2 INSTRUCTION FORMAT

There are eight basic formats for instruction words in the MICRO 400. Six of these formats are two byte commands and two are one byte. Byte one of the command will always contain the operation code and is the first byte fetched from memory during execution.

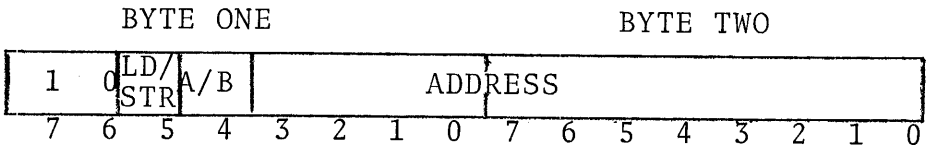
2.2.1 Memory Reference

LDA ADDR Load A from Memory

LDB ADDR Load B from Memory

STA ADDR Store A into Memory

STB ADDR Store B into Memory



Where: ADDR is a 12 bit address of the operand in a 4096 word page.

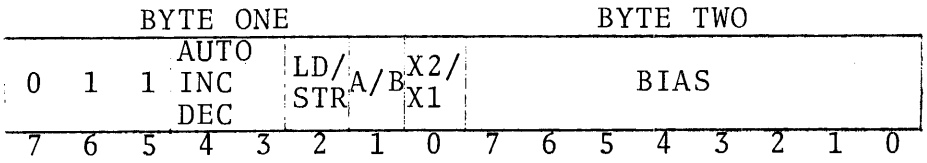
Bits 4 and 5 of the first byte determine source/destination for the command. Bit 5=0 indicates memory source, bit 5=1 indicates accumulator source.

Bit 4=0 indicates accumulator A, and bit 4=1 indicates accumulator B.

Bits 0 through 3 of byte one and bits 1 through 7 of byte two form the memory address with bit zero of the second byte being the least significant address bit.

2.2.2 Memory Reference Indexed

LDA	BIAS,X(,M)	Load A from Memory Indexed
LDB	BIAS,X(,M)	Load B from Memory Indexed
STA	BIAS,X(,M)	Store A into Memory Indexed
STB	BIAS,X(,M)	Store B into Memory Indexed



Where: BIAS is an 8-bit value (0-255) which is added to the selected Index register to form the operand address.

: X selects index register 1 or 2.

: M is an optional index modifier field

I - increment index by one

D - decrement index by one

blank - no change to index

Bits three and four of byte one decode as follows:

00 = no auto increment or decrement

01 = auto decrement

10 = auto increment

Bits one and two of byte one decode as follows:

00 = load accumulator A

01 = load accumulator B

10 = store accumulator A

11 = store accumulator B

Bit zero of byte one decodes the use of index register one (0) or the optional index register two (1).

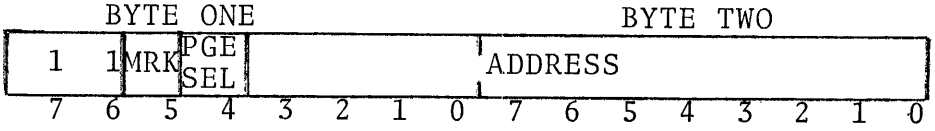
2.2.3 Jump and Jump Mark Commands

JPC ADDR Jump within current 4096 page

JPE ADDR Jump within extended 4096 page

JMC ADDR Jump Mark within current 4096 page

JME ADDR Jump Mark within extended 4096 page



Where: ADDR is a 12-bit address.

Bits 4 and 5 of byte one decode as follows:

00 = Jump in current page

01 = Jump into extended page and Switch page select

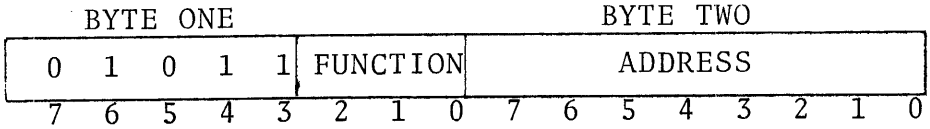
10 = Jump and Mark in Current Page

11 = Jump and Mark into extended page and Switch page select

2.2.4 Memory Test/Modify Instructions

Skip	SER ADDR	Skip if memory is Even, Rotate memory*
	SOR ADDR	Skip if memory is Odd, Rotate memory*
	SPM ADDR	Skip if Positive Memory
	SMM ADDR	Skip if Minus Memory
Shift Memory	RCS ADDR	Right shift one place, copy sense flip-flop
	RCZ ADDR	Right shift one place, Copy Zero
	RCO ADDR	Right shift one place, Copy One
	RCA ADDR	Right shift one place, Copy least significant bit of the A Accumulator

Where: ADDR is an 8-bit address of the first 256 words of the current 4096 word page.



*Rotation of memory will take place regardless of whether memory is odd or even.

Bits zero through two of byte one decode the memory modification that occurs as follows:

210

000 = Skip if even, rotate memory right one

001 = Skip if odd, rotate memory right one

010 = Skip if positive, do not rotate

011 = Skip if negative, do not rotate

100 = Right shift memory one, copy sense flip-flop

101 = Right shift memory one, copy zero

110 = Right shift memory one, copy one

111 = Right shift memory one, copy LS bit of A

2.2.5 General Skip Instructions

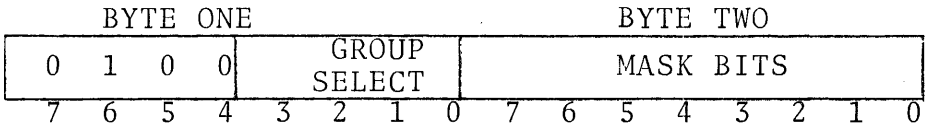
SNZ LIT Skip if all of the selected N-class conditions are Zero

SNO LIT Skip if any of the selected N-class conditions is One

SMZ LIT Skip if all of the selected M-class conditions are Zero

SMO LIT Skip if any of the selected M-class conditions is One

Where: LIT is an 8-bit literal mask for the N or M class conditions.



Bits zero through 3 of byte one are decoded as follows:

1000-Examine group (N), skip if all condition(s) are zero

1001-Examine group (N), skip if any condition is true

1010-Examine group (M), skip if all condition(s) are zero

1011-Examine group (M), skip if any condition is true

N-Class Conditions

Bit	Source
0	Most Significant Bit of A
1	Least Significant Bit of A
2	A Register Not Zero
3	Overflow
4	Most Significant Bit of B
5	Least Significant Bit of B
6	B Register Not Zero
7	Link

M-Class Conditions

Bit	Source
0	Interrupt Enabled
1	Sense Response
2	Teletype Read Ready
3	Teletype Write Ready
4	X1 Register Low Order 8 Bits Equal Zero*
5	X2 Register Low Order 8 Bits Equal Zero*
6	Sense Switch one Not Set
7	Sense Switch two Not Set

*Note - See Instructions SI1, SD1, SI2, SD2 for full definition.

2.2.6 Input/Output Instruction Format

Input/output instruction format is described below:

In the format box, Function (F) and Device address (D) are referred to only as F and D. The programmer must

substitute a valid three bit code for F and a five bit code for D. Also, in the case of Data Output and Data Input, the programmer must specify which accumulator he is requesting. A one in byte one, bit two will select accumulator B; a zero, accumulator A.

SEN F,D Sense

EXC F,D External Control

DIA F,D Data Input into the A Register

DOA F,D Data Output from the A Register

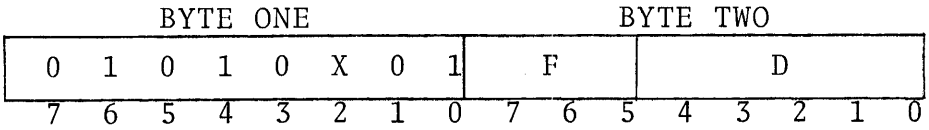
DIB F,D Data Input into the B Register

DOB F,D Data Output from the B Register

Where: F is a 3 bit function code

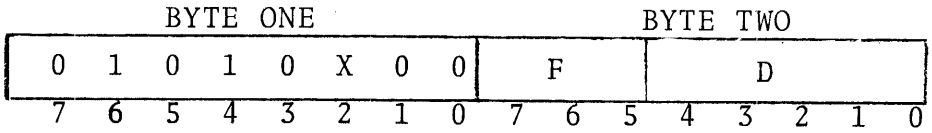
: D is a 5 bit device address

EXC F,D EXTERNAL CONTROL



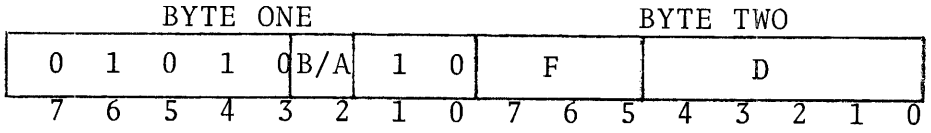
Perform the control function specified by F in device D.

SEN F,D SENSE DEVICE STATUS



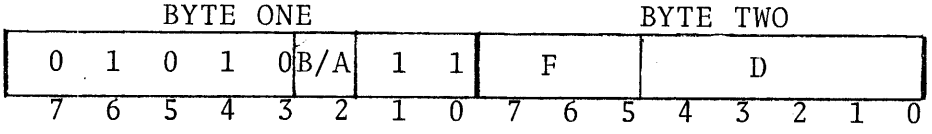
Set the central processor sense flip-flop according to the state of F in device D.

(DIA F,D) (DIB F,D) DATA TRANSFER INTO ACCUMULATOR A OR B



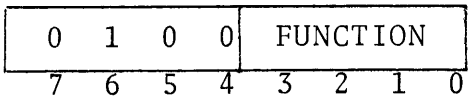
Bring data into accumulator A or B from device D. F may be used to control the type of data from device D such as status word, true data, or special data. F may also be used to specify action by the device.

(DOA F,D) (DOB F,D) DATA TRANSFER OUT OF ACCUMULATOR A OR B



Transfer the contents of accumulator A or B to the device specified by D. Special control may be exercised in the device by F.

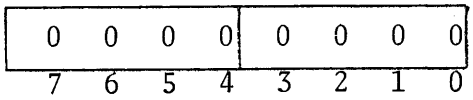
2.2.7 Control Instruction Format



Bits zero through three provide the following specific functions:

Mnemonic	3210
SOF	0000 = Set Overflow Flip-flop
ROF	0001 = Reset Overflow Flip-flop
EIN	0010 = Enable INterrupts
DIN	0011 = Disable INterrupts
NOP	0101 = No OPeration
SSF	0110 = Set Sense Flip-flop
RSF	0111 = Reset Sense Flop-flop

2.2.8 Halt

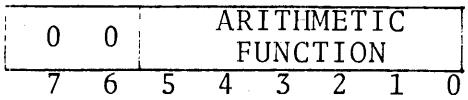


Halts the computer. The program counter register (P) will contain the address of the next byte after the HALT, STEP mode will be selected.

2.2.9 Arithmetic/Logical Instructions

The one byte arithmetic/logical instructions are executed in the accumulators. One or both of which are operand sources and one of which is a destination. Execution time is one cycle (1.6 microseconds).

The link and overflow flip-flops may be affected by the Arithmetic/Logical instructions.



MICRO 400 INSTRUCTION LISTING

ARITHMETIC

	MNEMONIC	OP CODE	DESCRIPTION	* LINK/ OVERFLOW
ADD	ABA	12	ADD B TO A	A
	AAB	09	ADD A TO B	A
SUBTRACT	SBA	14	SUBTRACT B FROM A TO A	A
	SAA	22	SUBTRACT A FROM B TO A	A
AND	NBA	1E	AND B WITH A TO A	R
	NAB	3B	AND A WITH B TO B	R
INCLUSIVE OR	OBA	29	INCLUSIVE OR B WITH A TO A	R
	OAB	0D	INCLUSIVE OR A WITH B TO B	R
EXCLUSIVE OR	XBA	2D	EXCLUSIVE OR B WITH A TO A	R
LEFT SHIFT	LSA	11	LEFT SHIFT A ONE BIT, INSERT 0	A
	LSB	0A	LEFT SHIFT B ONE BIT, INSERT 0	A
RIGHT SHIFT	RSA	1F	RIGHT SHIFT A ONE BIT, INSERT 0	-
	RSB	3F	RIGHT SHIFT B ONE BIT, INSERT 0	-
	RRA	0F	RIGHT ROTATE A ONE BIT	-
	RRB	2F	RIGHT ROTATE B ONE BIT	-
COMPLEMENT	OCA	20	ONES COMPLEMENT A	S
	OCB	04	ONES COMPLEMENT B	S
	TCA	26	TWOS COMPLEMENT A	A
	TCB	34	TWOS COMPLEMENT B	A
INCREMENT	IAA	16	INCREMENT A BY 1 TO A	A
	IAB	31	INCREMENT A BY 1 TO B	A
	IBB	32	INCREMENT B BY 1 TO B	A
DECREMENT	DAA	10	DECREMENT A BY 1 TO A	A
	DAB	01	DECREMENT A BY 1 TO B	A
	DBB	02	DECREMENT B BY 1 TO B	A
COPY	CAB	39	COPY A INTO B	R
	CBA	2B	COPY B INTO A	R
CLEAR	CLA	21	CLEAR A TO ZERO	S
	CLB	06	CLEAR B TO ZERO	S
SET	POA	17	POSITIVE ONE TO A	R
	POB	07	POSITIVE ONE TO B	R
	MOA	18	MINUS ONE TO A	R
	MOB	08	MINUS ONE TO B	R
LINK	CLL	0E	CLEAR LINK BIT TO ZERO	R
	LCB	30	SET LINK BIT AND CLEAR B	S

INSTRUCTION FORMAT: XXX

* NOTE - AFFECT ON LINK/OVERFLOW BITS
S - LINK ALWAYS SET
R - LINK ALWAYS RESET
A - LINK AFFECTED (MAY BE SET OR RESET) AND OVERFLOW MAY BE SET

2.3 INPUT/OUTPUT INSTRUCTION IMPLEMENTATION

The input/output instructions control all transfers of data to and from the peripheral equipment, and also perform various operations within the central processor. Instructions in this class are all two byte instructions and the format is discussed in this section. Bits three through seven of byte one are the operation code while bits zero through two are the specific input/output operation code (transfer data in, transfer data out, external control, and sense). For this discussion, "input of data" always refers to the transfer of data from a peripheral device into the central processor while an "output" command transfers data out of the central processor to a peripheral device.

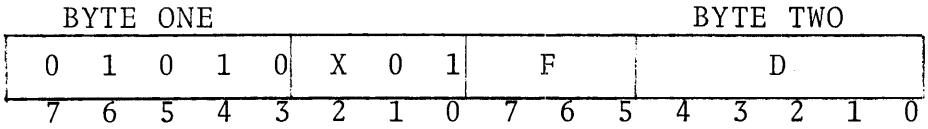
Bits five through seven of byte two of the input/output instruction contain special control bits that will be discussed later in this section. Bits zero through four contain device address bits that select the device that is to respond to the instruction. This format thus allows for 32 codes to be used for device selection. A table in Appendix B lists all devices for which codes have been assigned. Every device controller has a five bit device selection network that decodes the device address bits so that only the addressed device responds to the signals sent by the processor over the input/output bus.

2.4 INPUT/OUTPUT INSTRUCTION FORMAT

Input/output instruction format is described below:

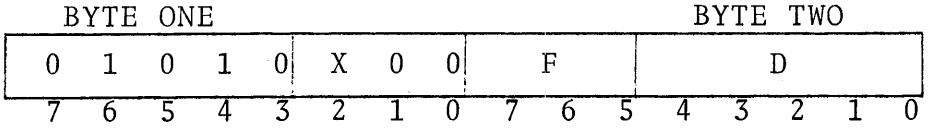
In the format box, function (F) and device address (D) are referred to only as F and D. The programmer must substitute a valid three bit code for F and a five bit code for D. Also, in the case of Data Output and Data Input, the programmer must specify which accumulator he is requesting. A one in byte one bit two will select accumulator B, a zero, accumulator A.

EXC F,D EXTERNAL CONTROL



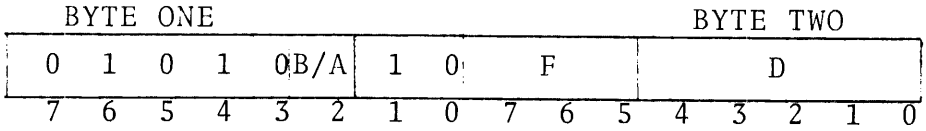
Perform the control function specified by F in device D.

SEN F,D SENSE DEVICE STATUS



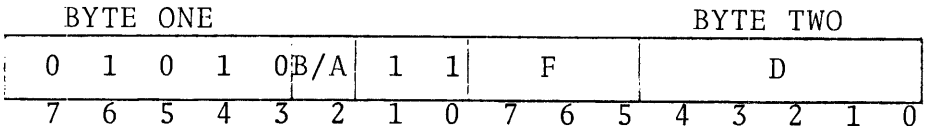
Set the central processor sense flip-flop according to the state of F in device D.

DIX F,D DATA TRANSFER INTO ACCUMULATOR A OR B



Bring data into accumulator A or B from device D. F may be used to control the type of data from device D such as status word, true data, or special data. F may also be used to specify action by the device

DOX F,D DATA TRANSFER OUT OF ACCUMULATOR A OR B



Transfer the contents of accumulator A or B to the device specified by D. Special control may be exercised in the device by F.

To place a device in operation, (assume an output of data is to be accomplished) the central processor first will test for a device ready condition. This will be done by either the Sense command or an input of status word from the device. When a true ready response is received, the first character of data will be transferred to the device controller. This will typically cause the device controller to write the data on the peripheral device (paper tape for example). At the same time, as soon as the device controller has accepted the data from the central processor, it will set the "busy" or "not ready" condition until all peripheral device action has taken place. At that time, the "busy" condition will be cleared and a "ready" signal made available for the next transfer of data.

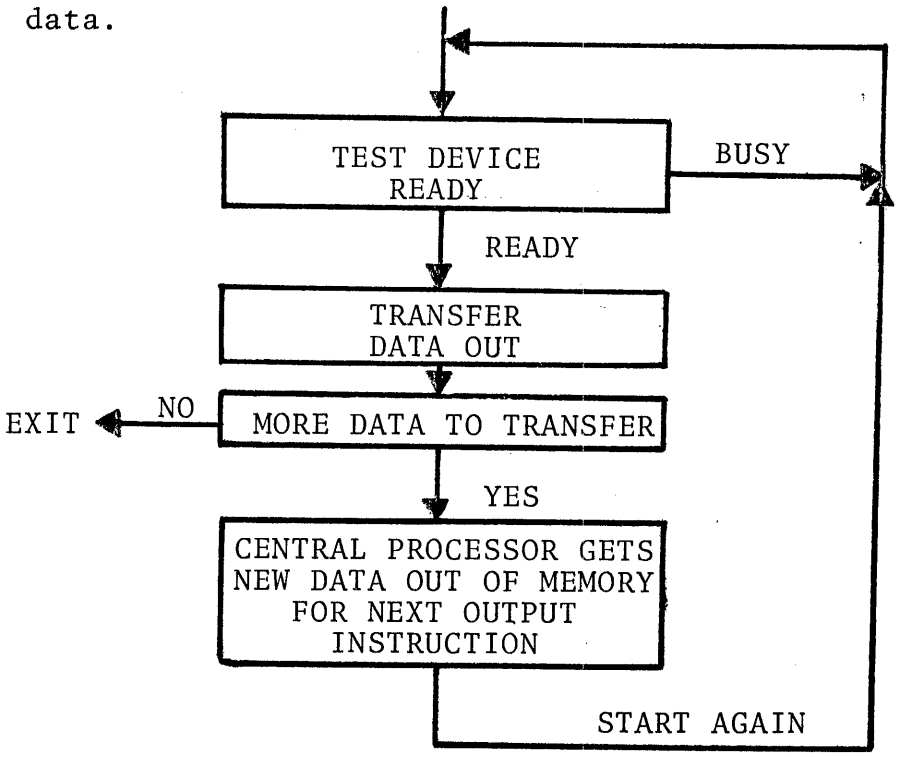


FIGURE 2-1

The above procedure is of course the most inefficient for optimum use of the central processor. To optimize program execution, an interrupt would be used to allow the central processor to execute instructions in the program instead of hanging in a small program loop waiting for the peripheral device to become ready. This would result in the following flow chart. Interrupts will be discussed in detail later in this section.

When all the data has been transferred, the program will generally inform the device in some manner; possibly by not allowing any further interrupts until new data is available for output.

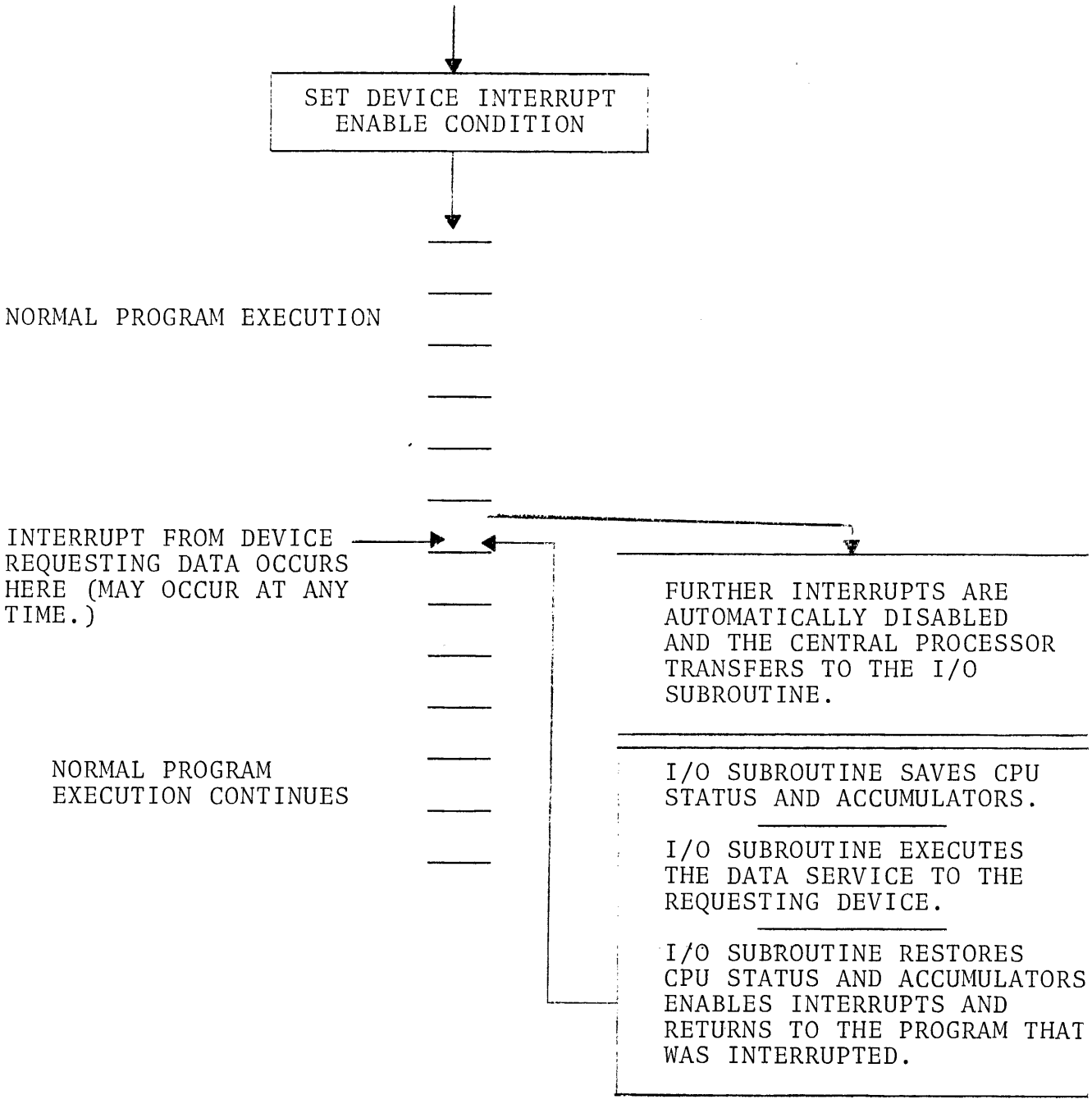

SET DEVICE INTERRUPT ENABLE CONDITION
NORMAL PROGRAM EXECUTION
INTERRUPT FROM DEVICE REQUESTING DATA OCCURS HERE (MAY OCCUR AT ANY TIME.)
NORMAL PROGRAM EXECUTION CONTINUES
FURTHER INTERRUPTS ARE AUTOMATICALLY DISABLED AND THE CENTRAL PROCESSOR TRANSFERS TO THE I/O SUBROUTINE.
I/O SUBROUTINE SAVES CPU STATUS AND ACCUMULATORS.
I/O SUBROUTINE EXECUTES THE DATA SERVICE TO THE REQUESTING DEVICE.
I/O SUBROUTINE RESTORES CPU STATUS AND ACCUMULATORS ENABLES INTERRUPTS AND RETURNS TO THE PROGRAM THAT WAS INTERRUPTED.

FIGURE 2-2

3.0 MEMORY

From the programmer's standpoint, all of memory is a set of contiguous 8-bit word length locations whose addresses range from hexadecimal $0000 to a maximum of $FFFF depending on the memory configuration for a particular installation. An address of $FFFF represents the maximum memory size of $65,536_{10}$ words of storage. However, the memory is actually made up of a number of memory modules each having a capacity of 1024, 4096, or 8192 (decimal) words. The configuration can also contain read-only memory modules for the storage of unalterable programs or data. From the programmer's viewpoint read only memory differs only in that its contents cannot be altered.

4.0 PROGRAM INTERRUPT

The devices on the I/O bus are assigned priority for control of external interrupts and for Automatic I/O operations. The priority is achieved by the physical location of the device controller on the I/O bus. That is to say the first device on the I/O bus has the highest priority and the last unit on the I/O bus the lowest. This is achieved by the signal PROT that passes priority down through the bus from one controller to the other. Before a controller can make an interrupt request (IREQ) it must receive priority from the next higher controller in the chain. A controller never passes priority along to a lower priority controller while making a request.

Interrupt requests from peripheral controllers are carried on the Interrupt Request Line (IREQ) of the I/O Bus. The central processor will recognize the IREQ at the end of the current instruction in process and will respond with the Interrupt Acknowledge signal (IACK). During the period IACK, the peripheral controller will drive the I/O bus lines EB00 through EB07 with its device address shifted left one to provide the interrupt response address. The shift in the address is necessary to allow

the peripheral device to select an even numbered address to provide as the interrupt response address. For example, the following chart gives the interrupt response address available to peripheral device number OF.

DEVICE ADDRESS	RESPONSE ADDRESS
OF(0000 1111)	1E(0001 1110)

It must be emphasized that the Interrupt REQuest (IREQ) occurs in a completely asynchronous fashion with regard to the sequencing of a program in the MICRO 400. It can occur at any time during an instruction but must wait until the end of the instruction to be recognized. This dictates that the longest wait for interrupt acknowledge will be the time necessary to complete the longest instruction, the Jump and Mark command, plus queing time which is 1.2 microseconds. This command requires 4 cycle times (6.4 microseconds) to complete its execution, therefore the maximum wait will be 6.4 microseconds plus 1.2 microseconds or 7.6 microseconds. The minimum wait will be 1.2 microseconds.

In addition, Interrupt Enable must be set before the central processor can recognize any interrupts other than the Powerfail/Restart interrupts. The Interrupt Enable condition is set and reset by the one word control commands outlined in Section 2.2

EIN

0	1	0	0	0	0	1	0
7	6	5	4	3	2	1	0

ENABLE INTERRUPT FORMAT

DIN

0	1	0	0	0	0	1	1
7	6	5	4	3	2	1	0

DISABLE INTERRUPT FORMAT

In the process of recognizing an interrupt, the central processor automatically disables further interrupts. It is necessary to do an Enable Interrupt command before any further interrupts will be acknowledged.

Note that when base register addressing is being used the "EBL" (extended base to lower Base) and the "LEB" (lower Base to extended base) instructions inhibit all interrupts (including power failure) until after the following extended jump (either "JPE" or "JME"). Thus the execution time for the EBL, LEB commands (3.2usec.) and any interspersed instructions must be considered in the timing analysis for interrupt responses.

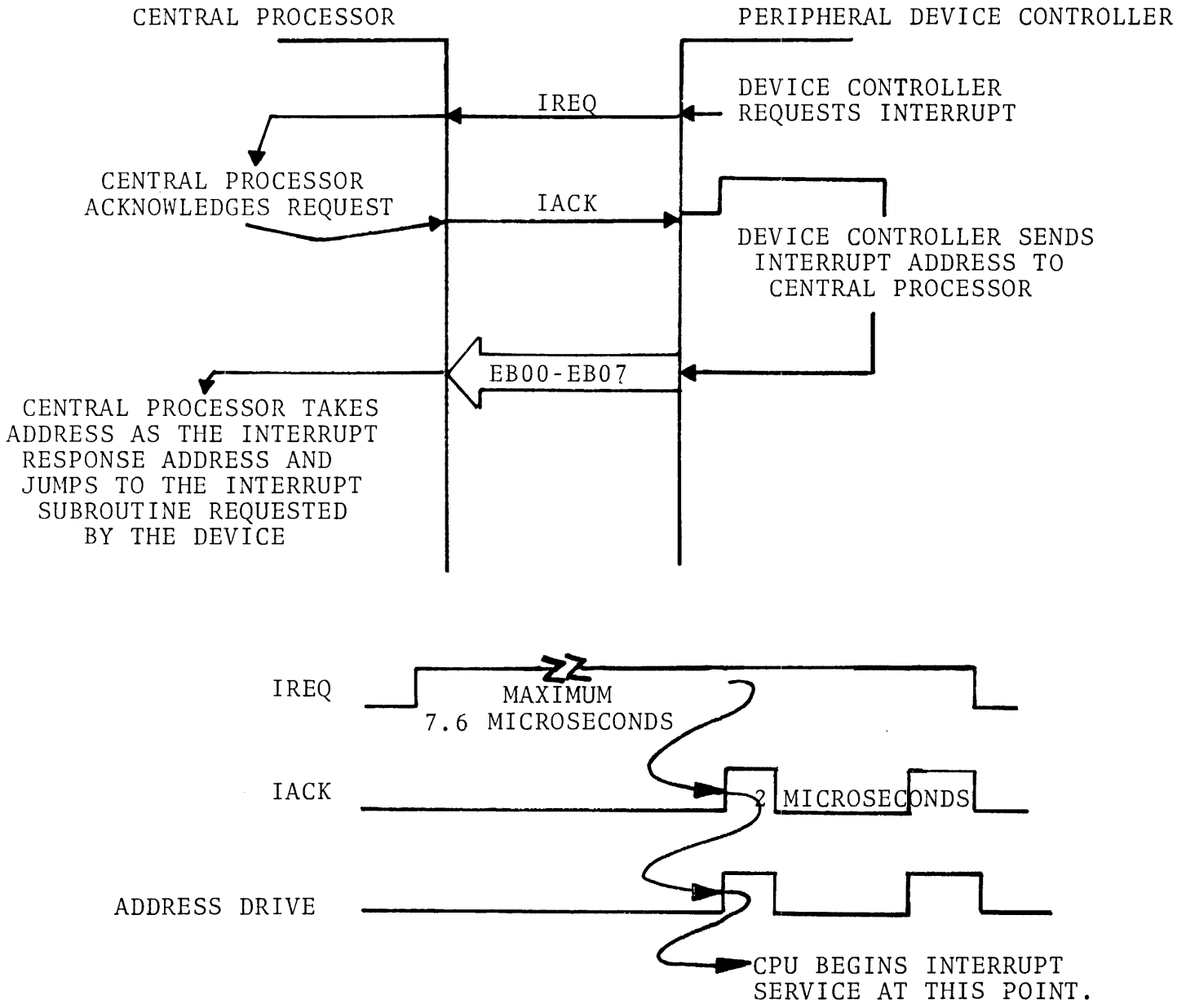


FIGURE 4-1

INTERRUPT CONTROL SIGNAL SEQUENCING

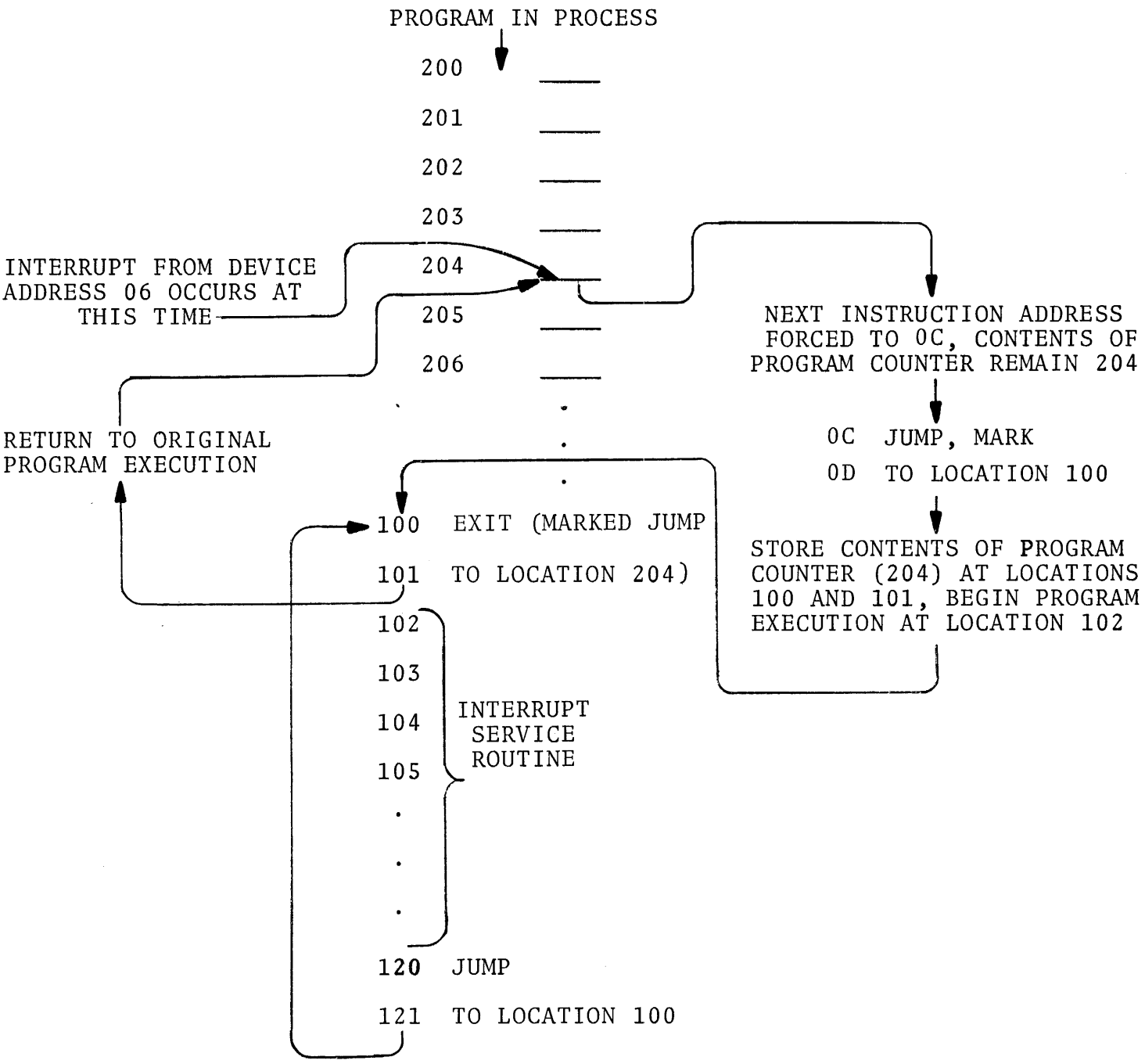


FIGURE 4-2

INTERRUPT SEQUENCING

5.0 CONTROL PANEL

The control panel is illustrated in Figure 5-1. The indicators in the upper right section display control conditions, while the indicators directly above the data switches display the accumulators, memory data or the program counter. The data switches can best be visualized as a hardware register through which the operator can supply addresses and data to the processor. The "up" position of the data switch represents a one.

The four switches on the right side of the operator's panel (EXC, STEP, INIT, RUN) are momentary contact spring return switches. They provide control signals to initiate action in the central processor.

NOTE

Since the control panel is not necessary to maintain central processor operation, it may be removed and all operator control initiated from the teletype using the TOS (teletype operating system) or from a special systems console. The auto bootstrap option would be required for startup without the front panel.

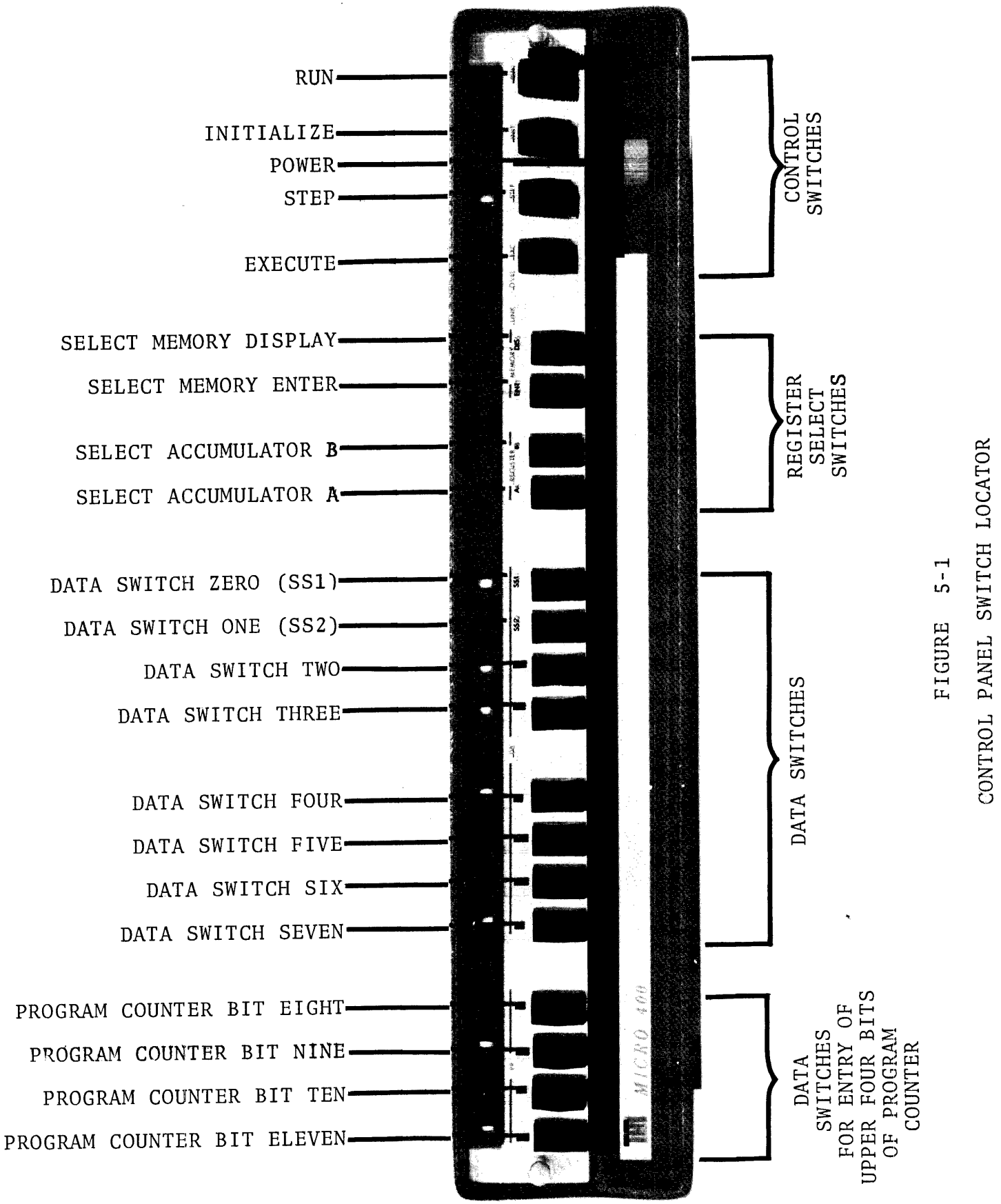


FIGURE 5-1

CONTROL PANEL SWITCH LOCATOR

FIGURE 5-1

All register and memory entry or display must be performed in the step mode. All switches except step and data entry switches zero and one, which are used as sense switches, are interlocked so they have no effect in the run mode.

INDICATORS

When any indicator is lit, the associated function is in the true (1) state. The indicators can display useful information in the run mode but most can change too rapidly to be valid. They are therefore discussed in terms of information displayed in the step mode.

There are 16 indicators on the MICRO 400 control panel. The eight DB indicators display the contents of the selected register; A, B, P, or Memory Data. The four PR indicators display the upper four bits of the P register in conjunction with the DB indicators displaying the low order eight bits. The PR indicators are only lit when P is selected for display. The control conditions OVFL (overflow), Link, Step and Run are also displayed.

5.1 REGISTER SELECT

The A, B, P and MD registers can be controlled from the control panel. The P register is displayed when the A, B, Memory Enter and Memory Display switches are all in the off (down) position. The front panel data switch contents can then be transferred to the P register by depressing the EXC switch. The A, B or Memory Data register can be selected by raising the respective switch. When the A or B register is selected, the contents of that register are displayed. When Memory ENT or Memory DIS are selected, the contents of the Memory Data register are displayed. Except for Memory DIS, the data switch contents can be transferred to the selected register by depressing the EXC switch. With Memory DIS selected, the contents of the memory cell addressed by the program counter will be displayed by depressing the EXC switch.

5.2 DATA ENTRY SWITCHES

Eight alternate action switches are used to enter data into the selected register or into memory. Four additional switches are associated with the upper 4-bits of the P register. Data Entry Switches zero and one also serve as sense switches in the RUN and instruction STEP modes.

5.3 MEMORY SELECT

The contents of memory can be displayed or altered from the front panel. The address of the memory location is

loaded into the P register. When ENT is selected and the EXC switch is depressed, the contents of P will be strobed into the memory address (MA) register and the contents of the eight data switches will be strobed into the memory data (MD) register and stored in core. When DIS is selected and the EXC switch is momentarily depressed, the contents of P are transferred to the MA register and the memory data contained in that location are strobed into the MD register and displayed on the control panel. The STEP switch increments P without executing instructions as long as either the ENT or DIS memory switch is selected, therefore when successive addresses are to be displayed or altered, the STEP switch can be used.

5.4 STEP

When the computer is in step mode and neither the memory or register select switches are selected, depressing the step switch will cause the next instruction (as addressed by P register) to be executed.

When the computer is in the run mode, depressing the Step switch will cause the computer to enter the step mode.

The Step switch will also perform an increment P if either the memory DIS or memory ENT switches are selected. Instructions are not executed in this mode.

5.5 RUN

If the memory ENT or DIS switches are not selected and the computer is in the Step mode, momentarily depressing the Run switch will cause the computer to enter the run mode.

5.6 INIT (Initialize)

The INIT is functional only in the step mode and is used to initialize the computer and external devices.

5.7 POWER ON/OFF INDICATOR AND SWITCH

The power on/off switch controls the relay that switches AC to the power supply and AC sequencer. The power on indicator is lighted by the same 12 VAC that is used to switch the relay when power is available to the CPU and memory (+5, ±12 volts).

SENSE SWITCHES SS1 AND SS2

The SS1, SS2 and STEP switches are the only switches active in the RUN mode. In the STEP mode, during data entry operations, the SS1 and SS2 switches are data switches bit 0 and 1. In stepping through a program or in the RUN mode, they are used as sense switches, and may be interrogated by the SKIP commands to cause breakpoint or branching of a control program. If the control panel has been removed, the SS1, SS2 response will always be not set.

6.0 TELETYPE

Three teletypewriter models are available for use with the MICRO 400, the ASR 33 TY, KSR 33 and KSR 35, all of which are capable of speeds up to ten characters per second. The program can type out characters and can read in the characters produced when keys are struck at the keyboard. With an ASR the program can also punch characters from paper tape.

The teletype is really two distinct devices in that its input and output functions are separated. When the program does an output command and places a character in the teletype controller output buffer, it causes the teletype to print the character. Striking a key places the code for the associated character in the controller input buffer where it can be retrieved by the program, but it does nothing at the teletype unless the program sends the code back as an output.

6.1 OPERATION

A KSR is actually two independent devices, keyboard and printer, which can be operated simultaneously. An ASR is really four devices, keyboard, printer, reader and punch, which can be operated in various combinations. Power must be turned on by the operator. On the 33 and 35 the switch is beside the keyboard and is labeled LINE/OFF/LOCAL or ON/OFF and has an unmarked third position opposite ON. When this switch is set to

LOCAL or the unmarked position, power is on but the machine is off line and can be used like a typewriter. Moreover, in an ASR turning on the punch allows the operator to punch a tape from the keyboard, and running the reader allows a tape to control the printer (if the punch is also on, it duplicates the tape).

Turning the switch to LINE or ON connects the unit to the computer and separates its input and output functions. Thus any information transmitted to the computer from the keyboard affects the printer only insofar as the computer sends it back. Turning on the reader places it under program control, and turning on the punch causes it to punch whatever is sent to the printer by the computer.

The only control on the reader is a 3-position switch. When the switch is in the FREE position, the tape can be moved by hand freely through the reader mechanism. The STOP position engages the reader clutch so the tape is stationary but the reader is still off. Turning the switch to START causes the reader to read the tape if the unit is in local, but places it under program control if on line.

The operator controls the punch by means of four push-buttons. The two on the right turn the punch on and off. Pressing the REL button releases the tape so it can be

moved by hand through the punch mechanism. Pressing B. SP. moves the tape backward one frame so the operator can delete a frame that is incorrect by striking the rubout key. Pressing HERE IS with the keyboard in local punches twenty lines of blank tape (lines with only a feed hole punched).

The keyboard resembles that of a standard typewriter. Codes for printable characters on the upper parts of the key tops on the 33 and 35 are transmitted by using the shift key; most control codes require use of the control key. The control key is used for some control characters, but many have separate keys. Note also that both the keyboard arrangement and the labels differ somewhat. On all models the line feed spaces the paper vertically at six lines to the inch, and must be combined with a return to start a new line. The local advance (feed) and return keys affect the printer directly and do not transmit codes. Appendix E lists the complete teletype code, ASCII characters, key combinations, and differences among the several models.

On the 33 and 35 is a repeat button REPT. Pressing this button and striking any character key causes transmission of the corresponding code so long as REPT is held down. Characters that require the shift key may also be repeated in this manner, but there is no repetition of control characters.

Teletype manuals supplied with the equipment give complete, illustrated descriptions of the procedures for loading paper and tape, changing the ribbon, and setting horizontal and vertical tabs. Setting tabs is usually left for maintenance personnel; in any event, the best and easiest way to learn how to do any of these things is to have someone who knows show you how. However, as a precautionary measure we describe here the things you may have to do yourself.

TAPE

The tape moves in the reader from back to front with the feed holes closer to the left edge. To load tape, set the switch to FREE, release the cover guard by opening the latch at the right, place the tape so that the sprocket wheel teeth engage the feed holes, close the cover guard, and set the switch to STOP.

To load tape in the punch, raise the cover, feed the tape manually from the top of the roll into the guide at the back, move the tape through the punch by turning the friction wheel, then close the cover. Turn on the punch with the unit in local and punch about two feet of leader by pressing HERE IS or the control, shift and P keys to generate null codes.

6.3 PAPER

The 33 printer has an 8-1/2-inch roll of paper at the back. Printed sections can be torn off against the edge of the glass window in front of the platen. To replenish the paper, snap open the cover, remove the old roll and slip a new one in its place. Draw the paper from the roll around the platen as in an ordinary typewriter.

The 35 printer has a sprocket feed and uses 8-1/2 X 11 fanfold form paper. The supply is held in a tray at the back. To replenish it, first remove the upper cover by pressing the cover release button on the right side. To free the remaining old paper for removal, lift the paper guides by pushing the handle marked PUSH at the right of the platen. To insert new paper from the tray, bring it up below the platen at the rear, line up the holes at the edges of the paper with the sprockets, and press line feed (in local) to draw the paper under the platen.

6.4 RIBBON

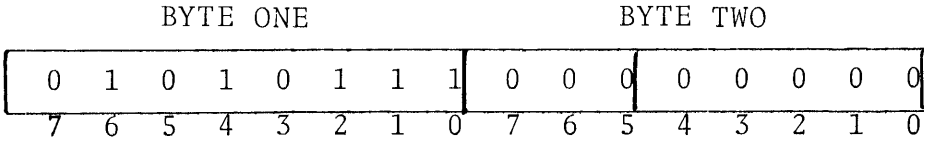
Replace the ribbon whenever it becomes worn or frayed or the printing becomes too light. Disengage the old ribbon from the ribbon guides on either side of the type block, and remove the reels by lifting the spring clips on the reel spindles and pulling the reels off.

Remove the old ribbon from one of the reels and replace the empty reel on one side of the machine; install a new reel on the other side. Push down both reel spindle spring clips to secure the reels. Unwind the fresh ribbon from the inside of the supply reel, over the guide roller, through the two guides on either side of the type block, out around the other guide roller, and back onto the inside of the takeup reel. Engage the hook on the end of the ribbon over the point of the arrow in the hub. Wind a few turns of the ribbon to make sure that the reversing eyelet has been wound onto the spool. Make sure the ribbon is seated properly and feeds correctly in operation.

6.5 TELETYPE OUTPUT

An output to the teletype requires only one transfer instruction. This command will transfer the contents of either accumulator and also initiates the printing of the character. The device code address is 00 and no control codes are used.

DOB TTY, TRANSFER DATA OUT OF B TO TELETYPE



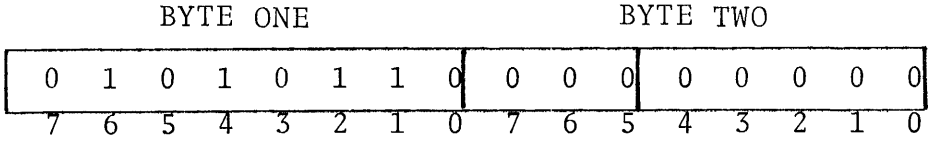
If Byte One bit two were a zero, the contents of A would be transferred and the mnemonic would be DOA TTY. If the punch is on, the character will be printed and also punched on tape.

The Model 33 and 35 can type or punch up to ten characters per second. After the DOB or DOA command, the program must send the next character 9.09 milliseconds later to maintain maximum rate.

6.6 TELETYPE INPUT

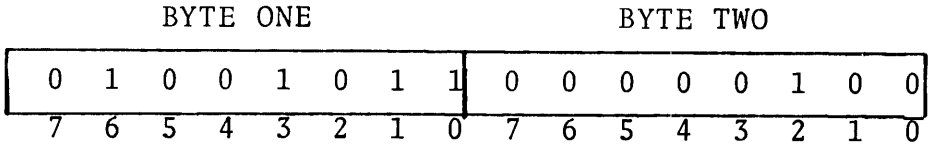
Reception from the keyboard requires no initiating action by the program; striking a key transmits the code for the character serially into the input buffer of the teletype controller and sets the Reader Ready flag. However, under program control, a start reader code may be transmitted to the teletype to start reading paper tape. This will cause the reader to start moving the tape to read all eight channels from the next line on tape and transmit the line serially into the buffer. (The presence of a hole in the tape produces a one in the buffer.) When the character is brought into an accumulator, tape channel 1 corresponds to bit position zero.

DIB TTY, DATA TRANSFER INTO B FROM THE TELETYPE



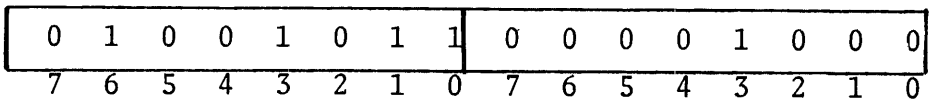
If byte one bit two were a zero, the accumulator A would be selected.

6.7 SKIP IF TELETYPE READER READY IS SET (SRS)



Two choices of Reader Ready test may be made, ready or not ready. If byte one bit zero is a zero the skip condition would be NOT READY. The Skip if Reader Ready command will cause a program skip if a character is in the teletype buffer. The Transfer in of this character will clear the reader ready flag in the controller.

6.8 SKIP IF TELETYPE WRITE READY IS SET (SWS)



Two choices of Writer Ready test may be made, ready or not ready. If Byte one bit zero is a zero the skip condition would be not ready. The Skip if Writer Ready command will cause a skip if the teletype is ready to accept an output of data to the printer and punch.

7.0 DIRECT MEMORY ACCESS (DMA)

The maximum rate for data transfers between external devices and core memory could be no greater than 77,000 words per second if the transfers were executed under program control. To allow data transfer rates up to 625,000 words per second, a Direct Memory Access (DMA) Channel is available as an option. The DMA option makes a channel available through which data can be transferred automatically using one processor cycle (1.6us) per word. At data rates lower than 625,000 words per second, the DMA channel frees processor time to allow execution of a program sequenced with data transfers for a device.

The program cannot affect the DMA channel because there are no instructions for it; instead, the program sets up the device (via the input/output bus) to use the channel. Then, when the device requires data service, it requests access to the memory via the DMA request line. At the beginning of every memory cycle, the processor synchronizes and recognizes any requests for memory access that are being made. This assures no more than 1.6 microseconds wait for DMA and as long as the DMA request remains true, every memory cycle is available.

The device requesting DMA service must supply the necessary control signals as well as address to the memory unit. As an option for the MICRO 400 the Block Automatic Transfer (BAT) is available to simplify use of the DMA channel. The BAT supplies control and sequences the memory address for all standard peripheral devices using the DMA channel.

The following data is presented for a better understanding of how the DMA channel operates and as design data for the systems engineer. See Appendix D for specific connector layout and pin listings.

7.1 DMA INTERFACE

The DMA option conditions certain control functions and upper address lines. The I/O board is required to allow transfers of address and data through it.

ADLD : The Address Load Enable is used to gate the address bits 00-07 through the I/O Interface to the memory.

DMRQ : The DMA Request Line is used to request a DMA cycle. DMRQ cannot be issued until the response DMRS is not busy.

DMRS : The DMA Response Line is used to determine if the DMA is in use.

DMMS: The DMA Memory Start is used to initiate the memory cycle. DMMS strobes the address in the address register on its leading edge.

DMCR: The DMA Clear Write is used when a transfer into memory is desired.

DAIG: The Data In Gate is used to gate the data bits 00 ⟶ 07 through the I/O interface.

DMIS: The Data in Strobe is used to strobe the data bits 00 ⟶ 07 into the memory data register.

DMOS: The Data Out Strobe is used to gate the data bits 00 ⟶ 07 from the memory data register through the I/O interface to the external device.

EB00⟶EB07: The I/O data bus is used to transfer address bits 00 ⟶ 07 to the memory address register and transfer data bits 00 ⟶ 07 to/from the memory data register.

DM08⟶DM15: The DMA address bits 08 ⟶ 15 are transferred as follows: Bits 08 ⟶ 11 are transferred to the memory and bits 12 ⟶ 15 are transferred to the Base Register Module to select which 4K module the DMA cycle is for.

7.2 START DMA CYCLE

Check if DMA is being used (DMRS will be high if not in use). If DMRS is high, a DMA request (Lower DMRQ) can be made. DMRS will go low when the CPU has accepted the DMRQ. There is a maximum of 1.6 microseconds and a minimum of "0" microseconds from DMRQ to DMRS.

Address load (ADLD) must precede memory start (DMMS) by 50 nanoseconds minimum and anything greater than 50 nanoseconds is acceptable. During ADLD place the address of the memory location that is to be accessed on the Address Bus (DM08 ⟶ DM15) (Most Significant 8 bits) and the I/O Data Bus (EB00 ⟶ EB07) (Least Significant 8 bits).

DMMS can be applied from 50 nanoseconds minimum to 300 nanoseconds maximum after the DMRS. If this is not followed, time will be lost since the DMA total cycle will take longer than 1.6 microseconds.

7.2.2 Transfer Into Memory

If the transfer is into the memory the clear write flag (DMCR) must be issued and be stabilized by 300 nanoseconds maximum after the leading edge of DMMS and remain stable for 1 microsecond.

The data to be stored into the memory must be placed on the Data Bus (EB00 ⟶ EB07) during DAIG which is 500 nanoseconds maximum after the leading edge of DMMS and must be removed by 1.3 Microseconds after the leading edge of DMMS. DAIG is 400 nanoseconds wide. DAIG is

issued 200 nanoseconds maximum after the leading edge of DAIG and is used to direct set the data into the memory data register. DMIS must be 100 nanoseconds minimum wide and must be totally windowed by DAIG · with 100 nanoseconds minimum margin from the trailing edge.

7.2.3 Transfer Out Of Memory

Data from the memory data register is gated to the data bus (EB00 ⟶ EB07) by the data from memory enable (DMOS). The leading edge of DMOS must be a minimum of 500 nanoseconds from the leading edge of DMMS and DMOS must be removed by 1.3 microseconds from the leading edge of DMMS. The data at the external device will be stable 100 nanoseconds after the leading edge of DMOS and during the rest of DMOS.

7.2.4 Single or Additional DMA Cycles

If a single DMA cycle is desired DMRQ must be removed by 1.1 microseconds after the leading edge of DMMS.

If additional DMA cycles are desired DMRQ is held low.

7.3 DMA CYCLE

The DMA cycle requires the following operations.

See Figure B-4.

7.3.1 Check if DMA is being used (DMRS will be high if not in use). If so, make a DMA request (Lower DMRQ).

7.3.2 Detect that the CPU is waiting to perform a DMA cycle (DMRS will go low).

7.3.3 Put the address of the memory location that is to be accessed on the Address Bus (Most Significant 8 bits) and the Data Bus (Lease Significant 8 bits).

7.3.4 At least 300 nanoseconds after 7.3.2 apply a memory start pulse (DMMS) which has a duration of 100 nanoseconds minimum (200 nanoseconds maximum).

7.3.5 The clear write flag (DMCR) must be stablized by 300 nanoseconds maximum after the leading edge of DMMS and remain stable for one microsecond.

7.3.6 If data is to be stored into the memory, it must be put on the Data Bus during DAIG which is 500 nanoseconds maximum after the leading edge of DMMS and removed before starting a new memory cycle.

7.3.7 If a clear write cycle (store cycle) is being executed, the data to memory strobe (DMIS) must be sent 100 nanoseconds minimum (200 nanoseconds maximum) after the leading edge of DMMS. It must be 100 nanoseconds wide minimum; 200 nanoseconds maximum.

7.3.8 Data from the memory is gated to the data bus (DBOO--DBO7) by the data from memory enable (DMOS), 500 nanoseconds from the leading edge of DMMS. The data at the external device will be stable 100 nanoseconds after the leading edge of DMOS.

7.3.9 If an additional DMA cycle is desired, start at 7.2.3. If no additional DMA cycles are desired, release the DMRQ signal 1.2 microseconds after the last DMMS.

7.3.10 The data from memory will remain on the D-bus as long as DMOS is present. Therefore, DMOS must be removed before the next memory address is required.

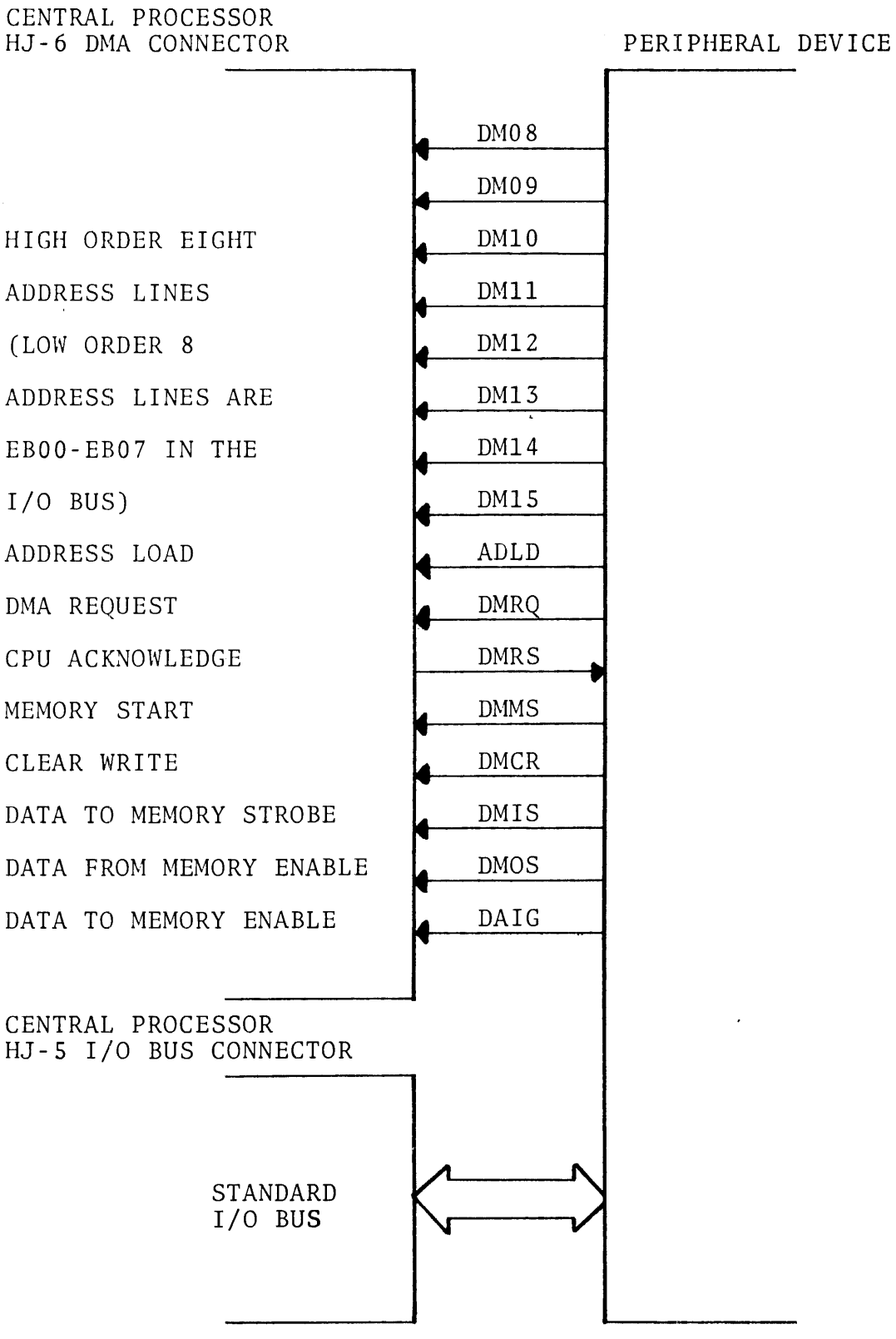
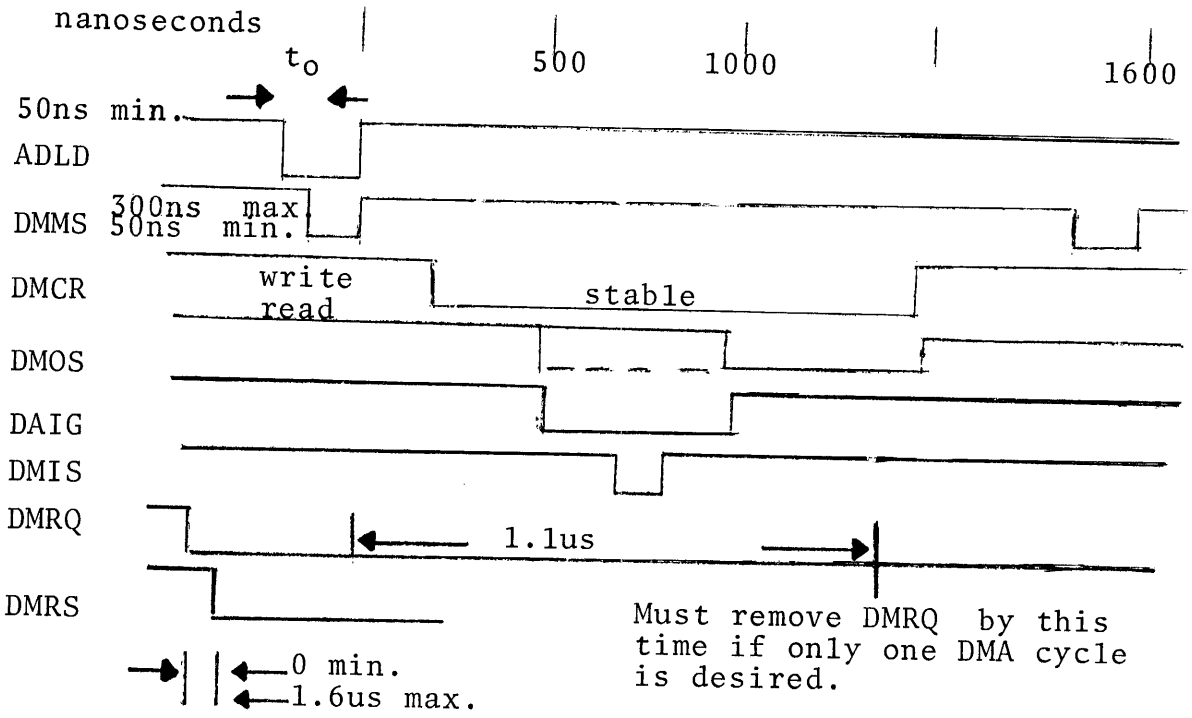


FIGURE 7-1

DMA BUS AND STANDARD I/O BUS RELATIONSHIP



NOTE: Signals are shown ground true as they appear on the I/O bus.

FIGURE B-4

DMA CYCLE TIMING

APPENDIX A

INSTRUCTION LISTING

MICRO 400 INSTRUCTION LISTING

ARITHMETIC

	MNEMONIC	OP CODE	DESCRIPTION	* LINK/ OVERFLOW
ADD	ABA	12	ADD B TO A	A
	AAB	09	ADD A TO B	A
SUBTRACT	SBA	14	SUBTRACT B FROM A TO A	A
	SAA	22	SUBTRACT A FROM B TO A	A
AND	NBA	1E	AND B WITH A TO A	R
	NAB	3B	AND A WITH B TO B	R
INCLUSIVE OR	OBA	29	INCLUSIVE OR B WITH A TO A	R
	OAB	0D	INCLUSIVE OR A WITH B TO B	R
EXCLUSIVE OR	XBA	2D	EXCLUSIVE OR B WITH A TO A	R
LEFT SHIFT	LSA	11	LEFT SHIFT A ONE BIT, INSERT 0	A
	LSB	0A	LEFT SHIFT B ONE BIT, INSERT 0	A
RIGHT SHIFT	RSA	1F	RIGHT SHIFT A ONE BIT, INSERT 0	-
	RSB	3F	RIGHT SHIFT B ONE BIT, INSERT 0	-
	RRA	0F	RIGHT ROTATE A ONE BIT	-
	RRB	2F	RIGHT ROTATE B ONE BIT	-
COMPLEMENT	OCA	20	ONES COMPLEMENT A	S
	OCB	04	ONES COMPLEMENT B	S
	TCA	26	TWOS COMPLEMENT A	A
	TCB	34	TWOS COMPLEMENT B	A
INCREMENT	IAA	16	INCREMENT A BY 1 TO A	A
	IAB	31	INCREMENT A BY 1 TO B	A
	IBB	32	INCREMENT B BY 1 TO B	A
DECREMENT	DAA	10	DECREMENT A BY 1 TO A	A
	DAB	01	DECREMENT A BY 1 TO B	A
	DBB	02	DECREMENT B BY 1 TO B	A
COPY	CAB	39	COPY A INTO B	R
	CBA	2B	COPY B INTO A	R
CLEAR	CLA	21	CLEAR A TO ZERO	S
	CLB	06	CLEAR B TO ZERO	S
SET	POA	17	POSITIVE ONE TO A	R
	POB	07	POSITIVE ONE TO B	R
	MOA	18	MINUS ONE TO A	R
	MOB	08	MINUS ONE TO B	R
LINK	CLL	0E	CLEAR LINK BIT TO ZERO	R
	LCB	30	SET LINK BIT AND CLEAR B	S

INSTRUCTION FORMAT: XXX

* NOTE - AFFECT ON LINK/OVERFLOW BITS
S - LINK ALWAYS SET
R - LINK ALWAYS RESET
A - LINK AFFECTED (MAY BE SET OR RESET) AND OVERFLOW MAY BE SET

MICRO 400 INSTRUCTION LISTING

MEMORY REFERENCE

MNEMONIC	OP CODE	DESCRIPTION
LDA	** **	LOAD A FROM MEMORY
LDB	** **	LOAD B FROM MEMORY
STA	** **	STORE A INTO MEMORY
STB	** **	STORE B INTO MEMORY

INSTRUCTION FORMAT: XXX ADDR (DIRECT ADDRESSING)
: XXX BIAS,X(,M) (INDEXED)

WHERE: ADDR IS A 12 BIT ADDRESS OF OPERAND IN 4K PAGE
: BIAS IS AN 8 BIT VALUE (0-255) WHICH IS ADDED TO THE SELECTED INDEX REGISTER TO FORM THE OPERAND ADDRESS
: X IS '1' OR '2' AND SELECTS DESIRED INDEX
: M (OPTIONAL) INDEX MODIFIER
'I' INCREMENT INDEX BY 1
'D' DECREMENT INDEX BY 1

BIT FORMAT OF UPPER BYTE OF INSTRUCTION:

INSTRUCTION BIT	DIRECT 7654 3210	INDEXED 7654 3210
LDA	1000 AAAA	011M M00X
LDB	1001 AAAA	011M M01X
STA	1010 AAAA	011M M10X
STB	1011 AAAA	011M M11X

WHERE: A IS UPPER 4 BITS OF 12 BIT ADDRESS
: MM IS INDEX MODIFIER
00 - NO CHANGE TO INDEX
01 - DECREMENT SELECTED INDEX BY ONE
10 - INCREMENT SELECTED INDEX BY ONE
: X IS INDEX SELECTION BIT
0 - USE INDEX REGISTER 1 (X1)
1 - USE INDEX REGISTER 2 (X2)

DETAIL LOAD/STORE INSTRUCTION FORMATS

A-REGISTER FORMATS

MNEMONIC	OPERAND	OP CODE
LDA	ADDR	8X XX
LDA	BIAS,1	60 XX
LDA	BIAS,2	61 XX
LDA	BIAS,1,D	68 XX
LDA	BIAS,2,D	69 XX
LDA	BIAS,1,I	70 XX
LDA	BIAS,2,I	71 XX
STA	ADDR	AX XX
STA	BIAS,1	64 XX
STA	BIAS,2	65 XX
STA	BIAS,1,D	6C XX
STA	BIAS,2,D	6D XX
STA	BIAS,1,I	74 XX
STA	BIAS,2,I	75 XX

B-REGISTER FORMATS

MNEMONIC	OPERAND	OP CODE
LDB	ADDR	9X XX
LDB	BIAS,1	62 XX
LDB	BIAS,2	63 XX
LDB	BIAS,1,D	6A XX
LDB	BIAS,2,D	6B XX
LDB	BIAS,1,I	72 XX
LDB	BIAS,2,I	73 XX
STB	ADDR	BX XX
STB	BIAS,1	66 XX
STB	BIAS,2	67 XX
STB	BIAS,1,D	6E XX
STB	BIAS,2,D	6F XX
STB	BIAS,1,I	76 XX
STB	BIAS,2,I	77 XX

SKIPS

MNEMONIC	OP CODE	DESCRIPTION
SAP	48 01	SKIP IF A POSITIVE (OR ZERO)
SAE	48 02	SKIP IF A IS EVEN
SAZ	48 04	SKIP IF A IS ZERO
SON	48 08	SKIP IF OVERFLOW NOT SET
SBP	48 10	SKIP IF B POSITIVE (OR ZERO)
SBE	48 20	SKIP IF B IS EVEN
SBZ	48 40	SKIP IF B IS ZERO
SLN	48 80	SKIP IF LINK NOT SET
SAM	49 01	SKIP IF A IS MINUS
SAO	49 02	SKIP IF A IS ODD
SAN	49 04	SKIP IF A IS NOT ZERO
SOS	49 08	SKIP IF OVERFLOW IS SET
SBM	49 10	SKIP IF B IS MINUS
SBO	49 20	SKIP IF B IS ODD
SBN	49 40	SKIP IF B IS NOT ZERO
SLS	49 80	SKIP IF LINK IS SET
SIN	4A 01	SKIP IF INTERRUPT-ENABLE NOT SET
SSN	4A 02	SKIP IF SENSE RESPONSE NOT SET
SRN	4A 04	SKIP IF TTY READ READY NOT SET
SWN	4A 08	SKIP IF TTY WRITE READY NOT SET
SI1	4A 10	SKIP IF X1L=0, INCREMENT X1 BY 1*
SD1	4A 20	SKIP IF X1L=0, DECREMENT X1 BY 1*
SN1	4A 40	SKIP IF SENSE SWITCH 1 NOT SET
SN2	4A 80	SKIP IF SENSE SWITCH 2 NOT SET
SIS	4B 01	SKIP IF INTERRUPT-ENABLE IS SET
SSS	4B 02	SKIP IF SENSE RESPONSE IS SET
SRS	4B 04	SKIP IF TTY READ READY IS SET
SWS	4B 08	SKIP IF TTY WRITE READY IS SET
SI2	4B 10	SKIP IF X2L=0, INCREMENT X2 BY 1*
SD2	4B 20	SKIP IF X2L=0, DECREMENT X2 BY 1*
SS1	4B 40	SKIP IF SENSE SWITCH 1 IS SET
SS2	4B 80	SKIP IF SENSE SWITCH 2 IS SET

INSTRUCTION FORMAT: XXX

NOTE - IF THE SKIP CONDITION IS TRUE, THE FOLLOWING TWO BYTES WILL BE SKIPPED.

* NOTE - THE SKIP CONDITION IS TESTED PRIOR TO THE INCREMENTING OR DECREMENTING. THE INCREMENTING OR DECREMENTING OCCURS WHETHER THE LOWER BYTE OF THE INDEX IS ZERO OR NOT.

GENERAL SKIPS

MNEMONIC	OP CODE	DESCRIPTION
SNZ	48 XX	SKIP IF ALL OF THE SELECTED N-GROUP CONDITIONS ARE ZERO
SNO	49 XX	SKIP IF ANY OF THE SELECTED N-GROUP CONDITIONS IS ONE.
SMZ	4A XX	SKIP IF ALL OF THE SELECTED M-GROUP CONDITIONS ARE ZERO.
SMO	4B XX	SKIP IF ANY OF THE SELECTED M-GROUP CONDITIONS IS ONE.

INSTRUCTION FORMAT: XXX LIT

WHERE: LIT IS AN 8 BIT LITERAL CONDITION MASK

N-GROUP CONDITIONS BIT	SOURCE
0	MOST SIGNIFICANT BIT OF A
1	LEAST SIGNIFICANT BIT OF A
2	ZERO IF A ZERO
3	OVERFLOW
4	MOST SIGNIFICANT BIT OF B
5	LEAST SIGNIFICANT BIT OF B
6	ZERO IF B ZERO
7	LINK

M-GROUP CONDITIONS BIT	SOURCE
0	INTERRUPTS ENABLED FLIP-FLOP
1	SENSE RESPONSE FLIP-FLOP
2	TTY READ READY
3	TTY WRITE READY
4	SEE INSTRUCTIONS SI1, SI2
5	SEE INSTRUCTIONS SD1, SD2
6	ZERO IF SENSE SWITCH 1 SET
7	ZERO IF SENSE SWITCH 2 SET

JUMPS

MNEMONIC	OP CODE	DESCRIPTION
JPC	CX XX	JUMP WITHIN CURRENT 4K PAGE
JPE	DX XX	JUMP TO EXTENDED 4K PAGE *
JMC	EX XX	JUMP & MARK IN CURRENT 4K PAGE
JME	FX XX	JUMP & MARK TO EXTENDED 4K PAGE *

INSTRUCTION FORMAT: XXX ADDR

WHERE: ADDR IS 12 BIT ADDRESS IN 4K PAGE

* NOTE - EXTENDED JUMPS ENABLE BASE REGISTER ADDRESSING AND TOGGLE THE BASE REGISTER SELECTION PRIOR TO THE JUMP UNLESS PRECEDED BY AN 'EBL' OR 'LEB' INSTRUCTION.

MICRO 400 INSTRUCTION LISTING

INPUT/OUTPUT

MNEMONIC	OP CODE	DESCRIPTION
SEN	50 XX	SENSE
EXC	51 XX	EXTERNAL CONTROL
DIA	52 XX	DATA INPUT INTO A
DOA	53 XX	DATA OUTPUT FROM A
DIB	56 XX	DATA INPUT INTO B
DOB	57 XX	DATA OUTPUT FROM B

INSTRUCTION FORMAT: XXX F,D

WHERE: F IS A 3 BIT FUNCTION CODE
: D IS A 5 BIT DEVICE ADDRESS

CONTROL

MNEMONIC	OP CODE	DESCRIPTION
HLT	00	HALT
SOF	40	SET OVERFLOW FLIP-FLOP
ROF	41	RESET OVERFLOW FLIP-FLOP
EIN	42	ENABLE INTERRUPTS
DIN	43	DISABLE INTERRUPTS
NOP	45	NO OPERATION
SSF	46	SET SENSE FLIP-FLOP
RSF	47	RESET SENSE FLIP-FLOP

INSTRUCTION FORMAT: XXX

MEMORY MODIFY/SKIP

	MNEMONIC	OP CODE	DESCRIPTION
SKIP	SER	58 XX	SKIP IF EVEN, ROTATE MEMORY *
	SOR	59 XX	SKIP IF ODD, ROTATE MEMORY *
	SPM	5A XX	SKIP IF POSITIVE MEMORY
	SMM	5B XX	SKIP IF MINUS MEMORY
SHIFT MEMORY	RCS	5C XX	RIGHT SHIFT, COPY SENSE
	RCZ	5D XX	RIGHT SHIFT, COPY ZERO
	RCO	5E XX	RIGHT SHIFT, COPY ONE
	RCA	5F XX	RIGHT SHIFT, COPY LEAST SIGNIFICANT BIT OF A

INSTRUCTION FORMAT: XXX ADDR

WHERE: ADDR IS AN 8 BIT ADDRESS OF THE FIRST 256 BYTES OF THE CURRENT 4K PAGE

* NOTE - THE MEMORY WORD IS CIRCULARLY ROTATED TO THE RIGHT ONE BIT POSITION. THE SKIP CONDITION IS TESTED PRIOR TO MEMORY ROTATION. ROTATION WILL TAKE PLACE REGARDLESS OF WHETHER THE SKIP CONDITION IS TRUE OR FALSE.

SPECIAL BASE-REGISTER/INDEX INSTRUCTIONS ----------

	MNEMONIC	OP CODE	DESCRIPTION
TRANSFER	TRA	** **	TRANSFER BYTE BETWEEN CPU AND BRM

INSTRUCTION FORMAT: TRA S,D

WHERE: S IS THE MNEMONIC FOR THE SOURCE REGISTER
: D IS THE MNEMONIC FOR THE DESTINATION REGISTER

NOTE - S AND D MUST BE FROM DIFFERENT MNEMONIC SETS (SEE BELOW)

BIT FORMAT OF TRANSFER (TRA) INSTRUCTIONS:

	FIRST BYTE	SECOND BYTE
BIT	7654 3210	7654 3210
TRA	0101 0A1D	HRN0 1010

WHERE: A IS ACCUMULATOR SELECTION
(0=A ACCUMULATOR, 1=B ACCUMULATOR)
D IS DIRECTION OF TRANSFER
(0=BRM TO CPU, 1=CPU TO BRM)
H IS REGISTER HALF SELECTION
(0=LOWER 8 BITS, 1=UPPER 8 BITS)
R IS REGISTER TYPE SELECTION
(0=BASE REGISTER, 1=INDEX REGISTER)
N IS REGISTER NUMBER
(0= X1 OR B1, 1= X2 OR B2)

DETAIL TRANSFER INSTRUCTION FORMATS

A-REGISTER FORMATS

MNEMONIC	OPERAND	OP CODE
TRA	X1L,A	52 4A
TRA	X2L,A	52 6A
TRA	X1U,A	52 CA
TRA	X2U,A	52 EA
TRA	A,X1L	53 4A
TRA	A,X2L	53 6A
TRA	A,X1U	53 CA
TRA	A,X2U	53 EA
TRA	B1L,A	52 0A
TRA	B2L,A	52 2A
TRA	B1U,A	52 8A
TRA	B2U,A	52 AA
TRA	A,B1L	53 0A
TRA	A,B2L	53 2A
TRA	A,B1U	53 8A
TRA	A,B2U	53 AA

B-REGISTER FORMATS

MNEMONIC	OPERAND	OP CODE
TRA	X1L,B	56 4A
TRA	X2L,B	56 6A
TRA	X1U,B	56 CA
TRA	X2U,B	56 EA
TRA	B,X1L	57 4A
TRA	B,X2L	57 6A
TRA	B,X1U	57 CA
TRA	B,X2U	57 EA
TRA	B1L,B	56 0A
TRA	B2L,B	56 2A
TRA	B1U,B	56 8A
TRA	B2U,B	56 AA
TRA	B,B1L	57 0A
TRA	B,B2L	57 2A
TRA	B,B1U	57 8A
TRA	B,B2U	57 AA

SPECIAL BASE REGISTER/INDEX INSTRUCTIONS - (CONTINUED)----------

	MNEMONIC	OP CODE	DESCRIPTION
CLEAR	CLR	** **	CLEAR BRM REGISTER TO ZERO

INSTRUCTION FORMAT: CLR REG

WHERE: REG IS A BRM MNEMONIC (SEE BELOW)

BIT FORMAT OF CLEAR (CLR) INSTRUCTIONS:

	FIRST BYTE	SECOND BYTE
BIT	7654 3210	7654 3210
CLR	0101 0001	HRN0 1010

WHERE: H IS REGISTER HALF SELECTION
(0=LOWER 8 BITS, 1=UPPER 8 BITS)
R IS REGISTER TYPE SELECTION
(0=BASE REGISTER, 1=INDEX REGISTER)
N IS REGISTER NUMBER
(0= X1 OR B1, 1= X2 OR B2)

BRM MNEMONICS (BASE-REGISTER/INDEX MODULE)

'X1U'	'X1L'	INDEX 1
'X2U'	'X2L'	INDEX 2
'B1U'	'B1L'	BASE 1
'B2U'	'B2L'	BASE 2

DETAIL CLEAR INSTRUCTION FORMATS

MNEMONIC	OPERAND	OP CODE	DESCRIPTION
CLR	B1L	51 0A	CLEAR BASE 1 LOWER TO ZERO
CLR	B2L	51 2A	CLEAR BASE 2 LOWER TO ZERO
CLR	X1L	51 4A	CLEAR INDEX 1 LOWER TO ZERO
CLR	X2L	51 6A	CLEAR INDEX 2 LOWER TO ZERO
CLR	B1U	51 8A	CLEAR BASE 1 UPPER TO ZERO
CLR	B2U	51 AA	CLEAR BASE 2 UPPER TO ZERO
CLR	X1U	51 CA	CLEAR INDEX 1 UPPER TO ZERO
CLR	X2U	51 EA	CLEAR INDEX 2 UPPER TO ZERO

SPECIAL BASE REGISTER/INDEX INSTRUCTIONS - (CONTINUED)

	MNEMONIC	OP CODE	DESCRIPTION
SET	SB1	50 2A	SET SENSE IF BASE 1 NOT ENABLED
	SLE	50 4A	SET SENSE IF 'LEB' SET
	SEL	50 8A	SET SENSE IF 'EBL' SET
	SEM	50 8B	SET SENSE IF IN EXTENDED MEMORY
	LEB	51 2B	SET 'LEB' *
	EBL	51 4B	SET 'EBL' *

NOTE - THE 'EBL' COMMAND ALSO INHIBITS THE NEXT EXTENDED JUMP (JME,JPE) FROM TOGGLING THE BASE REGISTER SELECTION.

INSTRUCTION FORMAT: XXX

* NOTE - 'LEB' IS LOWER CORE TO EXTENDED BASE FLIP-FLOP
'EBL' IS EXTENDED BASE TO LOWER CORE FLIP-FLOP

INSTRUCTION TIMING

FUNCTION		TIME (CYCLE)	
ARITHMETIC		1	
CLEAR REGISTER	(BRM)	2	
CONTROL		1	
INPUT		2	
JUMP		2	
JUMP MARK		4	
LOAD/STORE DIRECT		3	
LOAD/STORE INDEXED		3	
MEMORY MODIFY/SKIP		3	(4 IF SKIP OCCURS)
OUTPUT		2	
SHIFT		1	
SKIP		2	(3 IF SKIP OCCURS)
TRANSFER REGISTER	(BRM)	2	

(1 CYCLE EQUALS 1.6 MICROSECONDS)

MICRO 400 INSTRUCTIONS MNEMONIC SEQUENCE

INSTRUCTION		OP CODE	DESCRIPTION
AAB		09	ADD A TO B
ABA		12	ADD B TO A
CAB		39	COPY A INTO B
CBA		2B	COPY B INTO A
CLA		21	CLEAR A TO ZERO
CLB		06	CLEAR B TO ZERO
CLL		0E	CLEAR LINK BIT TO ZERO
CLR	B1L	51 0A	CLEAR BASE 1 LOWER TO ZERO
CLR	B2L	51 2A	CLEAR BASE 2 LOWER TO ZERO
CLR	X1L	51 4A	CLEAR INDEX 1 LOWER TO ZERO
CLR	X2L	51 6A	CLEAR INDEX 2 LOWER TO ZERO
CLR	B1U	51 8A	CLEAR BASE 1 UPPER TO ZERO
CLR	B2U	51 AA	CLEAR BASE 2 UPPER TO ZERO
CLR	X1U	51 CA	CLEAR INDEX 1 UPPER TO ZERO
CLR	X2U	51 EA	CLEAR INDEX 2 UPPER TO ZERO
DAA		10	DECREMENT A BY 1 TO A
DAB		01	DECREMENT A BY 1 TO B
DBB		02	DECREMENT B BY 1 TO B
DIA	F,D	52 XX	DATA INPUT INTO A
DIB	F,D	56 XX	DATA INPUT INTO B
DIN		43	DISABLE INTERRUPTS
DOA	F,D	53 XX	DATA OUTPUT FROM A
DOB	F,D	57 XX	DATA OUTPUT FROM B
EBL		51 4B	SET 'EBL'
EIN		42	ENABLE INTERRUPTS
EXC	F,D	51 XX	EXTERNAL CONTROL
HLT		00	HALT
IAA		16	INCREMENT A BY 1 TO A
IAB		31	INCREMENT A BY 1 TO B
IBB		32	INCREMENT B BY 1 TO B
JMC	ADDR	EX XX	JUMP & MARK IN CURRENT 4K PAGE
JME	ADDR	FX XX	JUMP & MARK TO EXTENDED 4K PAGE
JPC	ADDR	CX XX	JUMP WITHIN CURRENT 4K PAGE
JPE	ADDR	DX XX	JUMP TO EXTENDED 4K PAGE
LCB		30	SET LINK BIT AND CLEAR B
LDA	ADDR	8X XX	LOAD A FROM MEMORY
LDA	BIAS,1	60 XX	LOAD A INDEXED BY X1
LDA	BIAS,2	61 XX	LOAD A INDEXED BY X2
LDA	BIAS,1,D	68 XX	LOAD A INDEXED BY X1 DECREMENT X1
LDA	BIAS,2,D	69 XX	LOAD A INDEXED BY X2 DECREMENT X2
LDA	BIAS,1,I	70 XX	LOAD A INDEXED BY X1 INCREMENT X1
LDA	BIAS,2,I	71 XX	LOAD A INDEXED BY X2 INCREMENT X2

MICRO 400 INSTRUCTIONS MNEMONIC SEQUENCE

INSTRUCTION		OP CODE	DESCRIPTION
LDB	ADDR	9X XX	LOAD B FROM MEMORY
LDB	BIAS,1	62 XX	LOAD B INDEXED BY X1
LDB	BIAS,2	63 XX	LOAD B INDEXED BY X2
LDB	BIAS,1,D	6A XX	LOAD B INDEXED BY X1 DECREMENT X1
LDB	BIAS,2,D	6B XX	LOAD B INDEXED BY X2 DECREMENT X2
LDB	BIAS,1,I	72 XX	LOAD B INDEXED BY X1 INCREMENT X1
LDB	BIAS,2,I	73 XX	LOAD B INDEXED BY X2 INCREMENT X2
LEB		51 2B	SET 'LEB'
LSA		11	LEFT SHIFT A ONE BIT, INSERT 0
LSB		0A	LEFT SHIFT B ONE BIT, INSERT 0
MOA		18	MINUS ONE TO A
MOB		08	MINUS ONE TO B
NAB		3B	AND A WITH B TO B
NBA		1E	AND B WITH A TO A
NOP		45	NO OPERATION
OAB		0D	INCLUSIVE OR A WITH B TO B
OBA		29	INCLUSIVE OR B WITH A TO A
OCA		20	ONES COMPLEMENT A
OCB		04	ONES COMPLEMENT B
POA		17	POSITIVE ONE TO A
POB		07	POSITIVE ONE TO B
RCA	ADDR	5F XX	RIGHT SHIFT, COPY LEAST SIGNIFICANT BIT OF A
RCO	ADDR	5E XX	RIGHT SHIFT, COPY ONE
RCS	ADDR	5C XX	RIGHT SHIFT, COPY SENSE
RCZ	ADDR	5D XX	RIGHT SHIFT, COPY ZERO
ROF		41	RESET OVERFLOW FLIP-FLOP
RRA		0F	RIGHT ROTATE A ONE BIT
RRB		2F	RIGHT ROTATE B ONE BIT
RSA		1F	RIGHT SHIFT A ONE BIT, INSERT 0
RSB		3F	RIGHT SHIFT B ONE BIT, INSERT 0
RSF		47	RESET SENSE FLIP-FLOP
SAA		22	SUBTRACT A FROM B TO A
SAE		48 02	SKIP IF A IS EVEN
SAM		49 01	SKIP IF A IS MINUS
SAN		49 04	SKIP IF A IS NOT ZERO
SAO		49 02	SKIP IF A IS ODD
SAP		48 01	SKIP IF A POSITIVE (OR ZERO)
SAZ		48 04	SKIP IF A IS ZERO
SBA		14	SUBTRACT B FROM A TO A
SBE		48 20	SKIP IF B IS EVEN
SBM		49 10	SKIP IF B IS MINUS

MICRO 400 INSTRUCTIONS MNEMONIC SEQUENCE

INSTRUCTION		OP CODE	DESCRIPTION
SBN		49 40	SKIP IF B IS NOT ZERO
SBO		49 20	SKIP IF B IS ODD
SBP		48 10	SKIP IF B POSITIVE (OR ZERO)
SBZ		48 40	SKIP IF B IS ZERO
SB1		50 2A	SET SENSE IF BASE 1 NOT ENABLED
SD1		4A 20	SKIP IF X1L=0, DECREMENT X1 BY 1
SD2		4B 20	SKIP IF X2L=0, DECREMENT X2 BY 1
SEL		50 8A	SET SENSE IF 'EBL' SET
SEM		50 8B	SET SENSE IF IN EXTENDED MEMORY
SEN	F,D	50 XX	SENSE
SER	ADDR	58 XX	SKIP IF EVEN, ROTATE MEMORY
SIN		4A 01	SKIP IF INTERRUPT-ENABLE NOT SET
SIS		4B 01	SKIP IF INTERRUPT-ENABLE IS SET
SI1		4A 10	SKIP IF X1L=0, INCREMENT X1 BY 1
SI2		4B 10	SKIP IF X2L=0, INCREMENT X2 BY 1
SLE		50 4A	SET SENSE IF 'LEB' SET
SLN		48 80	SKIP IF LINK NOT SET
SLS		49 80	SKIP IF LINK IS SET
SMM	ADDR	5B XX	SKIP IF MINUS MEMORY
SMO	LIT	4B XX	SKIP IF ANY OF THE SELECTED M-GROUP CONDITIONS IS ONE
SMZ	LIT	4A XX	SKIP IF ALL OF THE SELECTED M-GROUP CONDITIONS ARE ZERO
SNO	LIT	49 XX	SKIP IF ANY OF THE SELECTED N-GROUP CONDITIONS IS ONE
SNZ	LIT	48 XX	SKIP IF ALL OF THE SELECTED N-GROUP CONDITIONS ARE ZERO
SN1		4A 40	SKIP IF SENSE SWITCH 1 NOT SET
SN2		4A 80	SKIP IF SENSE SWITCH 2 NOT SET
SOF		40	SET OVERFLOW FLIP-FLOP
SON		48 08	SKIP IF OVERFLOW NOT SET
SOR	ADDR	59 XX	SKIP IF ODD, ROTATE MEMORY
SOS		49 08	SKIP IF OVERFLOW IS SET
SPM	ADDR	5A XX	SKIP IF POSITIVE MEMORY
SRN		4A 04	SKIP IF TTY READ READY NOT SET
SRS		4B 04	SKIP IF TTY READ READY IS SET
SSF		46	SET SENSE FLIP-FLOP
SSN		4A 02	SKIP IF SENSE RESPONSE NOT SET
SSS		4B 02	SKIP IF SENSE RESPONSE IS SET
SS1		4B 40	SKIP IF SENSE SWITCH 1 IS SET
SS2		4B 80	SKIP IF SENSE SWITCH 2 IS SET

MICRO 400 INSTRUCTIONS MNEMONIC SEQUENCE

INSTRUCTION		OP CODE	DESCRIPTION
STA	ADDR	AX XX	STORE A INTO MEMORY
STA	BIAS,1	64 XX	STORE A INDEXED BY X1
STA	BIAS,2	65 XX	STORE A INDEXED BY X2
STA	BIAS,1,D	6C XX	STORE A INDEXED BY X1 DECREMENT X1
STA	BIAS,2,D	6D XX	STORE A INDEXED BY X2 DECREMENT X2
STA	BIAS,1,I	74 XX	STORE A INDEXED BY X1 INCREMENT X1
STA	BIAS,2,I	75 XX	STORE A INDEXED BY X2 INCREMENT X2
STB	ADDR	BX XX	STORE B INTO MEMORY
STB	BIAS,1	66 XX	STORE B INDEXED BY X1
STB	BIAS,2	67 XX	STORE B INDEXED BY X2
STB	BIAS,1,D	6E XX	STORE B INDEXED BY X1 DECREMENT X1
STB	BIAS,2,D	6F XX	STORE B INDEXED BY X2 DECREMENT X2
STB	BIAS,1,I	76 XX	STORE B INDEXED BY X1 INCREMENT X1
STB	BIAS,2,I	77 XX	STORE B INDEXED BY X2 INCREMENT X2
SWN		4A 08	SKIP IF TTY WRITE READY NOT SET
SWS		4B 08	SKIP IF TTY WRITE READY IS SET
TCA		26	TWOS COMPLEMENT A
TCB		34	TWOS COMPLEMENT B
TRA	X1L,A	52 4A	TRANSFER X1L TO A
TRA	X2L,A	52 6A	TRANSFER X2L TO A
TRA	X1U,A	52 CA	TRANSFER X1U TO A
TRA	X2U,A	52 EA	TRANSFER X2U TO A
TRA	A,X1L	53 4A	TRANSFER A TO X1L
TRA	A,X2L	53 6A	TRANSFER A TO X2L
TRA	A,X1U	53 CA	TRANSFER A TO X1U
TRA	A,X2U	53 EA	TRANSFER A TO X2U
TRA	X1L,B	56 4A	TRANSFER X1L TO B
TRA	X2L,B	56 6A	TRANSFER X2L TO B
TRA	X1U,B	56 CA	TRANSFER X1U TO B
TRA	X2U,B	56 EA	TRANSFER X2U TO B
TRA	B,X1L	57 4A	TRANSFER B TO X1L
TRA	B,X2L	57 6A	TRANSFER B TO X2L
TRA	B,X1U	57 CA	TRANSFER B TO X1U
TRA	B,X2U	57 EA	TRANSFER B TO X2U
TRA	B1L,A	52 0A	TRANSFER B1L TO A
TRA	B2L,A	52 2A	TRANSFER B2L TO A
TRA	B1U,A	52 8A	TRANSFER B1U TO A
TRA	B2U,A	52 AA	TRANSFER B2U TO A
TRA	A,B1L	53 0A	TRANSFER A TO B1L
TRA	A,B2L	53 2A	TRANSFER A TO B2L
TRA	A,B1U	53 8A	TRANSFER A TO B1U
TRA	A,B2U	53 AA	TRANSFER A TO B2U
TRA	B1L,B	56 0A	TRANSFER B1L TO B
TRA	B2L,B	56 2A	TRANSFER B2L TO B
TRA	B1U,B	56 8A	TRANSFER B1U TO B
TRA	B2U,B	56 AA	TRANSFER B2U TO B
TRA	B,B1L	57 0A	TRANSFER B TO B1L
TRA	B,B2L	57 2A	TRANSFER B TO B2L
TRA	B,B1U	57 8A	TRANSFER B TO B1U
TRA	B,B2U	57 AA	TRANSFER B TO B2U
XBA		2D	EXCLUSIVE OR B WITH A TO A

MICRO 400 INSTRUCTIONS OP CODE SEQUENCE

INSTRUCTION	OP CODE	DESCRIPTION
HLT	00	HALT
DAB	01	DECREMENT A BY 1 TO B
DBB	02	DECREMENT B BY 1 TO B
OCB	04	ONES COMPLEMENT B
CLB	06	CLEAR B TO ZERO
POB	07	POSITIVE ONE TO B
MOB	08	MINUS ONE TO B
AAB	09	ADD A TO B
LSB	0A	LEFT SHIFT B ONE BIT, INSERT 0
OAB	0D	INCLUSIVE OR A WITH B TO B
CLL	0E	CLEAR LINK BIT TO ZERO
RRA	0F	RIGHT ROTATE A ONE BIT
DAA	10	DECREMENT A BY 1 TO A
LSA	11	LEFT SHIFT A ONE BIT, INSERT 0
ABA	12	ADD B TO A
SBA	14	SUBTRACT B FROM A TO A
IAA	16	INCREMENT A BY 1 TO A
POA	17	POSITIVE ONE TO A
MOA	18	MINUS ONE TO A
NBA	1E	AND B WITH A TO A
RSA	1F	RIGHT SHIFT A ONE BIT, INSERT 0
OCA	20	ONES COMPLEMENT A
CLA	21	CLEAR A TO ZERO
SAA	22	SUBTRACT A FROM B TO A
TCA	26	TWO'S COMPLEMENT A
OBA	29	INCLUSIVE OR B WITH A TO A
CBA	2B	COPY B INTO A
XBA	2D	EXCLUSIVE OR B WITH A TO A
RRB	2F	RIGHT ROTATE B ONE BIT
LCB	30	SET LINK BIT AND CLEAR B
IAB	31	INCREMENT A BY 1 TO B
IBB	32	INCREMENT B BY 1 TO B
TCB	34	TWO'S COMPLEMENT B
CAB	39	COPY A INTO B
NAB	3B	AND A WITH B TO B
RSB	3F	RIGHT SHIFT B ONE BIT, INSERT 0
SOF	40	SET OVERFLOW FLIP-FLOP
ROF	41	RESET OVERFLOW FLIP-FLOP
EIN	42	ENABLE INTERRUPTS
DIN	43	DISABLE INTERRUPTS
NOP	45	NO OPERATION
SSF	46	SET SENSE FLIP-FLOP
RSF	47	RESET SENSE FLIP-FLOP

MICRO 400 INSTRUCTIONS OP CODE SEQUENCE

INSTRUCTION		OP CODE	DESCRIPTION
SAP		48 01	SKIP IF A POSITIVE (OR ZERO)
SAE		48 02	SKIP IF A IS EVEN
SAZ		48 04	SKIP IF A IS ZERO
SON		48 08	SKIP IF OVERFLOW NOT SET
SBP		48 10	SKIP IF B POSITIVE (OR ZERO)
SBE		48 20	SKIP IF B IS EVEN
SBZ		48 40	SKIP IF B IS ZERO
SLN		48 80	SKIP IF LINK NOT SET
SNZ	LIT	48 XX	SKIP IF ALL OF THE SELECTED N-GROUP CONDITIONS ARE ZERO
SAM		49 01	SKIP IF A IS MINUS
SAO		49 02	SKIP IF A IS ODD
SAN		49 04	SKIP IF A IS NOT ZERO
SOS		49 08	SKIP IF OVERFLOW IS SET
SBM		49 10	SKIP IF B IS MINUS
SBO		49 20	SKIP IF B IS ODD
SBN		49 40	SKIP IF B IS NOT ZERO
SLS		49 80	SKIP IF LINK IS SET
SNO	LIT	49 XX	SKIP IF ANY OF THE SELECTED N-GROUP CONDITIONS IS ONE
SIN		4A 01	SKIP IF INTERRUPT-ENABLE NOT SET
SSN		4A 02	SKIP IF SENSE RESPONSE NOT SET
SRN		4A 04	SKIP IF TTY READ READY NOT SET
SWN		4A 08	SKIP IF TTY WRITE READY NOT SET
SI1		4A 10	SKIP IF X1L=0, INCREMENT X1 BY 1
SD1		4A 20	SKIP IF X1L=0, DECREMENT X1 BY 1
SN1		4A 40	SKIP IF SENSE SWITCH 1 NOT SET
SN2		4A 80	SKIP IF SENSE SWITCH 2 NOT SET
SMZ	LIT	4A XX	SKIP IF ALL OF THE SELECTED M-GROUP CONDITIONS ARE ZERO
SIS		4B 01	SKIP IF INTERRUPT-ENABLE IS SET
SSS		4B 02	SKIP IF SENSE RESPONSE IS SET
SRS		4B 04	SKIP IF TTY READ READY IS SET
SWS		4B 08	SKIP IF TTY WRITE READY IS SET
SI2		4B 10	SKIP IF X2L=0, INCREMENT X2 BY 1
SD2		4B 20	SKIP IF X2L=0, DECREMENT X2 BY 1
SS1		4B 40	SKIP IF SENSE SWITCH 1 IS SET
SS2		4B 80	SKIP IF SENSE SWITCH 2 IS SET
SMO	LIT	4B XX	SKIP IF ANY OF THE SELECTED M-GROUP CONDITIONS IS ONE
S31		50 2A	SET SENSE IF BASE 1 NOT ENABLED
SLE		50 4A	SET SENSE IF 'LEB' SET
SEL		50 8A	SET SENSE IF 'EBL' SET
SEM		50 8B	SET SENSE IF IN EXTENDED MEMORY
SEN	F,D	50 XX	SENSE

MICRO 400 INSTRUCTIONS OP CODE SEQUENCE

INSTRUCTION		OP CODE	DESCRIPTION
CLR	B1L	51 0A	CLEAR BASE 1 LOWER TO ZERO
CLR	B2L	51 2A	CLEAR BASE 2 LOWER TO ZERO
LEB		51 2B	SET 'LEB'
CLR	X1L	51 4A	CLEAR INDEX 1 LOWER TO ZERO
EBL		51 4B	SET 'EBL'
CLR	X2L	51 6A	CLEAR INDEX 2 LOWER TO ZERO
CLR	B1U	51 8A	CLEAR BASE 1 UPPER TO ZERO
CLR	B2U	51 AA	CLEAR BASE 2 UPPER TO ZERO
CLR	X1U	51 CA	CLEAR INDEX 1 UPPER TO ZERO
CLR	X2U	51 EA	CLEAR INDEX 2 UPPER TO ZERO
EXC	F,D	51 XX	EXTERNAL CONTROL
TRA	B1L,A	52 0A	TRANSFER B1L TO A
TRA	B2L,A	52 2A	TRANSFER B2L TO A
TRA	X1L,A	52 4A	TRANSFER X1L TO A
TRA	X2L,A	52 6A	TRANSFER X2L TO A
TRA	B1U,A	52 8A	TRANSFER B1U TO A
TRA	B2U,A	52 AA	TRANSFER B2U TO A
TRA	X1U,A	52 CA	TRANSFER X1U TO A
TRA	X2U,A	52 EA	TRANSFER X2U TO A
DIA	F,D	52 XX	DATA INPUT INTO A
TRA	A,B1L	53 0A	TRANSFER A TO B1L
TRA	A,B2L	53 2A	TRANSFER A TO B2L
TRA	A,X1L	53 4A	TRANSFER A TO X1L
TRA	A,X2L	53 6A	TRANSFER A TO X2L
TRA	A,B1U	53 8A	TRANSFER A TO B1U
TRA	A,B2U	53 AA	TRANSFER A TO B2U
TRA	A,X1U	53 CA	TRANSFER A TO X1U
TRA	A,X2U	53 EA	TRANSFER A TO X2U
DOA	F,D	53 XX	DATA OUTPUT FROM A
TRA	B1L,B	56 0A	TRANSFER B1L TO B
TRA	B2L,B	56 2A	TRANSFER B2L TO B
TRA	X1L,B	56 4A	TRANSFER X1L TO B
TRA	X2L,B	56 6A	TRANSFER X2L TO B
TRA	B1U,B	56 8A	TRANSFER B1U TO B
TRA	B2U,B	56 AA	TRANSFER B2U TO B
TRA	X1U,B	56 CA	TRANSFER X1U TO B
TRA	X2U,B	56 EA	TRANSFER X2U TO B
DIB	F,D	56 XX	DATA INPUT INTO B
TRA	B,B1L	57 0A	TRANSFER B TO B1L
TRA	B,B2L	57 2A	TRANSFER B TO B2L
TRA	B,X1L	57 4A	TRANSFER B TO X1L
TRA	B,X2L	57 6A	TRANSFER B TO X2L
TRA	B,B1U	57 8A	TRANSFER B TO B1U
TRA	B,B2U	57 AA	TRANSFER B TO B2U
TRA	B,X1U	57 CA	TRANSFER B TO X1U
TRA	B,X2U	57 EA	TRANSFER B TO X2U
DOB	F,D	57 XX	DATA OUTPUT FROM B

MICRO 400 INSTRUCTIONS OP CODE SEQUENCE

INSTRUCTION		OP CODE	DESCRIPTION
SER	ADDR	58 XX	SKIP IF EVEN, ROTATE MEMORY
SOR	ADDR	59 XX	SKIP IF ODD, ROTATE MEMORY
SPM	ADDR	5A XX	SKIP IF POSITIVE MEMORY
SMM	ADDR	5B XX	SKIP IF MINUS MEMORY
RCS	ADDR	5C XX	RIGHT SHIFT, COPY SENSE
RCZ	ADDR	5D XX	RIGHT SHIFT, COPY ZERO
RCO	ADDR	5E XX	RIGHT SHIFT, COPY ONE
RCA	ADDR	5F XX	RIGHT SHIFT, COPY LEAST SIGNIFICANT BIT OF A
LDA	BIAS,1	60 XX	LOAD A INDEXED BY X1
LDA	BIAS,2	61 XX	LOAD A INDEXED BY X2
LDB	BIAS,1	62 XX	LOAD B INDEXED BY X1
LDB	BIAS,2	63 XX	LOAD B INDEXED BY X2
STA	BIAS,1	64 XX	STORE A INDEXED BY X1
STA	BIAS,2	65 XX	STORE A INDEXED BY X2
STB	BIAS,1	66 XX	STORE B INDEXED BY X1
STB	BIAS,2	67 XX	STORE B INDEXED BY X2
LDA	BIAS,1,D	68 XX	LOAD A INDEXED BY X1 DECREMENT X1
LDA	BIAS,2,D	69 XX	LOAD A INDEXED BY X2 DECREMENT X2
LDB	BIAS,1,D	6A XX	LOAD B INDEXED BY X1 DECREMENT X1
LDB	BIAS,2,D	6B XX	LOAD B INDEXED BY X2 DECREMENT X2
STA	BIAS,1,D	6C XX	STORE A INDEXED BY X1 DECREMENT X1
STA	BIAS,2,D	6D XX	STORE A INDEXED BY X2 DECREMENT X2
STB	BIAS,1,D	6E XX	STORE B INDEXED BY X1 DECREMENT X1
STB	BIAS,2,D	6F XX	STORE B INDEXED BY X2 DECREMENT X2
LDA	BIAS,1,I	70 XX	LOAD A INDEXED BY X1 INCREMENT X1
LDA	BIAS,2,I	71 XX	LOAD A INDEXED BY X2 INCREMENT X2
LDB	BIAS,1,I	72 XX	LOAD B INDEXED BY X1 INCREMENT X1
LDB	BIAS,2,I	73 XX	LOAD B INDEXED BY X2 INCREMENT X2
STA	BIAS,1,I	74 XX	STORE A INDEXED BY X1 INCREMENT X1
STA	BIAS,2,I	75 XX	STORE A INDEXED BY X2 INCREMENT X2
STB	BIAS,1,I	76 XX	STORE B INDEXED BY X1 INCREMENT X1
STB	BIAS,2,I	77 XX	STORE B INDEXED BY X2 INCREMENT X2
LDA	ADDR	8X XX	LOAD A FROM MEMORY
LDB	ADDR	9X XX	LOAD B FROM MEMORY
STA	ADDR	AX XX	STORE A INTO MEMORY
STB	ADDR	BX XX	STORE B INTO MEMORY
JPC	ADDR	CX XX	JUMP WITHIN CURRENT 4K PAGE
JPE	ADDR	DX XX	JUMP TO EXTENDED 4K PAGE
JMC	ADDR	EX XX	JUMP & MARK IN CURRENT 4K PAGE
JME	ADDR	FX XX	JUMP & MARK TO EXTENDED 4K PAGE

APPENDIX B

I/O INTERFACE

APPENDIX B

1. INTRODUCTION

This document describes the Micro 400 I/O Interface and its operation, timing diagrams are given to assist in understanding it.

1.1 INPUT/OUTPUT BUS

The Micro 400 communicates with peripheral device options on a word parallel I/O bus as shown in Figures B-2 and B-3.

Information exchanges with peripheral devices are normally synchronized and sequenced by a peripheral device controller. For discussion in this manual, a peripheral device option includes not only the device but the controller which is necessary to the operation of that device.

The standard I/O bus consists of the E-bus, the I-bus, the C-bus and five control lines: I/O Enable (IOEN), I/O Strobe (IOST), Initialize (INIT), Interrupt (IREQ), and Interrupt Acknowledge (IACK).

2.0 FUNCTIONAL DESCRIPTION

2.1 INTERFACE DEFINITION

2.1.1 Signal Buses

E-Bus: an 8-bit bidirectional data bus that is used to transfer data to or from the controller.

The E-bus is active only during IOEN.

C-Bus: an 8-bit unidirection bus that is used to transfer the device address CB00 → CB04 and function code information CB05 → CB07 contained in the instruction being executed to the controller. The contents of the C-bus are meaningful only during the time of IOEN.

F-Bus: a 2-bit unidirection bus that is used to transfer the command states data input transfer, data output transfer, execute, and sense information contained in the instruction being executed to the controller. The contents of the F-bus are meaningful only during the time of IOEN.

2.1.2 Control Lines

IOEN: The I/O enable signal is 800 nanoseconds wide and is used as a window during which time typical controller action takes place. All other time periods are indeterminate.

IOST: The I/O strobe signal is 100 nanoseconds wide and occurs 600 nanoseconds after the start of IOEN. It is used as follows:

a. Data on the E-bus to the controller is strobed into storage registers on the trailing edge of IOST.

b. Data on the E-bus from the controller is strobed into the CPU on the leading edge of IOST.

c. An indication that the sense response has been sampled.

d. Execution of external control functions should be accomplished with IOST.

INIT: The initialize signal is used to reset the controllers connected to the I/O. It is activated by the front panel reset switch or other options which control reset.

CLCK: A 5 MHz, 50% duty cycle, clock useful for timing and setting control functions in the controller. The phase relationship to IOEN is always the same.

2.1.3 Priority Line

PROT: Priority out line. A Low (0V) level which indicates that the first controller attached can make an interrupt or BATIO request.

2.1.4 Interrupt Lines

IREQ: Interrupt Request line. A Low (0V) indicates an interrupt request is being made. It is conditioned in the controller by the priority chain and a interrupt enable/disable flip-flop.

IACK: **Interrupt Acknowledge** line. A Low (0V) indicates an interrupt acknowledge is being made and that the controller should place its device address times 2 on the E-bus.

2.2 I/O COMMAND TIMING DESCRIPTION

The MICRO 400 I/O structure provides four basic types of I/O commands for control of peripheral device options. These are:

a. Data Input Transfer

b. Data Output Transfer

c. External Control

d. Sense

As can be seen by examination of the I/O timing diagram in Figure 2.1, all input/output operations occur in the same manner and have identical timing diagrams.

The following paragraphs discuss the operations associated with each command type.

> NOTE: Logic levels as discussed in the following sections are as they appear on the bus. The bus is ground true, + 5 false.

2.2.1 Data Input Transfer

The timing diagram in Figure B-1 can be used to understand the basic transfer of data into the MICRO 400 from an external device. When a DIA (or DIB) is executed the MICRO 400 begins to communicate with the addressed controller at time T_0.

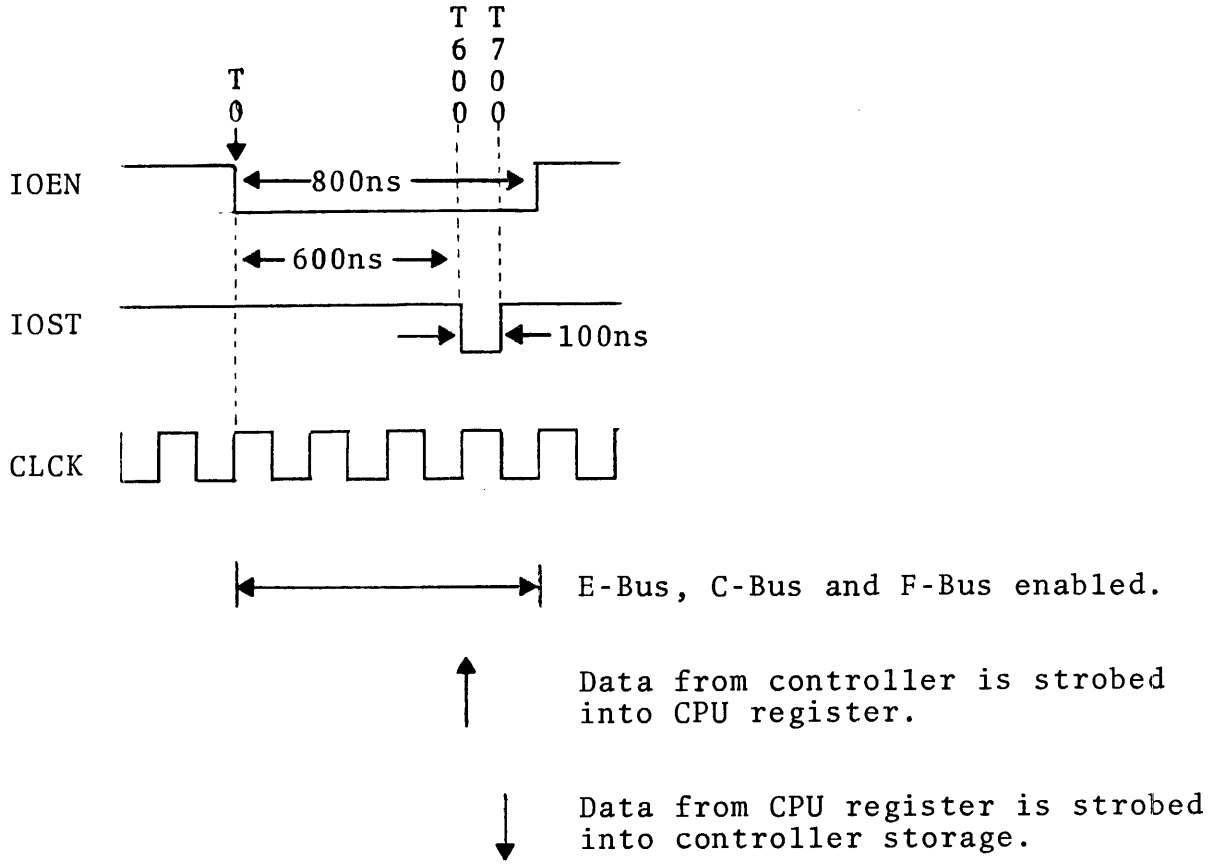
At T_0 the C-bus is driven by the MICRO 400 with the portion of the DIA command word that contains device address (CB00 → CB04) and control code (CB05 → CB07) information.

At T_0 time the I-bus (IB00, IB01) is driven with the portion of the DIA command word that defines that this is to be a command, in this instance IB00 will be true and IB01 will be false.

At T_0 the IOEN signal is driven and alerts the addressed device controller to place data on the E-bus.

From time T_0 to T_{800} the E-bus is driven by the addressed device controller with the data to be transferred to the MICRO 400. It must be stable by time T_{600}.

At T_{600} → T_{700} IOST is driven. In the case of DIA, the IOST informs the external device that the incoming data has been sampled. Thus it may be used for any housekeeping chore required by the peripheral device controller.



NOTE: Logic levels shown are as they appear on the I/O bus. Ground true, +5 false.

FIGURE B-1

At T_{800} the E-bus, I-bus, C-bus, and IOEN are removed from the I/O bus.

2.2.2 Date Output Transfer

The timing diagram in Figure B-1 can be used to understand the basic transfer of data out of the MICRO 400 and to a peripheral device controller. Assume that this is a DOA from register A, and that it already contains the data we intend to transfer out. When a DOA (or DOB) is executed the MICRO 400 begins to communicate with addressed controller at time T_0.

At T_0 the C-bus is driven with the portion of the DOA command word that contains device address (CB00 → CB04) and control code (CB05 → CB07) information.

At T_0 time the I-bus (IB00, IB01) is driven with the portion of the DOA command word that defines that this is to be a command, in this instance IB00 will be false and IB01 will be false.

At T_0 the IOEN signal is driven and alerts the addressed device controller to prepare to accept the data on the E-bus.

At T_0 the E-bus is driven with the data contained in register A.

At T_{600} → T_{700} IOST is driven. In the case of DOA, the IOST should be used to strobe the data on the E-bus into a holding register in the peripheral device controller.

At T_{800} the E-bus, F-bus, C-bus, and IOEN are removed from the I/O bus.

2.2.3 External Control (EXC)

The timing diagram in Figure B-1 can be used to understand the basic external control command sequence. The EXC will cause a prescribed action to take place inside the peripheral device controller or possibly an action inside the peripheral device; i.e., read cards, rewind tape, write EOF, etc.

At T_0 the C-bus is driven with the portion of the EXC command word that contains device address (CB00 → CB04) and control code (CB05 → CB07) information.

At T_0 time the F-bus (FB00, FB01) is driven with the portion of the EXC command word that defines that this is to be a command, in this instance FB00 is false and FB01 is true.

At T_0 the IOEN signal is driven and alerts the addressed controller to interrogate the C-bus for control codes.

No action on the E-bus should occur during the EXC.

At $T_{600} \rightarrow T_{700}$ IOST is driven. In the case of EXC, it may be used to strobe the external code generated by the decode of CB05, CB06, and CB07 during IOEN time.

At T_{800} the C-bus, F-bus, and IOEN are removed from the I/O bus.

2.2.4 Sense (SEN)

The timing diagram in Figure B-1 can be used to understand the basic Sense command sequence. The SEN command will cause a signal from the controller to be interrogated. The condition status will cause EB00 to be driven and the MICRO 400 to respond accordingly.

At T_0 the C-bus is driven with the portion of the SEN command word that contains device address (CB00 → CB04) and control code (CB05 → CB07) information

At T_0 time the F-bus (FB00, FB01) is driven with the portion of the SEN command word that defines that this is to be a command, in this instance FB00 is true and FB01 is true.

At T_0 the IOEN signal is driven and alerts the addressed controller to interrogate the condition and drive the signal EB00 accordingly.

At T_0 the E-bus is enabled but the only action that should occur on the E-bus during the SEN is the condition on EB00 placed there by the peripheral device.

At $T_{600} \rightarrow T_{700}$ IOST is driven. In the case of SEN, the IOST serves no specific purpose in the device controller. It may be used for any housekeeping chore required by the device controller.

At T_{800} the E-bus, C-bus, F-bus and IOEN are removed from the I/O bus.

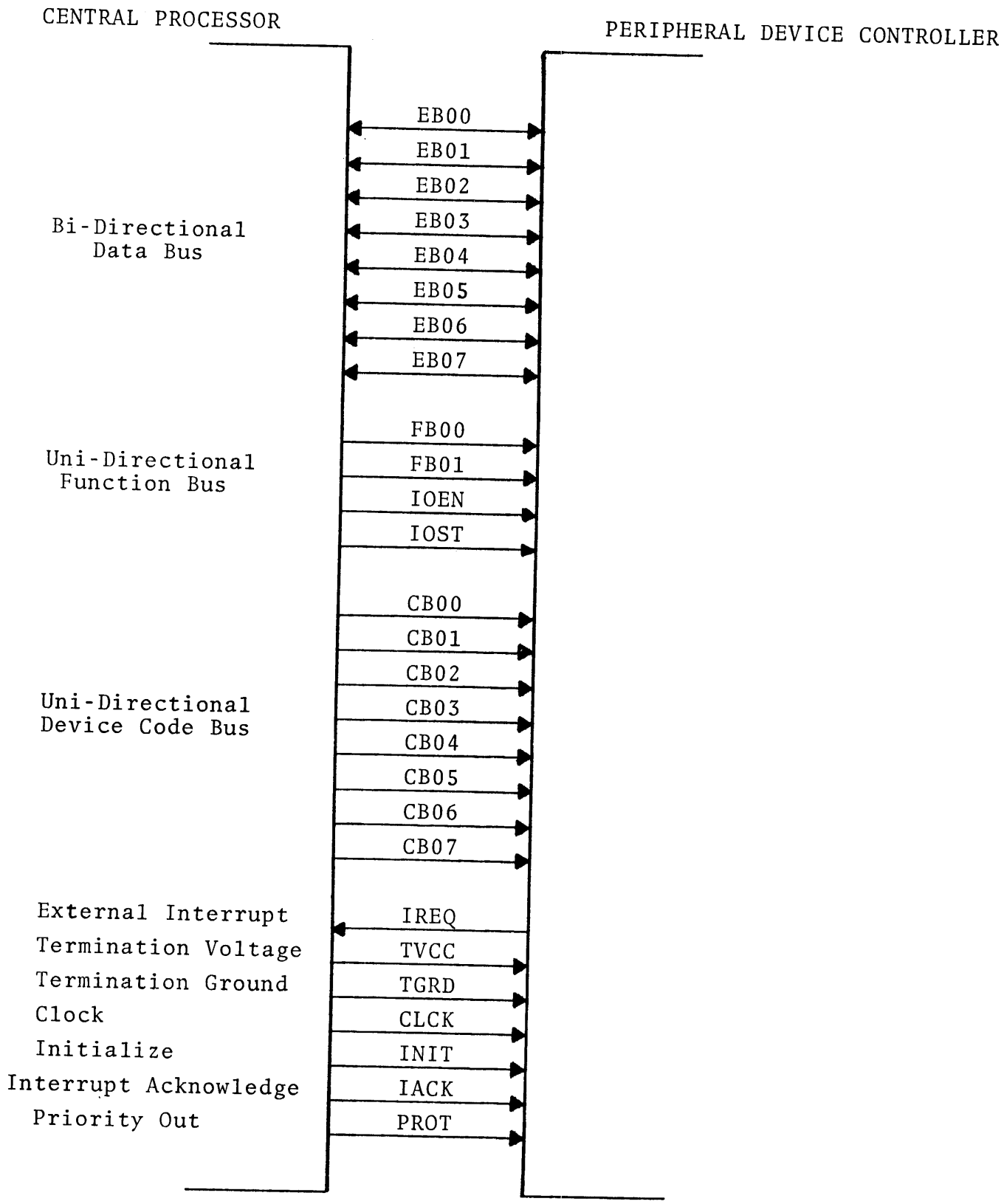


FIGURE B-2

STANDARD INPUT/OUTPUT DATA BUS

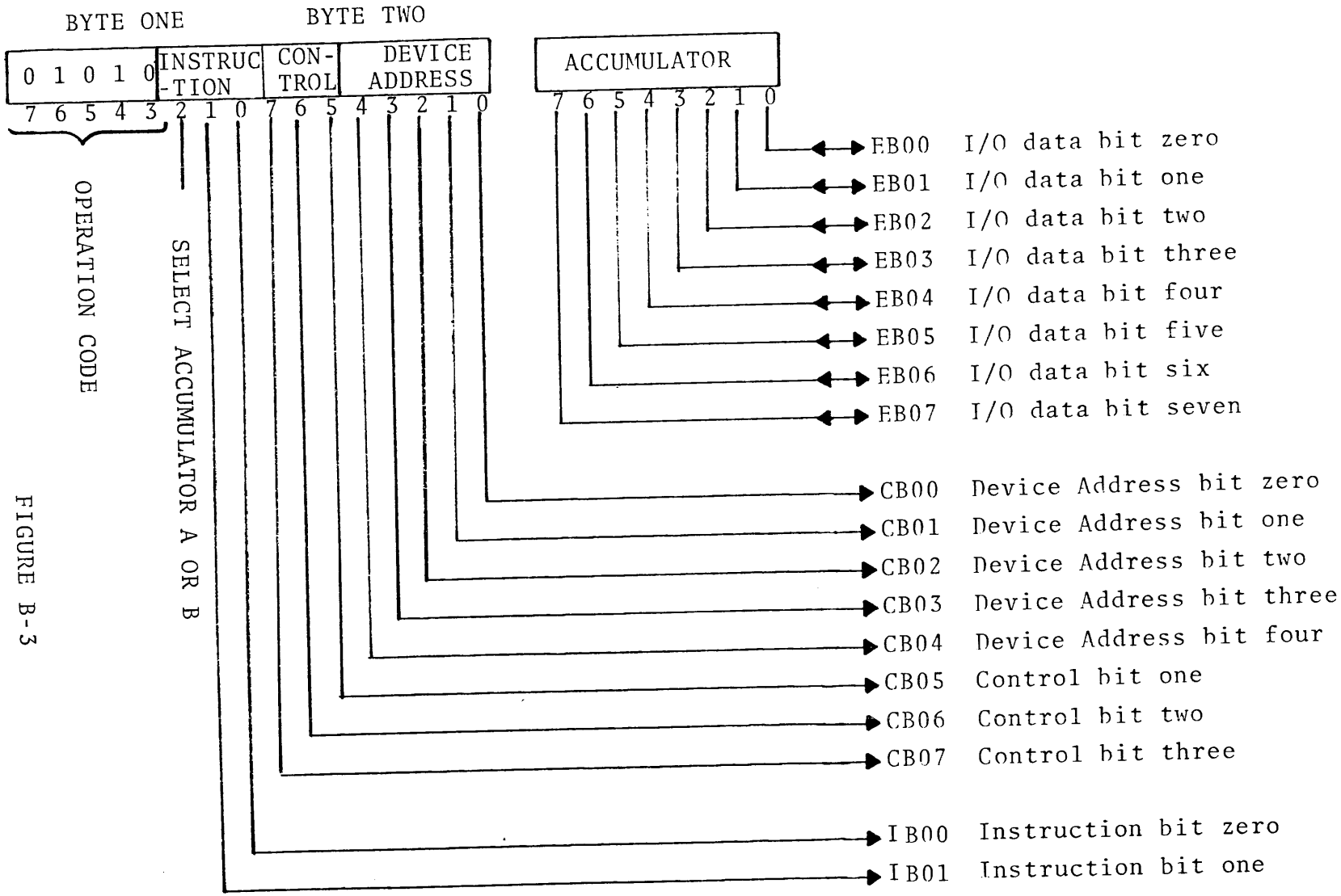


FIGURE B-3

Input/Output instruction and accumulator relationship to the Input/Output bus.
Logic levels appearing on the bus will be ground true(Ie ground = 1)
(+5 = 0)

2.3 PRIORITY DETERMINATION

Relative priority is determined by a hardwired priority chain consisting of the Priority IN (PRIN) and Priority Out (PROT) lines. This priority chain determines the priority for interrupt and Block Automatic transfer I/O (BATIO). Before a controller can make an interrupt or BATIO request, it must receive priority from the next higher priority controller on the chain in the form of a ground signal on the PRIN line. A Low PRIN (0V) indicates the controller can make a request. If the controller does not make a request, it makes PROT Low (0V). PROT is Low (0V) from the I/O board.

2.4 INTERRUPT

2.4.1 Interrupt Request

Interrupt request (IREQ) is conditioned by three terms: (a) the interrupt enable/disable flip-flop in the controller, (b) the priority in (PRIN) in the controller, and (c) the interrupt enable/disable flip-flop in the CPU. If both the CPU and controller interrupt flip-flop are enabled and PRIN is low, a controller can make IREQ low and it will receive the Interrupt Acknowledge within a maximum of 4 3/4 cycles (7.6 usec).

2.4.2 Interrupt Acknowledge

When the interrupt acknowledge (IACK) is received it indicates that the CPU is making an instruction fetch at memory locations hex 00 and hex 01 unless modified by the controller address. When the controller receives IACK it places its device address times 2 on the E-bus during the entire time of IACK. This action will modify the memory locations that the instruction is being fetched from. For example, if the controller device address is hex 08 the two memory locations fetched will be hex 10 and hex 11. This interrupt location contains a jump and mark instruction which will provide the exit to the interrupt routine and provide for the return to the normal program.

Interrupt Timing

Conditions: CPU and controller interrupt enable flip-flops are set.

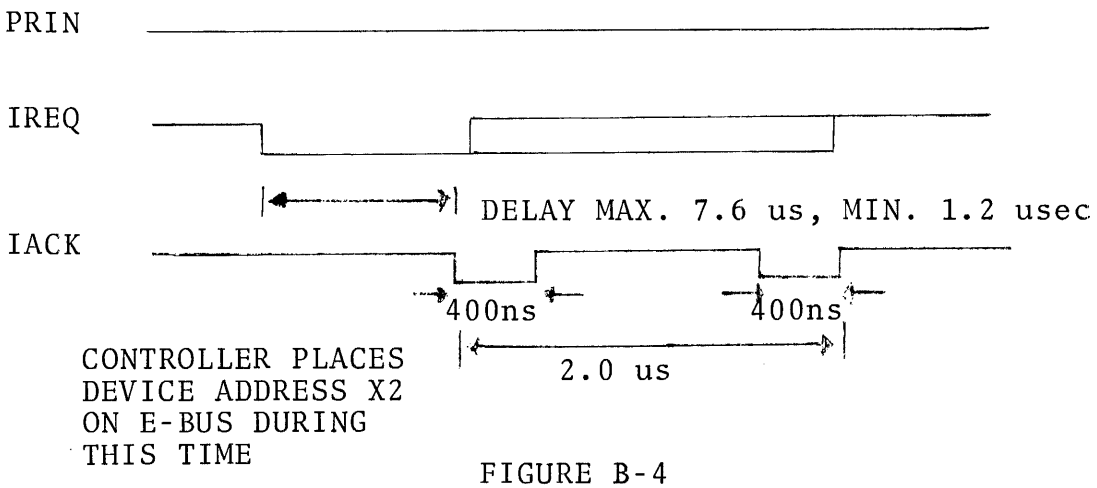
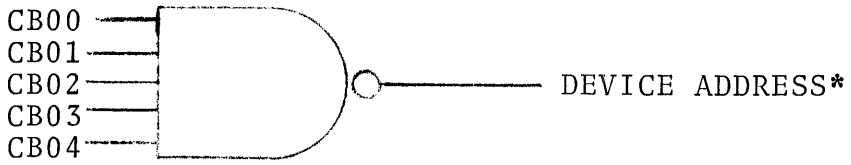
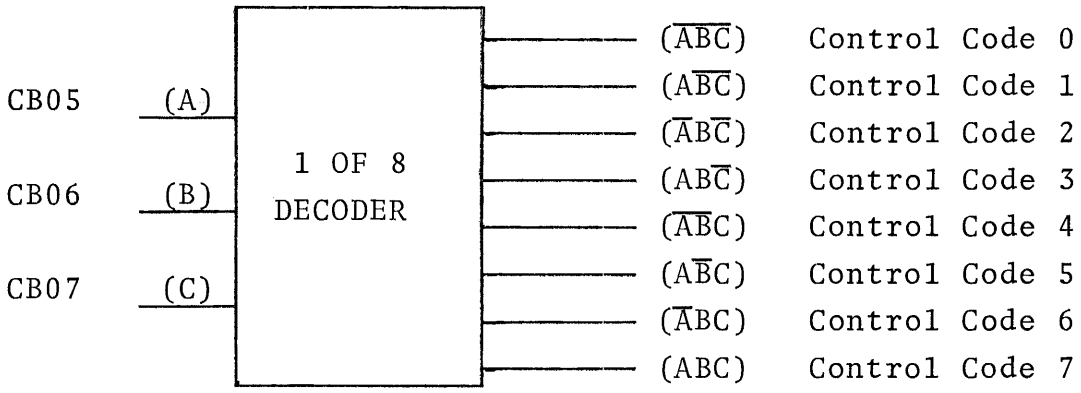


FIGURE B-4

NOTE: Logic levels shown are as they appear on the I/O bus. Ground true, + 5 volts false.

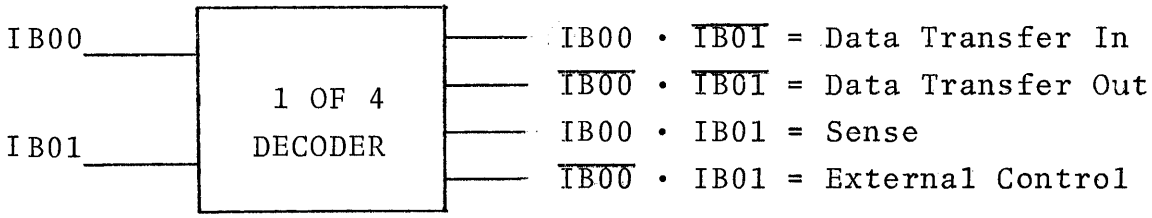


DEVICE ADDRESS DECODE



CONTROL CODE DECODE

The IB00 and IB01 signals as they appear on the I/O Bus produce the following equations (ground true).



COMMAND CODE DECODE

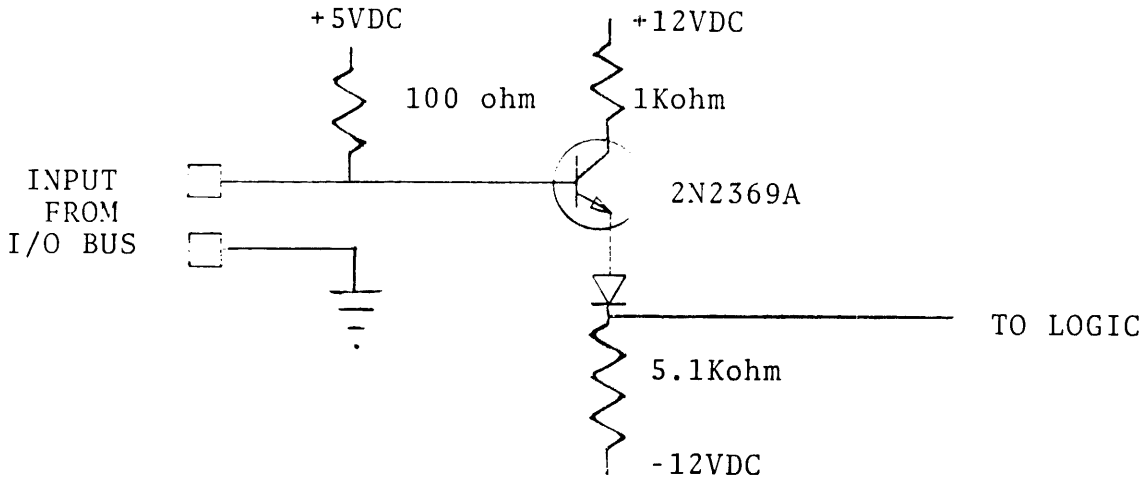
---

*The device address is a predetermined binary code and the detecting gate structure must be connected for proper decoding.

FIGURE B-7

BASIC CONTROLLER DECODING

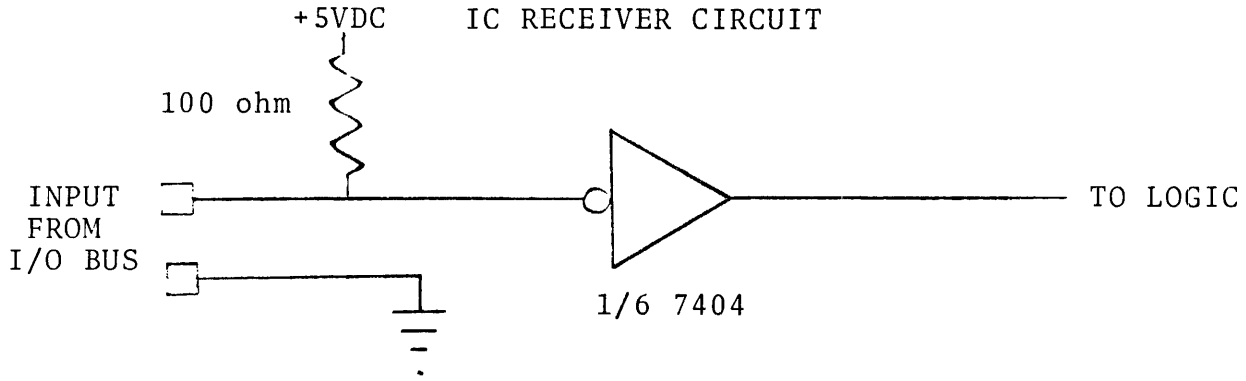
DISCRETE RECEIVER CIRCUIT



This discrete circuit has very desirable noise rejection characteristics for a long cable run and is recommended for any cable length exceeding twenty feet. This circuit is non-inverting.

FIGURE B-8

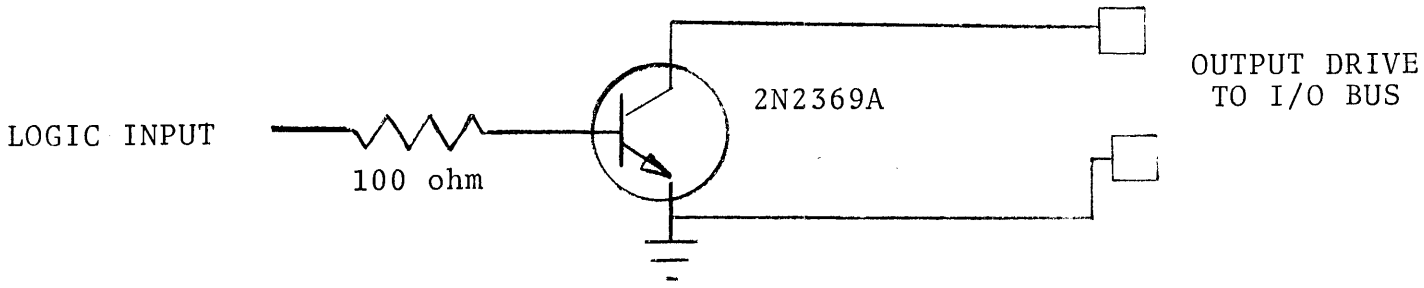
IC RECEIVER CIRCUIT



This receiver circuit is to be used only when the I/O bus cable length is less than twenty feet. It is an inverting circuit.

FIGURE B-8

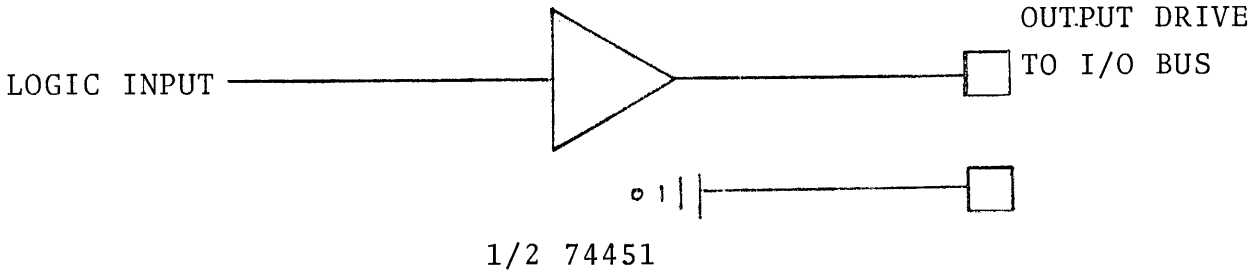
DISCRETE DRIVER CIRCUIT



This driver is relatively insensitive to the affects of cable length and is recommended for cable runs over twenty feet. The driver inverts the signal into it, therefore to achieve ground true on the bus the input must be +5 volt true.

FIGURE B-9

IC DRIVER CIRCUIT



The IC driver is recommended for cable runs of twenty feet or less. The IC driver does not invert the signal, therefore ground true in results in ground true out.

FIGURE B-10

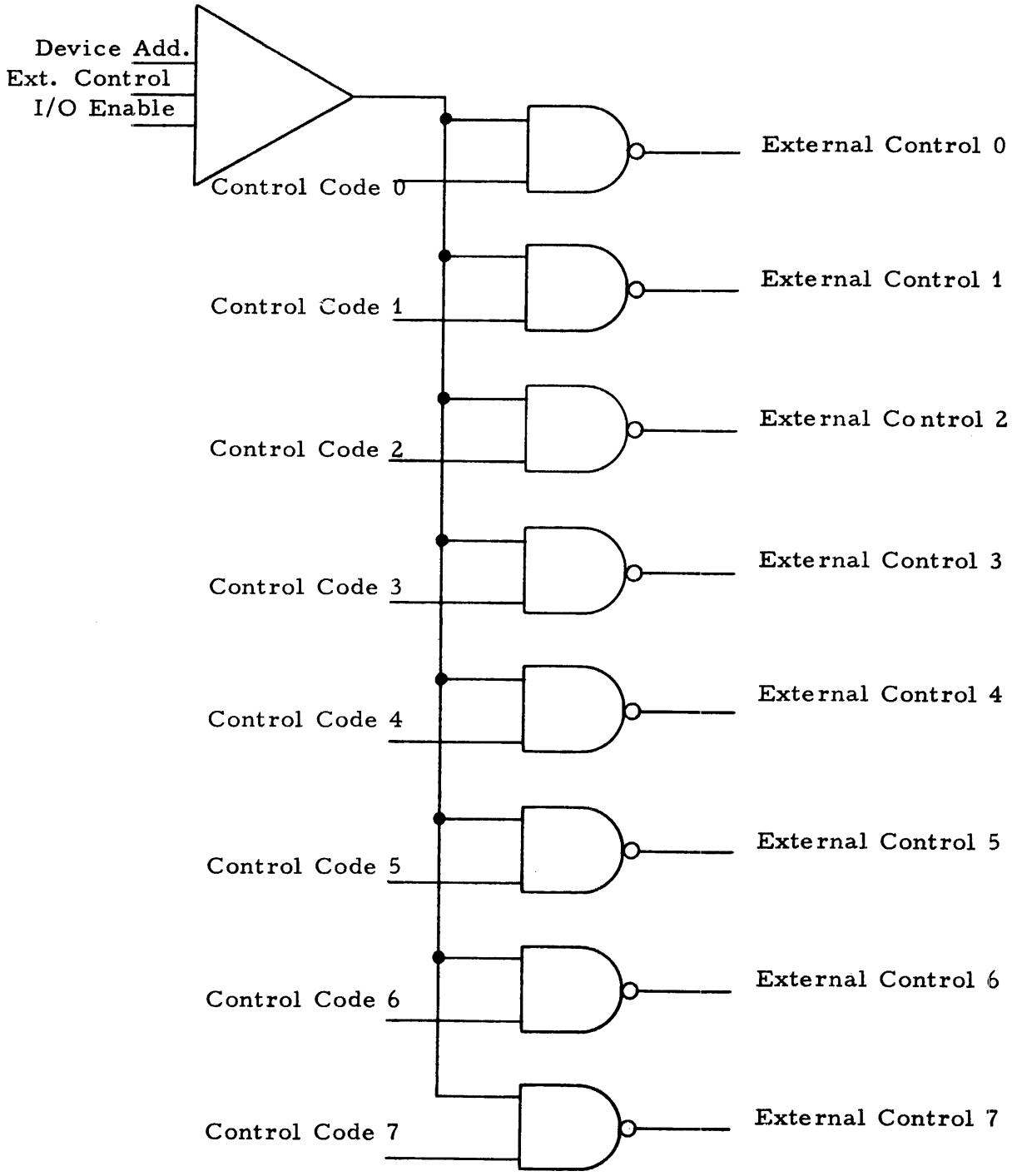
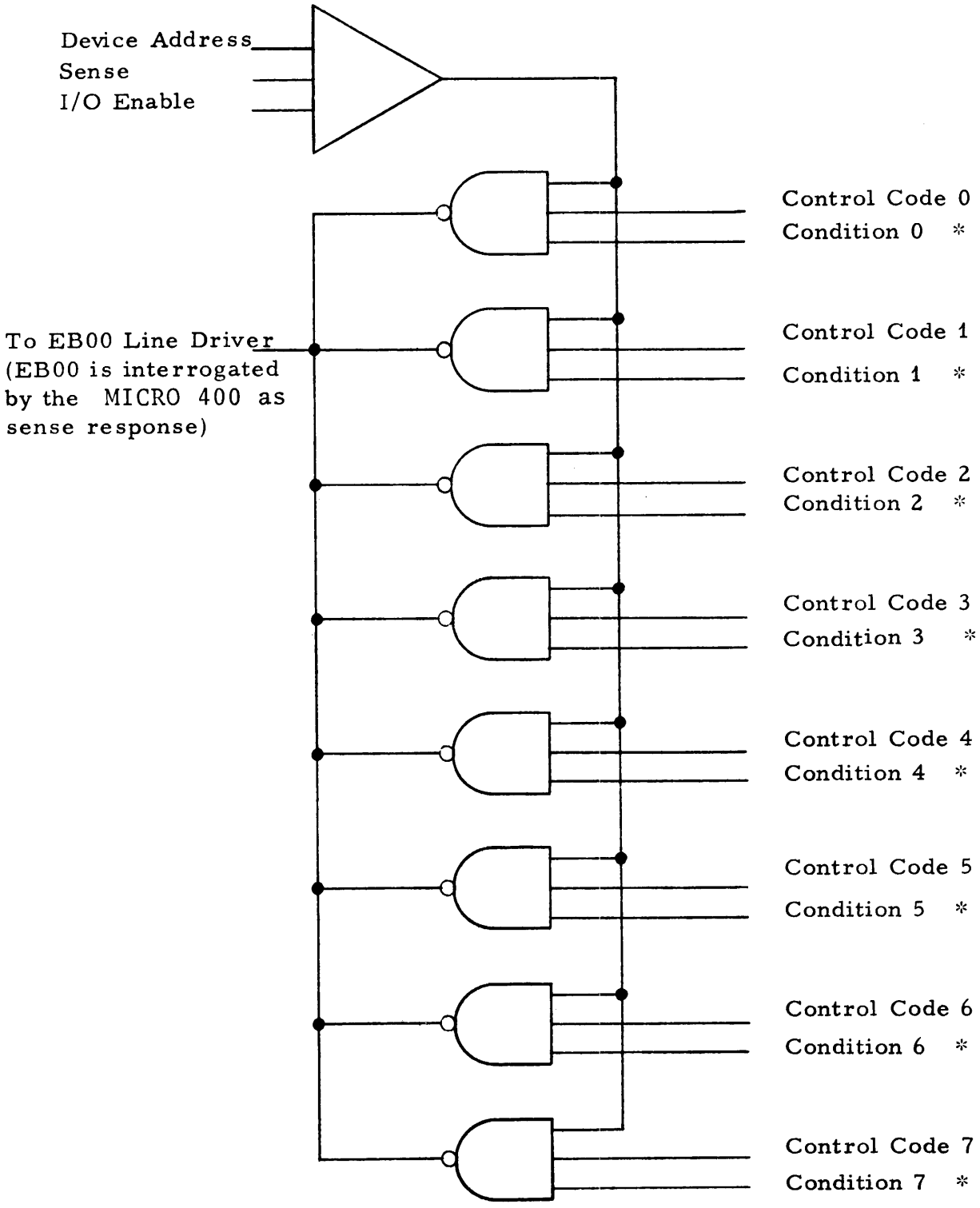


FIGURE B-11

EXTERNAL CONTROL COMMAND



* These are controller functions to be sensed; i.e. beginning of tape, ready, not ready, etc.

FIGURE B-12 - SENSE COMMAND

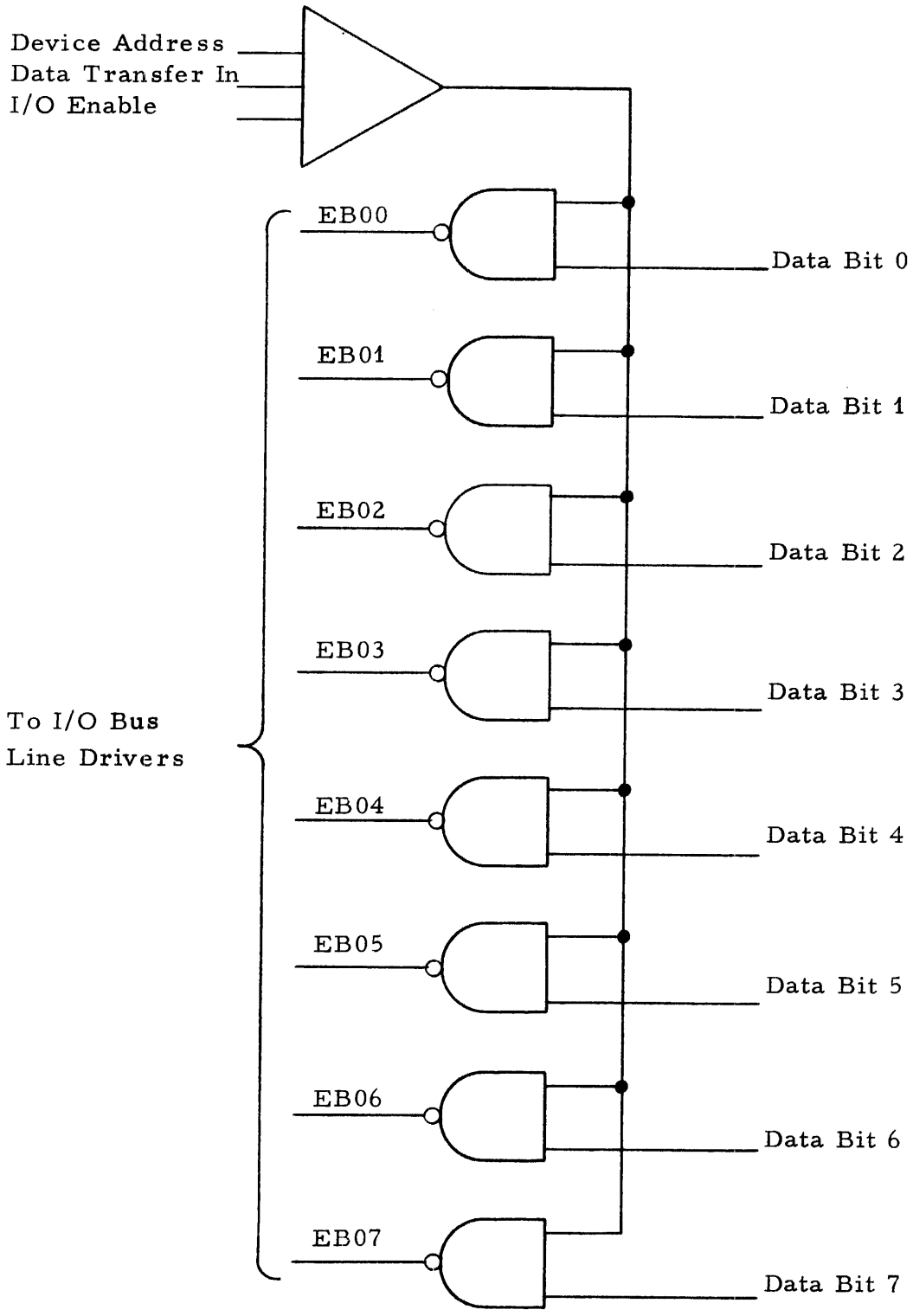
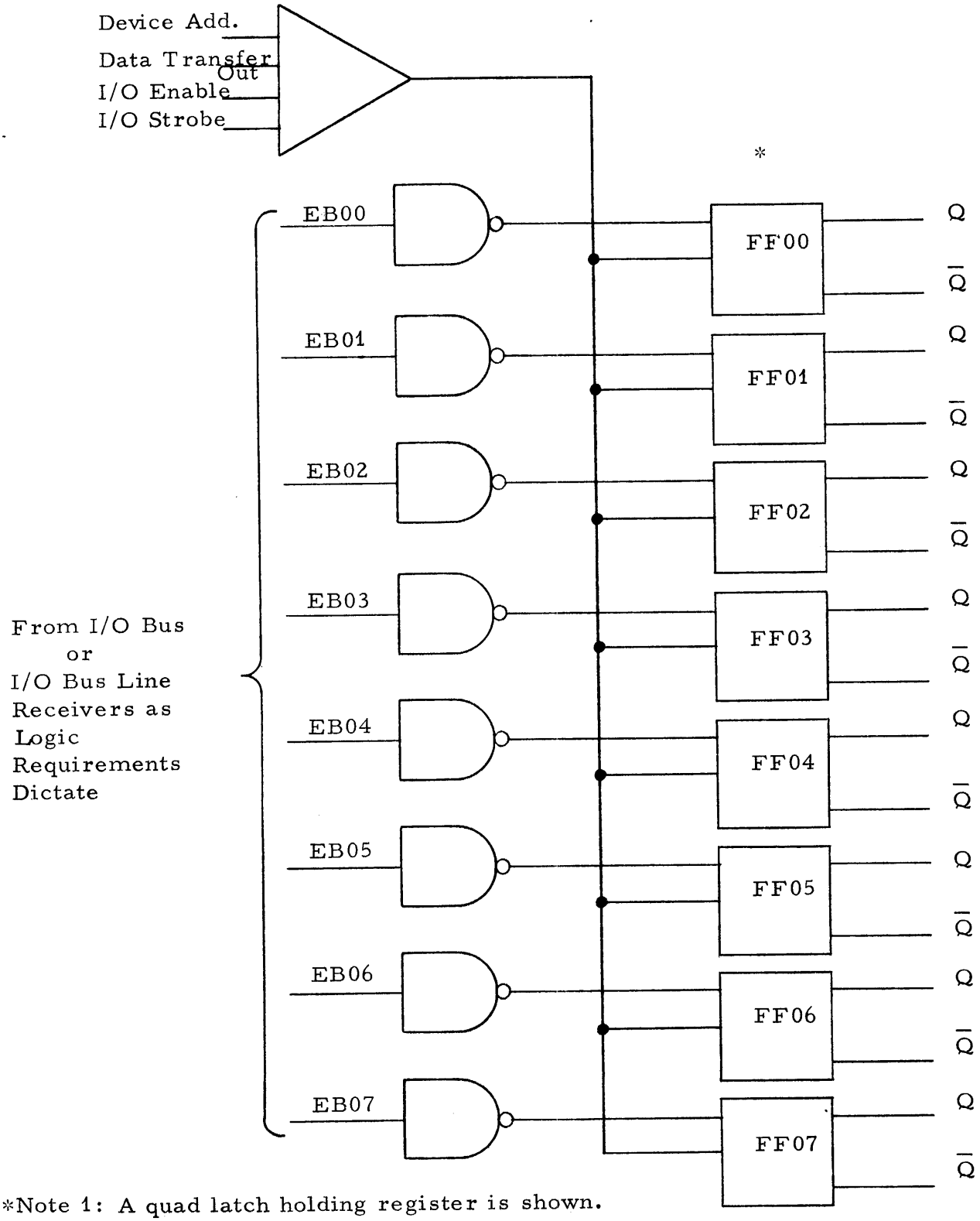


FIGURE B-13

DATA INPUT COMMAND



*Note 1: A quad latch holding register is shown.

FIGURE B-14

DATA OUTPUT COMMAND

APPENDIX C

LEFT BLANK INTENTIONALLY

APPENDIX D

CONNECTOR LISTING

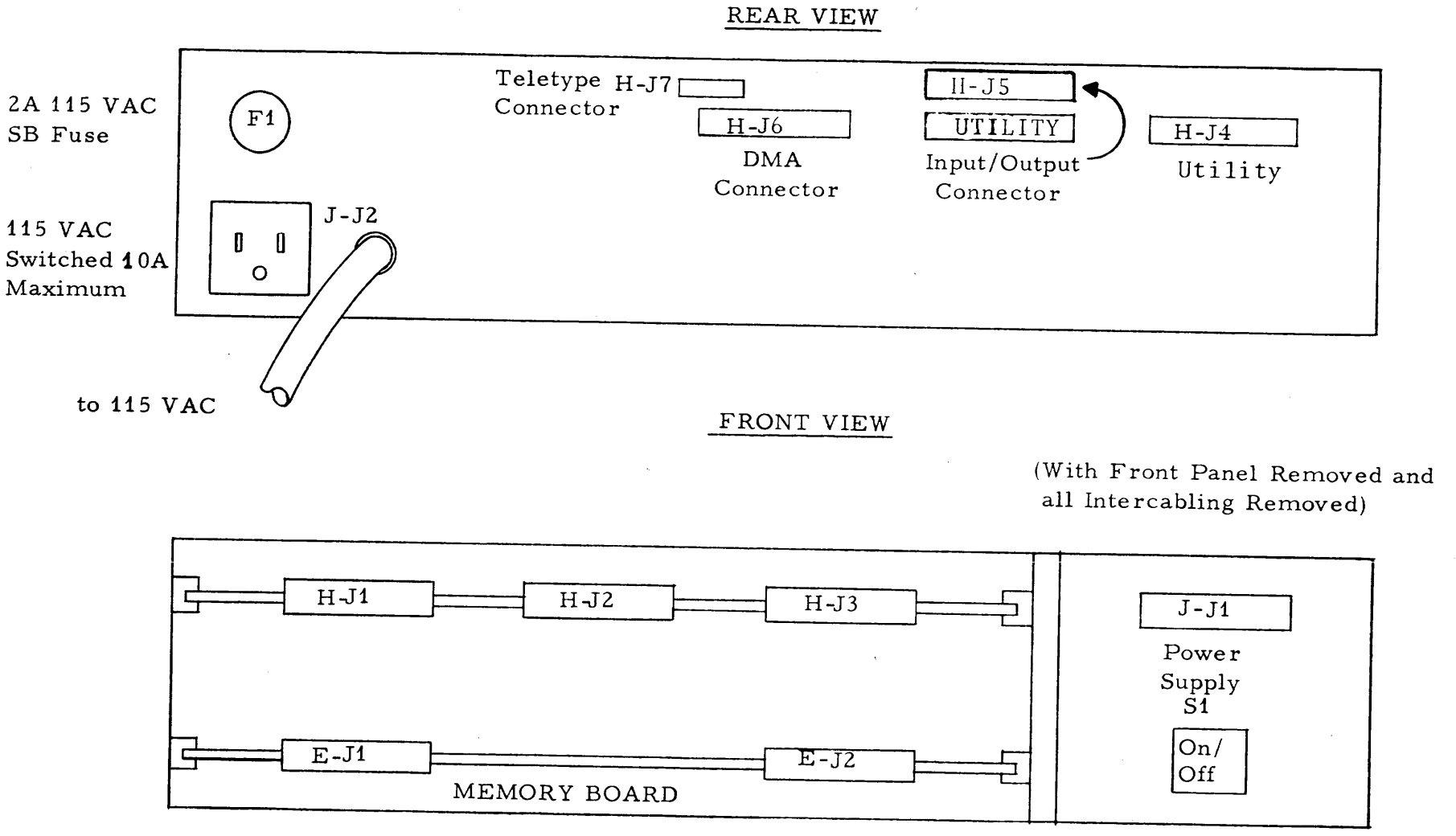


FIGURE D-1

M400 CPU CONNECTOR LOCATOR

FUNCTION: Connect CPU H-J2 to Front Panel F-J1 & F-J2

NAME: Control Panel Connector

Pin No.	Mnemonic	Description
01	AB03-	ADDRESS BIT 03
02	AB00-	ADDRESS BIT 00
03	PB00+	P REG BIT 0 ENABLE TERM
04	PHA2-	TIMING TERM
05	SPARE	
06	12 VAC	LAMP VOLTAGE
07	12 VAC	LAMP VOLTAGE RETURN
08	GRD	GROUND
09	GRD	GROUND
10	+5V	+5 VDC
11	OVFF+	OVERFLOW FLIP-FLOP
12	RUNF-	RUN FLIP-FLOP
13	DB03-	DATA BIT 03
14	SS01+	SENSE SWITCH 01
15	DB00-	DATA BIT 00
16	DB02-	DATA BIT 02
17	DPEN-	DISPLAY P REG ENABLE
18	RUNE-	RUN FLIP-FLOP
19	STPE-	STEP FLIP-FLOP
20	MEST-	MEMORY ENTER IN STEP
21	LDAD-	DISPLAY A ENABLE
22	LBEN-	LOAD B ENABLE
23	LAEN-	LOAD A ENABLE
24	LDBD-	DISPLAY B ENABLE
25	RSET-	RESET CPU IN STEP

Pin No.	Mnemonic	Description
26	AB01-	ADDRESS BIT 01
27	AB02-	ADDRESS BIT 02
28	PB02+	P REGISTER BIT 02
29	PB03+	P REGISTER BIT 03
30	PB01+	P REGISTER BIT 01
31	12 VAC	LAMP VOLTAGE
32	12 VAC	LAMP VOLTAGE RETURN
33	GRD	GROUND
34	GRD	GROUND
35	+5V	+5 VDC
36	TIMA-	TIMING TERM
37	SPARE	
38	RUNF+	RUN FLIP-FLOP
39	DB01-	DATA BIT 01
40	DB04-	DATA BIT 04
41	DB05-	DATA BIT 05
42	SS02+	SENSE SWITCH 02
43	DB07-	DATA BIT 07
44	DB06-	DATA BIT 06
45	12 VAC	LAMP VOLTAGE
46	CRYF+	CARRY FLIP-FLOP
47	DAIS-	DATA IN STROBE TO MEMORY
48	LPUE-	LOAD P REG ENABLE
49	DSPF-	DISPLAY P REG
50	CLWR-	CLEAR WRITE ENABLE

FIGURE D-2

CPU CONNECTOR H-J2

FUNCTION: Connect CPU H-J3 to Power Supply J-J1

NAME: Power Connector

Pin No.	Mnemonic	Description
01	GRD	
02	↑	
03		
04		Logic Ground
05		
06		
07		
08	↓	
09	GRD	
10	PFSN	Power Fail Sense
11	DASV	Data Save
12	+5V	
13	+5V	
14	+5V	
15	+5V	Logic Voltage +5 VDC
16	+5V	
17	+5V	
18	+5V	
19	REMS	Remote Switch
20	+12V	
21	+12V	Memory Voltage +12 VDC
22	+12V	
23	+12.6AC	Lamp Voltage Return
24	+12.6AC	Lamp Voltage
25	-12V	Memory Voltage -12 VDC

Pin No.	Mnemonic	Description
26	GRD	
27	↑	
28		
29		
30		Logic Ground
31		
32		
33	↓	
34	GRD	
35	PFSN	Power Fail Sense
36	DASV	Data Save
37	+5V	
38	+5V	
39	+5V	
40	+5V	Logic Voltage +5 VDC
41	+5V	
42	+5V	
43	+5V	
44	REMS	Remote Switch
45	+12V	
46	+12V	Memory Voltage +12 VDC
47	+12V	
48	+12.6AC	Lamp Voltage Return
49	+12.6AC	Lamp Voltage
50	-12V	Memory Voltage -12 VDC

Figure D-3

CPU CONNECTOR H-J3, MEMORY CONNECTOR E-J2 & POWER SUPPLY CONNECTOR J-J1

FUNCTION: DIRECT MEMORY ACCESS HJ6

Pin No.	Mnemonic	Description	Pin No.	Mnemonic	Description
01	+5VDC	Terminating Voltage	26	SPECIAL	
02	+5VDC	Terminating Voltage	27	GROUND	
03	+5VDC	Terminating Voltage	28	DM11	Memory Address Bit 11
04	+5VDC	Terminating Voltage	29	GROUND	
05	GROUND		30	DM10	Memory Address Bit 10
06	DM15	Memory Address Bit 15	31	GROUND	
07	GROUND		32	DM09	Memory Address Bit 09
08	DM14	Memory Address Bit 14	33	GROUND	
09	GROUND		34	DM08	Memory Address Bit 08
10	DM13	Memory Address Bit 13	35	GROUND	
11	GROUND		36	DMRQ	DMA Request
12	DM12	Memory Address Bit 12	37	GROUND	
13	GROUND		38	DMIS	Data to Memory Strobe
14	SPARE		39	GROUND	
15	GROUND		40	DMCR	Clear Write Enable
16	SPARE		41	GROUND	
17	GROUND		42	DMMS	Memory Start Pulse
18	SPARE		43	GROUND	
19	GROUND		44	DMOS	Data From Memory Enable
20	SPARE		45	GROUND	
21	GROUND		46	DAIG	Data to Memory Enable
22	SPARE		47	GROUND	
23	GROUND		48	ADLD	Address Load Enable
24	SPECIAL		49	GROUND	
25	GROUND		50	DMRS	CPU DMA Acknowledge

FIGURE D-4

DMA CONNECTOR HJ6

FUNCTION: INPUT/OUTPUT

Pin No.	Mnemonic	Description	Pin No.	Mnemonic	Description
01	+5 VOLTS	Terminating Voltage	26	IOST	I/O Strobe
02	+5 VOLTS	Terminating Voltage	27	GRND	Ground
03	+5 VOLTS	Terminating Voltage	28	IOEN	I/O Enable
04	+5 VOLTS	Terminating Voltage	29	GRND	Ground
05	IACK	Interrupt Acknowledge	30	FBO1	Function Bit 01
06	GRND	Ground	31	GRND	Ground
07	PROT	Priority Out	32	FBOO	Function Bit 00
08	CLCK	5 Mhz Clock	33	GRND	Ground
09	GRND	Ground	34	EBO7	Data Bus Bit 07
10	CBO7	Control Bit 3	35	GRND	Ground
11	GRND	Ground	36	EBO6	Data Bus Bit 06
12	CBO6	Control Bit 2	37	GRND	Ground
13	GRND	Ground	38	EBO5	Data Bus Bit 05
14	CBO5	Control Bit 1	38	GRND	Ground
15	GRND	Ground	40	EBO4	Data Bus Bit 04
16	CBO4	Device Address Bit 04	41	GRND	Ground
17	GRND	Ground	42	EBO3	Data Bus Bit 03
18	CBO3	Device Address Bit 03	43	GRND	Ground
19	GRND	Ground	44	EBO2	Data Bus Bit 02
20	CBO2	Device Address Bit 02	45	GRND	Ground
21	GRND	Ground	46	EBO1	Data Bus Bit 01
22	CBO1	Device Address Bit 01	47	GRND	Ground
23	GRND	Ground	48	EBOO	Data Bus Bit 00
24	CBOO	Device Address Bit 00	49	GRND	Ground
25	GRND	Ground	50	IREQ	Interrupt Request

Figure D-5

INPUT/OUTPUT CONNECTOR HJ5

FUNCTION; Connect CPU HJ1 to Memory EJ1

NAME: Memory Data Bus Connector

Pin No.	Mnemonic	Description	Pin No.	Mnemonic
01	SPARE		26	GRD
02	BRME	Special Ext. Address	27	
03	SP13	Special Ext. Address	28	
04	SPARE		29	
05	DASV	Data Save	30	
06	BUSY	Memory Busy	31	
07	SKL2	Memory Modify	32	
08	SKR1	Memory Modify	33	
09	SKR0	Memory Modify	34	
10	DAIS	Data In Strobe	35	
11	DAOS	Data Out Strobe	36	
12	CLWR	Clear Write	37	
13	MSTR	Memory Start	38	
14	AB00	Address Bit 8	39	
15	AB01	Address Bit 9	40	
16	AB02	Address Bit 10	41	
17	AB03	Address Bit 11	42	
18	DB00	Data Bit 0,Address Bit 0	43	
19	DB01	Data Bit 1,Address Bit 1	44	
20	DB02	Data Bit 2,Address Bit 2	45	
21	DB03	Data Bit 3,Address Bit 3	46	
22	DB04	Data Bit 4,Address Bit 4	47	
23	DB05	Data Bit 5,Address Bit 5	48	
24	DB06	Data Bit 6,Address Bit 6	49	
25	DB07	Data Bit 7,Address Bit 7	50	GRD

Figure D-6

CPU CONNECTOR HJ1 AND MEMORY CONNECTOR EJ1

APPENDIX E

CONVERSION TABLES

HEXADECIMAL ARITHMETIC

ADDITION TABLE

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
1	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10
2	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11
3	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12
4	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13
5	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14
6	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15
7	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16
8	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17
9	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18
A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19
B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A
C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B
D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C
E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D
F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E

MULTIPLICATION TABLE

1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2	04	06	08	0A	0C	0E	10	12	14	16	18	1A	1C	1E
3	06	09	0C	0F	12	15	18	1B	1E	21	24	27	2A	2D
4	08	0C	10	14	18	1C	20	24	28	2C	30	34	38	3C
5	0A	0F	14	19	1E	23	28	2D	32	37	3C	41	46	4B
6	0C	12	18	1E	24	2A	30	36	3C	42	48	4E	54	5A
7	0E	15	1C	23	2A	31	38	3F	46	4D	54	5B	62	69
8	10	18	20	28	30	38	40	48	50	58	60	68	70	78
9	12	1B	24	2D	36	3F	48	51	5A	63	6C	75	7E	87
A	14	1E	28	32	3C	46	50	5A	64	6E	78	82	8C	96
B	16	21	2C	37	42	4D	58	63	6E	79	84	8F	9A	A5
C	18	24	30	3C	48	54	60	6C	78	84	90	9C	A8	B4
D	1A	27	34	41	4E	5B	68	75	82	8F	9C	A9	B6	C3
E	1C	2A	38	46	54	62	70	7E	8C	9A	A8	B6	C4	D2
F	1E	2B	3C	4B	5A	69	78	87	96	A5	B4	C3	D2	E1

HEXADECIMAL-DECIMAL INTEGER CONVERSION TABLE

The table below provides for direct conversions between hexadecimal integers in the range 0–FFF and decimal integers in the range 0–4095. For conversion of larger integers, the table values may be added to the following figures:

Hexadecimal	Decimal	Hexadecimal	Decimal
01 000	4 096	20 000	131 072
02 000	8 192	30 000	196 608
03 000	12 288	40 000	262 144
04 000	16 384	50 000	327 680
05 000	20 480	60.000	393 216
06 000	24 576	70 000	458 752
07 000	28 672	80 000	524 288
08 000	32 768	90 000	589 824
09 000	36 864	A0 000	655 360
0A 000	40 960	B0 000	720 896
0B 000	45 056	C0 000	786 432
0C 000	49 152	D0 000	851 968
0D 000	53 248	E0 000	917 504
0E 000	57 344	F0 000	983 040
0F 000	61 440	100 000	1 048 576
10 000	65 536	200 000	2 097 152
11 000	69 632	300 000	3 145 728
12 000	73 728	400 000	4 194 304
13 000	77 824	500 000	5 242 880
14 000	81 920	600 000	6 291 456
15 000	86 016	700 000	7 340 032
16 000	90 112	800 000	8 388 608
17 000	94 208	900 000	9 437 184
18 000	98 304	A00 000	10 485 760
19 000	102 400	B00 000	11 534 336
1A 000	106 496	C00 000	12 582 912
1B 000	110 592	D00 000	13 631 488
1C 000	114 688	E00 000	14 680 064
1D 000	118 784	F00 000	15 728 640
1E 000	122 880	1 000 000	16 777 216
1F 000	126 976	2 000 000	33 554 432

Hexadecimal fractions may be converted to decimal fractions as follows:

1. Express the hexadecimal fraction as an integer times 16^{-n}, where n is the number of significant hexadecimal places to the right of the hexadecimal point.

   $$0.\,CA9BF3_{16} = CA9\,BF3_{16} \times 16^{-6}$$

2. Find the decimal equivalent of the hexadecimal integer

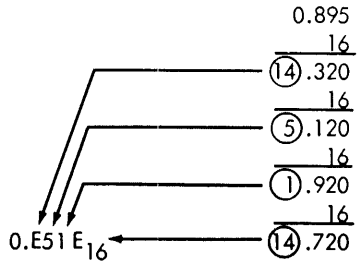
   $$CA9\,BF3_{16} = 13\,278\,195_{10}$$

3. Multiply the decimal equivalent by 16^{-n}

   $$\begin{array}{r} 13\,278\,195 \\ \times\ 596\,046\,448 \times 10^{-16} \\ \hline 0.791\,442\,096_{10} \end{array}$$

Decimal fractions may be converted to hexadecimal fractions by successively multiplying the decimal fraction by 16_{10}. After each multiplication, the integer portion is removed to form a hexadecimal fraction by building to the right of the hexadecimal point. However, since decimal arithmetic is used in this conversion, the integer portion of each product must be converted to hexadecimal numbers.

Example: Convert 0.895_{10} to its hexadecimal equivalent



	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
000	0000	0001	0002	0003	0004	0005	0006	0007	0008	0009	0010	0011	0012	0013	0014	0015
010	0016	0017	0018	0019	0020	0021	0022	0023	0024	0025	0026	0027	0028	0029	0030	0031
020	0032	0033	0034	0035	0036	0037	0038	0039	0040	0041	0042	0043	0044	0045	0046	0047
030	0048	0049	0050	0051	0052	0053	0054	0055	0056	0057	0058	0059	0060	0061	0062	0063
040	0064	0065	0066	0067	0068	0069	0070	0071	0072	0073	0074	0075	0076	0077	0078	0079
050	0080	0081	0082	0083	0084	0085	0086	0087	0088	0089	0090	0091	0092	0093	0094	0095
060	0096	0097	0098	0099	0100	0101	0102	0103	0104	0105	0106	0107	0108	0109	0110	0111
070	0112	0113	0114	0115	0116	0117	0118	0119	0120	0121	0122	0123	0124	0125	0126	0127
080	0128	0129	0130	0131	0132	0133	0134	0135	0136	0137	0138	0139	0140	0141	0142	0143
090	0144	0145	0146	0147	0148	0149	0150	0151	0152	0153	0154	0155	0156	0157	0158	0159
0A0	0160	0161	0162	0163	0164	0165	0166	0167	0168	0169	0170	0171	0172	0173	0174	0175
0B0	0176	0177	0178	0179	0180	0181	0182	0183	0184	0185	0186	0187	0188	0189	0190	0191
0C0	0192	0193	0194	0195	0196	0197	0198	0199	0200	0201	0202	0203	0204	0205	0206	0207
0D0	0208	0209	0210	0211	0212	0213	0214	0215	0216	0217	0218	0219	0220	0221	0222	0223
0E0	0224	0225	0226	0227	0228	0229	0230	0231	0232	0233	0234	0235	0236	0237	0238	0239
0F0	0240	0241	0242	0243	0244	0245	0246	0247	0248	0249	0250	0251	0252	0253	0254	0255

HEXADECIMAL - DECIMAL INTEGER CONVERSION TABLE (cont.)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
100	0256	0257	0258	0259	0260	0261	0262	0263	0264	0265	0266	0267	0268	0269	0270	0271
110	0272	0273	0274	0275	0276	0277	0278	0279	0280	0281	0282	0283	0284	0285	0286	0287
120	0288	0289	0290	0291	0292	0293	0294	0295	0296	0297	0298	0299	0300	0301	0302	0303
130	0304	0305	0306	0307	0308	0309	0310	0311	0312	0313	0314	0315	0316	0317	0318	0319
140	0320	0321	0322	0323	0324	0325	0326	0327	0328	0329	0330	0331	0332	0333	0334	0335
150	0336	0337	0338	0339	0340	0341	0342	0343	0344	0345	0346	0347	0348	0349	0350	0351
160	0352	0353	0354	0355	0356	0357	0358	0359	0360	0361	0362	0363	0364	0365	0366	0367
170	0368	0369	0370	0371	0372	0373	0374	0375	0376	0377	0378	0379	0380	0381	0382	0383
180	0384	0385	0386	0387	0388	0389	0390	0391	0392	0393	0394	0395	0396	0397	0398	0399
190	0400	0401	0402	0403	0404	0405	0406	0407	0408	0409	0410	0411	0412	0413	0414	0415
1A0	0416	0417	0418	0419	0420	0421	0422	0423	0424	0425	0426	0427	0428	0429	0430	0431
1B0	0432	0433	0434	0435	0436	0437	0438	0439	0440	0441	0442	0443	0444	0445	0446	0447
1C0	0448	0449	0450	0451	0452	0453	0454	0455	0456	0457	0458	0459	0460	0461	0462	0463
1D0	0464	0465	0466	0467	0468	0469	0470	0471	0472	0473	0474	0475	0476	0477	0478	0479
1E0	0480	0481	0482	0483	0484	0485	0486	0487	0488	0489	0490	0491	0492	0493	0494	0495
1F0	0496	0497	0498	0499	0500	0501	0502	0503	0504	0505	0506	0507	0508	0509	0510	0511
200	0512	0513	0514	0515	0516	0517	0518	0519	0520	0521	0522	0523	0524	0525	0526	0527
210	0528	0529	0530	0531	0532	0533	0534	0535	0536	0537	0538	0539	0540	0541	0542	0543
220	0544	0545	0546	0547	0548	0549	0550	0551	0552	0553	0554	0555	0556	0557	0558	0559
230	0560	0561	0562	0563	0564	0565	0566	0567	0568	0569	0570	0571	0572	0573	0574	0575
240	0576	0577	0578	0579	0580	0581	0582	0583	0584	0585	0586	0587	0588	0589	0590	0591
250	0592	0593	0594	0595	0596	0597	0598	0599	0600	0601	0602	0603	0604	0605	0606	0607
260	0608	0609	0610	0611	0612	0613	0614	0615	0616	0617	0618	0619	0620	0621	0622	0623
270	0624	0625	0626	0627	0628	0629	0630	0631	0632	0633	0634	0635	0636	0637	0638	0639
280	0640	0641	0642	0643	0644	0645	0646	0647	0648	0649	0650	0651	0652	0653	0654	0655
290	0656	0657	0658	0659	0660	0661	0662	0663	0664	0665	0666	0667	0668	0669	0670	0671
2A0	0672	0673	0674	0675	0676	0677	0678	0679	0680	0681	0682	0683	0684	0685	0686	0687
2B0	0688	0689	0690	0691	0692	0693	0694	0695	0696	0697	0698	0699	0700	0701	0702	0703
2C0	0704	0705	0706	0707	0708	0709	0710	0711	0712	0713	0714	0715	0716	0717	0718	0719
2D0	0720	0721	0722	0723	0724	0725	0726	0727	0728	0729	0730	0731	0732	0733	0734	0735
2E0	0736	0737	0738	0739	0740	0741	0742	0743	0744	0745	0746	0747	0748	0749	0750	0751
2F0	0752	0753	0754	0755	0756	0757	0758	0759	0760	0761	0762	0763	0764	0765	0766	0767
300	0768	0769	0770	0771	0772	0773	0774	0775	0776	0777	0778	0779	0780	0781	0782	0783
310	0784	0785	0786	0787	0788	0789	0790	0791	0792	0793	0794	0795	0796	0797	0798	0799
320	0800	0801	0802	0803	0804	0805	0806	0807	0808	0809	0810	0811	0812	0813	0814	0815
330	0816	0817	0818	0819	0820	0821	0822	0823	0824	0825	0826	0827	0828	0829	0830	0831
340	0832	0833	0834	0835	0836	0837	0838	0839	0840	0841	0842	0843	0844	0845	0846	0847
350	0848	0849	0850	0851	0852	0853	0854	0855	0856	0857	0858	0859	0860	0861	0862	0863
360	0864	0865	0866	0867	0868	0869	0870	0871	0872	0873	0874	0875	0876	0877	0878	0879
370	0880	0881	0882	0883	0884	0885	0886	0887	0888	0889	0890	0891	0892	0893	0894	0895
380	0896	0897	0898	0899	0900	0901	0902	0903	0904	0905	0906	0907	0908	0909	0910	0911
390	0912	0913	0914	0915	0916	0917	0918	0919	0920	0921	0922	0923	0924	0925	0926	0927
3A0	0928	0929	0930	0931	0932	0933	0934	0935	0936	0937	0938	0939	0940	0941	0942	0943
3B0	0944	0945	0946	0947	0948	0949	0950	0951	0952	0953	0954	0955	0956	0957	0958	0959
3C0	0960	0961	0962	0963	0964	0965	0966	0967	0968	0969	0970	0971	0972	0973	0974	0975
3D0	0976	0977	0978	0979	0980	0981	0982	0983	0984	0985	0986	0987	0988	0989	0990	0991
3E0	0992	0993	0994	0995	0996	0997	0998	0999	1000	1001	1002	1003	1004	1005	1006	1007
3F0	1008	1009	1010	1011	1012	1013	1014	1015	1016	1017	1018	1019	1020	1021	1022	1023

HEXADECIMAL - DECIMAL INTEGER CONVERSION TABLE (cont.)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
400	1024	1025	1026	1027	1028	1029	1030	1031	1032	1033	1034	1035	1036	1037	1038	1039
410	1040	1041	1042	1043	1044	1045	1046	1047	1048	1049	1050	1051	1052	1053	1054	1055
420	1056	1057	1058	1059	1060	1061	1062	1063	1064	1065	1066	1067	1068	1069	1070	1071
430	1072	1073	1074	1075	1076	1077	1078	1079	1080	1081	1082	1083	1084	1085	1086	1087
440	1088	1089	1090	1091	1092	1093	1094	1095	1096	1097	1098	1099	1100	1101	1102	1103
450	1104	1105	1106	1107	1108	1109	1110	1111	1112	1113	1114	1115	1116	1117	1118	1119
460	1120	1121	1122	1123	1124	1125	1126	1127	1128	1129	1130	1131	1132	1133	1134	1135
470	1136	1137	1138	1139	1140	1141	1142	1143	1144	1145	1146	1147	1148	1149	1150	1151
480	1152	1153	1154	1155	1156	1157	1158	1159	1160	1161	1162	1163	1164	1165	1166	1167
490	1168	1169	1170	1171	1172	1173	1174	1175	1176	1177	1178	1179	1180	1181	1182	1183
4A0	1184	1185	1186	1187	1188	1189	1190	1191	1192	1193	1194	1195	1196	1197	1198	1199
4B0	1200	1201	1202	1203	1204	1205	1206	1207	1208	1209	1210	1211	1212	1213	1214	1215
4C0	1216	1217	1218	1219	1220	1221	1222	1223	1224	1225	1226	1227	1228	1229	1230	1231
4D0	1232	1233	1234	1235	1236	1237	1238	1239	1240	1241	1242	1243	1244	1245	1246	1247
4E0	1248	1249	1250	1251	1252	1253	1254	1255	1256	1257	1258	1259	1260	1261	1262	1263
4F0	1264	1265	1266	1267	1268	1269	1270	1271	1272	1273	1274	1275	1276	1277	1278	1279
500	1280	1281	1282	1283	1284	1285	1286	1287	1288	1289	1290	1291	1292	1293	1294	1295
510	1296	1297	1298	1299	1300	1301	1302	1303	1304	1305	1306	1307	1308	1309	1310	1311
520	1312	1313	1314	1315	1316	1317	1318	1319	1320	1321	1322	1323	1324	1325	1326	1327
530	1328	1329	1330	1331	1332	1333	1334	1335	1336	1337	1338	1339	1340	1341	1342	1343
540	1344	1345	1346	1347	1348	1349	1350	1351	1352	1353	1354	1355	1356	1357	1358	1359
550	1360	1361	1362	1363	1364	1365	1366	1367	1368	1369	1370	1371	1372	1373	1374	1375
560	1376	1377	1378	1379	1380	1381	1382	1383	1384	1385	1386	1387	1388	1389	1390	1391
570	1392	1393	1394	1395	1396	1397	1398	1399	1400	1401	1402	1403	1404	1405	1406	1407
580	1408	1409	1410	1411	1412	1413	1414	1415	1416	1417	1418	1419	1420	1421	1422	1423
590	1424	1425	1426	1427	1428	1429	1430	1431	1432	1433	1434	1435	1436	1437	1438	1439
5A0	1440	1441	1442	1443	1444	1445	1446	1447	1448	1449	1450	1451	1452	1453	1454	1455
5B0	1456	1457	1458	1459	1460	1461	1462	1463	1464	1465	1466	1467	1468	1469	1470	1471
5C0	1472	1473	1474	1475	1476	1477	1478	1479	1480	1481	1482	1483	1484	1485	1486	1487
5D0	1488	1489	1490	1491	1492	1493	1494	1495	1496	1497	1498	1499	1500	1501	1502	1503
5E0	1504	1505	1506	1507	1508	1509	1510	1511	1512	1513	1514	1515	1516	1517	1518	1519
5F0	1520	1521	1522	1523	1524	1525	1526	1527	1528	1529	1530	1531	1532	1533	1534	1535
600	1536	1537	1538	1539	1540	1541	1542	1543	1544	1545	1546	1547	1548	1549	1550	1551
610	1552	1553	1554	1555	1556	1557	1558	1559	1560	1561	1562	1563	1564	1565	1566	1567
620	1568	1569	1570	1571	1572	1573	1574	1575	1576	1577	1578	1579	1580	1581	1582	1583
630	1584	1585	1586	1587	1588	1589	1590	1591	1592	1593	1594	1595	1596	1597	1598	1599
640	1600	1601	1602	1603	1604	1605	1606	1607	1608	1609	1610	1611	1612	1613	1614	1615
650	1616	1617	1618	1619	1620	1621	1622	1623	1624	1625	1626	1627	1628	1629	1630	1631
660	1632	1633	1634	1635	1636	1637	1638	1639	1640	1641	1642	1643	1644	1645	1646	1647
670	1648	1649	1650	1651	1652	1653	1654	1655	1656	1657	1658	1659	1660	1661	1662	1663
680	1664	1665	1666	1667	1668	1669	1670	1671	1672	1673	1674	1675	1676	1677	1678	1679
690	1680	1681	1682	1683	1684	1685	1686	1687	1688	1689	1690	1691	1692	1693	1694	1695
6A0	1696	1697	1698	1699	1700	1701	1702	1703	1704	1705	1706	1707	1708	1709	1710	1711
6B0	1712	1713	1714	1715	1716	1717	1718	1719	1720	1721	1722	1723	1724	1725	1726	1727
6C0	1728	1729	1730	1731	1732	1733	1734	1735	1736	1737	1738	1739	1740	1741	1742	1743
6D0	1744	1745	1746	1747	1748	1749	1750	1751	1752	1753	1754	1755	1756	1757	1758	1759
6E0	1760	1761	1762	1763	1764	1765	1766	1767	1768	1769	1770	1771	1772	1773	1774	1775
6F0	1776	1777	1778	1779	1780	1781	1782	1783	1784	1785	1786	1787	1788	1789	1790	1791

HEXADECIMAL - DECIMAL INTEGER CONVERSION TABLE (cont.)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
700	1792	1793	1794	1795	1796	1797	1798	1799	1800	1801	1802	1803	1804	1805	1806	1807
710	1808	1809	1810	1811	1812	1813	1814	1815	1816	1817	1818	1819	1820	1821	1822	1823
720	1824	1825	1826	1827	1828	1829	1830	1831	1832	1833	1834	1835	1836	1837	1838	1839
730	1840	1841	1842	1843	1844	1845	1846	1847	1848	1849	1850	1851	1852	1853	1854	1855
740	1856	1857	1858	1859	1860	1861	1862	1863	1864	1865	1866	1867	1868	1869	1870	1871
750	1872	1873	1874	1875	1876	1877	1878	1879	1880	1881	1882	1883	1884	1885	1886	1887
760	1888	1889	1890	1891	1892	1893	1894	1895	1896	1897	1898	1899	1900	1901	1902	1903
770	1904	1905	1906	1907	1908	1909	1910	1911	1912	1913	1914	1915	1916	1917	1918	1919
780	1920	1921	1922	1923	1924	1925	1926	1927	1928	1929	1930	1931	1932	1933	1934	1935
790	1936	1937	1938	1939	1940	1941	1942	1943	1944	1945	1946	1947	1948	1949	1950	1951
7A0	1952	1953	1954	1955	1956	1957	1958	1959	1960	1961	1962	1963	1964	1965	1966	1967
7B0	1968	1969	1970	1971	1972	1973	1974	1975	1976	1977	1978	1979	1980	1981	1982	1983
7C0	1984	1985	1986	1987	1988	1989	1990	1991	1992	1993	1994	1995	1996	1997	1998	1999
7D0	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015
7E0	2016	2017	2018	2019	2020	2021	2022	2023	2024	2025	2026	2027	2028	2029	2030	2031
7F0	2032	2033	2034	2035	2036	2037	2038	2039	2040	2041	2042	2043	2044	2045	2046	2047
800	2048	2049	2050	2051	2052	2053	2054	2055	2056	2057	2058	2059	2060	2061	2062	2063
810	2064	2065	2066	2067	2068	2069	2070	2071	2072	2073	2074	2075	2076	2077	2078	2079
820	2080	2081	2082	2083	2084	2085	2086	2087	2088	2089	2090	2091	2092	2093	2094	2095
830	2096	2097	2098	2099	2100	2101	2102	2103	2104	2105	2106	2107	2108	2109	2110	2111
840	2112	2113	2114	2115	2116	2117	2118	2119	2120	2121	2122	2123	2124	2125	2126	2127
850	2128	2129	2130	2131	2132	2133	2134	2135	2136	2137	2138	2139	2140	2141	2142	2143
860	2144	2145	2146	2147	2148	2149	2150	2151	2152	2153	2154	2155	2156	2157	2158	2159
870	2160	2161	2162	2163	2164	2165	2166	2167	2168	2169	2170	2171	2172	2173	2174	2175
880	2176	2177	2178	2179	2180	2181	2182	2183	2184	2185	2186	2187	2188	2189	2190	2191
890	2192	2193	2194	2195	2196	2197	2198	2199	2200	2201	2202	2203	2204	2205	2206	2207
8A0	2208	2209	2210	2211	2212	2213	2214	2215	2216	2217	2218	2219	2220	2221	2222	2223
8B0	2224	2225	2226	2227	2228	2229	2230	2231	2232	2233	2234	2235	2236	2237	2238	2239
8C0	2240	2241	2242	2243	2244	2245	2246	2247	2248	2249	2250	2251	2252	2253	2254	2255
8D0	2256	2257	2258	2259	2260	2261	2262	2263	2264	2265	2266	2267	2268	2269	2270	2271
8E0	2272	2273	2274	2275	2276	2277	2278	2279	2280	2281	2282	2283	2284	2285	2286	2287
8F0	2288	2289	2290	2291	2292	2293	2294	2295	2296	2297	2298	2299	2300	2301	2302	2303
900	2304	2305	2306	2307	2308	2309	2310	2311	2312	2313	2314	2315	2316	2317	2318	2319
910	2320	2321	2322	2323	2324	2325	2326	2327	2328	2329	2330	2331	2332	2333	2334	2335
920	2336	2337	2338	2339	2340	2341	2342	2343	2344	2345	2346	2347	2348	2349	2350	2351
930	2352	2353	2354	2355	2356	2357	2358	2359	2360	2361	2362	2363	2364	2365	2366	2367
940	2368	2369	2370	2371	2372	2373	2374	2375	2376	2377	2378	2379	2380	2381	2382	2383
950	2384	2385	2386	2387	2388	2389	2390	2391	2392	2393	2394	2395	2396	2397	2398	2399
960	2400	2401	2402	2403	2404	2405	2406	2407	2408	2409	2410	2411	2412	2413	2414	2415
970	2416	2417	2418	2419	2420	2421	2422	2423	2424	2425	2426	2427	2428	2429	2430	2431
980	2432	2433	2434	2435	2436	2437	2438	2439	2440	2441	2442	2443	2444	2445	2446	2447
990	2448	2449	2450	2451	2452	2453	2454	2455	2456	2457	2458	2459	2460	2461	2462	2463
9A0	2464	2465	2466	2467	2468	2469	2470	2471	2472	2473	2474	2475	2476	2477	2478	2479
9B0	2480	2481	2482	2483	2484	2485	2486	2487	2488	2489	2490	2491	2492	2493	2494	2495
9C0	2496	2497	2498	2499	2500	2501	2502	2503	2504	2505	2506	2507	2508	2509	2510	2511
9D0	2512	2513	2514	2515	2516	2517	2518	2519	2520	2521	2522	2523	2524	2525	2526	2527
9E0	2528	2529	2530	2531	2532	2533	2534	2535	2536	2537	2538	2539	2540	2541	2542	2543
9F0	2544	2545	2546	2547	2548	2549	2550	2551	2552	2553	2554	2555	2556	2557	2558	2559

HEXADECIMAL - DECIMAL INTEGER CONVERSION TABLE (cont.)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
A00	2560	2561	2562	2563	2564	2565	2566	2567	2568	2569	2570	2571	2572	2573	2574	2575
A10	2576	2577	2578	2579	2580	2581	2582	2583	2584	2585	2586	2587	2588	2589	2590	2591
A20	2592	2593	2594	2595	2596	2597	2598	2599	2600	2601	2602	2603	2604	2605	2606	2607
A30	2608	2609	2610	2611	2612	2613	2614	2615	2616	2617	2618	2619	2620	2621	2622	2623
A40	2624	2625	2626	2627	2628	2629	2630	2631	2632	2633	2634	2635	2636	2637	2638	2639
A50	2640	2641	2642	2643	2644	2645	2646	2647	2648	2649	2650	2651	2652	2653	2654	2655
A60	2656	2657	2658	2659	2660	2661	2662	2663	2664	2665	2666	2667	2668	2669	2670	2671
A70	2672	2673	2674	2675	2676	2677	2678	2679	2680	2681	2682	2683	2684	2685	2686	2687
A80	2688	2689	2690	2691	2692	2693	2694	2695	2696	2697	2698	2699	2700	2701	2702	2703
A90	2704	2705	2706	2707	2708	2709	2710	2711	2712	2713	2714	2715	2716	2717	2718	2719
AA0	2720	2721	2722	2723	2724	2725	2726	2727	2728	2729	2730	2731	2732	2733	2734	2735
AB0	2736	2737	2738	2739	2740	2741	2742	2743	2744	2745	2746	2747	2748	2749	2750	2751
AC0	2752	2753	2754	2755	2756	2757	2758	2759	2760	2761	2762	2763	2764	2765	2766	2767
AD0	2768	2769	2770	2771	2772	2773	2774	2775	2776	2777	2778	2779	2780	2781	2782	2783
AE0	2784	2785	2786	2787	2788	2789	2790	2791	2792	2793	2794	2795	2796	2797	2798	2799
AF0	2800	2801	2802	2803	2804	2805	2806	2807	2808	2809	2810	2811	2812	2813	2814	2815
B00	2816	2817	2818	2819	2820	2821	2822	2823	2824	2825	2826	2827	2828	2829	2830	2831
B10	2832	2833	2834	2835	2836	2837	2838	2839	2840	2841	2842	2843	2844	2845	2846	2847
B20	2848	2849	2850	2851	2852	2853	2854	2855	2856	2857	2858	2859	2860	2861	2862	2863
B30	2864	2865	2866	2867	2868	2869	2870	2871	2872	2873	2874	2875	2876	2877	2878	2879
B40	2880	2881	2882	2883	2884	2885	2886	2887	2888	2889	2890	2891	2892	2893	2894	2895
B50	2896	2897	2898	2899	2900	2901	2902	2903	2904	2905	2906	2907	2908	2909	2910	2911
B60	2912	2913	2914	2915	2916	2917	2918	2919	2920	2921	2922	2923	2924	2925	2926	2927
B70	2928	2929	2930	2931	2932	2933	2934	2935	2936	2937	2938	2939	2940	2941	2942	2943
B80	2944	2945	2946	2947	2948	2949	2950	2951	2952	2953	2954	2955	2956	2957	2958	2959
B90	2960	2961	2962	2963	2964	2965	2966	2967	2968	2969	2970	2971	2972	2973	2974	2975
BA0	2976	2977	2978	2979	2980	2981	2982	2983	2984	2985	2986	2987	2988	2989	2990	2991
BB0	2992	2993	2994	2995	2996	2997	2998	2999	3000	3001	3002	3003	3004	3005	3006	3007
BC0	3008	3009	3010	3011	3012	3013	3014	3015	3016	3017	3018	3019	3020	3021	3022	3023
BD0	3024	3025	3026	3027	3028	3029	3030	3031	3032	3033	3034	3035	3036	3037	3038	3039
BE0	3040	3041	3042	3043	3044	3045	3046	3047	3048	3049	3050	3051	3052	3053	3054	3055
BF0	3056	3057	3058	3059	3060	3061	3062	3063	3064	3065	3066	3067	3068	3069	3070	3071
C00	3072	3073	3074	3075	3076	3077	3078	3079	3080	3081	3082	3083	3084	3085	3086	3087
C10	3088	3089	3090	3091	3092	3093	3094	3095	3096	3097	3098	3099	3100	3101	3102	3103
C20	3104	3105	3106	3107	3108	3109	3110	3111	3112	3113	3114	3115	3116	3117	3118	3119
C30	3120	3121	3122	3123	3124	3125	3126	3127	3128	3129	3130	3131	3132	3133	3134	3135
C40	3136	3137	3138	3139	3140	3141	3142	3143	3144	3145	3146	3147	3148	3149	3150	3151
C50	3152	3153	3154	3155	3156	3157	3158	3159	3160	3161	3162	3163	3164	3165	3166	3167
C60	3168	3169	3170	3171	3172	3173	3174	3175	3176	3177	3178	3179	3180	3181	3182	3183
C70	3184	3185	3186	3187	3188	3189	3190	3191	3192	3193	3194	3195	3196	3197	3198	3199
C80	3200	3201	3202	3203	3204	3205	3206	3207	3208	3209	3210	3211	3212	3213	3214	3215
C90	3216	3217	3218	3219	3220	3221	3222	3223	3224	3225	3226	3227	3228	3229	3230	3231
CA0	3232	3233	3234	3235	3236	3237	3238	3239	3240	3241	3242	3243	3244	3245	3246	3247
CB0	3248	3249	3250	3251	3252	3253	3254	3255	3256	3257	3258	3259	3260	3261	3262	3263
CC0	3264	3265	3266	3267	3268	3269	3270	3271	3272	3273	3274	3275	3276	3277	3278	3279
CD0	3280	3281	3282	3283	3284	3285	3286	3287	3288	3289	3290	3291	3292	3293	3294	3295
CE0	3296	3297	3298	3299	3300	3301	3302	3303	3304	3305	3306	3307	3308	3309	3310	3311
CF0	3312	3313	3314	3315	3316	3317	3318	3319	3320	3321	3322	3323	3324	3325	3326	3327

HEXADECIMAL - DECIMAL INTEGER CONVERSION TABLE (cont.)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
D00	3328	3329	3330	3331	3332	3333	3334	3335	3336	3337	3338	3339	3340	3341	3342	3343
D10	3344	3345	3346	3347	3348	3349	3350	3351	3352	3353	3354	3355	3356	3357	3358	3359
D20	3360	3361	3362	3363	3364	3365	3366	3367	3368	3369	3370	3371	3372	3373	3374	3375
D30	3376	3377	3378	3379	3380	3381	3382	3383	3384	3385	3386	3387	3388	3389	3390	3391
D40	3392	3393	3394	3395	3396	3397	3398	3399	3400	3401	3402	3403	3404	3405	3406	3407
D50	3408	3409	3410	3411	3412	3413	3414	3415	3416	3417	3418	3419	3420	3421	3422	3423
D60	3424	3425	3426	3427	3428	3429	3430	3431	3432	3433	3434	3435	3436	3437	3438	3439
D70	3440	3441	3442	3443	3444	3445	3446	3447	3448	3449	3450	3451	3452	3453	3454	3455
D80	3456	3457	3458	3459	3460	3461	3462	3463	3464	3465	3466	3467	3468	3469	3470	3471
D90	3472	3473	3474	3475	3476	3477	3478	3479	3480	3481	3482	3483	3484	3485	3486	3487
DA0	3488	3489	3490	3491	3492	3493	3494	3495	3496	3497	3498	3499	3500	3501	3502	3503
DB0	3504	3505	3506	3507	3508	3509	3510	3511	3512	3513	3514	3515	3516	3517	3518	3519
DC0	3520	3521	3522	3523	3524	3525	3526	3527	3528	3529	3530	3531	3532	3533	3534	3535
DD0	3536	3537	3538	3539	3540	3541	3542	3543	3544	3545	3546	3547	3548	3549	3550	3551
DE0	3552	3553	3554	3555	3556	3557	3558	3559	3560	3561	3562	3563	3564	3565	3566	3567
DF0	3568	3569	3570	3571	3572	3573	3574	3575	3576	3577	3578	3579	3580	3581	3582	3583
E00	3584	3585	3586	3587	3588	3589	3590	3591	3592	3593	3594	3595	3596	3597	3598	3599
E10	3600	3601	3602	3603	3604	3605	3606	3607	3608	3609	3610	3611	3612	3613	3614	3615
E20	3616	3617	3618	3619	3620	3621	3622	3623	3624	3625	3626	3627	3628	3629	3630	3631
E30	3632	3633	3634	3635	3636	3637	3638	3639	3640	3641	3642	3643	3644	3645	3646	3647
E40	3648	3649	3650	3651	3652	3653	3654	3655	3656	3657	3658	3659	3660	3661	3662	3663
E50	3664	3665	3666	3667	3668	3669	3670	3671	3672	3673	3674	3675	3676	3677	3678	3679
E60	3680	3681	3682	3683	3684	3685	3686	3687	3688	3689	3690	3691	3692	3693	3694	3695
E70	3696	3697	3698	3699	3700	3701	3702	3703	3704	3705	3706	3707	3708	3709	3710	3711
E80	3712	3713	3714	3715	3716	3717	3718	3719	3720	3721	3722	3723	3724	3725	3726	3727
E90	3728	3729	3730	3731	3732	3733	3734	3735	3736	3737	3738	3739	3740	3741	3742	3743
EA0	3744	3745	3746	3747	3748	3749	3750	3751	3752	3753	3754	3755	3756	3757	3758	3759
EB0	3760	3761	3762	3763	3764	3765	3766	3767	3768	3769	3770	3771	3772	3773	3774	3775
EC0	3776	3777	3778	3779	3780	3781	3782	3783	3784	3785	3786	3787	3788	3789	3790	3791
ED0	3792	3793	3794	3795	3796	3797	3798	3799	3800	3801	3802	3803	3804	3805	3806	3807
EE0	3808	3809	3810	3811	3812	3813	3814	3815	3816	3817	3818	3819	3820	3821	3822	3823
EF0	3824	3825	3826	3827	3828	3829	3830	3831	3832	3833	3834	3835	3836	3837	3838	3839
F00	3840	3841	3842	3843	3844	3845	3846	3847	3848	3849	3850	3851	3852	3853	3854	3855
F10	3856	3857	3858	3859	3860	3861	3862	3863	3864	3865	3866	3867	3868	3869	3870	3871
F20	3872	3873	3874	3875	3876	3877	3878	3879	3880	3881	3882	3883	3884	3885	3886	3887
F30	3888	3889	3890	3891	3892	3893	3894	3895	3896	3897	3898	3899	3900	3901	3902	3903
F40	3904	3905	3906	3907	3908	3909	3910	3911	3912	3913	3914	3915	3916	3917	3918	3919
F50	3920	3921	3922	3923	3924	3925	3926	3927	3928	3929	3930	3931	3932	3933	3934	3935
F60	3936	3937	3938	3939	3940	3941	3942	3943	3944	3945	3946	3947	3948	3949	3950	3951
F70	3952	3953	3954	3955	3956	3957	3958	3959	3960	3961	3962	3963	3964	3965	3966	3967
F80	3968	3969	3970	3971	3972	3973	3974	3975	3976	3977	3978	3979	3980	3981	3982	3983
F90	3984	3985	3986	3987	3988	3989	3990	3991	3992	3993	3994	3995	3996	3997	3998	3999
FA0	4000	4001	4002	4003	4004	4005	4006	4007	4008	4009	4010	4011	4012	4013	4014	4015
FB0	4016	4017	4018	4019	4020	4021	4022	4023	4024	4025	4026	4027	4028	4029	4030	4031
FC0	4032	4033	4034	4035	4036	4037	4038	4039	4040	4041	4042	4043	4044	4045	4046	4047
FD0	4048	4049	4050	4051	4052	4053	4054	4055	4056	4057	4058	4059	4060	4061	4062	4063
FE0	4064	4065	4066	4067	4068	4069	4070	4071	4072	4073	4074	4075	4076	4077	4078	4079
FF0	4080	4081	4082	4083	4084	4085	4086	4087	4088	4089	4090	4091	4092	4093	4094	4095

HEXADECIMAL-DECIMAL FRACTION CONVERSION TABLE

Hexadecimal	Decimal	Hexadecimal	Decimal	Hexadecimal	Decimal	Hexadecimal	Decimal
.00 00 00 00	.00000 00000	.40 00 00 00	.25000 00000	.80 00 00 00	.50000 00000	.C0 00 00 00	.75000 00000
.01 00 00 00	.00390 62500	.41 00 00 00	.25390 62500	.81 00 00 00	.50390 62500	.C1 00 00 00	.75390 62500
.02 00 00 00	.00781 25000	.42 00 00 00	.25781 25000	.82 00 00 00	.50781 25000	.C2 00 00 00	.75781 25000
.03 00 00 00	.01171 87500	.43 00 00 00	.26171 87500	.83 00 00 00	.51171 87500	.C3 00 00 00	.76171 87500
.04 00 00 00	.01562 50000	.44 00 00 00	.26562 50000	.84 00 00 00	.51562 50000	.C4 00 00 00	.76562 50000
.05 00 00 00	.01953 12500	.45 00 00 00	.26953 12500	.85 00 00 00	.51953 12500	.C5 00 00 00	.76953 12500
.06 00 00 00	.02343 75000	.46 00 00 00	.27343 75000	.86 00 00 00	.52343 75000	.C6 00 00 00	.77343 75000
.07 00 00 00	.02734 37500	.47 00 00 00	.27734 37500	.87 00 00 00	.52734 37500	.C7 00 00 00	.77734 37500
.08 00 00 00	.03125 00000	.48 00 00 00	.28125 00000	.88 00 00 00	.53125 00000	.C8 00 00 00	.78125 00000
.09 00 00 00	.03515 62500	.49 00 00 00	.28515 62500	.89 00 00 00	.53515 62500	.C9 00 00 00	.78515 62500
.0A 00 00 00	.03906 25000	.4A 00 00 00	.28906 25000	.8A 00 00 00	.53906 25000	.CA 00 00 00	.78906 25000
.0B 00 00 00	.04296 87500	.4B 00 00 00	.29296 87500	.8B 00 00 00	.54296 87500	.CB 00 00 00	.79296 87500
.0C 00 00 00	.04687 50000	.4C 00 00 00	.29687 50000	.8C 00 00 00	.54687 50000	.CC 00 00 00	.79687 50000
.0D 00 00 00	.05078 12500	.4D 00 00 00	.30078 12500	.8D 00 00 00	.55078 12500	.CD 00 00 00	.80078 12500
.0E 00 00 00	.05468 75000	.4E 00 00 00	.30468 75000	.8E 00 00 00	.55468 75000	.CE 00 00 00	.80468 75000
.0F 00 00 00	.05859 37500	.4F 00 00 00	.30859 37500	.8F 00 00 00	.55859 37500	.CF 00 00 00	.80859 37500
.10 00 00 00	.06250 00000	.50 00 00 00	.31250 00000	.90 00 00 00	.56250 00000	.D0 00 00 00	.81250 00000
.11 00 00 00	.06640 62500	.51 00 00 00	.31640 62500	.91 00 00 00	.56640 62500	.D1 00 00 00	.81640 62500
.12 00 00 00	.07031 25000	.52 00 00 00	.32031 25000	.92 00 00 00	.57031 25000	.D2 00 00 00	.82031 25000
.13 00 00 00	.07421 87500	.53 00 00 00	.32421 87500	.93 00 00 00	.57421 87500	.D3 00 00 00	.82421 87500
.14 00 00 00	.07812 50000	.54 00 00 00	.32812 50000	.94 00 00 00	.57812 50000	.D4 00 00 00	.82812 50000
.15 00 00 00	.08203 12500	.55 00 00 00	.33203 12500	.95 00 00 00	.58203 12500	.D5 00 00 00	.83203 12500
.16 00 00 00	.08593 75000	.56 00 00 00	.33593 75000	.96 00 00 00	.58593 75000	.D6 00 00 00	.83593 75000
.17 00 00 00	.08984 37500	.57 00 00 00	.33984 37500	.97 00 00 00	.58984 37500	.D7 00 00 00	.83984 37500
.18 00 00 00	.09375 00000	.58 00 00 00	.34375 00000	.98 00 00 00	.59375 00000	.D8 00 00 00	.84375 00000
.19 00 00 00	.09765 62500	.59 00 00 00	.34765 62500	.99 00 00 00	.59765 62500	.D9 00 00 00	.84765 62500
.1A 00 00 00	.10156 25000	.5A 00 00 00	.35156 25000	.9A 00 00 00	.60156 25000	.DA 00 00 00	.85156 25000
.1B 00 00 00	.10546 87500	.5B 00 00 00	.35546 87500	.9B 00 00 00	.60546 87500	.DB 00 00 00	.85546 87500
.1C 00 00 00	.10937 50000	.5C 00 00 00	.35937 50000	.9C 00 00 00	.60937 50000	.DC 00 00 00	.85937 50000
.1D 00 00 00	.11328 12500	.5D 00 00 00	.36328 12500	.9D 00 00 00	.61328 12500	.DD 00 00 00	.86328 12500
.1E 00 00 00	.11718 75000	.5E 00 00 00	.36718 75000	.9E 00 00 00	.61718 75000	.DE 00 00 00	.86718 75000
.1F 00 00 00	.12109 37500	.5F 00 00 00	.37109 37500	.9F 00 00 00	.62109 37500	.DF 00 00 00	.87109 37500
.20 00 00 00	.12500 00000	.60 00 00 00	.37500 00000	.A0 00 00 00	.62500 00000	.E0 00 00 00	.87500 00000
.21 00 00 00	.12890 62500	.61 00 00 00	.37890 62500	.A1 00 00 00	.62890 62500	.E1 00 00 00	.87890 62500
.22 00 00 00	.13281 25000	.62 00 00 00	.38281 25000	.A2 00 00 00	.63281 25000	.E2 00 00 00	.88281 25000
.23 00 00 00	.13671 87500	.63 00 00 00	.38671 87500	.A3 00 00 00	.63671 87500	.E3 00 00 00	.88671 87500
.24 00 00 00	.14062 50000	.64 00 00 00	.39062 50000	.A4 00 00 00	.64062 50000	.E4 00 00 00	.89062 50000
.25 00 00 00	.14453 12500	.65 00 00 00	.39453 12500	.A5 00 00 00	.64453 12500	.E5 00 00 00	.89453 12500
.26 00 00 00	.14843 75000	.66 00 00 00	.39843 75000	.A6 00 00 00	.64843 75000	.E6 00 00 00	.89843 75000
.27 00 00 00	.15234 37500	.67 00 00 00	.40234 37500	.A7 00 00 00	.65234 37500	.E7 00 00 00	.90234 37500
.28 00 00 00	.15625 00000	.68 00 00 00	.40625 00000	.A8 00 00 00	.65625 00000	.E8 00 00 00	.90625 00000
.29 00 00 00	.16015 62500	.69 00 00 00	.41015 62500	.A9 00 00 00	.66015 62500	.E9 00 00 00	.91015 62500
.2A 00 00 00	.16406 25000	.6A 00 00 00	.41406 25000	.AA 00 00 00	.66406 25000	.EA 00 00 00	.91406 25000
.2B 00 00 00	.16796 87500	.6B 00 00 00	.41796 87500	.AB 00 00 00	.66796 87500	.EB 00 00 00	.91796 87500
.2C 00 00 00	.17187 50000	.6C 00 00 00	.42187 50000	.AC 00 00 00	.67187 50000	.EC 00 00 00	.92187 50000
.2D 00 00 00	.17578 12500	.6D 00 00 00	.42578 12500	.AD 00 00 00	.67578 12500	.ED 00 00 00	.92578 12500
.2E 00 00 00	.17968 75000	.6E 00 00 00	.42968 75000	.AE 00 00 00	.67968 75000	.EE 00 00 00	.92968 75000
.2F 00 00 00	.18359 37500	.6F 00 00 00	.43359 37500	.AF 00 00 00	.68359 37500	.EF 00 00 00	.93359 37500
.30 00 00 00	.18750 00000	.70 00 00 00	.43750 00000	.B0 00 00 00	.68750 00000	.F0 00 00 00	.93750 00000
.31 00 00 00	.19140 62500	.71 00 00 00	.44140 62500	.B1 00 00 00	.69140 62500	.F1 00 00 00	.94140 62500
.32 00 00 00	.19531 25000	.72 00 00 00	.44531 25000	.B2 00 00 00	.69531 25000	.F2 00 00 00	.94531 25000
.33 00 00 00	.19921 87500	.73 00 00 00	.44921 87500	.B3 00 00 00	.69921 87500	.F3 00 00 00	.94921 87500
.34 00 00 00	.20312 50000	.74 00 00 00	.45312 50000	.B4 00 00 00	.70312 50000	.F4 00 00 00	.95312 50000
.35 00 00 00	.20703 12500	.75 00 00 00	.45703 12500	.B5 00 00 00	.70703 12500	.F5 00 00 00	.95703 12500
.36 00 00 00	.21093 75000	.76 00 00 00	.46093 75000	.B6 00 00 00	.71093 75000	.F6 00 00 00	.96093 75000
.37 00 00 00	.21484 37500	.77 00 00 00	.46484 37500	.B7 00 00 00	.71484 37500	.F7 00 00 00	.96484 37500
.38 00 00 00	.21875 00000	.78 00 00 00	.46875 00000	.B8 00 00 00	.71875 00000	.F8 00 00 00	.96875 00000
.39 00 00 00	.22265 62500	.79 00 00 00	.47265 62500	.B9 00 00 00	.72265 62500	.F9 00 00 00	.97265 62500
.3A 00 00 00	.22656 25000	.7A 00 00 00	.47656 25000	.BA 00 00 00	.72656 25000	.FA 00 00 00	.97656 25000
.3B 00 00 00	.23046 87500	.7B 00 00 00	.48046 87500	.BB 00 00 00	.73046 87500	.FB 00 00 00	.98046 87500
.3C 00 00 00	.23437 50000	.7C 00 00 00	.48437 50000	.BC 00 00 00	.73437 50000	.FC 00 00 00	.98437 50000
.3D 00 00 00	.23828 12500	.7D 00 00 00	.48828 12500	.BD 00 00 00	.73828 12500	.FD 00 00 00	.98828 12500
.3E 00 00 00	.24218 75000	.7E 00 00 00	.49218 75000	.BE 00 00 00	.74218 75000	.FE 00 00 00	.99218 75000
.3F 00 00 00	.24609 37500	.7F 00 00 00	.49609 37500	.BF 00 00 00	.74609 37500	.FF 00 00 00	.99609 37500

HEXADECIMAL - DECIMAL FRACTION CONVERSION TABLE (cont.)

Hexadecimal	Decimal	Hexadecimal	Decimal	Hexadecimal	Decimal	Hexadecimal	Decimal
.00 00 00 00	.00000 00000	.00 40 00 00	.00097 65625	.00 80 00 00	.00195 31250	.00 C0 00 00	.00292 96875
.00 01 00 00	.00001 52587	.00 41 00 00	.00099 18212	.00 81 00 00	.00196 83837	.00 C1 00 00	.00294 49462
.00 02 00 00	.00003 05175	.00 42 00 00	.00100 70800	.00 82 00 00	.00198 36425	.00 C2 00 00	.00296 02050
.00 03 00 00	.00004 57763	.00 43 00 00	.00102 23388	.00 83 00 00	.00199 89013	.00 C3 00 00	.00297 54638
.00 04 00 00	.00006 10351	.00 44 00 00	.00103 75976	.00 84 00 00	.00201 41601	.00 C4 00 00	.00299 07226
.00 05 00 00	.00007 62939	.00 45 00 00	.00105 28564	.00 85 00 00	.00202 94189	.00 C5 00 00	.00300 59814
.00 06 00 00	.00009 15527	.00 46 00 00	.00106 81152	.00 86 00 00	.00204 46777	.00 C6 00 00	.00302 12402
.00 07 00 00	.00010 68115	.00 47 00 00	.00108 33740	.00 87 00 00	.00205 99365	.00 C7 00 00	.00303 64990
.00 08 00 00	.00012 20703	.00 48 00 00	.00109 86328	.00 88 00 00	.00207 51953	.00 C8 00 00	.00305 17578
.00 09 00 00	.00013 73291	.00 49 00 00	.00111 38916	.00 89 00 00	.00209 04541	.00 C9 00 00	.00306 70166
.00 0A 00 00	.00015 25878	.00 4A 00 00	.00112 91503	.00 8A 00 00	.00210 57128	.00 CA 00 00	.00308 22753
.00 0B 00 00	.00016 78466	.00 4B 00 00	.00114 44091	.00 8B 00 00	.00212 09716	.00 CB 00 00	.00309 75341
.00 0C 00 00	.00018 31054	.00 4C 00 00	.00115 96679	.00 8C 00 00	.00213 62304	.00 CC 00 00	.00311 27929
.00 0D 00 00	.00019 83642	.00 4D 00 00	.00117 49267	.00 8D 00 00	.00215 14892	.00 CD 00 00	.00312 80517
.00 0E 00 00	.00021 36230	.00 4E 00 00	.00119 01855	.00 8E 00 00	.00216 67480	.00 CE 00 00	.00314 33105
.00 0F 00 00	.00022 88818	.00 4F 00 00	.00120 54443	.00 8F 00 00	.00218 20068	.00 CF 00 00	.00315 85693
.00 10 00 00	.00024 41406	.00 50 00 00	.00122 07031	.00 90 00 00	.00219 72656	.00 D0 00 00	.00317 38281
.00 11 00 00	.00025 93994	.00 51 00 00	.00123 59619	.00 91 00 00	.00221 25244	.00 D1 00 00	.00318 90869
.00 12 00 00	.00027 46582	.00 52 00 00	.00125 12207	.00 92 00 00	.00222 77832	.00 D2 00 00	.00320 43457
.00 13 00 00	.00028 99169	.00 53 00 00	.00126 64794	.00 93 00 00	.00224 30419	.00 D3 00 00	.00321 96044
.00 14 00 00	.00030 51757	.00 54 00 00	.00128 17382	.00 94 00 00	.00225 83007	.00 D4 00 00	.00323 48632
.00 15 00 00	.00032 04345	.00 55 00 00	.00129 69970	.00 95 00 00	.00227 35595	.00 D5 00 00	.00325 01220
.00 16 00 00	.00033 56933	.00 56 00 00	.00131 22558	.00 96 00 00	.00228 88183	.00 D6 00 00	.00326 53808
.00 17 00 00	.00035 09521	.00 57 00 00	.00132 75146	.00 97 00 00	.00230 40771	.00 D7 00 00	.00328 06396
.00 18 00 00	.00036 62109	.00 58 00 00	.00134 27734	.00 98 00 00	.00231 93359	.00 D8 00 00	.00329 58984
.00 19 00 00	.00038 14697	.00 59 00 00	.00135 80322	.00 99 00 00	.00233 45947	.00 D9 00 00	.00331 11572
.00 1A 00 00	.00039 67285	.00 5A 00 00	.00137 32910	.00 9A 00 00	.00234 98535	.00 DA 00 00	.00332 64160
.00 1B 00 00	.00041 19873	.00 5B 00 00	.00138 85498	.00 9B 00 00	.00236 51123	.00 DB 00 00	.00334 16748
.00 1C 00 00	.00042 72460	.00 5C 00 00	.00140 38085	.00 9C 00 00	.00238 03710	.00 DC 00 00	.00335 69335
.00 1D 00 00	.00044 25048	.00 5D 00 00	.00141 90673	.00 9D 00 00	.00239 56298	.00 DD 00 00	.00337 21923
.00 1E 00 00	.00045 77636	.00 5E 00 00	.00143 43261	.00 9E 00 00	.00241 08886	.00 DE 00 00	.00338 74511
.00 1F 00 00	.00047 30224	.00 5F 00 00	.00144 95849	.00 9F 00 00	.00242 61474	.00 DF 00 00	.00340 27099
.00 20 00 00	.00048 82812	.00 60 00 00	.00146 48437	.00 A0 00 00	.00244 14062	.00 E0 00 00	.00341 79687
.00 21 00 00	.00050 35400	.00 61 00 00	.00148 01025	.00 A1 00 00	.00245 66650	.00 E1 00 00	.00343 32275
.00 22 00 00	.00051 87988	.00 62 00 00	.00149 53613	.00 A2 00 00	.00247 19238	.00 E2 00 00	.00344 84863
.00 23 00 00	.00053 40576	.00 63 00 00	.00151 06201	.00 A3 00 00	.00248 71826	.00 E3 00 00	.00346 37451
.00 24 00 00	.00054 93164	.00 64 00 00	.00152 58789	.00 A4 00 00	.00250 24414	.00 E4 00 00	.00347 90039
.00 25 00 00	.00056 45751	.00 65 00 00	.00154 11376	.00 A5 00 00	.00251 77001	.00 E5 00 00	.00349 42626
.00 26 00 00	.00057 98339	.00 66 00 00	.00155 63964	.00 A6 00 00	.00253 29589	.00 E6 00 00	.00350 95214
.00 27 00 00	.00059 50927	.00 67 00 00	.00157 16552	.00 A7 00 00	.00254 82177	.00 E7 00 00	.00352 47802
.00 28 00 00	.00061 03515	.00 68 00 00	.00158 69140	.00 A8 00 00	.00256 34765	.00 E8 00 00	.00354 00390
.00 29 00 00	.00062 56103	.00 69 00 00	.00160 21728	.00 A9 00 00	.00257 87353	.00 E9 00 00	.00355 52978
.00 2A 00 00	.00064 08691	.00 6A 00 00	.00161 74316	.00 AA 00 00	.00259 39941	.00 EA 00 00	.00357 05566
.00 2B 00 00	.00065 61279	.00 6B 00 00	.00163 26904	.00 AB 00 00	.00260 92529	.00 EB 00 00	.00358 58154
.00 2C 00 00	.00067 13867	.00 6C 00 00	.00164 79492	.00 AC 00 00	.00262 45117	.00 EC 00 00	.00360 10742
.00 2D 00 00	.00068 66455	.00 6D 00 00	.00166 32080	.00 AD 00 00	.00263 97705	.00 ED 00 00	.00361 63330
.00 2E 00 00	.00070 19042	.00 6E 00 00	.00167 84667	.00 AE 00 00	.00265 50292	.00 EE 00 00	.00363 15917
.00 2F 00 00	.00071 71630	.00 6F 00 00	.00169 37255	.00 AF 00 00	.00267 02880	.00 EF 00 00	.00364 68505
.00 30 00 00	.00073 24218	.00 70 00 00	.00170 89843	.00 B0 00 00	.00268 55468	.00 F0 00 00	.00366 21093
.00 31 00 00	.00074 76806	.00 71 00 00	.00172 42431	.00 B1 00 00	.00270 08056	.00 F1 00 00	.00367 73681
.00 32 00 00	.00076 29394	.00 72 00 00	.00173 95019	.00 B2 00 00	.00271 60644	.00 F2 00 00	.00369 26269
.00 33 00 00	.00077 81982	.00 73 00 00	.00175 47607	.00 B3 00 00	.00273 13232	.00 F3 00 00	.00370 78857
.00 34 00 00	.00079 34570	.00 74 00 00	.00177 00195	.00 B4 00 00	.00274 65820	.00 F4 00 00	.00372 31445
.00 35 00 00	.00080 87158	.00 75 00 00	.00178 52783	.00 B5 00 00	.00276 18408	.00 F5 00 00	.00373 84033
.00 36 00 00	.00082 39746	.00 76 00 00	.00180 05371	.00 B6 00 00	.00277 70996	.00 F6 00 00	.00375 36621
.00 37 00 00	.00083 92333	.00 77 00 00	.00181 57958	.00 B7 00 00	.00279 23583	.00 F7 00 00	.00376 89208
.00 38 00 00	.00085 44921	.00 78 00 00	.00183 10546	.00 B8 00 00	.00280 76171	.00 F8 00 00	.00378 41796
.00 39 00 00	.00086 97509	.00 79 00 00	.00184 63134	.00 B9 00 00	.00282 28759	.00 F9 00 00	.00379 94384
.00 3A 00 00	.00088 50097	.00 7A 00 00	.00186 15722	.00 BA 00 00	.00283 81347	.00 FA 00 00	.00381 46972
.00 3B 00 00	.00090 02685	.00 7B 00 00	.00187 68310	.00 BB 00 00	.00285 33935	.00 FB 00 00	.00382 99560
.00 3C 00 00	.00091 55273	.00 7C 00 00	.00189 20898	.00 BC 00 00	.00286 86523	.00 FC 00 00	.00384 52148
.00 3D 00 00	.00093 07861	.00 7D 00 00	.00190 73486	.00 BD 00 00	.00288 39111	.00 FD 00 00	.00386 04736
.00 3E 00 00	.00094 60449	.00 7E 00 00	.00192 26074	.00 BE 00 00	.00289 91699	.00 FE 00 00	.00387 57324
.00 3F 00 00	.00096 13037	.00 7F 00 00	.00193 78662	.00 BF 00 00	.00291 44287	.00 FF 00 00	.00389 09912

HEXADECIMAL - DECIMAL FRACTION CONVERSION TABLE (cont.)

Hexadecimal	Decimal	Hexadecimal	Decimal	Hexadecimal	Decimal	Hexadecimal	Decimal
.00 00 00 00	.00000 00000	.00 00 40 00	.00000 38146	.00 00 80 00	.00000 76293	.00 00 C0 00	.00001 14440
.00 00 01 00	.00000 00596	.00 00 41 00	.00000 38743	.00 00 81 00	.00000 76889	.00 00 C1 00	.00001 15036
.00 00 02 00	.00000 01192	.00 00 42 00	.00000 39339	.00 00 82 00	.00000 77486	.00 00 C2 00	.00001 15633
.00 00 03 00	.00000 01788	.00 00 43 00	.00000 39935	.00 00 83 00	.00000 78082	.00 00 C3 00	.00001 16229
.00 00 04 00	.00000 02384	.00 00 44 00	.00000 40531	.00 00 84 00	.00000 78678	.00 00 C4 00	.00001 16825
.00 00 05 00	.00000 02980	.00 00 45 00	.00000 41127	.00 00 85 00	.00000 79274	.00 00 C5 00	.00001 17421
.00 00 06 00	.00000 03576	.00 00 46 00	.00000 41723	.00 00 86 00	.00000 79870	.00 00 C6 00	.00001 18017
.00 00 07 00	.00000 04172	.00 00 47 00	.00000 42319	.00 00 87 00	.00000 80466	.00 00 C7 00	.00001 18613
.00 00 08 00	.00000 04768	.00 00 48 00	.00000 42915	.00 00 88 00	.00000 81062	.00 00 C8 00	.00001 19209
.00 00 09 00	.00000 05364	.00 00 49 00	.00000 43511	.00 00 89 00	.00000 81658	.00 00 C9 00	.00001 19805
.00 00 0A 00	.00000 05960	.00 00 4A 00	.00000 44107	.00 00 8A 00	.00000 82254	.00 00 CA 00	.00001 20401
.00 00 0B 00	.00000 06556	.00 00 4B 00	.00000 44703	.00 00 8B 00	.00000 82850	.00 00 CB 00	.00001 20997
.00 00 0C 00	.00000 07152	.00 00 4C 00	.00000 45299	.00 00 8C 00	.00000 83446	.00 00 CC 00	.00001 21593
.00 00 0D 00	.00000 07748	.00 00 4D 00	.00000 45895	.00 00 8D 00	.00000 84042	.00 00 CD 00	.00001 22189
.00 00 0E 00	.00000 08344	.00 00 4E 00	.00000 46491	.00 00 8E 00	.00000 84638	.00 00 CE 00	.00001 22785
.00 00 0F 00	.00000 08940	.00 00 4F 00	.00000 47087	.00 00 8F 00	.00000 85234	.00 00 CF 00	.00001 23381
.00 00 10 00	.00000 09536	.00 00 50 00	.00000 47683	.00 00 90 00	.00000 85830	.00 00 D0 00	.00001 23977
.00 00 11 00	.00000 10132	.00 00 51 00	.00000 48279	.00 00 91 00	.00000 86426	.00 00 D1 00	.00001 24573
.00 00 12 00	.00000 10728	.00 00 52 00	.00000 48875	.00 00 92 00	.00000 87022	.00 00 D2 00	.00001 25169
.00 00 13 00	.00000 11324	.00 00 53 00	.00000 49471	.00 00 93 00	.00000 87618	.00 00 D3 00	.00001 25765
.00 00 14 00	.00000 11920	.00 00 54 00	.00000 50067	.00 00 94 00	.00000 88214	.00 00 D4 00	.00001 26361
.00 00 15 00	.00000 12516	.00 00 55 00	.00000 50663	.00 00 95 00	.00000 88810	.00 00 D5 00	.00001 26957
.00 00 16 00	.00000 13113	.00 00 56 00	.00000 51259	.00 00 96 00	.00000 89406	.00 00 D6 00	.00001 27553
.00 00 17 00	.00000 13709	.00 00 57 00	.00000 51856	.00 00 97 00	.00000 90003	.00 00 D7 00	.00001 28149
.00 00 18 00	.00000 14305	.00 00 58 00	.00000 52452	.00 00 98 00	.00000 90599	.00 00 D8 00	.00001 28746
.00 00 19 00	.00000 14901	.00 00 59 00	.00000 53048	.00 00 99 00	.00000 91195	.00 00 D9 00	.00001 29342
.00 00 1A 00	.00000 15497	.00 00 5A 00	.00000 53644	.00 00 9A 00	.00000 91791	.00 00 DA 00	.00001 29938
.00 00 1B 00	.00000 16093	.00 00 5B 00	.00000 54240	.00 00 9B 00	.00000 92387	.00 00 DB 00	.00001 30534
.00 00 1C 00	.00000 16689	.00 00 5C 00	.00000 54836	.00 00 9C 00	.00000 92983	.00 00 DC 00	.00001 31130
.00 00 1D 00	.00000 17285	.00 00 5D 00	.00000 55432	.00 00 9D 00	.00000 93579	.00 00 DD 00	.00001 31726
.00 00 1E 00	.00000 17881	.00 00 5E 00	.00000 56028	.00 00 9E 00	.00000 94175	.00 00 DE 00	.00001 32322
.00 00 1F 00	.00000 18477	.00 00 5F 00	.00000 56624	.00 00 9F 00	.00000 94771	.00 00 DF 00	.00001 32918
.00 00 20 00	.00000 19073	.00 00 60 00	.00000 57220	.00 00 A0 00	.00000 95367	.00 00 E0 00	.00001 33514
.00 00 21 00	.00000 19669	.00 00 61 00	.00000 57816	.00 00 A1 00	.00000 95963	.00 00 E1 00	.00001 34110
.00 00 22 00	.00000 20265	.00 00 62 00	.00000 58412	.00 00 A2 00	.00000 96559	.00 00 E2 00	.00001 34706
.00 00 23 00	.00000 20861	.00 00 63 00	.00000 59008	.00 00 A3 00	.00000 97155	.00 00 E3 00	.00001 35302
.00 00 24 00	.00000 21457	.00 00 64 00	.00000 59604	.00 00 A4 00	.00000 97751	.00 00 E4 00	.00001 35898
.00 00 25 00	.00000 22053	.00 00 65 00	.00000 60200	.00 00 A5 00	.00000 98347	.00 00 E5 00	.00001 36494
.00 00 26 00	.00000 22649	.00 00 66 00	.00000 60796	.00 00 A6 00	.00000 98943	.00 00 E6 00	.00001 37090
.00 00 27 00	.00000 23245	.00 00 67 00	.00000 61392	.00 00 A7 00	.00000 99539	.00 00 E7 00	.00001 37686
.00 00 28 00	.00000 23841	.00 00 68 00	.00000 61988	.00 00 A8 00	.00001 00135	.00 00 E8 00	.00001 38282
.00 00 29 00	.00000 24437	.00 00 69 00	.00000 62584	.00 00 A9 00	.00001 00731	.00 00 E9 00	.00001 38878
.00 00 2A 00	.00000 25033	.00 00 6A 00	.00000 63180	.00 00 AA 00	.00001 01327	.00 00 EA 00	.00001 39474
.00 00 2B 00	.00000 25629	.00 00 6B 00	.00000 63776	.00 00 AB 00	.00001 01923	.00 00 EB 00	.00001 40070
.00 00 2C 00	.00000 26226	.00 00 6C 00	.00000 64373	.00 00 AC 00	.00001 02519	.00 00 EC 00	.00001 40666
.00 00 2D 00	.00000 26822	.00 00 6D 00	.00000 64969	.00 00 AD 00	.00001 03116	.00 00 ED 00	.00001 41263
.00 00 2E 00	.00000 27418	.00 00 6E 00	.00000 65565	.00 00 AE 00	.00001 03712	.00 00 EE 00	.00001 41859
.00 00 2F 00	.00000 28014	.00 00 6F 00	.00000 66161	.00 00 AF 00	.00001 04308	.00 00 EF 00	.00001 42455
.00 00 30 00	.00000 28610	.00 00 70 00	.00000 66757	.00 00 B0 00	.00001 04904	.00 00 F0 00	.00001 43051
.00 00 31 00	.00000 29206	.00 00 71 00	.00000 67353	.00 00 B1 00	.00001 05500	.00 00 F1 00	.00001 43647
.00 00 32 00	.00000 29802	.00 00 72 00	.00000 67949	.00 00 B2 00	.00001 06096	.00 00 F2 00	.00001 44243
.00 00 33 00	.00000 30398	.00 00 73 00	.00000 68545	.00 00 B3 00	.00001 06692	.00 00 F3 00	.00001 44839
.00 00 34 00	.00000 30994	.00 00 74 00	.00000 69141	.00 00 B4 00	.00001 07288	.00 00 F4 00	.00001 45435
.00 00 35 00	.00000 31590	.00 00 75 00	.00000 69737	.00 00 B5 00	.00001 07884	.00 00 F5 00	.00001 46031
.00 00 36 00	.00000 32186	.00 00 76 00	.00000 70333	.00 00 B6 00	.00001 08480	.00 00 F6 00	.00001 46627
.00 00 37 00	.00000 32782	.00 00 77 00	.00000 70929	.00 00 B7 00	.00001 09076	.00 00 F7 00	.00001 47223
.00 00 38 00	.00000 33378	.00 00 78 00	.00000 71525	.00 00 B8 00	.00001 09672	.00 00 F8 00	.00001 47819
.00 00 39 00	.00000 33974	.00 00 79 00	.00000 72121	.00 00 B9 00	.00001 10268	.00 00 F9 00	.00001 48415
.00 00 3A 00	.00000 34570	.00 00 7A 00	.00000 72717	.00 00 BA 00	.00001 10864	.00 00 FA 00	.00001 49011
.00 00 3B 00	.00000 35166	.00 00 7B 00	.00000 73313	.00 00 BB 00	.00001 11460	.00 00 FB 00	.00001 49607
.00 00 3C 00	.00000 35762	.00 00 7C 00	.00000 73909	.00 00 BC 00	.00001 12056	.00 00 FC 00	.00001 50203
.00 00 3D 00	.00000 36358	.00 00 7D 00	.00000 74505	.00 00 BD 00	.00001 12652	.00 00 FD 00	.00001 50799
.00 00 3E 00	.00000 36954	.00 00 7E 00	.00000 75101	.00 00 BE 00	.00001 13248	.00 00 FE 00	.00001 51395
.00 00 3F 00	.00000 37550	.00 00 7F 00	.00000 75697	.00 00 BF 00	.00001 13844	.00 00 FF 00	.00001 51991

HEXADECIMAL - DECIMAL FRACTION CONVERSION TABLE (cont.)

Hexadecimal	Decimal	Hexadecimal	Decimal	Hexadecimal	Decimal	Hexadecimal	Decimal
.00 00 00 00	.00000 00000	.00 00 00 40	.00000 00149	.00 00 00 80	.00000 00298	.00 00 00 C0	.00000 00447
.00 00 00 01	.00000 00002	.00 00 00 41	.00000 00151	.00 00 00 81	.00000 00300	.00 00 00 C1	.00000 00449
.00 00 00 02	.00000 00004	.00 00 00 42	.00000 00153	.00 00 00 82	.00000 00302	.00 00 00 C2	.00000 00451
.00 00 00 03	.00000 00006	.00 00 00 43	.00000 00155	.00 00 00 83	.00000 00305	.00 00 00 C3	.00000 00454
.00 00 00 04	.00000 00009	.00 00 00 44	.00000 00158	.00 00 00 84	.00000 00307	.00 00 00 C4	.00000 00456
.00 00 00 05	.00000 00011	.00 00 00 45	.00000 00160	.00 00 00 85	.00000 00309	.00 00 00 C5	.00000 00458
.00 00 00 06	.00000 00013	00 00 00 46	.00000 00162	.00 00 00 86	.00000 00311	.00 00 00 C6	.00000 00461
.00 00 00 07	.00000 00016	.00 00 00 47	.00000 00165	.00 00 00 87	.00000 00314	.00 00 00 C7	.00000 00463
.00 00 00 08	.00000 00018	.00 00 00 48	.00000 00167	.00 00 00 88	.00000 00316	.00 00 00 C8	.00000 00465
.00 00 00 09	.00000 00020	.00 00 00 49	.00000 00169	.00 00 00 89	.00000 00318	.00 00 00 C9	.00000 00467
.00 00 00 0A	.00000 00023	.00 00 00 4A	.00000 00172	.00 00 00 8A	.00000 00321	.00 00 00 CA	.00000 00470
.00 00 00 0B	.00000 00025	.00 00 00 4B	.00000 00174	.00 00 00 8B	.00000 00323	.00 00 00 CB	.00000 00472
.00 00 00 0C	.00000 00027	.00 00 00 4C	.00000 00176	.00 00 00 8C	.00000 00325	.00 00 00 CC	.00000 00474
.00 00 00 0D	.00000 00030	.00 00 00 4D	.00000 00179	.00 00 00 8D	.00000 00328	.00 00 00 CD	.00000 00477
.00 00 00 0E	.00000 00032	.00 00 00 4E	.00000 00181	.00 00 00 8E	.00000 00330	.00 00 00 CE	.00000 00479
.00 00 00 0F	.00000 00034	.00 00 00 4F	.00000 00183	.00 00 00 8F	.00000 00332	.00 00 00 CF	.00000 00481
.00 00 00 10	.00000 00037	.00 00 00 50	.00000 00186	.00 00 00 90	.00000 00335	.00 00 00 D0	.00000 00484
.00 00 00 11	.00000 00039	.00 00 00 51	.00000 00188	.00 00 00 91	.00000 00337	.00 00 00 D1	.00000 00486
.00 00 00 12	.00000 00041	.00 00 00 52	.00000 00190	.00 00 00 92	.00000 00339	.00 00 00 D2	.00000 00488
.00 00 00 13	.00000 00044	.00 00 00 53	.00000 00193	.00 00 00 93	.00000 00342	.00 00 00 D3	.00000 00491
.00 00 00 14	.00000 00046	.00 00 00 54	.00000 00195	.00 00 00 94	.00000 00344	.00 00 00 D4	.00000 00493
.00 00 00 15	.00000 00048	.00 00 00 55	.00000 00197	.00 00 00 95	.00000 00346	.00 00 00 D5	.00000 00495
.00 00 00 16	.00000 00051	.00 00 00 56	.00000 00200	.00 00 00 96	.00000 00349	.00 00 00 D6	.00000 00498
.00 00 00 17	.00000 00053	.00 00 00 57	.00000 00202	.00 00 00 97	.00000 00351	.00 00 00 D7	.00000 00500
.00 00 00 18	.00000 00055	.00 00 00 58	.00000 00204	.00 00 00 98	.00000 00353	.00 00 00 D8	.00000 00502
.00 00 00 19	.00000 00058	.00 00 00 59	.00000 00207	.00 00 00 99	.00000 00356	.00 00 00 D9	.00000 00505
.00 00 00 1A	.00000 00060	.00 00 00 5A	.00000 00209	.00 00 00 9A	.00000 00358	.00 00 00 DA	.00000 00507
.00 00 00 1B	.00000 00062	.00 00 00 5B	.00000 00211	.00 00 00 9B	.00000 00360	.00 00 00 DB	.00000 00509
.00 00 00 1C	.00000 00065	.00 00 00 5C	.00000 00214	.00 00 00 9C	.00000 00363	.00 00 00 DC	.00000 00512
.00 00 00 1D	.00000 00067	.00 00 00 5D	.00000 00216	.00 00 00 9D	.00000 00365	.00 00 00 DD	.00000 00514
.00 00 00 1E	.00000 00069	.00 00 00 5E	.00000 00218	.00 00 00 9E	.00000 00367	.00 00 00 DE	.00000 00516
.00 00 00 1F	.00000 00072	.00 00 00 5F	.00000 00221	.00 00 00 9F	.00000 00370	.00 00 00 DF	.00000 00519
.00 00 00 20	.00000 00074	.00 00 00 60	.00000 00223	.00 00 00 A0	.00000 00372	.00 00 00 E0	.00000 00521
.00 00 00 21	.00000 00076	.00 00 00 61	.00000 00225	.00 00 00 A1	.00000 00374	.00 00 00 E1	.00000 00523
.00 00 00 22	.00000 00079	.00 00 00 62	.00000 00228	.00 00 00 A2	.00000 00377	.00 00 00 E2	.00000 00526
.00 00 00 23	.00000 00081	.00 00 00 63	.00000 00230	.00 00 00 A3	.00000 00379	.00 00 00 E3	.00000 00528
.00 00 00 24	.00000 00083	.00 00 00 64	.00000 00232	.00 00 00 A4	.00000 00381	.00 00 00 E4	.00000 00530
.00 00 00 25	.00000 00086	.00 00 00 65	.00000 00235	.00 00 00 A5	.00000 00384	.00 00 00 E5	.00000 00533
.00 00 00 26	.00000 00088	.00 00 00 66	.00000 00237	.00 00 00 A6	.00000 00386	.00 00 00 E6	.00000 00535
.00 00 00 27	.00000 00090	.00 00 00 67	.00000 00239	.00 00 00 A7	.00000 00388	.00 00 00 E7	.00000 00537
.00 00 00 28	.00000 00093	.00 00 00 68	.00000 00242	.00 00 00 A8	.00000 00391	.00 00 00 E8	.00000 00540
.00 00 00 29	.00000 00095	.00 00 00 69	.00000 00244	.00 00 00 A9	.00000 00393	.00 00 00 E9	.00000 00542
.00 00 00 2A	.00000 00097	.00 00 00 6A	.00000 00246	.00 00 00 AA	.00000 00395	.00 00 00 EA	.00000 00544
.00 00 00 2B	.00000 00100	.00 00 00 6B	.00000 00249	.00 00 00 AB	.00000 00398	.00 00 00 EB	.00000 00547
.00 00 00 2C	.00000 00102	.00 00 00 6C	.00000 00251	.00 00 00 AC	.00000 00400	.00 00 00 EC	.00000 00549
.00 00 00 2D	.00000 00104	.00 00 00 6D	.00000 00253	.00 00 00 AD	.00000 00402	.00 00 00 ED	.00000 00551
.00 00 00 2E	.00000 00107	.00 00 00 6E	.00000 00256	.00 00 00 AE	.00000 00405	.00 00 00 EE	.00000 00554
.00 00 00 2F	.00000 00109	.00 00 00 6F	.00000 00258	.00 00 00 AF	.00000 00407	.00 00 00 EF	.00000 00556
.00 00 00 30	.00000 00111	.00 00 00 70	.00000 00260	.00 00 00 B0	.00000 00409	.00 00 00 F0	.00000 00558
.00 00 00 31	.00000 00114	.00 00 00 71	.00000 00263	.00 00 00 B1	.00000 00412	.00 00 00 F1	.00000 00561
.00 00 00 32	.00000 00116	.00 00 00 72	.00000 00265	.00 00 00 B2	.00000 00414	.00 00 00 F2	.00000 00563
.00 00 00 33	.00000 00118	.00 00 00 73	.00000 00267	.00 00 00 B3	.00000 00416	.00 00 00 F3	.00000 00565
.00 00 00 34	.00000 00121	.00 00 00 74	.00000 00270	.00 00 00 B4	.00000 00419	.00 00 00 F4	.00000 00568
.00 00 00 35	.00000 00123	.00 00 00 75	.00000 00272	.00 00 00 B5	.00000 00421	.00 00 00 F5	.00000 00570
.00 00 00 36	.00000 00125	.00 00 00 76	.00000 00274	.00 00 00 B6	.00000 00423	.00 00 00 F6	.00000 00572
.00 00 00 37	.00000 00128	.00 00 00 77	.00000 00277	.00 00 00 B7	.00000 00426	.00 00 00 F7	.00000 00575
.00 00 00 38	.00000 00130	.00 00 00 78	.00000 00279	.00 00 00 B8	.00000 00428	.00 00 00 F8	.00000 00577
.00 00 00 39	.00000 00132	.00 00 00 79	.00000 00281	.00 00 00 B9	.00000 00430	.00 00 00 F9	.00000 00579
.00 00 00 3A	.00000 00135	.00 00 00 7A	.00000 00284	.00 00 00 BA	.00000 00433	.00 00 00 FA	.00000 00582
.00 00 00 3B	.00000 00137	.00 00 00 7B	.00000 00286	.00 00 00 BB	.00000 00435	.00 00 00 FB	.00000 00584
.00 00 00 3C	.00000 00139	.00 00 00 7C	.00000 00288	.00 00 00 BC	.00000 00437	.00 00 00 FC	.00000 00586
.00 00 00 3D	.00000 00142	.00 00 00 7D	.00000 00291	.00 00 00 BD	.00000 00440	.00 00 00 FD	.00000 00589
.00 00 00 3E	.00000 00144	.00 00 00 7E	.00000 00293	.00 00 00 BE	.00000 00442	.00 00 00 FE	.00000 00591
.00 00 00 3F	.00000 00146	.00 00 00 7F	.00000 00295	.00 00 00 BF	.00000 00444	.00 00 00 FF	.00000 00593

MATHEMATICAL CONSTANTS

Constant	Decimal Value	Hexadecimal Value
π	3.14159 26535 89793	3.243F 6A89
π^{-1}	0.31830 98861 83790	0.517C C1B7
$\sqrt{\pi}$	1.77245 38509 05516	1.C5BF 891C
$\ln \pi$	1.14472 98858 49400	1.250D 048F
e	2.71828 18284 59045	2.B7E1 5163
e^{-1}	0.36787 94411 71442	0.5E2D 58D9
$\sqrt{e}$	1.64872 12707 00128	1.A612 98E2
$\log_{10} e$	0.43429 44819 03252	0.6F2D EC55
$\log_2 e$	1.44269 50408 88963	1.7154 7653
γ	0.57721 56649 01533	0.93C4 67E4
$\ln \gamma$	-0.54953 93129 81645	-0.8CAE 9BC1
$\sqrt{2}$	1.41421 35623 73095	1.6A09 E668
ln2	0.69314 71805 59945	0.B172 17F8
$\log_{10} 2$	0.30102 99956 63981	0.4D10 4D42
$\sqrt{10}$	3.16227 76601 68379	3.298B 075C
ln10	2.30258 50929 94046	2.4D76 3777

TABLE OF POWERS OF TWO

2^n	n	2^{-n}
1	0	1.0
2	1	0.5
4	2	0.25
8	3	0.125
16	4	0.062 5
32	5	0.031 25
64	6	0.015 625
128	7	0.007 812 5
256	8	0.003 906 25
512	9	0.001 953 125
1 024	10	0.000 976 562 5
2 048	11	0.000 488 281 25
4 096	12	0.000 244 140 625
8 192	13	0.000 122 070 312 5
16 384	14	0.000 061 035 156 25
32 768	15	0.000 030 517 578 125
65 536	16	0.000 015 258 789 062 5
131 072	17	0.000 007 629 394 531 25
262 144	18	0.000 003 814 697 265 625
524 288	19	0.000 001 907 348 632 812 5
1 048 576	20	0.000 000 953 674 316 406 25
2 097 152	21	0.000 000 476 837 158 203 125
4 194 304	22	0.000 000 238 418 579 101 562 5
8 388 608	23	0.000 000 119 209 289 550 781 25
16 777 216	24	0.000 000 059 604 644 775 390 625
33 554 432	25	0.000 000 029 802 322 387 695 312 5
67 108 864	26	0.000 000 014 901 161 193 847 656 25
134 217 728	27	0.000 000 007 450 580 596 923 828 125
268 435 456	28	0.000 000 003 725 290 298 461 914 062 5
536 870 912	29	0.000 000 001 862 645 149 230 957 031 25
1 073 741 824	30	0.000 000 000 931 322 574 615 478 515 625
2 147 483 648	31	0.000 000 000 465 661 287 307 739 257 812 5
4 294 967 296	32	0.000 000 000 232 830 643 653 869 628 906 25
8 589 934 592	33	0.000 000 000 116 415 321 826 934 814 453 125
17 179 869 184	34	0.000 000 000 058 207 660 913 467 407 226 562 5
34 359 738 368	35	0.000 000 000 029 103 830 456 733 703 613 281 25
68 719 476 736	36	0.000 000 000 014 551 915 228 366 851 806 640 625
137 438 953 472	37	0.000 000 000 007 275 957 614 183 425 903 320 312 5
274 877 906 944	38	0.000 000 000 003 637 978 807 091 712 951 660 156 25
549 755 813 888	39	0.000 000 000 001 818 989 403 545 856 475 830 078 125
1 099 511 627 776	40	0.000 000 000 000 909 494 701 772 928 237 915 039 062 5
2 199 023 255 552	41	0.000 000 000 000 454 747 350 886 464 118 957 519 531 25
4 398 046 511 104	42	0.000 000 000 000 227 373 675 443 232 059 478 759 765 625
8 796 093 022 208	43	0.000 000 000 000 113 686 837 721 616 029 739 379 882 812 5
17 592 186 044 416	44	0.000 000 000 000 056 843 418 860 808 014 869 689 941 406 25
35 184 372 088 832	45	0.000 000 000 000 028 421 709 430 404 007 434 844 970 703 125
70 368 744 177 664	46	0.000 000 000 000 014 210 854 715 202 003 717 422 485 351 562 5
140 737 488 355 328	47	0.000 000 000 000 007 105 427 357 601 001 858 711 242 675 781 25
281 474 976 710 656	48	0.000 000 000 000 003 552 713 678 800 500 929 355 621 337 890 625

STANDARD CHARACTER CODES

SYMBOL	ASCII (HEX)	EBCDIC (HEX)	HOLLERITH (029)	HOLLERITH (026)
blank	A0	40	blank	
!	A1	5A	11-8-2	
"	A2	7F	8-7	0-8-5
#	A3	7B	8-3	0-8-7
$	A4	5B	11-8-3	
%	A5	6C	0-8-4	11-8-7
&	A6	50	12	12-8-7
'	A7	7D	8-5	8-4
(	A8	4D	12-8-5	0-8-4
)	A9	5D	11-8-5	12-8-4
*	AA	5C	11-8-4	
+	AB	4E	12-8-6	12
,	AC	6B	0-8-3	
–	AD	60	11	
.	AE	4B	12-8-3	
/	AF	61	0-1	
0	B0	F0	0	
1	B1	F1	1	
2	B2	F2	2	
3	B3	F3	3	
4	B4	F4	4	
5	B5	F5	5	
6	B6	F6	6	
7	B7	F7	7	
8	B8	F8	8	
9	B9	F9	9	
:	BA	7A	8-2	8-5
;	BB	5E	11-8-6	
<	BC	4C	12-8-4	12-8-6
=	BD	7E	8-6	8-3
>	BE	6E	0-8-6	8-6
?	BF	6F	0-8-7	12-8-2
@	C0	7C	8-4	0-8-2
A	C1	C1	12-1	
B	C2	C2	12-2	
C	C3	C3	12-3	
D	C4	C4	12-4	
E	C5	C5	12-5	
F	C6	C6	12-6	
G	C7	C7	12-7	
H	C8	C8	12-8	
I	C9	C9	12-9	
J	CA	D1	11-1	
K	CB	D2	11-2	
L	CC	D3	11-3	
M	CD	D4	11-4	
N	CE	D5	11-5	
O	CF	D6	11-6	
P	D0	D7	11-7	
Q	D1	D8	11-8	
R	D2	D9	11-9	
S	D3	E2	0-2	
T	D4	E3	0-3	
U	D5	E4	0-4	
V	D6	E5	0-5	
W	D7	E6	0-6	
X	D8	E7	0-7	
Y	D9	E8	0-8	
Z	DA	E9	0-9	
[	DB	4F	12-8-7	12-8-5
\	DC	4A	12-8-2	0-8-6
]	DD	5F	11-8-7	11-8-5
↑	DE	6D	0-8-5	8-7
←	DF	6A	0-8-2	8-2

TELETYPE CONTROL AND TRANSMISSION CODES

FUNCTION	ASCII
NULL	80
SOM (Print on)	81
EAO	82
EOM	83
EOT (Print off)	84
WRU	85
RU	86
BELL	87
FEO	88
H.TAB	89
LINE FEED	8A
V.TAB	8B
FORM	8C
CARRIAGE RETURN	8D
SO	8E
SI	8F
DCO	90
X-ON (Reader on)	91
TAPE (Punch on)	92
X-OFF (Reader off)	93
TAPE OFF (Punch off)	94
ERROR	95
SYNC	96
LEM	97
S0	98
S1	99
S2	9A
S3	9B
S4	9C
S5	9D
S6	9E
S7	9F

APPENDIX F

LEFT BLANK INTENTIONALLY

APPENDIX G

FRONT PANEL USAGE

APPENDIX G

CONTROL PANEL USAGE

Loading or examining a program provides an excellent example of sequential operations on consecutive memory locations from the control panel. Any sequence of control panel operations must begin with supplying the initial address to the program counter. Assume the program to be examined starts at location 3F0. With all register select switches off, place 3F0 in the PR and DB switches, and depress EXC. This will place the contents of the switches into the program counter to provide a starting address. Then the following switch settings produce the effects listed.

EXAMINE MEMORY AND MAKE CHANGES

SWITCH POSITION	ACTION
Memory DIS selected, EXC	Contents of location 3F0 displayed
" " " , STEP	" " " 3F1 "
" " " , STEP	" " " 3F2 "
" " " , STEP	" " " 3F3 "
" " " , STEP	" " " 3F4 "
" " " , STEP	" " " 3F5 "
Memory DIS deselected	(assume contents of 3F5 to be in error.)
Memory ENT selected, EXC	Contents of data switches written into location 3F5 and displayed
Memory ENT deselected	
Memory DIS selected, STEP	Contents of location 3F6 displayed
" " " , STEP	" " " 3F7 "
" " " , STEP	" " " 3F8 "

The contents of the program counter can be examined at any time by deselecting the memory DIS switch.

LOAD DATA INTO MEMORY AND EXAMINE

SWITCH POSITION	ACTION
Memory ENT selected, EXC	Contents of data switches displayed and written into location 3F0
Memory ENT selected, STEP	Increment program counter to 3F1 and display contents of 3F1
Memory ENT selected, EXC	Contents of data switches displayed and written into location 3F1
Memory ENT selected, STEP	Increment program counter to 3F2 and display contents of 3F2 (assume contents of 3F2 are already correct)
Memory ENT selected, STEP	Increment program counter to 3F3 and display contents of 3F3
Memory ENT selected, EXC	Contents of data switches displayed and written into location 3F3
Memory ENT selected, STEP	Increment program counter to 3F4 and display contents of 3F4

The contents of the program counter can be examined at any time by deselecting the Memory ENT switch.

STARTING A PROGRAM

Load the starting address into the program counter as explained above, with all register select switches down, depress the INIT switch momentarily and then depress RUN. The INIT switch will clear any Link or OVFL flags,

and provide a reset to the I/O controllers. Depressing the RUN switch will start the central processor executing instructions beginning at the starting address contained in the program counter. The central processor will remain in the run mode until halted by a Halt command or depressing the Step switch.

STEPPING THROUGH A PROGRAM

Load the starting address into the program counter as explained above. Depress the INIT switch momentarily. With all register select swtiches down, depressing the STEP switch will execute one instruction at a time. The central processor may be placed in the RUN mode at any time by depressing the RUN switch and then halted by depressing the STEP switch. Stepping of the program may continue from that point.

CHANGING THE CONTENTS OF THE ACCUMULATORS

Place the desired accumulator select switch (A or B) in the up position. At that time, depressing the EXC switch momentarily will take the contents of the DB data switches into the selected accumulator.

If both accumulator select switches are up, the contents of the DB data switches will be loaded into both accumulators at the same time. This feature is useful for clearing both accumulators from only one EXC action.

APPENDIX H

THE BOOTSTRAP ROUTINE

1.0 THE BOOTSTRAP ROUTINE

To load programs from the teletype, a small bootstrap program must first be loaded by hand. It loads a more complex program, the binary loader, which will load all other programs. The binary loader is normally left in the machine so that this operation is done only once.

Using the "step through memory" feature as explained in Appendix G, load the following bootstrap program. All program codes and addresses are hexadecimal. A number annotated X'100' indicates hexadecimal 100, also $100 indicates hexadecimal 100.

HEX ADDRESS	LABEL	HEX CODE	MNEMONICS	COMMENTS
F31 F32	BEG	53 00	DOA TTY	Turn On Reader
F33	BP2	02	DBB	Decrement B Register
F34 F35		BF 3D	STB WSL	Store "B" in Working Store Lower
F36 F37	TRR	4B 04	SRS	Skip if Reader Ready
F38 F39		CF 36	JPC TRR	Jump Back to See If Reader Ready
F3A F3B		52 00	DIA TTY	Input Reader Character to the A Register
F3C F3D	WSL	AF AF	STA XX	Store the A Register at WSL
F3E F3F		CF 33	JPC BP2	Jump Back to Get New Word

1.2 With the first fully punched line of the binary loader tape placed over the teletype reader pins, close the reader cap.

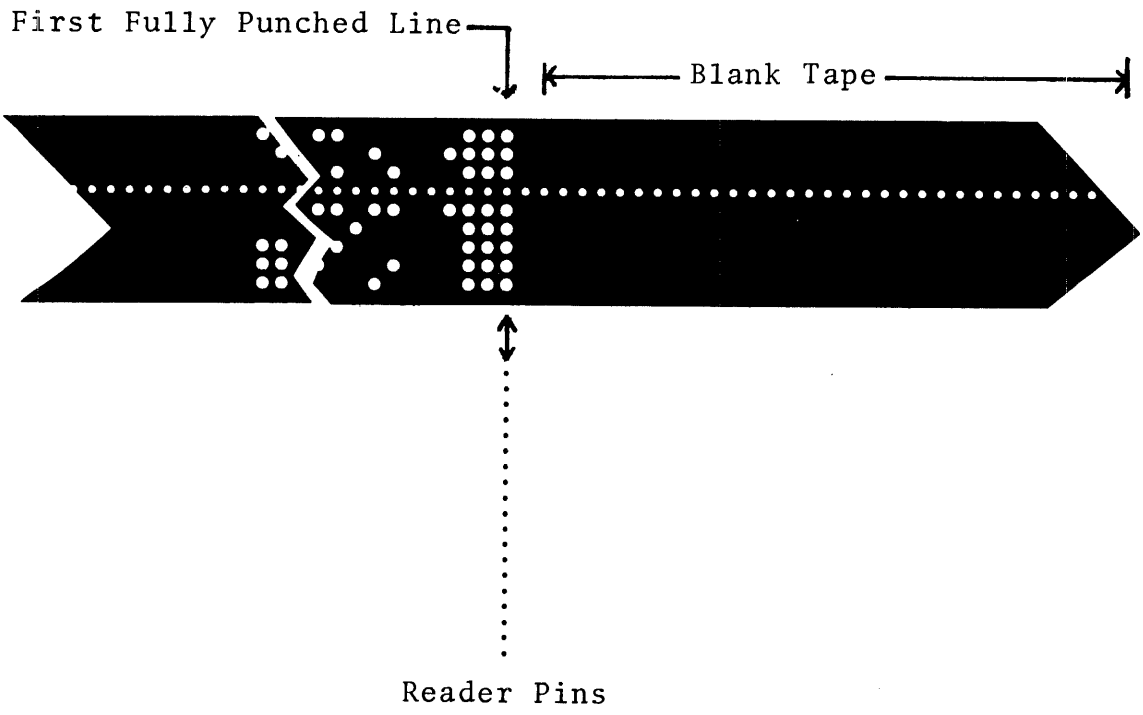


FIGURE H-1

1.3 Follow the procedure outlined below to load the binary loader program.

A. Set the "B" register to $00
Push "B" switch up ("A", "ENT" and "DIS" down).
Set "DB" switches to $00.
Depress and release "EXC" switch.

B. Set the "A" register to $11.
Push up only the "A" switch ("B", "ENT" and "DIS" down).
Set the "DB" switches equal to $11.
Depress and release the "EXC" switch.

C. Set the "P" register to F31.
Push down the "A", "B", "ENT" and "DIS" switches.
Set the "PR" and "DB" switches to F31.
Depress and release the "EXC" switch.

D. Depress and release the "INIT" (Initialize) switch.

E. Depress and release the "Run" switch.

F. The computer will turn the reader on and start loading tape.

1.1 SYSTEM FEATURES

Some of the standard features available with the MICRO 400 central processor include:

- Multi-Accumulator Arithmetic
- 105 Program Instructions
- Up to 32 Peripheral Devices
- Direct and Index Addressing
- Memory Directly Addressable in 4096 pages
- Two's Complement Arithmetic
- Local Operator's Control Panel (removable)
- Maintenance Programs
- Utility Debug Programs
- Full Line of Peripherals and Systems Interfaces
- Microbus Interconnection
- Programmed I/O bus
- Priority Interrupts

Some of the optional features available with the MICRO 400 central processor include:

- Base Register Addressing to 65K Words
- Automatic I/O
- Power Fail Protection from Run or Halt

1.2 ORGANIZATION

The MICRO 400 is organized around a single bus and consists of five 8-bit registers, two 12-bit registers and one 16-bit register. All information transmitted within the processor to and from all registers and to and from memory is transmitted in parallel on one data bus. The data on this bus at any one time are under the control of logic in the central processor. When the contents of one register are placed on the bus they are presented to the inputs of all registers and memory and are strobed into the proper destination by the control unit. This organization optimizes the MICRO 400 for maximum data throughtput and for minimum elapsed time between successive bus transfers. This unique organization makes possible the MICRObus concept.

The core memory is modular in 1024 or 4096 byte increments. Each module containing its own data and address registers. Memory addressing may be extended to 65,536 bytes with the Base Register Module option.

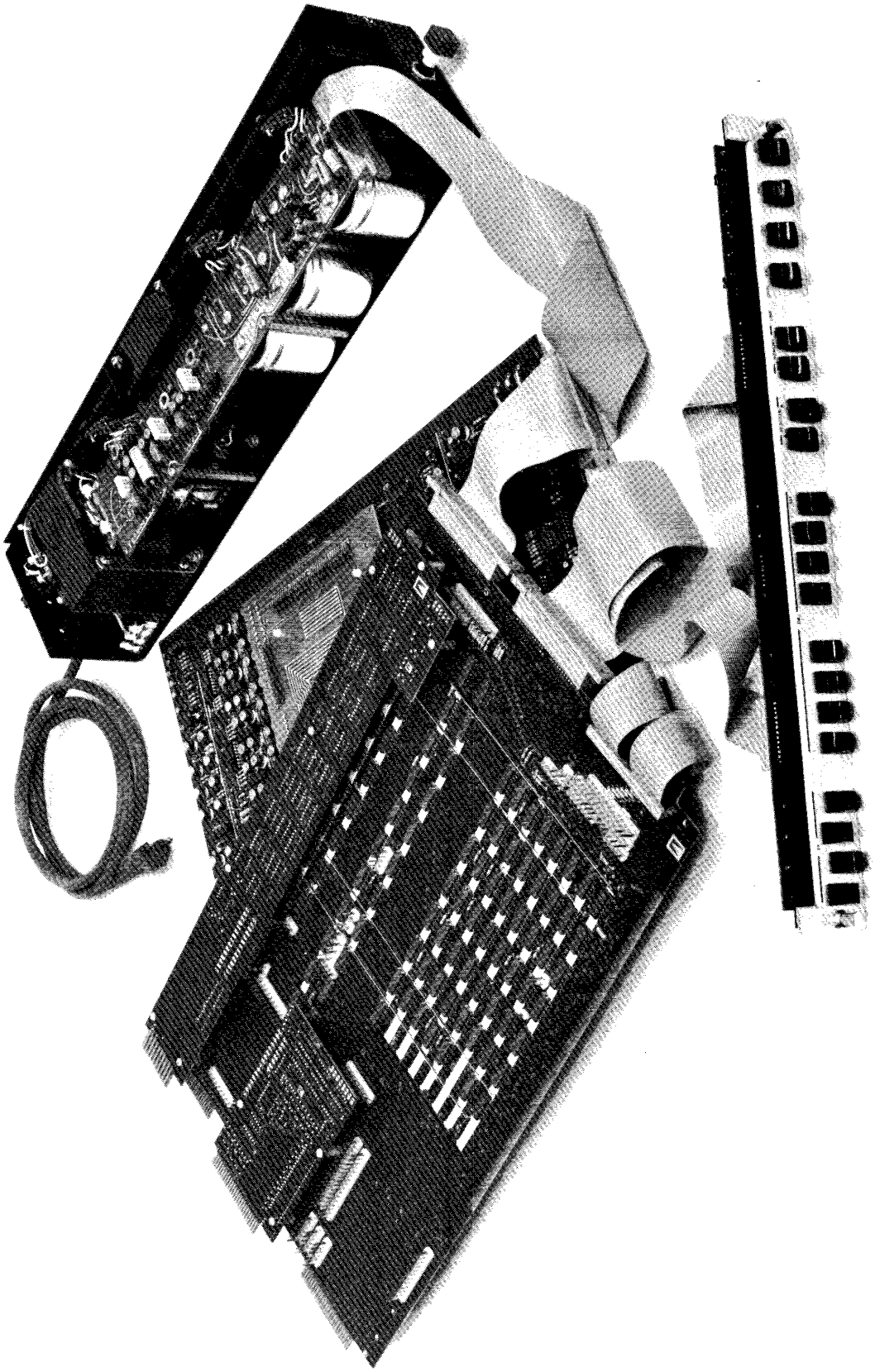


FIGURE 1-1
M400 CPU, MEMORY, POWER SUPPLY, FRONT PANEL INTERCONNECT