MIGHTY MUX

DMA MULTIPLEXER
USER MANUAL

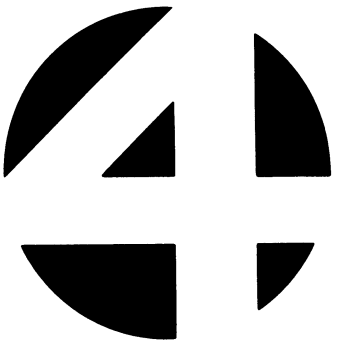
POINT 4
DATA CORPORATION

BRENDA

POINT 4 DATA CORPORATION
2569 McCabe Way / Irvine, California 92714

MIGHTY MUX

DMA MULTIPLEXER USER MANUAL



NOTICE

Every attempt has been made to make this reference manual complete, accurate and up-to-date. However, all information herein is subject to change due to updates. All inquiries concerning this manual should be directed to POINT 4 Data Corporation.

POINT 4 Data Corporation
2569 McCabe Way
Irvine, CA 92714
(714) 754-4114

REVISION RECORD

PUBLICATION NUMBER: HM-042-0015

Revision	Description	Date
A	Update of original manual (1977)	08/30/82

LIST OF EFFECTIVE PAGES

Changes, additions, and deletions to information in this manual are indicated by vertical bars in the margins or by a dot near the page number if the entire page is affected. A vertical bar by the page number indicates pagination rather than content has changed.

Page	Rev	Page	Rev	Page	Rev
Cover	-				
Title	-				
ii thru xi	A				
1-1 thru 1-10	A				
2-1 thru 2-28	A				
3-1 thru 3-46	A				
Appendix Title	-				
A-1 thru A-3	A				
B-1	A				
C-1	A				
D-1	A				
Comment Sheet	A				
Mailer	-				
Back Cover	-				

PREFACE

This manual includes general reference information for the POINT 4 MIGHTY MUX DMA Multiplexer. It describes system architecture, input/output interface, and specifications. It discusses aspects of hardware interface including installation, I/O bus interface signals, junction panel connections, hardware-selectable options and multiplexer timing. Programming information is also included.

Related manuals include:

Title	Pub. Number
MIGHTY MUX Diagnostics Manual	HM-042-0007
POINT 4 User Manual	(1979)
POINT 4 MARK 8 Reference Manual	HM-082-0021

CONTENTS

APPENDICES

FIGURES

TABLES

Section 1
INTRODUCTION

1.1 GENERAL DESCRIPTION

POINT 4 Data's MIGHTY MUX Multiplexer is a general purpose communications front-end for POINT 4 and Nova*-type computers. From four to 128 peripheral devices can be interfaced to the processor via the Direct Memory Access (DMA) channel, thus reducing I/O processing overhead on the CPU. A MIGHTY MUX system can handle up to 128 ports at 9600 baud, or up to 64 ports at 19.2K baud, all operating simultaneously.

The MIGHTY MUX consists of a Model 310 basic multiplexer with optional expansion boards (Model 301 series) for additional asynchronous ports. Model 310 is available as either a four-port (310-A4) or an eight-port (310-A8) controller. The 310 has, as standard features, a real-time clock, master terminal interface, and four or eight asynchronous serial ports which can handle CRTs, printing terminals, serial printers with RS-232 interface, or modem controls.

The following port parameters are under program control on a port-by-port basis: I/O buffer size and location, character size, parity, auto echo, interrupt conditions, and baud rate (a user can sign onto the system at one baud rate and switch to any other standard baud rate from the terminal).

The MIGHTY MUX multiplexer is software-compatible with the IRIS (Interactive Real-Time Information System) Operating System used on POINT 4 MARK 5/8 and Nova-type computers.

Throughput is limited primarily by the system software capabilities, since data-transfer handling is block oriented, rather than character-by-character oriented. The requirements for software handling of the transfer are reduced to starting transmission by setting appropriate control words, and receiving an interrupt when transmission is completed.

Figure 1-1 is an illustration of the MIGHTY MUX 310-A8 board. Figure 1-2 is a photograph of a 24-port system, consisting of 310-A8 and 301-A16 boards.

*Nova is a registered trademark of Data General Corporation.

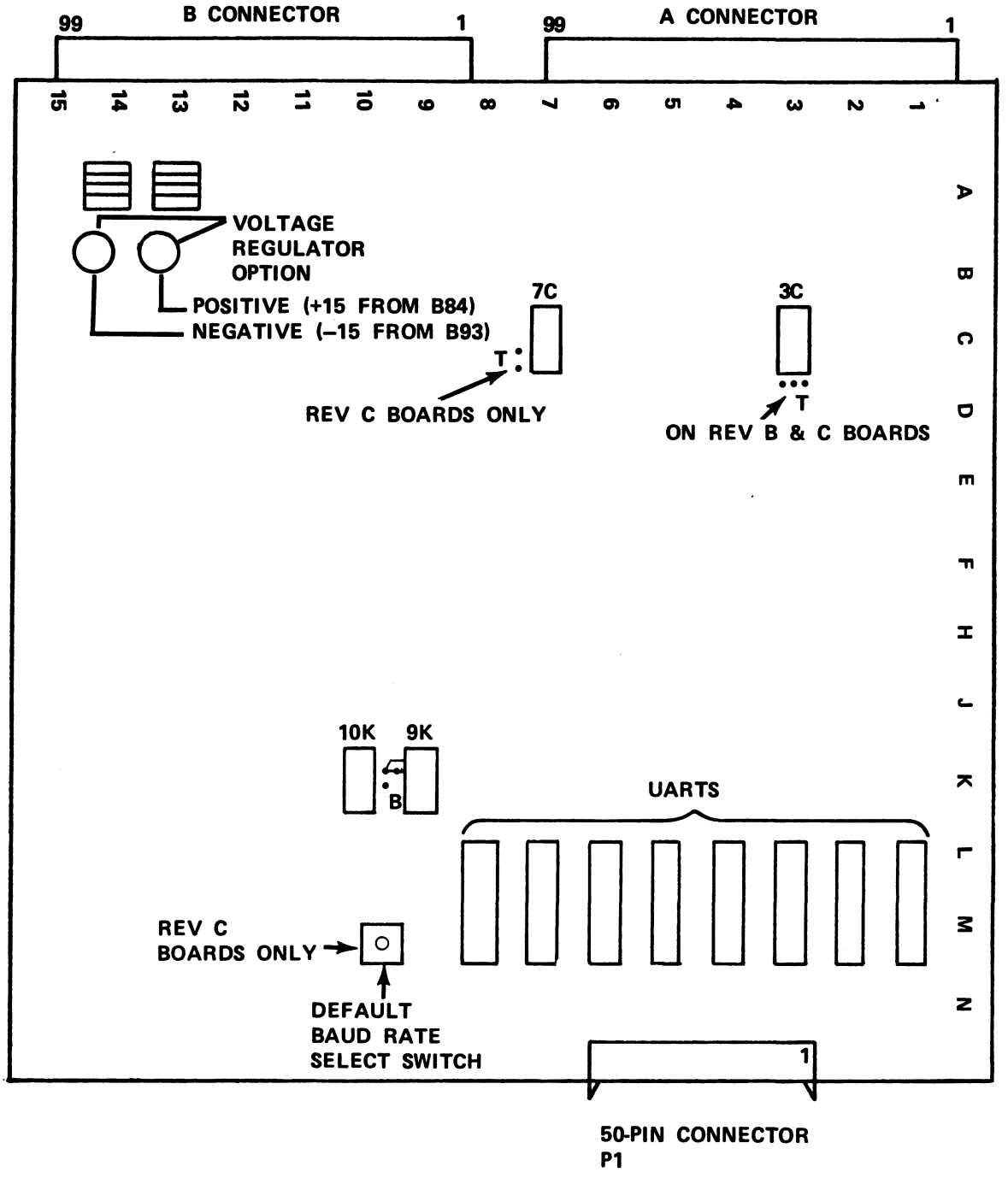


Figure 1-1. MIGHTY MUX 310-A8 Board

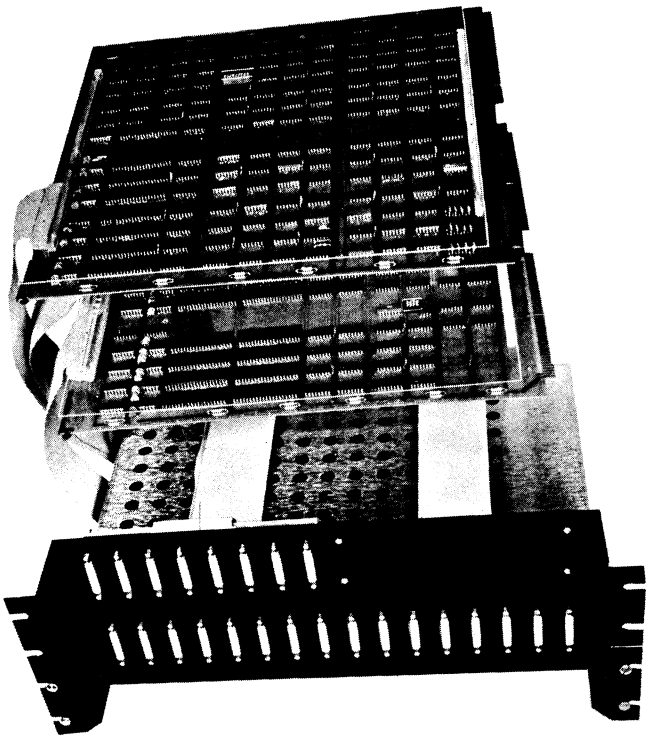


Figure 1-2. MIGHTY MUX 24-Port System, With 310-A8 and 301-A16 Boards

1.1.1 FEATURES

The MIGHTY MUX DMA Multiplexer greatly expands system performance capabilities. Its advanced design and versatility are evident in the following features:

- Direct memory access capability
- Interfaces terminals, line printers, modems
- Supports mixed line speeds and code levels
- Line speeds up to 9600 baud (19.2K baud optional)
- Supports character or block-oriented I/O tasks
- Single MIGHTY MUX 310 board includes four or eight multiplexer ports
- Expandable up to 128 ports
- Master Terminal Mode permits use of Port 0 as the master terminal interface (device code 10/11) or as a multiplexer port
- Software control of six port parameters
- Expansion boards offer 8, 16, or 24 ports per board
- Modem control available on each port
- Control character recognition
- Compatible with EIA RS-232C/CCITT V.24 or current loop (20, 40 or 60 milliamps)
- Switch-selectable master terminal baud rate

1.1.2 MULTIPLEXER MODELS

The MIGHTY MUX Multiplexer is available in several configurations. All configurations use the Model 310 basic multiplexer. Available models and their features are listed below. (See the Configuration Guide in Section 3.4.)

Model	Description
310-A4	4-port multiplexer (nonexpandable)
310-A8	8-port multiplexer expandable to 128 ports
301-A8	8-port expansion board
301-A16	16-port expansion board
301-A24	24-port expansion board
320	Junction panel for two connector/cable assemblies
322	Connector/cable assembly for 8 ports (RS-232C)
322-CL	Connector/cable assembly for 8 ports (RS-232C & current loop)
324-N	Interboard (Z) cable (N=number of 301 boards)
340-32	External Power supply for up to 32 ports
340-64	External Power supply for up to 64 ports
340-128	External Power supply for up to 128 ports
342-1	External Power supply for RS-232C and current loop operation; number of ports serviced is 32
342-2	External Power supply for RS-232C and current loop operation; number of ports serviced is 48
342-3	External Power supply for RS-232C and current loop operation; number of ports serviced is 64
342-4	External Power supply for RS-232C and current loop operation; number of ports serviced is 128
310-PS	On-board Power Supply Option for 310-A4 or 310-A8 (Replaces external supply for RS-232C; if current loop is desired, must use 342-X external power supply)

1.1.3 MULTIPLEXER OPERATION

Each asynchronous MUX port has two data lines (incoming and outgoing) and two control lines (the outgoing device control line, and the incoming device status line). The MUX transmitter receives the data from the computer, converts it from parallel to serial form, inserts start and stop bits, inserts a parity bit if enabled, and transmits the data at the rate selected for the port to a maximum rate of 9600 baud (19.2K baud optional).

The receiver portion reverses the process. Circuitry on the MUX checks each port once per 100 microseconds for an input request. When such a request is found, the MUX reads the control words for that port and carries out the indicated operations - including access of the automatic buffer or interrupt of the computer, depending upon the conditions.

1.2 SYSTEM ARCHITECTURE

The MIGHTY MUX Multiplexer interfaces from four to 128 I/O devices to a POINT 4 MARK 5/8 or Nova-type computer. The heart of the system is the POINT 4 310 board which contains all the common logic and the first eight interface ports. Each port contains the necessary control logic and buffering for full-duplex operation. The common logic contains a sequencer which interrogates the ports in turn, provides direct access to computer main memory via the DMA channel, and generates interrupts. Up to 120 more ports may be added by means of POINT 4 Data's 301 expansion boards. Each board contains 8, 16, or 24 ports. A full 128-port system requires one 310 and five 301 boards.

Figure 1-3 is a system block diagram of the MUX interface.

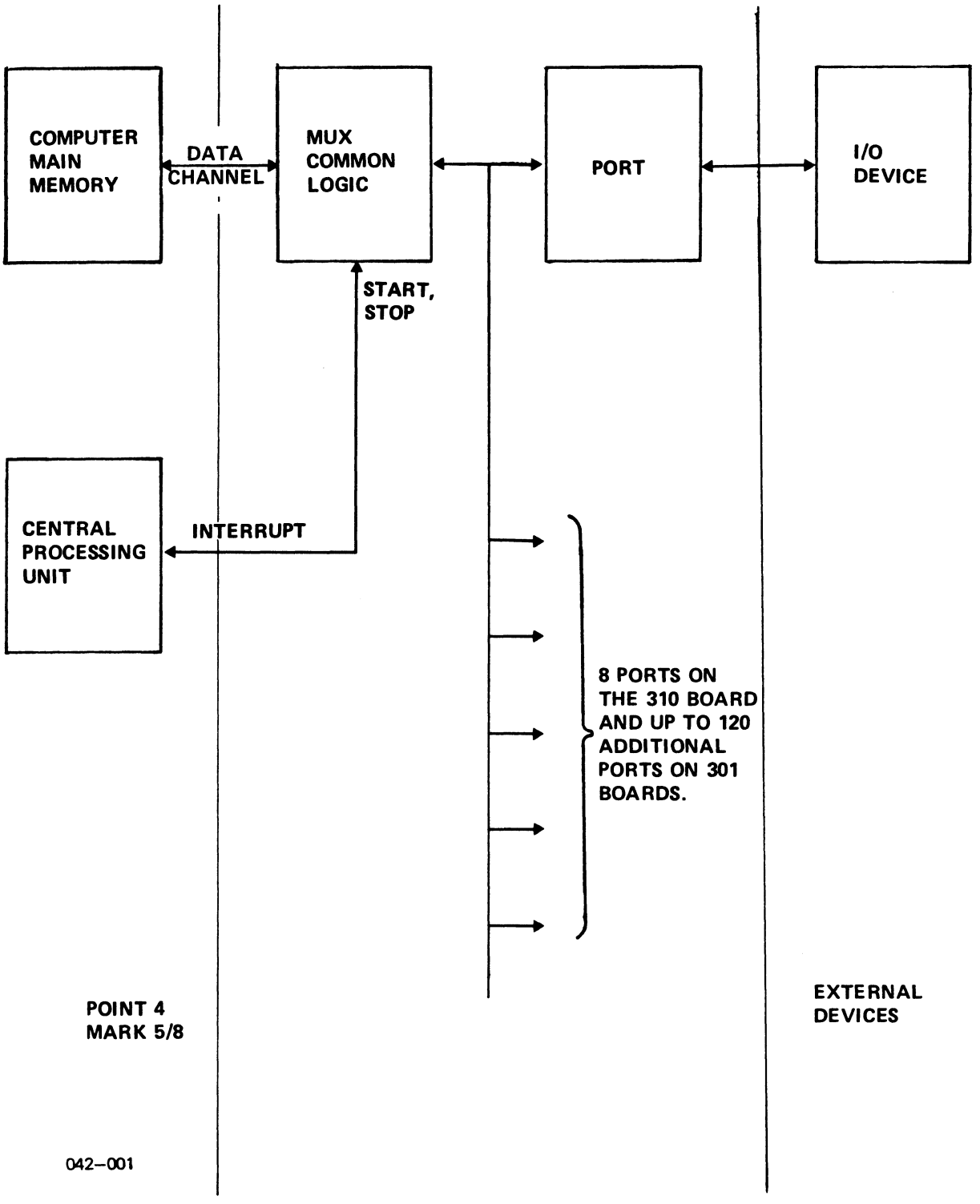


Figure 1-3. System Block Diagram of the MIGHTY MUX Interfacing External Devices to POINT 4 and Nova-type Computers

1.3 INPUT/OUTPUT INTERFACE

The MIGHTY MUX Multiplexer is designed to operate on POINT 4 MARK 5/8 and Nova-type computers which are compatible with specifications listed below under Computer Interface. MUX specifications are listed under Device Interface.

COMPUTER INTERFACE

I/O Bus	POINT 4 MARK 5/8 and Nova-type computer I/O bus-compatible
Backplane Wiring	None required for 310 boards; Z cable required for 301 boards
Device Code	Jumper selectable; standard = 25
Maximum DMA Transfer Rate	Data In: 900 nanoseconds/word Data Out: 1300 nanoseconds/word

DEVICE INTERFACE

Transmission Type	Asynchronous
Line Types	Full Duplex, Half Duplex
Line Speeds	Up to 9600 baud (19.2K baud optional)
Line Interfaces	Compatible with RS-232C/CCITT V.24 or current loop (20, 40 or 60 milliamps)
Number of Ports	128 maximum per MUX
Real Time Clock	100 Hz (period equals 10 milliseconds)
Modem Control	Supports Bell 103, 200 and 300 Data Sets or equivalent
Master Terminal Mode (Device Code 10/11)	Switch-selectable baud rate up to 9600 baud (19.2K baud optional)

1.4 SPECIFICATIONS

POINT 4 Data Corporation's MIGHTY MUX Multiplexer package includes: the MUX 310 board, junction panel, connector/cable assemblies, power supplies, and a diagnostic program. Physical, electrical and environmental specifications are listed below.

PHYSICAL

310, 301 boards:	Height: Width: 15 in. (38.10 cm) Depth: 15 in. (38.227 cm)
Junction Panel:	Height: 3-15/32 in. (8.81 cm) Width: 19 in. (48.26 cm) Depth: 2-1/2 in. (6.35 cm)
Power Supply:	Height: 3-3/8 in. (8.55 cm) Width: 19 in. (48.26 cm) Depth: 14-1/2 in. (36.83 cm)

POWER REQUIREMENTS (approximate)

Model	Current at +5VDC	+5 Power	Current at ±12VDC	±12 Power
310-A4	3.8A	19W	.1A	2.4W
310-A8	4.0A	20W	.2A	4.8W
301-A8	1.5A	8W	.2A	4.8W
301-A16	2.5A	13W	.4A	9.6W
301-A24	3.5A	18W	.6A	14.4W

OPERATING ENVIRONMENT

Operating Temperature:	0 to 50 degrees C (32 to 122 degrees F)
Relative Humidity:	0 to 90 percent noncondensing

340 POWER SUPPLY (REQUIRED FOR 301 BOARDS)

Input Voltage:	115V±10% 230V model available as an option
AC Current:	maximum .5A for 32 ports
Frequency:	47-440 Hz 60 Hz standard (output current based on this) For 50 Hz, derate output current by 10%
BTU Per Hour:	Less than 200

Section 2
PROGRAMMING

2.1 INTRODUCTION

This section outlines programming protocols for the interface of up to 128 peripheral devices via the MIGHTY MUX Multiplexer. Topics covered are: memory allocation requirements, I/O processing sequence, control word definitions, instructions for CPU control of MUX, and initialization procedures.

2.2 MEMORY ALLOCATION REQUIREMENTS

To control the MIGHTY MUX, the program stores appropriate I/O command words in a certain dedicated area in memory, consisting of a control block for each port used. Memory allocation of the MUX control block and of input and output buffers is shown in Figure 2-1. All control blocks are contiguous in main memory, and may start at any multiple of 400 (octal). This starting address is under software control. In the absence of appropriate software control, the MUX control-block starting address will default to a value preset by jumpers on the MUX board. The standard for this is 36000 (octal). All control blocks must be the same size. The standard control block size is 40 words (octal); it may be jumper modified to 20 or 10 (octal).

Within each port control block (PCB), six words are used for MUX control: 0, 1, 4, 5, 6, and 7. Even-numbered words control input; odd-numbered words control output. Each set of three words contains a general control word and two pointers indicating the beginning and end of the desired buffer areas in memory. The sizes and locations of these buffers are entirely under software control. Several ports may transmit from the same buffer area at one time, since each keeps track of its own pointers. If the port is operated in single-character mode, pointer words are not read by the MUX.

MIGHTY MUX control word definitions and format are discussed in Section 2.4.

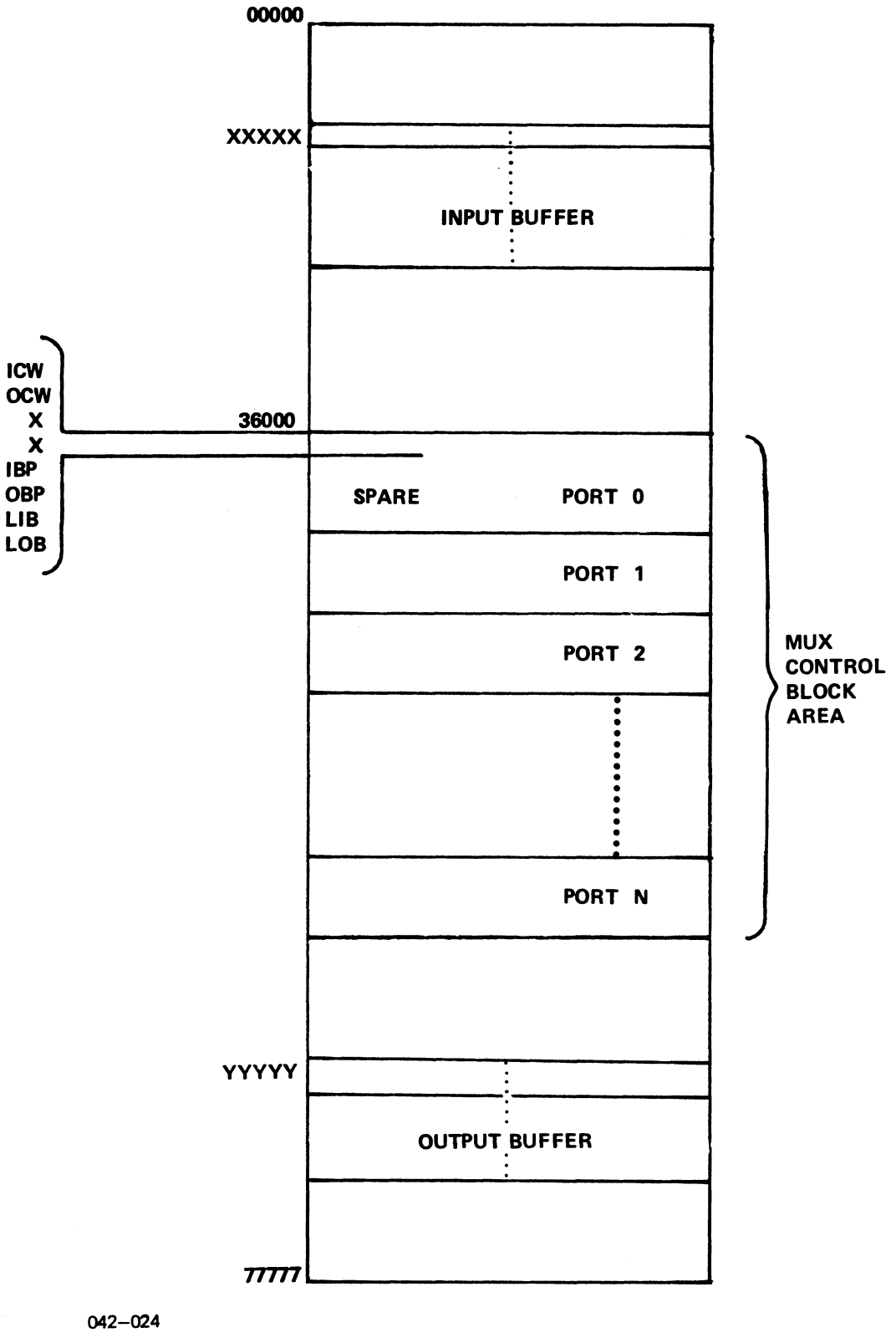


Figure 2-1. Memory Allocation for Control Block and Input and Output Buffers

2.3 INPUT/OUTPUT PROCESSING SEQUENCE

In order to start the MUX, it is necessary for the program to give a DOC ac,MUX instruction. The MUX then cyclically tests all ports, checking for an input request (a character has been received) or an output request (the port is ready to accept the next output character). When one of these requests is found, the MUX stops at that port and reads its input control word (ICW) or output control word (OCW). The MUX then carries out the indicated operations, including accessing the automatic buffer, if appropriate. When all required actions for the sensed input or output request are completed, the MUX tests each port in turn. If both input request and output request are true simultaneously, input takes precedence and output is deferred until that port is inspected again. See Figure 2-2 for an I/O processing flowchart.

When an input or output process is completed (see Section 2.4.1), the MUX sends an interrupt to the CPU. An interrupt may also be given by the real-time clock on the MIGHTY MUX once each 10 milliseconds. The program, by giving a DIAS ac,MUX instruction, can then clear the interrupt and simultaneously read the MUX status word, indicating which type of interrupt was given.

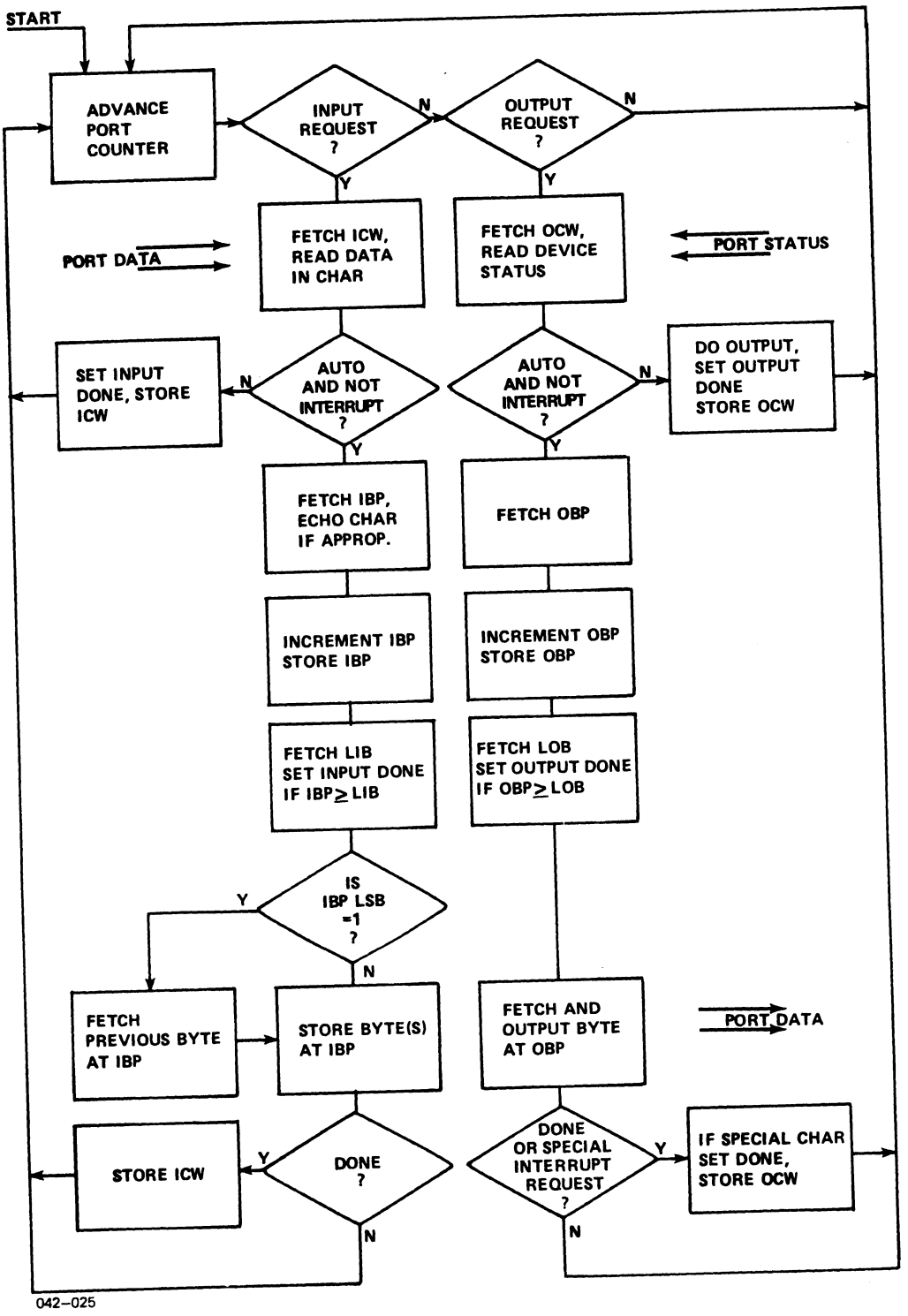


Figure 2-2. Input/Output Processing Flowchart

2.4 CONTROL WORD DEFINITIONS

Six words are used for MUX control within each port control block: 0, 1, 4, 5, 6, and 7. The three even-numbered words control the input; the three odd-numbered words control output. In each set of three words is a general control word, with two pointers indicating the beginning and end of desired buffer areas in memory. Formats for MUX control words are shown in Figure 2-3.

2.4.1 INPUT CONTROL WORD (ICW)

Table 2-1 defines the input control word (ICW), indicating each bit, its name and operation.

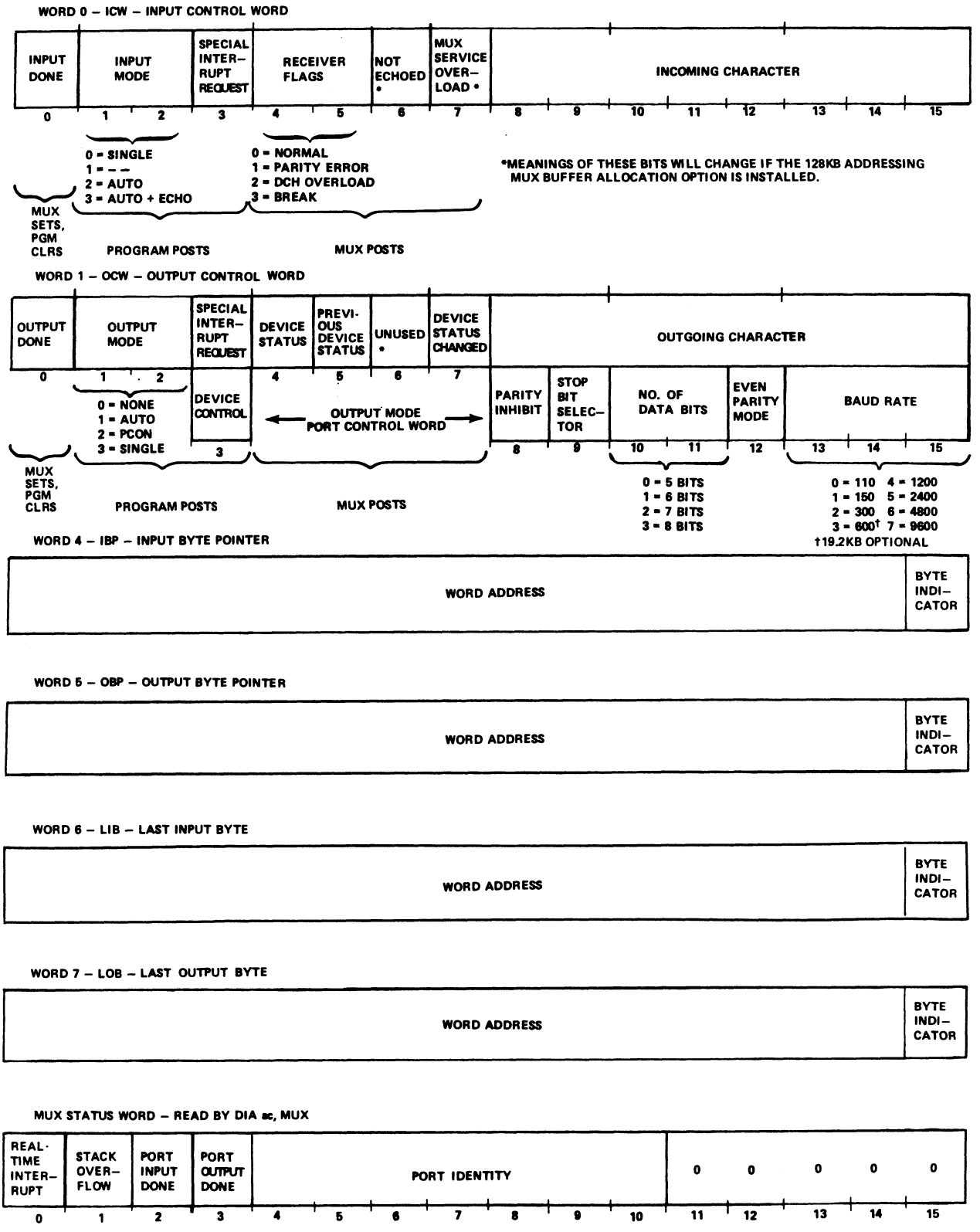


Figure 2-3. MIGHTY MUX Control Words

TABLE 2-1. INPUT CONTROL WORD DEFINITIONS
(WORD 0 OF EACH CONTROL BLOCK)

Bit	Name	Operation
0	Input Done	Set by MUX under the following conditions: • If MUX is in single-character-input mode when an incoming character is received and stored in bits 8-15. • If MUX is in automatic-input mode (with or without automatic echo) when the assigned input buffer is full; i.e., after an incoming character is placed in the byte specified by the last input byte (LIB). • If any of the following conditions is detected by the receiver (see bits 4 and 5 below): - Parity error - if parity inhibit is 0 (see OCW bit 8) - Data channel overload - Break (framing error) • If MUX is in automatic-input-with-echo mode, but automatic echo was not accomplished (see bit 6 below). • If special-interrupt request is enabled, and a special character is received (see bit 3 below). At the same time that input-done is set, an interrupt is generated. Input-done must be cleared by the MUX interrupt service program within one character time. If another input character is received while input-done is still set, it will override the character in bits 8-15 (automatic input is not permitted). Bit 7 is set, indicating that an incoming character has been lost, but a second interrupt is not generated. To inhibit any interrupts from a particular port, set input-done to 1.

TABLE 2-1. INPUT CONTROL WORD DEFINITIONS (Cont)
(WORD 0 OF EACH CONTROL BLOCK)

Bit	Name	Operation
1 & 2 Value	Input Mode	Written by the program, read by MUX to determine the type of input.
0 0		Single-character-input-mode - each incoming character is placed in bits 8-15 of ICW, input-done is set, and an interrupt given.
0 1		Not used (illegal).
1 0		Automatic-input - incoming characters are placed in the input buffer defined by input byte pointer (IBP) and last input byte (LIB), if no interrupt conditions are encountered. See bit 0 for interrupt conditions.
1 1		Automatic-input-with-echo - same as automatic input, except that each character placed in the input buffer is also automatically echoed (output). Note that any character producing an interrupt (other than buffer-full) is <u>not</u> automatically echoed.
Bit		
3	7-Bit ASCII Mode - Special Interrupt Request	Written by the program. If this bit is set, the MUX will examine each incoming character and generate an interrupt if it is a special character. The set of special characters is determined by a PROM (Programmable Read-Only Memory) on the 310 board and may be any set selected by the user. The standard set consists of all characters below octal 40 or above octal 173. Special characters are stored in ICW and produce interrupts. This allows immediate program response to such characters as backspace, carriage return and end-of-message. A second effect of the ASCII mode bit is that the MUX sets the MSB (most significant bit) of each incoming character to one. This allows the software operating system to distinguish incoming 7-bit ASCII data characters from nondata by means of the MSB.

TABLE 2-1. INPUT CONTROL WORD DEFINITIONS (Cont)
(WORD 0 OF EACH CONTROL BLOCK)

Bit	Name	Operation
4 & 5 Value	Receiver Flags	Posted by the MUX when an input is received.
0 0		Normal
0 1		Parity error - if parity is not inhibited (see OCW bit 8)
1 0		Data channel overload - an input character was received before the previous character from that port could be stored by the MUX. This occurs if another higher priority interface on the computer's data channel has been taking most of the computer memory cycles.
1 1		Break, or framing error - a stop bit was not received. This is used for detecting the break character, which is a zero (0) character with no stop bit. Each of the non-normal conditions produces an interrupt and sets input-done. If more than one of these occur at the same time, only the flag with the highest number is recorded.
Bit		
6	Not Echoed	Posted if automatic echo is not possible because output circuitry is busy with a previous output. This can only occur if the program started an output while the MUX was in automatic-input-with-echo mode. As this bit is posted, input-done is set and an interrupt generated. If MUX Buffer Allocation option is installed, bit 6 represents the most significant bit of input buffer word address. See also effect on bit 7.

(Table continues on next page.)

TABLE 2-1. INPUT CONTROL WORD DEFINITIONS (Cont)
(WORD 0 OF EACH CONTROL BLOCK)

Bit	Name	Operation
7	MUX Service Overload	Set by the MUX if an input character is stored when the input-done flag is still on from a previous input. The length of time available for the MUX to service an input interrupt is approximately one character-transmission time (1 millisecond for 9600 baud data rate). If MUX Buffer Allocation option is installed, this bit becomes a LOGICAL OR of Not-Echoed and MUX-service-overload.
8-15	Incoming Character	Each incoming character which produces an input-done is stored in these bits. In automatic-buffer mode, characters which are stored in the automatic buffer are not stored in ICW. However the last incoming character, which produces the buffer-full condition, is stored in both locations. Exception: If an input buffer ends in mid-word, the byte placed in ICW is the right half of the last buffer word, and not the last incoming character. The software must therefore check for buffer-full before it checks for special-character.

2.4.2 OUTPUT CONTROL WORD (OCW)

Table 2-2 defines the Output Control Word (OCW), indicating each bit, its name and operation.

TABLE 2-2. OUTPUT CONTROL WORD DEFINITIONS
(WORD 1 OF EACH CONTROL BLOCK)

Bit	Name	Operation
0	Output Done	Set by the MUX under the following conditions: • If MUX is in single-character or port-control-output mode when the output byte is read for transmission to the port. If output control words are reloaded within 1-2 character times, uninterrupted output will result. • If MUX is in automatic-buffer mode when the last byte of the automatic buffer is read for transmission. If the output control words are reloaded within 1-2 character times, uninterrupted output will result. • If MUX is in automatic-output-with-special-interrupt-request mode, and if the transmitted character is a special character, automatic output is terminated after transmitting the special character. • If the device-status-changed bit (bit 7) becomes a one (1), regardless of output mode. At the same time that output-done is set, an interrupt is generated. Output-done must be cleared by the program before another output can take place on that port. Leaving output-done set inhibits further interrupts from that port, even if the device status changes.

TABLE 2-2. OUTPUT CONTROL WORD DEFINITIONS (Cont)
(WORD 1 OF EACH CONTROL BLOCK)

Bit	Name	Operation
1 & 2 Value	Output Mode	Written by the program and read by MUX to determine the type of output.
0 0		No output.
0 1		Automatic-buffer output - MUX will output from the automatic buffer defined by output byte pointer (OBP) and last output byte (LOB).
1 0		Port-control-output - the data in bits 3 and 8-15 are sent to the port as control data for assigned control functions.
1 1		Single-character-output - the data byte in bits 8-15 is sent to the port for transmission to the external device.
Bit		
3	Special Interrupt Request	If output mode = 01 (auto): Written by the program; if this bit is set to one (1), the MUX will test each outgoing character to determine if it is a special character. If it is, the MUX will transmit it, set output-done, produce an interrupt, and terminate automatic output. This bit is useful in driving printers which require special service after a carriage return character, such as a line feed, delay time, or a print line command.
	Device Control	If output mode = 10 (port control output): Written by the program; transferred by the MUX to the port when the port-control-output is done. This bit is stored by a flip-flop in the port circuitry and applied as an EIA level on the device-control line to an external device. It may be used for any desired function. 1 = +10V = EIA Positive = function on 0 = -10V = EIA Negative = function off If output mode = 00 (none) or 11 (single) bit 3 has no effect.

TABLE 2-2. OUTPUT CONTROL WORD DEFINITIONS (Cont)
(WORD 1 OF EACH CONTROL BLOCK)

Bit	Name	Operation
4	Device Status	Written by the MUX each time OCW is inspected. This is the current value of the device-status line from the external device. This line may also be used for other desired functions; it does not interact with the port in any other way. 1 = EIA Positive, i.e. > +3V 0 = EIA Negative, i.e. ≤ 0V or open Note: Any voltage above +1.3 volts will produce a 1; any voltage below +0.7 volts will produce a 0; the result of voltages between these limits is indeterminate.
5	Previous Device Status	Written by the MUX each time OCW is inspected. The value read from Bit 4 is rewritten here. It is of no significance to the programmer.
6	Unused	Set to 0 by the MUX if MUX Buffer Allocation option is not installed. If MUX Buffer Allocation option is installed, bit 6 represents the most significant bit of output buffer word address.
7	Device Status Changed	Written by the MUX when the device-status line does not equal the device-status bit (bit 4). When this bit is set, bit 0 (output-done) is also set, and an interrupt is generated.

(Table continues on next page.)

TABLE 2-2. OUTPUT CONTROL WORD DEFINITIONS (Cont)
(WORD 1 OF EACH CONTROL BLOCK)

Bit	Name	Operation
8-15	Outgoing Character	If output mode is set to 11 (single character), this byte is sent to the port for transmission to the external device. If output mode is set to 00 or 01 (inactive or automatic), this byte is ignored. In automatic-output-with-special-interrupt request, this byte must not be a special character; otherwise the special character detection logic will not permit automatic output to start. When an output-done interrupt is produced, the MUX will store the character that produced the interrupt in this byte. If output mode is set to 10 (port control output), this byte is sent to the port as a control character, governing both output and input. In this case, the meanings of the eight bits are as follows:
8 Value 0 1	Parity Inhibit	 Parity is generated (transmit) and checked (receive). No parity
Bit 9 Value 0 1	Stop Bit Selector	 One stop bit. Two stop bits (output). For input, one stop bit is adequate.

TABLE 2-2. OUTPUT CONTROL WORD DEFINITIONS (Cont)
(WORD 1 OF EACH CONTROL BLOCK)

Bit	Name	Operation
10,11	Number of Data Bits	This field governs the data bits only; parity (if not inhibited) is an additional bit. If less than eight bits are selected, the character is right justified and the unused most significant bits are set to zero on input. For output, the character must be right justified, but the unused most significant bits are ignored.
Value		
0 0		5 bits
0 1		6 bits
1 0		7 bits
1 1		8 bits
Bit		
12	Parity Mode	In effect if bit 8 is set to 0.
Value		
0		Odd parity
1		Even parity
Bit		
13 14 15	Baud Rate	Frequency of transmission/reception
Value		
0 0 0		110 baud (standard Teletype)
0 0 1		150 baud
0 1 0		300 baud
0 1 1		600 baud (19.2K baud optional)
1 0 0		1200 baud
1 0 1		2400 baud
1 1 0		4800 baud
1 1 1		9600 baud

2.4.3 INPUT BYTE POINTER (IBP) - WORD 4 OF CONTROL BLOCK

IBP and LIB (last input byte) are only used in the automatic-input mode. IBP must be set by the program to one (1) less than the first byte address of the automatic input buffer. Each time the MUX stores an incoming byte, it will first increment IBP. The MUX uses the most significant 15 bits of the IBP as a memory word address, and the least significant bit (bit 15) as the byte indicator (BIN). The byte will be stored in the left half of the word addressed if BIN = 0; it will be stored in the right half if BIN = 1. IBP always points to the last input byte stored. If IBP $\geq$ LIB, an interrupt will be generated by the first incoming character, which will be stored at (IBP) +1.

2.4.4 OUTPUT BYTE POINTER (OBP) - WORD 5 OF CONTROL BLOCK

OBP and LOB (last output byte) are used only in the automatic-output mode. OBP must be set up by the program to one (1) less than the first byte address of the automatic-output buffer. Each time the MUX is ready for an output byte, it will increment OBP. The MUX fetches the byte for transmission from the appropriate half of the word address given by the most significant 15 bits of the OBP (word address) in conjunction with the byte indicator (BIN). OBP always points to the last byte transmitted.

2.4.5 LAST INPUT BYTE (LIB) - WORD 6 OF CONTROL BLOCK

Set up by the program to the last byte address of the auto-input buffer. The MUX will generate an input-done interrupt when a byte is stored at this address. IBP will then be equal to LIB.

2.4.6 LAST OUTPUT BYTE (LOB) - WORD 7 OF CONTROL BLOCK

Set by the program to the last byte address of the automatic-output buffer. When the byte at that address is picked up for transmission, the MUX will generate an output-done interrupt (OBP will then be equal to LOB). An automatic buffer may contain as little as one byte. In this case, initially (LOB) = (OBP) + 1. If initially (LOB) < (OBP) + 1, the MUX will transmit one byte from (OBP) + 1, set output-done and produce an interrupt.

2.4.7 STATUS WORD DEFINITIONS

The MUX status word is not one of the control words; it is the word which is read in when the program gives a DIA ac,MUX instruction. The MUX status word contains information indicating which type of interrupt was given (see Figure 2-4). The MUX contains a stack (FIFO - first-in, first-out) which can queue up to 40 interrupts. This stack is popped when a start pulse is given by the program. Therefore, by means of a DIAS ac,MUX instruction, the program can read the MUX status word and pop the stack. If the stack is not empty, the MUX done flag will remain set and another interrupt will be generated. If the stack is empty, a DIA ac,MUX instruction will read in a word with bits 0-3 all equal to zero (0); bits 4-10 (port identity bits) may not be zero (0). Table 2-3 defines the MUX status word, indicating each bit, its name and operation.

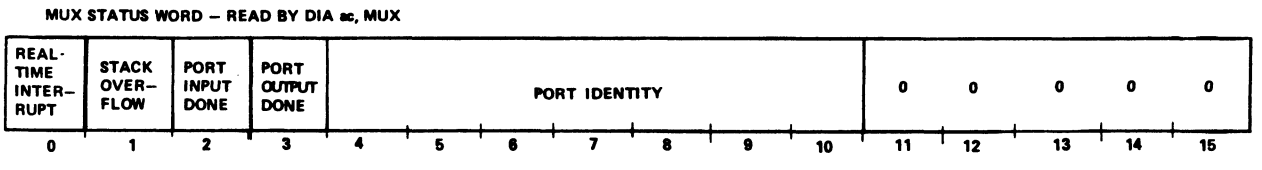


Figure 2-4. MUX Status Word

TABLE 2-3. MUX STATUS WORD DEFINITIONS

Bit	Name	Operation
0	RTI-100Hz Real-Time Interrupt	When the real-time clock is enabled, the MUX will generate an interrupt once each 10 milliseconds, with this bit set to one (1). For additional interrupts, this bit will be set to zero (0). Bit 0 is cleared by the DIA ac,MUX instruction. It is possible for a MUX status word to report both a real-time clock interrupt and an input or output interrupt simultaneously by having both bit 0 and bit 1, 2, or 3 set.
1	Stack Overflow	Set to one (1), this bit indicates that the MUX interrupt stack has overflowed ($\geq$ 40 interrupts pending). The software must inspect all ICWs and OCWs to determine those that require service. Set to zero (0), the following definitions apply:
2	Port Input Done	The port whose identity appears in bits 4-10 (below) has generated an input-done interrupt.
3	Port Output Done	The port whose identity appears in bits 4-10 (below) has generated an output-done interrupt.
4-10	Port Identity	This field identifies the port reporting an input-done or output-done interrupt. The identity of a port consists of seven bits which specify the 128 possible port control blocks: bits 4-10 of the port's control word address for 40-word (octal) control blocks, bits 5-11 for 20-word control blocks, or bits 6-12 for 10-word control blocks.
11-15		Always 0.

2.5 INSTRUCTIONS FOR CPU CONTROL OF MUX

The CPU controls MUX operation by means of the I/O instructions shown in Table 2-4.

TABLE 2-4. CPU CONTROL OF MUX

Instruction	Operation
IORST or CLEAR (e.g. NIOC)	Turns off both MUX and the real-time clock, clears done and busy flags, and terminates any output that may have been in process by setting all data output lines to -10 volts. IORST also sets: • all device-control lines to +10 volts • baud rate of Port 0 to hardware default baud rate (normally 9600 baud) • Port 0 to 8-bit character length, no parity, two stop bits • the port control block base address to its default value The Clear pulse does not have these additional effects. It allows the software to set up desired port parameters, shuts down the MUX and permits Port 0 to be used as the master terminal interface.
IOPLS (e.g. NIOP)	Starts the real-time clock. Has no effect on MUX action.
START (e.g. NIOS)	Pops the MUX status word stack. A DIAS ac,MUX instruction will read and pop the MUX status word stack. If the stack is empty, the MUX done flag will be cleared. If a stack overflow has occurred, the Start pulse clears the stack. **NOTE** CPUs such as the Nova 2 may allow a data channel cycle between the DIA and S pulses of a single DIAS instruction. To guard against this, the DIAS should be preceded by a DOB ac,MUX and followed by a DOC ac,MUX.

TABLE 2-4. CPU CONTROL OF MUX (Cont)

Instruction	Operation
DOA ac,MUX	This instruction sets up the base address of the PCB area if given before the MUX is turned on. Only the eight MSBs are used; therefore the PCB area must start at a multiple of 400 (octal). Once the MUX has been turned on, the effect of the DOA instruction (regardless of data output) is to prompt the MUX to inspect all OCWs for any pending output commands. In the absence of the DOA, the OCWs of all inactive ports are tested once each 50 milliseconds. This technique can be used to speed up output to peripherals such as line printers.
DOC ac,MUX	Starts the multiplexer proper. DOCP ac,MUX will start both the MUX and the real-time clock.

(Table continues on next page.)

TABLE 2-4. CPU CONTROL OF MUX (Cont)

<table>
<tr><th>Instruction</th><th>Operation</th></tr>
<tr><td>DOB ac,MUX</td><td>Stops all MUX data channel transfers, but does not interrupt any input or output already in process. Because the MUX takes successive data channel cycles when processing a particular port, the DOB instruction will always stop the MUX after completion of one port service and before beginning another. The DOB/DOC pair can be used when writing a new control word to prevent changes in the device status before the program stores a new OCW. The following sequence may be used in the MUX output routine to write a new OCW:

LDA 2,BASE Base address of port control block
LDA 1,C4000 Mask for bit 4, device status
INTDS Disable interrupts for minimum MUX pause
DOB 0,MUX Pause MUX operation
LDA 0,1,2 Pick up OCW
AND 0,1 Pick out the device-status bit
STA 1,1,2 Store idle OCW
DOC 0,MUX Continue MUX operation
INTEN Enable interrupts

A similar sequence may be used in the MUX input service routine to prevent an incoming character from being accidentally overwritten by the program.</td></tr>
<tr><td>DIA ac,MUX</td><td>Reads in the MUX status word and clears the real-time clock interrupt bit.

NOTE

To guard against occasional lost interrupts, the DIA should be preceded by a DOB ac,MUX and a 20-microsecond timeout.</td></tr>
</table>

2.6 INITIALIZATION PROCEDURES

MIGHTY MUX set-up procedures are as follows:

1. Give an IORST, or NIOC ac,MUX.

2. Give a DOA ac,MUX to change the memory location of the control block area from its hard-wired default value (usually 36000). (See Control Block Area Location Option, Section 3.7.7.)

3. Set up each OCW for the initial port-control output (see Table 2-2). For example, for a data-set ready for automatic answering at 110 baud with two stop bits and using 7-bit characters with even parity, OCW would be 50150 (see Figure 2-5).

4. Set up each ICW for the desired input mode (see Table 2-1). For single-character input, the ICW would be set to zero (0). If the MUX is to set the MSB to one (1), set the ICW to 10000.

5. Give a DOC ac,MUX. This starts the MUX, setting its busy flag. Busy remains set until an IORST, CLEAR, or DOB ac,MUX is given. Interrupts cause done to come on, but do not clear Busy. The MUX will now complete the indicated port-control outputs, setting the output-done bit and the MUX done flag for each port.

6. Enable interrupts. When the MUX-done flag is set, an interrupt is generated. By means of DIAS ac,MUX instructions, the interrupt service will pop MUX status words off the interrupt stack until all output-done bits are processed.

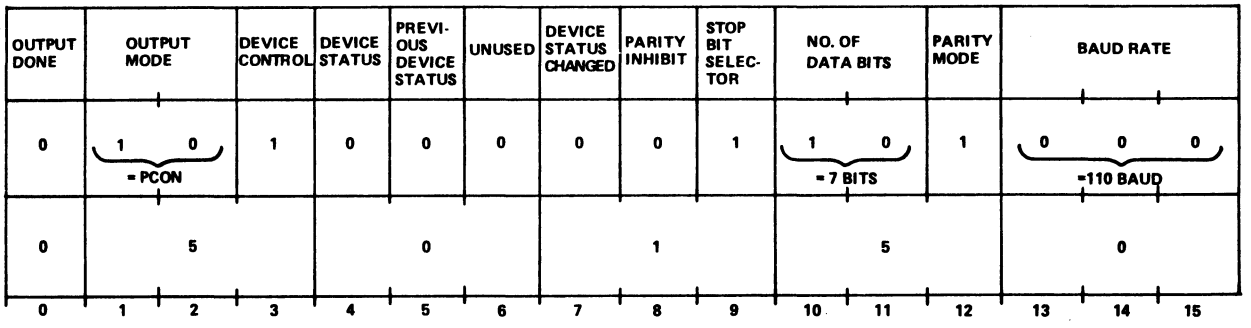


Figure 2-5. OCW Example

2.6.1 INPUT AND OUTPUT INITIALIZATION

The following subsections outline the initializing steps for input/output in single-character mode, automatic input/output to/from a string buffer, and changing port control parameters.

2.6.1.1 Input In Single-Character Mode

1. Set ICW to:

   a. 10000 for MSB = 1

   b. 0 for normal input (unmodified)

2. When an incoming character arrives, it will be stored in ICW, the input-done bit will be set, and an interrupt will be generated.

2.6.1.2 Output In Single-Character Mode

1. Set OCW to 6x000, plus the desired character, where x = 4 or 0, dependent upon the setting of the device-status bit (1 or 0).

2. An interrupt will be generated and output-done set when the character is picked up for transmission.

2.6.1.3 Input Automatically Into a String Buffer

1. Set input byte pointer (IBP) to one (1) less than the byte address of the first byte in the string.

NOTE

The byte address equals the word address shifted one place to the left with the least significant bit = 0 for the left byte or = 1 for the right byte.

2. Set last input byte (LIB) to the byte address of the last byte of the string buffer.

NOTE

The string buffer may end in the middle of a word without destroying the contents of the right half of that word.

3. Set ICW to:

40000 for no echo and no ASCII mode
50000 for no echo and ASCII mode
60000 for automatic echo and no ASCII mode
70000 for automatic echo and ASCII mode

4. An interrupt will be given and Input Done set if:

Condition	Indication
Buffer full	IBP = LIB
Parity error	ICW bits 4,5 = 0 1
Data channel late	ICW bits 4,5 = 1 0
Break	ICW bits 4,5 = 1 1
Auto echo ordered but not accomplished	ICW bit 6 = 1
Special character if ASCII mode set (special interrupt request)	Character is in ICW bits 8-15

NOTE

If the buffer is full, input-done cleared, ICW set up for automatic input, and another character is received, an interrupt will be produced. This character will produce buffer overflow, and will be stored in the next byte address.

2.6.1.4 Output Automatically From a String Buffer

1. Set output byte pointer (OBP) to one (1) less than the byte address of the first byte to be output.

2. Set last output byte (LOB) to the byte address of the last byte to be output.

3. Set OCW to:

   2x000 for regular automatic output

   3xyyy for interrupts on special control characters, where

   x = 4 or 0, dependent upon the setting of the device-status bit (1 or 0).

   yyy = any value that is not a special character

NOTE

If x is set to the wrong value, an immediate interrupt will be produced.

4. An interrupt will be produced and output-done set if:

Condition	Indication
Buffer empty	OBP = LOB
Device status changed	OCW bit 7 = 1 (OBP points to last character transmitted)
Special character if special interrupt requested	Character is in OCW bits 8-15 (OBP points to it in buffer)

2.6.1.5 Change Port Control Parameters

1. A new port-control mode may be set up at any time by storing the desired port control word in OCW. Such modes as baud rate, parity mode, or device control may be changed.

2. If a new port control word is to be issued after completing a transmission, it is necessary to add a null character at the end of the desired output string. The program must then wait for the output-done condition before it stores the new port control word in OCW. This is due to double-buffering in the port logic, and applies to both single-character mode and automatic-output mode.

2.6.2 INTERRUPT SERVICE

1. Give a DIAS ac,MUX. This reads in the MUX status word, and clears the MUX-done flag, unless there is another interrupt pending.

2. Test MUX status word bit 0. If it is set to one (1), perform real-time interrupt service. In either case, proceed to step 3.

3. Test Bit 1. If it is set to one (1), check all ICWs and OCWs to determine if service is required, then proceed to step 6. If bit 1 is set to zero (0), proceed to step 4.

4. Test Bit 2. If it is set to one (1), perform input service for the port whose identity appears in bits 4-10, then proceed to step 6.

5. Test Bit 3. If it is set to one (1), perform output service for the port whose identity appears in bits 4-10.

6. Give another DIAS ac,MUX. If bits 0-3 are all set to zero (0), exit interrupt service; otherwise, return to step 2.

2.7 MASTER TERMINAL MODE INSTRUCTIONS/PROCEDURES

The Master Terminal Mode permits Port 0 to be used as the master terminal (device codes 10 and 11) or as a time-shared port.

IORST enables the master terminal mode. Port 0 is the master terminal port. The initial port parameters are:

Baud rate = (selected by baud rate switch-see Section 3.3.1.1)
Character length = 8 bits
Parity = none
Number of stop bits = 2

Standard device code 10/11 software interface applies; there is no MUX action on ports one through seven.

2.7.1 INPUT

When a character is assembled, done is set and an interrupt produced. If busy was on, it is turned off. DIA ac,TTI reads in the character. The S or C pulse clears done: S sets busy and C clears busy. Paper tape reader control (RS-232) is driven from busy.

2.7.2 OUTPUT

DOA ac,TTO transfers the outgoing character to the MUX. The S pulse starts the transmission of the character, sets busy and clears done. When transmission is completed, busy is cleared and done set, producing an interrupt. The C pulse clears both busy and done.

NOTE

The Master Terminal Mode logic has the following anomaly. If a done condition exists in both TTI and TTO, but TTI is masked in while TTO is masked out, the response to an INTA instruction will be TTO rather than TTI. To avoid problems from this anomaly, always mask out TTI whenever TTO is masked out.

2.7.3 MASTER TERMINAL MODE-TO-MUX MODE

DOC ac,MUX turns off master terminal mode and turns on MUX mode, starting standard data channel MUX action. MUX mode immediately clears TTI, TTO busy and done flags, but allows any outgoing character to be completed.

2.7.4 MUX-TO-MASTER TERMINAL MODE

A C pulse to the MUX (NIOC MUX) turns off MUX mode and turns on master terminal mode without changing port control parameters. The software can go to MUX mode, set Port 0 to a different baud rate, revert to master terminal mode and use master terminal mode I/O interface logic at any desired baud rate.

2.7.5 REAL-TIME CLOCK

The real-time clock (RTC) on the MUX may be used in master terminal mode by giving an NIOP MUX. The MUX busy and done flags, interrupt logic, and MUX status word all operate normally for RTC only. No data channel action will occur, and ports one through seven will remain inactive. If the MUX board is not in a POINT 4 chassis, check to ensure that there is not another RTC.

Section 3
HARDWARE INTERFACE

3.1 INTRODUCTION

This section covers hardware characteristics essential to the installation and programming of the MIGHTY MUX Multiplexer. The following key topics are covered:

- Multiplexer Architecture
- Multiplexer Installation
- Multiplexer System Configuration
- Input/Output Bus Interface Signals
- Junction Panel Connections
- Hardware-Selectable Options
- Multiplexer Timing

3.2 MIGHTY MUX ARCHITECTURE

The MIGHTY MUX Multiplexer serves as an interface between the processor and up to 128 peripheral devices.

Interface to the processor is via the CPU data bus. MUX logic to handle information passed between the processor and the multiplexer includes:

- Data
- Data Channel Control
- Status Information
- Programmed I/O Control
- Interrupt Control Logic
- Address Logic

In addition to logic designed to process the information noted above, strictly internal logic functions handle:

- Port Counter
- MUX Control Logic
- Frequency Generator
- MUX/Port Interface

Interface to peripherals is via the port common logic.

Figure 3-1 is a block diagram of MUX logic.

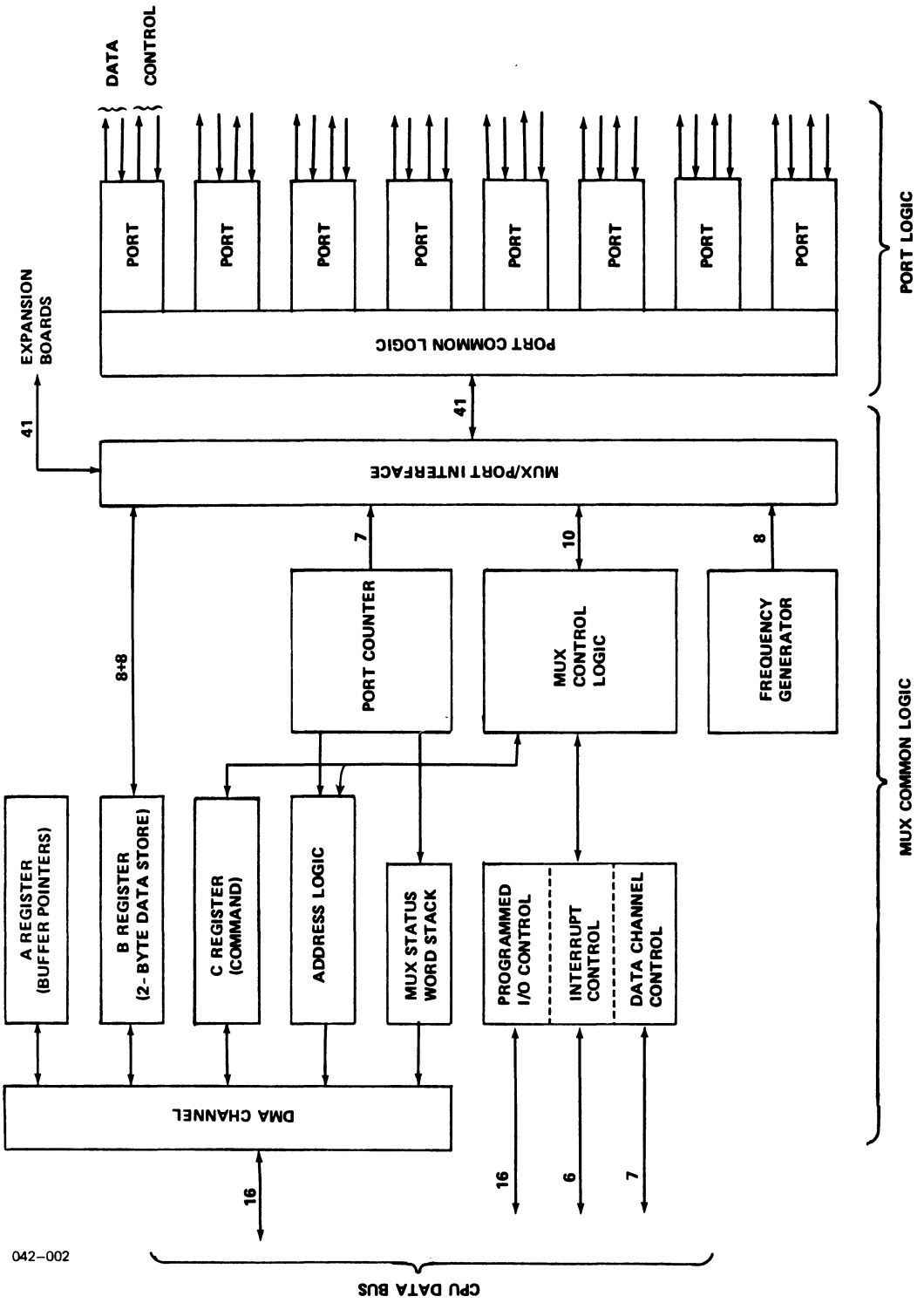


Figure 3-1. MIGHTY MUX Logic Block Diagram

3.3 MULTIPLEXER INSTALLATION

3.3.1 INSTALLATION PROCEDURES

Installation procedures for the MIGHTY MUX are described below, and illustrated in Figure 3-2.

The MIGHTY MUX Model 310-A4 or 310-A8 Multiplexer is a single-board design occupying one slot in the processor chassis.

Refer to the configuration guide in Section 3.4 for power source options and required associated equipment.

3.3.1.1 Default Baud Rate

Make sure that the rotary switch is set for the proper Port 0 default baud rate. When shipped, it is set at 9600 baud. The rotary switch can set rates from 110 to 9600. Rotary switch settings are as follows:

Setting	Baud Rate
0	110
1	150
2	300
3	600
4	1200
5	2400
6	4800
7	9600
8,9	Not Used

3.3.1.2 Chassis Position

With computer power off, insert the 310 MUX into any slot in the computer chassis. Make sure the MUX slot has a lower data channel priority than the disc controller slot.

3.3.1.3 Cabling

Mount the Junction Panel (Model 320) and Connector/Cable Assembly (Model 322) in a convenient location. Attach the free end of the 50-conductor ribbon cable to the connector on the 310 board. Pin 1 is to the right of the connector. The only effect of reversing the cable is that it will reverse the numbering of the eight connectors on the junction panel. If the External Power Source (Model 340 or 342) is to be used, the distribution cables should be connected to the back of the Connector/Cable Assembly (Model 322).

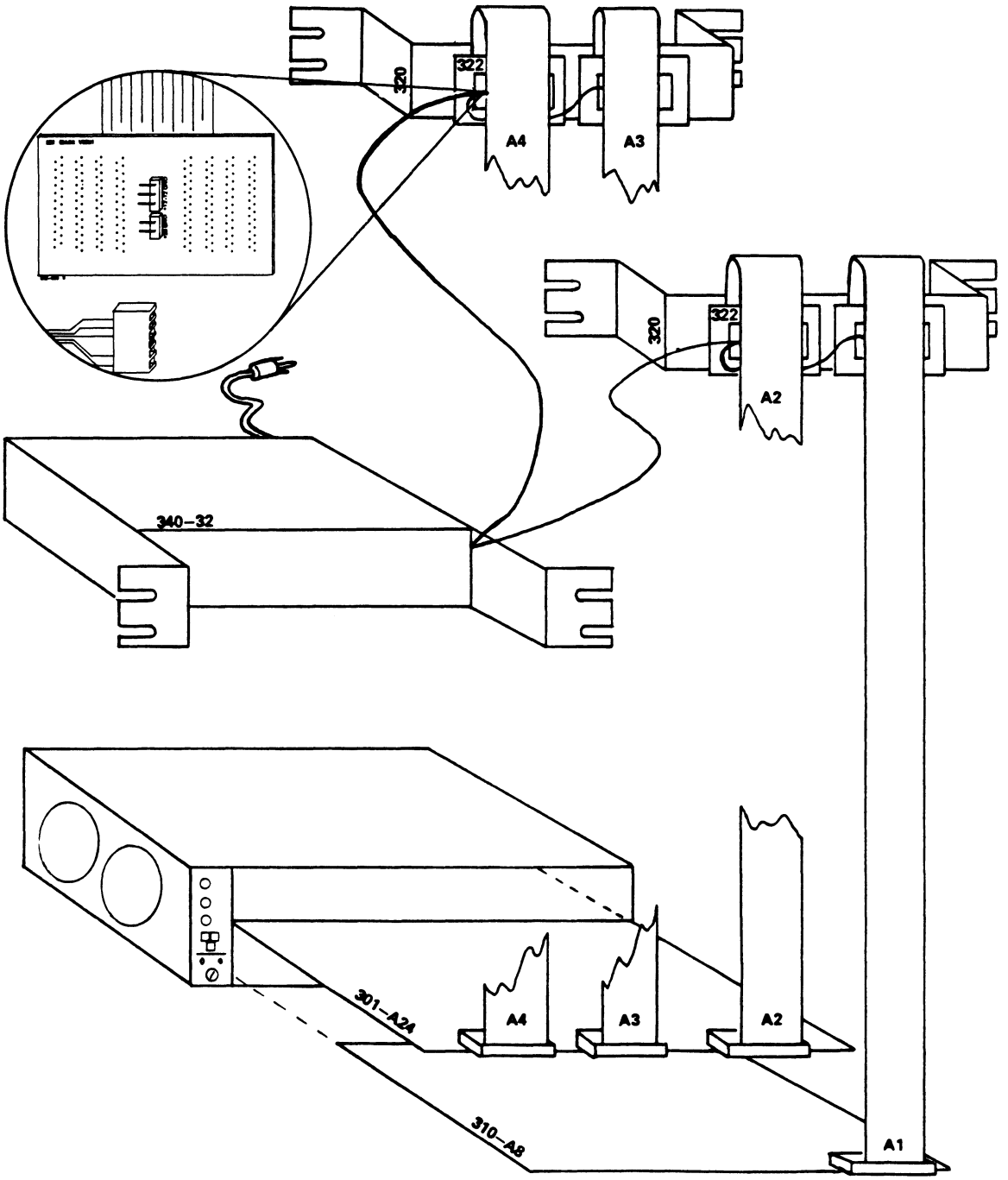


Figure 3-2. Multiplexer Installation

3.3.1.4 Priority Jumpers

For a computer other than a POINT 4 MARK 5/8, the following procedure is necessary. If there are any blank slots between the CPU board and the 310 board, the interrupt-priority and data-channel-priority signals must be jumpered to the 310 board on the backplane.

3.3.1.5 Data Channel Speed

Non-POINT 4 computers may have to be jumpered to provide standard-speed data channel on the MUX slot. The high-speed data channel may be used on POINT 4 computers.

3.3.1.6 Board Switches

If any 301 expansion boards are to be used, set the location of port control blocks to the desired control-block locations, using the DIP switches on the 301 expansion boards. The location of port control blocks is detailed in Section 3.7.8.

3.3.1.7 301 Board Z Cable

The 301 boards must be connected to the 310 (on the computer's backplane) by connecting 41 points on the 310 slot to 41 corresponding points on any slot containing a 301. This many be done by installing jumper wires (solder or wire-wrap), or by using the POINT 4 Model 324 Backplane Connector Cable (see Figure 3-3). The 41 connection points are listed below:

A47	A67	A78	A91	B31	B51
A49	A69	A79	B6	B34	B52
A57	A71	A81	B13	B36	B53
A59	A73	A83	B15	B38	B54
A61	A75	A85	B19	B40	B67
A63	A76	A87	B23	B48	B69
A65	A77	A89	B25	B49	

The MIGHTY MUX is now ready for use.

It is recommended that the MIGHTY MUX Diagnostic Program furnished with the MUX be run before the MUX is used on the system.

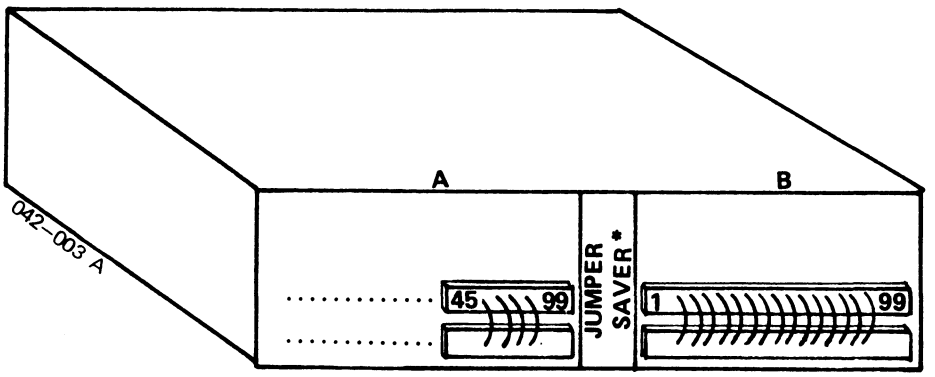


Figure 3-3. Backplane Connections for 301 Board

3.3.2 ADDITIONAL CONSIDERATIONS

3.3.2.1 RTC

If the MUX board is to be used with other boards that have a real-time clock, disable all but one clock.

3.3.2.2 Device 10/11

If the MUX board is to be used with other serial I/O boards (e.g., DG I/O or similar boards), make sure that only one board has device code 10/11 active.

3.3.2.3 Power

If only a 310 board is used (no 301 boards), power may be drawn from the backplane if all the following apply:

a. Voltages are available -
   +15V at B84 on backplane
   -15V at B93 on backplane

b. The On-Board Regulator option is installed
c. RS-232C interface voltages are used
   (Current Loop requires External Supply 342-xx)

3.3.2.4 Current Loop

If current-loop is desired, 322-CL Connector/Cables and 342-xx Power Supply must be used.

If the 322-CL Connector/Cable is used, current-loop data-out is on pin 25 and current-loop data-in is on pin 18.

Note that an RS-232C device can be used with current loop conector cable assembly (322-CL) without modifying the RS-232C cable. See Figure 3-4 for 322-CL Connector/Cable Wiring.

MUX current-loop interface is designed for terminals that are passive (no voltage source) on both input and output.

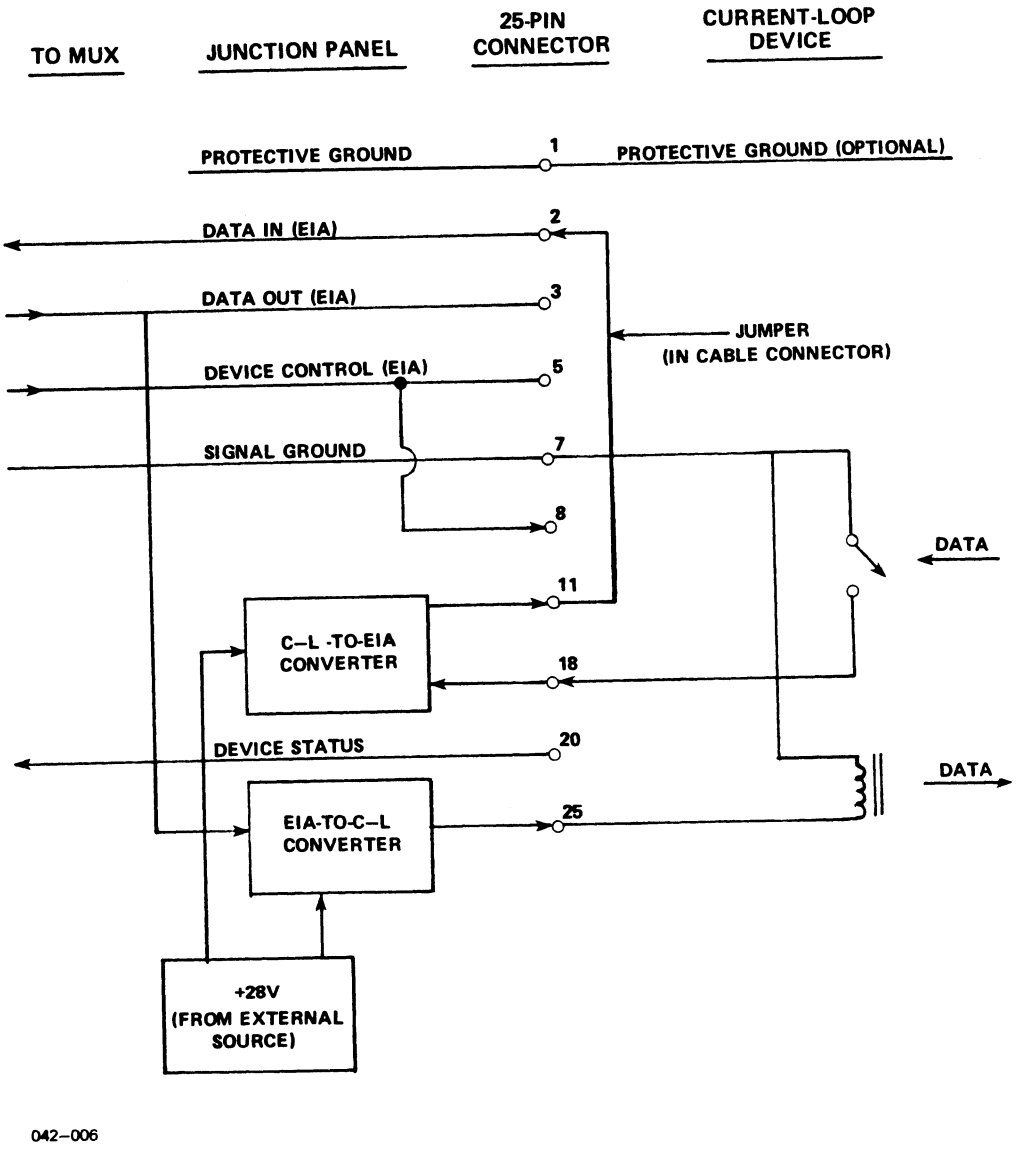


Figure 3-4. Junction Panel Connections for Current-Loop Interfaces

3.4 MULTIPLEXER SYSTEM CONFIGURATION

The MIGHTY MUX Multiplexer features flexibility in system configuration, determined by the number of ports required for asynchronous transmission. See the MUX configuration guide, Figure 3-5.

Using the configuration guide, a 32-port asynchronous multiplexer would include: one 310-A8; one 301-A24; two 320 panels; four 322 connector/cable assemblies; a 324-1 interboard (Z) cable; and a 340-32 power supply.

NO. OF PORTS REQ'D	BASIC MUX		ASYNCHRONOUS EXPANSION			PANEL	CONN/ CABLE ASSY	Z CABLE	POWER SUPPLY
	MODEL 310		MODEL 301			MODEL 320	MODEL 322	MODEL 324-n	
	-A4	-A8	-A8	-A16	-A24				
4	1					1	1		340-32 or 310 P/S
8		1				1	1		340-32 or 310 P/S
16		1	1			1	2	324-1	340-32
24		1		1		2	3	324-1	340-32
32		1			1	2	4	324-1	340-32
64		1	1		2	4	8	324-3	340-64
128		1			5	8	16	324-5	340-128

Figure 3-5. MUX Configuration Guide

3.5 INPUT/OUTPUT BUS INTERFACE SIGNALS

Input/output bus interface signals connect the processor logic to MUX logic. POINT 4 I/O bus and control signals can be classified into four major categories: data bus, data channel control, programmed I/O control, and interrupt control. Data bus and data channel control signals use the DMA channel. Figure 3-6 is a diagram of I/O signals across the I/O bus.

3.5.1 INPUT/OUTPUT INTERFACE SIGNAL CLASSIFICATION

Signals on the input/output bus can be grouped into the following signal classifications:

1. Bidirectional Data Bus (16 lines) - Used for transfer of all data and address words between the CPU and a peripheral device, for both programmed I/O and data channel transfers.

2. Device Codes (6 lines) - Codes used to designate the peripheral device involved in an input/output instruction.

3. Programmed Transfer Signals (6 lines) - These signals, generated by the CPU in response to input/output instructions for data transfers, are used to control data transfers for programmed input/output devices.

4. Device Control Signals (4 lines) - These signals are generated by the CPU in response to input/output instructions, and are used to initialize and control I/O devices. The signals affect only the device identified by its code in the instruction. The only exception is the IORST instruction, which resets all devices.

5. Skip Testing Flags (2 lines) - Flags supplied to the CPU, when skip-testing is required.

6. Interrupt Control Signals (6 lines) - Signals used to initialize and control the interrupt sequence.

7. Data Channel Transfer Signals (7 lines) - Signals used to control data channel transfers between memory and a peripheral device.

Table 3-1 identifies each signal by signal classification, indicates the signal name, the direction, and defines each signal function.

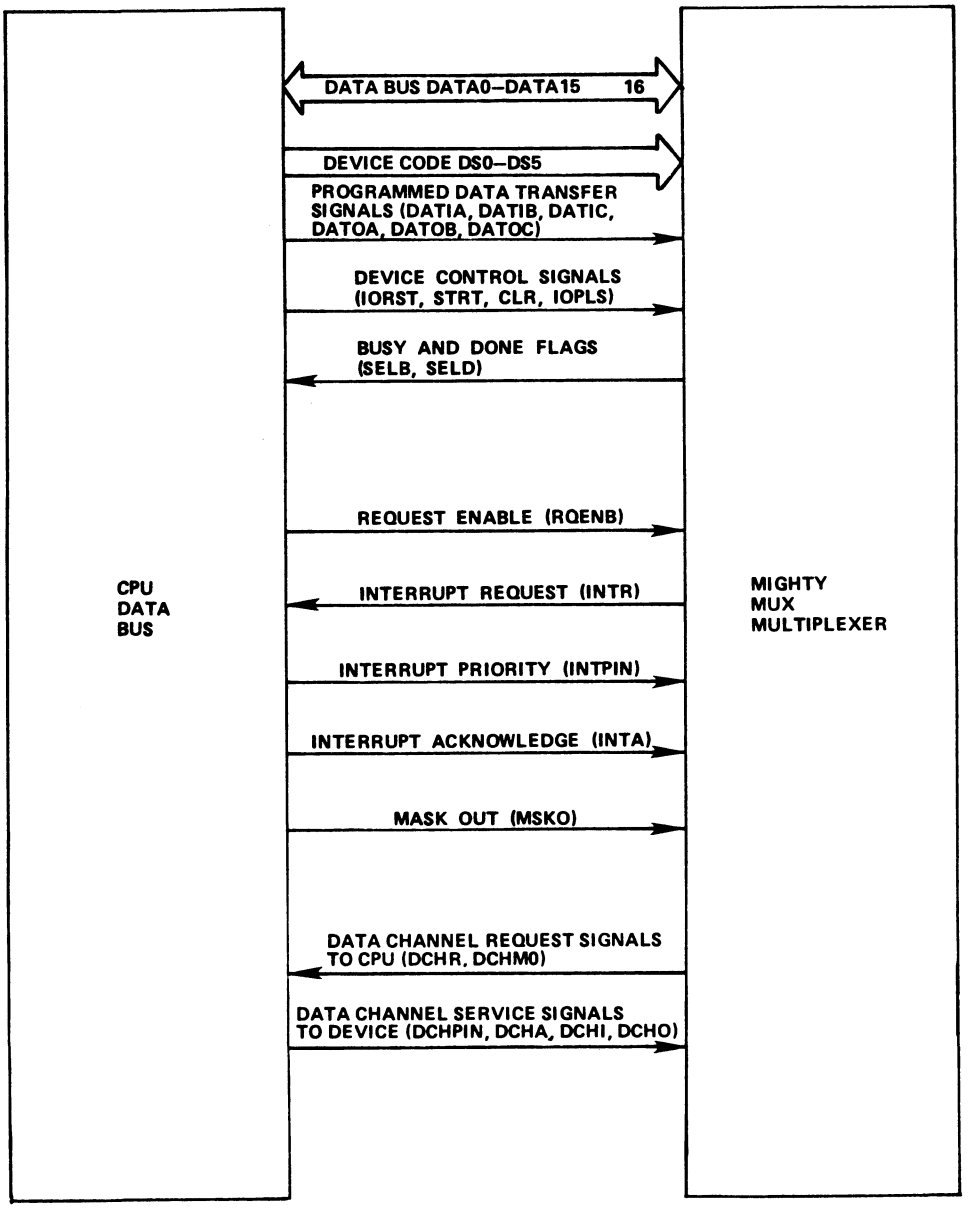


Figure 3-6. Input/Output Signals

TABLE 3-1. INPUT/OUTPUT SIGNALS BY CLASSIFICATION

Signal Group	Signal Name*	Direction	Description
Data Bus	DATA0- to DATA15-	Bidirectional	All data and addresses are supplied to and from the device via these lines. DATA0- is the MSB.
Device Code	DS0- to DS5-	From CPU	The CPU places the device code (bits 10-15 of the instruction word) on these lines during the execution of an input/output instruction. DS0- is the MSB.
Programmed Data Transfer Signals	DATIA+	From CPU	Data In A. Generated by a DIA instruction. Reads in Port 0 data (master terminal mode, device code 10), or MUX status word (MUX mode, device code 25).
	DATOA+	From CPU	Data Out A. Generated by a DOA instruction. Outputs Port 0 data (master terminal mode, device code 11). Outputs control area base address if not MUX mode (device code 25).
	DATIB+	From CPU	Not Used.
	DATOB+	From CPU	Data Out B. Generated by a DOB instruction. Causes a pause in MUX DMA transfer operation, regardless of data transferred.
	DATIC+	From CPU	Not Used.
	DATOC+	From CPU	Data Out C. Generated by a DOC instruction. Starts or resumes MUX DMA operation. The data transfer is ignored.

*Signal names ending with "+" are active high; those ending with "-" are active low.

TABLE 3-1. INPUT/OUTPUT SIGNALS BY CLASSIFICATION (Cont)

Signal Group	Signal Name*	Direction	Description
Device Control Signals	IORST+	From CPU	Input/Output Reset. Generated when an IORST instruction is being executed, and during power turn-on.
	STRT+	From CPU	Start. Generated when the CTRL field of an input/output transfer instruction contains code 01. It clears the done flag and interrupt request, and sets the busy flag. Starts output (master terminal mode, device code 11). Pops interrupt FIFO (MUX, device code 25).
	CLR+	From CPU	Clear. Generated when the CTRL field of an input/output transfer instruction contains code 10. It clears the busy and done flags and the interrupt request. The MUX resets to master terminal mode.
	IOPLS+	From CPU	I/O Pulse. Generated when the CTRL field of an input/output transfer instruction contains code 11. Starts real-time clock (device code 25).
Skip Testing Flags	SELB-	From MUX	Selected Device Busy. Indicates that the MUX is in DMA mode.
	SELD-	From MUX	Selected Device Done. Indicates interrupt pending.

*Signal names ending with "+" are active high; those ending with "-" are active low.

TABLE 3-1. INPUT/OUTPUT SIGNALS BY CLASSIFICATION (Cont)

Signal Group	Signal Name*	Direction	Description
Interrupt Control Signal	RQENB-	From CPU	Request Enable. Generated during each memory read/write cycle to synchronize INTR- and DCHR-.
	INTR-	From MUX	Interrupt Request. This signal goes low (following the leading edge of RQENB-) if MUX wants to request an interrupt.
	INTPIN-		Interrupt Priority Input. Produced by priority chain, or by Jumper-Saver on POINT 4 backplanes. This input is required for the MUX to respond to an interrupt-acknowledge. If a device of higher priority is requesting an interrupt, this signal will be false (high) into the MUX.
	INTPOUT-	From MUX	Interrupt Priority Output. Generated by MUX to the next board in the chain if the MUX is not requesting an interrupt and has INTPIN.
	INTA+	From CPU	Interrupt Acknowledge. Generated by an INTA instruction. Causes the device whose INTPIN- line is low and is requesting an interrupt to place its device code in bits 10-15 of the data bus for entry into accumulator specified in the instruction.
	MSKO-	From CPU	Mask Out. Generated by a MSKO instruction. Commands all I/O devices to set their interrupt-disable flags according to the state of the associated mask bit in the word on the data bus.

*Signal names ending with "+" are active high; those ending with "-" are active low.

TABLE 3-1. INPUT/OUTPUT SIGNALS BY CLASSIFICATION (Cont)

Signal Group	Signal Name*	Direction	Description
Data Channel Transfer Signals	DCHR-	From MUX	Data Channel Request. This signal goes low (following the leading edge of RQENB-) if MUX wants to request a data channel transfer.
	DCHPIN-		Data Channel Priority Input. Produced by priority chain, or by Jumper-Saver on POINT 4 backplanes. This input is required for the MUX to make a data-channel request. If a device of higher priority is requesting the data channel, this signal will be false (high) into the MUX.
	DCHPOUT-	From MUX	Data Channel Priority Output. Generated by MUX to the next board in the chain if the MUX is not requesting the data channel and has DCHPIN.
	DCHA-	From CPU	Data Channel Acknowledge. Generated by CPU in response to a data-channel request. Defines when MUX is to place memory address on data bus.
	DCHM0-	From MUX	Data Channel Mode. Indicates the type of data channel cycle being requested: 0 (high) Data Out (from CPU) 1 (low) Data In (to CPU)
	DCHI+	From CPU	Data Channel In. Defines when MUX is to place its input data on the data bus.
	DCHO+	From CPU	Data Channel Out. Defines when CPU has placed output data on the data bus.

*Signal names ending with "+" are active high; those ending with "-" are active low.

3.5.2 BACKPLANE PIN SIGNAL CONNECTION

All signal connections between the processor and the multiplexer take place via two 100-pin backplane connectors. Figure 3-7 shows the connector-pin layout for all I/O signals. The labeled pins refer to the I/O control signals, data transfer signals and the power lines used by the MUX.

BOTTOM		A		TOP
GND	2	1		GND
+5V	4	3		+5V
	6	5		
	8	7		
	10	9		
	12	11		
	14	13		
	16	15		
	18	17		
	20	19		
	22	21		
	24	23		
	26	25		
	28	27		
	30	29		
	32	31		
GND	34	33		GND
	36	35		
MSKO–	38	37		
INTA+	40	39		
DATIB+	42	41		
DATIA+	44	43		
DS3–	46	45		
DATOC+	48	47	*	
CLR+	50	49	*	
STRT+	52	51		
DATIC+	54	53		
DATOB+	56	55		
DATOA+	58	57	*	
DCHA–	60	59	*	
DS4–	62	61	*	
DS5–	64	63	*	
DS2–	66	65	*	
DS1–	68	67	*	
IORST+	70	69	*	
DSO–	72	71	*	
IOPLS+	74	73	*	
	* 76	75	*	
	* 78	77	*	
SELD–	80	79	*	
SELB–	82	81	*	
	84	83	*	
	86	85	*	
	88	87	*	
	90	89	*	
	92	91	*	
DCHPIN–	94	93		DCHPOUT–
INTPIN–	96	95		INTPOUT–
+5V	98	97		+5V
GND	100	99		GND

BOTTOM		B		TOP
GND	2	1		GND
+5	4	3		+5V
	* 6	5		
	8	7		
	10	9		
	12	11		
	14	13	*	
	16	15	*	
	18	17		DCHM0–
	20	19	*	
	22	21		
	24	23	*	
	26	25	*	
	28	27		
	30	29		INTR–
	32	31	*	
	* 34	33		DCHO+
	* 36	35		DCHR–
	* 38	37		DCHI+
	* 40	39		
	42	41		RQENB+
	44	43		
	46	45		
	* 48	47		
GND	50	49	*	
	* 52	51	*	
	* 54	53	*	
DATA14–	56	55		DATA7–
DATA11–	58	57		DATA5–
DATA8–	60	59		DATA12–
DATA0–	62	61		DATA4–
DATA13–	64	63		DATA9–
DATA15–	66	65		DATA1–
	68	67	*	
	70	69	*	
	72	71		
	74	73		DATA3–
	76	75		DATA10–
	78	77		
	80	79		
DATA2–	82	81		
+15V†	84	83		
	86	85		
	88	87		
	90	89		
GND	92	91		
	94	93		–15V†
	96	95		DATA6–
+5V	98	97		+5V
GND	100	99		GND

* 324 CABLE PINS – THESE PINS CONNECT THE 310 TO THE 301 BOARD(S)

†USED ONLY WITH ON-BOARD POWER OPTION

Figure 3-7. Backplane I/O Signals

3.6 JUNCTION PANEL CONNECTIONS

The Electronic Industries Association's (EIA) recommended standard RS-232C defines the interface between a data set (e.g., modem) and a data terminal (e.g., CRT). The standard delineates the electrical signal characteristics which have been accepted by equipment designers, and is the interface between equipment and most data sets supplied by common carriers.

The MIGHTY MUX can communicate with both data sets and data terminals. However, the 25-pin connectors on the junction panel are wired to make the MUX look like a data set. Therefore, data terminals may be plugged into it directly, but data sets must be connected through a cable which interchanges specific signals. All connectors used are standard 25-pin D-type connectors. The connectors on the junction panel are 25-pin, type D female connectors, model DB-25S. MUX cable requires male end; terminals and modems also typically require male end. Sample MUX cable connections are described below.

3.6.1 MIGHTY MUX-TO-TERMINAL CONNECTIONS

Device status, pin 20, is grounded (pin 7) at the MUX end, to inhibit noise which can cause automatic log-off in IRIS. If the terminal has a stable data-terminal-ready output on pin 20, it may be connected from pin 20 at the terminal end to pin 20 at the MUX end.

Figure 3-8 shows the MUX-to-terminal connections.

3.6.2 MIGHTY MUX-TO-MODEM CONNECTIONS

A full-duplex modem should be employed when using the IRIS Operating System.

In half-duplex systems, pin 20 of the data set must be connected at modem end to a +5 or +15-volt source, or jumpered to data-set-ready, pin 6.

Figure 3-9 indicates the MUX connections to modems.

MUX END	TERMINAL	FUNCTIONS (AS SEEN BY TERMINAL)
PIN NO.	PIN NO.	
2 ←	2	TRANSMIT DATA
3 →	3	RECEIVE DATA
JUMPERED 7	7	SIGNAL GROUND
JUMPERED 20		
(SOME TERMINALS MAY NOT REQUIRE THESE CONNECTIONS BUT THEY DO NO HARM) JUMPERED 4		REQUEST TO SEND
JUMPERED 5		CLEAR TO SEND
JUMPERED 6		DATA SET READY
JUMPERED 8		DATA SIGNAL CARRIER DETECT
JUMPERED 20		DATA TERMINAL READY

042–005 A

Figure 3-8. MUX-to-Terminal Connections

MUX END	FULL-DUPLEX MODEM	FUNCTION (AS SEEN BY MODEM)
PIN NO.	PIN NO.	
2 ←	3	RECEIVE DATA
3 →	2	TRANSMIT DATA
7 —	7	SIGNAL GROUND
8 →	20	DATA TERMINAL READY
20 ←	8	RECEIVED SIGNAL CARRIER DETECT

042–005 B

MUX END	HALF-DUPLEX MODEM	FUNCTION (AS SEEN BY MODEM)
PIN NO.	PIN NO.	
2 ←	3	RECEIVE DATA
3 →	2	TRANSMIT DATA
5 →	4	REQUEST TO SEND
7 —	7	SIGNAL GROUND
	JUMPERED 6	DATA SET READY
	JUMPERED 20	DATA TERMINAL READY
20 ←	5	CLEAR TO SEND

Figure 3-9. MUX-to-Modem Connections

3.6.3 CURRENT LOOP CONNECTIONS

The POINT 4 Data 322-CL Cable Assembly allows the user the versatility of plugging cables requiring either EIA-level or current-loop (C-L) interfacing into the same 25-pin connector. This is achieved by using pins 11, 18, and 25 (left unassigned by EIA spec RS-232C) for the current-loop connections (see Figure 3-4). Pin 25 carries the C-L output from the MUX; pin 18 receives the C-L input to the MUX; pin 7 (signal ground) serves as the return for both. Pin 11 must be jumpered to pin 2 whenever C-L is used; this should be done inside the 25-pin connector on the C-L cable at the MUX end.

The C-L circuitry on the 322-CL is driven by a +28-volt external power source, and is designed for 20 milliamps current in both input and output. This may be increased to 40 or 60 milliamps by adding two 1.4K (1-watt) or 680-ohm (2-watt) resistors on the 322 board, next to the selected 25-pin connector.

CAUTION

The MIGHTY MUX Current Loop Interface (322-CL) is designed for use with terminals that are passive (no voltage source) in both input and output. Do not plug an active terminal (internal C-L voltage source) into the 322-CL.

3.7 HARDWARE-SELECTABLE OPTIONS

The following options are available, via etch cuts and jumpering, on the MIGHTY MUX 310 board. Selection of these options is explained in the following subsections.

- Master Terminal Mode Removal Option
- 19.2K Baud-Rate Option
- Device Code Options
- Master Terminal Mode Device Code 50/51 Option
- Mask Bit Option
- Port Control Block Length Option
- Control Block Area Location Option
- Port Control Block Location Option
- Special Character Set Option
- MSB in ASCII Mode Option
- X Option
- 128KB-Addressing MUX Buffer Allocation Option
- Isochronous Option
- On-board Power Option

3.7.1 MASTER TERMINAL MODE REMOVAL OPTION

The Master Terminal Mode Removal Option provides a means of disabling the standard master terminal mode. The master terminal mode permits Port 0 to be used as the master terminal interface (device code 10/11) or a time-shared port.

Master terminal mode is disabled by the removal and addition of jumper wires at T, located between I.C. 3C and 3D, and the removal of a jumper between I.C. 7C and 8C. The disabling procedure is shown in Figure 3-10.

Effect On Diagnostic: None.

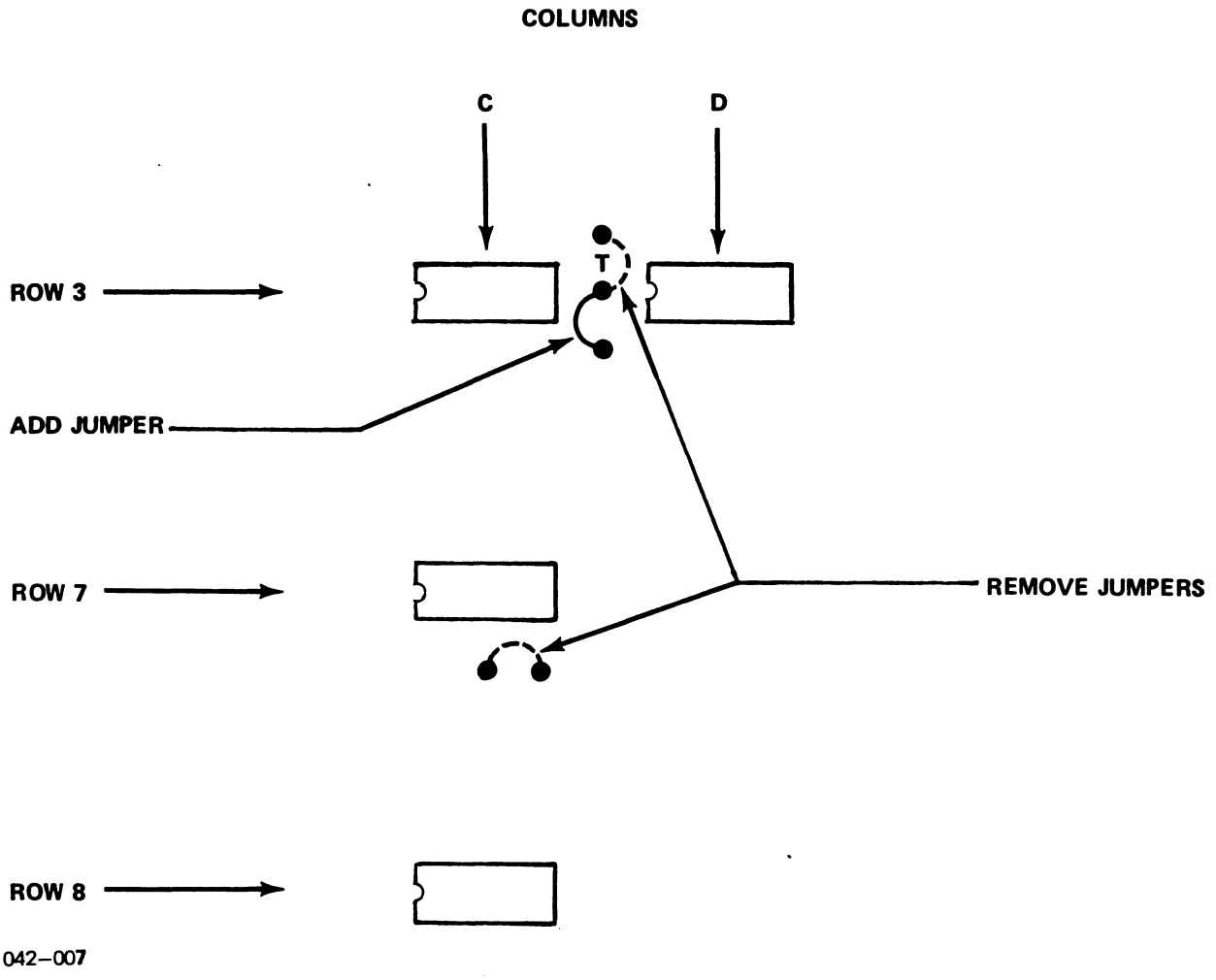
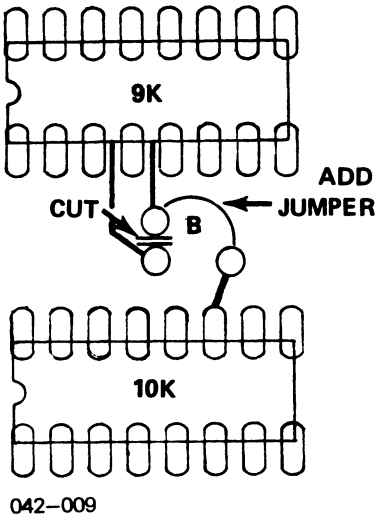


Figure 3-10. Disabling of Master Terminal Mode

3.7.2 19.2K BAUD-RATE OPTION (B OPTION)

The Baud-Rate Option allows the user to sacrifice 600 baud and replace it with 19.2K baud.

The B Option is enabled by cutting the etch between I.C. 9K and 10K at B, as indicated below. A jumper is installed over B, as shown.



Set up software to select 600 baud. Make sure that the terminal to be used will run at 19.2K baud.

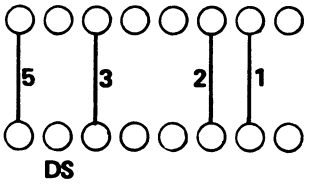
Effect On Diagnostic: Will cause diagnostic to fail in half-frequency test at 1200 and 600 baud. Refer to MIGHTY MUX Diagnostics Manual.

3.7.3 DEVICE CODE OPTIONS

These options preserve device codes 10 and 11 for the master terminal mode. The standard device code is 25 (octal). Options available (in octal) are:

4, 5, 14, 15, 20, 21, 24, 30, 31, 34, 35.

At location 3B of the 310 board, there are four jumpers (labelled DS) which control bits 1, 2, 3 and 5 of the six-bit device code. Bits 0 (the MSB) and 4 are permanently wired to zero. The jumper block at location 3B is as shown below:



042-010

The numbers represent the device-code bits. A jumper on the left makes that bit a 1; a jumper on the right makes it a 0. Thus, the standard-device code has bit 5 = 1, bit 3 = 1, bit 2 = 0, and bit 1 = 1.

0	1	2	3	4	5
0	1	0	1	0	1

Bits 0 and 4 both = 0, which makes the device code = 010,101 = 25 octal. To change the device code to any of the available options, decode which bits need to be changed, cut the appropriate jumpers and install their opposites.

When changing the incoming device code, it is necessary to change the code for 310 response to an INTA (interrupt-acknowledge) instruction. That code is controlled by jumpers (labelled I) at location 10F on the 310 board. For proper response of the 310 board to the new device code, the jumpers at location 10F should be changed in the same manner as those at location 3B.

Effect On Diagnostic: The MUX Diagnostic Program must be re-assembled to allow for the new device code. Refer to MIGHTY MUX Diagnostics Manual.

3.7.4 MASTER TERMINAL MODE DEVICE CODE 50/51 OPTION

This option changes the standard MUX device code from 25 to 45 and the master terminal mode device code from 10/11 to 50/51.

The standard device code has bit 0 (MSB) and bit 4 permanently wired to zero. MUX device code = 45 and master terminal mode device codes = 50/51 require bit 0 = 1. Therefore, in addition to standard jumper changes (for bit 1 = 0), at locations 3B and 10F, modification to bit 0 is also required as indicated below:

1. Cut etch between 2D pin 11 and edge connector pin A72. Cut etch between 3A pin 7 and 2D pin 10. Add jumper between 3A pin 7 and edge-connector pin A72.

2. Cut etch between 3B pin 7 and 3B pin 10. Add jumper between 3B pin 8 and 3B pin 9.

3. Cut etch between 10E pin 8 and 13E pin 1. Add jumper between 10E pin 8 and 13E pin 12.

4. Cut etch between 10F pin 7 and 10F pin 10. Add jumper between 10F pin 7 and 10F pin 12.

This results in:

	0	1	2	3	4	5	
MUX Device Code:	1	0	0 .	1	0	1	= 45 octal
master terminal mode Device Code:	1	0	1 .	0	0	0	= 50 octal TTI
	1	0	1 .	0	0	1	= 51 octal TTO

Effect On Diagnostic: Diagnostic will have to be re-assembled to test master-terminal mode. Refer to MIGHTY MUX Diagnostics Manual.

3.7.5 MASK BIT OPTION

The MIGHTY MUX offers the option of using any bit, 0 through 15, as the mask bit. Bit 5 (MSB is bit 0) is standard.

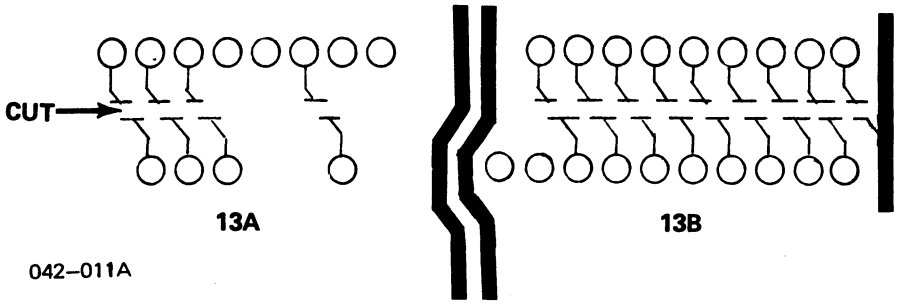
The mask-bit jumper (M) is located next to pin 12 of I.C. 9C. To change the mask bit to another bit, jumper M should be cut. The feed-thru closest to pin 12 of I.C. 9C is jumpered to the appropriate I.C. pin for the required mask bit, as specified below:

Required Mask Bit	Jumper To
0	I.C. 14E pin 13
1	I.C. 14E pin 6
2	I.C. 14E pin 10
3	I.C. 14E pin 3
4	I.C. 12E pin 13
6	I.C. 12E pin 10
7	I.C. 12E pin 3
8	I.C. 13E pin 13
9	I.C. 13E pin 6
10	I.C. 13E pin 10
11	I.C. 13E pin 3
12	I.C. 11E pin 13
13	I.C. 11E pin 6
14	I.C. 11E pin 10
15	I.C. 11E pin 3

Effect On Diagnostic: The MUX Diagnostic Program requires a patch for the new mask bit. For example, for mask bit = 3, change the word at location EIMUX-2 (in the 310 MIGHTY MUX Diagnostic Program) from 175777 to 167777. Instead of bit 5 = 0, bit 3 now = 0. The corresponding checksum at CKS1 must also be changed. Refer to MIGHTY MUX Diagnostic Manual.

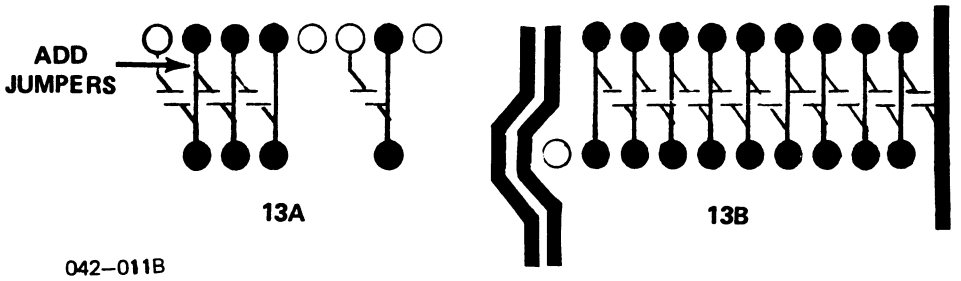
3.7.6 PORT CONTROL BLOCK LENGTH OPTION

In this option, block length on the model 310 MUX is controlled by jumpers at location 13A and 13B. To change block length from default (40 words octal), cut etch lines as shown below and add jumpers for appropriate new block length.



3.7.6.1 20-Word (Octal) Block Length

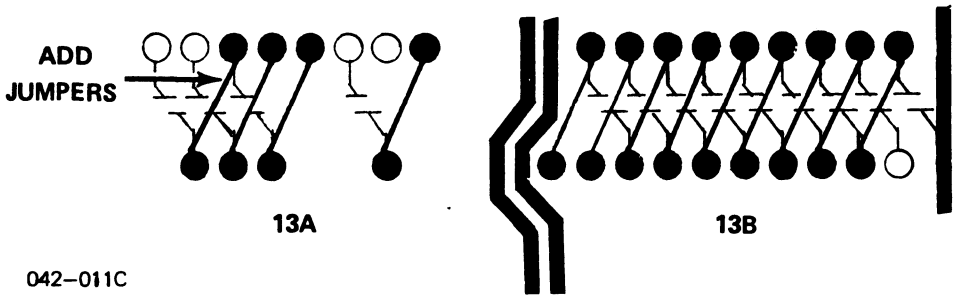
At locations 13A and 13B, jumper pins are added as shown below.



Test for accuracy of block length by running the MUX diagnostic and confirming block length typeout (in octal - 20).

3.7.6.2 10-Word (Octal) Block Length

At location 13A and 13B, jumper pins are added as shown below.



Test for accuracy of block length by running the MUX diagnostic and confirming block length typeout (in octal - 10).

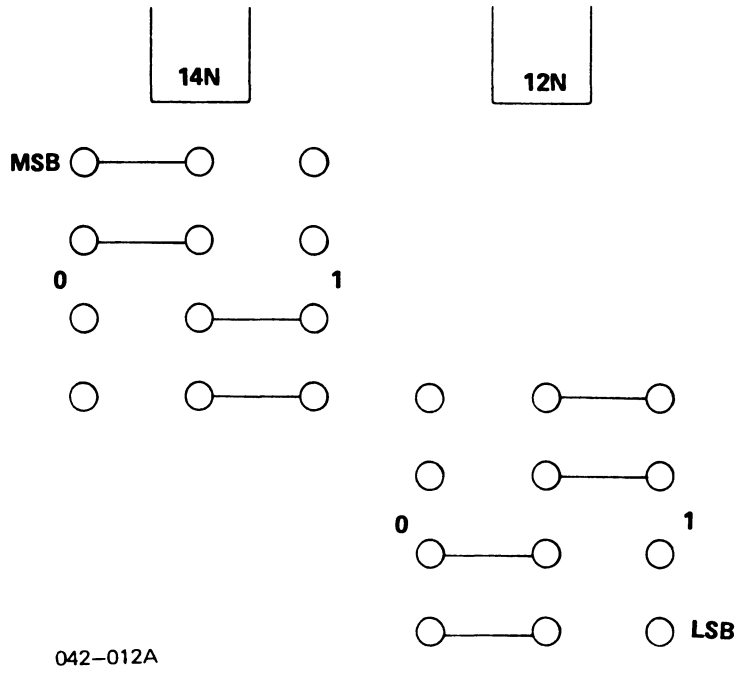
Effect On Diagnostic: None.

3.7.7 CONTROL BLOCK AREA LOCATION OPTION

All port control blocks are contiguous in memory. The beginning of the entire control block area is initialized (IORST) to a hardware-controlled default value, after which it may be changed by software (before the MUX is turned on).

The standard default value of the beginning of the control area is 36000 (octal). With the control block location option, the default value may be any other multiple of 400 (octal).

The hardware default value for the control block area base address is specified by the use of jumpers near locations 14N and 12N as shown below:



A jumper on the left signifies that a particular bit equals 0; a jumper on the right signifies a bit equals 1. MSB corresponds to the most significant bit (bit 0); LSB corresponds to the least significant bit (bit 7). By utilizing these hardware jumpers, it is possible to specify the eight most significant bits of the control block area base address to some multiple of 400 octal. The control block area base address specified by the jumpers shown above is:

```
MSB                     LSB
0   0  1  1   1  1  0   0  X  X   X  X  X   X  X  X
0 .    3    .    6    .    X    .    X    .    X
```

where the X values are supplied by the MUX control logic to specify the different addresses of each of the port control blocks within the control block area. The control block area base address = 36000, and each individual port control block area has an address = 36000 + XXX.

To change to a nonstandard hardware default value, it is necessary to decode the required address (octal to binary), then cut and install the appropriate jumpers.

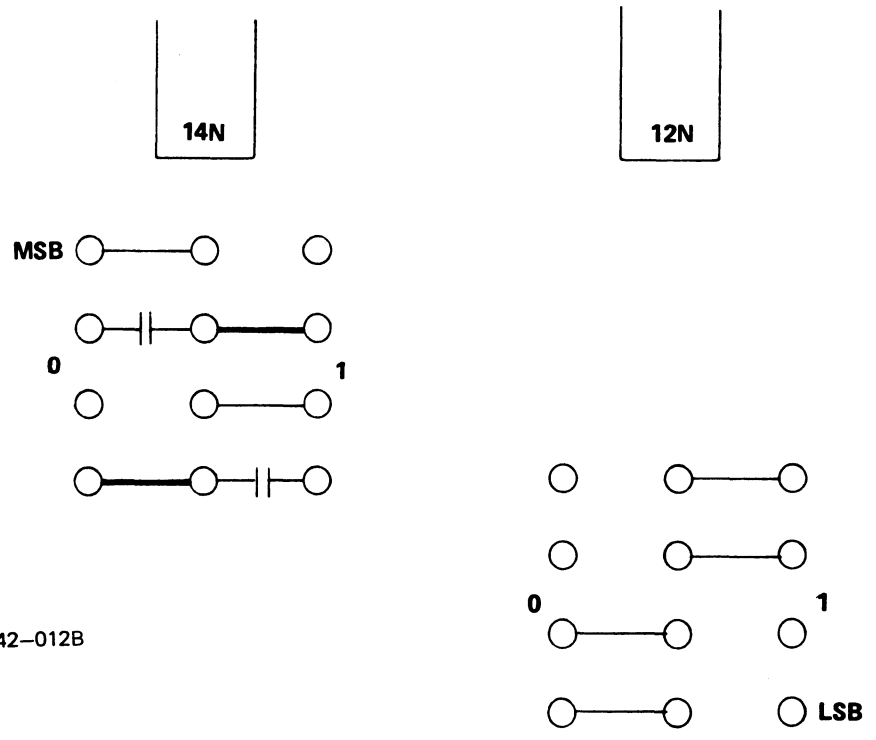
For example, to set up a control block area base address equal to 66000, the appropriate bits will decode as follows:

```
MSB                     LSB

0   1  1  0   1  1  0   0

0 .    6    .    6    .    0
```

Bits 1 and 3 are changed by cutting the standard etches for bits 1 and 3, and replacing them with their opposites, as shown below:



Effect On Diagnostic: These modifications have no effect on the MUX Diagnostic. Run the diagnostic, and test for accuracy of control block area base address by confirming the MUX control-area-base typeout. (In the above example, the MUX control area base typeout in octal should be equal to 66000). Refer to MIGHTY MUX Diagnostics Manual.

3.7.8 PORT CONTROL BLOCK LOCATION OPTION

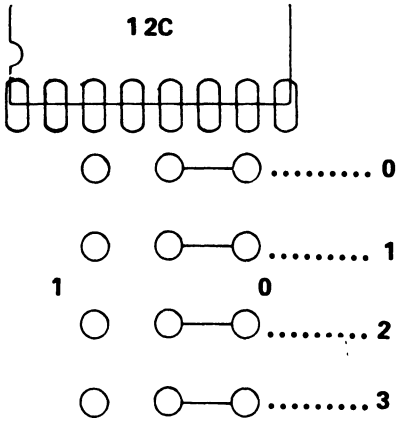
This option is applicable when using 301 expansion boards.

Eight ports (or fraction thereof) on a 310 or 301 board constitute a set whose port control blocks must be contiguous in memory. As the MIGHTY MUX is expandable up to 128 ports, there may be up to sixteen 8-port sets.

The eight ports on the 310 constitute set 0; their control blocks are at the beginning of the control block area.

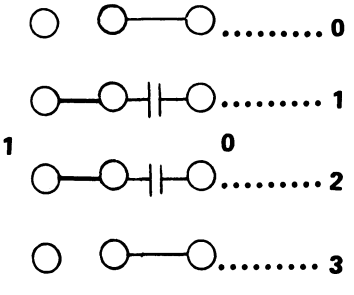
Using this option, the 310 ports may be any set from 1 thru 15.

The 310 port control block set is specified by jumpers situated next to the I.C. at location 12C, and is set up as shown below:



042-013 A

Standard jumpers, positioned as shown (all zeroes), specify the 310 port control blocks as set 0. To specify sets 1 through 15, cut etches and insert jumpers as required to change appropriate bits from 0 to 1. With bit 3 = LSB, and bit 0 = MSB, the number thus created is the binary equivalent of the required set number. For example, for 310 port control blocks = set 6, the zero jumpers of bits 1 and 2 would be cut, and their opposites inserted as shown below:



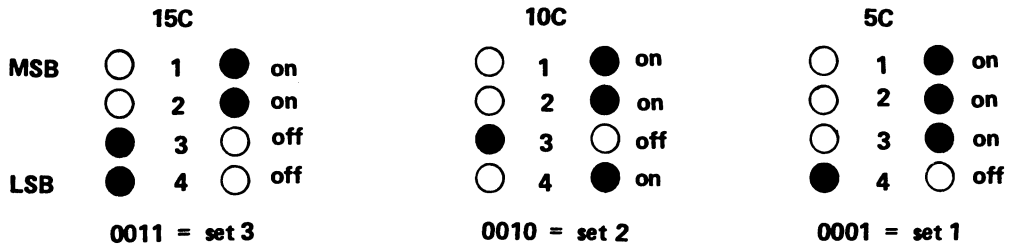
042-013 B

This results in: 0 1 2 3

0 1 1 0 = 6 octal

This port control block set-up technique also applies to expansion boards. Note that these boards are supplied with DIP switches rather than jumpers, to facilitate ease of set-up, and have no standard set values.

For example, the switches on a 24-port 301 board (three sets of eight ports), which are located at 5C, 10C, and 15C, could be set up for ports 8 through 15, 16 through 23, and 24 through 31 (with ports 0 through 7 on the 310 board). The 301 expansion board would then be set up for port control block areas corresponding to sets 1, 2, and 3 (the 310 board is set to 0) and this could be implemented as shown below:



042-014

In this case, "on" denotes a logical 0, and "off" a logical 1, with switch 4 being the least significant bit, and switch 1 being the most significant bit of the set number.

NOTE

Sets should be selected so as not to leave any gaps.

Effect On Diagnostic: These modifications have no effect on the MUX Diagnostic. Run the diagnostic and test for correctness of MUX control block area base address typeout. (For example, for set 6, assuming port control block size = 40 octal, the MUX control block base address will be = 36000 + 40x6x10 = 41000 octal, assuming that the diagnostic is run on the 310 board alone. If the diagnostic is run on the 310 board in conjunction with a 301 board whose port control block areas constitute sets 0 thru 5, the MUX control block area base address typeout will still be equal to 36000). Refer to MIGHTY MUX Diagnostics Manual.

3.7.9 SPECIAL CHARACTER SET OPTIONS

For Special-Character-Set options, 256x4 PROM (Programmable Read-Only Memory) is used to determine which characters will produce interrupts when requested by the software. The character itself is used as an address into the PROM and one of the outputs is used to identify a special character. The options relate to which output is used and whether the MSB of the address is the MSB of the character (full 8-bit character set), or another control signal in the MUX. This results in two different 7-bit character sets, such as one for input and the other for output, or one for 310 ports and the other for expansion ports.

3.7.9.1 Special Character Set Usage

For example, when we require different special-character sets for input and output, we can use the control signal QO+ as the most significant bit of the PROM address. For input, the most significant bit will be high, specifying one set of PROM addresses (one group of four special-character sets). For output, the most significant bit will be low, specifying another set of PROM addresses, and another group of four character sets.

Note that each PROM address specifies a four-bit memory location, corresponding to whether or not this particular PROM address is to be a special character in each of these four special-character sets.

In those cases where both input and output require the same special-character set, the most significant bit of the PROM address can be set either high or low, and the special-character set will be the same for both input and output. If the most significant bit were initially set high (or low), then by setting the most significant bit low (or high), another set of PROM addresses is specified which can be used to define another special-character-set, assuming that both input and output require the same special character set. For an example of the condition, see options 2 and 2a in Section 3.7.9.2.

Standard (Option #0): All ASCII characters <40 or >173 (testing only 7 bits) are "special" characters, for both input and output.

Option #1: 3, 4, 31 (both input and output)

Option #1a: 3, 12, 15, 177 (both input and output)

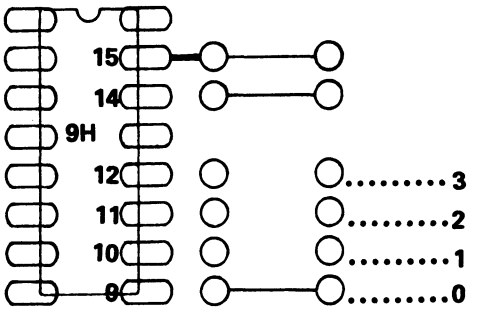
Option #2: 0, 4, 15, 21, 23 (both input and output)

Option #2a: 136 (both input and output)

Option #3: Input: 0,3,4,5,6,10,14,15,25,27,30,31,32,33
Output: 0,3,4,5,6,14,15,25,27,30,31

3.7.9.2 Special Character Set Implementation

The Standard Character Set (Option #0) is shown below. The jumper next to pin 15 of I.C. 9H corresponds to the most significant bit of the PROM address and is tied to QO+ which is high (1) for input and low (0) for output. The jumper next to pin 14 of I.C. 9H is connected to one of the PROM enable inputs, and is always grounded. It is designed to allow possible future expansion to a 512x4 PROM. The jumpers next to pins 9 through 12 of I.C. 9H are used to specify options 0 through 3 as shown:



042-015

To change from standard to some other special-character set, cut the jumper next to pin 9 (for Option #0) and replace it with the required Option jumper 1, 2 or 3.

To distinguish between Option #2 and Option #2a, the jumper next to pin 15 of I.C. 9H must also be cut and replaced by a jumper from the pad next to pin 15 to ground (for Option #2) or to a pull-up (for Option #2a). A convenient pull-up is located at pin 5 of I.C. 10H; pin 14 of I.C. 9H is grounded by the jumper next to it.

Effect On Diagnostic: The MUX Diagnostic requires an overlay or patch. Contact POINT 4 Data for the appropriate modification for a particular special-character set and/or the ASCII mode option. Refer to MIGHTY MUX Diagnostics Manual.

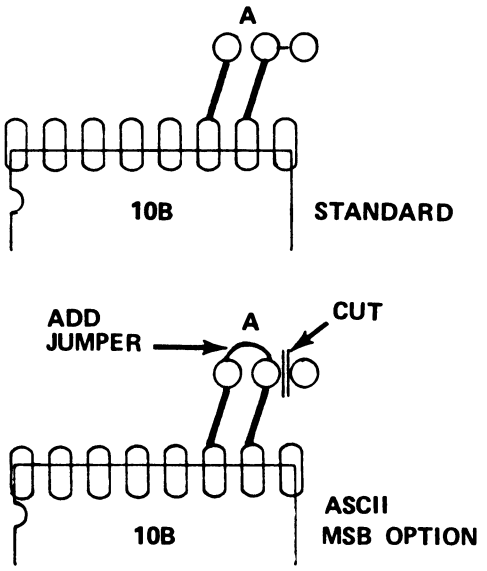
3.7.10 MSB IN ASCII MODE OPTION

This is an Input-Only option. When the special-character interrupt is requested by the software, the MUX can automatically set the MSB of the incoming character to 1. This will affect all incoming characters, whether special or not, but only on those ports where special-character interrupt is selected.

In the standard mode, the MSB is set to one (1) on all incoming characters if special-character interrupt is requested.

In the optional mode, the MSB is set to zero (0) if port-control output was set up for 7-bit character length, or left unchanged if port-control output was set up for 8-bit character length.

To enable this option (to prevent the MUX from setting the most significant bit of the incoming character to one), the etch next to pins 8 and 9 of I.C. 10B should be cut, and a jumper installed at "A" as shown below:



042-016

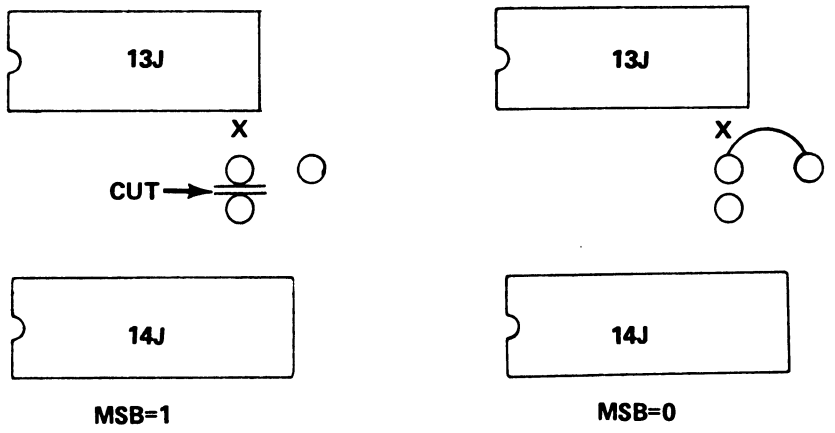
Effect On Diagnostic: The MUX Diagnostic requires an overlay or a patch. Contact POINT 4 Data for the appropriate modification for a particular special-character set and/or the ASCII mode option. Refer to MIGHTY MUX Diagnostics Manual.

3.7.11 X OPTION

In the standard mode, the MSB of the data buffer area always conforms to the MSB value of the port control block base address. The X Option can fix the MSB to one (1), zero (0), or default to the standard mode.

To fix the MSB to one (1), cut the etch between the two pins at "X", situated between I.C. 13J and 14, as shown below.

To fix the MSB to zero (0), install a jumper as indicated.



042–017

Effect On Diagnostic: None.

3.7.12 128KB-ADDRESSING MUX BUFFER ALLOCATION OPTION

The MUX Buffer Allocation option is a factory-installed, extra-cost option which provides buffers in both upper and lower memory.

In the standard format, bit 6 of the input control word (word 0) is the not-echoed bit, and bit 6 of the output control word (word 1) is not used (see Tables 2-1 and 2-2).

With this option, bit 6 of the ICW and bit 6 of the OCW become the most significant bits for direct-memory-access transfer.

The optional formats for the ICW and OCW utilizing the MUX Buffer Allocation option are shown in Figure 3-11.

Effect On Diagnostic: Will cause diagnostic to fail in Echo Test. Refer to MIGHTY MUX Diagnostics Maual.

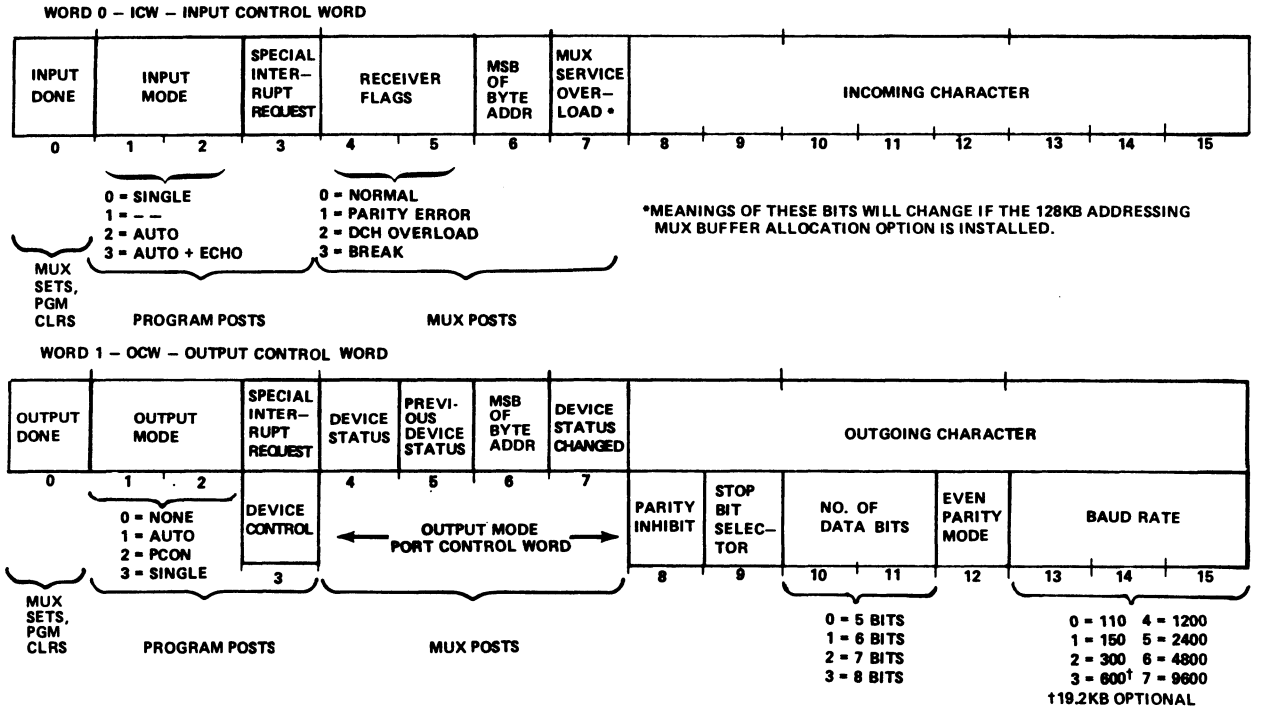


Figure 3-11. Optional ICW and OCW Formats For MUX Buffer Allocation Option

3.7.13 ISOCHRONOUS OPTION

The Isochronous Option is a method of utilizing a synchronous modem and its improved efficiency without requiring synchronous receiving circuitry. This hardware-selectable mode is a hybrid of asynchronous and synchronous operation.

Isochronous data transmission requires contiguous characters with the addition of a start and a single stop bit added to each character. Unlike asynchronous transmission, it operates on a bit-by-bit synchronization. After a stop bit, the next character starts at "n" bits, rather than an arbitrary time interval.

In its implementation, ports must operate in pairs (0 and 1, 2 and 3, etc.) at the same baud rate.

NOTE

It is posible for some port pairs to be isochronous while the rest are asynchronous.

To enable this option, the following board modifications are necessary (see Figure 3-12):

1. Implement the following I.C. changes, as required:

   - Remove I.C. 1H (9334)
   - Port 0&1: Remove I.C. 2K (74151); install I.C. 2.5N (74161)
   - Port 2&3: Remove I.C. 4K (74151); install I.C. 5.5N (74161)
   - Port 4&5: Remove I.C. 6K (74151); install I.C. 7.5N (74161)
   - Port 6&7: Remove I.C. 8K (74151); install I.C. 8.5N (74161)

2. Cut the etches between feedthroughs at a, d, g and m, as indicated in Figure 3-12. These locations are all in column K.

   Cut etches at b, c; e, f; h, k; n and p all in column N.

3. Add jumpers at a, b, c; d, e, f; g, h, k; m, n, and p, as indicated.

Effect On diagnostic: The diagnostic can not run on two ports in the same port pair. For example, if ports 0 and 1 are paired, as are ports 2 and 3, the diagnostic cannot be run between ports 0 and 1 or between ports 2 and 3. It may be run between 0 and 2, 0 and 3, 1 and 2, and 1 and 3.

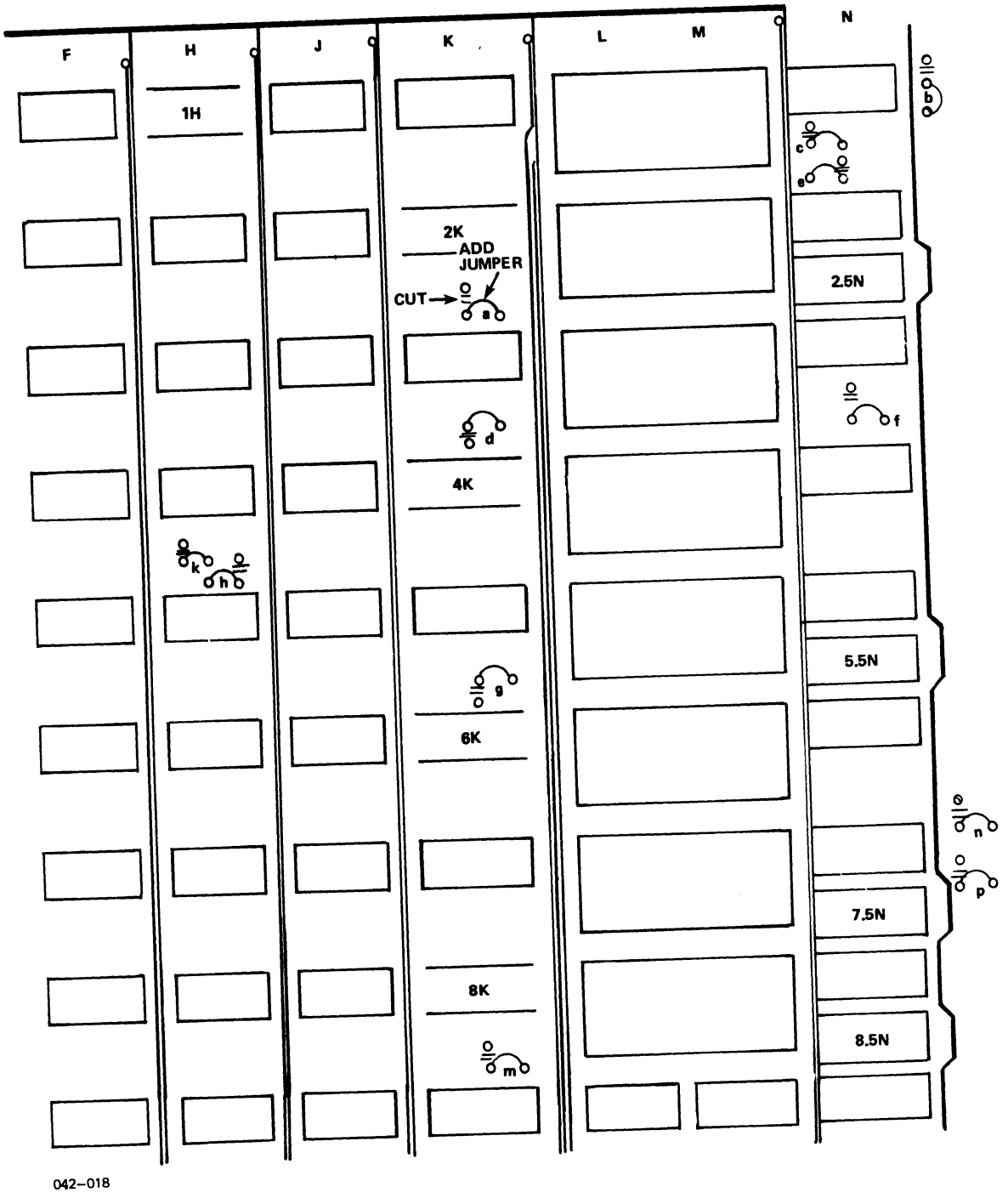


Figure 3-12. Isochronous Option Implementation

3.7.14 ON-BOARD POWER OPTION

The On-Board Power Option is a factory-installed, extra-cost option.

In standard mode, the MUX takes its ±12 volt power requirements from an external power supply, Model 340-32 or 340-64. The ribbon-cable power connection goes from the power supply to the 322 connector/cable assembly, to the 310 board.

In the optional mode, the MUX generates its ±12 volt requirements from the CPU power supply. MUX contains all necessary voltage regulation, protection, and filtering circuitry.

To implement this option, the customer must assure connection and availability of sufficient current from the CPU power supply:

VINH (nominally +15V unregulated) - 250ma - pin B84
-15V (nominal, may be unregulated) - 250ma - pin B93

NOTE

This option is available only on 310 boards. All 301 boards require a 340 or 342 power supply.

Effect On Diagnostic: None.

3.8 MULTIPLEXER TIMING AND ELECTRICAL SPECIFICATIONS

Essential I/O timing specifications and overall MUX timing sequences are described in the following sections.

3.8.1 INPUT/OUTPUT TIMING SPECIFICATIONS

The MIGHTY MUX I/O timing specs are determined primarily by the universal asynchronous receiver/transmitter (UART) chips used in each port; the electrical specs are determined by the TTL/EIA converter chips. The interrelationship between these circuits is shown in Figure 3-13.

The basic timing source is a 6.7584 MHz ±.005% crystal oscillator. This is followed by a frequency divider producing 16 times the selected baud rate, which in turn drives the UART chip. The UART receiver is designed to allow up to 47% time distortion, while the transmitter produces less than 1% distortion. This makes the MIGHTY MUX compatible with virtually all asynchronous modems and terminals currently in use.

3.8.2 ELECTRICAL SPECIFICATIONS

The voltage levels of the MIGHTY MUX standard I/O circuitry are in accordance with EIA Specification RS-232C. The 1488 chips are driven from a ±12-volt source, and the outgoing levels are:

```
Data = 0 or device control = 1:  +10 volts
Data = 1 or device control = 0:  -10 volts
```

For incoming lines, the 1489 chips give the following response:

```
> +1.3V:                  Data = 0 or device status = 1
< +0.7V or open:          Data = 1 or device status = 0
Between +.7V and +1.3V:   Indeterminate
```

The 1489 offers an input resistance of about 3.5K; the 1488 can drive any load above 2K.

The optional MIGHTY MUX current-loop interface is driven from a +28-volt source through a current-limiting resistor of 1.4K for the 20-milliamps version (or 700 ohm for 40 milliamps, or 470 ohm for 60 milliamps). These values apply to both the input and output circuits, as shown in Figure 3-14.

CAUTION

The MIGHTY MUX current-loop interface (322-CL) is designed for use with terminals that are passive (no voltage source) in both input and output. Do not plug an active terminal (internal C-L voltage source) into the 322-CL. See schematics in Figure 3-14.

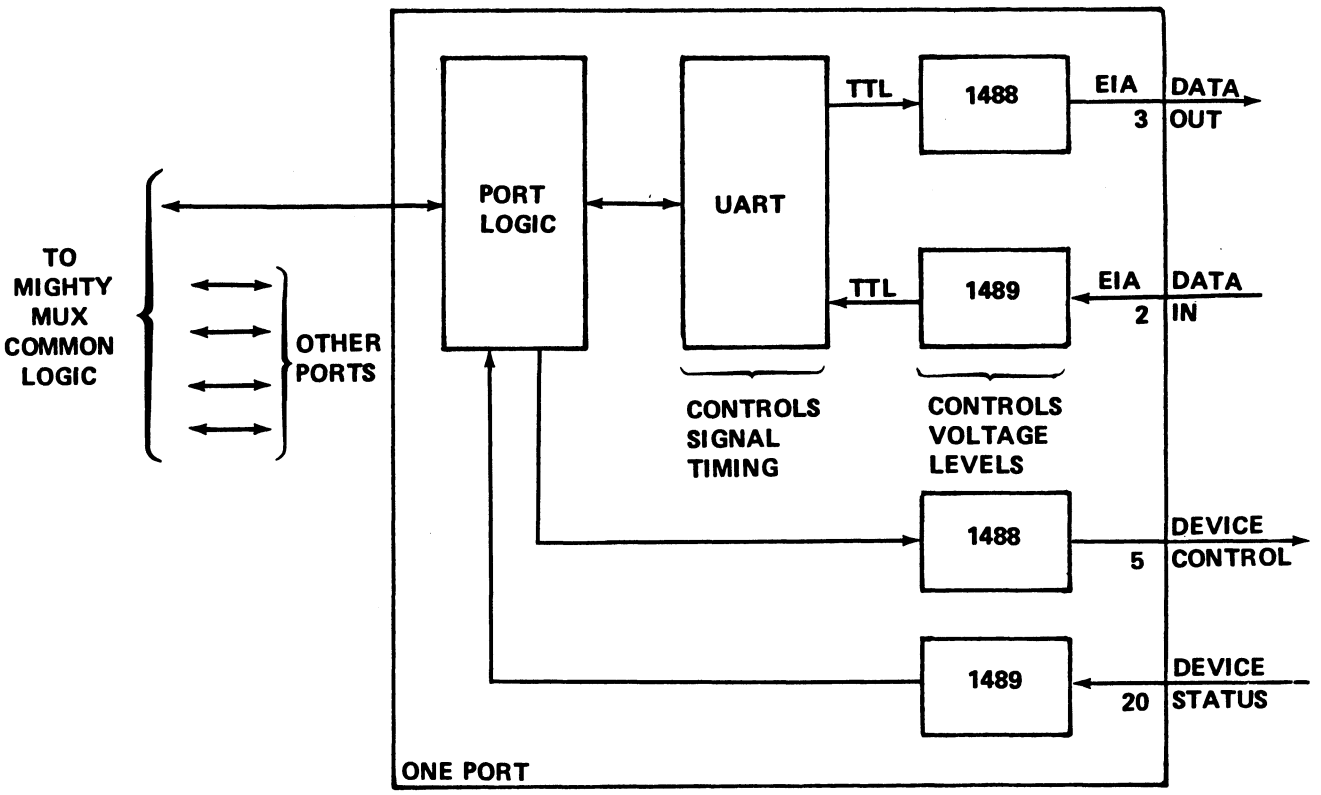


Figure 3-13. Block Diagram of EIA Interface

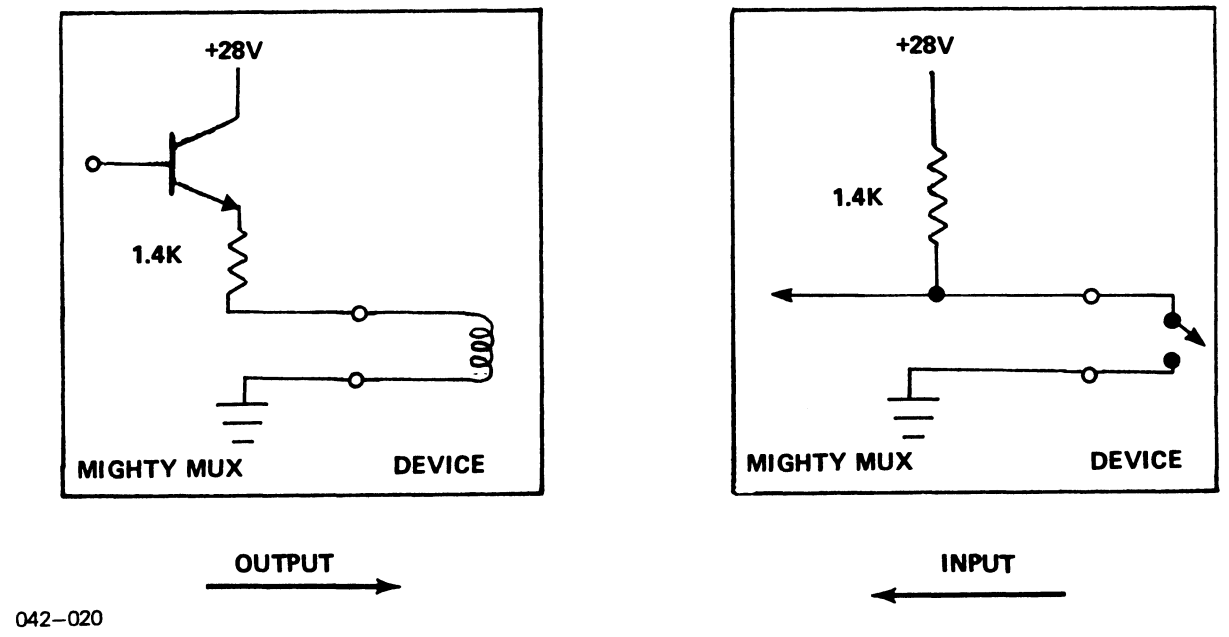


Figure 3-14. Current-Loop Circuitry

3.8.3 OVERALL MUX TIMING

When no input or output is taking place, the MUX inspects each OCW once every 50 milliseconds. Therefore, when the software starts an output by setting an OCW to an active output mode, there may be a delay of up to 50 milliseconds before the MUX begins transmission. This delay can be avoided by the use of a DOA ac,MUX instruction to "prod" the MUX into immediate action without waiting for the 50 milliseconds outtime delay. Once transmission has begun on a port, the MUX will retest that OCW for the next output command as soon as the port's output-buffer register is empty and the port counter has reached the appropriate port (typically 100 microseconds maximum).

Figure 3-15 is a diagram of MUX overall timing.

If the device status changes while there is no output on that port, there is again a delay of up to 50 milliseconds before the status-changed bit is posted and an interrupt is produced. When output is in process, a device status change produces the interrupt as soon as transmission of the current character is completed.

When an incoming character is received, the MUX will store it and produce an interrupt, if appropriate, within 100 microseconds after the center of the first stop bit. If multiple interrupts occur too quickly for the CPU interrupt service routine to handle, then the MUX status word for each port (including port identity) is stored by the MUX in a FIFO stack (up to 40 maximum) until the CPU interrupt service routine is able to service each of them in turn. The MUX presents interrupts to the CPU in the order in which they were detected.

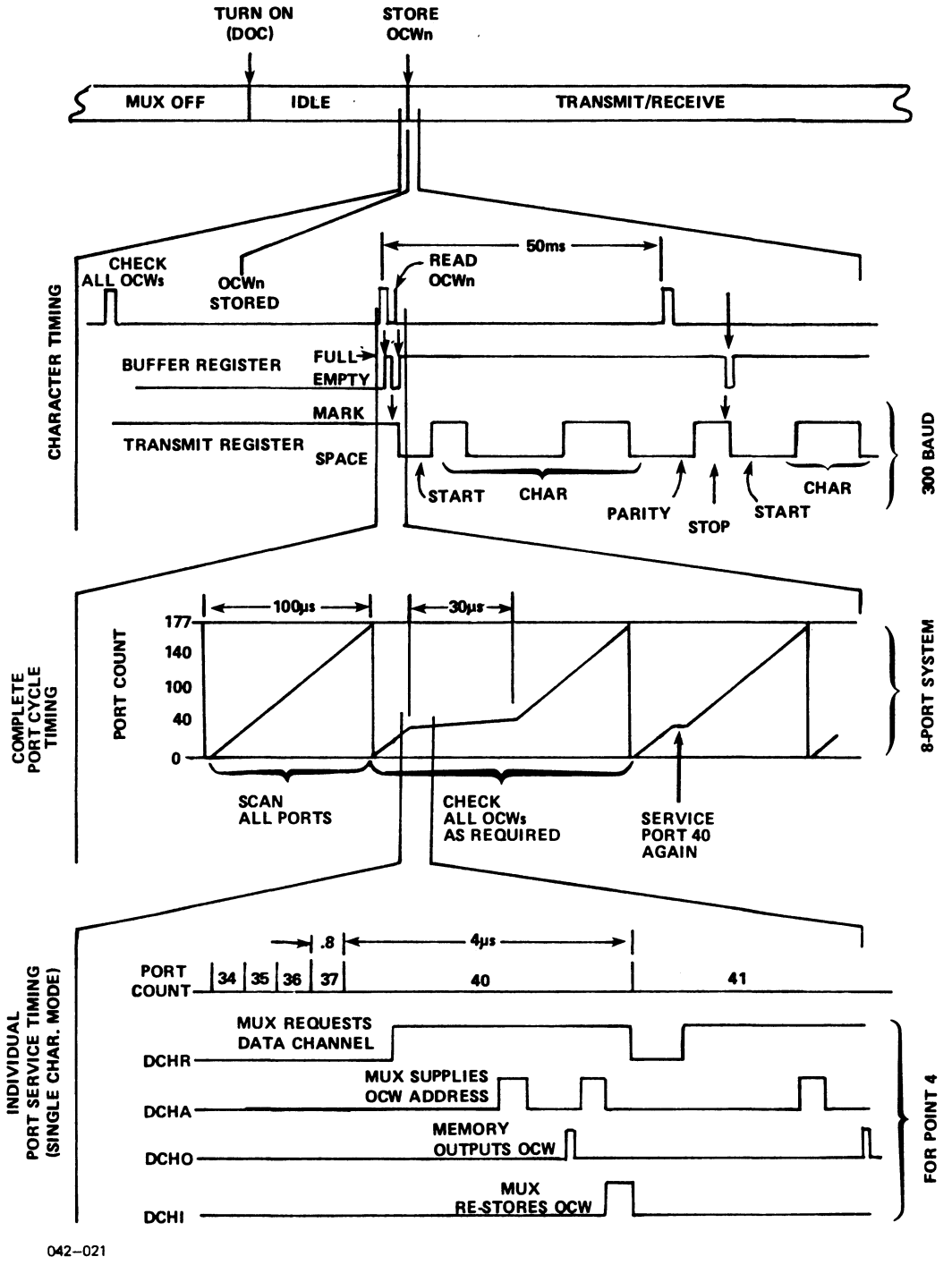


Figure 3-15. Overall Timing Diagram

3.8.4 MAXIMUM DATA RATE PER PORT

The relationship between number of ports and maximum data rate per port is shown in Figure 3-16. This figure assumes that all ports operate simultaneously at the same data rate. If some of the ports have a lower data rate, a correspondingly larger number of ports may operate simultaneously. Thus, for example, 80 ports at 9600 baud impose about the same load as 60 ports at 9600 baud plus 40 ports at 4800 baud. This is not an exact relationship, however, and it is wise to allow a reasonable safety margin when making such a conversion.

When the MUX operates at the maximum total data rate it requires approximately 80% of CPU time. This is due to the fact that outputting one character takes five data channel cycles (one to transfer it and four to set up for the transfer - see I/O flowchart in Section 2.3). After a transfer, the MUX will give up the data channel for at least one cycle. If the total data rate is less than the maximum, the overhead ratio decreases in direct proportion. For example, ten ports at 9600 baud plus 80 ports at 1200 baud result in an 18 percent overhead on a POINT 4 unit, because this is equivalent to 20 ports operating at 9600 baud, which is approximately 22 percent of the maximum rate of 44K baud for 20 ports. Eight ports operating simultaneously at 1200 baud would produce a 1.6 percent overhead on a POINT 4 unit; 1200 baud is approximately 2 percent of the maximum eight-port rate of 66K baud.

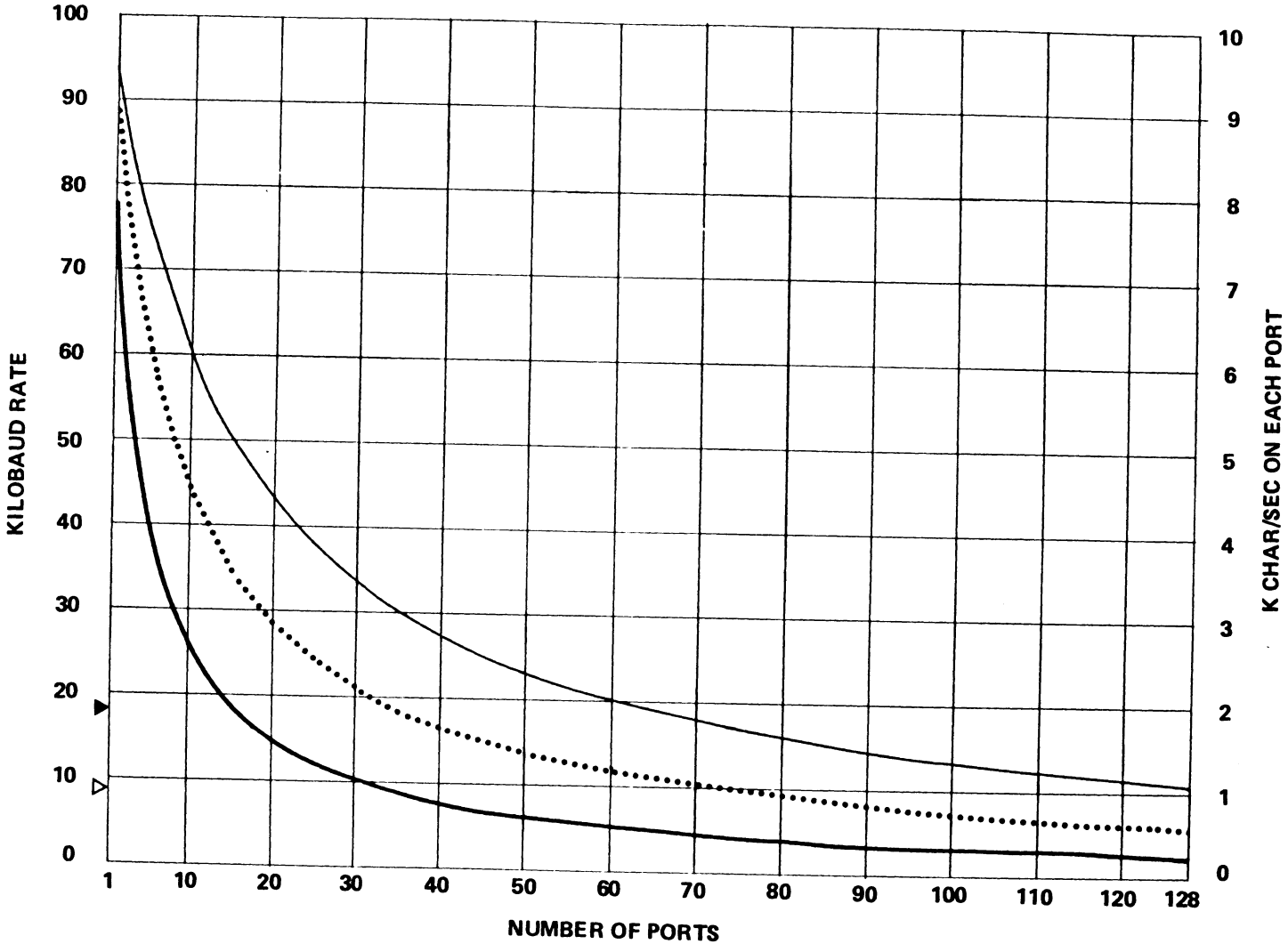


Figure 3-16. Relationship Between Number of Ports and Maximum Data Rate Per Port

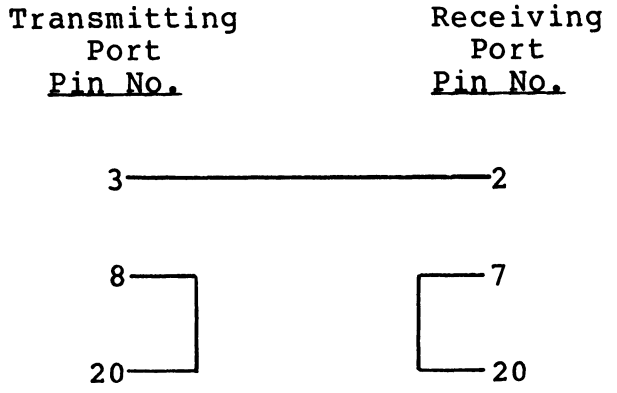
APPENDICES

Appendix A
MUX DIAGNOSTICS

This appendix contains cable connections for the MIGHTY MUX 310/301 Diagnostic Program and an elementary diagnostic test if the MUX Diagnostic is not available.

A.1 MUX DIAGNOSTIC CABLE CONNECTIONS

If there are any indications that the MUX is not operating properly, use the MIGHTY MUX Diagnostic Program. This program tests all MUX operational modes, and types out appropriate fault message if it finds errors. Test cable connections for the MUX Diagnostic are shown below:



NOTE: No other wires should be connected!

A.2 MINI-TEST USING COMPUTER FRONT PANEL

If the MIGHTY MUX Diagnostics Program is not available, the following elementary test may be used:

Set Switches	Press	Comment
Any ICW	Reset, Examine	
0	Deposit	Clear ICW
40000	Deposit Next	OCW = port control word
63025	Deposit Next	DOC 0,MUX; turns MUX on
400	Deposit Next	JMP .; allows data channel action
ICW+2	Start	
ICW	Reset, Examine	Should still be 0
	Examine Next	OCW should be 140000 or 146400

A.3 MINI-TEST USING VIRTUAL FRONT PANEL

If the operator uses the POINT 4 Data Virtual Front Panel, the Mini-Test is typed as follows (items in parentheses are typed out automatically by the computer):

```
 36000:      0
(36001:)     40000
(36002:)     63025
(36003:)     400
(36004:)     Escape
J36002
Press STOP, APL
D36000
(36000:      0 , 140000...)
```

If OCW is 146400, it means that the incoming device-status line has a positive voltage, and the MUX is working correctly. If ICW is 106777, it is a strong indication that the -12 volt supply is not reaching the 310 board (especially if the same occurs in all ICWs). This may be checked with a voltmeter or an oscilloscope at pin 2 of any of the 40-pin UART chips.

A.4 CONTROL AREA SEARCH PROGRAM

If ICW and OCW are still 0 and 40000, respectively, it is possible that the MUX is not properly searching for its control blocks. The following program can be used to search for the MUX's apparent control area.

```
 0:    20014  LDA     0,14       ;start here
 1:    30015  LDA     2,15
 2:    41000  STA     0,0,2      ;store 40000 in all words
 3:   151404  INC     2,2,SZR
 4:        2  JMP     2
 5:    63025  DOC     0,MUX      ;start MUX
 6:    30015  LDA     2,15
 7:    25000  LDA     1,0,2      ;check all words
10:   106414  SEQ     0,1
11:    63077  HALT               ;word not = 40000
12:   151400  INC     2,2
13:        7  JMP     7
14:    40000  40000              ;MUX control word
15:   100016  100016             ;initial address (MSB=1)
```

This program stores 40000 (port-control output or auto input) in each word above the program up to the end of memory, then starts the MUX and tests each word to see if it has changed. If so, it halts, with the changed word in A1, and its address (with MSB = 1) in A2.

Press Reset and start at 0.

The program should halt, with A2 containing the address of the first OCW, and A1 containing either 140000 or 146400, dependent upon whether the device status is zero (0) or one (1). If Continue is then pressed, the program should stop at each OCW, until it halts at the top of memory, with A2 indicating the amount of memory in the system.

If the program halts with an ICW address in A2, it indicates data (or noise) is coming in, or (if the data is all ones) that the -12 volt supply is missing. The program may also halt at the corresponding input byte pointer, which may have been incremented to 40001.

If the program halts with any other value in A2, the address switches on the 301 boards may be set incorrectly.

If it halts only at the top of memory, it may indicate that the MUX control block area default value is set to a value greater than the amount of memory available, that the MUX board is not plugged in properly, or that the board is not getting a +5 volt supply from the computer.

If the program does not halt at all, it indicates that the MUX is never releasing the data channel. The most likely cause is that the 310 board is not receiving the data-channel-priority signal.

Appendix B
CABLE LENGTH CONSIDERATIONS

The following considerations apply to the specification of maximum cable length between the POINT 4 Data MIGHTY MUX and a CRT or other terminal, using the RS-232 interface.

The Electronics Industries Association (EIA), in Recommended Standard RS-232C, states, "The use of short cables (each less than approximately 50 feet...) is recommended; however, longer cables are permissible, provided that the resulting load capacitance...does not exceed 2500 picofarads." The 50-foot recommendation is generally viewed as extremely conservative.

In normal noise environments, the limiting factor determining maximum cable length is waveshape distortion due to resistance-capacitance effects. This factor is directly proportional to line length and baud rate. The following maximum line lengths represent safe engineering practice:

Baud Rate	Maximum Length
9600	500 feet
4800	1000 feet
2400	1/2 mile
1200	1 mile

The same relationship should continue for several miles, after which the limiting factor will become resistive voltage drop.

Ordinary unshielded wire may be used such as telephone company interior wiring. In foot-traffic areas, standard 3-wire a.c. line cord with heavy insulation makes a sturdy, inexpensive cable.

In noisy environments (cable runs in close proximity to fluorescent lights or air conditioning or elevator motors), the limiting factor becomes noise pickup and shielded wire should be used. The same line lengths may be employed.

The only effect of excessive cable length will be the occasional incorrect transmission of a character. If this happens too frequently to be acceptable, the operator should switch to a lower baud rate.

Appendix C
ASCII CODE CHART

ASCII CODE in OCTAL

000	NUL	<CTRL-@>	040	BLANK	100	@	140	`
001	SOH	<CTRL-A>	041	!	101	A	141	a
002	STX	<CTRL-B>	042	"	102	B	142	b
003	ETX	<CTRL-C>	043	#	103	C	143	c
004	EOT	<CTRL-D>	044	$	104	D	144	d
005	ENQ	<CTRL-E>	045	%	105	E	145	e
006	ACK	<CTRL-F>	046	&	106	F	146	f
007	BEL	<CTRL-G>	047	'	107	G	147	g
010	BKSP	<CTRL-H>	050	(	110	H	150	h
011	HTAB	<CTRL-I>	051	)	111	I	151	i
012	LF	<CTRL-J>	052	*	112	J	152	j
013	VTAB	<CTRL-K>	053	+	113	K	153	k
014	FF	<CTRL-L>	054	,	114	L	154	l
015	CR	<CTRL-M>	055	-	115	M	155	m
016	SO	<CTRL-N>	056	.	116	N	156	n
017	SI	<CTRL-O>	057	/	117	O	157	o
020	DLE	<CTRL-P>	060	0	120	P	160	p
021	XON	<CTRL-Q>	061	1	121	Q	161	q
022	AUXON	<CTRL-R>	062	2	122	R	162	r
023	XOFF	<CTRL-S>	063	3	123	S	163	s
024	AUXOFF	<CTRL-T>	064	4	124	T	164	t
025	NAK	<CTRL-U>	065	5	125	U	165	u
026	SYN	<CTRL-V>	066	6	126	V	166	v
027	ETB	<CTRL-W>	067	7	127	W	167	w
030	CAN	<CTRL-X>	070	8	130	X	170	x
031	ENDMD	<CTRL-Y>	071	9	131	Y	171	y
032	SUB	<CTRL-Z>	072	:	132	Z	172	z
033	ESC	<CTRL-[>	073	;	133	[	173	{
034	F SEP	<CTRL-\>	074	<	134	\	174	|
035	G SEP	<CTRL-]>	075	=	135	]	175	}
036	R SEP	<CTRL-^>	076	>	136	^	176	~
037	U SEP	<CTRL-_>	077	?	137	_	177	DEL

Appendix D
ACRONYMS AND ABBREVIATIONS

ac	Accumulator
BIN	Byte Indicator
C-L	Current Loop
CARR	Carrier (Received Line Signal Detector)
CPU	Central Processing Unit (Computer)
CTS	Clear To Send
DIP	Dual-Inline Package
DMA	Direct Memory Access (Data Channel)
DSR	Data Set Ready
DTR	Data Terminal Ready
EIA	Electronic Industries Association
FDX	Full Duplex
FIFO	First-in, First-out
HDX	Half Duplex
IBP	Input Byte Pointer (Word 4 of PCB)
ICW	Input Control Word (Word 0 of PCB)
I.C.	Integrated Circuit
IRIS	Interactive Real-Time Information System
LIB	Last Input Byte (Word 6 of PCB)
LOB	Last Output Byte (Word 7 of PCB)
LSB	Least Significant Bit
MSB	Most Significant Bit
MUX	MIGHTY MUX DMA Multiplexer
OBP	Output Byte Pointer (Word 5 of PCB)
OCW	Output Control Word (Word 1 of PCB)
PCB	Port Control Block
PCON	Port Control
PROM	Programmable Read-Only Memory
RTC	Real Time Clock
RTI	Real-Time Interrupt
RTS	Request To Send
TTI	Teletype Input - refers to Master Terminal, device code 10
TTL	Transistor-Transistor Logic
TTO	Teletype Output - refers to Master Terminal, device code 11
UART	Universal Asynchronous Receiver-Transmitter

COMMENT SHEET

MANUAL TITLE MIGHTY MUX DMA Multiplexer User Manual

PUBLICATION NO. HM-042-0015 REVISION A

FROM: NAME/COMPANY: _____

BUSINESS ADDRESS: _____

CITY/STATE/ZIP: _____

COMMENTS: Your evaluation of this manual will be appreciated by POINT 4 Data Corporation. Notation of any errors, suggested additions or deletions, or general comments may be made below. Please include page number references where appropriate.

NO POSTAGE NECESSARY IF MAILED IN U.S.A.
Fold on Dotted Lines and Tape

NO POSTAGE
NECESSARY
IF MAILED IN
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 5755 SANTA ANA, CA.

POSTAGE WILL BE PAID BY ADDRESSEE:

POINT 4 Data Corporation
PUBLICATIONS DEPARTMENT
2569 McCabe Way
Irvine, CA 92714

CUT ON THIS LINE

POINT 4 DATA CORPORATION

2569 McCabe Way / Irvine, California 92714 / (714) 754-4114