UNIDEX™ XI AND UNIDEX IV

IEEE-488

USER'S MANUAL

(OPTION 2)

REV. 0-2.L

AUGUST, 1987

AEROTECH, INC., 101 Zeta Drive, Pittsburgh, PA 15238
(412) 963-7470 TWX 710-795-3125

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

DISCLAIMER

The information contained in this manual is subject to change due to improvements in design.

Though this document has been checked for inaccuracies, Aerotech does not assume responsibility for any errors contained herein.

CHAPTER 1: INSTALLATION

SECTION 1-1 INTRODUCTION

The IEEE-488 option for the Unidex XI or IV makes it possible to control Unidex XI or IV from a host computer via the IEEE-488 bus. A simple command sequence via the IEEE-488 interface gains control of the device. The host computer may from this point on perform different tasks using Unidex XI or IV. These tasks may include executing a command block in the immediate mode, downloading a program into the user memory, running a program from the memory in auto-run or block-run mode, reading the axis position values, reading the directory, reading a program, or reading the entire memory.

Unidex XI or IV may also be set up for interactive control. In this mode, Unidex XI or IV sets the service request after executing a program, a command block, or if there is an error condition. Subsequently, the host computer is required to serial poll Unidex XI or IV before proceeding.

Before continuing, it is recommended that the user review the Unidex XI or Unidex IV Operator's Manual (2 or 4 axis).

SECTION 1-2 REQUIRED HARDWARE

It is required that the option board OP4 be installed with the control board CB4 for the IEEE-488 option. The connector J15 on the rear panel board is the IEEE-488 interface connector (refer to figure 1-1). This is a standard 24 pin connector implementing the bus signal lines as discussed below.

B. IEEE-488 INTERFACE

IEEE-488 has 8 data lines and 8 control lines. It can accommodate up to 14 devices and provides a service request line from all devices to the controller. All of these properties lead to a more rapid form of communication between

Unidex XI or IV and the controller. You need not concern yourself with bus disciplines if your controller has IEEE-488 interface and device driver software that "hooks" into the Basic, Fortran, Pascal or whatever language you intend to use.

Different ways of connecting multiple devices to the IEEE-488 interface is shown in figure 1-2.

The devices connected to the bus have certain roles assigned to them. The roles represent the three basic functional elements necessary for effective communication. These three roles are:

1. Listener
2. Talker
3. Controller

1. Device As Listener

A "listener" is a device that has the capability of receiving data from the bus. It can be addressed by an interface message to listen. When addressed to listen, the listener will receive data placed on the bus.

2. Device As Talker

A "talker" is a device with the capability of sending data via the bus, when addressed by an interface message to talk.

3. Device As Controller

A "controller" is a device with the capability of controlling and directing the activity on the bus. A controller can address other devices to listen or to talk. It can also send interface messages to command specific actions from the other devices connected to the bus. You will need a device to act as a controller when implementing the IEEE-488 interface.

Listener, talker and controller capabilities can occur individually or in combinations. For instance, devices such as the Unidex XI, Unidex IV or a terminal can be implemented to talk or to listen, but not to control. Many computers, however, are capable of talking, listening and controlling.

4. Signal Lines of the IEEE-488 Bus

The IEEE-488 transfers data and commands between devices on 16 signal wires.

Eight of the lines are for the transfer of data (DIO1 to DIO8).

Data and message transfers are asynchronous and are coordinated by the three handshake lines.

The remaining five lines, for example "ATN" (attention) and "SRQ" (service request), are for bus management. Each line, when asserted low (ground), represents a single line message sent on the bus. A description of these lines is given in table 1-1.

5. Cable Restrictions of the IEEE-488 Bus

The devices in a system are connected together by a 24-wire cable using 24-pin connectors as specified in the IEEE-488 standard.

There are certain limitations on the length of the cables and the number of devices on the bus.

The maximum number of devices on the bus is limited to 14. The total length of the cable is limited to 20 meters or 2 meters multiplied by the number of devices (whichever is shorter in length). For a complete cable listing, refer to table 1-2.

6. Parallel And Serial Polling

Parallel polling is done to identify which device on the IEEE-488 bus is requesting service (SRQ). Serial polling is then done on the device requesting service in order to determine why.

a. Parallel Polling

When parallel polled by the controller, a device asserts a preassigned data line if it is requesting service.

The parallel poll bit assigned to each Unidex XI or IV may be selected via the front panel keyboard in the set-up mode, as described in section 2-1.

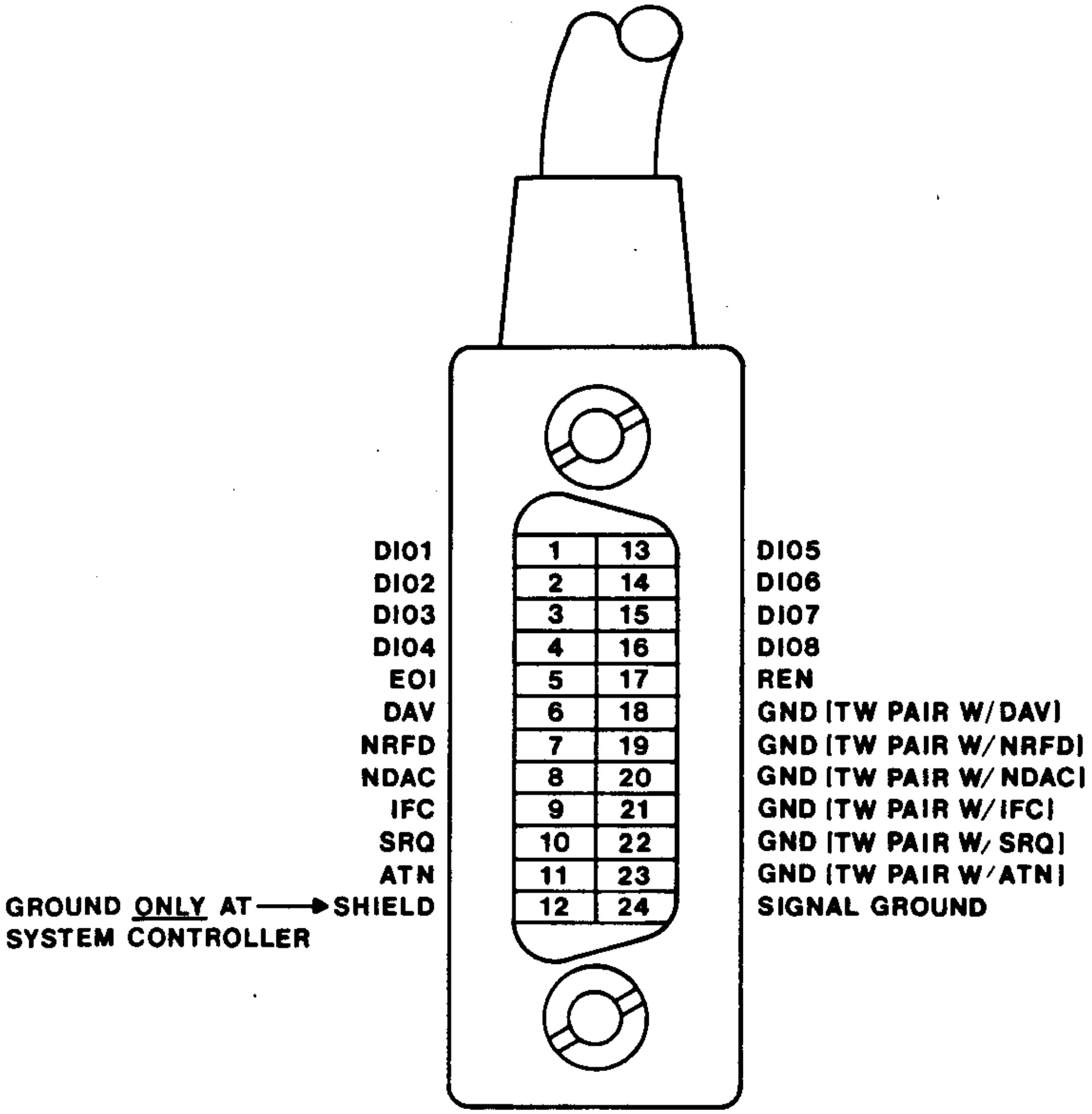
b. Serial Polling

In the serial poll, each of the devices requesting service is polled one at a time. You may serial poll any device at any time, regardless of the number of devices on the line.

A Unidex XI or Unidex IV will request service (set SRQ) at specific times, such as when a program is completely executed. At such a time, further operations will be suspended until Unidex XI or IV is serial polled by the controller. Upon being polled, the Unidex XI or IV will transmit its status.

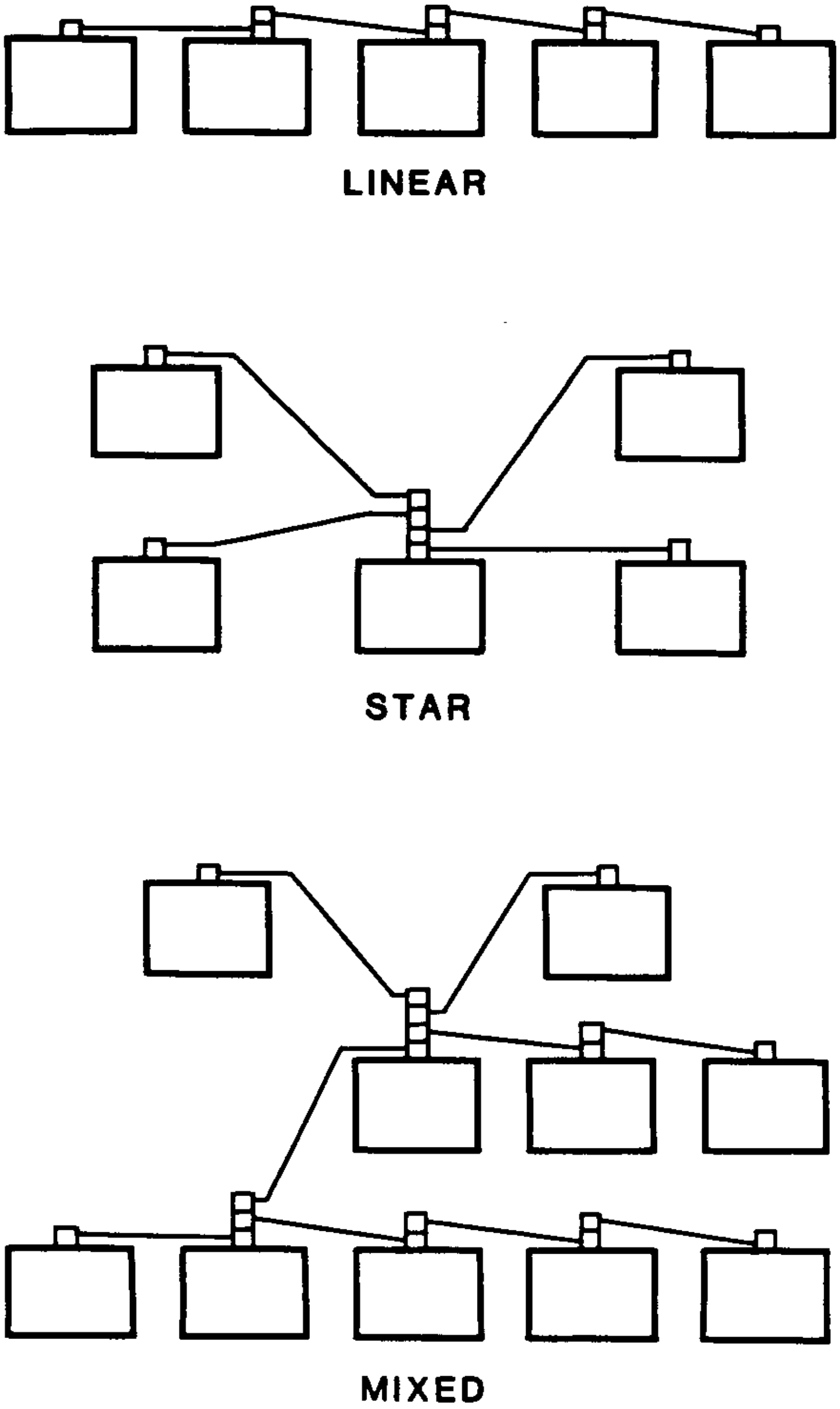
MNEMONIC	NAME	FUNCTION
		Bus Management Lines
IFC	INTERFACE CLEAR	System controller alone can assert this line, to place all devices in the unaddressed state. Devices go into talker idle/listener idle state. If control has been passed to another device, system controller again becomes active by asserting IFC.
ATN	ATTENTION	Asserted true by active controller to send bus interface messages on the bus. When ATN is asserted, signals on the data lines are interpreted as messages. ATN asserted with EOI to do a parallel poll. When ATN is false, data may be sent over the bus by a designated talker.
REN	REMOTE ENABLE	Asserted to program devices on the bus remotely. Any device addressed to listen while REN is true, is placed in remote mode of operation.
SRQ	SERVICE REQUEST	Asserted by a device to indicate its need for interaction with the controller.
EOI	END OR IDENTIFY	When asserted, indicates the termination of flow of data. Asserted when the last data byte is placed on the bus.
		DATA HANDSHAKE LINES
DAV	DATA VALID	Asserted by the talker to indicate to all listeners that data on the bus is valid.
NRFD	NOT READY FOR DATA	When true, indicates to talker that all listeners are not ready for data.
NDAC	NOT DATA ACCEPTED	When true, indicates to the talker that all listeners have not accepted the data placed on the bus
		DATA LINES
DI01-DI08	DATA LINES	Used for sending data (ATN lines false) or bus interface messages (ATN line true).

TABLE 1-1: IEEE-488 STANDARD INTERFACE BUS SIGNAL LINE



DIO1	1	13	DIO5
DIO2	2	14	DIO6
DIO3	3	15	DIO7
DIO4	4	16	DIO8
EOI	5	17	REN
DAV	6	18	GND (TW PAIR W/DAV)
NRFD	7	19	GND (TW PAIR W/NRFD)
NDAC	8	20	GND (TW PAIR W/NDAC)
IFC	9	21	GND (TW PAIR W/IFC)
SRQ	10	22	GND (TW PAIR W/SRQ)
ATN	11	23	GND (TW PAIR W/ATN)
GROUND <u>ONLY</u> AT → SHIELD SYSTEM CONTROLLER	12	24	SIGNAL GROUND

FIGURE 1-1: IEEE-488 PIN CONNECTIONS



LINEAR

STAR

MIXED

FIGURE 1-2: IEEE-488 CABLING CONFIGURATIONS

HEWLETT-PACKARD
Palo Alto, California 94304

PN	LENGTH
HP 10833D	.5 Meter
HP 10833A	1 Meter
HP 10833B	2 Meters
HP 10833C	4 Meters
HP 10834A	Adapter

Belden Corporation
Richmond, Indiana 47374

PN	Length
9642	1 Meter
9643	2 Meters
9644	4 Meters
9645	8 Meters
9646	16 Meters

TABLE 1-2: IEEE-488 CABLE MANUFACTURERS

CHAPTER 2: OPERATION OF UNIDEX XI or IV THROUGH IEEE-488 BUS

SECTION 2-1 SETTING UP THE IEEE-488 PARAMETERS

It is required that the device address of Unidex XI or IV be set to a number that the controller uses to address it prior to any data transfer. Device addresses that can be set vary from "00" to "15". To change the device address of Unidex XI or IV, put the system in the SET-UP mode as described in section 2-2, item K, of the Unidex XI or Unidex IV Operator's Manual.

Press SELECT until the OPTION BOARD set up screens are displayed. They will be displayed after the last axis' "Number of Digits after Point" screen. The display screen will show:

```
* * * * * * * * * * * * * * * * * * * * * *
*          Insert Block Nos. when          *
*          Printing programs : NO          *
* * * * * * * * * * * * * * * * * * * * * *
```

Press SELECT to see:

```
* * * * * * * * * * * * * * * * * * * * * *
*           IEEE-488  INTERFACE            *
*          Device  Address   : 02          *
* * * * * * * * * * * * * * * * * * * * * *
```

Now the +/- key may be pressed repeatedly until the address changes to the required value.

Press select again for the last set up screen related to the IEEE-488 interface.

```
* * * * * * * * * * * * * * * * * * * * * *
*           IEEE-488  INTERFACE            *
*         Par.  Poll  Resp.  : PPR1        *
* * * * * * * * * * * * * * * * * * * * * *
```

This screen sets up the identity byte for the device. Each device on the IEEE-488 bus may be assigned one data line that the device asserts when an IDENTIFY command is sent by the controller (when the controller parallel polls the device). The identity bytes and the corresponding PPRx set ups are shown below:

IEEE - 488 PARALLEL POLL RESPONSE

(SIGNAL ON THE DATA LINES)

	8	7	6	5	4	3	2	1
PPR 1	0	0	0	0	0	0	0	1
PPR 2	0	0	0	0	0	0	1	0
PPR 3	0	0	0	0	0	1	0	0
PPR 4	0	0	0	0	1	0	0	0
PPR 5	0	0	0	1	0	0	0	0
PPR 6	0	0	1	0	0	0	0	0
PPR 7	0	1	0	0	0	0	0	0
PPR 8	1	0	0	0	0	0	0	0

FIGURE 2-1: PARALLEL POLL RESPONSE

CHAPTER 3: PROGRAMMING UNIDEX XI or IV VIA IEEE-488 BUS

SECTION 3-1 COMMUNICATING WITH UNIDEX XI OR IV

Unidex XI or IV is ready for communication at power up if the device address has been correctly set up and the system has a battery back up or if the default values are the required format. Default values are:

```
    DEVICE ADDRESS      :   02
PARALLEL POLL RESPONSE  :  PPR1
```

If the keyboard is operated, the communication interface is disabled and Unidex XI or IV will not respond to commands on the IEEE-488 bus until the interface is enabled from the keyboard. To enable IEEE-488 interface, bring up the fifth screen of the main menu:

```
* * * * * * * * * * * * * * * * * * * * * * *
*     1. DIG.  OUT        3. COMM ENAB      *
*     2. ACL/DCL          4. PRINT          *
* * * * * * * * * * * * * * * * * * * * * * *
```

Pressing key #3 will take you to the following screen:

```
* * * * * * * * * * * * * * * * * * * * * * *
*           1.  RS-232/IEEE-488             *
*           2.  PARALLEL PORT               *
* * * * * * * * * * * * * * * * * * * * * * *
```

Press key #1 and Unidex XI or IV gets ready to communicate via IEEE-488.

```
* * * * * * * * * * * * * * * * * * * * * * *
*           IEEE-488 INTERFACE              *
*         COMMUNICATION ENABLED             *
* * * * * * * * * * * * * * * * * * * * * * *
```

The host device must now address Unidex XI or IV to listen. Most controllers provide the user with a high level language statement that can be executed in the controller to accomplish this. When addressed to listen, Unidex XI or IV or IV will display:

```
* * * * * * * * * * * * * * * * * * * * * * *
*            IEEE-488 INTERFACE               *
*          COMMUNICATION ACTIVE               *
* * * * * * * * * * * * * * * * * * * * * * *
```

SECTION 3-2 TYPES OF COMMANDS

A command sent to Unidex XI or IV via the IEEE-488 interface may be classified into two types:

1. **SYSTEM COMMANDS:** These are commands needed to interact with Unidex XI or IV as a device and perform operations such as resetting Unidex XI or IV, printing a program, printing position values, running a program, downloading a program, etc. Each system command establishes a mode of operation.

2. **PROGRAM (MOTION) COMMANDS:** These are the user program blocks in a motion program that Unidex XI or IV executes when running the program or in the immediate mode. Program commands are valid only in the immediate or edit mode.

SECTION 3-3 SERVICE REQUEST AND SERIAL POLL

Service request is an important concept in device control when there is a controller (host computer) as the master and the controlled device (such as a printer, Unidex XI or Unidex IV) as the slave. The purpose of Service Request is for the slave device to catch the attention of the master controller.

Typically, a controller has more than one device being controlled by it and it would be very inefficient for the master controller to continually read the statuses of all the devices to check for error states.

The controlled (slave) device therefore has the capacity to send a signal (service request) to the master controller whenever the attention of the master is required. The reason for requesting service may be an error condition or to signal the completion of a task.

Unidex XI or IV implements a Service Request by asserting the SRQ line on the IEEE-488 bus. The controller (master) may be set up to be interrupted by a Service Request and to take necessary action.

The minimum necessary action that the controller must take once Unidex XI or IV has sent the Service Request signal is to serial poll Unidex XI or IV.

Unidex XI or IV waits until it is serial polled by the controller and will not respond to any other system command until this is done. The purpose of the serial poll is to read a status byte from Unidex XI or IV.

The 8 bits of this status byte represent different internal states of Unidex XI or IV. Responding to a serial poll is a hardware function and therefore the controller may serial poll Unidex XI or IV at any time.

The status byte may be analyzed by the controller to determine the cause of the Service Request. Each of the bits in the status byte is described below.

SERIAL POLL STATUS BYTE

	ZERO	ONE
BIT 0	Incremental mode	Absolute mode
BIT 1	Not running a program	Running a program
BIT 2	Block run mode	Auto run mode
BIT 3	Non-corner rounding mode	Corner rounding mode
BIT 4	Communication disabled	Communication enabled
BIT 5	Not executing a command	Executing a command
BIT 6	No Service Request signal sent	Service Request signal sent; waiting for serial poll
BIT 7	No errors detected	Error detected

Unidex XI or IV may be put into the Service Request mode by the system command: **J <CR><LF>**

The Service Request mode may be cancelled by sending the system command **K <CR><LF>**. In this mode Unidex XI or IV will not send a service request signal for any reason. This is the default mode. In this mode, to determine if an immediate command or a program has been completed, a serial poll may be done, and the status byte analyzed (bit 1 and bit 5). When these bits are clear (zero), Unidex XI or IV is ready to take the next command.

A. CONDITIONS FOR SERVICE REQUEST

When in the service request mode, Unidex XI or IV sets a service request (SRQ) under the following conditions:

1. When an Immediate command execution is complete.
2. When a program is completely executed in the Auto Run mode.
3. When a block is executed in the Block Run mode.
4. When a run time error condition is generated and the program is aborted (refer to chapter 4 for error code definitions).
5. When an axis limit is activated.
6. When a program or immediate command move is stopped by pressing the STOP key on the front panel keypad.
7. At the end of a program download operation if an error was generated while downloading. (The SRQ is set by Unidex XI or IV after the "%" that ends the downloading of the program.)
8. When it is requested that a nonexistent program be printed. (If "Pnn <CR><LF>" is sent to Unidex XI or IV and program #nn does not exist, Unidex XI or IV will set the SRQ line.)

NOTE: For more information on error bytes see chapter 4.

NOTE: For more information on IEEE-488, refer to the "ANSI/IEEE Std 488-1978" standard, published by:

The Institute of Electrical and Electronics
Engineers, Inc.
345 East 47th Street
New York, NY 10017

CHAPTER 4: ERROR CODES

An error condition may be detected by the host computer by checking the most significant bit (bit 7) of the serial poll status byte (section 3-3). If this bit is set to "1", an error has occurred.

To further determine the type of error, the system status may be read by the host by sending the command "PS <CR><LF>". Unidex XI or IV will send back 13 bytes followed by <CR><LF>. These 13 bytes represent a complete status report of Unidex XI or IV. They are described as follows:

BYTE 1 Same as Serial Poll status byte described in section 3-3.

BYTE 2 **EDITOR ERROR STATUS**

		ZERO	**ONE**
Bit 0	-	No illegal character during download	Illegal character during download (illegal command code)
Bit 1	-	Memory not full during download	Memory full during download
Bit 2	-	No user memory checksum error	Checksum error during download of program
Bit 3	-	No illegal command	Illegal command during download (illegal command code)
Bit 4 ¦¦ Bit 7		Not used	

NOTE: If one of these errors is generated during download operation, Unidex XI or IV will assert SRQ (service request), if in the Service Request mode. It is recommended that the user then edit and correct that program.

BYTE 3 RUNTIME ERROR STATUS 1

	ZERO	ONE
Bit 0 * -	X axis not in limit	X axis in limit
Bit 1 * -	Y axis not in limit	Y axis in limit
Bit 2 * -	U axis not in limit	U axis in limit
Bit 3 * -	V axis not in limit	V axis in limit
Bit 4 * -	No illegal byte in memory	Illegal byte in memory
Bit 5 * -	Program number valid	Invalid program called out for run
Bit 6 * -	Memory not clear	No programs in memory (Memory clear)
Bit 7 * -	No user memory check sum error	User memory check sum error

BYTE 4 RUNTIME ERROR STATUS 2

	ZERO	ONE
Bit 0 -	Stop key not pressed	Stop key pressed to stop motion/program
Bit 1 * -	No "Repeat Loop End Invalid" error	"Repeat Loop End" invalid
Bit 2 * -	No "Repeat Loop Incomplete" error	Repeat loop incomplete
Bit 3 * -	Eight repeat loops not exceeded	Eight repeat loops exceeded
Bit 4 * -	No "Return From Subroutine Invalid" error	"Return From Subroutine" invalid

Bit 5 * -	No "Incomplete subroutine" error	Incomplete subroutine
Bit 6 * -	Eight subroutines not exceeded	Eight subroutines exceeded
Bit 7 * -	No "Missing Label" error	Missing label

* FOR ADDITIONAL INFORMATION, SEE CHAPTER 5 OF THE UNIDEX XI OR UNIDEX IV OPERATOR'S MANUAL.

BYTE 5 COMMUNICATION STATUS 1

	ZERO	ONE
Bit 0 -	No RS-232 hardware on OP4 board	RS-232 hardware on OP4 board
Bit 1 -	No IEEE-488 hardware on OP4 board	IEEE-488 hardware on OP4 board
Bit 2 -	RS-232 communication not active	RS-232 communication active
Bit 3 -	IEEE-488 communication not active	IEEE-488 communication active
Bit 4 -	">" or "#" not received	">" or "#" received
Bit 5 -	Not used	
Bit 6 -	Not in SRQ mode	In SRQ mode
Bit 7 -	Not in Hold mode	In Hold mode

BYTE 6 COMMUNICATION STATUS 2

	ZERO	ONE
Bit 0 -	LCD Display 1 included in system (X and Y display present)	LCD Display 1 not included in system
Bit 1 -	LCD Display 2 included in system (U and V display present)	LCD Display 2 not included in system
Bit 2 -	In Hold mode, but no Trigger command received	In Hold mode, and Trigger command received
Bit 3 -	Unidex XI or IV "receive buffer" not full	Unidex XI or IV "receive buffer" full
Bit 4 -	Xon received during transmit	Xoff received during transmit
Bit 5 -	Not in program download mode	In program download mode
Bit 6 -	Status bytes printed in binary format	Status bytes printed in hex-ASCII format
Bit 7 -	I/O parallel port input not enabled	I/O parallel port input enabled (option 5)

BYTE 7 AXIS VALIDITY STATUS

	ZERO	ONE
Bit 0 -	X axis not in system	X axis in system
Bit 1 -	Y axis not in system	Y axis in system
Bit 2 -	U axis not in system	U axis in system
Bit 3 -	V axis not in system	V axis in system

Bit 4
¦¦ - Not used
Bit 7

BYTE 8 RAMPER BOARD STATUS

		ZERO	ONE
Bit 0	-	X axis does not have ramping	X axis has ramping (through 1MR)
Bit 1	-	Y axis does not have ramping	Y axis has ramping (through 1MR)
Bit 2	-	U axis does not have ramping	Y axis has ramping (through 1MR)
Bit 3	-	V axis does not have ramping	V axis has ramping (through 1MR)
Bit 4 ¦¦ Bit 7	-	Not used	

BYTE 9 AXIS MOTION STATUS

		ZERO	ONE
Bit 0	-	X axis not moving	X axis moving
Bit 1	-	Y axis not moving	Y axis moving
Bit 2	-	U axis not moving	U axis moving
Bit 3	-	V axis not moving	V axis moving
Bit 4 ¦¦ Bit 7	-	Not used	

BYTE 10 FREE RUN MODE STATUS

		ZERO	ONE
Bit 0	-	X axis not in free run mode	X axis in free run mode
Bit 1	-	Y axis not in free run mode	Y axis in free run mode
Bit 2	-	U axis not in free run mode	U axis in free run mode
Bit 3	-	V axis not in free run mode	V axis in free run mode
Bit 4 ¦¦ Bit 7	-	Not used	

BYTE 11 I/O STATUS

		ZERO	ONE
Bit 0	-	Input 1 is a 0 (I1)	Input 1 is a 1 (I1)
Bit 1	-	Input 2 is a 0 (I2)	Input 2 is a 1 (I2)
Bit 2	-	Input 3 is a 0 (I3)	Input 3 is a 1 (I3)
Bit 3	-	Input 4 is a 0 (I4)	Input 4 is a 1 (I4)
Bit 4	-	Output 1 is a 0 (O1)	Output 1 is a 1 (O1)
Bit 5	-	Output 2 is a 0 (O2)	Output 2 is a 1 (O2)
Bit 6	-	Output 3 is a 0 (O3)	Output 3 is a 1 (O3)
Bit 7	-	Output 4 is a 0 (O4)	Output 4 is a 1 (O4)

BYTE 12 AND BYTE 13 DIGITAL OUTPUT STATUS (OPTION 5)

Bit 0
 ¦¦   - 12 bit value of digital output
Bit 11

Bit 12
 ¦¦   - Not used
Bit 15

CHAPTER 5: TYPES OF COMMANDS

SECTION 5-1 SYSTEM COMMANDS

System commands interact with Unidex XI or IV as a device and perform operations such as resetting it, printing a program, printing position values, running a program, downloading a program, transferring status byte information from Unidex XI or IV, etc. Each system command establishes a mode of operation once it is received by Unidex XI or IV. Each system command must be entered as a capital letter.

A. GETTING UNIDEX XI OR IV'S ATTENTION

To activate the IEEE-488 interface in Unidex XI or IV, it must be addressed to listen. The controller may do this by asserting the ATN and REN line and sending the appropriate Listen Address Code. Generally, the user need not be concerned with the low level bus operations. Most controllers provide means to communicate on the IEEE-488 bus from a high level language via familiar statements such as OUTPUT, PRINT, ENTER, READ, INPUT, etc.

B. AUTO MODE (A)

Executing a program in the auto mode enables the program to run automatically, executing the motion commands of the program with no need of further user intervention.

To run a program in the auto mode, send "A", the program number ("nn") and <CR><LF>. Example:

A 10 <CR><LF>

If in service request mode (see section 3-3), once the program has been executed, Unidex XI or IV will set the service request and wait for a serial poll. After the serial poll, you may execute the same program again by sending another <CR><LF>. To run a different program, send "A nn <CR><LF>" again.

C. BLOCK MODE (B)

A motion program can be run one block at a time, instead of automatically as discussed in the above subsection. To run a program in the block mode, send "B" for block, the program number ("nn") and a <CR><LF>. Example:

B 10 <CR><LF>

If in service request mode, Unidex XI or IV will set SRQ after each block has been executed. If this is the case, the host must serial poll Unidex XI or IV after the execution of each block.

After the execution of the first command block and the serial poll, send a <CR><LF> to execute the next block. Bit 1 of the status byte (section 3-3) may be checked to detect completion of the program. This bit is cleared after the last block in the program.

D. REMOTE RESET (C)

Sending the command "C" followed by <CR><LF> resets Unidex XI or IV. This will take it back to power up conditions. Example:

C <CR><LF>

The IEEE-488 bus functions "Device Clear" (DCL) and "Selected Device Clear" (SDC) also effect the same response from Unidex XI or IV.

E. DISABLING (D) JOYSTICK MODE OR REMOTE MODE (AVAILABLE WITH OPTION 7 ONLY)

The system command:

D <CR><LF>

will do one of the following:

1. Disable the Computer Enabled Joystick Mode and return control to the host controller. (The position registers will be updated with the absolute position values before returning control.)

2. Disable the Remote Mode and return control to the host. (The position registers are updated with the absolute position values before returning control.)

When either the Computer Enabled Joystick Mode or Remote Mode is active, Unidex XI or IV will only recognize the "D" command and the Serial Poll command. All other system commands will be ignored.

F. DOWNLOADING A PROGRAM TO UNIDEX XI or IV FROM HOST (E)

The "E" command, followed by a program number ("nn") and an end-of-block character (* or /), will put Unidex XI or IV into the edit mode and set it up to enter the program commands into program "nn" in the user memory. If an existing program with the same number already resides in Unidex XI or IV, it will be deleted automatically when the new program "nn" is downloaded. Example:

```
E10 * HXY * XF100D1000YF1000D-2000 * %
```

In the above example, the commands following "E10*" will be downloaded into the Unidex XI or IV user memory and stored in program #10. The motion commands that may be included in this program will be discussed in section 5-2, Program Commands.

G. DELETING A PROGRAM (E$)

In order to delete a program from the Unidex XI or IV user memory, send the command "E", followed by the character "$", the program number "nn" and an end-of-block character, either "*" or "/". Example:

E $ 10 / (or *)

Program 10 will be erased.

H. DELETING ALL PROGRAMS (USER MEMORY) (E$)

In order to delete <u>all</u> programs from the Unidex XI or IV user memory, send the command "E", followed by the character "$", two zeros ("00") and an end-of-block character, either "*" or "/". Example:

E $ 00 /

All programs will be erased.

I. BLOCK NUMBERING (F)

If you want programs to be printed with block numbers, send the "F" command and a <CR><LF>. Block numbering may make editing the program easier. Example:

F <CR><LF>

After this command is sent to Unidex XI or IV, any programs will be printed with block numbers.

J. BLOCK NUMBERING CANCEL (G)

In order to cancel block numbering when a program is printed, as established in the above subsection, send the command "G" along with <CR><LF>. Example:

G <CR><LF>

After this command has been sent to Unidex XI or IV, programs will be printed without block numbering.

These system commands F and G do not change the system set up feature stored in the battery back up memory.

K. HOLD (H)

The command to "hold" the execution of a command string or an entire program is established by the "H" command and a <CR><LF>. Example:

H <CR><LF>

The above command will cause Unidex XI or IV to suspend execution of any Immediate, Auto or Block commands which may follow it. This is useful if synchronization of axis motion with some other action is necessary. Unidex XI or IV will only execute the commands when it receives a "T" (for Trigger) command or a "Group Execute Trigger"

(GET) bus message (discussed in a following subsection). Example:

```
H <CR><LF>
A 20 <CR><LF> Program #20 not executed (held)
T <CR><LF>    Now Program #20 executes
   OR
(GET) bus interface message
```

L. IMMEDIATE MODE (I)

The "I" command, followed by motion program commands, an end-of-block character (* or /) and a <CR><LF>, allows a block of motion commands to be executed immediately instead of being entered as a motion program. Each block of immediate commands must begin with an I. For example:

```
I X F10000 D20000 * <CR><LF>
```

The above immediate command will send the X axis 20000 steps at a feedrate of 10000 steps per second (or whatever units might be set in the system). If in SRQ mode, Unidex XI or IV will send a service request and wait for a serial poll after the command is executed. After being polled, Unidex XI or IV is ready to execute another block of commands.

M. SERVICE REQUEST (SRQ) SET UP (J)

In order to establish the service request mode, send the "J" command, followed by <CR><LF>. After the SRQ mode has been established via the J command, Unidex XI or IV will assert the SRQ line under conditions described in section 3-3. It will then wait until it is serial polled by the controller before executing any further commands. (For more detailed information on SRQ, see section 3-3).

N. SERVICE REQUEST (SRQ) CANCEL (K)

In order to cancel the service request (SRQ) mode established by the J command (above subsection), send a K command, followed by <CR><LF>. Example:

K <CR><LF>

SRQ cancelled is the default status.

O. STATUS BYTE IN BINARY FORMAT (M)

To establish the format of the status bytes as binary upon transmission, sent command "M", followed by <CR><LF>. Example:

M <CR><LF>

Transmission of the status bytes in binary format is the default status.

P. STATUS BYTE IN HEX-ASCII FORMAT (N)

To establish the status bytes in the Hex-ASCII format upon transmission, send the command "N", followed by <CR><LF>. Example:

N <CR><LF>

Q. PRINT AXIS POSITION

1. Print X Axis Position (PX)

In order to print the X axis position register, send:

PX <CR><LF>

When Unidex XI or IV is addressed to talk, the axis position is sent in the following format:

<space> or <negative sign> <10 digits> <CR><LF>

2. Print Y Axis Position (PY)

In order to print the Y axis position register, send:

PY <CR><LF>
(and address Unidex XI or IV to talk)

3. Print U Axis Position (PU)

In order to print the U axis position register, send:

PU <CR><LF>
(and address Unidex XI or IV to talk)

4. Print V Axis Position (PV)

In order to print the V axis position register, send:

PV <CR><LF>
(and address Unidex XI or IV to talk)

R. PRINTING DIRECTORY LISTING (PD)

To get a listing of the programs in the Unidex XI or IV directory send:

PD <CR><LF>
(and address Unidex XI or IV to talk)

Bytes of memory remaining in Unidex XI or IV will be printed as well.

S. PRINTING A PROGRAM (Pnn)

To have one program printed, send the command "P", the program number ("nn") and <CR><LF>. Example:

P10 <CR><LF>
(and address Unidex XI or IV to talk)

The above command will cause program #10 to be printed out.

T. PRINTING ALL PROGRAMS

To have all programs printed, send the command "P", two zeros (00) and <CR><LF>. Example:

P00 <CR><LF>
(and address Unidex XI or IV to talk)

The above command will cause all programs in memory to be printed out.

U. PRINTING STATUS BYTES (PS)

To have the status bytes listed in chapter 4 printed out, send:

PS <CR><LF>
(and address Unidex XI or IV to talk)

V. REMOTE MODE (R)
(AVAILABLE WITH OPTION 7 ONLY)

The system command:

R <CR><LF>

will enable Unidex XI or IV to be driven via the auxiliary controls. The display shows the remote mode tracking screen:

```
* * * * * * * * * * * * * * * * * * * *
*    rtm    X: 0000023456  step       *
*    rt     Y:-0002345.12   mm        *
* * * * * * * * * * * * * * * * * * * *
```

The host controller may now signal an external device to take control of Unidex XI or IV.

Unidex XI or IV keeps track of the axes' positions during external control.

W. ENABLING JOYSTICK (S) (AVAILABLE WITH OPTION 7 ONLY)

The system command:

S <CR><LF>

puts Unidex XI or IV into the joystick mode. The display changes to the joystick mode tracking display (see the Unidex XI Operator's Manual or the Unidex IV Operator's Manual, section 2-2H).

In a system with more than 2 axes, the initially active axes will be X and Y. The joystick may now be operated to move the axes.

X. TRIGGER (T)

To execute the program that is suspended with a Hold command (H), send:

T <CR><LF>
(or execute the bus function "Group Execute Trigger" **(GET)**)

SECTION 5-2 MOTION PROGRAM COMMANDS

The motion program commands are the user program blocks in a motion program that Unidex XI or IV executes when running a program in the auto or block mode. Program commands are valid only if entered in the immediate or edit mode.

A. END OF BLOCK (* OR /)

An end-of-block terminates a block of a program. It may be one of two characters:

*** or /**

If two or more axes are to run simultaneously, keep those axes commands within one block, i.e., place an end-of-block character after the axes moves have been entered. For example:

```
X F10000 D150000
Y F10000 D150000
V F500 D10000 * (or /)
```

B. AXIS MOTION COMMANDS (X,Y,U,V,F,D,R)

The axis to move must, of course, be specified by an axis command (X, Y, U or V).

The speed with which it travels must be specified by a feedrate command (F).

The distance which it is to travel (or the position it is to attain if in the Absolute Mode) is specified with a distance command (D).

An example of a program block utilizing the above commands is:

```
X F10000 D150000
```

The above command would send the X axis a distance of 150000 system units at a feedrate of 10000 system units/second.

NOTE: When programming via IEEE-488 communication, the feedrate need only be entered with the first index block. After that the feedrate must only be entered if it is to be changed.

1. Axis Free-Run

The axes are commanded to free-run by the command R and a "+" or "-" sign to signify CW (+) or CCW (-). Example:

```
Y F10000 R+ *
```

The above command tells the Y axis to free-run in the CW direction at a speed of 10000 system units/second.

C. DWELL (DW)

A program dwell can be entered into your program with a DW command, followed by the duration of the dwell in tenths of seconds. Example:

```
DW 10 *    or    DW 1.0 *
```

The above command will cause a 1 second dwell within your program.

D. HOME (H)

Send an axis Home with command H followed by the axis or axes to be sent home and an end-of-block character. Example:

```
H X *
  or
H XYUV *
```

The first example will send X axis home. The second command will send all axes home.

E. OUTPUT STATUS (OT)

To establish the output as a 1 (true), a 0 (false) or an X (don't care), send the command "OT" followed by a 1, 0 or X. Example:

OT 10XX *

In the above example, O1 is to be programmed to a 1, O2 to a 0, O3 is a "don't care" and O4 is also a "don't care". "Don't care" leaves the status of an output unchanged.

F. INPUT STATUS (IT)

To set up the status you wish the inputs to attain before the program continues, program an "IT" command, followed by the desired statuses and an end-of-block character. Example:

IT X001 *

The above command states that the program should wait until I2 is a 0, I3 is a 0 and I4 is a 1. State of Input I1 does not matter since it is programmed as a "don't care".

G. OUT/STOP STATE (OS)

To put out values to the outputs when the program is stopped via the STOP key, a feedhold or pressing key #3 on the front panel, enter the command "OS" followed by the desired values and an end-of-block character. Example:

OS 0011 *

When the program is stopped, a zero will be output on O1 and O2, a one will be output on O3 and O4.

H. OUT/RUN (OR)

To output values when the program is allowed to run again by pressing RUN or releasing the feedhold, (after the STOP key, key #3, or feedhold has been pressed), program "OR" followed by the desired values and an end-of-block. Example:

OR XXXX *

In the above example, when the program run is allowed to continue, the outputs will remain unchanged (because "don't care" has been programmed for each output).

I. REPEAT LOOP START (RS)

The command to start a repeat loop in your program and the number of times the loop executes is established with a "RS" command, followed by the number of times to repeat and an end-of-block. Example:

RS 8 *

The repeat loop beginning has been marked, and the loop is to be repeated 8 times.

J. REPEAT LOOP END (RE)

To mark the end of the repeat loop started as discussed in the above subsection, program command "RE" followed by an end-of-block. Example:

RE *

The repeat loop may be nested 8 levels deep.

K. CONDITIONAL REPEAT LOOP END (RC)

To end the repeat loop based on input conditions, program "RC" followed by the required input state and an end-of-block. Example:

RC 10XX *

The above example states that the repeat loop will end when the inputs are as follows: I1 is a 1, I2 is a 0. I3 and I4 have no control over the above program block.

L. STARTING FREE RUN AFTER A STOP FREE RUN COMMAND (R)

After a free run has been stopped (discussed in the next subsection), programming an "R" followed by the axis or axes to be restarted, will start the axis or axes again. Example:

R X *
or
R XYUV *

The first example restarts the X axis free run. The second example restarts all axes free run.

M. STOP AXES FREE RUN (S)

To stop an axis or axes' free run, program an "S" command, followed by those axes that are to be stopped, and an end-of-block character. Example:

```
  S X *
   or
  S XY *
   or
 S XYUV *
```

The first example stops the X axis free run. The second stops X and Y axes free run. The third stops all axes free run.

N. REPEAT PROGRAM (RP)

To repeat the entire program from the start, enter command "RP" and an end-of-block. Example:

```
RP *
```

Remember, any commands following this command within your program will not be executed.

O. LOAD POSITION REGISTERS (L)

You may load any of the axes position registers with an "L" command, followed by a distance and an end-of-block. The value, which is in system units, may be a positive or negative number or may be a zero. Example:

```
LX0Y0
```

In the above example, the X and Y position registers are loaded with zeros. This command may be used to establish an absolute reference position. The axes may be moved to this reference position by programming, in the absolute mode, an Index block with distance values equal to the reference position.

P. INCREMENTAL MODE (IN)/ABSOLUTE MODE (AB)

In the incremental mode, a distance command tells Unidex XI or IV how much further to move the axes. For example:

```
X F10000 D1000
```

This command would move the X axis 1000 steps in the CW direction each time it was executed. Program "IN", followed by an end-of-block, to establish the incremental mode. Example:

```
IN *
```

In the absolute mode, on the other hand, a distance command is an absolute position. For example:

```
X F10000 D1000
```

When in the absolute mode, the above example tells Unidex XI or IV to send the X axis to the position 1000. Once there, the reexecution of the above command will not move the axis any further since it is already at the position commanded. To establish the absolute mode of programming, enter:

```
AB *
```

Q. BEEPER (BN/BF)

To turn the beeper ON, program "BN" followed by end-of-block. To turn it OFF, program "BF" followed by end-of-block. Example:

```
 BN *
DW .5
 BF *
```

The above example turns on the beeper for 5/10 of a second and then turns it off.

R. LABEL (LB)

A label (0 to 99) labels a block of program as the place to which the program goes when a GOSUB (go to subroutine) or a GOTO command is encountered. It is programmed with a "LB" command, followed by the number and an end-of-block.

```
LB55*
```

S. GOTO (GT)

This command directs program flow to a label. Enter "GT" and a label number, followed by an end-of-block. Example:

```
GT 20 *
```

The above command tells Unidex XI or IV to continue program execution at label 20.

T. GOSUB (GS)

The command that tells Unidex XI or IV to execute a subroutine at label #nn is the "GS" command, followed by the block label number and an end-of-block. Example:

GS 15 *

The subroutine to be executed is located at label #15.

U. SUBROUTINE RETURN (SR)

This command causes Unidex XI or IV to return from the subroutine execution to the program block immediately after the "GS nn" block that called the subroutine. Every subroutine should end with a Return from Subroutine.

SR *

Subroutines may be nested 8 levels deep

V. PROGRAM STOP (PS)

Program stop marks the place in the program at which program execution ends. Enter:

PS *

Subroutines may be placed after this block.

W. CONDITIONAL GOTO (CT)

The command "CT", followed by a label number, an "I" and four input values, states that the program should go to a specific block if the input statuses match the specified values. For example:

```
CT 22 I 10X0 *
```

The above command tells Unidex XI or IV to go to the program block labeled "22" when I1 is 1, I2 is 0, and I4 is 0. If the inputs are not these values, continue with the next program block.

X. CONDITIONAL GOSUB (CS)

The command to send the program to a subroutine if the input statuses match specified values is "CS", followed by the label number, an "I" and the desired input values. For example:

```
CS 33 I 110X
```

The above command tells Unidex XI or IV to go to the subroutine labeled "33" when the value of I1 is 1, I2 is 1, I3 is 0. If these input conditions do not exist, continue with the next program block.

Conditional subroutines, in combination with regular subroutines, may be nested 8 levels deep.

Y. CORNER ROUNDING (CO)/NON-CORNER ROUNDING (NC)

The command to implement corner rounding is "CO" followed by an end-of-block. Example:

CO *

The above example would enable corner rounding for all four axes.

The command to go back to non-corner rounding is "NC" followed by an end-of-block. Example:

NC *

The above command would take all four axes out of the corner rounding mode.

Z. ACCELERATION/DECELERATION RAMP TIME (AD)

The accel/decel ramp time can be programmed in milliseconds by the command "AD" followed by the desired time and an end-of-block character. Example:

AD 10 * or AD 5821 *

In the above example, the acceleration ramp time as well as the deceleration ramp time will be 10 milliseconds. The maximum value is 9999. This command will be executed only when the required option is included in the system.

AA. BINARY DIGITAL OUTPUT (DD)/BCD DIGITAL OUTPUT (DB)

To program the Digital Output to be a binary number, enter command "DD" followed by the number ("nnnn") to be output and an end-of-block. Note: (nnnn <= 4095). The binary equivalent of the number you enter will be output. Example:

DD 22 *

In the above example the binary equivalent of the number 22 (000000010110) will be output on the 12 output lines.

To program the Digital Output to be a BCD number, enter command "DB", followed by number ("nnn") to be output and an end-of-block. Note: (nnn <= 999). Example:

DB 22 *

The number 22 will be output in BCD format (0000 0010 0010). A mSec strobe will be output on the strobe line after the 12 bit value settles.

BB. END EDIT (%)

The character that is placed at the end of a downloadable file is the percent sign (%). Just place it at the end of your file without an end-of-block. Example:

%

When downloading a program, this character take Unidex XI or IV out of the edit mode and back to the system command mode.

SECTION 5-3 SAMPLE IEEE-488 COMMANDS

A few brief samples will help demonstrate the commands discussed in the last section.

SAMPLE IMMEDIATE COMMANDS

```
Address Unidex XI or IV to listen   ; Interface active
I H XY * <CR><LF>                  ; Send home X and Y
I X F10000 D10000 * <CR><LF>       ; Move X axis
I Y F10000 D10000 * <CR><LF>       ; Move Y axis
I BN * <CR><LF>                    ; Beeper ON
I BF * <CR><LF>                    ; Beeper OFF
I X F100 D1000 Y F100 D1000 * <CR><LF> ; X and Y axes move
```

SAMPLE MOTION PROGRAM

```
Address Unidex XI or IV to listen   ; Interface active
E 01 *                             ; Edit program 1
H XY *                             ; Send X and Y axes home
X F10000 D10000 *                  ; Move X axis
Y F10000 D10000 *                  ; Move Y axis
BN *                               ; Beeper ON
DW .2 *                            ; Dwell for 2/10 second
BF *                               ; Beeper OFF
%                                  ; End edit mode
```

Send F <CR><LF> to set Unidex XI or IV to the block number printing mode.

P01 <CR><LF>

The above command will cause program #1 to be printed with block numbers when Unidex XI or IV is addressed to talk.

Cancel numbering with a G command.

Program #1 may be run block by block by sending:

B01 <CR><LF>

and successive <CR><LF> for successive blocks.

To run this same program in the Auto Mode, send:

A01 <CR><LF>

It may be deleted by sending:

E $ 01 *

Check your directory with a **PD** command to verify that program #1 has been deleted.

The X position register may be read by sending:

PX <CR><LF>
(and addressing Unidex XI or IV to talk)

The Y position register may be read by sending:

PY <CR><LF>
(and addressing Unidex XI or IV to talk)

Send J <CR><LF> to put Unidex XI or IV in the Service Request Mode. When Unidex XI or IV asserts SRQ, perform a serial poll before continuing. If running the program in the auto mode, the service request will come after the program execution. If running a program one block at a time, it will follow each block.

Send command K <CR><LF> to cancel the service request mode.

APPENDIX 1

COMMAND SUMMARY FOR UNIDEX XI AND UNIDEX IV

APPENDIX 1

COMMAND SUMMARY
FOR UNIDEX XI OR IV : IEEE-488

NOTE: All system commands must be entered as upper case letters.

The following is a list of commands for the Unidex XI or IV when operating via the IEEE-488 communication bus in the system command mode.

SYSTEM COMMANDS

A nn <CR><LF> : Run program # nn in AUTO mode (nn = 0 to 99).

B nn <CR><LF> : Run program # nn in BLOCK mode (subsequent <CR><LF> will execute successive program blocks).

C <CR><LF> : Reset Unidex XI or IV.

D : Cancel S or R command.

E nn * : Begin downloading program #nn. Existing program #nn will get deleted automatically. Program blocks described in following section, User Motion Program Commands.

E $ nn * : Delete program # nn.

E $ 00 * : Clear program memory (all programs).

F <CR><LF> : Insert block numbers when printing programs.

G <CR><LF> : Cancel block number printing (default state)

H <CR><LF> : Put Unidex XI or IV in HOLD mode (Trigger required to execute programs). Hold mode cancelled by **O <CR><LF>**.

I (string) * <CR><LF> : Execute program block (string) in the immediate mode.

J <CR><LF> : Set up Unidex XI or IV to assert Service Request after execution.

K <CR><LF> : Cancel set up to send SRQ (default state)

L <CR><LF> : Not used

M <CR><LF> : Set up to transmit status in binary format (default state)

N <CR><LF> : Set up to transmit status in Hex-ASCII format

O : Cancel HOLD mode (default state)

PX <CR><LF> : Print X axis position register value when addressed to talk

PY <CR><LF> : Print Y axis position register value when addressed to talk

PU <CR><LF> : Print U axis position register value when addressed to talk

PV <CR><LF> : Print V axis position register value when addressed to talk

PD <CR><LF> : Print Directory listing when addressed to talk

Pnn <CR><LF>: Print program #nn when addressed to talk

P00 <CR><LF>: Print all programs in memory when addressed to talk

PS <CR><LF> : Print Status bytes when addressed to talk

Q : Not used

R : Enable Remote Mode from host controller

S : Enable Joystick Mode from host controller

T : Trigger to start program execution

APPENDIX 1

MOTION PROGRAM COMMANDS

* or /	: End of block (terminates block)
X F ffffff D ddddddddd	: X axis move at feedrate ffffff steps/sec a distance of ddddddddd steps, CW
Y F ffff R +	: Y axis feedrate ffff steps/sec free-run CW
U F fffff R -	: U axis feedrate fffff steps/sec free-run CCW
V F ff D - dddd	: V axis move at feedrate ff steps/ sec a distance of dddd steps, CCW

(* placed as required)

DW nnn.n *	: Dwell nnn.n seconds
H X *	: X axis home
H Y *	: Y axis home
H XYUV *	: All four axes home
OT 10XX *	: Output O1, O2, O3, O4 1: True 0: False X: Don't care
IT X001 *	: Wait until input state I1, I2, I3, I4 matches X001
OS 0011 *	: OUT/STOP state
OR XXXX *	: OUT/RUN state
RS nnnn *	: Repeat loop start nnnn times
RE *	: Repeat loop end
RC 10X0 *	: End Repeat loop on input condition 10X0
R X *	: Start free run axis X after Stop (S X)
R YUV *	: Start free run axes Y, U and V
S X *	: Stop free run axis X
S YUV *	: Stop free run axes Y, U and V
RP *	: Repeat Program
L X ddddddddd	: Load position register X with ddddddddd
L Y ddd	: Load Y axis position register with ddd
L U 0	: Load U axis register with 0 (zero).
L V -dddd	: Load V axis position register with -dddd

(* placed as required)

IN *	: Incremental mode
AB *	: Absolute mode
BF *	: Beeper OFF
BN *	: Beeper ON
LB nn *	: Label # nn
GT nn *	: GoTo label #nn
GS nn *	: GoSub label #nn
SR *	: Subroutine Return

APPENDIX 1

PS *	: Program Stop (end of program execution)
CT nn I 10X0 *	: Conditionally GoTo label # nn if input state is 10X0, else continue
CS nn I 1XXX *	: Conditionally GoSub label #nn if input state is 1XXX, else continue
CO *	: Corner rounding mode
NC *	: Non-corner rounding mode
AD nnnn *	: Accel/Decel ramp time in milliseconds
DD nnnn *	: Binary digital output (nnnn <= 4095)
DB nnn *	: BCD digital output (nnn <= 999)
%	: End edit (downloading)

APPENDIX 2

IEEE-488 BUS INTERFACE FUNCTIONS

APPENDIX 2

IEEE-488 BUS INTERFACE FUNCTIONS

These are messages sent on the bus with the ATN line asserted. The bus functions listed below are valid only when Unidex XI or IV is in the Interface Active mode or the Communication Enabled mode. If Unidex XI or IV is in the Local mode, bus handshake protocol may not be completed.

BUS FUNCTION	UNIDEX XI or IV RESPONSE
DCL (Not addressed) SDC (Addressed)	Goes into Power-up state if not already there.
GTL (Addressed)	Goes into Local mode with Communication Enabled. Position registers and previous command block are not cleared.
GET (Addressed)	Executes command block and sets SRQ if in Hold mode. No response if not in Hold mode.
SPE (Addressed) SPD (Addressed)	These two components of a Serial Poll cause Unidex XI or IV to put out the status byte and reset SRQ if set.
UNL UNT MLA MTA LAG TAG	These are components of Data Transfers. They cause appropriate data transfer sequences
IDY (Identify) (Parallel Poll)	Puts the identity on the bus as a response to a parallel poll; transparent to Unidex XI or IV.

APPENDIX 2

BUS FUNCTION	UNIDEX XI or IV RESPONSE
TCT (Addressed)	Bus hangs up
LLO	No response
UUCG (Undefined Universal Command Group)	No response
UACG (Undefined Addressed Command Group)	No response

INDEX

Warranty and Field Service Policy

Aerotech, Inc. warrants its products to be free from defects caused by faulty materials or poor workmanship for a period of one year from date of shipment from Aerotech. Seller's liability is limited to replacing, repairing or issuing credit, at its option, for any products which are returned by the original purchaser during the one-year period. Seller makes no warranty that its products are fit for the use or purpose to which they may be put by the buyer, whether or not such use or purpose has been disclosed to seller in specifications or drawings previously or subsequently provided seller, and whether or not seller's products are specifically designed and/or manufactured by seller for buyer's use or purpose. Aerotech's liability on any claim for loss or damage arising out of the sale, resale or use of any of its products shall in no event exceed the selling price of the unit.

Returning Goods Procedure

Claims for incorrect or defective materials must be filed within thirty (30) days from delivery at buyer's place of business. No units or systems may be returned, in or out of warranty, without first obtaining approval from the seller, and no claim will be allowed nor credit given for units or systems returned without such approval.

Returned Goods Warranty Determination

If possible, after approval from Aerotech, the defective unit or system is to be returned to the factory with statement of problem and transportation prepaid (no c.o.d. or collect freight shipments will be accepted). After Aerotech's in-plant examination, warranty or out-of-warranty status will be determined. If upon Aerotech's examination of such unit or system, warrantied defects exist, then the unit or system will be repaired at no charge and shipped, prepaid, back to the buyer. If an out-of-warranty situation exists, the buyer shall be notified of the repair cost immediately. At such time, the buyer must issue a purchase order to cover the cost of the repair or authorize the unit or system to be shipped back as is, at the buyer's expense.

On-Site Warranty Repair

If the system or unit cannot be made functional by telephone assistance or by sending and having customer install replacement parts, and cannot be returned to the Aerotech factory for repair, and if it is determined that the problem could be warranty-related, then the following policy applies:

Aerotech will provide an on-site field service representative in a reasonable amount of time, provided that the customer issues a bona-fide purchase order to Aerotech covering all transportation and subsistence costs. For warranty repairs, customer will not be charged for cost of labor and material.

If during the on-site repair it is determined the problem is not warranty related, then the terms and conditions stated in the following "On-Site Non-Warranty Repair" section apply.

On-Site Non-Warranty Repair

If system or unit cannot be made functional by no-charge telephone assistance or purchased replacement parts cannot be returned to the Aerotech factory for repair, then the following field service policy applies:

Aerotech will provide an on-site field service representative in a reasonable amount of time, provided that the customer issues a bona-fide purchase order to Aerotech covering all transportation and subsistence costs and the prevailing cost per hour including travel time necessary to complete the repair.

AEROTECH,INC., 101 Zeta Drive, Pittsburgh, Pennsylvania 15238 • (412) 963-7470
TWX 710-795-3125 • FAX 412-963-7459