REFERENCE
DOCUMENT

UNIDEX 400 SERIES
MACHINE CONTROLLERS

OPERATION & TECHNICAL MANUAL

P/N: EDU123

AEROTECH, INC.
101 Zeta Drive
Pittsburgh, PA 15238-2897

AEROTECH, INC.

Revision History

Revision 1.0	Preliminary	7/92
Revision 1.1	1st Printing	10/92

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION

SECTION 1-1: USING THIS MANUAL

The Unidex 400 User's Manual is divided into the following chapters:

Chapter 1: Introduction
This Chapter outlines the contents of this manual.

Chapter 2: Getting Acquainted With the Unidex 400
This Chapter contains a general description of the Unidex 400, an explanation of the Units serial number and delineation of general operating functions.

Chapter 3: Edit Mode
This Chapter provides detailed information to be used when creating or editing a program. Each of the menu selections are described and examples of the command block created are provided.

Chapter 4: Command Summary and Sample Programs
This Chapter provides a list of the Machine Tool Codes used by the Unidex 400 as well as a reference to corresponding menu selections.

Chapter 5: File Mode
This Chapter provides information concerning Unidex 400 file management. File retrieval, loading, copying, and deletion procedures are described.

Chapter 6: Machine Mode
This Chapter provides information necessary to run programs to create motion.

Chapter 7: Setup Mode
This Chapter contains an explanation of each of the general and axis operating parameters. These parameters establish the manner in which the Unidex 400 will proccess a program and execute a move.

Chapter 8: The Test Mode

This Chapter describes the internal testing proccess performed by the Unidex 400 upon Power Up or Reset.

Chapter 9: Remote Operation

This Chapter provides the User with the information necessary to operate the Unidex 400 remotely from a Terminal or a Host Controller.

Chapter 10: Unidex 400 Hardware

This Chapter contains a description of each of the standard circuit boards and Jumper information as well as Wiring Diagrams for each of the various models.

Chapter 11: Interfacing the Unidex 400

Details of each of the Unidex 400 Rear Panel Interface connectors are provided, including pinouts and electrical characteristics.

Chapter 12: Error Messages

This Chapter contains a list of error messages that may be encountered during operation, the source of the error and a possible solution, when applicable, is also provided.

Chapter 13: Troubleshooting

This Chapter lists symptoms, that may be evident during operation of the Unidex 400, indicating conditions requiring User attention. Where applicable, possible causes and solutions are provided.

Chapter 14: Maintenance

Instructions are provided for battery replacement, software updates and service.

Chapter 15: Options

This Chapter contains an operational description of options that may be included with the Unidex 400.

SECTION 1-2: LIST OF FIGURES

(THIS PAGE INTENTIONALLY LEFT BLANK)

CHAPTER 2: GETTING ACQUAINTED WITH THE UNIDEX 400

The Unidex 400 is a multi-dimensional Motion Controller capable of controlling up to four axes of motion. The position, speed and torque of a variety of motor sizes and types (Step, Servo, and Brushless) as well as amplifiers (linear or PWM) may be controlled. Encoder, Resolver, Inductosyn or Interferometer position feedback is accommodated. Interface to the Unidex 400 is done through the rear connector panel except for the keyboard connector which is located on the Front Panel. The U400N is interfaced both front and rear.

SECTION 2-1: DESCRIPTION

U400B

The Unidex 400, Model U400B, is a rack mount version, internally equipped with the necessary motor drivers and power supply for independant operation. It requires remote communication since there is no integral Front Panel. This is accomplished by means of interfacing the Unidex 400 to a Remote Front Panel, a Terminal or a Host Controller. (See Figure 2-2.)

U400K

The Unidex 400, Model U400K, is a rack mount version, providing current command outputs that must be interfaced to an external drive chassis. Communication is through interface to a Remote Front Panel, a Terminal or Host Controller since the U400K does not contain an integral Front Panel.

U400N

The Unidex 400, Model U400N, provides current command outputs that must be interfaced to an external drive chassis. Communication is through interface to a Remote Front Panel, a Terminal or Host Controller, since the U400N does not contain an integral Front Panel. (See Figure 2-3).

U400R

The Unidex 400, Model U400R, is a rack mount version, internally equipped with the necessary motor drivers and power supply for operation. It contains an integral Front Panel providing complete stand-alone operation (See Figure 2-1). Remote Operation, through a Terminal or Host Controller is also possible.

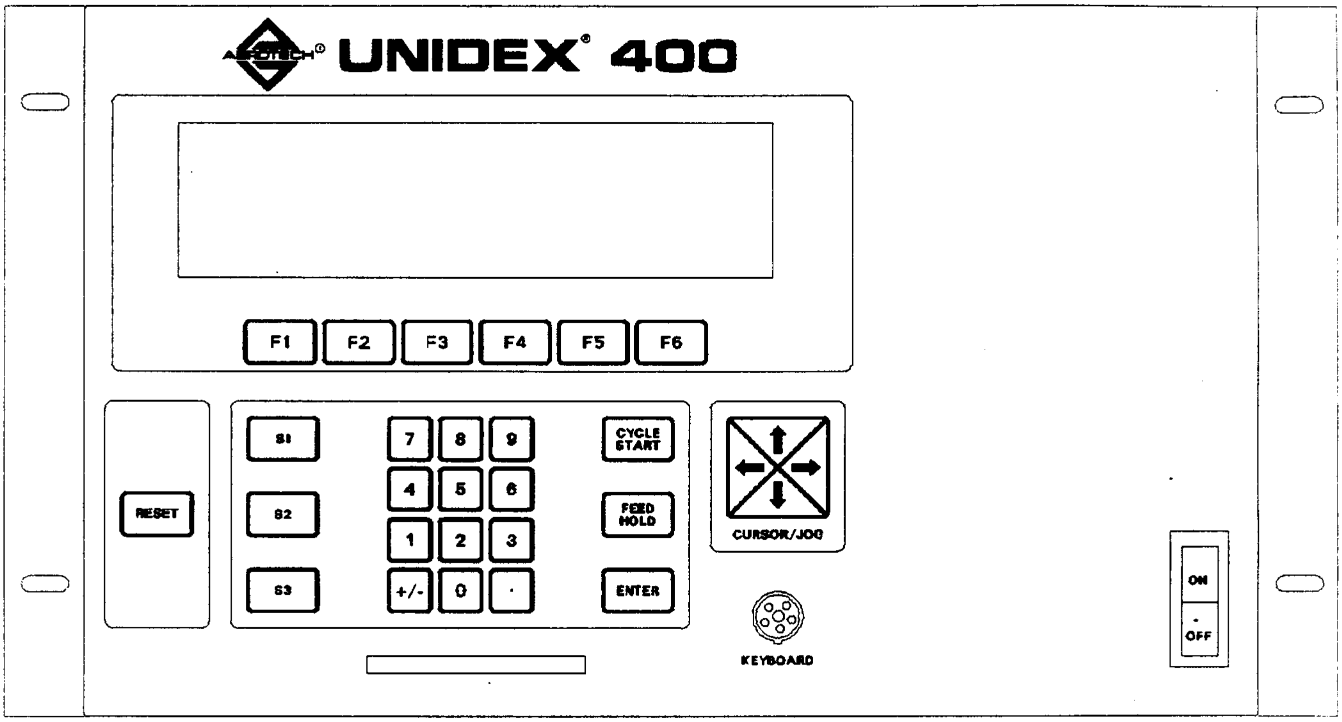


Figure 2-1: Unidex 400R (Front Panel Display)

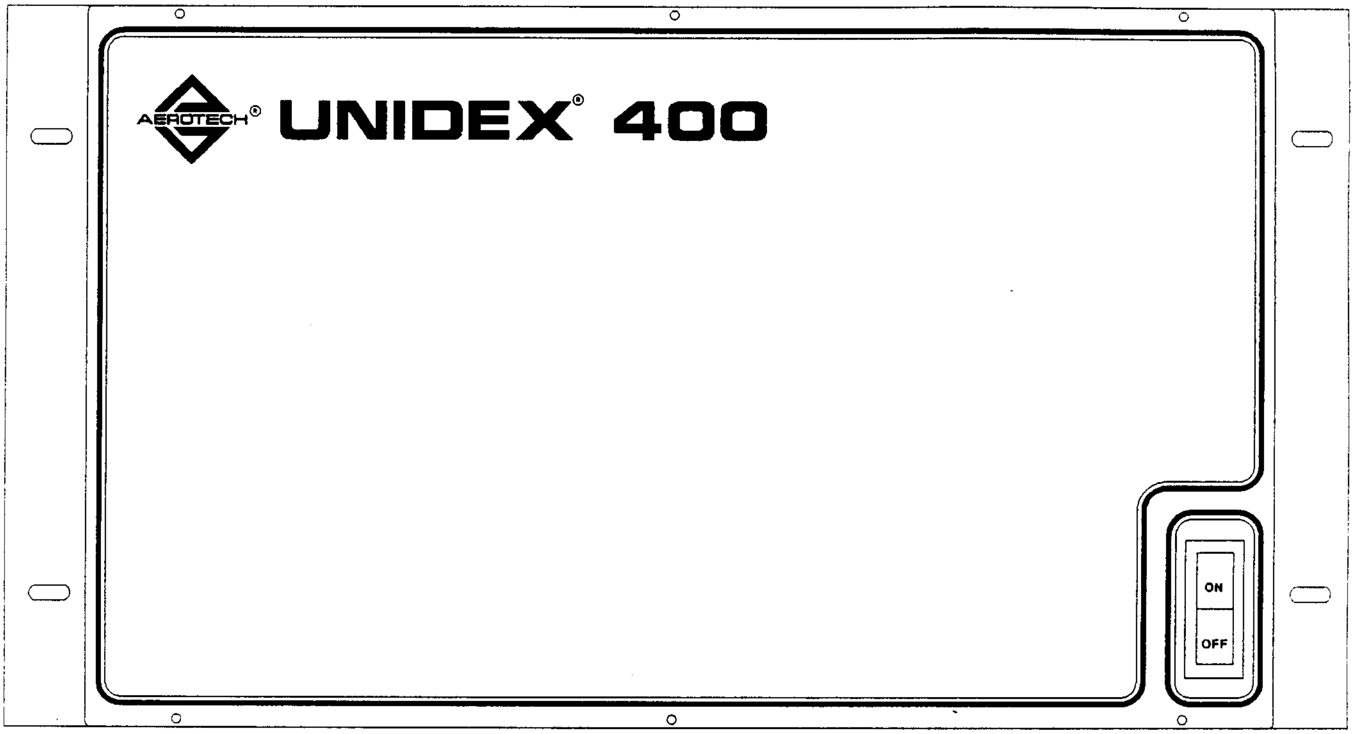


Figure 2-2: Unidex 400 B (Blank Front Panel)

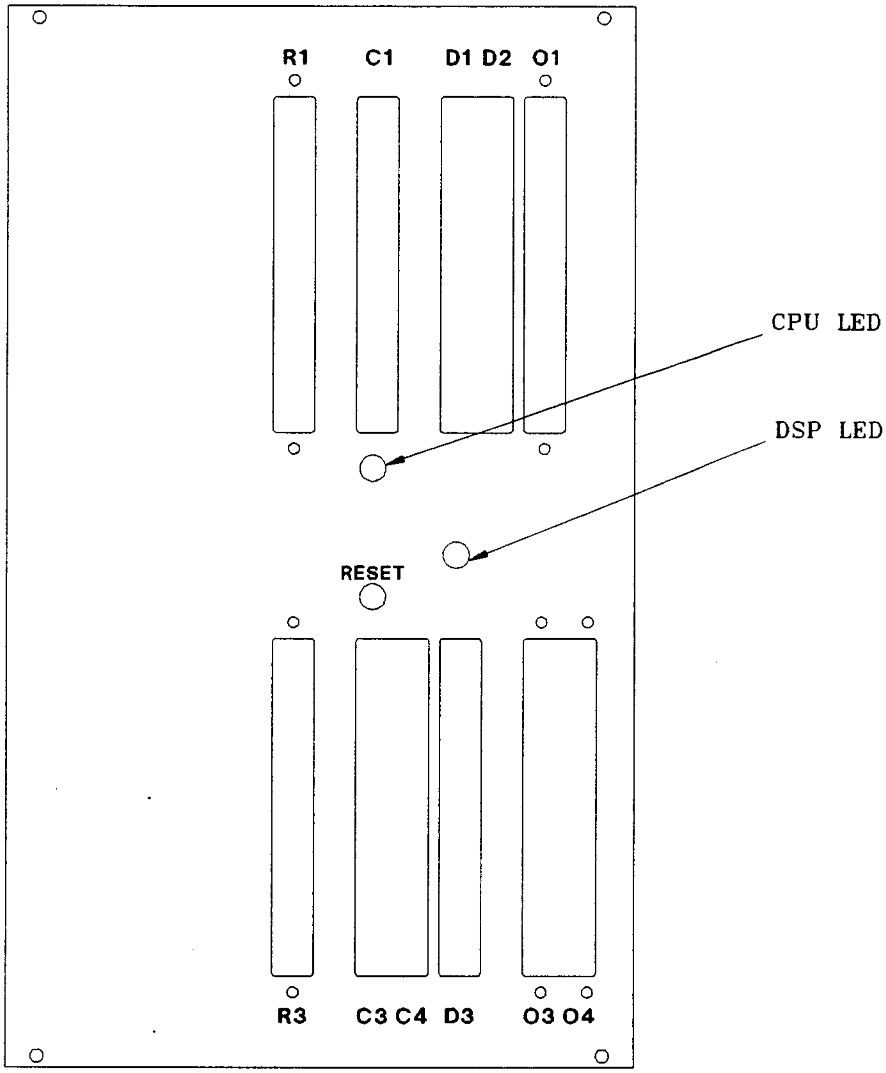


Figure 2-3: Unidex 400N (Industrial Package)

2-2: FACTORY INSTALLED UNIDEX 400 OPTIONS

The Unidex 400 may be equipped with several factory installed options. To determine which options have been installed in your controller, examine the serial number. The following serial number contains a variety of available options. This serial number is for illustration only since no controller would actually be so equipped.

U400R-4-B-40 / X2-DL4020C-20-F5/X2-DS16020C-10-F3/DTC/FDD/JBV/ES12345
1 2 3 4

/1/ Unidex 400 Chassis Configuration:

U400B - 19 inch Rack mount version with blank front panel, accommodates up to four integral motor drivers and includes DC bus power supply.

U400K - 19 inch rack mount version with blank front panel , current command motor drive outputs (Refer to listing of external drive chassis).

U400N - panel mount industrial version, current command motor drive outputs (Refer to listing of external drive chassis).

U400R - 19 inch rack mount version, integral operator panel, accommodates up to four integral motor drivers, includes DC bus power supply.

-**1** thru **4** - potential number of axes directly controlled by the Unidex 400

-**A** -115V Operation
-**B** - 230V Operation
-**C** - 100 VAC, 50Hz, single phase for system, 115 VAC for drivers
-**D** - 208 VAC, 50/60 Hz, single phase for system and drivers

-**20** , **40**, **60**, **80**, **160**, and **320** - available DC bus voltages

/2/ Axis Motor Drivers - One to four fields may specify the Motor Drivers that are installed in the Unidex 400 (Models U400B and U400R only). Each Driver field has a suffix indicating resolution (in hundreds of steps per rev.) followed by a code indicating the fuse rating (-Fnn).

In the previous example, DL4020C drivers with a 2000 steps/rev. resolution and a 5A current are specified for axes X and Y. (Axes are always listed in the order X, Y, Z, U; X2 indicates both the first and second axes, X and Y have the same drivers). Axes Z and U have DS16020C Drivers with 1000 steps/rev. resolution and 3A current.

The Motor Drivers used with the Unidex 400 are:

AM16020-	Variable Bus, 10A cont., 3U stepping motor driver
DL4020C-	40V, 20A pk, 5A cont., linear 6U DC servo driver
DS16020C-	160V, 20A pk, 10A cont., 20kHz pwm, 3U DC servo driver
DS16030C-	160V, 30A pk, 15A cont., 20kHz pwm, 3U DC servo driver
AS16020C-	160V, 20A pk, 7.5A cont., 20kHz pwm, 6U brushless driver
AS32030C-	320V, 30A pk, 15A cont., 18kHz pwm, 6U brushless driver

Unidex 400 Models U400K and U400N are equipped with one of the following Mounting Chassis for one of the above Drivers.

DSR-3 -A-	3-Axis, Panel-Mount Chassis for 3U DC Servo Drivers
DSFR-3-A-	3-Axis, 19 inch Rack-Mount Chassis for 3U DC Servo Drivers
DSFR-6-A-	6-Axis, 19 inch Rack-Mount Chassis for 3U DC Servo Drivers
APR-4-A-	8-axis, Panel-Mount Chassis for 6U Drivers
APRF-8-A-	8-Axis, 19-inch Rack-Mount Chassis for 6U Drivers

/3/ Options - One or more fields of the part number may indicate options that are included in your Unidex 400 system.

FUNCTION CARDS

MPI-B	Opto 22™ PAMUX™ I/O bus interface Card
MPI-C	IEEE-488 parallel communications card
MPI-E	Combination IEEE-488 and PAMUX™ I/O bus interface card
PSO	Position Synchronized Output (laser firing control) card
RDP	Resolver-to-digital conversion interface card for 4 axes
RMX	Resolution multiplier (up to x256) interface card for 4 axes. Requires sinewave output encoders.

HARDWARE

EXB	Extension bus backplane
VMB	VME bus backplane
BCM	Z axis brake control module
BPS	Brake power supply, 24VDC at 450mA
DTC	Desktop Enclosure case for U400R or U400B
JBV	Joystick Controller with digitizing
TBV	Three-axis Trackball Controller with digitizing
M1B	One megabyte battery-backed static RAM card
M2B	Two megabyte battery-backed static RAM card

TRANSFORMERS FOR EXTERNAL DRIVER CONFIGURATIONS

TV1.5	1.5kVA, 104/115/208/230 VAC, 50/60 Hz, single-phase
TV2.5	2.5kVA, 104/115/208/230 VAC, 50/60 Hz, single-phase
TV5	5kVA, 104/115/208/230 VAC, 50/60 Hz, single-phase

WIRING OPTIONS FOR ENCLOSURE MOUNTED SYSTEMS

ACW-	AC input wiring for TV1.5, TV2.5, TV5; mounted on 17" x 17" (432mm x 432mm) subplate.(Voltage A, B, C, or D)
EMG	Per Axis emergency stop contactor and circuitry for DC or brushless drives rated to one horsepower
EMW	Per Axis encoder and motor wiring
MWP	Rack-mount panel to accommodate up to eight axis of EMW wiring
BOM	Breakout module assembly equipped with three foot of ribbon cable

I/O ACCESSORIES

PB16	Opto 22™ I/O mounting rack for 8IN/8OUT
PB24	Opto 22™ I/O mounting rack for 16IN/8OUT
LM16	Latched M function relay card, each relay rated 120VAC or 28VDC at 5A resistive; maximum three per system
LM16B	Latched M function relay card, each relay rated 240VAC at 2A resistive; maximum one per system
PB8H-B5	Opto 22™ I/O rack for 8 digital I/O on PAMUX™ bus
PB16H-B5	Opto 22™ I/O rack for 16 digital I/O on PAMUX™ bus
PB4AH-B6	Opto 22™ I/O rack for 4 analog I/O on PAMUX™ bus
PB8AH-B6	Opto 22™ I/O rack for 8 analog I/O on PAMUX™ bus
PB16AH-B6	Opto 22™ I/O rack for 16 analog I/O on PAMUX™ bus
OAC5A	AC output module, 24 to 280VAC, 2A
IAC5	AC input module, 90 to 140VAC
ODC5	DC output module, 5 to 60VDC, 2A
IDC5	DC input module, 10 to 32VDC
IDC5B	DC input module, 4 to 16VDC
AD3	4 to 20mA current analog input module
AD6	DC analog input module, 0 to 5VDC
AD7	DC analog input module, 0 to 10VDC
AD12	DC analog input module, -10 to 10VDC
DA3	DC analog current output 4 to 20mA
DA4	DC analog voltage output 0 to 5VDC
DA5	DC analog voltage output 0 to 10VDC
DA7	DC analog voltage output -10 to 10VDC
TERM1	PAMUX™ bus terminator card
SDA	S-command-to-analog output; 12 bit resolution

SOFTWARE ACCESSORIES

CIMLITE Computer Integrated Manufacturing software package; provides CAD-to U400 postprocessor facility plus CAD file editing features.

CIMCAD Computer Integrated Design and Manufacturing software package; provides all of CIMLITE features plus complete CAD design features

IBM Graphics Software **xxxxx**

/4 Engineering Specification - A modified or special version of a Unidex 400 will have an Aerotech, Inc. Engineering Specification number assigned to it, to provide traceability. The "ES" number occupies the last field of the Serial Number.

2-3: GENERAL OPERATING CONVENTIONS

The manner in which the Unidex 400 is operated is dependent upon the model type being used. The following functions, though initiated differently, are common to all Unidex 400's:

RESET

The Unidex 400's RESET function serves to clear the system and return to initial power-up conditions. General Setup Parameter 024 may be configured to retain Servo positions following a RESET.

NOTE: **Any data entered while in the Edit Mode and not saved, will be lost following a RESET.**

S1, S2, S3

Special function keys that are used primarily to clear errors. The User will be prompted as to their use.

CYCLE START

The CYCLE START function is used to step through a program one block at a time while running a program in the Machine-Single Mode. This function may also be used in conjunction with the Dwell (G4) command to signal the restart of a program.

FEEDHOLD

The FEEDHOLD function is used to stop program flow and motion of all axes. A FEEDHOLD is canceled by repeating the key sequence used to initiate it.

ENTER

ENTER is used to accept the current status.

SECTION 2-4: UNIDEX 400 SPECIFICATIONS

2-4-1 DIMENSIONS

U400B/U400K/U400R

Width	19.0 in	483,0 mm
Height	10.5 in	266,7 mm
Depth	14.6 in	370,0 mm

U400 Remote Front Panel

Width	19.0 in	483,0 mm
Height	10.5 in	266,7 mm
Depth	1.9 in	48,3 mm

U400N

Width	7.8 in	198,1 mm
Height	12.4 in	315,0 mm
Depth	9.4 in	238,8 mm

2-4-2 WEIGHTS

U400B/U400K	U400R	U400N
53 lbs (24kg)	62lbs (28,2kg)	15 lbs (7,3kg)

2-4-3 POWER REQUIREMENTS (LESS DRIVERS)

270 Watt Power Supply, 100-120 Volts at 4A or 220-240 Volts at 1.9 A.

2-4-4 ENVIRONMENTAL CONSIDERATIONS

Ambient Temperature

Operating: 0 to 35 degrees C (32 to 95 degrees F)
Storage: (-) 20 to 70 degrees C ((-)4 to 158 degrees F)_

Humidity

0 to 95%, at 35 degrees C (95 degrees F) noncondensing.

(THIS PAGE INTENTIONALLY LEFT BLANK)

CHAPTER 3: EDIT MODE

SECTION 3-1: INTRODUCTION

The Unidex 400 enables the User to edit programs by two methods; "On Screen Editing" and "Menu Driven Editing".

"On Screen Editing" is done directly by using an IBM keyboard to enter the desired machine tool code into the program block. This editing function provides a "what-you-see-is-what-you-get" type of editing. The entire program may be altered or created character-by-character.

"Menu Driven Editing" provides program editing functions through the use of the Unidex 400's Front Panel Keypad. Menus are provided from which the User makes the desired edit selections. The Unidex 400 internally translates the English-like mnemonics of the menu selections to machine code. Menu selections affecting the entire program as well as those to create or edit a specific program block are available.

This Chapter provides a detailed description of each of the editing methods.

SECTION 3-2: GETTING STARTED

Following Power-Up, the Initial Selection screen shown below will be displayed:

```
    UNIDEX 400                    Version x.xx

     RAM:ok   ROM:ok   SETUP:ok
     User RAM: xxxxxx

edit     file     setup     test     machine     PSWD
```

NOTE: **The PSWD option (F6) is displayed only if the Password function has been activated from the Setup Mode.**

Enter the Edit Mode by pressing F1.

The following message will be displayed:

```
editing & digitizing

   input file name (0 to 9900) =


ins-Y/N     del     last-E          last-R     quit
```

Key in the program number to be edited.

The following functions are available during program selection:

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (No). When ON characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

last-E Press the F3 key to recall the name of the last program that was edited.

last-R Press the F5 key to recall the name of the last program that was run.

quit Press the F6 key to abort the command and return to the initial selection menu.

Press the **ENTER** key to accept the program name.

SECTION 3-3: GENERAL EDIT FUNCTIONS

The Unidex 400 contains two general purpose Edit Screens, each containing six distinct edit functions. Functions are selected by pressing the corresponding Function key on the integral Unidex 400 keypad or an IBM-AT compatible keyboard. The "m" at the far right of the functions indicates additional functions are available, pressing the "up/down arrow" key changes from one function screen to another. The two general edit screens are illustrated below:

```
> 0: (program)

insrt   alter   delete   up   down   end   m
```

```
> 0: (program)

KEYB   paste   copy   merge   find   erase   m
```

A description of each of the general edit functions follows:

insrt The Line Insert function is used to insert a new program block into a new or existing program file. The new line will be inserted at the current cursor location. The options displayed following this selection are dependent upon whether the Unidex 400 is in the KEYBOARD or MENU Mode as established in the second general edit screen.

alter Press the F2 key to modify the program line at the cursor location. The options displayed following this selection are dependent upon whether the Unidex 400 is in the KEYBOARD or MENU Mode as established in the second general edit screen.

delete Press the F3 key to delete the line at the cursor location. All subsequent program blocks will be renumbered.

up Select **up** to scroll the program text up one line.

down Select **down** to scroll the program text down one line.

end Select **end** to exit the Edit Mode. The following options are presented :

Save The modified program is saved to a file. The display returns to the Initial Selection Menu.

Abort The modified program is abandoned. The display returns to the Initial Selection Menu.

Redo No action is taken on the program. The display returns to the first general edit screen.

KEYB/MENU This function is used to select the edit mode that will be active when using the **insert** or **alter** functions. If **KEYB** is displayed, program editing or creation is accomplished by direct entry of machine tool code with an IBM AT compatible keyboard. If **MENU** is displayed, program editing or creation is accomplished by selection of appropriate menu options.

paste This function retrieves data sent to the internal buffer by the **Copy** command. The data will be placed starting at the cursor location. Cursor placement is achieved using **up** or **down** from the first edit screen.

copy This function enables the User to copy data to an internal buffer. Prior to using this function, the cursor must be positioned at the beginning of the data to be copied. Cursor movement is achieved by using **up** or **down** from the first edit screen.

Press the F3 key to activate the Copy function. The display will be:

```
    > 0: (program)

copy thru Line/Label:
line      label      up      down        quit
```

Use the F1 and F2 keys to toggle between designating a Line number or a Label as the end of the data to be copied. Enter the Line number or Label to be used to signal the end of the data to be copied to the buffer.

merge The **merge** function enables the User to add the contents of another file to the file currently being edited. Prior to using this function, the cursor must be positioned at the location at which the external file is to be placed. Cursor movement is achieved by using **up** or **down** from the first edit screen.

Press the F4 key to merge a file into the current file. The display will be:

```
    > 0: (program)

Merge from file:      up      down        quit
```

Enter the File name to be merged with the current file. The external file will be added at the cursor location and all subsequent program blocks will be renumbered.

find The **find** function allows the User to specify a location within the program to place the cursor. Press the F5 key to activate the Find function. The display will be:

```
> 0: (program)

find Line/Label:
line     label          up          down               quit
```

Use the F1 and F2 keys to toggle between designating a Line number or a Label for the new cursor location. Enter the Line number or Label to which to move the cursor.

erase The **erase** function is used to delete single or multiple program blocks from the file. Prior to using this function, the cursor must be positioned at the beginning of the data to be erased. Press the F6 key to activate the **erase** function. The display will be:

```
    > 0: (program)
    > 1:
Erase to line:
Line        Label        up          down               quit     up
```

Use the F1 and F2 keys to toggle between designating a Line number or a Label to signify the end of the data to be erased. Enter the Line number or Label signifying the end of the data to be erased

SECTION 3-4: JOYSTICK DIGITIZING

Press the "down arrow" key to display the third Edit screen.

	> 0: (program)				
D_idx	**D_lin**	**D_cir**	**D__l+c**	**D_c+c**	**m**

The functions available on Edit Screen #3 are used when digitizing positions with the optional Joystick. The Joystick is used to move an axis or multiple axes to a given position. Pressing the top Joystick button will record the current position and write it to a program. The type of motion between these points is determined by functions selected from this menu screen. Refer to the Programming and Options Section of this manual for details concerning Joystick operation.

Following is a brief description of the motions available from this menu screen for Joystick digitizing:

D_idx Digitize in the Index Mode. Motion is point to point for each axis (see the **G0** program command). Refer to the Options Section of this manual for details concerning Joystick digitizing in the Indexing Mode.

D-lin Digitize in the Linear Contour Mode. Multiple axes with synchronized motion having a vectored feedrate for contouring (see the **G1** program command). Refer to the Options Section of this manual for details concerning Joystick digitizing in the Linear Contour Mode.

D_cir Digitize in the Circular Mode. Motion is generated in the appropriate circular profile from the current point through the next two digitized points (see the **G2** and **G3** program commands). Refer to the Options Section of this manual for details concerning Joystick digitizing in the Circular Contour Mode.

D_l+c Digitize in the Linear and Circular Mode. Motion is first generated in a circular profile from the current point through the next two digitized points (see the **G2** and **G3** program commands), and then a different axis pair may be selected for linear motion (see the **G2** command).Refer to the Options Section of this manual for details concerning Joystick digitizing in the Linear and Circular Mode.

D_c+c Digitize two Circles. Motion is generated in the appropriate circular profile from the current point through the next two digitized points (see the **G2** and **G3** program commands). Refer to the Options Section of this manual for details concerning Joystick digitizing in the Circular Mode.

Press the **ENTER** key to accept the digitized program block.

SECTION 3-5: WRITING A PROGRAM

Instructions for program block creation through the menu driven mode are provided in this section. A full range of program creation and edit functions are available through this mode by the selection of appropriate menu options from the Unidex 400 Front Panel Keypad. Examples are provided of both machine code and menu driven code as well as the respective program block formats.

Place the Unidex 400 in the Menu Driven Mode by selecting **MENU** from Edit Screen #2. Once established in the Menu Mode, the program command options are available by selecting "insert" (F1) or" alter" (F2) from Edit Screen #1. As program commands are selected, a program block is created in english-like format. NOTE: Up to 38 characters may be displayed for each program block being edited, additional characters are truncated.

The program block may be viewed in Machine Tool Code by returning to Edit Screen #2 and selecting **KEYBD.**

After selection of **insert** or **alter** from Edit Screen #1 the display will be:

```
>0: (program)
>1:

motion    I/O    mode    flow    servo    quit    m
```

NOTE: **The display selections necessary to insert or alter a program block with the function being described, are provided at the top of the subsequent pages of this Chapter. It should be assumed that all EDIT Mode selections are preceded with selection of EDIT and either INSERT or ALTER.**

3-5-1: WRITING A PROGRAM BLOCK TO PRODUCE MOTION

General Information - Axis motion may be specified by a direct floating point number, such as "X=1.". This will move the X axis one User unit (mm,inches,degrees etc.). The statement "X=1" (no decimal point) will move the X axis .0001 User units (assuming four decimal places, see Chapter 7, General Parameter #4). Axis motion may also be specified through the use of system variables.

Example:

(menu driven)	(Machine Tool)	
0:linear F100.	0:G1 F100.	:Sets feedrate to 100 User units/minute
1:V10=100	1:G97 V10=100	;Sets the system variable V10 to 100
2: linear XV10	2:G1 XV10	;Produces a linear contour move of 100 User Units

Note that the decimal point is always implied when using variables.

Feedrate - The feedrate input recognizes an established decimal point, if none is specified, the Unidex 400 internally places a decimal point two places to the left. Example: "F100" is read as 1.00, "F100." is read as 100 Feedrates may also be defined by variables and need only be entered once.

Select "Motion" to write a program block to initiate axis movement.

The following display provides the options for entering a Motion command.

```
>0: (program)
>1:

index    cntour    home    free        quit
```

MOTION --> INDEX

3-5-1-1: INDEX

Select **Index** for non-synchronized motion of any or all of the four axes at a rapid traverse rate. The Acceleration/Deceleration ramping trajectory for this command is parabolic. The parabolic coefficient is calculated based on the current acceleration deceleration rate. This menu command establishes the **G0** and **F** commands in machine tool language. This command is modal.

Axis Parameters 8, 9 and 30, and General Parameter 5 settings affect this command. The Index display is:

			INDEXING			
X		F		Z	F	
Y		F		U	F	
X	Xfedr	Y	Yfedr	Z	Zfedr	m

Press the "down" arrow for additional Indexing options.

			INDEXING			
X		F		Z	F	
Y		F		U	F	
ins-Y/N	del	U	Ufedr	Varb	Quit	m

X-U Press the appropriate Function key to enter the desired move distance for each of the axes.

Xfedr-Ufedr Press the appropriate Function key to enter the desired feedrate for each of the axes.

ins-Y/N Press the F1 key of the second Index menu to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

del Press the F2 key of the second Index menu to delete the character at the cursor position.

Varb Select **Varb** to insert a pre-established variable (refer to the Math Mode for variable definition) for a move distance or feedrate.

MOTION – > CONTOUR – > LINEAR

quit Select **quit** to abort the command and return to the initial Motion selection menu.

Press the **ENTER** key to write the program block.

Program Block Format: move axis name distance feedrate

Example: index X5. F100. Z2. F200. or G0 X5. F100. Z2. F200. ; point-to-point move of X and Z at 200 units/min.

CONTOUR

3-5-1-2: Contour

Select **cntour** (F2) from the initial Motion menu to produce synchronized, vectored feedrate motion. All enabled axes will start and stop the requested motion at the same time. The feedrate need only be entered once for all contour moves.

General Parameters 5, 11, 37, 38, 39, 40 and 41 affect this command.

The following menu screen is provided for Contour Motion selections:

linear	**2 cir**	**cir+1**	**quit**

CONTOUR-LINEAR

linear Select **linear** to establish a program block resulting in a linear move. This menu command establishes the **G1** and/or **F** commands in machine tool language. The **G8** command is created if Velocity Profiling is enabled.

MOTION-- > CONTOUR-- > LINEAR

The following displays provide the options for entering a Linear Contour Motion command:

LINEAR, Vel Profile :				F		
X			Z			
Y			U			
X	Y	Z	U	F	Varb	m

Press the "down" arrow for additional Linear Contouring options.

LINEAR, Vel Profile :			F		
X		Z			
Y		U			
ins-Y/N	del	Velpof		quit	m

X-U Press the appropriate Function key to enter the desired move distance for each of the axes.

F Select **F** to enter the Feedrate that all axes are to move.

Varb Select **Varb** to insert a pre-established variable (refer to the Math Mode for variable definition) for a move distance or the feedrate.

ins-Y/N Press the F1 key of the second Linear menu to toggle character insert ON (Y) or OFF (No). When ON, characters are inserted without overwriting existing text.

del Press the F2 key of the second Linear menu to delete the character at the cursor position.

MOTION – > CONTOUR – > LINEAR – > VELPRO

Velpof Select **Velpof** to enable or disable Velocity Profiling for this Linear Contour move. During Velocity Profiling, priority is given to attaining and maintaining the programmed feedrate, with the secondary emphasis on achieving the end point. If position overshoot occurs, the value of this overshoot is automatically subtracted from the programmed position of the next motion command in the program stream. The system uses the ramp time to accelerate or decelerate from the current velocity to the desired velocity.

If Velocity Profiling is not enabled, the default provides for both acceleration and deceleration to zero velocity.

To maintain smooth transitions from block to block motion, the **G8** command requires that consecutive moves be tangent to each other.

Refer to Appendix A for additional information concerning Velocity Profiling.

Press the **ENTER** key to write the program block.

Program Block Format: vel profile (if enabled) move axis name distance feedrate

Example: vel linear X5. F100. or G8 G1 X5. F100. ; linear contour move of the X axis at a feedrate of 100 units/second with no deceleration.

MOTION – > CONTOUR – >2CIR – >VELPRO

2cir Select **2cir** to establish a program block resulting in a move consisting of one or two circles.

The following displays provide the options for entering a Circular Motion command.

2CIR, Vel Profile :			**F**			
1dir	**axes**	**2dir**	**axes**			
1dir	**1axes**	**1data**	**2dir**	**2axis**	**2data**	**m**

Press the "down" arrow for additional Circular Motion options.

2CIR, Vel Profile:			**F**			
1dir	**axes**	**2dir**	**axes**			
INS-y/n	**del**	**velpof**	**F**	**Varb**	**quit**	**m**

Key in the desired Feedrate of the Circular motion.

Velpof Select **Velpof** to enable or disable Velocity Profiling for this Circular move. During Velocity Profiling, priority is given to attaining and maintaining the programmed feedrate, with the secondary emphasis on achieving the end point. If position overshoot occurs, the value of this overshoot is automatically subtracted from the programmed position of the next motion command in the program stream. The system uses the ramp time to accelerate or decelerate from the current velocity to the desired velocity.

If Velocity Profiling is not enabled, the default provides for both acceleration and deceleration to zero velocity.

To maintain smooth transitions from block to block motion, the **G8** command requires that consecutive moves be tangent to each other.

Refer to Appendix A for additional information concerning Velocity Profiling.

MOTION --> CONTOUR -->2CIR

1dir Select **1dir** to establish the direction of the first circular move. The selection toggles between Clockwise and Counterclockwise direction. This menu command establishes the **G2** or **G3** command in machine tool language.

1axis Select **1axis** to designate the axes to be involved in the first circular move. Two axes must be selected. The following selections will be displayed:

```
2CIR, Vel Profile :                    F
1 dir        axes        2dir        axes


  X        Y        Z        U        data        back
```

Select the axes to do the first circular move.

data The end and center points of the circular move may be entered by selecting **data**. (Circular move data may also be entered through the **1data** and **2data** selections of the initial circular motion menu.)

NOTE: **Endpoint and centerpoint data must be established for each axis involved in a circular move.**

The following displays provide the options for entering circular motion data:

```
2CIR, Vel Profile :                    F
1dir         axes        2dir        axes


  1end1       end2       cent1       cent2      Varb        back        m
```

Press the "down" arrow for additional Circular Motion data entry options.

```
2CIR, Vel Profile :                    F
 1 dir        axes        2dir        axes


  INS-y/n       del       velpof       F       Varb       quit       m
```

MOTION --> CONTOUR -->2CIR

1end1 Select **1end1** to establish the endpoint for the first axis in the first circular move. Key in the desired end point at the cursor location.

end2 Select **end2** to establish the endpoint for the second axis in the first circular move. Key in the desired endpoint at the cursor location.

cent1/cent2 Select **cent1/cent2** to establish the centerpoint for the first/second axis in the first/second circular move. Key in the desired centerpoint at the cursor location. This will establish the **I, J, K,** or **P** command in machine tool language. This code programs the Offset Vector which is the incremental distance from the beginning of the arc to the arc center. The Offset Vector is determined incrementally regardless of whether the Unidex 400 is in the Absolute (**G90**) or Incremental (**G91**) mode.

Varb Select **Varb** to insert a pre-established variable (refer to the Math Mode for variable definition) for a centerpoint, endpoint or feedrate.

back Select **back** to return to the previous menu.

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

del Press the F2 key of the second Linear menu to delete the character at the cursor position.

At the completion of data entry for the first circle, select "**back**" to return to the initial Circular Motion menu.

The display will be:

```
2CIR, Vel Profile :                   F
1dir      axes      2dir      axes


 1 dir     1axei     1data     2dir     2axes     2data    m
```

MOTION --> CONTOUR -->2CIR

1data The end and center points of the first circular move may be entered by selecting **1data.** (Data entry with this selection is done in the same way as previously described for the **data** selection of the **1axis** menu.)

At the completion of data entry for the first circle, select **back** to return to the initial Circular Motion menu.

A second circle may be defined within this program block by entering the required information for the **2dir, 2axis,** and **2data** selections.

Press the "down" arrow for the following additional Circular Motion data entry options.

2CIR, Vel Profile :			**F**		
1 dir	**axes**	**2dir**	**axes**		
ins-Y/N	**del**				**m**

ins-Y/N Press the F1 key of the second Circular motion menu to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

del Press the F2 key of the second Circular motion menu to delete the character at the cursor position.

Press the "down" arrow to return to the previous data entry display.

Varb Select **Varb** to insert a pre-established variable (refer to the Math Mode for variable definition) for a move distance or the feedrate.

quit Select **quit** to abort the command and return to the initial Motion selection menu.

MOTION --> CONTOUR -->2CIR

At the completion of data entry, press the **ENTER** key to write the program block.

Program Block Format:
vel profile (if enabled) direction axis1 axis2 centerpoint1 centerpoint2 feedrate

Example: vel cw X5. Y1. I2. J3. ccw Z1. U2. K2. P2. F100.
or vel G8 G2 X5. Y1. I2. J3. G3 Z1. U2. K2. P2. F100.
; contour motion to produce two circles the first in the clockwise direction and the second in the counterclockwise direction.

MOTION --> CONTOUR --> CIR+LIN

cir+l Select **cir+l** to establish a program block resulting in one axis pair executing a linear move and another axis pair executing a circular move.

The following displays provide the options for producing both a Circular and a Linear Motion command.

```
CIR, + L, Vel Profile :                F
cir          axes        linear        axes

  Cdir       Caxes       Cdata      Laxes      Ldata      F        m
```

Press the "down" arrow for additional Circular and Linear Motion options.

```
CIR, + L, Vel Profile :           F
1 dir        axes       2dir      axes

INS-y/n      del        velpof           Varb      quit      m
```

Cdir Select **Cdir** to establish the direction of the circular move. The selection toggles between Clockwise and Counterclockwise direction. This menu command establishes the **G2** or **G3** command in machine tool language.

Caxes Select **Caxes** to designate the axes to be involved in the circular move. Two axes must be selected. The following selections will be displayed:

```
CIR, + L, Vel Profile :                F
cir          axes        linear        axes

  X        Y        Z        U        data        back
```

MOTION --> CONTOUR -->CIR+LIN

Select the axes to do the circular move. (Two axes must be selected.)

data The end and center points of the circular move may be entered by selecting **data**. (Circular move data may also be entered through the **Cdata** selection of the initial circular and linear motion menu.)

NOTE: **Endpoint and centerpoint data must be established for each axis involved in the circular move.**

The following displays provide the options for entering circular motion data:

CIR + L, Vel Profile :				**F**		
cir	**axes**		**linear**	**axes**		
1end1	**end2**	**cent1**	**cent2**	**Varb**	**back**	**m**

Press the "down" arrow for additional Circular and Linear Motion options.

CIR, + L, Vel Profile::			**F**		
cir	**axes**	**linear**		**axes**	
ins-Y/N	**del**				**m**

1end1 Select **1end1** to establish the endpoint for the first axis in the circular move. Key in the desired endpoint at the cursor location.

end2 Select **end2** to establish the endpoint for the second axis in the circular move. Key in the desired endpoint at the cursor location.

MOTION --> CONTOUR --> CIR+LIN

centl/cent2 Select **centl/cent2** to establish the centerpoint for the first/second axis in the first/second circular move. Key in the desired centerpoint at the cursor location. This will establish the **I, J, K,** or **P** command in machine tool language. This code programs the Offset Vector which is the incremental distance from the beginning of the arc to the arc center. The Offset Vector is determined incrementally regardless of whether the Unidex 400 is in the Absolute (**G90**) or Incremental (**G91**) mode.

Varb Select **Varb** to insert a pre-established variable (refer to the Math Mode for variable definition) for a centerpoint, endpoint or feedrate.

back Select **back** to return to the previous menu.

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

del Press the F2 key of the second Circular and Linear menu to delete the character at the cursor position.

At the completion of data entry for the first circle, select **back** to return to the initial Circular and Linear Motion menu.

CIR, + L, Vel Profile :			**F**			
cir	**axes**	**linear**	**axes**			
Cdir	**Caxes**	**Cdata**	**Laxes**	**Ldata**	**F**	**m**

Cdata The end and center points of the circular move may be entered by selecting **Cdata**. (Data entry with this selection is done in the same way as previously described for the "data" selection of the **Caxes** menu.)

MOTION --> CONTOUR --> CIR+LIN

At the completion of data entry for the circular move, select **back** to return to the initial Circular and Linear Motion menu.

The Linear move is defined within this program block by entering the required information for the **Laxes**, and **Ldata** selections. This menu command establishes the **G1** command in machine tool language.

Laxes Select **Laxes** to designate the axes to be involved in the linear move. Two axes must be selected. The following selections will be displayed:

CIR, + L, Vel Profile :				**F**		
cir	**axes**		**linear**	**axes**		
X	**Y**	**Z**	**U**	**data**	**back**	**m**

Select the axes to do the linear move.

data The distance of the linear move may be entered by selecting **data**. (Linear move data may also be entered through the **Ldata** selection of the initial circular and linear motion menu.)

NOTE: Move distance must be established for each axis involved in the linear move.

The following display provides the options for entering linear motion data:

CIR, + L, Vel Profile:			**F**		
cir	**axes**	**linear**	**axes**		
ins-Y/N	**del**	**axis1**	**axis2**	**Varb**	**back**

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

MOTION --> CONTOUR -->CIR+LIN

axis1 Select **axis1** to establish the distance of the first axis linear move.

axis2 Select **axis2** to establish the distance of the second axis linear move.

Varb Select **Varb** to insert a pre-established variable (refer to the Math Mode for variable definition) for a linear move distance.

back Select **back** to return to the previous menu.

At the completion of data entry for the linear move, select **back** to return to the initial Circular and Linear Motion menu.

```
CIR, + L, Vel Profile                         F
dir          axes          linear           axes


 Cdir         Caxes        Cdata       Laxes      Ldata      F        m
```

Ldata The distance of the linear move may be entered by selecting **Ldata.** (Data entry with this selection is done in the same way as previously described for the **data** selection of the **Laxes** menu.)

F Select **F** to enter the Feedrate that all axes are to move.

Press the "down" arrow for additional Circular and Linear Motion options.

```
CIR, + L, Vel Profile :                  F
cir          axes          linear          axes


INS-Y/N        del         velpof              Varb       quit       m
```

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

MOTION – > CONTOUR – >CIR+LIN– >VELPROF

del Press the F2 key to delete the character at the cursor position.

Velpof Select **Velpof** to enable or disable Velocity Profiling for this Circular move. During Velocity Profiling, priority is given to attaining and maintaining the programmed feedrate, with the secondary emphasis on achieving the end point. If position overshoot occurs, the value of this overshoot is automatically subtracted from the programmed position of the next motion command in the program stream. The system uses the ramp time to accelerate or decelerate from the current velocity to the desired velocity.

If Velocity Profiling is not enabled, the default provides for both acceleration and deceleration to zero velocity.

To maintain smooth transitions from block to block motion, the **G8** command requires that consecutive moves be tangent to each other.

Refer to Appendix A for additional information concerning Velocity Profiling.

Varb Select **Varb** to insert a pre-established variable (refer to the Math Mode for variable definition) for a move distance or the feedrate.

quit Select **quit** to abort the command and return to the initial Motion selection menu.

Press the **ENTER** key to write the program block.

.Program Block Format:
vel profile (if enabled) direction axis1 axis2 centerpoint1 centerpoint2 linear axis1 axis2 feedrate

Example: vel cw X5. Y1. I2. J3. linear Z1. U2. F100.
or vel G8 G2 X5. Y1. I2. J3. G1 Z1. U2. F100.
; contour motion to produce a clockwise circle with axes X and Y and a linear move with axes Z and U, all axes will move at 100 units/second.

MOTION-- > HOME

3-5-1-3: Home

Select **Home** (F3) from the initial Motion menu to send any or all axes to the Home Position. The specified axes will first move to the Home Limit Switch, then move away from the switch toward the Marker, a specified number of machine steps. The characteristics of a Home move are established by the settings of Axis Parameters x02, x03,x04,x05, x06, and x07.

This command establishes the **G7** command in machine tool language.

A Software Home position **(G92)** may also be established through this menu selection.

The following selection menu will be displayed when **Home** is selected from the initial motion menu:

```
HARDWARE HOME, OR SET AXES POSITION
which axes go home:

X      Y      Z      U      Posreg      quit      m
```

Press the "down" arrow for additional Home options.

```
HARDWARE HOME, OR SET AXES POSITION
which axes go home:

ins-Y/N      del                                  m
```

MOTION-- > HOME

X-U Press the appropriate Function key to select the axes to be sent Home.

Posreg Select **Posreg** to establish a software Home position at the current position of either all or the specified axes. This command establishes the **G92** command in machine tool language.

The display will be:

```
SET AXES POSITION  REGISTER
X                        Z
Y                        U

X       Y       Z       U       all 0     Varb     m
```

Press the "down" arrow for additional Position Register options.

```
SET AXES POSITION  REGISTER
X                        Z
Y                        U

ins-Y/N      del                         quit      m
```

X-U Press the appropriate Function key to establish a software reference position for an axes.

all 0 Select **all 0** to set the software reference position of all axes to zero.

Varb Select **Varb** to insert a pre-established variable (refer to the Math Mode for variable definition) for an axes software position.

quit Select **quit** to abort the command and return to the initial Motion selection menu.

MOTION-- > HOME

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

Press the **ENTER** key to write the program block.

Program Block Format: home axes name

Example: home X Y or G7 XY
; the X and Y axes are sent Home simultaneously at a feedrate established by an axis parameter

Example: reset relative posn X2. Y3. or G92 X2. Y3.
; sets a software home position for the X Axis as 2inches/millimeters and Y as 3 inches/millimeters. All other axes remain the same.

Example: reset relative posn or G92
; sets a software home position for all axes as zero

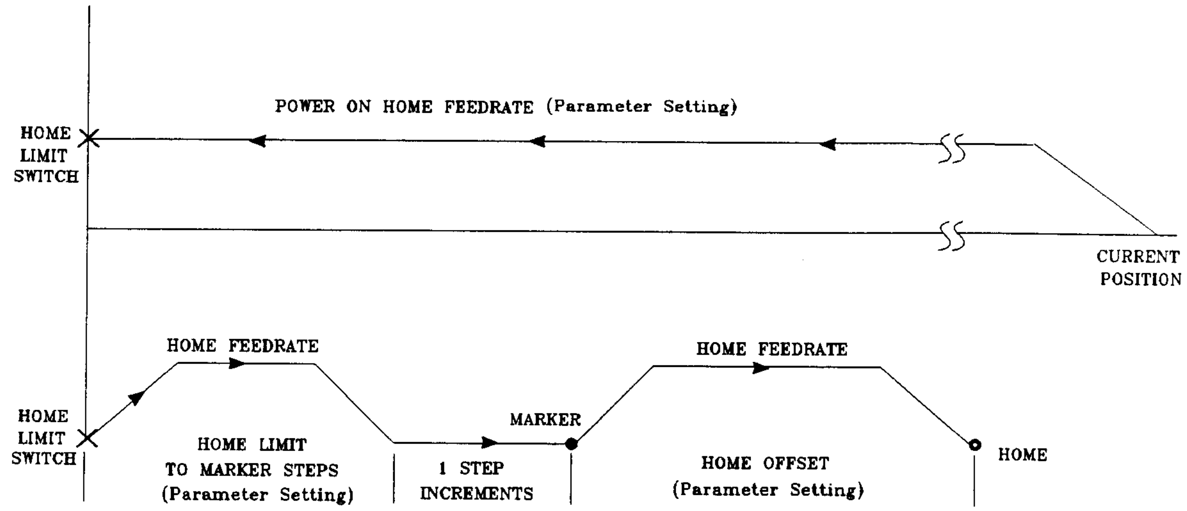


Figure 3-1: Illustration of the Home Command Following a Power Up

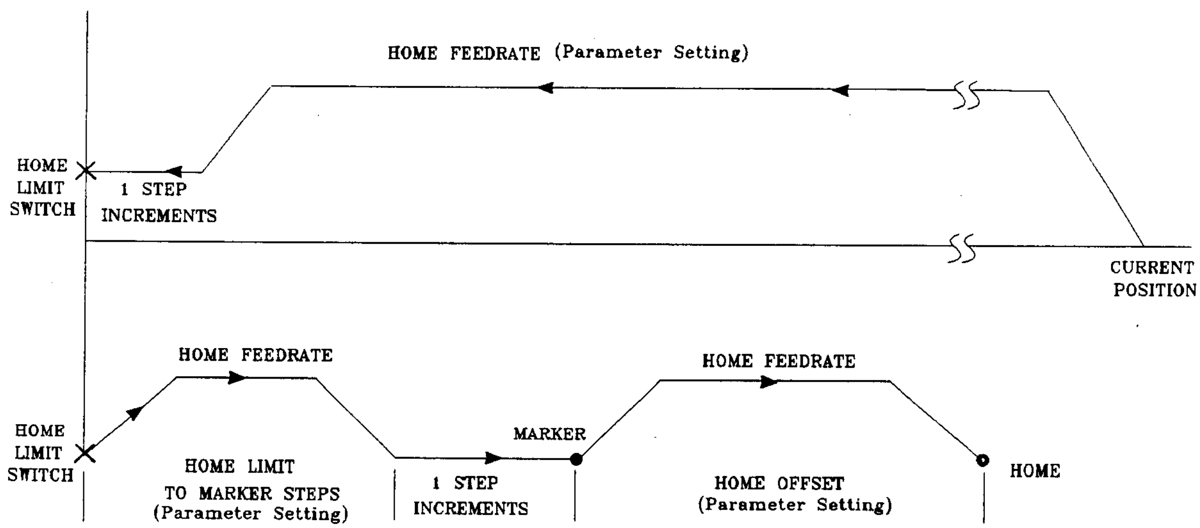


Figure 3-2: Illustration of the Home Command From a Known Position

MOTION-- > FREE

3-5-1-4: free Select **free** from the initial Motion menu to establish non-synchronized motion for any or all axes. This command provides the User the ability to perform a non-contour move almost simultaneously with a contour move.

This command establishes the **G10** command in machine tool language. The display will be:

```
      > 0: (program)
      > 1:

rotate      1shot      rept      #_rept    stop     quit     m
```

Press the "down" arrow for additional Free Run options.

```
      > 0: (program)
      > 1:

output     no       status       wait                         m
```

rotate Select **rotate** to establish continuous non-synchronized motion in one direction at a specified feedrate.

The display will be;

```
    free run, rotation case, axis =
    +/- direction = +/-          feedrate =

    X        Y        Z        U       Varb      quit       m
```

Press the "down" arrow for additional Free Run - Rotate options.

```
    free run, rotation case, axis =
    +/- direction = +/-          feedrate =

    ins-Y/N        del                  Varb     quit        m
```

MOTION-- > FREERUN-- > ROTATE

X-U Press the appropriate Function key to select the axis to perform the **Rotate** motion.

The following menu selection screen will be displayed:

```
free run, rotation case, axis =
+/- direction = +/-          feedrate =

ins-Y/N     del                    Varb    quit        m
```

The cursor will be at the feedrate insertion point. Key in the desired feedrate.

Use the +/- key to toggle between (+) positive direction (CW) or (-) negative direction (CCW) for the rotating axis.

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

Varb Select **Varb** to insert a pre-established variable (refer to the Math Mode for variable definition) for the "Rotate" feedrate.

quit Select **quit** from the second **Rotate** menu to return to the initial **Rotate** selection menu. Select **quit** from the initial **Rotate** menu to abort the command and return to the initial Freerun selection menu

Press the **ENTER** key to write the program block.

Program Block Format: free: freerun type axis and case feedrate direction

Example: Free: cont X1 F200. or G10 X1 F200. ; continuous free run motion in CW direction at a rate of 200 units/min
Free: cont X1 F200. or G10 X1 F-200. ; continuous free run motion in CCW direction at a rate of 200 units/min

MOTION-- > FREERUN-- > 1SHOT

1shot Select **1shot** to establish non-synchronized motion in one direction, for a specified distance, at a specified feedrate.
This command establishes the **G10** command in machine tool language.

The display will be:

```
free run, 1shot case, axis =
+/- direction = +/-
feedrate =                          distance=
X      Y      Z      U      Varb      quit      m
```

Press the "down" arrow for additional **Free Run - 1 Shot** options.

```
free run, 1shot case, axis =
+/- direction = +/-
feedrate =                          distance=
  ins-Y/N    del    F    dist    Varb    quit    m
```

X-U Press the appropriate Function key to select the axis to perform the **1shot** motion.

The following menu selection screen will be displayed:

```
free run, 1shot case, axis =
+/- direction = +/-
feedrate =                          distance=
  ins-Y/N    del    F    dist    Varb    quit    m
```

The cursor will be at the feedrate insertion point. Key in the desired feedrate.

Use the +/- key to toggle between (+) positive direction (CW) or (-) negative direction (CCW) for the axis.

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

MOTION-- > FREERUN-- > 1SHOT

del Press the F2 key to delete the character at the cursor position.

F Select **F** to enter the axis Feedrate if it has not previously been entered.

dist Select **dist** to enter the distance, in machine steps, that the axis is to travel.

Varb Select **Varb** to insert a pre-established variable (refer to the Math Mode for variable definition) for the "1shot" feedrate or distance.

quit Select **quit** from the second **1shot** menu to return to the initial **1shot** selection menu. Select **quit** from the initial **1shot** menu to abort the command and return to the initial **Freerun** selection menu

Press the **ENTER** key to write the program block.

Program Block Format:

free: freerun type axis and case feedrate direction distance (machine steps)

Example: free: 1shot Y2 F100. D4. or G10 Y2 F100. D4.
;free run motion in a CW direction for a distance of 4 machine steps
at a rate of 200 units/min

Example: free: 1shot Y2 F-100. D4. or G10 Y2 F-100. D4.
;free run motion in a CCW direction for a distance of 4 machine steps
at a rate of 200 units/min

MOTION-- > FREERUN-- > REPT

rept Select **rept** to establish non-synchronized, continuous back-and-forth motion, at a specified initial direction, for a specified distance, at a specified feedrate.

The display will be:

```
free run, back-forth no stop, axis =
+/- direction = +/-
feedrate =                              distance=
X        Y        Z        U        Varb      quit        m
```

Press the "down" arrow for additional Free Run - Repeat options.

```
free run, back-forth no stop, axis =
+/- direction = +/-
feedrate =                              distance=
ins-Y/N     del       F       dist      Varb      quit        m
```

X-U Press the appropriate Function key to select the axis to perform the **rept** motion.

The following menu selection screen will be displayed:

```
free run, back-forth no stop, axis =
+/- direction = +/-
feedrate =                              distance=
ins-Y/N     del       F       dist      Varb      quit        m
```

The cursor will be at the feedrate insertion point. Key in the desired feedrate.

Use the +/- key to toggle between (+) positive direction (CW) or (-) negative direction (CCW) for the rotating axis.

MOTION-- > FREERUN-- > REPT

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

F Select **F** to enter the axis Feedrate if it has not previously been entered.

dist Select **dist** to enter the distance, in machine steps, that the axis is to travel in one direction.

Varb Select **Varb** to insert a pre-established variable (refer to the Math Mode for variable definition) for the "rept" feedrate or distance.

quit Select **quit** from the second **rept** menu to return to the initial **rept** selection menu. Select **quit** from the initial **rept** menu to abort the command and return to the initial **Freerun** selection menu.

Press the **ENTER** key to write the program block.

Program Block Format:

free: freerun type axis and case feedrate direction distance (machine steps)

Example: free: rept Z3 F400. D4. or G10 Z3 F400. D4.
; continuous back-and-forth motion,starting in the CW direction, for a distance of 4 machine steps at a rate of 400 units/min

free: rept Z3 F-400. D4. or G10 Z3 F400. D4.
; continuous back-and-forth motion,starting in the CCW direction, for a distance of 4 machine steps at a rate of 400 units/min

MOTION-- > FREERUN-- > #_REPT

#_rept Select **#_rept** to establish non-synchronized, back-and-forth motion, for a specified number of cycles. The initial direction, distance, and feedrate are all User specified.

The display will be:

```
free run, back-forth # cycle, axis =
+/- direction = +/-              #_cycle =
feedrate =                       distance=
X      Y      Z      U      Varb      quit      m
```

Press the "down" arrow for additional **Free Run - #_Repeat** options.

```
free run, back-forth # cycle, axis =
+/- direction = +/-              #_cycle =
feedrate =                       distance=
ins-Y/N   del   F   dist   cycle   quit   m
```

X-U Press the appropriate Function key to select the axis to perform the **#_rept** motion.

The following menu selection screen will be displayed:

```
free run, back-forth # cycle, axis =
+/- direction = +/-              #_cycle =
feedrate =                       distance=
ins-Y/N   del   F   dist   cycle   quit   m
```

The cursor will be at the feedrate insertion point. Key in the desired feedrate.

Use the +/- key to toggle between (+) positive direction (CW) or (-) negative direction (CCW) for the rotating axis.

MOTION-- > FREERUN-- > #_REPT

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

F Select **F** to enter the axis Feedrate if it has not previously been entered.

dist Select **dist** to enter the distance, in machine steps, that the axis is to travel in one direction.

cycle Select **cycle** to insert a the number of cycles to be performed.

quit Select **quit** from the second **#_rept** menu to return to the initial **#_rept** selection menu. Select **quit** from the initial **#_rept** menu to abort the command and return to the initial **Freerun** selection menu

Press the **ENTER** key to write the program block.

Program Block Format:
free: axis and case +number of cycles feedrate direction distance (machine steps)

Example: free: X4 +5 F100. D3 or G10 X4 +F100. D3.
; 5 cycles of back-and-forth motion for a distance of 3 machine steps (starting in the CW direction) at a rate of 100 units/min

Example: free: X4 +5 F-100. D3 or G10 X4 +F-100. D3.
; 5 cycles of back-and-forth motion for a distance of 3 machine steps (starting in the CW direction) at a rate of 100 units/min

MOTION-- > FREERUN-- > STOP

stop Select **stop** to stop Freerun motion for any or all axes.

The display will be:

```
stop  axes free run, axes =

X      Y      Z      U      del      quit
```

X-U Press the appropriate Function key to stop the free run of that axes.

del Press the F5 key to delete the character at the cursor position.

quit Select **quit** to abort the command and return to the initial **Freerun** selection menu.

Press the **ENTER** key to write the program block.

Program Block Format: free: stop axis zero

Example: free: stop X0. or G10 X0 ; stop the free run for the X axis

MOTION-- > FREERUN-- > OUTPUT I/O

Press the "down" arrow from the initial **Freerun** selection menu for additional Freerun options.

```
>0: (program)
>1:

output   no   status   wait
```

output The Unidex 400 is factory configured to output data through either a 8 out, 12in I/O bus or a 16 out MST bus. The selections available from the **Freerun Output** menu , are dependent upon the I/O configuration of the system being used.

This command establishes the **G11** command in machine tool language.

A description of both of the **Output** menu types follows:

I/O Bus Output

Select **output** to send data to the I/O bus , at the completion of a Freerun of the specified axes. Up to eight bits may be set by this command.

```
set output at freerun done, axis =
output bit =

X   Y   Z   U   Varb   quit   m
```

Press the "down" arrow for additional Free Run I/O Output options.

```
set output at freerun done, axes =
output bit =

ins-Y/N   del   high   low   skip   quit   m
```

MOTION-- > FREERUN-- > OUTPUT
I/O

X-U	Press the appropriate Function key to establish the bit pattern to be output at the conclusion of the freerun of that axes.
Varb	Select **Varb** to insert a pre-established variable (refer to the Math Mode for variable definition) for the bit setting.
quit	Select **quit** to abort the command and return to the initial Freerun selection menu.
ins-Y/N	Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.
del	Press the F2 key to delete the character at the cursor position.
high	Select **high** to establish a bit setting of "1".
low	Select **low** to establish a bit setting of "0".
skip	Select **skip** to allow a bit setting to remain unchanged.
quit	Select **quit** from the second **output** menu to return to the initial output selection menu. Select **quit** from the initial **output** menu abort the command and return to the initial Freerun selection menu

Press the **ENTER** key to write the program block.

Program Block Format: free done, output axis bit pattern

Example: free done, output X HHHHLLL* or G11 X HHHHLLL*
; at the completion of the X axis free run 1111000* will be output on the I/O bus, * indicates bit remains the same

MOTION-- > FREERUN-- > OUTPUT

M, S, T Output

Select **output** to send data to the M,S, T bus, at the completion of a Freerun of a specified axes. Up to four sets of M, S, T outputs may be specified by this command.

The duration of the output and the delay time to receive an acknowledge, is established in the Setup-General Parameter Mode (see Chapter 7 of this manual).

The display will be:

```
set output at freerun done, axis =
at most 4 sets =

X        Y        Z        U    Varb    quit    m
```

Press the "down" arrow for additional Free Run I/O Output options.

```
set output at free run done, axis =
at most 4 sets =

ins-Y/N    del    M    S    T    quit    m
```

X-U Press the appropriate Function key to select the axis that will send an output on the MST Bus at the conclusion of a Free Run.

Varb Select **Varb** to insert a pre-established variable (refer to the Math Mode for variable definition) for the bit setting.

The following menu selection screen will be displayed:

```
set output at free run done, axis =
at most 4 sets =

ins-Y/N    del    M    S    T    quit    m
```

MOTION-- > FREERUN-- > OUTPUT

MST

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

M Select **M** to establish an M function output at the conclusion of the freerun of the selected axis. M(0-999)

S Select **S** to establish an S function output at the conclusion of the freerun of the selected axis. S(0-999)

T Select **T** to establish an T function output at the conclusion of the freerun of the selected axis. T(0-999)

quit Select **quit** from the second **output** menu to return to the initial **output** selection menu. Select **quit** from the initial "output" menu to abort the command and return to the initial **Freerun** selection menu

Certain M codes (M0-M50) are assigned a command function, if an output of this designation is required, then a zero must precede the bit specification.

Example: An output of M0 is entered as M00

Press the **ENTER** key to write the program block.

Program Block Format: free done, output axis bit pattern

Example: free done, output X MV100M80S254 or G11 X MV100M80S254
; Upon the completion of the freerun of the X axis, the bit pattern contained in Variable 100, M80 and S254 will be output on the MST Bus.

MOTION-- > FREERUN-- > NO

no Select **no** to cancel data output for a specified axis following the Freerun.

This command establishes the **G12** command in machine tool language.

The following menu selection screen will be displayed:

cancel output, axes =					
X	**Y**	**Z**	**U**	**del**	**quit**

X-U Press the appropriate Function key to cancel an output at the conclusion of a Free Run of that axis.

del Press the F2 key to delete the character at the cursor position.

quit Select **quit** to abort the command and return to the initial Free Run selection menu.

Press the **ENTER** key to write the program block.

Program Block Format: freerun status axis

Example: stop free, output XY or G12 XY ;at the conclusion of the freerun of axes X and Y's there is no Bus output

MOTION-- > FREERUN-- > STATUS

status Select **status** to save the freerun status to a variable. The freerun status specifies the axes that are currently involved in a freerun. The status information is in binary format which is transcribed to integer format for storage within the variable, ie:

[(bits 8-31 always clear)7 6 5 4 3 2 1 0]	(H,Variable is hex format)
[0 0 0 0 2^3 2^2 2^1 1]	Binary status data
[0 0 0 0 U Z Y X]	status of axis free run stored to "least significant bit"

Example: If axes X, Y, and Z are in a free run, the status will be "7" (2^2 +2 + 1)

This command establishes the **G14** command in machine tool language. The following menu selection screen will be displayed:

```
free run status to Variable =

ins-Y/N        del                                   quit
```

Key in the name of the Variable to store the free run status.

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

quit Select **quit** to abort the command and return to the initial Freerun selection menu.

Press the **ENTER** key to write the program block.

Program Block Format: set freerun status variable number
Example: set free status V23 or G14 V23 ; send the axes free run status to variable 0023

MOTION-- > FREERUN-- > WAIT

wait Select **wait** to pause the program run until the specified axes have completed their free run.

This command establishes the **G13** command in machine tool language.

The following menu selection screen will be displayed:

wait free run done, axes =					
X	**Y**	**Z**	**U**	**del**	**quit**

X-U Press the appropriate Function key to pause the program run until the conclusion of the Free Run of that axis.

del Press the F2 key to delete the character at the cursor position.

quit Select **quit** to abort the command and return to the initial Free Run selection menu.

Press the **ENTER** key to write the program block.

Program Block Format: wait for freerun axis name

Example: wait for free XY or G13XY ; program will pause until completion of the freerun of axes X and Y.

I/O

3-5-2: I/O

The **I/O** selection provides the User with the ability to select the format of the Input/Output data.

The Unidex 400 is factory configured to input and output data through either an 8 out, 12in I/O bus or a 16 out MST bus. The selections available from the I/O selection menus , are dependent upon the I/O configuration of the system being used.

A description of both of the Output menu types follows:

I/O Bus Output

Inputs and Outputs data through the CPU interface bus. This bus is a general purpose 8 out, 12 in, non-opto isolated parallel interface. (Refer to the Chapter 11: Hardware for a pin-out of the CPU/OPTO 22 Bus connector.)

Program data for the I/O port can be in one of these formats:

HEX, DEC, FFP,
BCD, BIT.

A brief description of each follows:

HEX : base 16, Hexadecimal format. Specified with a "H," or "#H:" placed before a variable name, depending on the application. (Refer to the BTF and FTB commands within the Math Mode for additional information.) Hex numbers are 32 bits long.

DEC : base 10, default mode. No prefix is required.

FFP : Fast Floating Point Format. All variables are in this format. (see the BTF and FTB commands within the Math Mode). Floating numbers have a 24 bit mantissa, 7 bit exponent, and 1 sign bit, providing more than six decimal places of precision.

BCD : Binary Coded Decimal. Each decimal digit is represented by a unique arrangement of binary digits (usually four).

BIT : specified as "H" for High or logic 1, "L" for Low or logic 0, or "*" for "don't care".

I/O-- > I-BCD
I/O BUS

Following the selection of **insrt** or **alter** from Edit Screen #1 the display will be:

```
>0: (program)
>1:

motion   I/O   mode   flow   servo   quit   m
```

Select **I/O** to establish the format to input or output data.

The displays will be:

```
I-bcd   I-dec   I-hex   O-bit   O-ffp   quit   m
```

Press the "down" arrow for additional **I/O** options.

```
OVhex   O-dec   O-hex   O-cont   O-stop   m
```

I-bcd This selection specifies that BCD (Binary Coded Decimal) format is to be used to interpret data input to the designated variable location.

This command establishes the **G39** command in machine tool language.

I/O-- > I-BCD

I/O BUS

Following this selection the display will be:

```
input BCD format to Variable =

ins-Y/N     del                                  quit
```

Key in the variable location from which data will be interpreted as being in BCD format (up to four characters).

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

Press the **ENTER** key to write the program block.

Program Block Format: input to variable number format

Example: input to V100 (BCD format) or G39 B100 ; input to variable 100 is to be interpreted in BCD format.

I/O-- > I-DEC

I/O BUS

I-dec This selection specifies that DEC (Decimal) format be used to interpret input to the designated variable location.

This command establishes the **G39** command in machine tool language.

Following this selection the display will be:

```
input DEC format to Variable =


ins-Y/N     del                                   quit
```

Key in the variable location from which data will be interpreted as being in DEC format (up to four characters).

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

Press the **ENTER** key to write the program block.

Program Block Format: input to variable number format

Example: input to V300 (DEC format) or G39 D300 ; input to variable 300 is to be interpreted in DEC format.

I/O-- > I-HEX
I/O BUS

I-hex This selection specifies that Hex (Hexadecimal) format is to be used to interpret data input to the designated variable location.

This command establishes the **G39** command in machine tool language.

Following this selection the display will be:

input HEX format to Variable =		
ins-Y/N	**del**	**quit**

Key in the variable location from which data will be interpreted as being in Hex format (up to four characters).

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

Press the **ENTER** key to write the program block.

Program Block Format: input to variable number format

Example: input to V200 (HEX format) or G39 H2 ; input to variable 200 is to be interpreted in Hex format.

SAMPLE PROGRAM

The following sample program establishes and displays the formats designated by the previous examples of variables 100, 300 and 200.

Unidex 400	or	Unidex 400
Filename: 0204		Filename: 0204
%0: dwell until input = HLHLHLHL		%G4 IHLHLHLHL
1: dwell 100 ms		G4 F100
2: dwell until cycle start		G4 C
3: input to V100 (BCD format)		G39 B100
4: input to V300 (DEC format)		G39 D 300
5: input to V200 (HEX format)		G39 H200
6: MESG #V0		G21 #V0
7: dwell until cycle start		G4 C
8: MESG #V1		G21 #V1
9: dwell until cycle start		G4 C
10: MESG #H:V2		G21 #H:V2
11: dwell until cycle start		G4 C
12: MESG #V2		G21 #V2
13: dwell until cycle start		G4 C
14: end prog		M2
15:		

I/O-- > O-BIT

I/O BUS

O-bit This selection is used to establish the output bit pattern.

This command establishes the **G38** command in machine tool language.

Following this selection the display will be:

output bit pattern =					
ins-Y/N	**del**	**high**	**low**	**skip**	**quit**

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

high Select **high** to establish a bit setting of "1". (Bit pattern entries are always right justified.)

low Select **low** to establish a bit setting of "0". (Bit pattern entries are always right justified.)

skip Select **skip** to allow a bit setting to remain unchanged or "don't care". (Bit pattern entries are always right justified.)

quit Select **quit** to abort the command and return to the initial **I/O** selection menu.

Program Block Format: output bit pattern

Example: output bit pattern HHHHLLL* or G38 HHHHLLL*
; 1111000* will be output on the I/O bus, * indicates bit remains the same

NOTE: Output bit pattern HL will be read as ******HL.

I/O – > Vffp

I/O BUS

OVffp This selection is used to establish that the output from a designated variable will be assumed to be in FFP (Fast Floating Point) format. Any fraction is rounded to the nearest integer and then output to the I/O bus.

This command establishes the **G34** command in machine tool language.

Following this selection, the display will be:

output 1 byte of FFP Variable =		
ins-Y/N	**del**	**quit**

Key in the Variable location to output data in FFP (Fast Floating Point) format.

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

quit Select **quit** to abort the command and return to the initial I/O selection menu

Press the **ENTER** key to write the program block.

Program Block Format: output byte variable number

Example: output byte V0203 (ffp format) or G34 0203
; output from variable location 0203 is to be interpreted in FFP format.

I/O-- > O-VHEX

I/O BUS

O-Vhex This selection is used to establish that the output from the designated variable will be assumed to be in Hex (Hexadecimal) format. It will be output directly to the I/O bus.

This command establishes the **G36** command in machine tool language.

Following this selection, the display will be:

output 1 byte of HEX Variable =		
ins-Y/N	**del**	**quit**

Key in the Variable location to output data in HEX (Hexadecimal) format.

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

quit Select **quit** to abort the command and return to the initial I/O selection menu

Press the **ENTER** key to write the program block.

Program Block Format: output byte variable number

Example: output byte V3456 (hex format) or G36 V3456
; output from variable location 3456 is to be interpreted in HEX format.

I/O-- > O-DEC
I/O BUS

O-dec This selection establishes decimal data for immediate output. Decimal values 0 through 255 may be output.

This command establishes the **G37** command in machine tool language.

Following this selection, the display will be:

```
output DEC pattern =

ins-Y/N        del                              quit
```

Key in the decimal data for output.

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

Press the **ENTER** key to write the program block.

Program Block Format: decimal output byte

Example: decimal output 123 or G37 123
; initiates an immediate output of the decimal 123 output

I/O-- > O-HEX

I/O BUS

O-hex This selection establishes HEX (Hexadecimal) data for immediate output. Hex values 0 through FF may be output.

This command establishes the **G35** command in machine tool language.

Following this selection, the displays will be:

output HEX pattern =						
A	**B**	**C**	**D**	**E**	**F**	**m**

Press the "down" arrow for additional Hex data output options.

output HEX pattern =			
ins-Y/N	**del**	**quit**	**m**

A-F Select the appropriate letter(s) or key in the number(s) for Hex data output.

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

Press the **ENTER** key to write the program block.

Program Block Format: HEX output

Example: HEX out AA or G35 AA
; initiates an immediate output of the Hexadecimal AA

I/O-- > OCONT

I/O BUS

Ocont This selection specifies a bit pattern, to be continuously output while the program is running, and feedhold is not active.

This command establishes the **G25** command in machine tool language. Following this selection, the display will be:

cont/feedhold off, output =					
ins-Y/N	**del**	**high**	**low**	**skip**	**quit**

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

high Select **high** to establish a bit setting of "1". (Bit pattern entries are always right justified.)

low Select **low** to establish a bit setting of "0". (Bit pattern entries are always right justified.)

skip Select **skip** to allow a bit setting to remain unchanged or "don't care". (Bit pattern entries are always right justified.)

quit Select **quit** to abort the command and return to the initial I/O selection menu.

Press the **ENTER** key to write the program block.

Program Block Format: output type bit pattern

Example: output during run HHHHLLL* or G25 HHHHLLL*
; 1111000* will be continuously output during the program run, * indicates bit remains the same

NOTE: Output bit pattern HL is read as ******HL.

I/O-- > OSTOP

I/O BUS

Ostop This selection specifies a bit pattern, to be output when the program run is stopped and feedhold is activated.

This command establishes the **G26** command in machine tool language. Following this selection, the display will be:

```
stop/feedhold on, output =

ins-Y/N      del      high      low      skip      quit
```

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

high Select **high** to establish a bit setting of "1". (Bit pattern entries are always right justified.)

low Select **low** to establish a bit setting of "0". (Bit pattern entries are always right justified.)

skip Select **skip** to allow a bit setting to remain unchanged or "don't care". (Bit pattern entries are always right justified.)

quit Select **quit** to abort the command and return to the initial **I/O** selection menu.

Press the **ENTER** key to write the program block.

Program Block Format: output type bit pattern

Example: output during stop HHHHLLL* or G26 HHHHLLL*

; 1111000* will be output when the program run is stopped, * indicates bit remains the same

NOTE: Output bit pattern HL is read as ******HL.

I/O

MST BUS

M, S, T Output

The MST bus provides 16 output lines and 1 acknowledgment line.

Following the selection of **insrt** or **alter** from Edit Screen #1 the display will be:

```
>0: (program)
>1:

motion    I/O    mode    flow    servo    quit    m
```

Select **I/O** to establish the MST output data.

The displays will be:

```
M-out    S-out    T-out                    quit    m
```

Press the "down" arrow for additional I/O options.

```
M-stb    S-stb    T-stb    M-ack    S-ack    T-ack    m
```

I/O

MST BUS

Select **M-out, S-out**, or **T-out** to establish the data to be output on one of these lines. The display will be:

M/S/T function output =						
A	**B**	**C**	**D**	**E**	**F**	**m**

Press the "down" arrow for additional options.

M/S/T function output =			
ins-Y/N	**del**	**quit**	**m**

Key in or select the 4 bit Hex characters to be output to the M, S or T bus.

NOTE: **M2 and M47 are assigned special functions in Machine Tool code. If M2 or M47 output is required, it is necessary to enter them as three digits.**

Example: M2 ; signifies the end of the program
M002 ; permits M2 to be output

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

quit Select **quit** to abort the command and return to the initial I/O selection menu.

Press the **ENTER** key to write the program block.

Program Block Format: M/S/T-output 4 bit Hex character

Example: M12DD ; M function output will be 12DD

NOTE: **Output AA is read as **AA.**

I/O

MST BUS

STROBE DELAY

The **M-stb, S-stb, or T-stb** commands establish the length of time that the M,S, or T signal will remain ON to allow an external device time to respond. The time may be set at 0 - 65535 mSec. If the time is set at "0" the output will be sent for 5 microseconds.

These selections will override the general parameter settings for this program only. The parameter settings are not changed by this command entry. Refer to the Setup Chapter of this manual for additional parameter information.

These commands establish the **G58** command in machine tool language.

Select **M-stb, S-stb, or T-stb** from the second I/O MST menu to establish the length of time that the M, S or T output signal will remain ON

The display will be:

M/S/T strobe delay =		
ins-Y/N	**del**	**quit**

Key in the strobe delay time for the M, S or T function (0- 65535).

ins-Y/N Press the F1 key to toggle character insert ON (Y) or Off (No). When ON, characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

quit Select **quit** to abort the command and return to the initial selection menu.

Press the **ENTER** key to write the program block.

Program Block Format: MST length of signal (mSec)

Example: MST strobe M100 or G58 M100
; M function output will be 100 mSec in duration

I/O

MST BUS

ACKNOWLEDGE DELAY

The **M-ack, S-ack, or T-ack** commands establish the length of time that the Unidex 400 will wait for an Acknowledge of the M,S, or T output signal. The time may be set at 0 - 65535 mSec. If the time is set at "0", it is interpreted as "no Acknowledge needed." If the time is set at 65535, the Unidex 400 will scan indefinitely and check every 1 mSec for the Acknowledge signal.

These selections override the general parameter settings for this program only. The parameter settings are not changed by this command entry. Refer to the Setup Chapter of this manual for additional parameter information.

These commands establish the **G59** command in machine tool language.

Select **M-ack, S-ack, or T-ack** to establish the length of time that the Unidex 400 will wait for an Acknowledge.

The display will be:

```
M/S/T ack delay =


ins-Y/N        del                              quit
```

Key in the Acknowledge time for the M, S or T function (0-65535).

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (No). When ON, characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

quit Select **quit** to abort the command and return to the initial selection menu.

Press the **ENTER** key to write the program block.

Program Block Format: MST type of Acknowledge wait (mSec)

Example: MST ack delay S200 or G59 S200

; the Unidex 400 will wait for a Strobe Acknowledge signal for 200 mSec

3-5-3: MODE

The **mode** selection provides the User with the ability to create the format of the program or program block.

Following the selection of **insrt** or **alter** from Edit Screen #1 the display will be:

```
>0: (program)
>1:

motion    I/O    mode    flow    servo    quit    m
```

Select **mode** to establish the program or program blocks format.

The display will be:

```
>0: (program)
>1:

abs    inc    metric        english    quit    m
```

Press the "down" arrow for additional options.

```
>0: (program)
>1:

corner        n-corn        filter        m
```

An explanation of each of the modes follows:

MODE -- > ABS

MODE -- > INC

abs Select **abs** to initiate Absolute programming. When absolute positioning is in effect, all requested moves are referenced from the previously established software home location.

This command establishes the **G90** command in machine tool language.

Press the **ENTER** key to write the program block.

Example: absolute mode or G90

; program moves are referenced from the software home location.

inc Select **inc** to initiate Incremental programming. When incremental positioning is in effect, all requested moves are referenced from the previous position.

This command establishes the **G91** command in machine tool language.

Press the **ENTER** key to write the program block.

Example: incremental mode or G91

; program moves are referenced from the previous position.

MODE -- > METRIC

MODE -- > ENGLISH

metric Select **metric** to write all positioning values in the Metric format. The default for this selection is established in the General Parameter mode. This command selection overrides the parameter setting for this program only. Refer to the Setup Chapter of this manual for additional parameter information.

This command establishes the **G71** command in machine tool language.

Press the **ENTER** key to write the program block.

Example: Metric Mode or G71
; all position values are expressed in Metric format.

english Select **english** to write all positioning values in the English format. The default for this selection is established in the General Parameter mode. This command selection overrides the parameter setting for this program only. Refer to the Setup Chapter of this manual for additional parameter information.

This command establishes the **G70** command in machine tool language.

Press the **ENTER** key to write the program block.

Example: English Mode or G70
; all position values are expressed in English format.

quit Select **quit** to abort the command and return to the initial selection menu.

MODE – > CORNER
MODE – > N-CORN

corner Select **corner** to activate the corner rounding mode. While in this mode the Unidex 400 sets the trajectory generator to execute all subsequent motion commands at the commanded feedrate, disregarding any predefined "ramptime" settings. The trajectory generator will not wait for the given axes to stabilize on the final commanded position of the just completed motion command before sending the next motion command(s) to the Servo Loops. This command is most often used when several consecutive moves are requested.

Refer to Appendix A for additional information concerning the corner rounding mode.

This command establishes the **G23** command in machine tool language.

Press the **ENTER** key to write the program block.

Example: corner round on or G23
; Unidex 400 will begin the next motion
command when the previous command is nearing completion.

n-corn Select **n-corn** to activate the non-corner rounding mode. While in this mode the "ramp time" is enabled allowing acceleration/deceleration to take effect between two adjacent motion commands with differing feedrates. Unidex 400 waits for the completion of the requested move (count zero) before processing the next commands.

Refer to Appendix A for additional information concerning the corner rounding mode.

This command establishes the **G24** command in machine tool language.

Press the **ENTER** key to write the program block.

Example: corner round off or G24
; Unidex 400 will not begin a move until the previous
move is complete

MODE – >FILTER

filter Select **filter** to establish the level of digital filtering which can occur on the milisecond position/velocity command being sent to the Unidex 400's DSP Servo Control card. This selection overrides but does not change the default value established by General Parameter 40. This setting is modal, and remains until changed or the system is reset.

The Unidex 400 provides a high performance Proportional, Integral, Derivative Servo Loop that also implements two feedforward gains: velocity and acceleration. As a result, the system follows (tracks) very closely to the actual command with minimal servo following error or lag. When moves are programmed with tangency, i.e. corners are programmed with small arcs, the Unidex 400 will smoothly follow the path using the G8 Velocity Profiling mode.

However a large number of CNC's exist that use an error based type of servo loop closure. For these cases programmed moves must not necessarily have tangency and corners but may be programmed as two linear moves. When executed with the G23 Corner Rounding selection, these systems will inherently round the corners and provide relatively continuous velocity contouring even with non-tangent moves. Refer to Appendix A for further information on Velocity Profiling and Corner Rounding

Selection of the **filter** command in conjunction with appropriate setting of General Parameter 41 allows the Unidex 400 to digitally replicate the more traditional CNC type control functionality.

The digital filter operates with the following relationships:

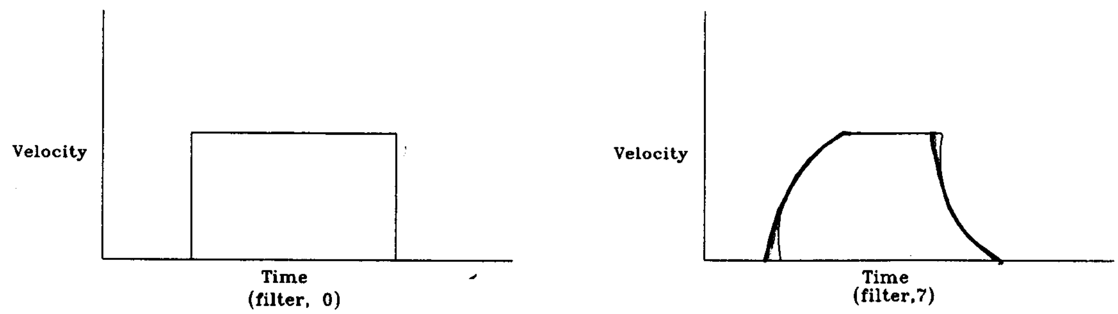
Filter Total	=	Filter + Command In
Command Out	=	Filter /2n; (N equals corner round filter #)
Filter Total	=	Filter Total - Command Out

Use of the **filter** command provides an exponential ramp to the motion, eliminating the need for **ramp** values except as a supplement (refer to General Parameter 41).

MODE – > FILTER

Values for the **filter** selection range from 0 (no filter) to 7 (maximum filter). For standard systems a value of 4, 5, or 6 provides the best performance. Higher filter values result in more corner rounding.

An illustration of the effect of the **filter** value on a point-to-point move with no ramping during Corner Rounding is provided below.



The filter must not be turned **OFF** (set to zero) while the axes are moving or a sharp velocity transition may result. It may be turned **OFF** only following a **G9** or **G24** type motion. Filter status is normally latched and cannot be changed until motion is stopped.

Select **filter** to establish the program or program blocks format.
The display will be:

```
corner round filter # =

ins-Y/N     del                    Varb     quit
```

Key in the desired filter number.

Select **ins-Y/N** or **del** to insert or delete a character at the cursor location.

Select **Varb** to insert a pre-established variable (refer to the Math Mode for variable definition) for the **filter** setting.

MODE – > FILTER

Select **quit** to abort the command and return to the initial **servo** selection menu.

Press the **ENTER** key to write the program block.

Program Block Format: filter level

Example: filter 5 or G6 6 5
; the corner rounding filter is set at level 5

3-5-4: FLOW

The **FLOW** selection provides the User with the ability to define the direction of the program. Sub-menus within this selection may be used to direct the program to another location, perform the defined program blocks and then return to the original point in the program and continue to process the program. Other selections allow the creation of program loops, program titles and line labels. The following sections describe these functions.

Following the selection of **insrt** or **alter** from Edit Screen #1, the display will be:

```
>0: (program)
>1:

motion    I/O    mode    flow    servo    quit    m
```

Select **flow** to establish the program direction.

The display will be:

```
>0: (program)
>1:

gosub    goto    IFthen    loop    return    label    m
```

Press the "down" arrow for additional options.

```
>0: (program)
>1:

end    re-run    title    quit    m
```

FLOW – > GOSUB – > LABEL

I/O Bus

Gosub Selection of **Gosub** from the Flow menu allows the User to direct the program flow to a previously defined subroutine. The **Return** command (also within the **Flow** menu) must be used in conjunction with this command.

The Unidex 400 may be factory configured to output data through either a 8-out, 12-in I/O bus or a 16-out MST bus. The selections available from the **Gosub** menu are dependent upon the I/O configuration of the system being used.

This command establishes the **G61** or **G63** commands in machine tool language.

A description of the **Gosub** command used in conjunction with each of the Output menu types follows:

I/O Bus Output

Following selection of **Gosub** the display will be:

```
                  GOSUB  CASE
   label =

 ins-Y/N      del                              back
```

label Select **label** to designate the label that represents the beginning of the subroutine to be processed. The Label must have been previously defined at the correct location by the **label** command selection of the main **FLOW** menu prior to running the program.

FLOW – > GOSUB – > PROGRAM

I/O Bus

The display will be:

GOSUB CASE		
label =		
ins-Y/N	del	back

Key in the label number (not the line number) that signifies the beginning of the subroutine.

Select **ins-Y/N** or **del** to insert or delete a character.

Select **back** to return to the initial **FLOW** selection menu.

Press the **ENTER** key to write the program block.

Example: GOSUB LABEL 200 or G61 L200
; program execution will jump to the location of Label 200.

program Select **program** to designate the file number of the program to be called as a subroutine. The program subroutine may be any program that is resident in the Unidex 400 memory.

At the completion of the subroutine, the program run will return to the next program block of the original program. A **return** is not required at the end of the program called as a subroutine.

The display will be:

GOSUB CASE		
program =		
ins-Y/N	del	back

FLOW– >GOSUB– >VARB

I/O Bus

Key in the file number of the program to be run as a subroutine.

Select **ins-Y/N** or **del** to insert or delete a character.

Select **back** to return to the initial FLOW selection menu.

Press the **ENTER** key to write the program block.

Example: GOSUB P100 or G61 P100
; program execution will run program 100 as a subroutine.

varb Select **varb** to specify the variable location containing the Label or Program number to be called as a subroutine.

The display will be:

		GOSUB CASE		
variable =			Label/Program =	
ins-Y/N	del	label	progrm	back

Key in the variable number.

Select **label** or **progrm** to designate whether the selected variable contains a Label number or a Program file number.

Select **ins-Y/N** or **del** to insert or delete a character.

Select **back** to return to the initial FLOW selection menu.

Press the **ENTER** key to write the program block.
Program Block Format: GOSUB Program/Label Variable number
Example: GOSUB PV100 or G61 PV100
; program execution will go to variable 100 for the program number to be run as a subroutine.

FLOW – > GOSUB – > INPUT

I/O Bus

input Select **input** to configure the Unidex 400 to receive the format of an input from an external device for the purpose of calling a subroutine.

The format of the input through the I/O Bus may be either Binary Coded Decimal or Hexadecimal. The input number may represent either a Label, to call a subroutine from within the current program, or a program file number, indicating that the subroutine is another program.

The display will be:

		GOSUB CASE		
input Bcd/Hexdec =			**Label/Program =**	
Bcd	**Hexdec**	**Label**	**Progrm**	**back**

Select **Bcd** or **Hexdec** to designate whether the input is in Binary Coded Decimal or Hexadecimal format.

Select **label** or **progrm** to designate whether the input contains a Label number or a Program file number.

Select **back** to return to the initial FLOW selection menu.

Press the **ENTER** key to write the program block.

Program Block Format: GOSUB Program/Label Bcd/Hexdec

Format Example: GOSUB LB or G61 LB
; a Binary Coded Decimal input will specify the label number of the subroutine to be processed.

FLOW — > GOSUB — > IF

I/O Bus

if Select **if** to establish input conditions that will be used to determine whether an input calls a subroutine.

The display will be:

		GOSUB CASE		
I-hi	I-lo	I-bit	intrp	back

Select **I-hi** or **I-lo** to designate whether a High or Low input bit calls a subroutine.

The display will be:

	GOSUB CASE		
if input bit =		is high/low	
ins-Y/N	del	next	back

Key in the input bit number (0-11).

Select **next** to specify whether the conditional input bit will designate a **label**, **program** or **variable** to identify the subroutine.

The display will be:

	GOSUB CASE	
if input bit pattern =		is high/low
label/program =		
ins-Y/N	del	back

FLOW – > GOSUB – > IF

I/O Bus

Select **label** or **progrm** to designate whether the input is to signal the call of a subroutine identified by a Label number or a Program file number.

The display will be:

```
                      GOSUB CASE
   if input bit =
   label/program =

  ins-Y/N       del                             back
```

Select **varb** if the input is to signal the call of a subroutine identified by a variable.

The display will be:

```
                      GOSUB CASE

 variable =                          Label/Program =

  ins-Y/N       del         label     progrm     back
```

Key in the Label number, the Program file number or the Variable number that identifies the subroutine to be called. If a variable contains the subroutine identifier, specify by selecting **label** or **progrm**, which of these the variable represents.

Press the **ENTER** key to write the program block.

Program Block Format:

GOSUB Program/Label/Variable number High/Low input bit number

Example: GOSUB LABEL V1111 H3 or G63 LV1111 H3
; if input bit 3 is high, the subroutine represented by the Label contained in Variable 1111 will be called.

FLOW– >GOSUB– >IF

I/O Bus

Select **I-bit** to designate a specific bit pattern that will signal the call of a subroutine.

The display will be:

		GOSUB CASE			
if input bit pattern =					
del	high	low	skip	next	back

Select **high, low** and/or **skip** to establish a 12-bit pattern. When this **GOSUB** command is encountered within the program, the Unidex 400 will "look" to see if the designated bit pattern is being input. If it is, a designated subroutine will be called. If the bit pattern is not as specified, the **GOSUB** command will be ignored.

NOTE: **Bit pattern entries are always right justified. Output bit pattern HL will be read as **********HL.**

Select **del** to delete the character at the cursor position.

Select **next** to specify whether the input bit pattern will designate a **label, program** or **variable** to identify the subroutine.

The display will be:

		GOSUB CASE	
if input bit pattern =			
label	progrm	varb	back

FLOW-- >GOSUB-- >IF

I/O Bus

Select **label** or **progrm** to designate whether the input bit pattern is to signal the call of a subroutine identified by a Label number or a Program file number.

The display will be:

```
                      GOSUB CASE
   if input bit =
    label/program =

  ins-Y/N        del                               back
```

Select **varb** if the input bit pattern is to signal the call of a subroutine identified by a variable.

The display will be:

```
                      GOSUB CASE
   if input bit =                  is high/low
   variable = .                    Label/ Program =

  ins - Y/N      del      label      program      back
```

Key in the Label number, the Program file number or the Variable number that identifies the subroutine to be called. If a variable contains the subroutine identifier, specify by selecting **label** or **progrm**, which of these the variable represents.

Press the **ENTER** key to write the program block.

Program Block Format:
GOSUB Program/Label/Variable number input bit pattern

Example: GOSUB PV2222 IHHHHHHHHHL
or G63 PV2222HHHHHHHHHHHL
; input bits 1-11 must be high and input bit 12 must be low when the GOSUB command is read in order to call the subroutine represented by the Program file number contained in Variable 2222.

FLOW– >GOSUB– >IF

I/O Bus

Select **intrp** to designate an Interrupt signal to initiate the call of a subroutine.

Upon receiving an Interrupt signal, the Unidex 400 "looks" to see if any subroutine has been specified for that Interrupt. If a subroutine has been specified in the previously processed portion of the program, the "jump" to the subroutine will occur in one of two ways; the program block currently being processed is completed before the subroutine is entered, or the current program block is aborted and the subroutine is entered immediately . These command blocks should be located at or near the beginning of the program.

At the completion of the subroutine, the Unidex 400 will go to the Program block after the block where the Interrupt occurred.

Following selection of **intrp**, the display will be:

```
                    GOSUB  CASE

  if USER INTERRUPT

    1      1abort      2      2abort      3      3abort      m
```

Press the "down" arrow for additional options.

```
                    GOSUB  CASE

  if USER INTERRUPT

    4      4abort                                back        m
```

Select Interrupt **1** through **4** to designate the interrupt signal that will initiate the subroutine. The current program block will be completed prior to processing the subroutine.

Select Interrupt **1abort** through **4abort** to designate the interrupt signal that will initiate the subroutine. The current program block will be aborted and the subroutine will immediately be processed.

FLOW – > GOSUB – > IF

I/O Bus

The display will be:

```
                    GOSUB CASE
   if USER INTERRUPT   1-4  or  1abt-4abt
   label/program =

 label      progrm      varb                        back
```

Select **label** or **progrm** to designate whether the interrupt signal is to initiate the call of a subroutine identified by a Label number or Program file number.

The display will be:

```
                    GOSUB CASE
   if USER INTERRUPT   1-4  or  1abt-4abt
   label/program =

 ins-Y/N     del                                back
```

Select **varb** if the input bit pattern is to signal the call of a subroutine identified by a variable.

The display will be:

```
                    GOSUB CASE
   if USER INTERRUPT   1-4  or  1abt-4abt
   variable =                  Label/ Program =

 ins - Y/N      del        label        program     back
```

Key in the Label number, the Program file number or the Variable number that identifies the subroutine to be called. If a variable contains the subroutine identifier, specify by selecting **label** or **progrm**, which of these the variable represents.

FLOW – > GOSUB – > IF

I/O Bus

Press the **ENTER** key to write the program block.

Program Block Format:

GOSUB Program/Label/Variable number interrupt type and number

Example: GOSUB PV1234 A1
or G63 PV1234A1
; if an Interrupt signal is received on Interrupt line #1 at any point in the program following this command, the current program block is aborted and the Program number contained in Variable #1234 will be run as a subroutine.

Example: GOSUB Label V3333 T4
or G63 LV3333T4
; if an Interrupt signal is received on Interrupt line #4 at any point in the program following this command, the current program block is completed and the Label number contained in Variable #3333 will be the location of the beginning of the subroutine.

FLOW – > GOSUB – > LABEL

MST Bus

MST Bus Output

Following selection of **Gosub** the display will be:

		GOSUB CASE		
label	program	varb	if	back

label Select **label** to designate the label that represents the beginning of the subroutine to be processed. The Label must have been previously defined at the correct location by the "label" command from the main **FLOW** menu, prior to running the program.

The display will be:

	GOSUB CASE	
label =		
ins-Y/N	del	back

Key in the label number (not the line number) that signifies the beginning of the subroutine.

Select **ins-Y/N** or **del** to insert or delete a character.

Select **back** to return to the initial **FLOW** selection menu.

Press the **ENTER** key to write the program block.

Example: GOSUB LABEL 200 or G61 L200
; program execution will jump to the location of Label 200.

FLOW– >GOSUB– >PROGRAM
MST Bus

progrm Select **progrm** to designate the file number of the program to be called as a subroutine. The program subroutine may be any program that is resident in the Unidex 400 memory.

At the completion of the subroutine, the program run will return to the next program block of the original program. A "return" is not required at the end of the program called as a subroutine.

The display will be:

	GOSUB CASE	
program =		
ins-Y/N	**del**	**back**

Key in the file number of the program to be run as a subroutine.

Select **ins-Y/N** or **del** to insert or delete a character.

Select **back** to return to the initial **FLOW** selection menu.

Press the **ENTER** key to write the program block.

Example: GOSUB program 100 or G61 P100
; program execution will run program 100 as a subroutine.

FLOW-- > GOSUB-- > VARB

MST Bus

varb Select **varb** to specify the variable location containing the Label or Program number to be called as a subroutine.

The display will be:

		GOSUB CASE		
variable =		Label/ Program =		
ins - Y/N	del	label	progrm	back

Key in the variable number.

Select **label** or **progrm** to designate whether the selected variable number contains a Label number or a Program file number.

Select **ins-Y/N** or **del** to insert or delete a character.

Select **back** to return to the initial FLOW selection menu.

Press the **ENTER** key to write the program block.

Example: GOSUB PV100 or G61 PV100
; program execution will run the program specified in variable 100 as a subroutine.

FLOW– >GOSUB– >IF

MST Bus

if Select **if** to establish input conditions that will be used to determine if an input will call a subroutine.

The display will be:

```
                    GOSUB  CASE



      ack h        ack l          intrp       back
```

Select **ack h** or **ack l** to designate whether a High or Low input bit calls a subroutine.

The display will be:

```
                    GOSUB  CASE

  if MST ack high/low=

   label    progrm   varb                        back
```

label Select **label** to designate the label that represents the beginning of the subroutine to be processed. The Label must have been previously defined at the correct location by the **label** command from the main **FLOW** menu, prior to running the program.

The display will be:

```
                    GOSUB  CASE

  if MST ack high/low=
  label =
  ins-Y/N   del                                  back
```

FLOW-- >GOSUB-- >IF

MST Bus

Key in the label number (not the line number) that signifies the beginning of the subroutine.

Select **ins-Y/N** or **del** to insert or delete a character.

Select **back** to return to the initial **FLOW** selection menu.

Press the ENTER key to write the program block.
Example: GOSUB LABEL 200 HA or G63 L200 HA
; program execution will jump to the location of Label 200 upon receipt of a high acknowledge signal.

progrm Select **progrm** to designate the file number of the program to be called as a subroutine. The program subroutine may be any program that is resident in the Unidex 400 memory.

At the completion of the subroutine, the program run will return to the next program block of the original program. A **return** is not required at the end of the program called as a subroutine.

The display will be:

GOSUB CASE	
if MST ack high/low=	
program =	
ins-Y/N del	**back**

Key in the file number of the program to be run as a subroutine.

Select **ins-Y/N** or **del** to insert or delete a character.

Select **back** to return to the initial **FLOW** selection menu.

Press the **ENTER** key to write the program block.

Example: GOSUB P100 HA or G63 P100 HA
; program execution will run program 100 as a subroutine upon receipt of a high acknowledge signal.

varb Select **varb** to specify the variable location containing the Label or Program number to be called as a subroutine.

The display will be:

		GOSUB CASE		
if MST ack high/low=				
variable =		**Label/Program =**		
ins-Y/N	**del**	**label**	**progrm**	**back**

Key in the variable number.

Select **label** or **progrm** to designate whether the selected variable number contains a Label number or a Program file number.

Select **ins-Y/N** or **del** to insert or delete a character.

Select **back** to return to the initial **FLOW** selection menu.

Press the **ENTER** key to write the program block.

Example: GOSUB PV100 LA or G63 PV100 LA
; program execution will run the program specified in variable 100 as a subroutine.

FLOW – > GOSUB – > IF

MST Bus

Select **intrp** to designate an Interrupt signal to initiate the call of a subroutine.

Upon receiving an Interrupt signal, the Unidex 400 "looks" to see if any subroutine has been specified for that Interrupt. If a subroutine has been specified in the previously processed portion of the program, the "jump" to the subroutine will occur in one of two ways; the program block currently being processed is completed before the subroutine is entered, or the current program block is aborted and the subroutine is entered immediately . These command blocks should be located at or near the beginning of the program.

At the completion of the subroutine the Unidex 400 will go to the Program block following the block where the Interrupt occurred.

Following selection of **intrp**, the display will be:

			GOSUB CASE			
if USER INTERRUPT						
1	**1abort**	**2**	**2abort**	**3**	**3abort**	**m**

Press the "down" arrow for additional options.

			GOSUB CASE			
if USER INTERRUPT						
4	**4abort**				**back**	**m**

Select Interrupt **1** through **4** to designate the interrupt signal that will initiate the subroutine. The current program block will be completed prior to processing the subroutine.

Select Interrupt **1abort** through **4abort** to designate the interrupt signal that will initiate the subroutine. The current program block will be aborted and the subroutine will immediately be processed.

FLOW– >GOSUB– >IF

MST Bus

The display will be:

```
                    GOSUB CASE
   if USER INTERRUPT



  label     progrm     varb                        back
```

Select **label** or **progrm** to designate whether the interrupt signal is to initiate the call of a subroutine identified by a Label number or Program file number.

The display will be:

```
                    GOSUB CASE
   if USER INTERRUPT   1-4  or  1abt-4abt
   label/program =

  label     progrm     varb                        back
```

Select **varb** if the input bit pattern is to signal the call of a subroutine identified by a variable.

The display will be:

```
                    GOSUB CASE
   if USER INTERRUPT   1-4  or  1abt-4abt
   variable =                   Label/ Program =

  ins - Y/N       del        label       progrm      back
```

Key in the Label number, the Program file number or the Variable number that identifies the subroutine to be called. If a variable contains the subroutine identifier, specify by selecting **label** or **progrm**, which of these the variable represents.

FLOW-- > GOSUB-- > IF

MST Bus

Press the **ENTER** key to write the program block.

Program Block Format:

GOSUB Program/Label/Variable number interrupt type and number

Example: GOSUB PV1234 A1

or G63 PV1234A1

; if an Interrupt signal is received on Interrupt line #1 at any point in the program following this command, the current program block is aborted and the Program number contained in Variable #1234 will be run as a subroutine.

Example: GOSUB Label V3333 T4

or G63 LV3333T4

; if an Interrupt signal is received on Interrupt line #4 at any point in the program following this command, the current program block is completed and the Label number contained in Variable #3333 will be the location of the beginning of the subroutine.

Goto

Select **Goto** from the Flow menu to direct program flow to a previously defined program or label. All of the options for I/O and MST Bus operation that have been previously discussed for the **Gosub** command are applicable when using the **Goto** command, however when using the **Goto** Command, a return is not recognized.

This command establishes the **G60** or **G62** commands in machine tool language.

FLOW– > IFTHEN

IFthen Select **IFthen** from the **Flow** menu to establish a conditional "Goto" command with a variable designator. When encountered during program execution, the **IFthen** command causes the Unidex 400 to compare the specified variable with the condition selected, if the condition is met, program execution will commence at the designated label or program.

This command establishes the **G62** command in machine tool language.

The display will be:

```
        IF .. CONDITION .. THEN  GOTO  CASE
condition : V


=        <        >        -        goto        varb        m
```

Press the "down" arrow for additional options.

```
        IF .. CONDITION .. THEN  GOTO  CASE
condition : V


ins-Y/N     del              H:      goto      back      m
```

From the initial **IFthen** menu key in the variable (condition) number.

Select the operation (=, <, >, < >, or -) for the "IFthen" statement.

Key in a number or select **varb** (and key in the variable number) to be used as the comparator.

If the comparison is to be done in Hex, select **H:**

Select **goto** to designate the result of the **IFthen** statement.

The display will be:

```
        IF .. CONDITION .. THEN  GOTO  CASE
   condition : Vxxxxxxx


   label        progrm        varb                          back
```

Select **label, progrm**, or **varb** to designate the destination of program flow if the conditions of the "IFthen" statement are met.

The display will be:

```
        IF .. CONDITION .. THEN  GOTO  CASE
   condition : Vxxxxxxx
   label /program  =

   ins - Y/N      del                                       back
```

Select **varb** if the input bit pattern is to signal the call of a subroutine identified by a variable.

The display will be:

```
                 IF_CONDITION_ THEN  GOTO  CASE
     condition : Vxxxxxxx
     variable  =                      Label/ Program  =

   ins - Y/N      del        label        progrm        back
```

Key in the Label number, the Program file number or the Variable number that identifies the subroutine to be called. If a variable contains the subroutine identifier, specify by selecting **label** or **progrm**, which of these the variable represents.

FLOW – > IFTHEN

Select **back** to abort this selection and return to the previous display.

Press the **ENTER** key to write the program block.

Program Block Format:
GOTO Prog./Label/Variable number Variable (Condition) number operation comparator

Example: GOTO LABEL 400 C1 = 2 or G62 L400 C1 = 2
; if Condition 1 equals 2 then program execution will goto Label 400

Example: GOTO P678 C123 < > V345 or G62 P678 C123 < > V345
; if Condition 123 does not equal the value contained in variable 345 then program execution will goto program 678

FLOW – > LOOP

loop Select **loop** from the **Flow** menu to establish the beginning of a group of program blocks to be repeated a designated number of times or, if using an I/O Bus, until a specified input bit pattern is detected.

This command establishes the **G28** command in machine tool language.

The display will be:

```
repeat loop times =
or until input =

    ins-Y/N      del      until      end      varb      quit
```

Key in the desired number of times that the group of program blocks are to be repeated or select **varb** and key in the variable number containing the number of times the loop is to be repeated.

NOTE: **Program blocks designated as a part of a loop will be run at least one (1) time.**

Select **ins-Y/N** or **del** to insert or delete a character.

Select **until** to establish an input bit pattern that will terminate processing of the loop.

NOTE: **The input bit pattern is monitored following the processing of each loop. If the bit pattern specified by the "until" selection is detected, processing of the loop will stop.**

The display will be:

```
repeat loop times =
or until input =

    del      high      low      skip      back      quit
```

FLOW– >LOOP

Select **del** to delete a character at the cursor.

Select **high** to establish a bit setting of "1". (Bit pattern entries are always right justified.)

Select **low** to establish a bit setting of "0". (Bit pattern entries are always right justified.)

Select **skip** to allow a bit setting to remain unchanged or "don't care". (Bit pattern entries are always right justified.)

NOTE: Bit pattern HL will be read as **********HL.

Select **back** to return to the previous **loop** selection menu.

Select **end** to establish the endpoint of the group of program blocks to comprise the loop. This selection establishes the **N-2** command in machine tool language.

Select **quit** to abort the command and return to the initial **Flow** selection menu.

Press the **ENTER** key to write the program block.

Program Block Format: repeat number of times
Example: repeat 10 or G28 10
; the specified group of program blocks will be repeated 10 times

Program Block Format: repeat number of times Bit Pattern
Example: repeat 11 HL or G28 11 HL
; the specified group of program blocks will be repeated 11 times or until an input bit pattern of **********HL is detected

Example: end loop or N-2
; signifies the last block of program to be included in the loop.

FLOW – > RETURN

return Select **return** from the Flow menu to indicate the last program block of a subroutine. This command is used in conjunction with the **GOSUB** command.

This selection establishes the **N-1** command in machine tool language.

Press the **ENTER** key to write the program block.

Program Block Format: Return

Example: Return or N-1
; returns program flow to main program

FLOW– >LABEL

label Select **label** from the **Flow** menu to define an entry point within the program. The Label may be up to four characters in length and must occupy its own program block.

This selection establishes the **N** command in machine tool language.

Press the **ENTER** key to write the program block.

Program Block Format: Label number

Example: LABEL 123 or N123
; Label 123 is written to a program block

FLOW– >END

FLOW– >RERUN

Press the "down" arrow for additional **Flow** options.

```
      >0: (program)
      >1:

end         re-run          title               quit    m
```

end Select **end** to designate the program's end. Program execution will stop at this point.

This selection establishes the **M2** command in machine tool language.

Example: end prog or M2
; program execution will stop

re-run Select **re-run** to initiate a "jump" to the programs beginning and start a re-run of the program.

This selection establishes the **M47** command in machine tool language.

Example: repeat prog or M47
; program execution will stop and begin again at the program's beginning

FLOW– > TITLE

title Select **title** to add a name to the current program. A maximum of 48 characters may be used in the title. The **title** command (**G30**) should be the first line in a program. When in the **File- Directory** Mode, the title designated by this command will appear next to the program number.

This command establishes the **G30** command in machine tool language.

The display will be:

(TITLE < 48 char)				
ins-Y/N	del		quit	m

The following displays provide character selections to create a title. Press the "down" arrow to initiate each of these displays.

(TITLE < 48 char)						
A	B	C	D	E	F	m

(TITLE < 48 char)						
G	H	I	J	K	L	m

(TITLE < 48 char)						
M	N	O	P	Q	R	m

FLOW – > TITLE

(TITLE < 48 char)

S T U V W X m

(TITLE < 48 char)

Y Z [\ ] ^ m

(TITLE < 48 char)

_ ` { | } ~ m

(TITLE < 48 char)

space ! " # $ % m

(TITLE < 48 char)

& ' ( ) * + m

FLOW- > TITLE

```
(TITLE < 48 char)

,     _     .     /     :     ;     m
```

```
(TITLE < 48 char)

<     =     >     ?     @     m
```

Select **ins-Y/N** or **del** to insert or delete a character at the cursor location.

At the completion of title character selection, press the **ENTER** key to write the program block.

Program Block Format: Title

Example: TITLE OF PROGRAM or G30 TITLE OF PROGRAM
; TITLE OF PROGRAM will be written as the Program's title.

3-5-5: SERVO

Selection of **servo** permits modification of several Servo related parameters from within the program, these settings are valid for the current program only, system parameters are not effected. The following sections describe these functions.

Following the selection of **insrt** or **alter** from Edit Screen #1, the display will be:

```
>0: (program)
>1:

motion   I/O   mode   flow   servo   quit   m
```

Select **servo** to establish program specific Servo parameters.

The display will be:

```
ramp   ac/de   pid   traj   coef   quit
```

SERVO – > RAMP

ramp Select **ramp** to set the time it will take all axes to reach the desired velocity. Ramp time is used for coordinated motion only, non-coordinated motion (**G0**, **Home**, etc.) uses the ac/de rate. This selection overrides but does not change the setting of General Parameter 5. The time established by this selection remains in effect until updated by a subsequent Ramp entry or system reset.

This command establishes the **G6** command in machine tool language.

The display will be:

```
axes ramp time  ms =


ins-Y/N        del                                  varb      quit
```

Key in the desired Ramp time or select **varb** and key in the variable number that contains the desired Ramp time.

Select **ins-Y/N** or **del** to insert or delete a character at the cursor location.

Select **quit** to abort the command and return to the initial **servo** selection menu.

Press the **ENTER** key to write the program block.

Program Block Format: ramp time in mSec

Example: ramp ms 11111 or G6 11111
; the Ramping time of all axes will be 11111 milliseconds

SERVO – > AC/DE

ac/de The **ac/de** selection provides the ability to set the acceleration/deceleration rate for each axis, overriding the setting of Axis Parameter 9. All acceleration/deceleration rates are established in steps/sec/sec, not decimals. The rate established by this selection remains in effect until updated by a subsequent **ac/de** entry or a system reset.

This command establishes the **G29** command in machine tool language.

Select **ac/de**, the display will be:

```
X      Y      Z      U                    quit
```

Select the desired axis, the display will be:

```
_ axis ac/de steps/sec/sec =

ins -Y/N     del                 varb      quit
```

Key in the desired acceleration/deceleration rate or select **varb** and key in the variable number that contains the desired acceleration/deceleration rate.

Select **ins-Y/N** or **del** to insert or delete a character at the cursor location.

Select **quit** to abort the command and return to the initial **servo** selection menu.

Press the **ENTER** key to write the program block.

Program Block Format: ac/de axis acceleration/deceleration rate steps/sec/sec

Example: ac/de X 111111111 or G29 111111111
; the X axis ac/de rate is 111111111 steps/sec/sec

SERVO – > PID

PID The **pid** selection is used to set the Proportional Gain, the Integral Gain, and the Derivative Gain for each axis, overriding the settings of Axis Parameters 9, 10, 11 and 12. These Gain settings are used by the Unidex 400 to generate the Servo Loop. Refer to Chapter 7 : The Setup Mode, Section 7-8 for details concerning Servo Loop Gain settings.

The settings established by this selection remain in effect until updated by a subsequent **pid** entry or a system reset.

This command establishes the **G27** command in machine tool language.

Select **pid** from the **servo** menu. The display will be:

```
 X      Y      Z      U                    quit
```

Select the desired axis. The display will be:

```
X Kp =                      Ki =
  Kd =                      Kf1 =
  Kf2 =
   Kp     Ki     Kd     Kf1    Kf2    Varb    m
```

NOTE: The following X Axis display is representative of axes Y, Z, and U.

Press the "down" arrow for additional **pid** options.

```
X Kp =                      Ki =
  Kd =                      Kf1 =
  Kf2 =
  ins-N   del                               m
```

Select the PID option to be altered.

Key in the desired Gain values.

Select **ins-Y/N** or **del** to insert or delete a character at the cursor location.

Select **quit** to abort the command and return to the initial **servo** selection menu.

Press the **ENTER** key to write the program block.

Program Block Format:
set PID axis Kp value Ki value Kd value F1 value F2 value

Example: set PID X P1111 I2222 D3333 F14444 F2555555
or G27 X P1111 I2222 D3333 F14444 F2555555
; Gain settings for the X axis are established.

SERVO -- > TRAJ

traj The **traj** selection provides the ability to set the axis ramping trajectory for either Linear or Parabolic, overriding the settings of Axis Parameter 29 (software versions prior to version 1.13) or General Parameter 38 (software versions 1.13 and above).

The trajectory type established by this selection, remains in effect until updated by a subsequent trajectory entry or a system reset which will reinstate the parameter values.

For software versions 1.13 and above, all contour moves of individual axes are sychronized with a "virtual" Master axis, therefore the Acceleration/Deceleration Ramping trajectory for each axis follows the Ramping Trajectory established for the Master axis.

This command establishes the **G32 M0** or **G32 M1** command in machine tool language.

Select **traj**, the display will be:

X	**Y**	**Z**	**U**	**M**	**quit**

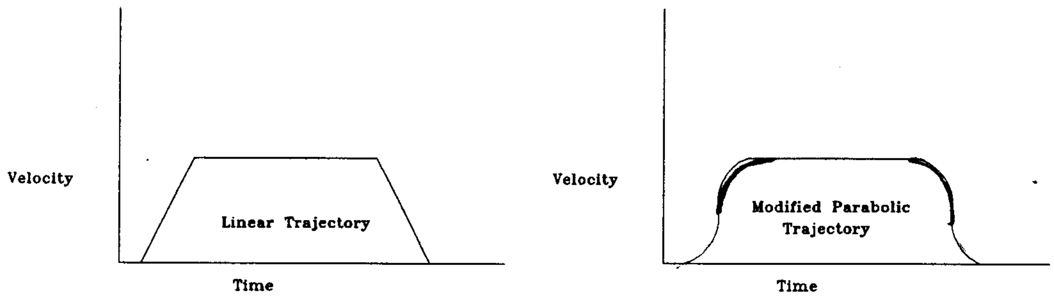
NOTE: **The above display is used in Unidex 400 software versions 1.13 and above. When using earlier software, only axes designations X through U are present. When using software version 1.13 and above, only the M (Master)selection is valid.**

Select the desired axis, the display will be:

X /M axis traj type =		
linear	**parab**	**quit**

Select **linear** to establish a linear ramping trajectory (**G32 M0**).

Select **parab** to establish a parabolic ramping trajectory (**G32 M1**).A Parabolic Trajectory is derived by applying a "Parabolic Trajectory Coefficient" to the maximum slope. The Parabolic Trajectory Coefficient is set by Axis Parameter 30 (software versions prior to version 1.13) or General Parameter 38 (software versions 1.13 and above). The "Parabolic Coefficient" may be changed while in the Edit mode by selecting **coef** from the initial **Servo** menu. A parabolic ramping trajectory will round the corners of the velocity profile as illustrated below:



NOTE: Index type moves (G0) always are performed on a per axis basis and do not implement the "Master" axis concept.

Select **quit** to abort the command and return to the initial **Servo** selection menu.

Program Block Format:
trajectory type axis/M case 0 = linear, 1 = parabolic
Example: trajectory type M0 or G32 M0
; The "Master" axis will have a linear ramping trajectory

SERVO – > COEF

coef The **coef** selection is used to provide the factor that is used in conjunction with a parabolic trajectory to convert the linear velocity/time slope into a Parabolic Trajectory. This setting will override the setting of Axes Parameter 30 (software versions prior to version 1.13) or General Parameter 38 (software versions 1.13 and above). Refer to the **SERVO – > TRAJ** Command for additional information

It is suggested that coefficient selections be made in steps of 10,000. The following formula is applied to the coefficient value selected:

$$\text{maximum slope during ramp} = \frac{\text{parabolic coefficient}}{65535} + 1$$

The parabolic coefficient established by this selection, remains in effect until updated by a subsequent coefficient entry or a system reset.

This command establishes the **G33 Mn** command in machine tool language.

NOTE: **Index type moves (G0) always are performed on a per axis basis and do not implement the "Master" axis concept.**

Select **coef**, the display will be:

```
    X      Y      Z      U              M       quit
```

NOTE: **The above display is used in Unidex 400 software versions 1.13 and above. When using earlier software, only axes designations X through U are present.**

Select the desired axis, the display will be:

```
 X/M axis traj coefficient =

   ins -Y/N    del                        Varb      quit
```

SERVO – > COEF

Key in the desired parabolic trajectory coefficient or select **Varb** and key in the variable number that contains the desired parabolic trajectory coefficient rate.

Select **ins-Y/N** or **del** to insert or delete a character at the cursor location.

Select **quit** to abort the command and return to the initial **servo** selection menu.

Press the **ENTER** key to write the program block.

Program Block Format: trajectory coeff = M value

Example: trajectory coeff = M22222 or G33 M 22222
; The "Master" axis will have a parabolic trajectory coefficient of 22222

3-5-6: OPTIONS

Selection of **OPTIONS** provides the ability to initiate use of an optional Joystick, Handwheel or the PSO function from within a program.

Following the selection of **insrt** or **alter** from Edit Screen #1, the display will be:

```
>0: (program)
>1:

motion    I/O    mode    flow    servo    quit    m
```

Press the "down" arrow for additional options.

```
>0: (program)
>1:

option    dwell    MSG    MISC    INTRP    math    m
```

Select **option** to initiate use of a Joystick, Handwheel or the PSO function.

The display will be:

```
jstik    handwl    PSO                quit
```

OPTION – > JSTIK

jstik The **jstik** selection is used to initiate the Unidex 400 for use of a Joystick for jogging and/or digitizing axis position. Axis Parameters 18, 19 and 20 and General Parameter 8, affect Joystick operation. A detailed description of the Joystick option is contained in the Options section of this manual.

This selection establishes the **G19** command in machine tool language.

Example: call joystick or G19
; the Unidex 400 is initiated for Joystick use.

OPTION -- > HANDWHEEL

handwl The **handwl** selection is used to initiate or disable use of an optional Handwheel to manually control motion of an axis. The handwheel scale factor is used to convert handwheel increments to machine steps. The data input format is established in General Parameter18 and/or 19. General Parameter 22 also affects Handwheel operation. A detailed description of the Handwheel operation is contained in the Options section of this manual.

The handwheel characteristics established by this selection remain in effect until changed by a handwheel disable entry or a system reset.

This selection establishes the **G95** command in machine tool language.

Select **handwl** to initiate or disable Handwheel use.

The display will be:

```
handwheel scale (1-255) =
axis =              input =

  X       Y       Z       U       none       input       m
```

Press the "down" arrow for additional handwheel options.

```
                         DWELL

dwell time in ms =

  ins-Y/N    del                            quit
```

Key in the handwheel scale (the relationship between handwheel increments and the number of machine steps of axis movement) or select **Varb** and key in the variable number that contains the desired handwheel scale.

Select the axis that is to be moved by the handwheel.

Select **none** to disable handwheel use for all axes.

OPTION – > HANDWHEEL

Select **input** to delineate which input (1 or 2) is assigned to this handwheel.

Select **ins-Y/N** or **del** to insert or delete a character at the cursor location.

Select **quit** to abort the command and return to the initial **motion** selection menu.

Press the **ENTER** key to write the program block.

Program Block Format: handwheel axis input number scale factor
handwheel disable input number

Example: handwheel X1 200 or G95 X1 200
; handwheel control is enabled for the X axis on Input 1 using a scale factor of 200 machine steps equal to one handwheel increment..

handwheel D1 or G95 D1
; input 1 handwheel control is disabled for all axes

OPTION – > PSO

PSO The **PSO** option is a programmable, DSP based Position Synchronized Output Control card that precisely coordinates the functioning of a peripheral device with the motion initiated by the Unidex 400 Controller. A detailed description of the **PSO** option is contained in the Options section of this manual.

This selection establishes the **G80** command in machine tool language.

3-5-7: DWELL

The **dwell** selection provides a time delay of a programmed or established duration. The duration may be entered in either decimal or integer form. If the Dwell duration value is followed by a ".", (decimal form) the Dwell will be in seconds. If the Dwell value is not followed by a ".", (integer form) the Dwell will be in tenths of a second. The Dwell command must occupy it's own block within a program.

The Unidex 400 may be factory configured to output data through either a 8-out, 12-in I/O bus or a 16-out MST bus. The selections available from the **dwell** menu are dependent upon the I/O configuration of the system being used.

This command establishes the **G4** command in machine tool language.

A description of the **dwell** command used in conjunction with each of the Output menu types follows:

I/O Bus Output

Following selection of **dwell,** the display will be:

DWELL			
time	**input**	**Cstart**	**quit**

DWELL

I/O Bus

The display will be:

DWELL		
dwell time in ms =		
ins-Y/N	del	quit

Key in the desired duration (mSec) of the delay in the program.

Select **ins-Y/N** or **del** to insert or delete a character at the cursor location.

Select **quit** to abort the command and return to the initial **motion** selection menu.

Press the **ENTER** key to write the program block.

Program Block Format: dwell time (mSec)

Example: dwell 234. ms or G4 F234.
; a delay of 234 mSec. will occur within the program

Select **input** from the initial **dwell** menu to establish an input bit pattern that will signify the completion of the program delay.

DWELL

I/O Bus

Select **time** to establish the duration of the program delay.

The display will be:

		DWELL			
dwell until input =					
ins-Y/N	**del**	**high**	**low**	**skip**	**quit**

Select **high, low** and/or **skip** to establish a 12-bit pattern. When this **dwell** command is encountered within the program, the Unidex 400 will stop program execution until the designated bit pattern is input. When it is, program execution will continue.

NOTE: **Bit pattern entries are always right justified. Input bit pattern HL will be read as **********HL.**

Select **ins-Y/N** or **del** to insert or delete a character at the cursor location.

Select **quit** to abort the command and return to the initial **motion** selection menu.

Press the **ENTER** key to write the program block.

Program Block Format: dwell until input = bit pattern

Example: dwell until input = *******HLLLL or G4 I*******HLLLL
; Program execution will be delayed until the specified bit pattern is input

DWELL

I/O Bus

Select **Cstart** from the initial **dwell** menu to establish a program delay that will continue until the front panel **CYCLE START** key is depressed.

Program Block Format: dwell until cycle start

Example: dwell until cycle start or G4 C
; Program execution will be delayed until the front panel **CYCLE START** key is pressed.

MST Bus Output

Following selection of **dwell** the display will be:

```
                              DWELL



   time      MSTack              Cstart                    quit
```

Select **time** to establish the duration of the program delay.

The display will be:

```
                              DWELL

  dwell time in ms =

   ins-Y/N    del                                          quit
```

Key in the desired duration (mSec) of the delay in the program.

Select **ins-Y/N** or **del** to insert or delete a character at the cursor location.

Select **quit** to abort the command and return to the initial **motion** selection menu.

Press the **ENTER** key to write the program block.

Program Block Format: dwell time (mSec)

Example: dwell 234. ms or G4 F234.
; a delay of 234 mSec. will occur within the program

DWELL

MST Bus

Select **MSTack** from the initial **dwell** menu to establish a program delay that will continue until an MST Acknowledge signal is received.

Program Block Format: dwell M

Example: dwell M or G4 M
; Program execution will be delayed until an MST Acknowledge signal is received.

Select **Cstart** from the initial **dwell** menu to establish a program delay that will continue until the front panel **CYCLE START** key is depressed.

Program Block Format: dwell until cycle start

Example: dwell until cycle start or G4 C
; Program execution will be delayed until the front panel **CYCLE START** key is pressed.

3-5-8: MSG

The **MSG** selection permits a message to be entered into a program, and be displayed on the screen at the time of program execution. The message may also be sent through the RS232 or IEEE488 ports. All numeric data can be read to system variables via the Serial port. The transmitted data must be in ASCII format and separated by a comma only. Data should be terminated by a <CR>. Data will be written sequentially from the specified start variable to the specified end variable. The transfer may be terminated prematurely by sending a <CR>. The Unidex 400 will ignore any extra data sent.

The displayed message may consist of text, User variables, and system variables and may also receive input from the User. Input in the form of characters must be enclosed in " ".

The following formats provide conversions for variable data (the default format is floating point):

#H: Hexadecimal Format
Example: #H:V5 Variable 5 is displayed in Hex format.

#C: Character Format
Example: #C:V5 Variable 5 is displayed in Character format.

#:: Variable String
Example: #::<V5,V10> Character strings for variables 5 thru 10 will be displayed.
Input that is to be displayed as a variable string must be in the following format:
"name",0 (string display will stop upon encountering a 0)

#V Value of the Variable
Example: #V5 The decimal value of variable 5 will be displayed.

This command establishes the **G21** command in machine tool language.

MSG

Following selection of **MSG**, the display will be:

RS232	**dispy**	**RS +ds**	**IE488**	**IE +ds**	**quit**	**m**

Press the "down" arrow for additional message options.

F9999	**m**

The selections of these two displays provide choices as to the destination of the message. Following is a description of each of the selections.

RS232 The **RS232** selection configures the Unidex 400 to send an ASCII message of less than 48 characters to the serial port. If input is desired, the Unidex 400 will wait for less than 40 ASCII characters. It stores numeric data to specified User variables. A comma must delimit the data values.

dspy The selection of **dsply** sends an ASCII message to the Unidex 400's front panel display. Input is accomplished through the front panel keyboard.

RS + ds The **RS + ds** selection configures the Unidex 400 to send a message to both the serial port and the Unidex 400 display. Input may be received from either the serial port or front panel keyboard.

IE488 The **IEEE488** selection configures the Unidex 400 to send an ASCII message of less than 48 characters to the IEEE488 port. If input is desired, the Unidex 400 will wait for less than 40 ASCII characters. It stores numeric data to specified User variables. A comma must delimit the data values.

IE + ds The **IE + ds** selection configures the Unidex 400 to send a message to both the IEEE488 port and the Unidex 400 display. Input may be received from either the IEEE port or front panel keyboard.

FILE 9999 An internal file, **File 9999** is reserved for data. Selection of **FILE 9999** from this display writes the data in the message field to this file in ASCII format. Prior to editing , this file must be copied to a standard file.

NOTE: **The data format for the message can be a variable or ASCII data. A variable may displayed in floating point format (default), hex format or character format.**

Select **quit** to abort the command and return to the initial **motion** selection menu.

The following displays provide characters that are not available from the Unidex 400's front panel.

NOTE: **The display selections provided for the RS232 selection are the same for the display, RS + ds, IE488, and IE + ds selections. The display for File 9999 contains slight differences in nomenclature but no variation in data entry .**

Press the "down" arrow, the first display will be:

```
    RS232
 (message < 48 char)_
 input < 40 char, from V        to V

 ins-Y/N    del     crlf    space    input    quit     m
```

Press the "down" arrow, the display will be:

```
    RS232
 (message < 48 char)_
 input < 40 char, from V        to V

   ,         #       V        H        C        :       m
```

MSG

Press the "down" arrow, the display will be:

```
    RS232
(message < 48 char)_
input < 40 char, from V        to V

 A        B        C        D        E        F        m
```

Press the "down" arrow, the display will be:

```
    RS232
(message < 48 char)_
input < 40 char, from V        to V

 G        H        I        J        K        L        m
```

Press the "down" arrow, the display will be:

```
    RS232
(message < 48 char)_
input < 40 char, from V        to V

 M        N        O        P        Q        R        m
```

Press the "down" arrow, the display will be:

```
    RS232
(message < 48 char)_
input < 40 char, from V        to V

 S        T        U        V        W        X        m
```

Press the "down" arrow, the display will be:

```
RS232
(message < 48 char)_
input < 40 char, from V          to V

Y        Z        [        \        ]        ^        m
```

Press the "down" arrow, the display will be:

```
RS232
(message < 48 char)_
input < 40 char, from V          to V

_        '        {        |        }        ~        m
```

Press the "down" arrow, the display will be:

```
RS232
(message < 48 char)_
input < 40 char, from V          to V

space    !        "        #        $        %        m
```

Press the "down" arrow, the display will be:

```
RS232
(message < 48 char)_
input < 40 char, from V          to V

&        '        (        )        *        +        m
```

MSG

Press the "down" arrow, the display will be:

```
    RS232
(message < 48 char)_
input < 40 char, from V        to V

  ,       -       .       /       :       ;       m
```

Press the "down" arrow, the display will be:

```
    RS232
(message < 48 char)_
input < 40 char, from V        to V

  <      =      >      ?      @              m
```

Select **ins-Y/N** or **del** to insert or delete a character at the cursor location.

Select **crlf** to insert a "carriage return/line feed" into the message.

Select **space** to insert a space into the message line.

Select **input** to move the cursor to the **input** section of the display. Key in the first Variable number to which input is to be written, at the **from V** location . Select **to** to move the cursor to the **to V** location and key in the last variable number to which input is to be written.

NOTE: Input may be activated for variables in conjunction with or without a message entry.

To provide the variable input function the Unidex 400 must be in the Machine Mode.

Example: MESG < 1 3 > or G21 < 1 3 >
;when the program block is run in the Machine Mode, the requested input data must be entered as input 1,input2,input3.

Select **quit** to abort command entry and return to the initial selection menu.

Press the **ENTER** key to write the program block.

The following is a sample program illustrating many of the MSG options:

```
0: DISPLAY EXAMPLE                          G30 DISPLAY EXAMPLE
1: LABEL 100                                N100
2: MESG <0 20> Enter Your Name              G21 <0 20> Enter Your Name
3: MESG <21 21> Enter Part Number           G21 <21 21> Enter Part Number
4: MESG <22 22> Enter Move Length           G21 <22 22> Enter Move Length
5: MESG <23 23> Enter Feedrate              G21 <23 23> Enter Feedrate
6: MESG <24 24> Verify #::0,20 P.N.:#V21 L:#V22 F:#V23 0/1?
        G21 <24 24> Verify #::0,20 P.N.:#V21 L:#V22 F:#V23 0/1?
7: GOTO LABEL 100 C24 = 0                   G62 L100 C24 = 0
```

3-5-9: MISC

The **MISC** selection permits modification of several basic system functions.

Following the selection of **MISC**, the display will be:

```
> 0: (program)
> 1:

track     beep     focus     + limit     -limit     quit     m
```

Press the "down" arrow, the display will be:

```
> 0: (program)
> 1:

UMFO     MIR                                                  m
```

track The **track** selection provides the ability to enable or disable the Machine Mode Tracking Display. Disabling the Tracking Display results in reduced processing time that may be required when processing certain motion commands. The default is for the Tracking Display to be enabled.

Following the selection of **track**, the display will be:

tracking display enable/disable		
enable	**disabl**	**quit**

enable Select **enable** to display tracking data while in the Machine Mode.

disabl Select **disabl** to eliminate the tracking information from the Machine Mode Display.

Select **quit** to abort command entry and return to the initial selection menu.

This command establishes the **G15** and **G16** command in machine tool language.

Program Block Format: track disp on
track disp off

Example: track disp on or G16
track disp off or G15

MISC – > BEEP

beep The **beep** selection provides the ability to enable or disable the Unidex 400's beeper. The default is for the beeper to be disabled.

NOTE: **Once enabled, the beeper will remain ON until it is disabled.**

Following the selection of **beep**, the display will be:

beeper enable/disable		
enable	**disabl**	**quit**

enable Select **enable** to enable the Unidex 400's beeper.

disabl Select **disabl** to disable the Unidex 400's beeper.

Select **quit** to abort command entry and return to the initial selection menu.

This command establishes the **G93** and **G94** command in machine tool language.

Program Block Format: beep on
beep off

Example: beep on or G93
beep off or G94

focus The **focus** selection provides auto-focus control, i.e. each axis has the ability to lock into either of the two Handwheel inputs. The linked axes will move to null the input pulses. Multiple axes may be locked into a single input.

The auto-focus command is designed to be used in a closed loop system for camera focus. It may also be used to command an axis to move at a velocity proportional to Handwheel displacement.

Data input format is established in General Parameter18 and/or 19.

The auto-focus scale factor is used to convert the input clock to machine steps. Scale entries may be used to override, but not change, Axis Parameters 31 and 32.

The input direction may be changed by establishing the polarity as **Reverse.**

The auto-focus mode established by this command remains in effect until a subsequent auto-focus command is entered, or the system is reset.

The auto-focus command is disabled by selecting "D" from the **polarity or disable** selections.

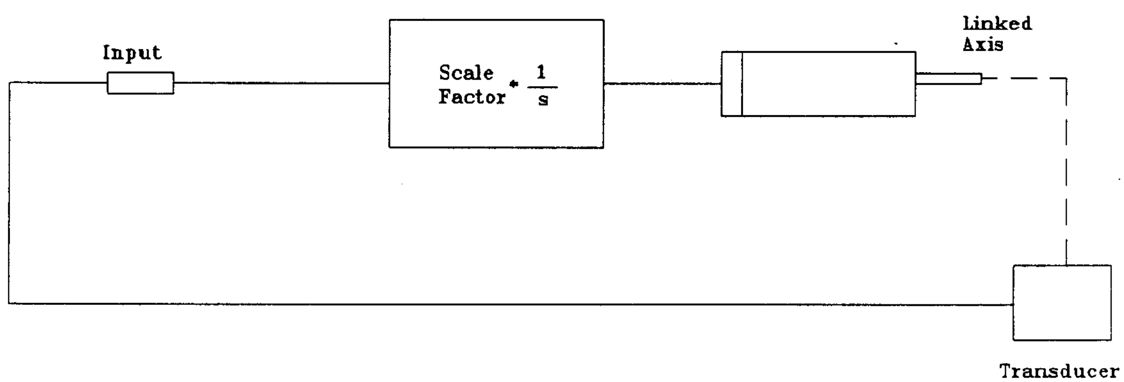


Figure 3-3: Illustration of the Auto-Focus Function

MISC – > FOCUS

Following the selection of **focus**, the display will be:

```
auto-focus axis =

X        Y        Z        U                 quit
```

Select the axis for which auto-focus is to be established (X has been selected in this example). The display will be:

```
auto-focus axis = X
scale < 127.99999 0-default =
input =        polarity or disable =

ins-Y/N    del    input    polar    Varb    quit
```

Key in the desired scale factor. The scale factor may be any whole number or fraction less than 127.9999. Select 0 to retain the default setting.

The scale factor is applied as follows:

machine steps = input * (scale factor/ 65536)

If the Scale Factor is set at "1" and "1" pulse is input, the axis will move at "1" machine step/mSec.

Select **ins-Y/N** or **del** to insert or delete a character at the cursor location.

Select **input** to establish the input that is to be used with this command. Pressing the F3 key will toggle between Input 1 or Input 2.

Select **polar** to establish the direction of axis motion or to disable the auto-focus command. Pressing the F4 key will toggle between the following:

S/1 - Axis motion will be the same as specified Input.
R/2 - Axis motion will be reverse of the specified Input.
D/3 - Auto-Focus will be disabled.

varb Select **varb** if the Scale Factor is contained in a variable. Key in the variable number.

Select **quit** to abort command entry and return to the initial selection menu.

Press the **ENTER** key to write the program block.

This command establishes the **G96** command in machine tool language.

Program Block Format:

autofocus axis input number (1/2) polarity condition (1/2/3) scale factor

Example: autofocus X1 2 5 or G96 X1 2 5

; the auto-focus command while be in effect for the X axis through input 1 with reverse polarity at rate determined by a scale factor of 5.

MISC-- > +LIMIT

+limit The **+limit** selection is used to establish a CW software limit (in machine steps) as referenced from the Hardware Home position. The value of this selection will override, but not change the value in Axis Parameter 39. This value will remain in effect until a subsequent **+limit** selection is entered, the limit command is disabled or the system is reset.

This command establishes the **G43**, **G44** and **G45** command in machine tool language.

Following the selection of **+limit**, the display will be:

```
set CW limit axis =

X        Y        Z        U                  quit
```

Select the axis for which a CW limit is to be established (X has been selected in this example). The display will be:

```
set CW limit axis = X
limit =
enable or disable =

ins-Y/N     del     enable     disabl     Varb     quit
```

Key in the CW limit (in machine steps) as referenced from the Hardware Home position or select **Varb** and key in the variable number containing the limit.

NOTE: **Entering the CW limit value does not enable the command. The command must be enabled as a separate program block.**

MISC – > +LIMIT

Select **ins-Y/N** or **del** to insert or delete a character at the cursor location

enable Select **enable** to enable the CW Limit.

disabl Select **disabl** to disable the CW Limit.

Select **quit** to abort command entry and return to the initial selection menu.

Press the **ENTER** key to write the program block.

Program Block Format:

set + limit for axis machine steps
enable +limit for axis
disable +limit for axis

Example: set + limit for axes X100. or G45 X100.
; the CW Limit is established at 100 machine steps

enable + limit for axes X or G43 X
; the established CW Limit is enabled

disable + limit for axes X or G44 X
; the CW Limit is disabled

MISC – > -LIMIT

-limit The **-limit** selection is used to establish a CCW software limit (in machine steps) as referenced from the Hardware Home position. The value of this selection will override, but not change the value of Axis Parameter 38. This value will remain in effect until a subsequent **-limit** selection is entered, the limit command is disabled or the system is reset.

This command establishes the **G53**, **G54** and **G55** command in machine tool language.

Following the selection of **-limit**, the display will be:

```
set CCW limit axis =


X        Y        Z        U                quit
```

Select the axis for which a CW limit is to be established (X has been selected in this example). The display will be:

```
set CCW limit axis = X
limit =
enable or disable =

ins-Y/N     del     enable     disabl     Varb     quit
```

Key in the CCW limit (in machine steps) as referenced from the Hardware Home position or select **Varb** and key in the variable number containing the limit.

NOTE: **Entering the CCW limit value does not enable the command. The command must be enabled as a separate program block.**

Select **ins-Y/N** or **del** to insert or delete a character at the cursor location.

enable Select **enable** to enable the CCW Limit.

disabl Select **disabl** to disable the CCW Limit.

Select **quit** to abort command entry and return to the initial selection menu.

Press the **ENTER** key to write the program block.

Program Block Format:

set - limit for axis machine steps
enable -limit for axis
disable -limit for axis

Example: set - limit for axes X100. or G55 X100.
; the CCW Limit is established at 100 machine steps

enable - limit for axes X or G53 X
; the established CCW Limit is enabled

disable - limit for axes X or G54 X
; the CCW Limit is disabled

MISC – > UMFO

UMFO The **UMFO** selection is used to enable or disable the Manual Feedrate Override. MFO scales the current feedrate linearly. When disabled, the MFO may be fixed to (0-199%). When the **UMFO** selection is enabled, any external MFO adjustments are disabled.

This selection establishes the **G5** command in machine tool language.

Following the selection of **UMFO**, the display will be:

```
set MFO at (0-199)% =
enable or disable =

ins-Y/N     del     enable     disabl     Varb     quit
```

Key in the percentage of desired Feedrate Override, or select **Varb** and key in the variable number containing the percentage of desired Feedrate Override.

Select **ins-Y/N** or **del** to insert or delete a character at the cursor location.

enable Select **enable** to enable Feedrate Override.

disabl Select **disabl** to disable the Feedrate Override.

Select **quit** to abort command entry and return to the initial selection menu.

Press the **ENTER** key to write the program block.

Program Block Format:

MFO status (1 = enable, 0 = disable) percentage

Example: MFO enable % 100 or G5 1 100
; a Feedrate Override of 100% is enabled

MFO disable % or G5 0
; Feedrate Override is disabled

MISC – >MIR

MIR The **MIR** selection is used to enable or disable the mirror image function. The mirror image function is used to reverse the polarity of a programmed move. The mirror image function is available for each of the four axes. The Mirror function operates in both the incremental and absolute mode. While in the absolute mode, all functions are referenced to the Software Home (established by the G92/Position Register command).

This selection establishes the **G65** command in machine tool language.

Following the selection of **MIR**, the display will be:

```
MIRROR XYZU =

del      Yes      No      skip                quit
```

Select **del** to delete a character at the cursor location.

Select **Yes** to activate an axis for the Mirror function. The axes are activated in the order that they appear on the display. A "Y" will appear for an active axis.

Select **No** to deactivate an axis for the Mirror function. The axes are deactivated in the order that they appear on the display. A "N" will appear for an deactivated axis.

Select **skip** to allow the status of an axis to remain the same. The axes are designated to be skipped in the order that they appear on the display. A "*" will appear for a skipped axis.

MISC – > MIR

Select **quit** to abort command entry and return to the initial selection menu.

Press the **ENTER** key to write the program block.

Program Block Format:

MIRROR Y/N/*

Example: MIRROR Y*NY or G65 Y*NY
; the Mirror function is active for X and U axes, status is not changed for Y axis and the mirror function is deactivated for Z axis

Fhold The **Fhold** selection allows one of the four User Interrupt Inputs to activate a Feedhold.

This selection establishes the **G81** command in machine tool language.

Following the selection of **Fhold,** the display will be:

Interrupt line for FEEDHOLD =					
INT1	**INT2**	**INT3**	**INT4**	**none**	**quit**

Select **INT1, INT2, INT 3** or **INT4** to designate the interrupt as a Feedhold input.

Select **none** to disable all interrupts as Feedhold inputs.

Select **quit** to abort command entry and return to the initial selection menu.

Press the **ENTER** key to write the program block.

Program Block Format:

FEEDHOLD Interrupt Number

Example: FEEDHOLD 3 or G81 3
; Interrupt 3 is designated as a Feedhold input

FEEDHOLD N or G81 N
; All interrupts disabled as Feedhold input

MISC – > MOTOR

Motor The **Motor** selection is used to enable or disable the current command feedback from the motor of a particular axis or axes.

This selection establishes the **G82** or **G83** command in machine tool language.

Following selection of **Motor**, the display will be:

Set motors current command		
on	**off**	**quit**

on Select **on** to enable the motor current command for an axis.

The display will be:

Set motors current command					
ON, axes =					
X	Y	Z	U	del	quit

Select the axis or axes for which the motor current command will be enabled.

Select **del** to delete the character at the cursor location.

Select **quit** to abort command entry and return to the initial selection menu.

Press the **ENTER** key to write the program block.

off Select **off** to disable the motor current command for an axis.

The display will be:

	Set motors current command				
ON, axes =					
X	**Y**	**Z**	**U**	**del**	**quit**

Select the axis or axes for which the motor current command will be disabled .

Select **del** to delete the character at the cursor location.

Select **quit** to abort command entry and return to the initial selection menu.

Press the **ENTER** key to write the program block.

Program Block Format:

MOTOR ON axes
MOTOR OFF axes

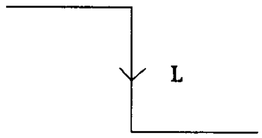
Example:

MOTOR ON XY or G82 XY
; Feedback of Current Command for X and Y axes ON.

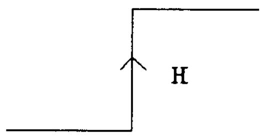
MOTOR OFF ZU or G83 ZU
; Feedback of Current Command for Z and U axes OFF.

INTERRUPT

INTRP The **INTRP** selection is used to configure the four opto-isolated User Interrupts. The Interrupt signals are edge triggered. The Interrupt is a edge triggered signal (see sketch below).



L Triggers an Interrupt set Low



H Triggers an Interrupt set High

Following the selection of **INTRP**, the display will be:

INT 1234 =					
del	**high**	**low**	**disabl**	**skip**	**quit**

This selection establishes the **G64** command in machine tool language.

Select **del** to delete a character at the cursor location.

Select **high** to configure the Interrupt signal as an edge triggered high signal. The Interrupts are configured in the order that they appear on the display. A "H" will appear for a configured Interrupt.

Select **low** to configure the Interrupt signal as an edge triggered low signal. The Interrupts are configured in the order that they appear on the display. A "L" will appear for a configured Interrupt.

Select **disabl** to disable all signals of an Interrupt. The Interrupts are disabled in the order that they appear on the display. A "D" will appear for a disabled Interrupt.

INTERRUPT

Select **skip** to allow the status of an Interrupt to remain the same. The Interrupts are skipped in the order that they appear on the display. A "*" will appear for a skipped Interrupt.

NOTE: The Unidex 400 accepts only one Interrupt Signal input at a time.

Select **quit** to abort command entry and return to the initial selection menu.

Press the **ENTER** key to write the program block.

Program Block Format:

USER INT Interrupt Configuration

Example: USER INT HL*D or G64 HL*D
; Interrupt 1 is edge triggered High
; Interrupt 2 is edge triggered Low
; Interrupt 3 remains as previously configured
; Interrupt 4 is disabled

MATH

MATH FUNCTIONS

The Unidex 400 contains a library of math routines. These routines allow data manipulation in either Floating Point or Hex format. Floating Point format is the default. Hex format is specified by a preceding "H," or "#H:" and is a 32 bit integer.

0:G97 V1 = 1234	define V1 in Floating Point format
1:G21 IN DEC V1 = #V1	display message and V1 value
2:G21 IN HEX V1 = #H:V1	display Hex value of Floating Point format
3:G97 V1 = H,123ABCDE	define V1 in Hex format, 32 bit
4:G21 IN HEX V1 = #H:V1	display Hex value

The Math function also provides several selections for User defined variables. The Unidex 400 has memory allocated for 256 User variables, defined as V0 - V255. Additional variables may be defined from V256 - V9899. User memory is reallocated for these variables from the subheading **more.** Adding one variable will create "V256". Added variables may be cleared under the **clear** heading. Note that the amount of available memory decreases as additional variables are defined (see description of File 9999 later in this section).

Unidex 400 also has predefined system variables named V9900 to V9999. The system variables are defined as follows:

V21

V9900,V9901,V9902,V9903 - X,Y,Z,U absolute position - \$XAP
V9904,V9905,V9906,V9907 - X,Y,Z,U relative position \$XRP
\$XDP

Example:
To display X axis absolute and relative position:

Select **insert, MSG, dispy,** and enter abs pos = #V9900, and rel pos = #V9900

or

G21 abs pos = #V9900, rel pos = #V9904

File 9999

Unidex 400 has predefined file names from 9900 to 9999. File 9999 is for additional variables. The size of this file must be specified under **machine-mdi/edit-math-more** or G98. This file is identical to variable V256-V????. The file data size is four bytes for every variable added under **more**. If "1" was entered under the heading **more**, then four bytes of data could be stored. The first four bytes of file 9999 are accessed by the variable V256, the next four by V257, etc.

The file 9999 may be deleted within the File mode. Parts of file 9999 (V256-V????) may be set to zero using the **clear** command. **Clear** "V256 to V257" sets the first eight bytes of file 9999 and V256,V257 to zero. Note that file 9999 may not be directly edited, however, it can be copied to another file for viewing.

Data may be written to file 9999 one byte at a time under the selection **MSG, F9999**. The start and end byte must be specified. Always allocate 2 extra bytes for system purposes.

Example:

```
0:G98 100                           ; DEFINE VARIABLE 100*4 BYTES
1:G97 V10=10.123                    ; FLOATING FORMAT
2:G97 V11=H,12345678                ; HEX FORMAT
3:G97 V12="ABCD"                    ; ASCII FORMAT
4:G31 <0 1 100> WRITE THIS DATA TO FILE F9999
5:G31 APPEND THIS LINE AFTER ABOVE LINE
6:G31 <0 100> THIS WILL OVER WRITE LINE ONE
7:G31 #V10 #H:V11 #C:V12            ; APPEND
8:G21 #C:V256 #C:V257               ; FIRST EIGHT ASCII BYTES
```

MATH

Files 9902 - 9905 are reserved for axis calibration data. These files may not be deleted or edited. Data is loaded to these files from file 9999 using the **copy** command. This is done following calibration by AEROTECH, Inc. The User should never overwrite these files.

9902,9903,9904,9905 - X,Y,Z,U axis calibration data

File 9901 is used for Step Response data. This file contains the percentage of desired Position and Velocity during a Step Response. This information is useful for manual tuning of the Servo gains.

Following selection of **MATH**, the display will be:

MATH :						
Varb	**=**	**+**	**-**	*****	**/**	**m**

Press the "down" arrow, the display will be:

MATH :						
ins-Y/N	**del**	**clear**	**more**	**arithm**	**logic**	**m**

MATH – > VARB

Press the "down" arrow, the display will be:

MATH :					
(	)	hex	"	quit	m

Press the "down" arrow, the display will be:

MATH :						
A	B	C	D	E	F	m

varb Select **varb** to specify a variable number to be configured.

This selection establishes the **G97** command in machine tool language.

The following selections provide various Math functions:

= Selection of = provides an assignment operator

+ Selection of + provides for addition of two Floating Point numbers. The result will also be a Floating Point number

- Selection of - provides for subtraction of two Floating Point numbers. The result will also be a Floating Point number

* Selection of * provides for multiplication of two Floating Point numbers. The result will also be a Floating Point number

/ Selection of / provides for division of two Floating Point numbers. The result will also be a Floating Point number

MATH – > CLEAR

Select **ins-Y/N** or **del** to insert or delete a character at the cursor location.

clear Select **clear** to return variable(s) settings to zero.

The display will be:

CLEAR VARIABLE from		**to**			
ins - Y/N	**del**	**from**	**to**	**quit**	**m**

This selection establishes the **G99** command in machine tool language.

Select **ins-Y/N** or **del** to insert or delete a character at the cursor location.

from Select **from** to key in the initial variable number to be cleared.

to Select **to** to key in the last variable number to be cleared.

NOTE: **If 100 variables are used, they may be cleared by entering "clear from 256 to 355".**

Select **quit** to abort command entry and return to the initial selection menu.

Press the **ENTER** key to write the program block.

Program Block Format:

clear variables from number to number

Example: clear V 1 20 or G99 1 20

; Variables 1 through 20 will be cleared to zero

MATH – > MORE

more Select **more** to activate additional variables (V256-V9899).

The display will be:

0 - 9644 FOR V255 - V9899 =
ins - Y/N **del** **quit**

Key in the number of desired additional variables.

This selection establishes the **G98** command in machine tool language.

Select **ins-Y/N** or **del** to insert or delete a character at the cursor location.

Select **quit** to abort command entry and return to the initial selection menu.

Press the **ENTER** key to write the program block.

Program Block Format:
define number of additional variables

Example: define V 50 or G98 50
; 50 additional Variables will be available for use (V256-V306)

MATH – > ARITHM

ARITHM The **ARITHM** selection provides a variety of Floating Point functions.

Following selection of **ARITHM**, the display will be:

```
MATH :

!<exp>    SQR    SIN    COS    TAN    quit    m
```

Press the "down" arrow, the display will be:

```
MATH :

ABS    INT    RAD    DEG    ATN    BTF    m
```

! The **!** selection is used to specify an exponent of a Floating Point number.

SQR The **SQR** selection is used to derive the Square Root of a Floating Point number.

NOTE: **The Unidex 400 recognizes only positive Floating Point numbers**

SIN The **SIN** selection is used to derive the Sine value of a floating point angle. The result is a Floating Point number. All angles are expressed in decimal degrees.

COS The **COS** selection is used to derive the Cosine value of a Floating Point angle. The result is a Floating Point number. All angles are expressed in decimal degrees.

TAN The **TAN** selection is used to derive the Tangent value of a Floating Point angle. The result is a Floating Point number. All angles are expressed in decimal degrees.

MATH – > ARITHM

Select **quit** to abort command entry and return to the initial selection menu.

ABS The **ABS** selection is used to express the Absolute value of a Floating Point number. The result is always a positive Floating Point number.

INT The **INT** selection is used to"round-off" the fractional part of any Floating Point number. The result will be a Floating Point integer.

RAD The **RAD** selection is used to convert degrees to radians (both are Floating Point numbers). The result is a Floating Point number in radians.

DEG The **DEG** selection is used to convert radians to degrees (both are Floating Point numbers). The result is a Floating Point number in decimal degrees.

ATN The **ATN** selection is used to derive the Arctangent (Inverse of Tangent) value of a Floating Point number. The result is a Floating Point number in radians.

BTF The **BTF** selection is used to convert a Binary number to a Floating Point number. Floating Point numbers and Binary numbers cannot be used within the same equation unless one or the other is converted. The result is a Floating Point number.

Press the **ENTER** key to write the program block.

The **ARITH** selection establishes the **G97** command in machine tool language.

Program Block Format:

variable name ARITH function

Example: V1 = SIN(40) or G97 V1 = SIN(40)
; Value of variable 1 will be the Sine of 30 degrees or 0.5

MATH – > LOGIC

LOGIC The **LOGIC** selection provides a variety of Binary functions.

Following selection of **LOGIC**, the display will be:

MATH :						
.NOT4.	**.ADD4.**	**.SUB4.**	**.AND4.**	**.OR4.**	**quit**	**m**

Press the "down" arrow, the display will be:

MATH :						
.XOR4.	**.LSL4.**	**.LSR4.**	**.ROL4.**	**.ROR4.**	**FTB**	**m**

NOTE: **The "4" following each logic function represents "four byte" operation, ie. 32 bits.**

.NOT4. The **.NOT4.** selection is used to perform the Logic "NOT", the "Inverse" or the "1's complement" function. The result is a Binary number. This function requires only one operand.

Format ; Binary Number = .NOT4. Binary number

.ADD4. The **.ADD4.** selection is used to perform addition of two Binary numbers. The result is a Binary number.

Format; Binary Number = Binary Number.ADD4.Binary Number

.SUB4. The **.SUB4.** selection is used to perform subtraction of Binary numbers. The result is a Binary number.

Format; Binary Number = Binary Number.SUB4.Binary Number

MATH – > LOGIC

.AND4. The **.AND4.** selection is used to perform the Logic "AND" function. The result is a Binary number.

Format ; Binary Number .AND4. Binary number

.OR4. The **.OR4.** selection is used to perform the Logic "OR" function. The result is a Binary number.

Format ; Binary Number .OR4. Binary number

Select **quit** to abort command entry and return to the initial selection menu.

.XOR4. The **.XOR4.** selection is used to perform the Logic "EXCLUSIVE OR" function. The result is a Binary number.

Format ; Binary Number .XOR4. Binary number

.LSL4. The **.LSL4.** (Logic Shift Left) selection is used to shift a specified number of bits left. It requires two operands in a specified order. The first operand provides the number to be shifted, the second operand specifies the number of bits of the shift. The result is a Binary number.

Format ; Binary Number .LSL4. Binary number

.LSR4. The **.LSR4.** (Logic Shift Right) selection is used to shift a specified number of bits right. It requires two operands in a specified order. The first operand provides the number to be shifted, the second operand specifies the number of bits of the shift. The result is a Binary number.

Format ; Binary Number .LSR4. Binary number

MATH LOGIC

.ROL4. The **.ROL4.** (Rotate Left) selection is used to shift a specified number of bits left and rotate the displaced bits to the far right. It requires two operands in a specified order. The first operand provides the number to be shifted, the second operand specifies the number of bits of the rotation. The result is a Binary number.

Format ; Binary Number .XOR4. Binary number

.ROR4. The **.ROR4.** (Rotate Right) selection is used to shift a specified number of bits right. The displaced bits replace the empty spaces on the far left. It requires two operands in a specified order. The first operand provides the number to be shifted, the second operand specifies the number of bits of the rotation. The result is a Binary number.

Format ; Binary Number .XOR4. Binary number

FTB The **FTB** selection is used to convert a Floating Point number to a Binary number. Floating Point numbers and Binary numbers cannot be used within the same equation, unless one or the other is converted. The result is a Binary number.

Format ; Binary Number .AND4. Binary number

NOTE: **The fractional part of the Floating Point number is rounded to the nearest integer.**

The **LOGIC** selection establishes the **G97** command in machine tool language.

Press the **ENTER** key to write the program block.

Program Block Format:

variable name LOGIC function

Example: V1.SUB4.V14 or G97 V1.SUB4.V14
; Value of variable 14 will be subtracted from the value of variable 1

MATH

The following is the third display of the **MATH** function:

MATH :					
(	**)**	**hex**	**"**	**quit**	**m**

( ) The "(" and ")" selections provide for creation of mathematical phrases that are to be treated as a single term.

hex The **hex** selection is used to specify that a value will be in Hexadecimal format.

" The " selection is used as a delimitter for an ASCII string.

Select **quit** to abort command entry and return to the initial selection menu.

Press the "down" arrow, the display will be:

MATH :						
A	**B**	**C**	**D**	**E**	**F**	**m**

A-F Selections **A** through **F** are used to create Hexadecimal characters.

Press the **ENTER** key to write the program block.

(THIS PAGE INTENTIONALLY LEFT BLANK)

CHAPTER 4: COMMAND SUMMARY AND SAMPLE PROGRAMS

SECTION 4-1: COMMAND SUMMARY

This Chapter consists of a list of machine tool codes that are recognized by the Unidex 400. It is provided as a quick reference, for a more detailed description of any function refer to Chapter 3: The Edit Mode.

Throughout this Chapter lower case letters indicate a value that is required to be entered by the User.

The applicable Edit Mode page number is indicated in bold type with the description.

CODE	COMMAND FORMAT	DESCRIPTION
G0	G0 Xnn Fnn Ybb Fbb Zcc Fcc Udd Fdd	Point-to-Point Index move of 1-4 axes each with a specified Feedrate. **3-11**
G1	G1 Xnn Ynn Znn Unn Fnn	Linear Contouring move of 1-4 axes with one specified Feedrate. **3-12**
G2	G2 Xnn Ynn Znn Unn Inn Jnn Knn Pnn Fnn	Clockwise Circular Contouring of 1-4 axes with one specified Feedrate. **3-16**
G3	G3 Xnn Ynn Znn Unn Inn Jnn Knn Pnn Fnn	Counterclockwise Circular Contouring of 1-4 axes with one specified Feedrate. **3-16**
G4	G4 Fnn	Dwell or system pause in mSec. **3-118/121**
	G4 Inn	Dwell or system pause until I/O input. **3-119**
	G4 M	Dwell or system pause until MST Acknowledge. **3-122**
	G4 C	Dwell or system pause until CYCLE START. **3-120/122**

CODE	COMMAND FORMAT	DESCRIPTION
G5	G5 1/0 nn	User MFO (1 enable, 0 disable) with specified percentage. **3-140**
G6	G6 nn	Ramping time for all axes. **3-104**
G7	G7 XYZU	Send designated axis to the Hardware Home position. **3-26**
G8/G9	G8 G1 Xnn Ynn Znn Unn Fnn G9 G1 Xnn Ynn Znn Unn Fnn	Used for contouring in conjunction with G1, G2, and G3. G8 enables Velocity Profiling, G9 disables Velocity Profiling. **3-14**
G10	G10 X/Y/Z/U 0	Stop Freerun. **3-38**
	G10 X/Y/Z/U1 F+/-nn	Continuous Rotate Freerun **3-30**
	G10 X/Y/Z/U2 F+/-nn Dnn	Freerun Fixed Distance **3-32**
	G10 X/Y/Z/U3 F+/-nn Dnn	Freerun Back /Forth Fixed Distance **3-34**
	G10 X/Y/Z/U4 F+/-nn Dnn	Freerun Back and Forth a set number of Cycles and Distance. **3-36**
G11	G11 X/Y/Z/U nn	Send designated output at conclusion of Freerun. **3- 39**
G12	G12 X/Y/Z/U	Cancel output at conclusion of axis Freerun. **3-43**
G13	G13 X/Y/Z/U	Wait for finish of axis Freerun. **3-45**
G14	G14 Vnn	Set Freerun status to Variable. **3-44**

CODE	COMMAND FORMAT	DESCRIPTION
G15	G15	Disable Machine Mode Tracking Display. **3-131**
G16	G16	Enable Machine Mode Tracking Display. **3-131**
G17	G17 <Vnn> message	Send message to Display and Service Request to the General Purpose Interface Bus (IEEE-488) and Display. **3-124**
G18	G18 <Vnn> message	Send Service Request to the General Purpose Interface Bus (IEEE-488). **3-124**
G19	G19	Enable Joystick. **3-113**
G20	G20 <Vnn> message	Display Message and Send through RS-232 Serial Port. **3-124**
G21	G21 <Vnn> message	Display Message. **3-123**
G22	G22 <Vnn> message	Display Message and Send through IEEE-488 Port. **3-125**
G23	G23	Enable Corner Rounding Mode. **3-66**
G24	G24	Disable Corner Rounding Mode. **3-66**
G25	G25 nn	Send output during continuous run or Feedhold Off. **3-57**
G26	G26 nn	Send output at Stop or Feedhold. **3-58**

CODE	COMMAND FORMAT	DESCRIPTION
G27	G27 X/Y/Z/U Pnn Inn Dnn F1nn F2nn	Override PIDF1F2 Parameters. **3-103**
G28	G28 nn or nn	Repeat Loop "nn" number of times or until input equal "nn" **3-95**
G29	G29 X/Y/Z/U nn	Override Acceleration/Deceleration rate Parameter. **3-102**
G30	G30 nn	Inserts title or comment. **3-97**
G31	G31 < Vnn >	Write message to File 9999. **3-125**
G32	G32 X/Y/Z/U /M0/1	Override Trajectory Type Parameter (0-Linear, 1-Parabolic). **3-108**
G33	G33 X/Y/Z/U/M nn	Override Trajectory Coefficient Parameter (0-Linear, 1-Parabolic). **3-109**
G34	G34 nn	Output the Lower byte of Variable nn in Floating Point format. **3-53**
G35	G35 nn	Immediate Output in Hex format. **3-56**
G36	G36 nn	Output the Lower byte of Variable nn in Hex format. **3-54**
G37	G37 nn	Establish Output in Decimal format. **3-55**
G38	G38 nn	Establish Output in Binary format. **3-52**

CODE	COMMAND FORMAT	DESCRIPTION
G39	G39 B/H/D nn	Input Binary, Hex, or Decimal format to Variable nn. **3-47**
G43	G43 X/Y/Z/U	Enables the positive (CW)limit. **3-136**
G44	G44 X	Disables the positive (CW) limit. **3-136**
G45	G45 X/Y/Z/U nn	Sets the positive (CW) limit. **3-136**
G53	G53 X/Y/Z/U	Enables the negative (CCW) limit. **3-138**
G54	G54 X/Y/Z/U	Disables the negative (CCW) limit. **3-138**
G55	G55 X/Y/Z/U nn	Sets the negative (CCW) limit. **3-138**
G58	G58 M/S/T nn	Sets the M/S/T function Strobe width at nn mSec. **3-61**
G59	G59 M/S/T nn	Sets the M/S/T function Acknowledge delay at nn mSec. **3-62**
G60	G60 Lnn	Directs program flow to a designated Label, no return. **3-91**
	G60Pnn	Directs program flow to a designated Program, no return. **3-91**
	G60 LVnn	Directs program flow to a Label specified by a designated variable, no return. **3-91**

CODE	COMMAND FORMAT	DESCRIPTION
	G60 LB/Hnn	Directs program flow to a Label specified by a Binary or Hex Input, no return. **3-91**
	G60 PVnn	Directs program flow to a Program specified by a designated variable, no return. **3-91**
	G60 PB/Hnn	Directs program flow to a Program specified by a Binary or Hex Input, no return. **3-91**
G61	G61 Lnn	Directs program flow to a designated Label, with return. **3-72/3-82**
	G61 Pnn	Directs program flow to a designated Program, with return. **3-72/3-83**
	G61 LVnn	Directs program flow to a Label specified by a designated variable, with return. **3-73/3-84**
	G61 LB/Hnn	Directs program flow to a Label specified by a Binary or Hex Input, with return. **3-74/3-85**
	G61 PVnn	Directs program flow to a Program specified by a designated variable, with return. **3-73/3-85**
	G61 PB/Hnn	Directs program flow to a Program specified by a Binary or Hex Input, with return. **3-74/3-85**

CODE	COMMAND FORMAT	DESCRIPTION
G62	G62 Lnn Cnn	Directs program flow to Label nn if Condition nn is true, no return. **3-92**
	G62 Lnn Inn	Directs program flow to Label nn if Input is as designated, no return. **3-93**
	G62 Lnn Hnn/Lnn	Directs program flow to Label nn if Input bit is High/Low, no return. **3-93**
	G62 Lnn HA/LA	Directs program flow to Label nn if MST ACK is High/Low, no return. **3-93**
	G62 Pnn Cnn	Directs program flow to Program nn if Condition nn is true, no return. **3-93**
	G62 Pnn Inn	Directs program flow to Program nn if Input is as designated, no return. **3-93**
	G62 Pnn Hnn/Lnn	Directs program flow to Program nn if Input bit is High/Low, no return. **3-93**
	G62 Pnn HA/LA	Directs program flow to Program nn if MST ACK is High/Low, no return. **3-93**

CODE	COMMAND FORMAT	DESCRIPTION
	G62 LVnn Cnn	Directs program flow to Label specified by Variable nn if Condition nn is true, no return. 3-93
	G62 LVnn Inn	Directs program flow to Label specified by Variable nn if Input is as designated, no return. 3-93
	G62 LVnn Hnn/Lnn	Directs program flow to Label specified by Variable nn if Input bit is High/Low, no return. 3-93
	G62 LVnn HA/LA	Directs program flow to Label specified by Variable nn if MST ACK is High/Low, no return. 3-93
	G62 PVnn Cnn	Directs program flow to Program specified by Variable nn if Condition nn is true, no return. 3-93
	G62 PVnn Inn	Directs program flow to Program specified by Variable nn if Input is as designated, no return. 3-93
	G62 PVnn Hnn/Lnn	Directs program flow to Program specified by Variable nn if Input bit is High/Low, no return. 3-93
	G62 PVnn HA/LA	Directs program flow to Program specified by Variable nn if MST ACK is High/Low, no return. 3-93

CODE	COMMAND FORMAT	DESCRIPTION
	G62 Lnn Tn	Directs program flow to Label nn if Interrupt signal n is received, no abort. **3-93**
	G62 Pnn Tn	Directs program flow to Program nn if Interrupt signal n is received, no abort. **3-93**
	G62 Lnn An	Directs program flow to Label nn if Interrupt signal n is received, abort. **3-93**
	G62 Pnn An	Directs program flow to Program nn if Interrupt signal n is received, abort. **3-93**
	G62 LVnn Tn	Directs program flow to Label specified by Variable nn if Interrupt signal n is received, no abort. **3-93**
	G62 PVnn Tn	Directs program flow to Program specified by Variable nn if Interrupt signal n is received, no abort. **3-93**
	G62 LVnn An	Directs program flow to Label specified by Variable nn if Interrupt signal n is received, abort. **3-93**
	G62 PVnn An	Directs program flow to Program specified by Variable nn if Interrupt signal n is received, abort. **3-93**

CODE	COMMAND FORMAT	DESCRIPTION
G63	G63 Lnn Inn	Directs program flow to Label nn if Input is as designated, with return. **3-75**
	G63 Lnn Hnn/Lnn	Directs program flow to Label nn if Input bit is High/Low, with return. **3-75**
	G63 Lnn HA/LA	Directs program flow to Label nn if MST ACK is High/Low, with return. **3-85**
	G63 Pnn Inn	Directs program flow to Program nn if Input is as designated, with return. **3-75**
	G63 Pnn Hnn/Lnn	Directs program flow to Program nn if Input bit is High/Low, with return. **3-77**
	G63 Pnn HA/LA	Directs program flow to Program nn if MST ACK is High/Low, with return. **3-83**
	G63 LVnn Inn	Directs program flow to Label specified by Variable nn if Input is as designated, with return. **3-77**
	G63 LVnn Hnn/Lnn	Directs program flow to Label specified by Variable nn if Input bit is High/Low, with return. **3-75**
	G63 LVnn HA/LA	Directs program flow to Label specified by Variable nn if MST ACK is High/Low, with return. **3-87**

CODE	COMMAND FORMAT	DESCRIPTION
	G63 PVnn Inn	Directs program flow to Program specified by Variable nn if Input is as designated, with return. **3-75**
	G63 PVnn Hnn/Lnn	Directs program flow to Program specified by Variable nn if Input bit is High/Low, with return. **3-75**
	G63 PVnn HA/LA	Directs program flow to Program specified by Variable nn if MST ACK is High/Low, with return. **3-85**
	G63 Lnn Tn	Directs program flow to Label nn if Interrupt signal n is received, no abort. **3-79**
	G63 Pnn Tn	Directs program flow to Program nn if Interrupt signal n is received, no abort. **3-79**
	G63 Lnn An	Directs program flow to Label nn if Interrupt signal n is received, abort. **3-79**
	G63 Pnn An	Directs program flow to Program nn if Interrupt signal n is received, abort. **3-79**

CODE	COMMAND FORMAT	DESCRIPTION
	G63 LVnn Tn	Directs program flow to Label specified by Variable nn if Interrupt signal n is received, no abort. **3-79**
	G63 PVnn Tn	Directs program flow to Program specified by Variable nn if Interrupt signal n is received, no abort. **3-79**
	G63 LVnn An	Directs program flow to Label specified by Variable nn if Interrupt signal n is received, abort. **3-79**
	G63 PVnn An	Directs program flow to Program specified by Variable nn if Interrupt signal n is received, abort. **3-79**
G64	G64nnnn	Configures User Interrupt Signals 1 thru 4 as either High, Low, Disabled or Skip. **3-146**
G65	G65 nnnn	Enables or disables the mirror image function. **3-141**
G66	G66 n	Selects the amount of filtering for corner-rounding, 0 thru 7. **3-67**
G70	G70	English Mode **3-65**
G71	G71	Metric Mode **3-65**
G80	G80	Enables PSO option board **3-116**

CODE	COMMAND FORMAT	DESCRIPTION
G81	G81 n	Enables User Interrupt n to activate a Feedhold. **3-143**
G82	G82 X/Y/Z/U	Enable designated axis motor **3-144**
G83	G83 X/Y/Z/U	Disable designated axis motor **3-144**
G90	G90	Absolute Mode **3-64**
G91	G91	Incremental Mode **3-64**
G92	G92 Xnn/Ynn/Znn/Unn	Establishes a Software Home (Position Register) for desired axes. **3-27**
	G92	Sets Software Home Position of all axes to zero. **3-27**
G93	G93	Beeper ON **3-132**
G94	G94	Beeper OFF **3-132**
G95	G95 X/Y/Z/Un nn	Enables optional Handwheel control for the designated axes on Input n with a scale factor of nn. **3-114**
	G95 Dn	Disables optional Handwheel for all axes on Input n. **3-114**

CODE	COMMAND FORMAT	DESCRIPTION
G96	G96 X/Y/Z/U na nb nn	Establishes Auto-Focus control for designated axes through Input na with a polarity of nb at a scale factor of nn. **3-133**
G97	G97	Enables Math Function as well as establishes System Variables (V=nn). **3-151** Math Function Summary Varb, =, +, -, *, / **Floating Point Numbers:** ! exponent, SQR Square Root, SIN Sine, COS Cosine, TAN Tangent, ABS Absolute Value, INT Integer, RAD Radian, DEG Radians to Degrees, ATN Arctangent, BTF Binary to Floating **Logic/Binary Number Functions:** .Not4., .ADD4., .SUB4., .AND4., .OR4., .XOR4., .LSL4., .LSR4., .ROR4., FTB
G98	G98 nn	Activates nn additional Variables. **3-153**
G99	G99 Vn nn	Clear Variable settings n through nn to zero. **3-152**

CODE	COMMAND FORMAT	DESCRIPTION
M2	M2	Indicates End of Program. **3-99**
M47	M47	Initiates a re-run of the program. **3-99**
M	Mnnnn	M Function Output (0000 to 0FFFF) must consist of 4 digits. M2 entered as M002 M47 entered as M047 **3-62**
S	Snnnn	S Function Output (0000 to 0FFFF) must consist of 4 digits. **5-62**
T	Tnnnn	T Function Output (0000 to 0FFFF) must consist of 4 digits. **3-62**
N	Nnn	Establishes Label nn. **3-98**
N-1	N-1	Carriage Return. **3-97**

SECTION 4-2: SAMPLE PROGRAMS

PROGRAM #1

This program moves an X-Y stage to positions on a circle with a radius 2 inches and then reads the input bus, masking out the top 8 bits.

Machine Tool	Menu Driven	Comments
0:G30 U400 EXAMPLE PROGRAM #1		;program title
1:G90	absolute mode	;absolute positioning
2:G70	English mode	;English conversion factor
3:G1 F1000.	linear F1000.	;feedrate 1000. in/min
4:G97 V1=0	V1=0	;clear angle
5:N1	Label 1	;label
6:G97 V2=2*COS(V1)	V2=2*COS(V1)	;calculate move
7:G97 V3=2*SIN(V1)	V3=2*SIN(V1)	;calculate move
8:G1 XV2 YV3	linear XV2 YV3	;linear move to position
9:G39 H4	input to V4 (HEX format)	;input to V4 in Hex format
10:G97 V4=H,FF.AND4.V4	V4=H,FF.AND4.V4	;logical AND of data
11:G97 V5=BTF(V4)	V5=BTF(V4)	;convert to Floating Format
12:G21 INPUT AT #V1 IS #H:V4 HEX AND #V5 DECIMAL	MESG INPUT AT #V1 IS #H:V4 HEX AND #V5 DECIMAL	;display both formats
13:G4 C	dwell until cycle start	;wait for Cycle Start
14:G97 V1=V1+45	V1=V1+45	;increment angle (degrees)
15:G62 L1 C1<360	G62 Label 1 C1<360	;jump to label "1" if V1<360
16:G97 V1=0	V1=0	;reset angle
17:G60 L1	GOTO LABEL 1	;jump to label "1"
18:M2	end prog	;end of program

PROGRAM #2

This program moves the X and Y axes back and forth with a 100mSec dwell between moves.

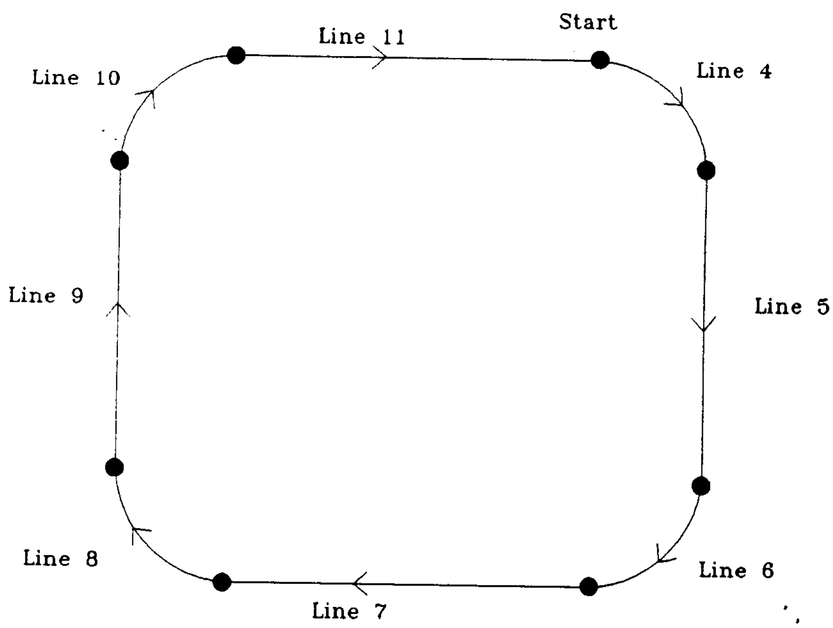
Machine Tool	Menu Driven	Comments
0:G30 U400 EXAMPLE PROGRAM #2		;program title
1: G92	reset relative posn	;set software home
2: G70	English mode	;English conversion factor
3: G1 F1000.	linear F1000.	;feedrate 1000. in/min.
4: N1	LABEL 1	;label
4: G1 X10. Y10.	linear X10. Y10.	;linear move to position
5: G4 F100	dwell 100 ms	;wait for 100 mSec.
6: G1 X-10. Y-10.	linear X-10. Y-10.	;linear move to position
7: G4 F100	dwell 100 ms	;wait for 100 mSec.
8: G60 L1	GOTO LABEL 1	;jump to label "1"

PROGRAM #3

This program produces a square with three inch sides and one inch radius corners at a constant velocity with 90 degree arc segments.

Machine Tool	Menu Driven	Comments
0:G30 U400 EXAMPLE PROGRAM		;program title
1: G92	reset relative posn	;set software home
2: G70	English mode	;English conversion factor
3: G1 F1000.	linear F1000.	;feedrate 1000. in/min.
4: G8 G2 X1. Y-1. I0. J-1.	vel cw X1. Y-1. I0. J-1.	;enables velocity profiling for CW circular contouring
5: G8 Y-3.	vel linear Y-3.	;enables velocity profiling for linear contouring (constant velocity square edge)
6: G8 G2 X-1. Y-1. I-1. J0.	vel cw X-1. Y-1. I-1. J0.	;enables velocity profiling for CW circular contouring
7: G8 G1 X-3.	vel linear X-3.	;enables velocity profiling for linear contouring (constant velocity square edge)
8: G8 G2 X-1. Y1. I0. J1.	vel cw X-1. Y1. I0. J1.	;enables velocity profiling for CW circular contouring
9: G8 G1 Y3.	vel linear Y3.	;enables velocity profiling for linear contouring (constant velocity square edge)

Machine Tool	Menu Driven	Comments
10: G8 G2 X1. Y1. I1. J0.	vel cw X1. Y1. I1. J0.	;enables velocity profiling for CW circular contouring
11: G9 G1 X3.	vel linear X3.	;enables velocity profiling for linear contouring (constant velocity square edge)
12: G21 CIRCLE DONE	MESG CIRCLE DONE	;display message
13: G4 C	dwell until cycle start	;wait for Cycle Start



PROGRAM #4
Input Bus/Data Example

Machine Tool	Menu Driven	Comments
0:G30 U400 EXAMPLE PROGRAM #4		;program title
1: N1	LABEL 1	;label
2:: G39 B0	input to V0 (BCD format)	;store 12 Input Bits to V0 in BCD Format
3: G39 D1	input to V1 (DEC format)	;store 12 Input Bits to V1 in DEC Format
4: G39 H2	input to V2 (HEX format)	;store 12 Input Bits to V2 in HEX Format
5: G21 #V0	MESG #V0	;display contents of V0
6: G21 #V1	MESG #V1	;display contents of V1
7: G21 #H:V2	MESG #H:V2	;display contents of V2 in Hex format
8: G21 #V1 #V0	MESG #V1 #V0	;display contents of V1 and V0
9: G4 F1000	dwell 1000 ms	;wait for 1000 ms
10: G60 L1	GOTO LABEL 1	;jump to label "1"

CHAPTER 5: FILE MODE

SECTION 5-1: INTRODUCTION

The Unidex 400 provides a full range of of file management capabilities. Files may be listed, output through RS-232 or IEEE-488, input through RS-232 or IEEE-488, copied or erased.

Valid User file names consist of numbers 0 through 9900. Files 9901 through 9999 are reserved for system data (See below).

Files 9902 through 9905 are reserved for axis calibration data.
File 9999 is reserved for additional variables.

Reserved files may not be deleted or edited, data within these files may be viewed by copying the file to a User file name and entering the Edit Mode. Refer to Chapter 3: Edit Mode, Math Functions for additional information concerning reserved files.

The following sections provide detailed information concerning the File Mode.

SECTION 5-2: GETTING STARTED

Following Power-Up, the initial selection screen below is displayed:

```
UNIDEX 400                    Version x.xx

  RAM:ok   ROM:ok   SETUP:ok
  User RAM: xxxxxx

edit    file    setup    test    machine    PSWD
```

NOTE: **The PSWD option (F6) is displayed only if the Password function has been activated from the Setup Mode.**

Press the F2 key to enter the File Mode. The following screen will be displayed:

```
    file management



dir        out        in        copy        erase        quit
```

A description of the display follows:

SECTION 5-3: DIRECTORY

Press the F1, **DIR** key to display a list of all of the file numbers currently resident in the Unidex 400 memory. Files are listed in the order that they were created.

> NOTE: **Titles may be added to a file number by use of the G30 command while in the Edit Mode. If a title has been created, it will be displayed next to the file number when a directory listing is requested.**

A sample Directory display is illustrated below:

```
 9902
0002 0006 0071 0055 0022
0000 : DISPLAY EXAMPLE
0056 0003 0011

 dir        out        in        copy        erase        quit
```

Press the F1 key to move through the Directory listing.

Following the Directory listing, the following information is displayed:

```
0056 0003 0011
BACKGROUND : 0   USER DATA : 0
FREE : 174736


dir      out      in      copy      erase      quit
```

BACKGROUND The number following **BACKGROUND** indicates the amount of memory that has been allocated for background functions such as position fetching. Currently background memory allocation is not implemented.

USER DATA The number following **USER DATA** indicates the amount of memory that is allocated for program stacks. Refer to Chapter 7: Setup Mode, General Parameter 010.

FREE The number following **FREE** indicates the total amount of RAM available to the User. Refer to Chapter 7: Setup Mode, General Parameter 028, if additional memory has been added.

SECTION 5-4: OUTPUT

The F2, **out** selection provides for file transfer through either the RS-232 Serial Port (**PORT A**-P30) or the **IEEE-488** connector (P11). File output format and an output example is provided in Item 5-4-1.

Following selection of **out** from the initial File menu, the display will be:

```
        file management



  output                  to              form    D   <cr>
    del        all       RS232      I488       form      quit      m
```

Key in the File number to be output.

del Select **del** to delete a character at the cursor location.

all Select **all** to output all of the files stored in the Unidex 400's memory to an external device.

RS232 Select **RS232** to send files through the RS-232 Serial Port (**PORT A**) to an external device. An **R** will appear on the display.

I488 Select **I488** to send files through the **IEEE-488** Serial Port to an external device. An **I** will appear on the display.

form Select **form** to toggle between Indirect and Direct format. The designated file will be transferred in one of these formats. The Indirect format translates the "G" machine tool code into English-like mnemonics, adds line numbers and adds a <LF> after each <CR>. The Direct format sends the "G" machine tool code only. Both formats send data in ASCII format and send a brief header.

quit Select **quit** to abort file output and return to the initial file management display.

Press the "down" arrow for additional options.

ins-Y/N Select **ins-Y/N** to toggle character insert ON(Y) or OFF(N). When ON, characters are inserted without overwriting existing text.

5-4-1 FILE OUTPUT FORMAT

Following is the suggested format of a file to be output from the Unidex 400:

1) Send File Header
   Send File Name
2) then "%"
3) IF (indirect format)
   - Send Line Number (ASCII)
   - Send English translation of U400 commands
   - Send CRLF after each line
   - Repeat until End of File (EOF)

   ELSE (direct format)
   - Send direct U400 "G codes"
   - Send CRLF after each line
   - Repeat until EOF

   ENDIF
4) Send "End 1 File" character. (Refer to General Parameter 006)
5) IF ("all file" mode)
   - IF (not all transferred)
     - GOTO 1)
6) Send "End All File" (EAF) character (General Parameter 007)
7) IF (General Parameter 035 = 1)
   - Wait for two byte checksum feedback
     - BIN format : LSB first, MSB second
   - Display error if checksum does not match

   ENDIF

NOTE:

LF = "Line Feed" - 0A Hexadecimal or 10 Decimal
CR = "Carriage Return" - 0D Hexadecimal or 13 Decimal

The Checksum is accumulated on all characters sent up to and including the EOF and EAF character. It is accumulated as an unsigned 8 bit quantity to a 16 bit signed accumulator.

Example:
Output all files through the RS-232 port in Indirect format.

```
-->Unidex 400      <CRLF>
   File Name ????%
   line 1          <CRLF>
1: line 2          <CRLF>
    ...
  <EOF>            - End of File Command  (Refer to General Parameter 006)
  <EAF>            - End All File Command  (Refer to General Parameter 007)
  (wait for checksum?)
   done
```

SECTION 5-5: INPUT

The F3, **in** selection provides for file input through either the RS-232 Serial Port (**PORT** A-P30) or the **IEEE-488** connector (P11).

All Serial input data is ignored for single file input until a "%" is received. Data that follows the "%" is transferred to a User's file. For multiple file input, the file name must precede the "%" character."

Following selection of **in** from the initial File menu, the display will be:

```
     file management


 input               from
  ins-Y/N     del      all      RS232      I488      quit
```

Key in the File name (number) to be input.

ins-Y/N	Select **ins-Y/N** to toggle character insert ON(Y) or OFF(N). When ON, characters are inserted without overwriting existing text.
del	Select **del** to delete a character at the cursor location.
all	Select **all** to input all of the files available from the source until a EAF is received.
RS232	Select **RS232** to receive files through the RS-232 Serial Port (**PORT A**). An **R** will appear on the display.
I488	Select **I488** to receive files through the **IEEE-488** Serial Port. An **I** will appear on the display.
quit	Select **quit** to abort file input and return to the initial file management display.

5-5-1 FILE INPUT FORMAT

The following are the suggested formats for Unidex 400 file input.

Single File Input

1) Enter file name from file menu

2) IF (file already exists) feedback error code

   ELSE

   send XON

3) Read Serial Characters until % is encountered

4) Dump Data to File Memory
   until End of File and End of All File characters are received

5) Feedback Checksum?
   (BIN format : LSB first, MSB second)

DONE

NOTE:

1) Set General Parameter 034 to 1, to enable the checksum echo.

2) XON is 17 Decimal, 11 Hexidecimal
XOFF is 19 Decimal, 13 Hexadecimal

3) Unidex 400 will send XOFF, an error code, and stop accepting characters if the memory becomes full.

4) End of File (EOF) is the End of File character as set by General Parameter 006. The default value is 9 Decimal.

5) End All File (EAF) is the End of All File character as set by General Parameter 007. The default value is 17 Decimal.

6) The Checksum is accumulated on all characters sent up to and including the EOF and EAF character. It is accumulated as an unsigned 8 bit quantity to a 16 bit signed accumulator.

Multiple File Input

1) Send XON command

2) Scan input stream for last four ASCII numbers preceeding "%". (this is the file name), or for the End All File character. If End All File is received, Goto 5)

3) Send XOFF. Check file name from step 2) above, if the file already exists, feedback error code and stop, otherwise send XON.

4) Dump Data to file memory until End of File character is received

5) Feedback Checksum?

DONE

NOTE:

1) Set General Parameter 034 to 1, to enable the checksum echo.

2) XON is 17 Decimal, 11 Hexidecimal
XOFF is 19 Decimal, 13 Hexadecimal

3) Unidex 400 will send XOFF, an error code, and stop accepting characters if the memory becomes full.

4) End of File (EOF) is the End of File character as set by General Parameter 006. The default value is 9 Decimal.

5) End All File (EAF) is the End of All File character as set by General Parameter 007. The default value is 17 Decimal.

6) The Checksum is accumulated on all characters sent up to and including the EOF and EAF character. It is accumulated as an unsigned 8 bit quantity to a 16 bit signed accumulator.

Example:
Transfer of two files to Unidex 400

Start Transmission

--> Unidex 400 is fun to use	- don't care data
10%	- send following data to file "0010"
{... Program body ... }	- this data to file
<9>	- default End of File command
abcdefghijkln	- don't care data
1234%	- File Name
{... Program body ...}	
<9>	- default End of File command
<17>	- default End All File command
(Echo Checksum ?)	

SECTION 5-6: COPY

The F4, **copy** selection allows file duplication. A file may not be copied to a name that already exists.

Following selection of **copy** from the initial File menu, the display will be:

```
        file management


copy S              T             <cr>
  ins-Y/N     del      source                target      quit
```

ins-Y/N Select **ins-Y/N** to toggle character insert ON(Y) or OFF(N). When ON, characters are inserted without overwriting existing text.

del Select **del** to delete a character at the cursor location.

source Moves cursor to the display location where the Source file name (number) is to be entered. The **source** file is the file to be copied.

target Moves cursor to the display location where the Target file name (number) is to be entered. The **target** file is the File that will be created.

quit Select **quit** to abort copy function and return to the initial file management display.

SECTION 5-7: ERASE

The F5, **erase** selection is used to erase a file(s) from memory.
Following selection of **erase** from the initial File menu, the display will be:

```
      file management



  erase                   <cr>
   ins-Y/N      del         all       ALL                   quit
```

ins-Y/N Select **ins-Y/N** to toggle character insert ON(Y) or OFF(N). When ON, characters are inserted without overwriting existing text.

del Select **del** to delete a character at the cursor location.

all Select **all** to erase all files from the Unidex 400 directory. As a safeguard, the following procedure is necessary to erase all files.

Select **all** from the Erase menu.

Press **ENTER** to initiate the Erase procedure.

Select **ALL** from the Erase menu.

Press **ENTER** to finalize the Erase procedure.

ALL Used in conjunction with **all** to erase all of the files from the Unidex 400 directory. (See above)

quit Select **quit** to abort the Erase function and return to the initial file management display. Select **quit** again to return to the main selection menu.

(THIS PAGE INTENTIONALLY LEFT BLANK)

CHAPTER 6: THE MACHINE MODE

SECTION 6-1: INTRODUCTION

The Machine Mode is the mode in which programs are run and axis motion will occur. From within this mode, the User can run previously created programs, enter direct motion commands, jog an axis, move an axis pair by joystick control or send an axis or axes Home. The Machine Mode display updates the axis position and MFO input every .25 seconds.

Prior to entering the Machine Mode, make certain that the appropriate Parameters have been established. (See Chapter 7: The Setup Mode)

WARNING: MOST DISPLAY SELECTIONS FROM THE MACHINE MODE PRODUCE AXIS MOTION. PRIOR TO ENTERING THE MACHINE MODE, MAKE CERTAIN ALL HARDWARE IS CONFIGURED SUCH THAT NO INJURY TO PERSONNEL OR DAMAGE TO EQUIPMENT WILL OCCUR.

The following sections provide detailed information concerning Machine Mode programming and operation.

SECTION 6-2: GETTING STARTED

Following Power-Up, the initial selection screen below is displayed:

```
UNIDEX 400                    Version x.xx

  RAM:ok   ROM:ok   SETUP:ok
  User RAM: xxxxxx

edit    file    setup    test    machine    PSWD
```

NOTE: **The PSWD option (F6) is displayed only if the Password function has been activated from the Parameter Mode.**

Press the F5 key to enter the Machine Mode. The following screen will be displayed:

```
X   0000.0000        Z   0000.0000            MFO  100 %
Y   0000.0000        U   0000.0000


home      jog      mdi      run      slew      quit      m
```

A description of the display follows:

The upper left portion of the display contains tracking position information for each active axis (as determined by existence checking - Chapter 7, Axis Parameter x37).

```
X   0000.0000        Z   0000.0000            MFO  100 %
Y   0000.0000        U   0000.0000


home      jog      mdi      run      slew      quit      m
```

Relative tracking position is displayed for each axis as the program is run. Position is displayed either in machine or program steps (See Chapter 7, General Parameter 009).

The upper right portion of the display contain s the **MFO %** (Manual Feedrate Override) indication.

```
X   0000.0000        Z   0000.0000              MFO  100 %
Y   0000.0000        U   0000.0000



home      jog      mdi      run      slew      quit      m
```

The **MFO %** setting is the number by which the programmed feedrate will be multiplied during programmed moves.

The **MFO %** value displayed at this location has been derived in one of the following ways:

> If an external MFO Potentiometer is connected, and General Parameter 23 is set at a non zero number, the MFO indication will reflect the external MFO Pot's setting.
>
> If the program being run contains an enabled **UMFO/G5** command, the **MFO %** indication will reflect the percentage set by this command.
>
> If General Parameter 23 is set to zero, the **MFO %** indication will be 100% and the external MFO Pot will be disabled (Default).

SECTION 6-3: THE MACHINE MODE

The following operations are available from the Machine Mode.

6-3-1 HOME

home The **home** function of the Machine Mode is used to send any active axis to the Home position . A Home move is one of two types; 1) a Home move following a Power-Up, or, 2) a run time Home move. (Refer to Figures 3-1 and 3-2.)

1) Power-Up Home

Following Power-Up the selected axis is sent in the direction specified by Axis Parameter x02. The axis then "looks" for the the Home Limit Switch at a feedrate specified by Axis Parameter x05. When the Home Limit Switch is encountered, the axis moves away from the switch a distance specified by Axis Parameter x06 at a feedrate specified by Axis Parameter x05.

When this move is complete, the axis moves at .5 machine steps/mSec. until the Marker pulse is found. When the Marker is located, the axis continues to move the distance (in machine steps) specified by Axis Parameter x07. This location then becomes the Home position. The absolute and relative positions 1are cleared at this point.

NOTE: **The only time the absolute position is cleared is following a Power-Up Home move. The relative position may be cleared at any time , by use of the G92 command.**

2) Run Time Home

A standard run time Home cycle is done in the same manner as the Power-Up Home cycle, except that the absolute position is known. The standard Home cycle feedrate is set by Axis Parameter x04.

Refer to Chapter 3, Edit Mode - **Motion-- >Home** and Chapter 7, Setup Mode - Axis Parameters, for additional information on the Home cycle.

Following selection of Home from the initial Machine Mode menu, the display will be:

```
X   0000.0000        Z   0000.0000        MFO  100 %
Y   0000.0000        U   0000.0000

HOME AXES
   X        Y        Z        U                     quit
```

X-U Select the axis to be sent Home.

quit Select **quit** to return to the initial Machine Mode display.

6-3-2 JOG

jog The **jog** function is used to provide axis movement by use of the arrow keys. The horizontal (left/right) keys are used to move either the X or Z axes, the vertical (up/down) keys will move either the Y or U axes, depending on the selected axis pair.

NOTE: **The jog function is not available when communication is taking place through the Unidex 400 Serial Ports (Port A and the Terminal Port).**

Select **jog** from the initial Machine Mode display.

The display will be:

```
X   0000.0000        Z   0000.0000        MFO  100 %
Y   0000.0000        U   0000.0000

JOG : XY    velocity    F/4
   Z-U        high        index      inc      dec      quit
```

Z-U Press the F1 key to toggle between axis pairs XY and ZU. This selection activates the axes pair for arrow key control. The active axes pair is displayed in the **JOG :** status line.

high/low Select F2 to toggle between a **high** and **low** feedrate range. The feedrate "F" is established by Axis Parameter x08. This feedrate is scaled by the value selected by the **inc/dec** selections. The **high/low** range operates in conjunction with the **inc/dec** values, scaling the selected values by 2.

vel/index Select F3 to toggle between a **velocity** or **index** type of axis movement. When performing a velocity type move, the axis will move at a constant feedrate the entire time that the arrow key is depressed. When performing an index type move, the axis will move a fixed distance at the constant feedrate for each key depression. For a velocity type move, the feedrate may be increased or decreased by use of the **inc/dec** selections. For an index type move, both the move distance and feedrate may be increased or decreased.

inc/dec Select F4 or F5 to increase or decrease the jog feedrate, and distance (for an index move only). The **high/low** selection directly effects this value, scaling the selected values by 2.

quit Select **quit** to return to the initial Machine Mode display.

6-3-3: MDI

mdi Select **mdi** (Machine Direct Interface) to enter a single command to the Unidex 400. Command selections may be made from the Unidex 400's Front Panel Menu Mode or the Keyboard Mode . The menu selections in this mode are identical to those offered in the Edit Mode with exception of the Flow commands which are not applicable.

Following selection of **mdi**, the display will be:

```
X   0000.0000        Z   0000.0000        MFO  100 %
Y   0000.0000        U   0000.0000



  ins-Y/N     del      clear      size              quit
```

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

clear Select **clear** to delete the entire command line at the cursor location.

size Select **size** to move the command line down on the Machine Mode display.

NOTE: **Press the ENTER key to repeat a command line.**

quit Select **quit** to return to the initial Machine Mode display.

6-3-3 RUN

run Select **run** from the initial Machine Mode display to execute a User program. The display will be:

```
X   0000.0000        Z   0000.0000        MFO  100 %
Y   0000.0000        U   0000.0000

file (0 to 9900):
  ins-Y/N     del     SINGLE     lastE     lastR     quit
```

Key in the file number to be run. The following selections are also available :

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

SINGLE Press the F3 key to toggle between **SINGLE** and **AUTO**. When in the **SINGLE** mode, one program line is executed and the program is stopped until the Front Panel's **CYCLE START** key is depressed. The program is processed one block at a time in this mode. This mode is useful when debugging programs.

While in the **SINGLE** mode, the Front Panel's **S3** key may be used to toggle between a display of several blocks of the program being run, or the axis tracking display.

NOTE: **If the Tracking display has been disabled by the G15 command, the program block display function will not be active.**

Press S3 from the standard Run display to see (a sample program is displayed):

```
xxxx  : G 93   : G4  F5000   : G94  : G4  C   :   G21  123
  : G4    F1000   : G21  :   G98     :    M47


file (0 to 9900):
  ins-Y/N       del        SINGLE       lastE       lastR      quit
```

The Program line may be interpreted as follows:

xxxx = Program name (number)
G93 (first program block after program name) = program block just completed
G4 F5000 (second program block after program name) = program block that will run when the **CYCLE START** key is pressed
G94 .... = program blocks yet to run

AUTO The **AUTO** mode will run a program continuously from start to finish.

lastE Select **lastE** to recall the number (name) of the last edited program. Press **ENTER** to initiate the program run.

lastR Select **lastR** to recall the number (name) of the last run program. Press **ENTER** to initiate the program run.

quit Select **quit** to return to the initial Machine Mode display.

After a program run has been initiated, the following is displayed:

```
X   0000.0000        Z   0000.0000        MFO  100 %
Y   0000.0000        U   0000.0000        FEEDHOLD


xxxx PROGRAM RUNNING  !
  stop              SINGLE      abort                  quit
```

FEEDHOLD The **FEEDHOLD** function is toggled ON and OFF by the Unidex 400's Front Panel **FEEDHOLD** key. When ON the **FEEDHOLD** message is displayed and all axes motion is stopped. When toggled OFF, axes motion resumes.

xxxx Indicates the name (number) of the program being run.

stop Press the F1 key to halt program execution. The block currently being processed will be completed then the display will return to the initial Machine Mode menu.

NOTE: **Press the "down" arrow of the initial Machine Mode display for the "cont" selection. This selection will resume execution of the "stopped" program.**

SINGLE Press the F3 key to toggle between **SINGLE** and **AUTO**. When in the **SINGLE** mode, one program line is executed and the program is stopped until the Front Panel's **CYCLE START** key is depressed. The program is processed one block at a time in this mode. This mode is useful when debugging programs.

While in the **SINGLE** mode, the Front Panel's **S3** key may be used to toggle between a display of several blocks of the program being run, and the axis tracking display..

The **AUTO** mode will run a program continuously from start to finish.

abort Select **abort** to terminate execution of the current program block and return to the initial Machine Mode display.

quit Select **quit** to stop program execution after current block is completed, then return to the initial Machine Mode display.

6-3-5 SLEW

slew Select **slew** from the initial Machine Mode display to provide manual axis movement by means of a Joystick. As a safety feature, the Slew Mode cannot be entered if a Joystick is not connected to the Unidex 400.

NOTE: **If a Joystick is connected to the Unidex 400 after power-up, the system must be "reset" in order for the Joystick or Trackball to be recognized. The display will be:**

In addition to being accessible from the main Machine Mode it may also be activated from the "Mdi" mode or from the User's program (G19).

Following selection of **slew**, the display will be:

```
X    0000.0000        Z    0000.0000        MFO  100 %
Y    0000.0000        U    0000.0000

JOY_STICK    SLEW     h-x      v-y  p
                                abort                   quit
```

The following selections may be made from the buttons of the optional Joystick:

The left-most button is used to toggle between the horizontal (h) and vertical (v) axis pairs as established by General Parameter 008 (see Chapter 7: The Setup Mode). The active axes pair is displayed in the **JOY_STICK** status line.

The upper left-center button is used to select one of two modes of Joystick motion:

The Absolute Position Mode coordinates axis motion to the amount of Joystick deflection (refer to axis parameter x18).

The Velocity Mode correlates axis speed to the amount of Joystick deflection. The velocity may be configured as high or low by Axis Parameters x19 and x20.

Depending on the selection, the last character of the Joystick status line will be **p**, **h** or **l** for Absolute Position, high velocity or low velocity.

The button on the end of the Joystick is used to abort Joystick control and return to the initial Machine Mode display.

abort Press the F4 key to abort Joystick control and return to the initial Machine Mode display.

quit Select **quit** to stop program execution after the current block is completed, then return to the initial Machine Mode display.

From the initial Machine Mode display, press the "down" arrow for additional Machine Mode options:

X	0000.0000	Z	0000.0000	MFO	100 %
Y	0000.0000	U	0000.0000		
cont	wheel			keyb	m

6-3-6 CONT

The **cont** selection is displayed on this menu only when program execution has been halted by use of the **stop** selection (from the **run** menu). Press the F1 key to resume the run of the program.

6-3-7 WHEEL

The **wheel** selection enables one or both of the Handwheel inputs and links them to the specified axis. Following this selection, the display will be:

```
X   0000.0000         Z   0000.0000            MFO  100 %
Y   0000.0000         U   0000.0000


HANDWHEEL 1  X  SCALE  (S2  +, S3 -)  : 1
     X          Y         Z          U        input       quit
```

The Handwheel status line illustrated in the display above is interpreted as follows:

HANDWHEEL 1 (input) **X** (axis) **SCALE** (**S2 +**, **S3** - Front Panel keys) : **1** (scale factor)

The following operations are available from the **wheel** display:

X-U Select the axis that is to be controlled by the Handwheel.

input Select **input** to establish which input (1 or 2) is assigned to this handwheel.

Press the Front Panel **S2** key to increase the handwheel scale factor, or the Front Panel S3 key to decrease the handwheel scale factor. A scale factor of "1" (unity scaling) is the initial value presented. Pressing the **S2** or **S3** keys increment or decrement the scale factor by the value established in general Parameters 020 and 021. (Refer to Chapter 7 - The Setup Mode.)

quit Select **quit** to abort the Handwheel Mode and return to the initial Machine Mode display.

6-3-8 KEYBOARD/MENU

The F6 key is used to toggle between the **Menu** and **Keyboard** modes of command entry. If KEYB is displayed, program command creation is accomplished by direct entry of machine tool code with an IBM AT compatible keyboard. If MENU is displayed, program command creation is accomplished by selection of appropriate menu options.

(THIS PAGE INTENTIONALLY LEFT BLANK)

CHAPTER 7: THE SETUP MODE

SECTION 7 -1: INTRODUCTION

System parameters for the Unidex 400 are viewed or altered through the Setup Mode. Default operating parameters are factory set and applicable for most program applications. Prior to entering the Setup Mode, the User should have established a thorough understanding of Unidex 400 operation.

System parameters are stored in battery backed RAM. Default parameter settings are stored in ROM. In the event of a baterry failure the Unidex 400 will prompt the User to set all parameters to the default (preset) values. Default values are "standard" settings" . Prior to shipment, the Unidex 400 is configured with non-standard settings per Customer requirements. Upon receipt of the Unidex 400 it is strongly suggested the system parameters be saved to a file. Refer to Section 7-3 and 7-4 for details concerning saving and loading parameter settings.

NOTE: **To facilitate re-configuration in the event of a RAM failure, upon receiving the Unidex 400, system parameters should be saved to a file (refer to Section 7-3 and 7-4).**

SECTION 7-2: GETTING STARTED

Following Power-Up, the Initial Selection screen shown below will be displayed:

```
UNIDEX 400                          Version xxx

  RAM:ok    ROM:ok    SETUP:ok
  User RAM: xxxxxx

edit    file    setup    test    machine    PSWD
```

Enter the Setup Mode by selecting **setup** (F3), the display will be:

```
Set Up Mode



save      load      axes      general      tune      quit
```

SECTION 7-3: SAVING SETUP DATA

Select **save** to write the current setup data to a file. The display will be:

```
Set Up Mode
Save Setup Data to File:



ins-Y/N      del                                    quit
```

Key in the File number that is to be used to store the setup data settings. The system setup data is saved in (Hex-ASCII). Each parameter is labeled with its parameter number. This file may be edited in the standard way and need not contain all parameters.

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

quit Press the F6 key to abort the command and return to the initial selection menu.

Press the **ENTER** key to accept the file number

The message **DONE** will appear on the display when the setup data has been saved to the file.

SECTION 7-4: LOADING SETUP DATA FROM A FILE

Select **load** to load a previously saved setup file. The current parameters are overwritten by the loaded values. The only requirement of the file to be loaded is that it be in the same format as the file generated by the Setup Modes **SAVE** command.

The display will be:

```
Set Up Mode
Load File to Setup Data:



ins-Y/N        del                              quit
```

Key in a File number from which to load the setup data.

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

quit Press the F6 key to abort the command and return to the initial selection menu.

Press the **ENTER** key to accept the file number

The message **DONE** will appear on the display when the parameter data has been loaded. Reset the Unidex 400 to initialize the new parameter values.

NOTE: **Following loading of a setup file, the Unidex 400 must be Reset to initialize the values.**

SECTION 7-5: SETTING AXIS PARAMETERS

Select **axes** to view and/or alter the axis parameter settings.

The following is the initial axis parameter display:

```
x00 : METRIC SCALE FACTOR
x01 : ENGLISH SCALE FACTOR
x02 : HOME DIRECTION IS CCW ?                          0/1
 x= 1,2,3,4 for Axis X,Y,Z,U
item:                       data:
ins-Y/N     del     item             preset     quit     m
```

The following selections are available for viewing or altering axis parameter settings:

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

item Select **item** to move the cursor to the location on the display where an axis parameter number may be entered. The axis parameter item number is entered as follows:

Key in 1,2,3 or 4 to designate Axis X,Y,Z, or U.

Key in the two digits that designate the parameter number.

For example:

Key in **100** to set the **Metric Scale Factor** for the X axis.
Key in **200** to set the **Metric Scale Factor** for the Y axis.

Press **ENTER** to accept the item number and to move the cursor to the **data** location on the display.

The current parameter setting will be displayed.

Key in the new parameter setting and then press **ENTER** to accept the new setting.

> **NOTE:** **Following the setting of a parameter, the Unidex 400 must be Reset to initialize the values.**

If a parameter setting is entered that is not appropriate for that parameter, the message "**data out of bounds**" will appear. The item number and data must then be re-entered for that parameter.

preset Select **preset** to display the default value of the parameter setting. Press **ENTER** to accept the default setting as the parameter value or select **Item** to return to Item field without saving the default value.

quit Press the F6 key to abort the command and return to the initial setup menu.

Press the "down" arrow for additional parameters. Press the "Left" and "Right" arrows to move the cursor in these directions.

The following section provides an explanation of each of the axis parameters.

SECTION 7-6: AXIS PARAMETERS

x00 : METRIC SCALE FACTOR

The Metric Scale Factor is used to convert programmed (mm) units to machine units (steps). This scale factor is used when the **G71** command is encountered in a program or the system default is the Metric mode. The number selected is incorporated into the following calculation:

$$\text{scale factor} = \frac{\text{machine steps}}{\text{prog steps}} * \frac{1000000}{1\text{x}10^{\text{NDEC}}}$$

The NDEC value is the number of decimal places implied in program units. It is set by general parameter 4.

The setting range is 0-99999999. The default setting is 1000000.

x01 : ENGLISH SCALE FACTOR

The English Scale Factor is used to convert programmed (inches) units to machine units (steps). This scale factor is used when the **G70** command is encountered in a program or the system default is the English mode. The number selected is incorporated into the following calculation:

$$\text{scale factor} = \frac{\text{machine steps}}{\text{prog steps}} * \frac{1000000}{1\text{x}10^{\text{NDEC}}}$$

The NDEC value is the number of decimal places implied in program units. It is set by general parameter 4.

The setting range is 0-99999999. The default setting is 2540000.

The following is an example of an English Scale Factor: (The procedures for Metric Scale Factor setting is theoretically the same.)

Scale Factor Example:

This example determines the Scale Factor for the English Unit Mode such that:

1.) Programmed Units = Inches
2.) Resolution of .1 mil, smallest move = .0001 inches
3.) System has 500 Line Encoders and a 5 rev/inch Ball Screw.

Solution:

Determine Machine Steps/Program Steps (inches)

500 Line x 4 (quadrature) x 5 (pitch)

$$\frac{10000 \text{ machine steps}}{\text{inch}}$$

The NDEC is selected so that the maximum programmed resolution is .0001 inches = $1\text{x}10^{-4}$, therefore NDEC = 4.

$$\text{Scale Factor} = 10000 \times \frac{1000000}{1 \times 10^4} = 1{,}000{,}000$$

Enter this number into axis parameter x01 and Reset the system. The command G1 X1. F100 will now move the X axis one inch at a rate of 100 inches per minute.

x02 : HOME DIRECTION IS CCW?

The Home Direction sets the direction the motor must rotate to encounter the Home limit during a HOME cycle (0-CW,1-CCW).

The default setting is 1.

x03 : HOME SWITCH NORMALLY OPEN ?

This parameter must be configured to correspond to the active polarity of the Home Limit Switch being used (0-Closed, 1-Open).

The default setting is 1.

x04 : HOME FEEDRATE (STEP/SEC)

This parameter establishes the Feedrate during a runtime Home cycle.

The setting range is 0-99999999. The default is 20000 steps per second.

x05 : POWER ON HOME FEEDRATE (STEPS/SEC)

Sets the Feedrate specifically for a Home move that immediately follows a Power Up. To avoid possible equipment damage it is suggested that this Feedrate be set considerably slower than programmed Home Feedrate since axis disorientation is likely .

The setting range is 0-99999999.The default setting is 10000 steps per second.

x06 : HOME LIMIT TO MARKER OFFSET (STEPS)

The distance (in machine steps) that the axis may move at maximum speed before slowing to search for a Marker. Selection of this distance is dependent upon system configuration and hardware variances.

The setting range is 0-99999999. The default setting is 1000 machine steps.

x07 : HOME OFFSET (STEPS)

This parameter establishes offsets to provide for a Home position that is not at a Marker. The distance from the Home Marker to the desired Home position must be measured and then converted to machine steps.

The setting range is 0-99999999. The default setting is zero (Home position is located at the Marker).

x08 : TOP FEEDRATE (STEPS/SEC)

The Top Feedrate is the highest speed for which the axis is mechanically configured. It is used as a basis for the Jog Feedrate. This parameter also establishes the maximum Feedrate that may be programmed before a "Top Feedrate Trap" occurs.

The setting range is 0-1966020. The default setting is 128000 steps per second.

x09 : MAX ACCEL/DECEL (STEPS/SEC/SEC)

This parameter is used to set the maximum acceleration/deceleration used during a freerun or a Home cycle for ramping.

The setting range is 0-99999999. The default setting is 1000000 steps per second per second.

x10 : PROPORTIONAL GAIN Kp

The Proportional Gain parameter sets the "P" Gain in the PIDF1F2 servo loop. The Proportional Gain produces an output directly proportional to the position error and thus produces a constant counteracting force to the error. Prior to changing this parameter, refer to the Auto-tune section of this chapter (Section 7-8).

The setting range is 0-8388008. The default setting is 29000.

x11 : INTEGRAL GAIN Ki

The Integral Gain parameter sets the "I" gain value in the PIDF1F2 Servo Loop. The Integral Gain produces an output which is a summation of the position errors, producing an increasing counteracting force for a constant or increasing position error. Prior to changing this parameter, refer to the Auto-tune section of this chapter (Section 7-8).

The setting range is 0-8388008. The default setting is 2000.

x12 : DERIVATIVE GAIN Kd

The Derivative Gain parameter sets the "D" gain value in the PIDF1F2 Servo Loop. The Derivative Gain dampens system response by producing a damping effect as long as the system is progressing toward error reduction. Prior to changing this parameter, refer to the Auto-Tune section of this chapter (Section 7-8).

The setting range is 0-8388008. The default setting is 276000.

x13 : VELOCITY FEEDFORWARD Kf1

This parameter permits the entry of a Velocity Feedforward value. The Velocity Feedforward optimizes the Gain settings of the PID Servo loop. The Velocity Feedforward value is a function of system inertia and the torque constant. Since it is not strictly error driven, it complements the PID Gain by reducing the PID compensation effort required, thus reducing system lag. Prior to changing this parameter, refer to Section 7-8 for further information concerning this setting.

The setting range is 0-8388008. The default setting is 0 (Velocity Feedforward disabled).

x14 : ACCELERATION FEEDFORWARD Kf2

T he Acceleration Feedforward optimizes the Gain settings of the PID Servo loop. The Acceleration Feedforward value is a function of system inertia and the torque constant. Since it is not strictly error driven, it complements the PID Gain by reducing the PID compensation effort required thus reducing system lag. Prior to changing this parameter, refer to Section 7-8 for further information concerning this setting.

The setting range is 0-8388008. The default setting is 0 (Acceleration Feedforward disabled).

x15 : FEEDHOLD IN FREERUN ? 0/1

This parameter determines whether the Front Panel's **FEEDHOLD** Key affects Freerun motion for this axis. (0-No, 1-Yes)

The default setting is 1 (the **FEEDHOLD** Key may be used to control Freerun motion).

x16 : ENABLE MFO IN FREERUN ? 0/1

This parameter allows an external MFO (Manual Feedrate Override) potentiometer to affect Freerun speed. (0-No, 1-Yes)

The default setting is 1 (an external MFO potentiometer will be active during a Freerun).

x17 : AXIS CALIBRATION ON ? 0/1

The Unidex 400 is capable of performing axis calibration, if valid calibration data is present. The appropriate data must be contained in the assigned files 9902, 9903, 9904 and 9905 for Axes X, Y, Z and U respectively. Refer to Section 7-8 for axis calibration information. (0-No, 1-Yes)

The default setting is 0 (No axis calibration).

x18 : JOYSTICK ABS MODE SCALE FACTOR

This parameter is used to establish a "window" of axis movement for fine positioning when in the Joystick's Absolute Mode.

The setting range is 0-255. The default setting is 10, providing Joystick positioning in increments of 10 program steps.

x19 : JOYSTICK HIGH VELOCITY

Resolution ratio's vary between axes. When a Joystick move is requested, the speed/distance ratio for each affected axis must be compatible. This parameter establishes the speed at which the axis will move when a Joystick move is requested at the "High" speed setting. To derive the Joystick "High" speed setting for this axis, proceed as follows:

Determine the distance (mm or in.) that the axis will move per second at the maximum speed.

Multiply this value by the axis resolution ratio (number of machine steps that equal 1 mm or in.).

The resulting number (steps/sec) is the Joystick-High value that may be entered for this parameter.

The setting range is 0-99999999. The default setting is 50000 steps/second.

x20 : JOYSTICK LOW VELOCITY (STEPS/SEC)

Resolution ratio's vary between axes. When a Joystick move is requested, the speed/distance ratio for each affected axis must be compatible. This parameter establishes the speed at which the axis will move when a Joystick move is requested at the "Low" speed setting. To derive the Joystick "Low" speed setting for this axis, proceed as follows:

Determine the distance (mm or in.) that the axis will move per second at the low speed.

Multiply this value by the axis resolution ratio (number of machine steps that equal 1 mm or in.).

The resulting number (steps/sec) is the Joystick-Low value that may be entered for this parameter.

The setting range is 0-99999999. The default setting is 10000 steps/second.

x21 : START STOP SPEED (STEP/SEC)

The Start/Stop Speed parameter establishes the maximum speed that this axis may be started or stopped without initiating an error condition.

The default setting is 2000 steps/sec. This parameter setting should not be changed by the User.

x22 : DISPLAY MODULO (STEP)

This parameter is used to establish the machine step number that will signal the Unidex 400 to return it's axis display counters to zero, and begin another count of steps.

The default setting is zero (no counter reset).

x23 : LIMIT SWITCH NORMALLY OPEN ? 0/1

This parameter must be configured to reflect the type of CW and CCW Limit switches being used, normally open or normally closed. (0-No, 1-Yes)

The default setting is 1, the Limit Switch for this axis is a normally open type Limit Switch.

x24 : LIMIT TO MECHANICAL STOP (STEP)

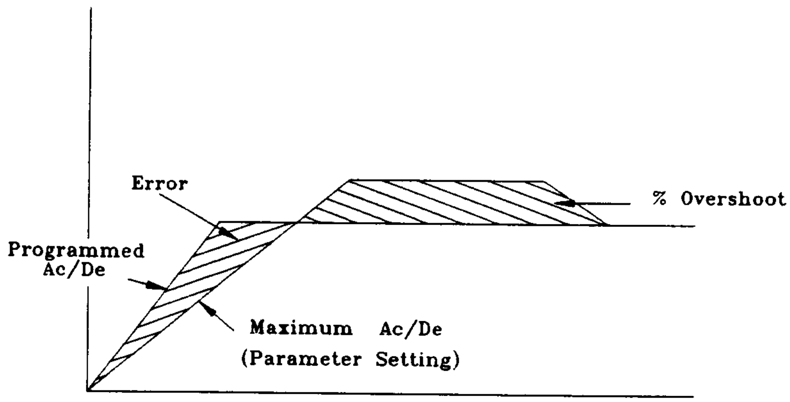
This parameter sets the number of machine steps that will separate the Limit Switch from the Mechanical Stop.

The setting range is 0-99999999. The default setting is 1000 machine steps.

x25 : MAXIMUM OVERSHOOT %

The maximum amount of axis overshoot must be entered for this parameter, 127 = 100%. The Unidex 400 uses this parameter to determine the amount of overshoot permitted when dissipating errors that result from an acceleration at a rate greater than the parameter setting of the Acceleration/Deceleration rate (see Parameter x09).

The setting range is 0-127. The default setting is 10.



x26 : MAXIMUM VELOCITY ERROR

This parameter is used to input the maximum amount of velocity error (the difference between the actual velocity and programmed velocity). The Unidex 400 will display a "Velocity Trap" error message if this parameter setting is exceeded during operation.

The setting range is 0-65535. The default setting is 1000.

x27 : MAXIMUM POSITION ERROR

This parameter is used to establish the maximum amount of position error (the difference between the actual and the requested position). The Unidex 400 will display a "Position Trap" error message if this parameter setting is exceeded during operation.

The setting range is 0-65535. The default setting is 1000.

X28 : MAXIMUM INTEGRAL ERROR (x100)

This parameter is used to input the maximum amount of allowable integrated error. The Unidex 400 will display a Integral error message if this parameter setting is exceeded during operation. The desired integral error value must be divided by 100 before it is entered, for example; if an error value of 200,000 is desired, enter 2000.

The setting range is 0-65535/100. The default setting is 1000.

x29 : TRAJECTORY TYPE IS LIN/PAR 0/1

The Acceleration/Deceleration ramping trajectory may be set as Linear or Parabolic. The Parabolic Mode uses a parabolic coefficient to round the edges from the velocity profile.

The default setting is 0 (Linear Trajectory).

x30 : PARABOLIC TRAJECTORY COEFFICIENT < 65535

The Parabolic Coefficient is the factor used to convert the linear velocity/time slope into a Parabolic Trajectory. It is suggested that coefficient selections be made in steps of 10,000. The following formula is applied to the coefficient value selected:

$$\text{maximum slope} = \frac{\text{Parabolic Coefficient}}{65535} + 1$$

The setting range is 0-65535. The default setting is 24576.

x31 : AUTOFOCUS 1 - SCALE FACTOR x65535

This parameter provides the ability to establish the value of the conversion factor used in conjunction with the Autofocus command (G96). This conversion factor is used to convert Input 1 Clock to Machine Steps. The following formula is applied:

$$\text{machine steps} = \text{Input Clock} * \frac{n}{65535}$$

Input 1 is also used as Handwheel Input 1.

The setting range is 0-8388608. The default setting is 65535 (1 to 1 scale).

x32 : AUTOFOCUS 2 - SCALE FACTOR x65535

This parameter provides the ability to establish the value of the conversion factor used in conjunction with the Autofocus command (G96). This conversion factor is used to convert Input 2 Clock to Machine Steps. The following formula is applied:

$$\text{machine steps} = \text{Input Clock} * \frac{n}{65535}$$

Input 2 is also used as Handwheel Input 2.

The setting range is 0-8388608. The default setting is 65535 (1 to 1 scale).

x33 : FEEDBACK DEVICE TYPE 0-3

This parameter is used to specify the type of device that the Feedback Servo Loop is to use. The options are; 0-Encoder, 1-Resolver, 2-Resolution Multiplier Board, 3- Open Loop (no feedback).

The default setting is 0, Encoder feedback.

x34 : DRIVE TYPE 0-3

This parameter configures the Unidex 400 for the type of amplifier/motor combination that is used on this axis. The options are 0-Brush type, 1-AC Brushless type, 2-Stepper type, 3-reserved.

The default setting is 0, Brush type motor.

x35 : BRUSHLESS COMMUTATION FACTOR

This parameter is valid only when a brushless type motor is being used. The commutation factor is derived as follows:

> If the feedback device is an encoder or a RMX Board, the commutation factor is the number of machine steps per electrical cycle.
>
> If the axis position feedback device is a resolver, the commutation factor is the number of electrical cycles per revolution.

The default setting is 2 indicating a brushless motor with resolver feedback.

x36 : FIXED BACKLASH COMPENSATION 0-OFF (STEP)

This parameter provides greater positioning accuracy by allowing entry of the number of machine steps (0-65535) needed to compensate for any backlash that may be present in the system with an axis direction change. The value of this parameter is added to the new direction.

The default setting is zero indicating backlash compensation is OFF.

x37 : EXISTENCE CHECKING 0-3

This parameter establishes active axis status conditions that are monitored by the Unidex 400 upon power-up or reset. The use of this parameter provides added awareness of the current configuration of the hardware. The following options are available for this parameter:

0-the status of the axis is determined internally by the Servo.
1-this axis must be active or an error message is displayed
2-this axis is not active, if detected as active, an error message is displayed.
3-axis is assumed as present but existence is not checked, disables auto-tune.

The default setting is 0, axis status determined internally by the Servo.

x38 : CCW SOFTWARE LIMIT (STEP)

This parameter is used to establish a CCW software travel limit (in machine steps) that is in reference to the hardware home.

The setting range is 0-99999999. The default setting is zero, indicating that a CCW software limit has not been established.

x39 : CW SOFTWARE LIMIT (STEP)

This parameter is used to establish a CW software travel limit (in machine steps) that is in reference to the hardware home.

The setting range is 0-99999999. The default setting is zero, indicating that a CW software limit has not been established.

x40 : SIZE FOR STEP RESPONSE (STEP)

This parameter sets the optimum step size to produce the desired step response. Typically the step size is 1/4 of the motor's revolution, i.e., a step size of 500 steps would be used for a 2000 count encoder.

The setting range is 0-99999999. The default setting is 500 steps.

x41 : SECOND FEEDBACK THRU AXIS #1-4,0

This parameter sets the type of Servo Loop that will be used by this axis. The options for this parameter are:

0 The standard Servo Feedback Loop using single feedback will be in effect. See Figure 7-1.

1-4 A Dual Servo Feedback Loop will be used, with the velocity feedback coming from the designated channel. See Figure 7-2.

Example:

If this parameter is activated by entry of 141 and 1 is entered as the parameter setting, both the position and velocity feedback will come from the X axis.

The default setting is zero, designating that the standard Servo Feedback Loop will be used.

x42 : CURRENT COMMAND SCALE 1-65535

This parameter establishes the value that will be used as the current command scale factor (K) in the PID Servo Feedback Loop. (Refer to Figures 7-1 and 7-2 and the PID/G27 command.)

The setting range is 1-65535. The default setting is 1.

SECTION 7-7: SETTING GENERAL PARAMETERS

Following entry into the Setup Mode, the display will be:

```
Set Up Mode



save    load    axes    general    tune    quit
```

Select **general** to view and/or alter the general parameter settings.

The following is the initial general parameter display:

```
000 : PASSWORD
001 : SERIAL AUTO ECHO  ?              0/1
002 : DEFAULT SF. ENGLISH/METRIC       0/1
003 : RS232 SETUP
item:                          data:
ins-Y/N    del    item        preset    quit    m
```

The following selections are available for viewing or altering general parameter settings:

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

item Select **item** to move the cursor to the location on the display that a general parameter number may be entered.

Press **ENTER** to accept the item number and to move the cursor to the **data** location on the display.

The current parameter setting will be displayed.

Key in the new parameter setting and then press **ENTER** to accept the new setting. This setting will remain in effect until overwritten by a new setting or the default value is recalled by use of the **preset** selection.

> NOTE: **Following the setting of a parameter, the Unidex 400 must be Reset to initialize the values.**

If a parameter setting is entered that is not appropriate for that parameter, the message "**data out of bounds**" will appear. The data must then be re-entered for that parameter.

preset Select **preset** to display the default value of the parameter setting. Press **ENTER** to accept the default setting as the parameter value.

quit Press the F6 key to abort the command and return to the initial setup menu.

Press the "down" arrow for additional parameters. Press the "Left: and "Right" arrows to move the cursor in these directions.

The following section provides an explanation of each of the general parameters.

SECTION 7-8: GENERAL PARAMETERS

000 : PASSWORD

Key in **0** to enter or change a system password. Passwords may be configured with different privledges in order to enable or restrict certain Modes for certain User's. Once established, a password must be entered following each power-up.

NOTE: **The password need only be entered following a power-up. If it is desired to change the password currently being recognized, a different password may be entered by returning to the initial selection menu.**

Following the **0** entry, the display will be:

```
        PASSWORD NUMBER [1]
                                    EFMST
    PRIVILEGE LEVEL                 [00000]
    PASSWORD:

      add      clear     word     priv      next      quit
```

add Select **add** to create a new password. Key in the password (number). Up to ten (10) passwords may be entered.

clear Select **clear** to remove the current password from memory and re-number the remaining passwords.

word Select **word** to move the cursor to the password location of the display.

priv Select **priv** to establish the modes for which this password will have privilege. The letters **EFMST** designate the privilege mode of the password. They correspond to the five major operating modes of the Unidex 400; Edit, File, Machine, Setup, Test. A password is privileged in a mode if the corresponding mode bit is set to "1". The cursor is moved from mode to mode by use of the left and right arrow keys.

NOTE: **In order to enter this password display, the User must be privileged in the Setup field (S).**

Special Password, S_1 S_2 S_3 3 1 5 8 9

next Select **next** to step through and display all established passwords.

Press **ENTER** to write the password to memory. The password mode will be active following a subsequent power-down/power-up.

quit Select **quit** to return to the main parameter menu.

NOTE: **If "quit" is selected before the password is entered, the password will not be active.**

001 : SERIAL AUTO ECHO ? 0/1

This parameter is used to configure the Serial Port to automatically echo any character it receives. (0-No character echo, 1-Character echo).

The default setting is 1.

002 : DEFAULT SF. ENGLISH/METRIC 0/1

This parameter is used to select whether the English or Metric Mode scale factor is currently active. This scale factor will be in effect upon power-up until the mode is cancelled by G70/G71. The scale factor is established by Axis Parameters x00 or x01.

Entering 0 sets the English Mode (G70) on power-up, entering 1 sets the Metric Mode (G71) on power-up.

The default setting is 1.

003 : RS232 SETUP

This parameter is used to configure the RS-232 serial port settings. Following selection of the RS232 Setup Parameter, the display will be:

```
Number of Data Bits          [8 ]
Number of Stop Bits          [1]
Parity                       [N]
Baud Rate                    [9600]

Data    Stop    Parity    Baud    Preset    Quit
```

Data Select **Data** to toggle through the options available for the number of data bits to be used. The available settings are; 5, 6, 7, or 8 character bytes. The default setting is 8.

Stop Select **Stop** to toggle through the stop bit settings. The selections are 1 or 2 (1.5 is allowed for the "5" Data Bit setting). The default setting is 1.

Parity Select **Parity** to toggle through the parity setting selections. The available settings are None, Odd or Even. The default setting is "N".

Baud Select **Baud** to toggle between the available baud rate settings. The available settings are 75, 110, 134, 150, 300, 600, 1200, 2400, 4800, 7200, 9600, and 38400. The default setting is 9600.

preset Select **preset** to display and store all of the default values for the RS-232 serial port setting.

quit Press the F6 key to return to the general parameter display.

004 : DIGITS AFTER DECIMAL POINT

This parameter is used to specify the number of decimal places implied in motion commands. This parameter specifies the resolution of the program units. For example, if the setting is 4, the programming resolution is .0001 User units.

This number is used to determine the English and Metric scale factors, Axis Parameters x00 and x01.

The setting range is 0-7. The default setting is 4.

005 : AXES RAMP TIME (ms)

This parameter is used to set the time, in mSec, that all axes are to be at their final or commanded velocity.

The setting range is 0-32767. The default setting is 100 mSec.

006 : END OF ONE FILE CHARACTER (EOF)

This parameter establishes a number that is transmitted or received, through Serial Port A as a code to indicate that the transfer of one file has been completed.

The setting range is 0-127. The default setting is 9.

007 : END OF ALL FILE CHARACTER (EAF)

This parameter establishes a number that is transmitted or received, through Serial Port A as a code to indicate that the transfer of all files has been completed.

The setting range is 0-127. The default setting is 17.

008 : JOYSTICK PAIR

This parameter is used to setup combinations of axis pairs. Following setup, the pairs may be toggled through by use of the upper left joystick button. Eight sets of joystick pairs may be specified.

Following selection of the Joystick Pair parameter, the display will be:

```
JOYSTICK SET UP NUMBER - 1

Horizontal                        Vertical
XYZU = [1000]                     XYZU = [0100]


   horiz     vert     number                 preset     Quit
```

horiz Select **horiz** to move the cursor to the **Horizontal** field on the display, key in a "1" for the axis that is to be the horizontal axis for this axis pair (only one axis may be set to 1). The left and right arrow keys are used to move the cursor between the axes.

vert Select **vert** to move the cursor to the **Vertical** field on the display, key in a "1" for the axis that is to be the vertical axis for this axis pair (only one axis may be set to 1). The left and right arrow keys are used to move the cursor between the axes.

number Select **number** to step through all axis pair settings (Joystick Setup Number 1-8).

preset Select **preset** to display the default values for the Joystick pair number displayed.

quit Press the F6 key to save the displayed settings and return to the general parameter display.

009 : DISPLAY IN MACH/PROG UNITS 0/1

This parameter sets the Unidex 400 to display units in "Program" (scaled) units or in direct machine units.

Entering 0 sets the display in machine units, entering 1 sets the display in program units.

The default setting is 1.

010 : PARTS PROGRAM STACK SIZE

This parameter is used to determine the number of bytes RAM allocated for program stacks.

The setting range is 0-2097151. The default setting is 320 bytes.

011 : AXES FOR GANTRY

The Gantry parameter establishes the master/slave relationship for multiple axes motors that are used in pairs. A standardly configured system may have two pairs. The format for configuring the pairs is as follows:

	msms	m = master s = slave 1234 = XYZU
Example:		
	1342	X and Z axes are paired, X is the master, Z is the slave
		U and Y axes are paired, U is the master, Y is the slave

The setting range is 0-87654321 though the standard Unidex 400 uses only axes 1-4.

The default setting is 0.

012 : M STROBE DELAY (ms)

This parameter is used to specify the length of time that the system output signal stays ON to allow a slow device time to respond.

The setting range is 0-65535. The default setting is 0, no M strobe output.

013 : M ACKNOWLEDGE DELAY (ms)

The MST "M" Acknowledge input parameter is used to specify the length of time that the system will wait for an Acknowledge signal. A "0" setting indicates no Acknowledge is required. A setting of 65535 will cause the Unidex 400 to scan every mSec, indefinitely.

The setting range is 0-65535. The default setting is 0, no Acknowledge required.

014 : S STROBE DELAY (ms)

This parameter is used to specify the length of time that the system output signal stays ON to allow a slow device time to respond.

The setting range is 0-65535. The default setting is 0, no S strobe output.

015 : S ACKNOWLEDGE DELAY (ms)

The MST "S" Acknowledge input parameter is used to specify the length of time that the system will wait for an Acknowledge signal. A "0" setting indicates no Acknowledge is required. A setting of 65535 will cause the Unidex 400 to scan every mSec, indefinitely.

The setting range is 0-65535. The default setting is 0, no Acknowledge required.

016 : T STROBE DELAY (ms)

This parameter is used to specify the length of time that the system output signal stays ON to allow a slow device time to respond.

The setting range is 0-65535. The default setting is 0, no T strobe output.

017 : T ACKNOWLEDGE DELAY (ms)

The MST "T" Acknowledge input parameter is used to specify the length of time that the system will wait for an Acknowledge signal. A "0" setting indicates no Acknowledge is required. A setting of 65535 will cause the Unidex 400 to scan every mSec, indefinitely.

The setting range is 0-65535. The default setting is 0, no Acknowledge required.

018 : WHEEL 1 TYPE 0-3

This parameter is used to configure the input data format of the RS-422 Wheel 1 connector, to be in conjunction with the signal requirements of the input device.

The selections are:

0 - CW/CCW pulses
1 - CLK/DIR
2 - QUADRATURE x 1
3 - QUADRATURE x 2

The default setting is 2, for a Clock and Direction type input.

019 : WHEEL 2 TYPE 0-3

This parameter is used to configure the input data format of the RS-422 Wheel 2 connector, to be in conjunction with the signal requirements of the input device.

The selections are:

0 - CW/CCW pulses
1 - CLK/DIR
2 - QUADRATURE x 1
3 - QUADRATURE x 2

The default setting is 2, for a Clock and Direction type input.

020 : WHEEL 1 SCALE FACTOR

This parameter is used to establish a relationship between handwheel increments and machine steps of axes movement for Wheel 1 input.

machine steps = scale factor * wheel steps

The setting range is 0-255. The default setting is 10.

021 : WHEEL 2 SCALE FACTOR

This parameter is used to establish a relationship between handwheel increments/decrement setting and machine steps of axes movement for Wheel 2 input.

machine steps = scale factor * wheel steps

The setting range is 0-255. The default setting is 10.

022 : MFO ADJUSTS HANDWHEEL ? 0\1

This parameter is used to enable the MFO setting (established by the **G5/UMFO** command or the external MFO Potentiometer) to provide the Handwheel scale factor. Enabling this function provides the ability to change the Handwheel increment/machine step relationship without entering the parameter mode.

Example:

100 handwheel counts * 50% MFO setting results in 50 counts

0-No the MFO setting will not provide the Handwheel scale factor, 1-the MFO setting will provide the Handwheel scale factor.

The default setting is 1.

023 : POT MFO 0 OFFSET 0-255

This parameter enables or disables an external MFO and sets the lower level noise threshold.

If this parameter is set to "0" the Unidex 400 will not recognize an external MFO potentiometer. If this setting is a non-zero number, the Unidex 400 will "look" for an external MFO and use the value as the minimum MFO %

NOTE: **If signal "jitter" is evident, it is suggested that the value of this parameter be increased.**

The setting range is 0-255. The default setting is 0.

024 : PERPENDICULAR AXIS PAIR

This parameter identifies to the Unidex 400 the horizontal and vertical relationship of axes pairs. The pair identification is used for perpendicularity error compensation (see general parameter 26 and 27. Four axes may be identified for a total of two pairs.

The format for configuring the pairs is as follows:

hvhv	h = horizontal	v = vertical	1234 = XYZU
Example:			
1342	X and Z axes are paired, X is horizontal, Z is vertical		
	U and Y axes are paired, U is horizontal, Y is vertical		

The default setting is 0 for no perpendicularity error compensation.

025 : RESET KEEPS POSITION ? 0/1

This parameter allows the Unidex 400 to be configured such that Servo positions are not lost after a software reset. (Positioning continues to be lost following a power-down.) Entering "0" disables position retention after reset, entering "1" enables position retention after reset.

The default is 1, the Servo position is retained following a reset.

026 : 1ST PERPENDICULAR PAIR ERROR (ARC SEC)

Positioning accuracy is directly related to the perpendicularity of the paired axis (refer to General Parameter 24). This parameter permits the amount of error, calculated for the first axis pair in arc seconds, to be entered into the system. Perpendicularity error is defined as the positive or negative deviation of the two paired stages from 90 degrees. An error in the CW direction (less than 90 degrees) is entered as a negative value. An error in the CCW direction (more than 90 degrees) is entered as a positive value.

The setting range is 0-324000. The default setting is 0.

027 : 2ND PERPENDICULAR PAIR ERROR (ARC SEC)

Positioning accuracy is directly related to the perpendicularity of the paired axis (refer to General Parameter 24). This parameter permits the amount of error, calculated for the second axis pair in arc seconds, to be entered into the system. Perpendicularity error is defined as the positive or negative deviation of the two paired stages from 90 degrees. An error in the CW (less than 90 degrees) direction is entered as a negative value. An error in the CCW direction (more than 90 degrees) is entered as a positive value.

The setting range is 0-324000. The default setting is 0.

028 : ADDITIONAL USER RAM (KBYTES)

Additional RAM may be added to the Unidex 400 VME bus. This parameter is used to configure the Unidex 400 for the amount of RAM to be added. A maximum of 8 MB RAM may be added.

The setting range is 0-8193 Kilo Bytes. The default setting is 0, the Unidex 400 will not "look" for any additional RAM.

029 : AXIS TRAP, OUTPUT LINE LOW (0-8)

This parameter is used to designate the output line that will go low in the event of a Servo error. This signal may then be used by an auxiliary device to initiate some function.

NOTE: **If the CPU Bus is configured for MST, the "T" strobe is asserted along with the output bit line specified.**

The default setting is 0, no output line is driven low upon an axis trap.

030 : QUICK STOP HI-LO TRIGGER TYPE 0/1

A Quick Stop or Fast Feedhold is an input signal used to feedhold all axes of the Unidex 400 as quickly as possible. A Quick Stop may be initiated by either a High/Low or a Low/High trigger signal. This parameter establishes the active polarity of the trigger signal. Entering "0" initiates a Quick Stop on a Low/High trigger signal, entering a "1" initiates a Quick Stop on a High/Low trigger signal.

The default setting is 1, a Quick Stop will occur on a High/Low trigger signal.

031 : QUICK STOP RETURN TO TRIGGER POINT 0/1

When a standard Quick Stop is initiated, deceleration occurs and axis movement halts at the conclusion of the Quick Stop Ramp Time (as established by General Parameter 036). If this parameter is activated (1) then the Unidex 400 moves the axis back to the point at which the Quick Stop trigger occurred, thus eliminating the ramp time distance.

The default setting is 1, the axes will return to the point at which the Quick Stop trigger occurred.

032 : IEEE 488 SETUP (GPIB)

This parameter is used to configure a variety of IEEE-488 settings. Following selection of this parameter, the display will be:

```
0 : GPIB ADDRESSING MODE
     - 0  talk only        - 1 listen only
     - 2 talker/listener
1 : DEVICE ADDRESS  (0-31)
item : _                   data :
 ins-Y/N     del       item       data       preset       quit    m
```

The following selections are available for viewing or altering IEEE-488 settings:

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

item Select **item** to move the cursor to the location on the display that a general parameter number may be entered.

Press **ENTER** to accept the item number and to move the cursor to the **data** location on the display.

The current setting will be displayed.

Key in the new parameter setting and then press **ENTER** to accept the new setting. This setting will remain in effect until overwritten by a new setting or the default value is recalled by use of the **preset** selection.

NOTE: **Following the configuration of IEEE-488 , the Unidex 400 must be Reset to initialize the values.**

preset Select **preset** to display the default value of the parameter setting. Press **ENTER** to accept the default setting as the parameter value.

quit Press the F6 key to abort the command and return to the initial setup menu.

Press the "down" arrow for additional IEEE-488 settings. Press the "Left: and "Right" arrows to move the cursor in these directions.

The following section provides an explanation of each of the IEEE-488-488 settings.

032 : 0 : GPIB ADDRESSING MODE

This setting is used to establish the addressing mode of the Unidex 400 on the IEEE-488 Bus. The address may be configured as one of the following:

0 - talk only the Unidex 400 should be configured as a "talker" when it is needed only to send data.

1 - listener the Unidex 400 should be configured as a "listener" when it is needed only to receive data.

2 - talker/listener the Unidex 400 will respond to the GPIB Controller, both sending and receiving data.

The default setting is 2, the Unidex 400 is both a "Listener" and a "Talker".

032 : 1 : DEVICE ADDRESS (0-31)

This setting is used to configure the address of the Unidex 400 on the Bus. Care should be taken that the address allocated to the Unidex 400 by this parameter is not already set for another device.

The setting range is 0-31. The default setting is 2.

032 : 2 : PARALLEL POLL RESPONSE (BIT)

This parameter determines the bit number that the Unidex 400 will set when the Controller does a parallel poll of the devices on the Bus. The parallel poll response may be disabled by setting this parameter to zero. If this parameter is set between 9 and 16 the Unidex 400 will respond with a reverse polarity parallel poll.

Example:

If set to 1 : Unidex 400 will set DI1 to a logic high during a parallel poll.

If set to 9 : Unidex 400 will set DI1 to a logic low during a parallel poll.

If set to 0 : Unidex 400 will not drive any line during a parallel poll.

The setting range is 0-16. The default setting is 1.

032 : 3 : EOS CHARACTER (0-255)

The End of String character indicates the end of a block of data. This parameter is typically set to the line-feed character (10 decimal). The Unidex 400 will send the EOS character when returning strings of data.

The Unidex 400 does not use the EOS character to stop data during the file transfer mode since the line feed character is used to delimit program lines. During file transfer, the Unidex 400 uses only the User specified End of File (EOF) and End All File (EAF) characters to separate files and terminate transfer.

The EOS character may be any value between 0 and 255. This also determines the size of the EOS character, 7 or 8 bits.

The Unidex 400 expects a <CR> (13 dec.) as a terminating character for the data received. (Refer to general parameter 006).

The setting range is 0-255. The default setting is 10.

032 : 4 : TERMINATE WRITE EOS-0/EOI-1 0/1

The Unidex 400 will send either the End of String (EOS) character or send an End of Item (EOI) character along with the EOS character to terminate data transfer. During string input to the Unidex 400, only <CR> may be used to terminate the string. String input occurs during use of Remote mode "2" as a command string or as a message using the **G17** or **G19** commands.

The default setting is 0, only the EOS character will terminate data transfer.

The following display is provided for information only:

```
Command  Input:  U400 will wait for Line
                  Feed.
File Transfer:   U400 will use EOF/EAF
                 only. See Gen Par   6,7
item : _                   data :
 ins-Y/N      del        item              preset        quit      m
```

033 : EDIT MODE BUFFER SIZE

This parameter specifies the number of bytes of memory to allocate for the Edit Mode buffer. The Edit Mode buffer is used to store program blocks that are copied or erased while in the Edit Mode. One complete line of the display is 40 bytes. If the Paste or Erase function does not operate while in the Edit Mode, this parameter setting must be increased.

The setting range is 0-99999999. The default setting is 500 bytes.

034 : RS232 U400 RECEIVE FILE(s)...
RETURN CHECKSUM AFTER EAF? 0/1

This parameter enables or disables a return of a 2 byte Checksum (in binary format) following receipt of the End All File character during file input. Refer to the Chapter 5: The File Mode, for file transfer information. Entering "0" disables a checksum return, entering "1" enables a checksum return.

The default is 0, a checksum is not returned following file input.

035 : RS232 U400 TRANSMIT FILE (s)...
WAIT FOR CHECKSUM AFTER EAF ? 0/1

When this parameter is enabled, the Unidex 400 will wait for the 2 byte Checksum (in binary format) to be returned after receiving the End All File character. Refer to Chapter 5: The File Mode, for file transfer information. Entering "0" disables a checksum return, entering "1" enables a checksum return.

The default is 0, a checksum is not returned following file input.

036: QUICK STOP RAMP TIME (ms)

A Quick Stop or Fast Feedhold is an input signal used to feedhold all axes of the Unidex 400 as quickly as possible. This parameter is used to specify a deceleration ramp time to be used when a Quick Stop input signal is received. This ramp time is used only under Quick Stop conditions.

The setting range is 0-65535 ms. The default is 100 ms.

037: IDX SEG. CALCULATE BASE (1-5)

The Segment Calculation Base Rate parameter is used to establish the period of time (in milliseconds) a move is broken down into for processing by the Indexing Board. The standard rate is 1 millisecond. If motion is jerky, an increase of this parameter may be indicated. It is recommended however, that this setting be as low as possible. The available settings are as follows:

Enter: "1" for a 1ms rate
"2" for a 2 ms rate
"3" for a 4 ms rate
"4" for a 8 ms rate
"5" for a 16 ms rate

The Unidex 400 uses this parameter setting as a baseline calculation rate, i.e. when a move is requested requiring a greater calculation rate the Undex 400 automatically generates such a rate. When the move is complete, the calculation rate value returns to the value established by this parameter.

The setting range is 1-5 ms. The default is 1 ms.

038: MASTER TRAJ. TYPE IS LIN/PAR 0/1

For all contour type moves each individual axis is synchronized with a "virtual" Master axis, therefore the Acceleration/Deceleration Ramping Trajectory for each axis follows the Acceleration/Deceleration Ramping Trajectory of the Master Axis.

This parameter defines the Master trajectory as either Linear (0) or Parabolic (1). A Parabolic trajectory is derived by applying a "Parabolic Trajectoy Coefficient" to the maximum slope. The Parabolic Trajectory Coefficient is set by General Parameter 039 or the **SERVO – > COEF** command. Refer also to the **SERVO – > TRAJ** command description in Chapter 3 of this manual.

The default is 0, the MasterTrajectory is Linear.

039: PARABOLIC TRAJECTORY COEFF < 65535

The Parabolic Trajectory coefficient is the factor that is used to convert the Linear velocity/time slope into a Parabolic Trajectory. It is suggested that the coefficient selections be made in steps of 10,000. The following formula is applied to the coefficient value selected:

$$\text{Maximum Slope} = \frac{\text{Parabolic Coefficient}}{65535} + 1$$

A Coefficent of zero represents a Linear Trajectory. Greater Coefficent values result in a steeper trajectory.

The setting range is 0-65535. The default setting is 45000.

040: G23 EXP. FILTER LEVEL <0-7>

During the G23 (Corner Rounding) operation, the Unidex 400 applies an Exponential Filter on the millisecond Position Velocity Trajectory command being sent to the DSP Servo Control Card. This filter allows the Uniudex 400's PIDF Servo Loop to digitally replicate the traditional error based type of Servo Loop.

Filter Total = Filter Total + Command In
Command Out = Filter /2n; (n equals corner rounding filter #)
Filter Total = Filter Total - Command Out

Values for theis parameter range from 0 (no filter) to 7 (maximum filter). For standard systems a value of 4, 5, or 6 provides the best performance. High filter values result in more corner rounding.

The default filter level is 3.

041: RAMPING DURING G23? 0/1

The G23 (Corner Rounding) Mode is used to program moves that are not necessarily tangent such that the controller can provide relatively continuous Velocity Contouring. In most applications such as Raster scanning, it is desirable to permit the User to program a set of linear moves at right angles to each other, and have the controller round the corners. That is, as soon as the first axis begins it's deceleration ramp, the second axis is to begin it's acceleration ramp. This is acheived by setting the Ramp Time (refer to the **SERVO -> RAMP** command in Chapter 3) to one millisecond each time that the User enters the G23 (Corner Rounding) Mode and restoring a normal Ramp Time (established by General Parameter 005) when in the G24 (Non-Corner Rounding) Mode.

Some applications however, use the G23 command to blend step velocities resulting from consecutive non-tangental moves. To provide for smooth motor operation, the normal Ramp Time (established by General Parameter 005) may be used in conjunction with a low filter **(G66, MODE -> FILTER)** setting.

If this parameter is set to No (0) a Ramp Time of one (1) millisecond will be used while in the G23 (Corner Rounding) Mode and the normal Ramp Time parameter setting (established by General Parameter 005) will be used while in the G24 (Non-Corner Rounding)Mode.

If this parameter is set to Yes (1), the normal Ramp time (established by General Parameter 005) will be in effect for both the G23 (Corner Rounding) and G24 (Non-Corner Rounding) Modes.

The default value is 1.

SECTION 7-9: TUNE

This section allows the User access to the Servo Loop Gains. The Unidex 400 contains an Auto-Tune feature which will automatically select gains for the system, providing that certain conditions are met. Gain values may also be entered manually. A positive or negative step response may be executed with optional graphic output.

Following selection of **Tune** from the Setup menu, the display will be:

```
Set Up Mode
AUTO TUNING,     STEP RESPONSE



X       Y       Z       U       image       quit
```

NOTE: **If an image is to be generated from the Tune procedures, the Unidex 400 must be configured for the image type, prior to selecting the axis to be tuned.**

X-U Select X, Y, Z or U to designate the axis to be tuned.

The display for the X axis will be:

```
X Kp  (29000      ) 29000          step  500
N Ki  (200 0      ) 200 0          current % 30
  Kd  (276000     ) 276000
  Kf1   (0        ) 0
  Kf2   (0        ) 0
del      auto     +SR      -SR      save      next
```

image Select **image** from the initial Auto-Tune menu to configure the Unidex 400 to generate an image file as a result of a step response.

Following this selection, the display will be:

Set Up Mode				
AUTO TUNING,	**STEP RESPONSE**			
no	**rs232**	**F9901**	**Graph**	**quit**

no Select **no** if an image file is not to be generated.

rs232 Select **rs232** to send two blocks of 220 data points (1 mSec sample rate) through the RS-232 Serial Port. The first block represents the attained percentage of the desired position, the second block represents the percentage of deviation from the desired velocity. This data is in ASCII format.

F9901 Select **F9901** to send 512 data points (1 mSec sample rate) to internal file 9901 This file is a system file and cannot be edited without first copying it to another filename.

Graph Select **Graph** to output data in a "special" format that is utilized by the optional IBM Graphics Software (contact an Aerotech representative for information on this option).

quit Select **quit** to abort the image command and return to the initial Auto-Tune display.

Following configuration of the image file, select **quit** to return to the initial Auto-Tune display.

THE TUNE DISPLAY

Select the axis to be tuned (the X axis has been selected for this example), the display will be:

```
X Kp   (29000      ) 29000             step  500
N Ki   (200        ) 200               current % 30
  Kd   (10000      ) 10000
  Kf1  (0          ) 0
  Kf2  (0          ) 0
del      auto      +SR       -SR       save       next
```

The **X**, at the upper left corner of the display, designates the name of the axis to be tuned.

The **N** , directly below the axis name, designates the type of image file that will be generated. The designations at this location may be interpreted as follows:

```
X Kp   (29000      ) 29000             step  500
N Ki   (2000       ) 2000              current % 30
  Kd   (276000     ) 276000
  Kf1  (0          ) 0
  Kf2  (0          ) 0
del      auto      +SR       -SR       save       next
```

N = No image file will be generated
R = RS-232
F = File 9901
G = IBM optional graphics software

The next column of the display contains the names of the gains. The system default values (available from Axis Parameters x10 through x14 - **preset**) are shown below.

```
X Kp   (29000        ) 29000               step  500
N Ki   (2000         ) 2000                current % 30
  Kd  (276000        ) 276000
  Kf1   (0           ) 0
  Kf2   (0           ) 0
del         auto     +SR        -SR        save          next
```

The third column of the Tune display contains Gain values that have been saved to the Setup Memory. They are retained following a Reset or Power-Down.

```
X Kp   (29000        ) 29000               step  500
N Ki   (200          ) 200                 current % 30
  Kd  (10000         ) 10000
  Kf1   (0           ) 0
  Kf2   (0           ) 0
del         auto     +SR        -SR        save          next
```

These values may be set in two ways, from the Axis Parameter Mode (x10 through x14) or through this display by "saving" the "Intermediate" values.

NOTE: **System default values may be re-loaded for these settings by returning to the Axis Parameter menu and selecting "preset" for parameters x10 through x14.**

The fourth column of the Tune display contains the "intermediate" values. These values are established either by initiating the Auto-Tune function, or by manual setting.

```
X Kp   (29000        ) 29000               step  500
N Ki   (200          ) 200                 current % 30
  Kd   (10000        ) 10000
  Kf1   (0           )  0
  Kf2   (0           )  0
del         auto     +SR        -SR        save          next
```

Manual setting of these values is accomplished by use of the "left", right", and "up" and "down" arrow keys to move the cursor to the desired location, then keying in the new value. Refer to the description of **+SR** and **-SR** for details on setting these values.

The upper right of the Tune display contains the **Step** Size setting.

```
X Kp   (29000        ) 29000               step  500
N Ki   (200          ) 200                 current % 30
  Kd   (10000        ) 10000
  Kf1   (0           )  0
  Kf2   (0           )  0
del         auto     +SR        -SR        save          next
```

The Step Size is the number of machine steps that the motor moves during a step response.

Typically, the Step Size is 1/4 of the motor's resolution, i.e. a Step Size of 500 steps per resolution would be used for a 2000 count encoder. The Step Size is not generated by the Auto-Tune procedure. It may be entered by Axis Parameter x40, or at this display.

NOTE: **If a Step Size value is selected that is too large, Response Overshoot will consistently occur, regardless of Gain variations.**

Directly below the Step Size setting is the **Current** Command setting.

```
X Kp   (29000        ) 29000              step   500
N Ki   (2000         ) 2000               current % 30
  Kd   (276000       ) 276000
  Kf1    (0          ) 0
  Kf2    (0          ) 0
del        auto      +SR        -SR       save         next
```

The Current Command setting is used in conjunction with the Auto-Tune procedure only. The Current Command setting may be increased or decreased in steps of 5% to increase stabilization in the percentage of Overshoot and/or Settling Time. Refer to Figure 7-3 for an illustration of Overshoot and Settling Time.

TUNE OPERATIONS

The following operations are available from the Tune display:

```
X Kp   (29000        ) 29000              step   500
N Ki   (2000         ) 2000               current % 30
  Kd   (276000       ) 276000
  Kf1    (0          ) 0
  Kf2    (0          ) 0
del        auto      +SR        -SR       save         next
```

del Select **del** to delete a character at the cursor location.

DANGER: **THE AUTO-TUNE PROCEDURE WILL CAUSE SUDDEN MECHANICAL MOVEMENT. PRIOR TO BEGINNING THE AUTO-TUNE PROCEDURE, MAKE CERTAIN THE STAGE IS LOCATED AT MID-TRAVEL AND ALL PERSONS ARE CLEAR OF ALL MOVING PARTS.**

auto Select **auto** to initiate the Auto-Tune procedure. The Auto-Tune function may be used for the "Standard" Servo Loop Type only (Axis Parameter x41 set to 0). When initiated, the Auto-Tune excites the axis open loop at the percentage of maximum amplifier current, as specified under the **current** command %. The Unidex 400 then analyzes the feedback and determines "safe" gain settings It must be noted however, that the gain values derived from the Auto-Tune procedure may not be the optimum values with respect to ramp time, overshoot, settling time and system stability for a particular application and may need further manual adjustment.

The following limitations exist for Auto-Tune use:

Auto-Tune cannot be used with axes gantry (disconnect Gantry and tune individually if possible).

Auto-Tune cannot be used in conjunction with the Existence Checking parameter (x37) set at "3".

Auto-Tune cannot be used with the RMX Feedback option.

Auto-Tune cannot be used on a vertical axis.

Auto-Tune cannot be used for an axis with a high mass or high inertial load.

DANGER: **REQUESTING A +/- STEP RESPONSE WILL CAUSE SUDDEN MECHANICAL MOVEMENT. PRIOR TO SELECTING +SR/-SR, MAKE CERTAIN ALL PERSONS ARE CLEAR OF ALL MOVING PARTS.**

+SR/-SR The **+SR and -SR** selections are used to command a Step function of the magnitude specified by the step setting and "intermediate" gain values (fourth column of the Tune display). If the Step Size is set at "0", and +SR or -SR is selected, a step command is not generated but the gain values are sent to the Servo Loop. This may then be used to evaluate the effect of the new gains on the Servo Loop.

Manual Tuning is used to optimize Auto-Tune values or in cases where the Auto-Tune procedure is not appropriate. It is suggested that if the Auto-Tune procedure is not being used, the "Dual Feedback Servo Loop" be incorporated (Refer to Axis Parameter x41).

NOTE: **Prior to manually changing any gain settings, the initial settings should be recorded for future reference.**

NOTE: **Prior to changing axis parameters x26, x27 and x28 the current values should be saved to a file for reloading following the Tune procedure.**

While in the process of manual tuning, Servo Loop Trap errors may frequently occur. In order to facilitate the tuning process, Axis Parameters x26, x27, and x28 may be configured with maximum values. (Following the Tune procedure these parameters should be returned to their original values.)

Manual Tuning may also be accomplished from within the Axis Parameter menu with the settings of Axis Parameters x10, x11, x12 and x13. (Manual Tuning done in this manner requires continuous resetting of the system to activate the new gains.)

The following guidelines may be used to adjust the gain settings:

1. When activated, if the axis makes a high pitched, loud buzzing type noise, one or more of the gain settings are too large.

NOTE: **The axis motor fuse often will "blow" after a few seconds of continued loud oscillation. If such a condition occurs, establish the integrity of the fuse before continuing. If necessary replace the fuse (located on the Unidex 400's Rear Panel) with a fuse of the same type.**

DANGER: **THE POSSIBILITY OF ELECTRICAL SHOCK EXISTS, PRIOR TO REMOVING OR INSERTING A FUSE INTO THE REAR PANEL OF THE UNIDEX 400, MAKE CERTAIN POWER TO THE UNIDEX 400 IS OFF.**

2. When activated, if the axis appears to move about the desired position at a low rate of speed, seemingly "hunting" for it's position, one or more of the gain settings are set too low.

3. If Feedback resolution of an axis is increased, the gain values should be decreased.

4. If the (inertial) load of an axis is increased, the gain values should be increased.

An Aerotech 1050 motor has the following specifications:

Rotor Intertia = .00807 in. sec. ^{2}

Torque Constant = 9.3207 in/amp.

500 Line Encoder x 4 = 2000 counts/rev.

Stable Gain settings for this motor with an unloaded shaft are:

Kp	200,000
Ki	20
Kd	1000
Kf1	1
Kf2	0

These gain values are used in conjunction with the above steps (1-4) to select starting gains for the Tune procedure.

The following additional steps may also be used to determine gain settings:

5. Increase the Kp (Velocity Loop Gain) until the motor offers considerable resistance to movement as determined by "feel" of the motor shaft. If the Kp value is too low, the motor shaft will feel "spongy". If the Kp value is too large, the motor will "buzz".

6. The Kd (Position Loop Gain) should be increased such that the motor visibly positions quickly, demonstrating little overshoot. This setting also inversely affects the settling time. (see Figure 7-4). This value is usually less than the Kp value and greater than the Ki value. If the Kd value is too large the system will also oscillate.

7. Increase the Ki (Integral Gain) value to decrease the settling time (see Figure 7-4). Increasing the Ki value however, also increases the percentage of overshoot. The Ki value is usually the smallest of the three main Servo gain values. (Kp, Kd, Ki) Setting the Ki value too large will increase system instability.

Two Servo Feedback Loop types are available with the Unidex 400. The standard type (default) is illustrated in Figure 7-1. The alternate Dual Loop type is illustrated in Figure 7-2. Depending upon the loop type being used, the Gain Settings may vary.

NOTE: Systems configured for Dual Loop Servo Feedback must be manually tuned.

The following table illustrates PID Gain usage for each loop type:

GAIN SETTING	STANDARD	DUAL LOOP
Kp	Output is directly proportional to position error, producing a constant counteracting force to the error. Increasing Kp increases overshoot and oscillation and decreases risetime steady state error and damping effect.	Effects damping within the system. Increasing Kp reduces velocity loop gain overshoot and oscillation, and increases damping and rise time. This is the Velocity Loop Gain.
Ki	Output is a summation of the position errors, producing an increasing counteracting force for a constant or increasing position error. Increasing the Ki value increases the overshoot and settling time and decreases the rise time and damping effect.	Effects the "stiffness" of the Servo Loop. Increasing the Ki value decreases the axis settling time and steady state error.This is the Velocity Loop Integral Gain.
Kd (PGain)	Dampens system response by producing a restraint as the system progresses toward error reduction. Increasing the Kd value increases the rise time and damping effect and decreases the overshoot and settling time.	Effects the position error of the system. Increasing the Kd value decreases the position error and decreases the axis settling time during a move. This is the Position Loop Gain.

GAIN SETTING	STANDARD	DUAL LOOP
Kf1	The Velocity Feedforward value is a function of the system friction and torque constant. It complements PID by reducing the compensation effort required thus reducing system lag.	Enables/disables velocity feedforward. Assumes a binary value (i.e. ON/OFF). If Kf1 does not equal zero the velocity feedforward is enabled. When Kf1 equals zero, velocity feedforward is disabled.
Kf2	The Acceleration Feedforward value is a function of the system inertia and torque constant. It complements the PID by reducing the compensation required, thus reducing system lag.	Reduces the peak velocity/position error during acceleration or deceleration.

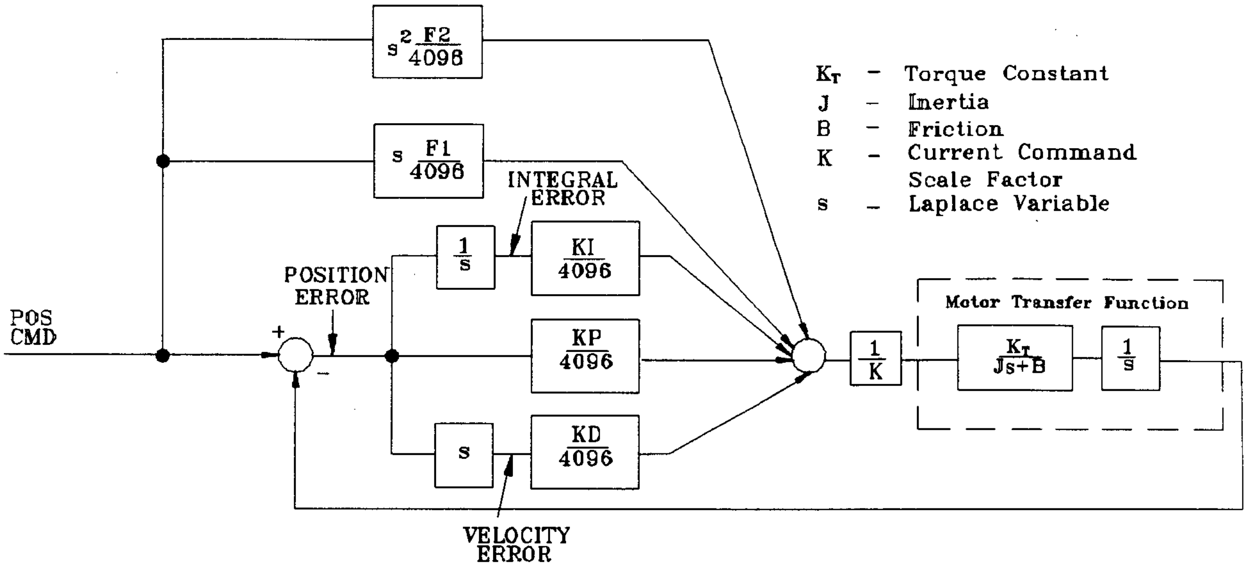


Figure 7-1: Standard Unidex 400 Servo Loop

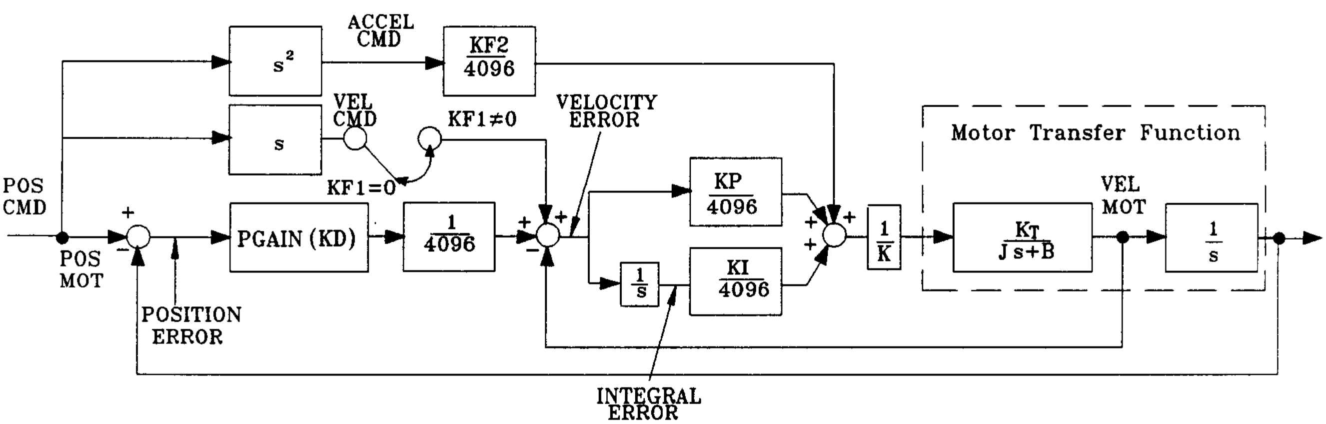
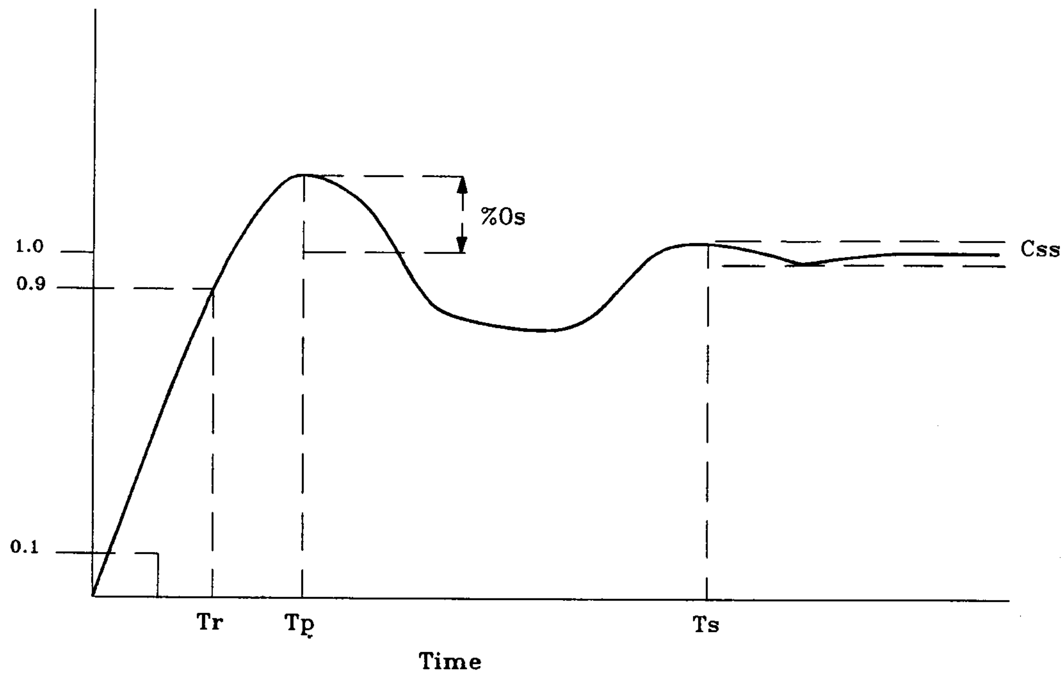


Figure 7-2: Dual Feedback Servo Loop



Tr	Rise Time of the Step Response is defined as the time that is required for the response to rise from 10 percent of the final value to 90 percent of the final value.
Tp	Peak Time is the time that the response takes to reach it's peak value.
Ts	Settling Time is the time required for the output to stabilze to within 5 percent of it's final value.
Css	Steady State is the final value of the Step Response.
%Os	Percent Overshoot is derived by the following formula:

$$\text{Percent Overshoot} = \frac{\text{Peak Value} - \text{Steady State Value}}{\text{Steady State Value}} \times 100$$

Figure 7-3: Typical Step Response

PARAMETER	Kp Increase	Ki Increase	Kd Increase
Rise Time	Decrease	Decrease	Increase
Overshoot	Increase	Increase	Decrease
Settling Time	Increase	Increase	Decrease
Steady State Error	Decrease	Decrease	—
Damping Factor	Decrease	Decrease	Increase

Figure 7-4: Effects of Gain Adjustments on System Response

(THIS PAGE INTENTIONALLY LEFT BLANK)

CHAPTER 8: THE TEST MODE

SECTION 8-1: INTRODUCTION

The Unidex 400 contains a self test mode that may be initiated through the main selection menu. It is not necessary to do a system power-down or reset prior to performing the test.

The Test Mode provides the ability to test three storage locations, EPROM, RAM and the SETUP Parameters.

SECTION 8-2: GETTING STARTED

Following power-up, the following selection menu is displayed:

```
UNIDEX 400                          Version x.xx

  RAM:ok    ROM:ok    SETUP:ok
  User RAM: xxxxxx

edit    file    setup    test    machine    PSWD
```

NOTE: **The PSWD option (F6) is displayed only if the Password function has been activated from the Parameter Mode**

Press the F4 key to enter the Test Mode. The following screen will be displayed:

```
test mode



RAM        ROM        SETUP                              quit
```

RAM Select **RAM** (F1) to read / write verify each byte of User RAM. If an error is detected, the Unidex 400 will show the RAM address of the error. This is a Fatal Error and indicates a system hardware failure. If such an error occurs, contact an Aerotech Representative.

A checksum verification is also performed on all User programs in memory. If a file is found to be corrupt, the Unidex 400 will display the file's name. The file may or may not be recoverable. This type of error is generally due to Battery failure.

ROM Select **ROM** (F2) to generate a ROM checksum to be compared with the stored checksum. If an error is detected, contact an Aerotech representative.

SETUP Select **SETUP** (F3) to generate a checksum of Setup data and compare it with the stored Setup data checksum. This type of error is generally due to Battery failure. If an error is detected, contact an Aerotech representative.

quit Select **quit** to abort file output and return to the initial file management display.

CHAPTER 9: REMOTE OPERATION

All models of the Unidex 400 may be remotely operated by a Terminal or Host Controller. This chapter contains operating information for each of these methods. Also contained within this Chapter is System Status information.

SECTION 9-1 REMOTE OPERATION THROUGH A TERMINAL

9-1-1 HARDWARE CONFIGURATION

Prior to using a Terminal to communicate with the Unidex 400, the following Jumper configurations must be present on the CPU Board. This may be accomplished as follows:

1. Move the Unidex 400 Power ON/OFF Switch into the OFF position. Disconnect the Unidex 400 from the Power Source.

DANGER: **POSSIBILITY OF ELECTRICAL SHOCK EXISTS. PRIOR TO OPENING THE UNIDEX 400 ENCLOSURE, MAKE CERTAIN ALL POWER SWITCHES ARE IN THE "OFF" POSITION AND ALL POWER CONNECTIONS HAVE BEEN DISCONNECTED.**

2. Remove the Unidex 400 Front Panel. Slide the Chassis from the Front Panel.

WARNING: **STATIC DISCHARGE MAY SEVERELY DAMAGE THE UNIDEX 400 CIRCUIT BOARDS. A STATIC STRAP, CONNECTED TO EARTH GROUND, MUST BE WORN BY ANYONE HANDLING THE CIRCUIT BOARDS.**

3. Remove the CPU Board, labeled A4, from the Chassis. (Refer to Figure 10-1 and 10-2 for board position).

   Locate Jumpers JP 5 and JP 8. (Refer to Figure 10-9.)

   Set Jumpers 5 and 8 to 2-3.

4. Re-install the CPU Board (A4) into the proper location in the card cage .

5. Replace the Front Panel onto the Chassis.

DANGER: PRIOR TO CONNECTING THE UNIDEX 400 TO THE POWER SOURCE, MAKE CERTAIN ALL CONNECTIONS ARE SECURE.

6. Connect the Unidex 400 to the power source.

A Video Display Terminal may interfaced to the Unidex 400 through the Rear Panel's TERMINAL Connector (P21). The hardware communication signals CTS and RTS are not used, instead the XON/XOFF protocol is used. The TERMINAL port is configured as a DCE with 9600 baud, 8 data bits, no parity and 1 stop bit.

9-1-2 TERMINAL MODES OF OPERATION

There are three modes possible for Terminal use:

(1) Dumb Terminal — A complete screen refresh is done whenever new display data is present. The only control characters to which the dumb terminal will respond are a carriage return (13) and a line feed (10). When in the Machine Mode, the tracking display is disabled.

(2) Smart terminal — Utilizes the default TeleVideo 905 cursor commands. When in the Machine Mode, the tracking display is enabled.

(3) Custom terminal — Permits the User to input cursor positioning commands. When in the Machine Mode, the tracking display is enabled. The following ASCII sequences should be input in Hexadecimal format:

CURSOR OFF - 3 bytes
CURSOR ON - 3 bytes
CLEAR LINE - 2 bytes
PLACE CURSOR - 2 bytes
CURSOR OFFSET - 1 byte

Custom or default Smart Terminal commands are stored in battery backed RAM. If data corruption occurs due to battery failure or other unusual circumstances, the display will default to the Dumb Terminal case. Smart terminal commands may be re-entered following battery replacement.

A new terminal type may be selected by pressing the **Ctrl** and ^ keys along with a number (1 thru 3) that corresponds to the case selected from the previous page. When in the Dumb Terminal Mode, the **ESC** key toggles between the current terminal screen and a Help file.

Terminal keys that are equivalent to Front Panel keys are listed in Table 9-4 , Mode 3 Command Table.

SECTION 9-2: REMOTE OPERATION THROUGH A HOST CONTROLLER

9-2-1 HARDWARE CONFIGURATION

A Host Controller may interfaced to the Unidex 400 through the Rear Panel's PORT A (P30) or IEEE-488 (P11) Connectors.

No Jumper changes need be made to the standard Unidex 400.

9-2-2 HOST CONTROLLER MODES OF OPERATION

There are five modes possible when using a Host Controller to control the Unidex 400. The selected mode may be changed at any time regardless of the current operating mode, by one of the following methods:

From an IBM keyboard connected to the Unidex 400's Front Panel, press the **Ctrl** and] keys along with a number (0 thru 4) that corresponds to the selected case.

or

Send the following two bytes of decimal data, 29 and the number that corresponds to the selected case, 0 thru 4 (48 thru 52), where "0" represents the ASCII value of the character zero.

Example: Set U400 to Remote Mode 1

Send two bytes to U400 PORT A

First Byte	= 29 (decimal) or 1D hexadecimal	=	access code
Second Byte	= 49 (decimal) or 31 hexadecimal	=	mode 1

NOTE: **While in the Remote Mode, all entries made through the IBM keyboard (or Front Panel) are blocked except for the entries to change Modes as illustrated above.**

MODES OF OPERATION

MODE 0	Normal mode with front panel keyboard and vacuum fluorescent display. This is the default mode of operation.
MODE 1	When in this mode, ASCII sequences that are followed by a carriage return (13) or a line feed (10), or both, are decoded to perform specific remote functions. (Refer to 9- 3, MODE 1 Command Table.) Return strings are terminated with the EAF character. Interface is through Port A. Refer to Chapter 7: The Setup Mode, General Parameters 003, 006 and 007.
MODE 2	Operation is the same as for MODE 1, except interface is through the IEEE-488 port. Refer to Chapter 7: The Setup Mode, General Parameter 032. Sequences are terminated with a EOS or EOI character.
MODE 3	Operation is similar to that of MODE 0 except the Front Panel Display and Keyboard are not used. Data is input from Port A. The MODE 3 Command Table lists codes that are equivalent to Front Panel Keyboard inputs. Unidex 400 returns the same character to Port A when it has successfully read and decoded it. If an error should occur during remote operation, the Unidex 400 will feedback two bytes, "87H" then the specific error code. (Refer to Section 9-5.)
MODE 4	Operation is the same as for MODE 3 except that data is input through the IEEE-488 port. Refer to Chapter 7: The Setup Mode, General Parameter 032.

SECTION 9-3: MODE 1 COMMAND TABLE

COMMAND	DESCRIPTION
PD	Print Directory of files in memory
Pnn	Print Program number nn
PS	Print Status bytes (see Section 9-5)
PX/PY/PZ/PU	Print axis X,Y,Z,U commanded position
Px/Py/Pz/Pu	Print axis X,Y,Z,U encoder position (actual)
PA	Print All programs in memory
Q	Query (Serial Poll), (See Section 9-5, Byte 1)
C	Reset system
F	Insert block number when printing program
G	Cancel block number printing (default)
M	Transmit status/position in binary (default)
N	Transmit status/position in hex - ASCII
J	Enable SRQ (default)
K	disable SRQ

< I (I cmd) send SRQ with C∅H

SECTION 9-4: MODE 3 COMMAND TABLE

UNIDEX 400 KEY	TERMINAL KEYS	DECIMAL
F1	Ctrl A	1
F2	Ctrl B	2
F3	Ctrl C	3
F4	Ctrl D	4
F5	Ctrl E	5
F6	Ctrl F	6
up arrow	Ctrl G	7
dwn arrow	Ctrl H	8
lft arrow	Ctrl I	9
right arrow	Ctrl J	10
"0"-"9"	"0"-"9"	48-57
+/-	Ctrl U	21
.	Ctrl V	22
Enter	Ctrl W	23

UNIDEX 400 KEY	TERMINAL KEYS	DECIMAL
C-Start	Ctrl X	24
Feed hold	Ctrl Y	25
S1	Ctrl Z	26
S2	Ctrl [	27
S3	Ctrl \	28
set remote	Ctrl]	29
toggle HELP	ESC	27
terminal type	Ctrl ^	30
RESET	Ctrl del del	31,31

All other ASCII codes valid

NOTE: Unimplemented Status Bits are read as zero.

SECTION 9-5: SYSTEM STATUS

System Status is a group of thirteen bytes which represent the current state of the Unidex 400.

BYTE 1 : Serial Poll

		ZERO	ONE
	BIT 0	Incremental mode	Absolute Mode
	BIT 1	Not running program	Running a program
	BIT 2	Block mode	Auto mode
	BIT 3	Non-corner rounding	Corner rounding
	BIT 4	Communication disabled	Communication enabled
	BIT 5	Inactive,not executing an Immediate command	Active, executing an Immediate command
*	BIT 6	No service request sent	Service request sent
	BIT 7	No error detected	Error detected

* = not implemented

BYTE 2 : Editor Error Status

		ZERO	ONE
*	BIT 0	No Illegal character during download.	Illegal character during download.
*	BIT 1	Memory not full during download	Memory full
*	BIT 2	User memory checksum OK	User memory checksum error
	BIT 3	No Illegal command	Illegal command
	BIT 4-BIT 7	not used, always zero	

BYTE 3 : Run Time Error Status

		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	Z axis not in limit	Z axis in limit
	BIT 3	U axis not in limit	U axis in limit
*	BIT 4	No Illegal byte in Memory	Illegal byte in memory
	BIT 5	Program number valid	Program number invalid
	BIT 6	Memory not clear	Memory clear
	BIT 7	No user memory checksum error	User memory checksum error

* = not implemented

BYTE 4 : Run Time Error Status 2

		ZERO	ONE
*	BIT 0	Stop key not pressed	Stop key pressed
*	BIT 1	No repeat loop invalid	Repeat loop invalid error
*	BIT 2	No repeat loop incomplete	Repeat loop incomplete error
*	BIT 3	8 repeat loops not exceeded	8 repeat loops exceeded
*	BIT 4	Subroutine return ok	Subroutine return error
*	BIT 5	No incomplete subroutine	Incomplete subroutine
*	BIT 6	8 subroutines not exceeded	8 subroutine exceeded
*	BIT 7	No missing label error	Missing label

BYTE 5 : Communication Status 1

		ZERO	ONE
	BIT 0	Always zero	N/A
*	BIT 1	NO IEEE-488 hardware	IEEE-488 hardware
	BIT 2	Always zero	N/A
*	BIT 3	No IEEE-488 communication	IEEE-488 communication
	BIT 4	Always zero	N/A
*	BIT 5	No Serial poll	Serial poll
*	BIT 6	Not in SRQ mode	In SRQ mode
*	BIT 7	Not in HOLD mode	In HOLD mode

* = not implemented

BYTE 6 : Communication Status 2

		ZERO	ONE
	BIT 0	Front Panel present	Front Panel not present
	BIT 1	Always zero	N/A
*	BIT 2	In HOLD mode, no trigger	In HOLD mode, trigger received
*	BIT 3	U400 receive buffer not full	U400 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 in binary	Status bytes in Hex - ASCII
*	BIT 7	Not in TDT mode	In TDT mode

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	Z axis not in system	Z axis in system
BIT 3	U axis not in system	U axis in system
BIT 4	reserved	
BIT 5	reserved	
BIT 6	reserved	
BIT 7	reserved	

* = not implemented

BYTE 8 : Position Status

		ZERO	ONE
*	BIT 0	X axis not in position	X axis in position
*	BIT 1	Y axis not in position	Y axis in position
*	BIT 2	Z axis not in position	Z axis in position
*	BIT 3	U axis not in position	U axis in position
	BIT 4	X axis no Soft Limit	X axis in Soft Limit
	BIT 5	Y axis no Soft Limit	Y axis in Soft Limit
	BIT 6	Z axis no Soft Limit	Z axis in Soft Limit
	BIT 7	U axis no Soft Limit	U axis in Soft Limit

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	Z axis not moving	Z axis moving
BIT 3	U axis not moving	U axis moving
BIT 4 -BIT 7	not used, always zero	

* = not implemented

BYTE 10: Free Run Mode Status

	ZERO	ONE
BIT 0	X axis not in free run	X axis in free run
BIT 1	Y axis not in free run	Y axis in free run
BIT 2	Z axis not in free run	Z axis in free run
BIT 3	U axis not in free run	U axis in free run
BIT 4-BIT 7	not used, always zero	

BYTE 11: Input Status 1

	ZERO	ONE
BIT 0	Input 0 is 0	Input 0 is 1
BIT 1	Input 1 is 0	Input 1 is 1
BIT 2	Input 2 is 0	Input 2 is 1
BIT 3	Input 3 is 0	Input 3 is 1
BIT 4	Input 4 is 0	Input 4 is 1
BIT 5	Input 5 is 0	Input 5 is 1
BIT 6	Input 6 is 0	Input 6 is 1
BIT 7	Input 7 is 0	Input 7 is 1

* = not implemented

BYTE 12: Input Status 2

	ZERO	ONE
BIT 0	Input 9 is 0	Input 9 is 1
BIT 1	Input 10 is 0	Input 10 is 1
BIT 2	Input 11 is 0	Input 11 is 1
BIT 3	Input 12 is 0	Input 12 is 1
BIT 4 -BIT 7	not used, always zero	

BYTE 13: Output Status

BIT 0-BIT 7 8 bit value of digital output

SECTION 9-6: ERROR TRANSMISSION

During Data transmission, the Unidex 400 may encounter error conditions. When in the Remote Mode, the format of the Error Codes is dependent upon the type of Remote interface.

If control interface to the Unidex 400 is through the **TERMINAL** connector, error messages appear in the same format as those displayed on the integral Front Panel display. (Refer to Chapter 12 for Error message description.)

If interface to the Unidex 400 is through the RS-232 **PORT A** connector, Error Codes will be sent as follows:

Hardware/Fatal Errors :
87H followed by **E0H** followed by Error Code (1 byte)

Standard Errors:
87H followed by Secondary Error Code (1 byte)

If interface to the Unidex 400 is through the **IEEE-488** connector, Error Codes will be in the formats illustrated below:

Hardware/Fatal Errors:
Service Request followed by **E0H** followed by Secondary Error Code (1 byte)

Standard Errors:
Service Request followed by Secondary Error Code (1 byte)

The following section contains a list of each of the Error Codes and a brief description of each.

SECTION 9-7: ERROR CODES

9-7-1 Hardware/Fatal Errors

These Errors will cause the system to stop processing.

Code	Description	Source/Suggested Solution
80H	68000 Bus Error	Power Down Unidex 400. Contact Aerotech, Inc.
81H	68000 Address Error	Power Down Unidex 400. Contact Aerotech, Inc.
82H	68000 Illegal Address Error	Power Down Unidex 400. Contact Aerotech, Inc.
83H	68000 Zero Divide	Power Down Unidex 400. Contact Aerotech, Inc.
84H	68000 Line 1010	Power Down Unidex 400. Contact Aerotech, Inc.
85H	68000 Line 1111	Power Down Unidex 400. Contact Aerotech, Inc.
86H	68000 Uninitialized Interrupt	Power Down Unidex 400. Contact Aerotech, Inc.
87H	68000 Spurious Interrupt	Power Down Unidex 400. Contact Aerotech, Inc.
88H	(Not Used)	
89H	(Not Used)	
8AH	(Not Used)	

Code	Description	Source/Suggested Solution
8BH	(Not Used)	
8CH	(Not Used)	
9FH	(Not Used)	
A0H	Axis in Limit	A Limit Switch is active. Send the Remote equivalent of the Front Panel "S1" Key to move the axis out of the Limit.
A1H	Axis Trap	A Servo Loop condition has been exceeded, producing an error. This error will occur when the Servo Loop performance is undesirable (Refer to Chapter 7: The Setup Mode) or, if a component in the drive has failed (Amplifier, Motor, Encoder or Servo Board).
A2H	M Function Error	Acknowledge signal was not detected in amount of time specified by General Parameter 13.
A3H	S Function Error	Acknowledge signal was not detected in amount of time specified by General Parameter 15.
A4H	T Function Error	Acknowledge signal was not detected in amount of time specified by General Parameter 17.
A5H	DSP Feedback Error	Servo Board has detected an invalid code. Reset system, if error repeats, contact Aerotech, Inc.
A6H	Brake Engaged	Brake monitor input to Unidex 400 is in it's Active (Low) State
A7H	Gantry or Perpendicularity Error	The same axis has been specified as being Master and Slave or as Horizontal and Vertical. Reconfigure General Parameters 011 or 024.
AFH	(Not Used)	
B0H	MFO = 0, Feedhold or Quick Stop	MFO Pot completely CCW, Feedhold or Quickstop active
B1H	(Not Used)	
B2H	Joystick Case Axis in Limit	A Limit Switch has become active while in Slew (Joystick) mode.

9-7-2 Non-Fatal Errors

These errors indicate a correctable condition that prevents processing. If any of these Errors occur, correct the condition and Reset the system before continuing.

Initial Selection Menu

10H	An invalid soft key selection has been made
11H	Axes requested for motion do not match those activated in Axis Parameter x37, Existence Checking
12H	Entry has been requested into a Mode for which the current password does not have privilege
13H	Remote Protocol Error (Parity, Stop, Frame Overrun)

Edit Mode

20H	An invalid soft key selection has been made
21H	An illegal filename (non-numerical)
22H	Out of Memory - No room to load Edit File
23H	Requested File cannot be found
24H	Menu Driven input error, enter equivalent of the Front Panel S1 key
25H	Reserved filename

File Mode

30H	An invalid soft key selection has been made
31H	Invalid File Name
32H	Source File does not exist
33H	File to be input already exists
34H	Memory Full

Setup Mode

40H	An invalid soft key selection has been made
41H	File named to receive Setup Data is invalid
42H	File named, from which to load Setup Data, not found
43H	File named to receive Setup Data already exists
44H	Memory Full
45H	(Not Used)
46H	Parameter data entered is beyond limits
47H	Item Number selection not valid

Test Mode

50H	An invalid key selection has been made
51H	Read/Write Error in System RAM
52H	Read/Write Error in User RAM
53H	Checksum error in User's program
54H	Checksum Error in ROM
55H	Checksum Error within Setup values

NOTE: The Checksum format is such that each byte is added to a 16 bit Accumulator as an unsigned quantity.

Machine Mode

60H	An invalid key selection has been made
61H	A format error exists in data to be processed
62H	File requested to be Run cannot be found
63H	File requested to be Run is a reserved file
64H	Miscellaneous Error, press equivalent of the Front Panel S1 key

9-7-3 Run Time Errors

These errors indicate a correctable condition encountered during program execution. If any of these Errors occur, return to the Edit Mode, correct the condition and Rerun the program.

Code	Description	Source/Suggested Solution
80H	Unrecognized Command	
81H	G Code Error	Command Syntax is not expected
82H	Same G Code	Command repeated within program block
83H	G0 Syntax Error	Axis move must be specified before Feedrate
84H	Too many BCD Digits	
85H	Unrecognized Variable name	Variable has not been correctly defined. Refer to Chapter 3, Edit Mode, Math, File 9999 for addition of variables
86H	G2/G3 Syntax Error	Center Point not provided for requested Circle
87H	No Feedrate provided	If G0 used, feedrate must be specified. If G1,G2 or G3 is used previous feedrate will be used
88H	No MST Bus	Jumpers not properly configured. Refer to Chapter10, Hardware Information
89H	No I/O Bus	Jumpers not properly configured. Refer to Chapter10, Hardware Information
8AH	Undefined Math Function	Math Syntax Error
8BH	Undefined Variable	Variable has not been defined. Refer to Chapter 3, Edit Mode, Math, File 9999 for addition of variables
8CH	Illegal I/O Format	Incorrect Command Syntax
8DH	Illegal Math Format	Math Syntax Error
8FH	Unable to read File 9999	File does not exist

Code	Description	Source/Suggested Solution
90H	Unable to write to system variable	System variables cannot be overwritten
91H	Insufficient space for additional variable	File 9999 cannot be created because there is not enough available memory
92H	IEEE-488 Option requested, not available	Unidex 400 not equipped with MPI Board
93H	Memory allocated for User stack gone	Reconfigure General Parameter 010
94H	Repeat Loop requested but cannot be executed	Repeat Loop Syntax Error
95H	File requested not a Parts Program	File requested is a reserved file
96H	File not found	
97H	Label not Found	Label Syntax Error
98H	No return from GOSUB command	Illegal placement of "RTS"
99H	File 9999 Overflow	Too many variables defined

C0H I CMD complete

CHAPTER 10: HARDWARE INFORMATION FOR THE UNIDEX 400

This Chapter provides the User with hardware information for the Unidex 400. Due to the addition of various available options, certain differences may exist between a particular unit and that which is described here.

SECTION 10-1: DESCRIPTION OF THE MAJOR COMPONENTS

The Unidex 400 consists of the following major components; the CPU Board, the DSP Board, the Rear Panel Interface Board and the Axis Amplifiers. Common optional boards are the Front Panel Board, the Remote Front Panel Board and the MPI Board. Internal arrangement of these boards is illustrated in Figures 2-1, 2-2 and 2-3.

A brief description of each follows:

CPU Board

The Unidex 400 CPU Board is the front end processor of the Unidex 400. It contains a 10 mHz Motorola 68000 processor to provide file editing, file management, system parameters, machine modes and peripheral communications via RS-232. It also contains eight (8) Opto Isolated outputs and twelve (12) Opto-Isolated inputs. A 16 output MST bus is also optionally available.

The CPU Board communication consists of two RS-232 ports. Port A is a multi-purpose port which may be used to interface various peripheral equipment. The second RS-232 port is dedicated to an external terminal (TeleVideo 905 Display Terminal or equivalent). When a terminal is connected to this port, the Unidex 400 is fully operational from the Terminal. (Certain functions require specific key entries, see the section of this manual dealing with Remote Operation of the Unidex 400.)

The CPU Board also provides the User four (4) Interrupt inputs. All of these Interrupts may be defined within a program to perform a User specified function.

DSP Board

The DSP Board contains a Motorola 56001 Digital Signal Processor which controls the Servo loops of the axes by the use of software alogrithms. The alogrithms read the Encoder Feedback and generate current commands via D/A converters.

The DSP Board accepts Encoder and limit data from the four axes through DSP connector P3. The Encoder inputs accepted are differential line driven, square wave Encoder signals. The DSP Board is configured for a standard Quadrature Encoder. Clock and direction Encoders as well as CW and CCW Pulse Train Encoders may be used with appropriate jumper configurations. (Refer to Section 10-3.)

The Limit inputs consist of a CW, CCW, and Home Limit line for each axis. The Limit inputs may be active high or active low. Polarity is established by a parameter in the Setup Mode.

A variety of functions are interfaced to the DSP Board through the P2 connector. Of most significance are the Current Command output signals used to command the axis amplifiers. A shutdown signal that may be used to disable the drives is also included in this signal group.

The DSP Board contains two opto-coupled interrupt sources that are interfaced through the P2 connector. One is a "fast feedhold" input capable of stopping a current axis move in 1 milli-second as opposed to the implementation of a Feedhold command through the CPU Board, which may take up to 300 milli-seconds. The second interrupt is the "position-grabbing" interrupt which, when activated, stores the current axis position and reports it to the CPU Board.

When a Brushless Motor System is used, the DSP Board's P2 connector contains the Hall Effect Switch inputs. These switches (three(3) per axis) are located on the motors and output a six (6) step pattern which is read by the DSP56001 at power-up to initialize the motor position. Thereafter the pattern is read on a 0.25 milli-second basis to provide accurate motor commutation.

The P2 connector also provides the Vertical Axis Brake interface. Two lines are provided for this function. The first is a brake output line that activates a brake whenever a fault is encountered that causes the vertical axis to lose torque. The second brake line is an input to the DSP Board that signals the DSP56001 brake/axis integrity.

Rear Panel Interface Board

The Unidex 400 Interface Board contains all of the connectors normally required for operation. A description of each of the connectors is provided in Chapter 11.

A 1 Amp Slo-Blo fuse for Encoder power is also contained on this board.

Axis Amplifiers

The type of Axis Amplifier or Translator included in the Unidex 400 (excluding the Unidex 400N) is correspondent to the type of Motor used for each axis. The Unidex 400 is capable of controlling Stepper, DC Brush and AC Brushless motor types.

The AM16010 Drive Translator is used in conjunction with Stepper motors. The Stepper Motors may be wired for up to 160 volts at 10 Amps.

The DC Brush type motors use a PWM DC Servo Amplifier . The DC Amplifier may be wired for up to 160 volts PWM. A 20 Amp peak, 10 Amp continuous version (DS16020) is available, as well as a 30 Amp Peak, 15 Amp continuous (DS16030) version.

The AC Brushless Servo Amplifier is used for brushless motors. The AC Brushless Motors may be wired for 160 Volts, 20 Amps (AS16020 Amplifier) or 320Volts, 10 Amps (AS32010 Amplifier).

Front Panel Board

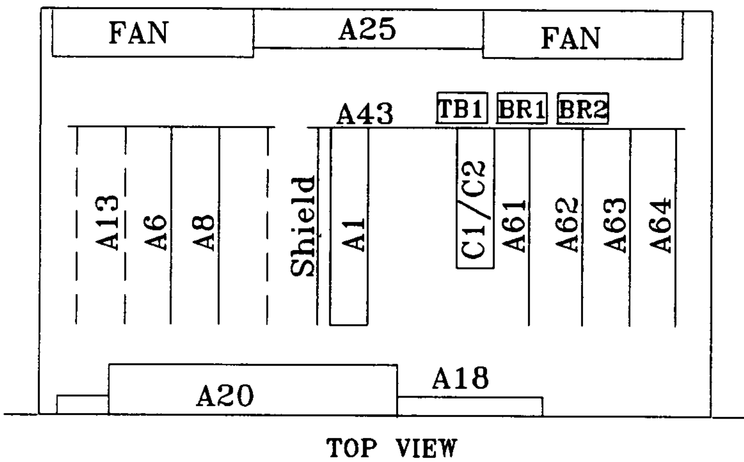
The Unidex 400 optional Front Panel Board provides several functions. It receives keyboard input and sends this input to the CPU Board, and drives the vacuum fluorescent display. The Beeper and Reset Switch are routed through the Front Panel Board as well as the switch and lamp signals for Power On/Off. If a RS-422 Board is included the Front Panel converts the differential signals received to standard logic levels.

Remote Front Panel Board

The optional Remote Front Panel Board is used to convert the Front Panel signals to differential signals in order to drive the signals through long lengths of cable.

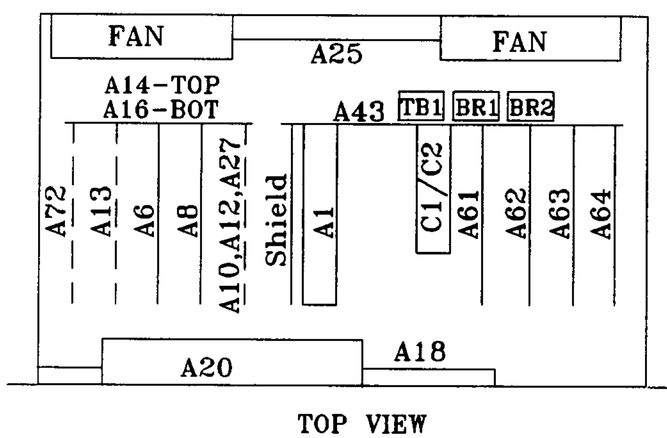
MPI Board

The optional MPI Board provides IEEE-488, Motorola I/O or a Pamux interface.



A1 Power Supply
A6 CPU Board
A8 DSP Board
A13 MPI Board
A18 Front Panel Board
A20 VF Display
A25 Rear Panel Interface Board
A43 Mother Board
A61 X Axis Amplifier
A62 Y Axis Aplifier
A63 Z Axis Amplifier
A64 U Axis Amplifier
BR1/BR2 Bridge Rectifier 1/2
C1/C2 Capacitors 1&2
TB1 Terminal Strip

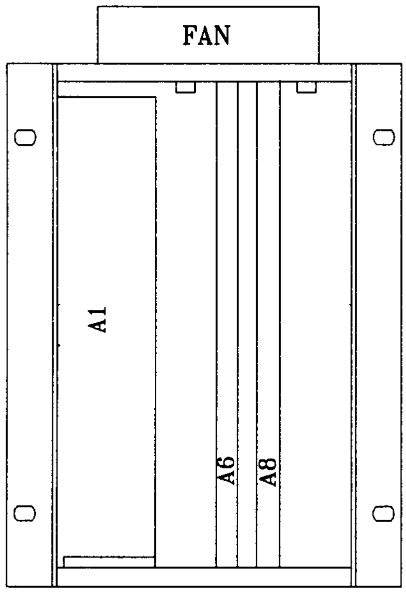
Figure 10-1: Internal Configuration of the Standard Unidex 400



A1 Power Supply
A6 CPU Board
A8 DSP Board
A13 MPI Board
A14 VME Backplane
A16 Extension Bus
A18 Front Panel Board
A20 VF Display
A25 Rear Panel Interface Board
A43 Mother Board
A61 X Axis Amplifier
A62 Y Axis Aplifier
A63 Z Axis Amplifier
A64 U Axis Amplifier
BR1/BR2 Bridge Rectifier 1/2
C1/C2 Capacitors 1&2
TB1 Terminal Strip

A10 R/D Board
A12 PSO Board
A27 RMX Board
A72 Remote Front Panel Board

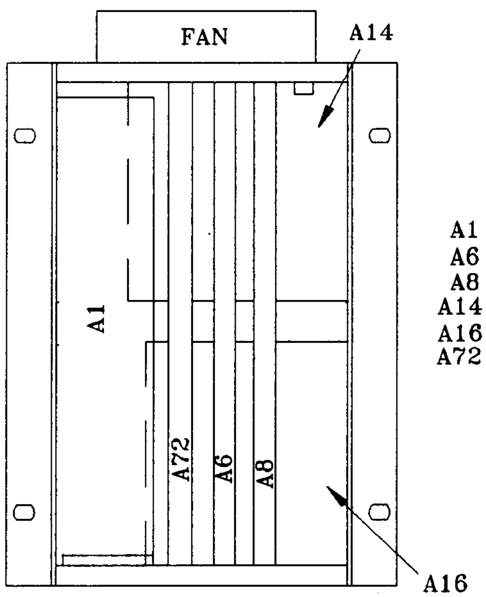
Figure 10-2: Internal Configuration of the Unidex 400 with Options



A1 Power Supply
A6 CPU Board
A8 DSP Board

FRONT VIEW

Figure 10-3: Internal Configuration of the Standard Unidex 400N



A1 Power Supply
A6 CPU Board
A8 DSP Board
A14 VME Backplane
A16 Extension Bus
A72 Remote Front Panel Board

FRONT VIEW

Figure 10-4: Internal Configuration of a Unidex 400N with Options

SECTION 10-2: SYSTEM WIRING

A basic Unidex 400 consists of a CPU Board, a DSP Board, a Rear Panel Interface Board, a Mother Board and the Axis Amplifiers. Refer to Figure 10-5 for a wiring diagram of a standard configuration.

Most controllers are equipped with one or more integral options. Refer to Chapter 2 of this manual for a list of these options.

Figure 10-6 provides a wiring diagram for a Unidex 400 equipped with integral options .

A wiring diagram for the standard Unidex 400N is provided in Figure 10-7.

Figure 10-8 provides a wiring diagram for a Unidex 400N that is equipped with integral options.

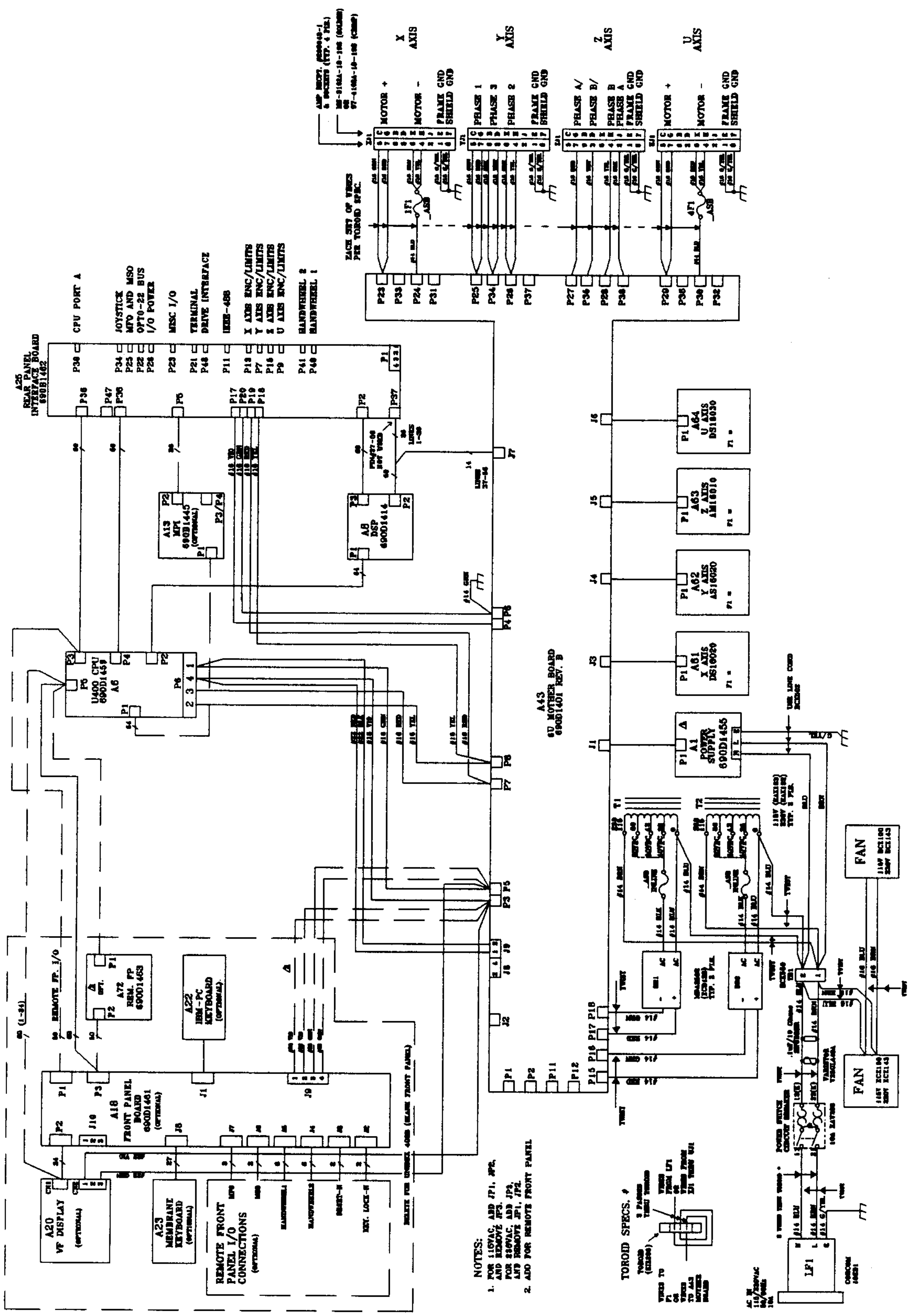


Figure 10-5: Unidex 400 Standard Wiring

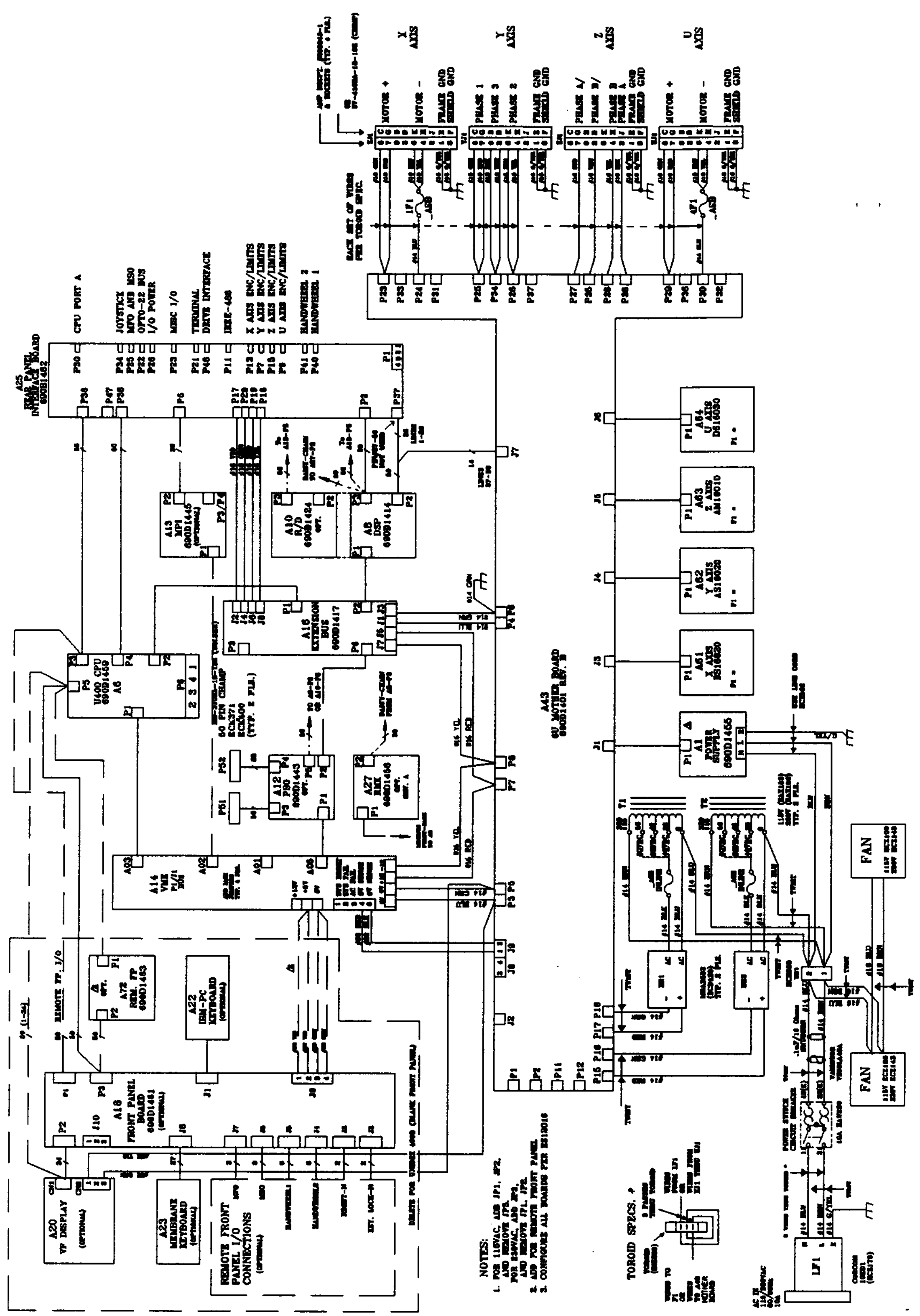


Figure 10- 6: Unidex 400 Optional Wiring

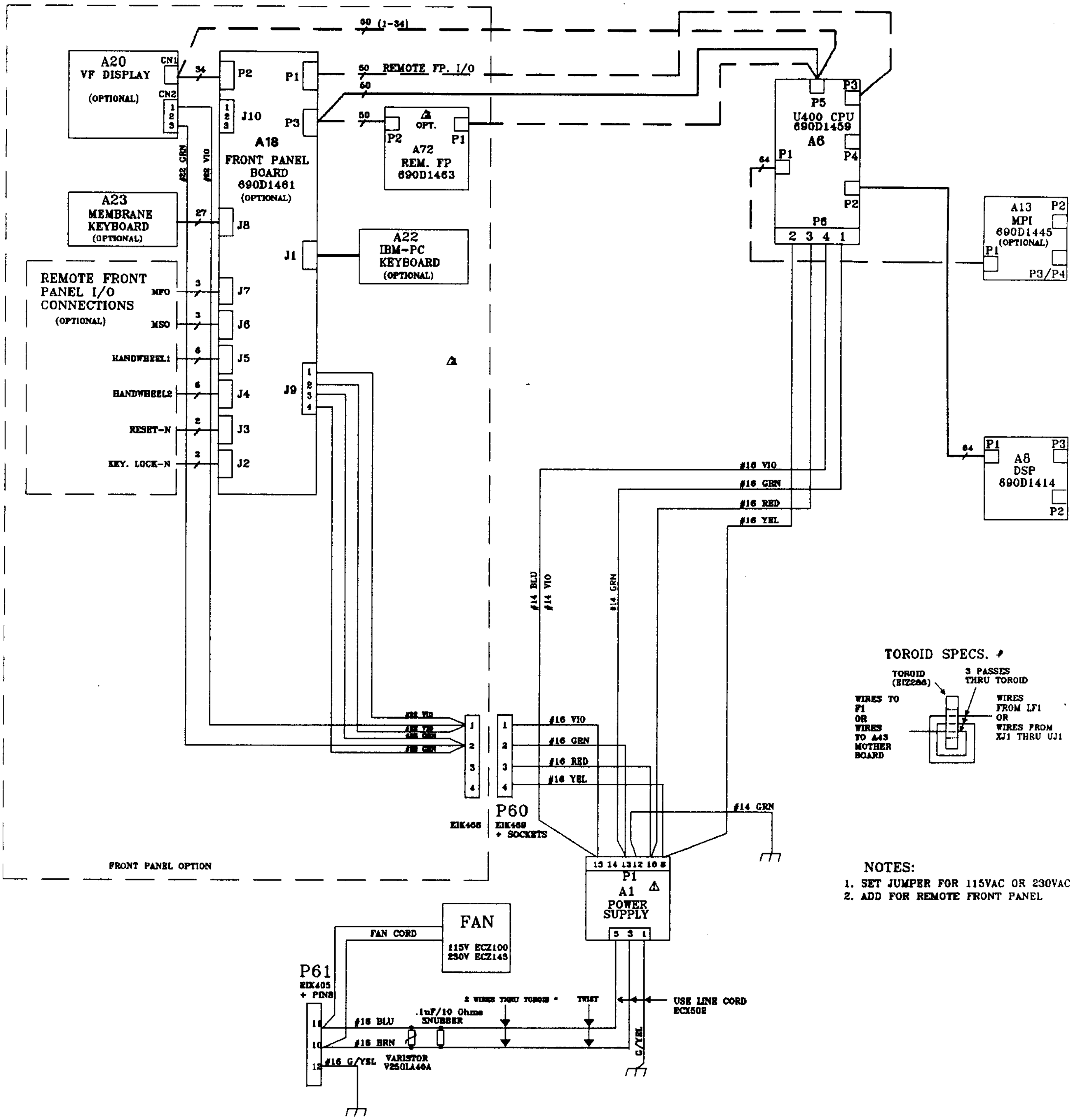


Figure 10-7: Unidex 400N Standard Wiring

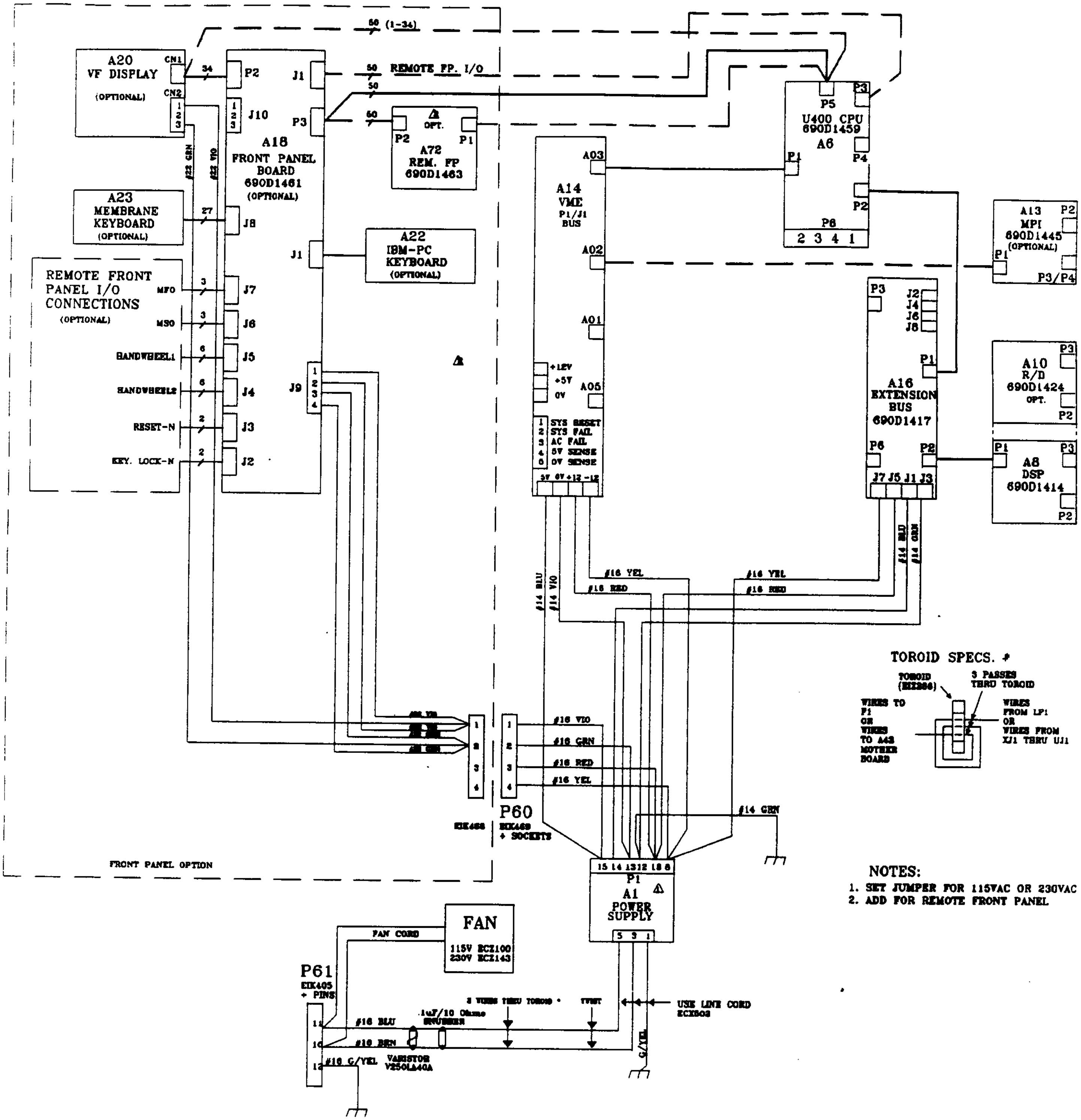


Figure 10- 8: Unidex 400N Optional Wiring

SECTION 10-3: JUMPERS AND SWITCHES

Many of the circuit boards that make up the Unidex 400 contain Jumpers and/or Switches. The following sections provide descriptions of the Jumper and Switch configurations for each of the boards. All Jumpers are factory set and should not be reconfigured by the User without consulting an Aerotech Service Representative. Refer to Figures 10-1 through 10-4 for relative board locations.

CPU Board - Jumpers/Switches

The following is a list of Jumpers that are located on the CPU Board:
(Refer to Figure 10-9 for Jumper locations).

Jumper	Conditions	Function
JP1	1-2	VME standby power disabled (default)
	2-3	VME standby power enabled
JP2	1-2	Disables Battery Backup
	2-3	Enables Battery Backup (default)
JP3	1-2	Enables MST Bus
	2-3	Enables I/O Bus (default)
JP4	1-2	Vacuum Fluorescent display connected through a Front Panel Board (default)
	2-3	Vacuum Fluorescent display connected directly to CPU Board
JP5	1-2	Configures Serial Port B for external IBM Keyboard Input (default)
	2-3	Configures Serial Port B for auxiliary RS-232C Serial Input
JP6	1-2	Enables Watchdog Timer (default)
	2-3	Disables Watchdog Timer
JP7	1-2	VME Bus Reset is not initiated from the CPU Board (default)
	2-3	VME Bus Reset is initiated from the CPU Board

Jumper	Conditions	Function
JP8	1-2	Front Panel option included on the Unidex 400 (default)
	2-3	Front Panel option not included on the Unidex 400
JP9	1-2	On Power-up the Unidex 400 resets the MST Bus
	2-3	On Power-up the Unidex 400 does not reset the MST Bus (default)
JP10	IN	Configured in conjunction with JP11 being OUT for use with 128k bytes or 256k bytes EPROMS (default)
	OUT	Configured in conjunction with JP11 being IN for use with 512k bytes EPROMS

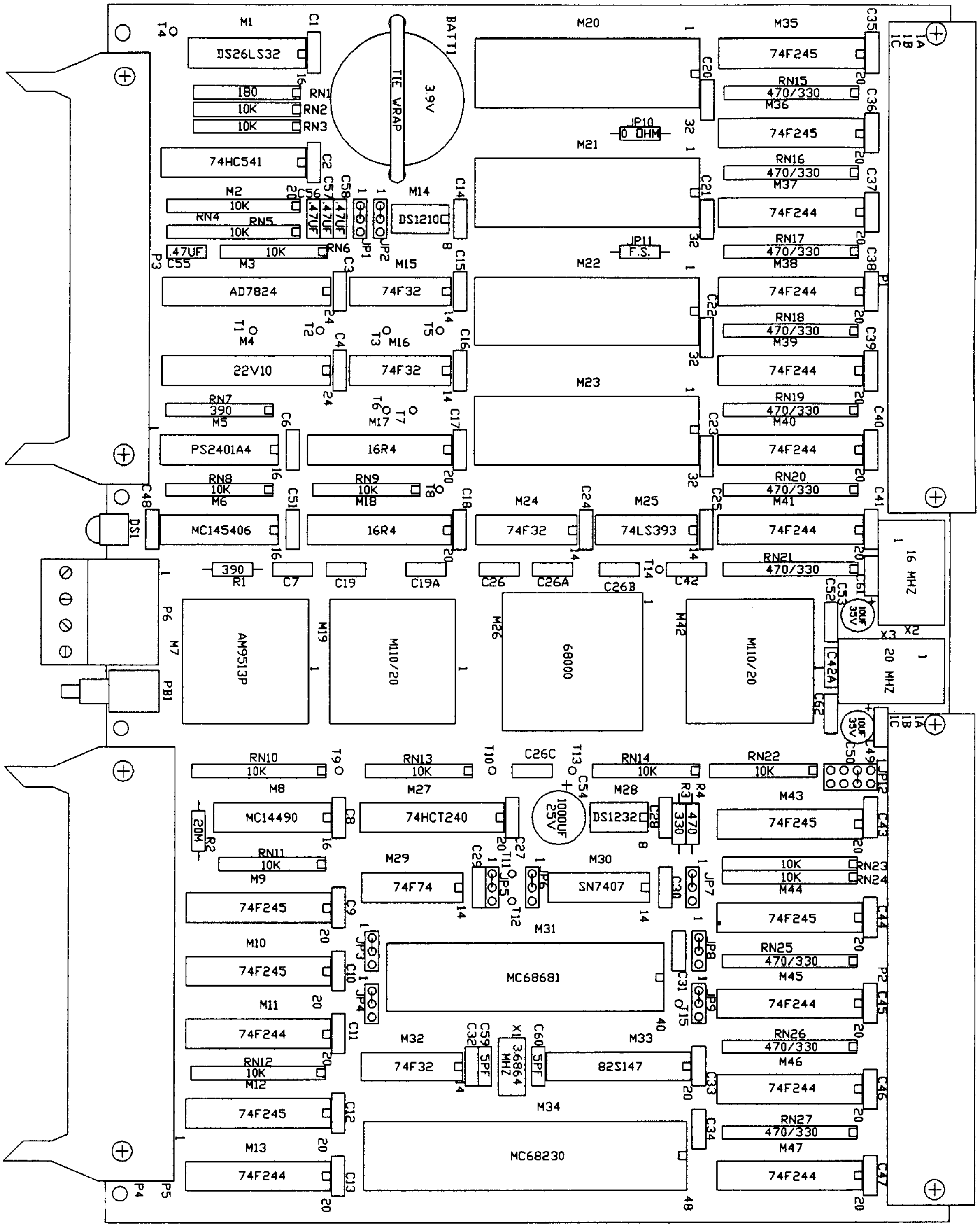


Figure 10-9: Unidex 400 CPU Board Assembly

DSP Board - Jumpers/Switches

The following is a list of Jumpers that are located on the DSP Board (Refer to Figure 10-10 for Jumper locations). These Jumpers are factory set and should not be reconfigured by the User.

Jumper	Conditions	Function
JP1	1-2	Enables Watchdog Timer (default)
	2-3	Disables Watchdog Timer
JP2	1-2	Establishes Opto-Isolator,M15 as HCPL2601
	2-3	Establishes Opto-Isolator,M15 as 6N136 (default)
JP3	1-2	Establishes Opto-Isolator,M13 as HCPL2601
	2-3	Establishes Opto-Isolator,M13 as 6N136 (default)
JP14	1-2	Establishes active low state to initiate shut-down (default)
	2-3	Establishes active high state to initiate shut-down
JP21	1-2	Configured in conjunction with JP22 and 23 also being set at 1-2 to designate an 8kx8 RAM
	2-3	Configured in conjunction with JP22 and 23 also being set at 2-3 to designate a 32kx8 or 128kx8 RAM (default)

JP24 thru JP26 — Establishes Extension Bus Address

JP24	JP25	JP26	Address
1-2	1-2	1-2	1x0xxxH, (default)
2-3	1-2	1-2	1x1xxxH,
1-2	2-3	1-2	1x2xxxH,
2-3	2-3	1-2	1x3xxxH,
1-2	1-2	2-3	1x4xxxH,
2-3	1-2	2-3	1x5xxxH,
1-2	2-3	2-3	1x6xxxH,
2-3	2-3	2-3	1x7xxxH,

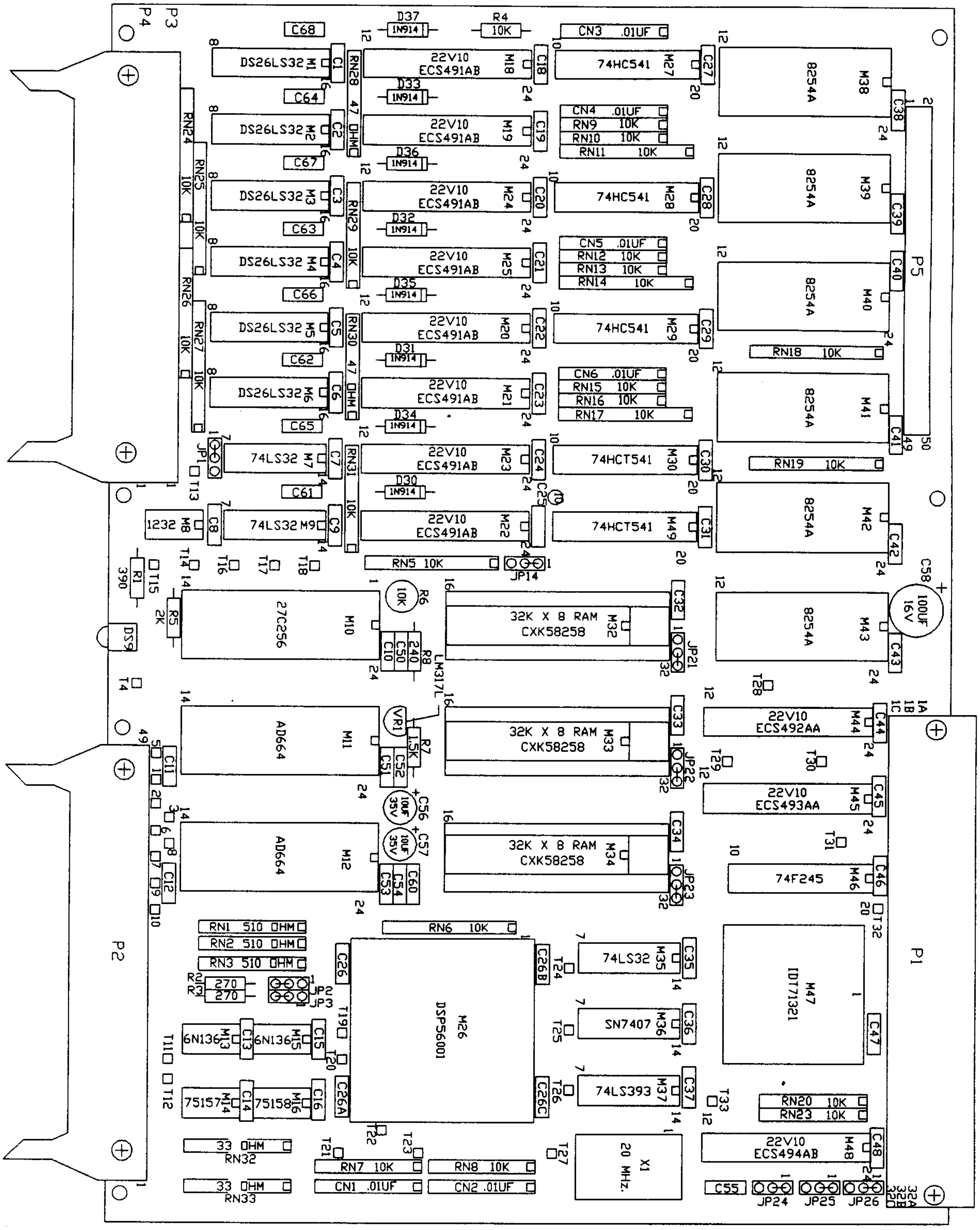


Figure 10-10: DSP Board Assembly

Mother Board - Jumpers/Switches

The following is a list of Jumpers that are located on the Mother Board (Refer to Figure 10-11 for Jumper locations). These Jumpers are factory set and should not be reconfigured by the User.

Jumper		Conditions		Function
JP1 thru JP4				Configures Unidex 400 for Dual Bus Voltage
JP1	JP2	JP3	JP4	Bus Voltage
IN	IN	IN	IN	No Dual Bus Voltage (default)
IN	IN	OUT	OUT	Shunt Regulated Bus #1
OUT	OUT	IN	IN	Shunt Regulated Bus #1
JP5, JP10, JP11				Configures Unidex 400 for the Drive Module
JP5	JP10	JP11		Drive Module
IN	IN	IN		Bipolar Linear Brush type
OUT	OUT	OUT		All other types (default)
JP6		1-2		X Axis is Brushless or Stepper type
		2-3		X Axis is Brush type Servo (default)
JP7		1-2		Y Axis is Brushless or Stepper type
		2-3		Y Axis is Brush type Servo (default)
JP8		1-2		Z Axis is Brushless or Stepper type
		2-3		Z Axis is Brush type Servo (default)
JP9		1-2		U Axis is Brushless or Stepper type
		2-3		U Axis is Brush type Servo (default)

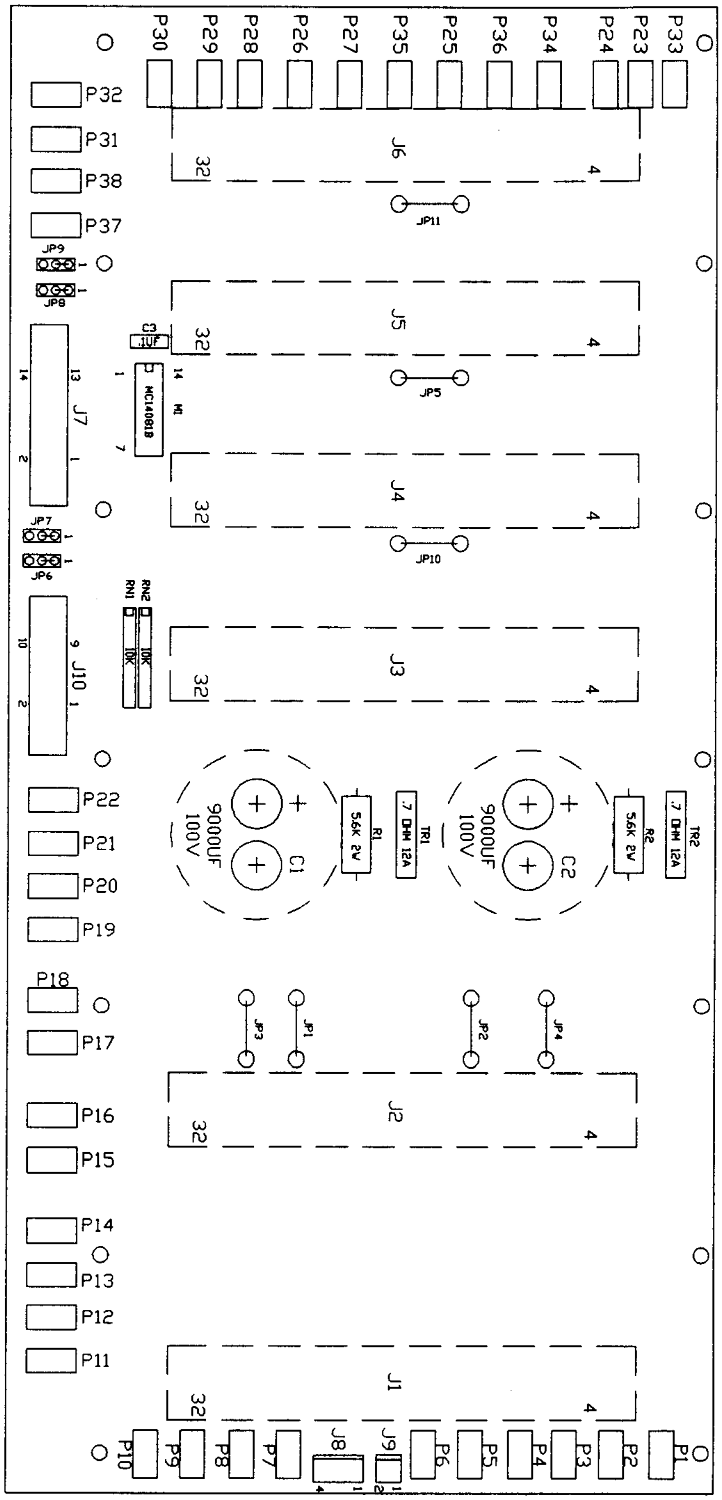


Figure 10-11: Mother Board Assembly

Front Panel Board - Jumpers/Switches

The following is a list of Jumpers that are located on the optional Front Panel Board: (Refer to Figure 10-12 for Jumper locations.)

Jumper	Conditions	Function
JP1	1-2	Disables Test Mode for the Vacuum Fluorescent Display (default)
	2-3	Enables Test Mode for the Vacuum Fluorescent Display
JP2	1-2	Enables Internal Speaker (default)
	2-3	Disables Internal Speaker
JP3	6-7 IN	Increases debounce time of the membrane Keyboard Matrix
	All Others Out	(Reserved)

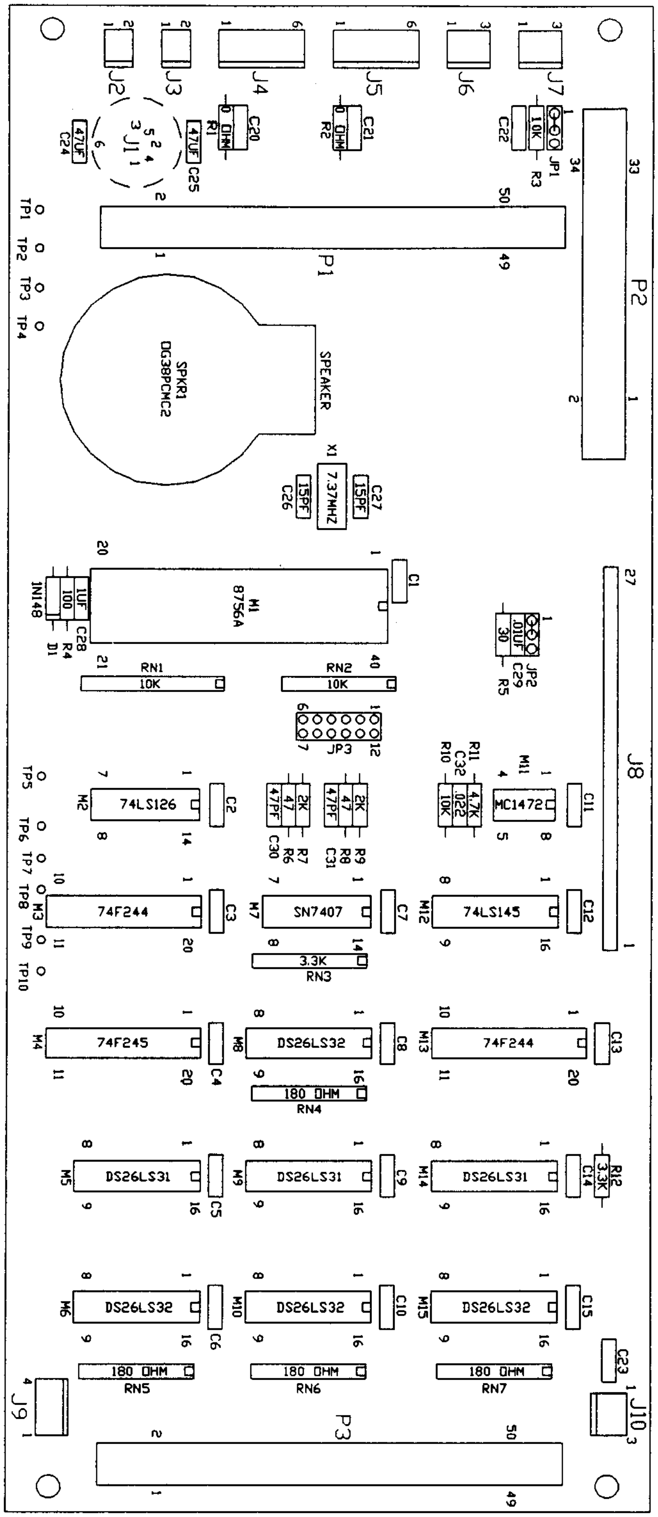


Figure 10-12: Front Panel Board Assembly

SECTION 10-4: FUSES AND CIRCUIT BREAKERS

The following table provides information for the fuses that are contained in the Unidex 400 system.

Fuse/Location	Rating	Aerotech P.N.
Main Power Switch/Breaker	5A	EAW234
	10A	EAW260
	15A	EAW255
Motor Fuses/Rear Panel (Dependant upon motor type)		
	Motor 1017-01 4ASB	EIF104
	Motor 1035-01 4ASB	EIF104
	Motor 1050-01 5ASB	EIF120
	Motor 1075-01 5ASB	EIF120
	Motor 1135-01 5ASB	EIF120
	Motor 1210-01 5ASB	EIF120
	Motor 1410-02 8ASB	EIF122
	Motor 1410-02 8ASB	EIF122
	Motor 1960-02 12ASB	EIF173
	Motor DS16020 12ASB	EIF173
	Motor DS16030 20ASB	EIF173
Motor Bus-Power Fuses, T1 T2 Transformer set for:		
	28VAC = 40VDC 15A	EIF116
	43VAC = 60VDC 12A	EIF173
	56VAC = 80VDC 10A	EIF110
Encoder Power/Rear Panel		
	1A	EIF189

CHAPTER 11: INTERFACE SPECIFICATIONS FOR THE UNIDEX 400

This Chapter contains interface specifications for the Unidex 400. The first section of this Chapter provides detailed information for the standard Unidex 400. Refer to Figure 11-1 for an illustration of the Unidex 400 rear panel.

SECTION 11-1: UNIDEX 400 INTERFACE

The standard configuration of the Unidex 400 is interfaced almost exclusively through the rear panel, the exception being the keyboard interface which is located on the front. Each of these connectors will be detailed in the subsequent sections.

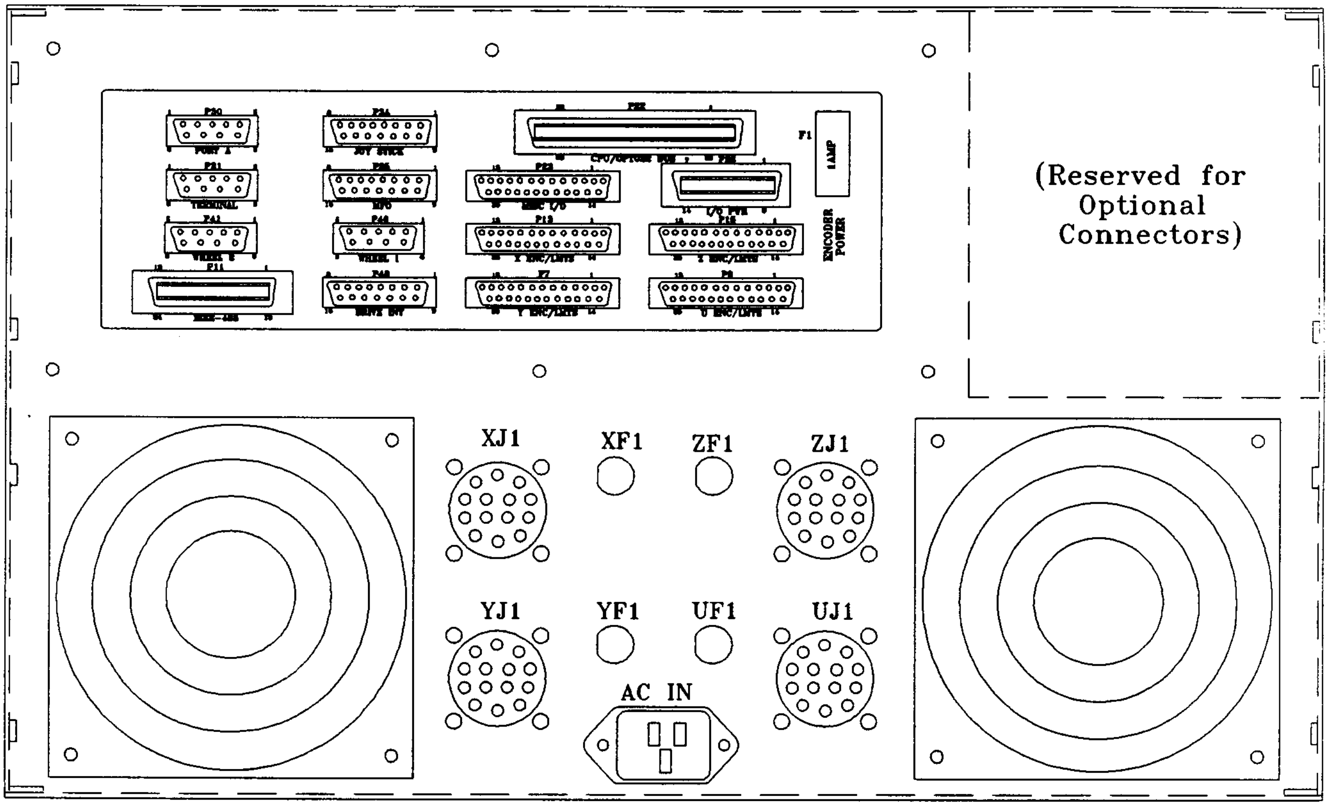


Figure 11-1: Unidex 400 Rear Panel

11-1-1 POWER CONNECTIONS

The Unidex 400 may be factory wired for an input power of either 115 VAC, single phase, 50/60 Hz 30 amps or 230 VAC, single phase, 50/60 Hz, 15 amps.

WARNING: ON THE SIDE PANEL OF EACH UNIDEX 400 IS A POWER REQUIREMENT LABEL. DEVIATION FROM THE POWER SPECIFICATIONS ON THIS LABEL COULD RESULT IN EXTENSIVE DAMAGE TO THE EQUIPMENT. CONSULT YOUR AEROTECH SERVICE REPRESENTATIVE FOR FURTHER INFORMATION.

Prior to making any power connections, refer to the Power Requirement Label for input power specifications.

Figure 11-2 shows an outline of the AC IN power connector. The Input power Cable is supplied with the Unidex 400.

WARNING: PRIOR TO MAKING ANY ELECTRICAL CONNECTIONS, MAKE CERTAIN THAT ALL POWER SWITCHES ARE IN THE OFF POSITION.

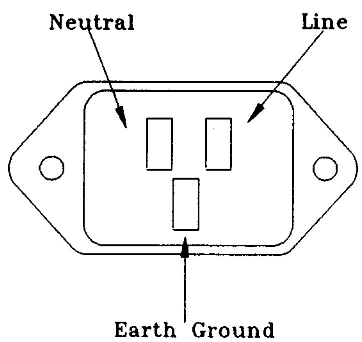


Figure 11-2: Unidex 400 AC IN Connector

11-1-2 MOTOR INTERFACE

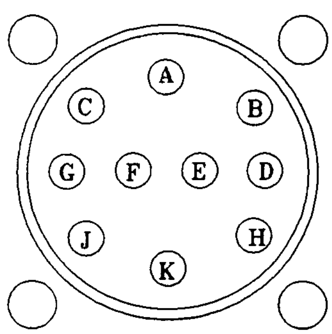
The Unidex 400 may be configured to interface with three motor types; Step, Servo and Brushless. Wiring details for each of these motor types are shown in Figures 11-4, 11-5 and 11-6.

Four motor output connections are located on the Unidex 400 Rear Panel. Details of these connectors are provided in Figure 11-3. Each of the connectors are identically wired.

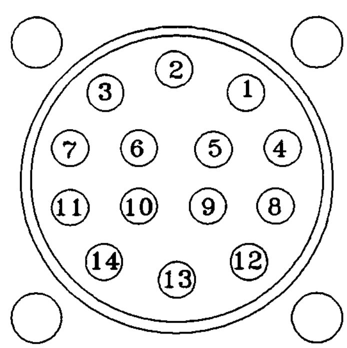
The following is a list of connectors that may be used as mating connectors to provide motor interface.

Amphenol PN	**Aerotech PN**	**Description**
10 Pin Metal (MS) Connector		
MS3106A-18	MCM475	Connector
MS-18-19P	MCM476	Insert
AN3057-1010	MCM477	Backshell

AMP PN	Aerotech PN	Description
14 Pin Plastic Connector		
206044-1	ECK131	Connector
206070-1	ECK134	Backshell
66098-7	EIK194	#16 Gauge Pin
66102-7	EIK194	#22 Gauge Pin



10 Pin Metal Style



14 Pin Plastic Style

Figure 11-3: Unidex 400 Motor Output Receptacles

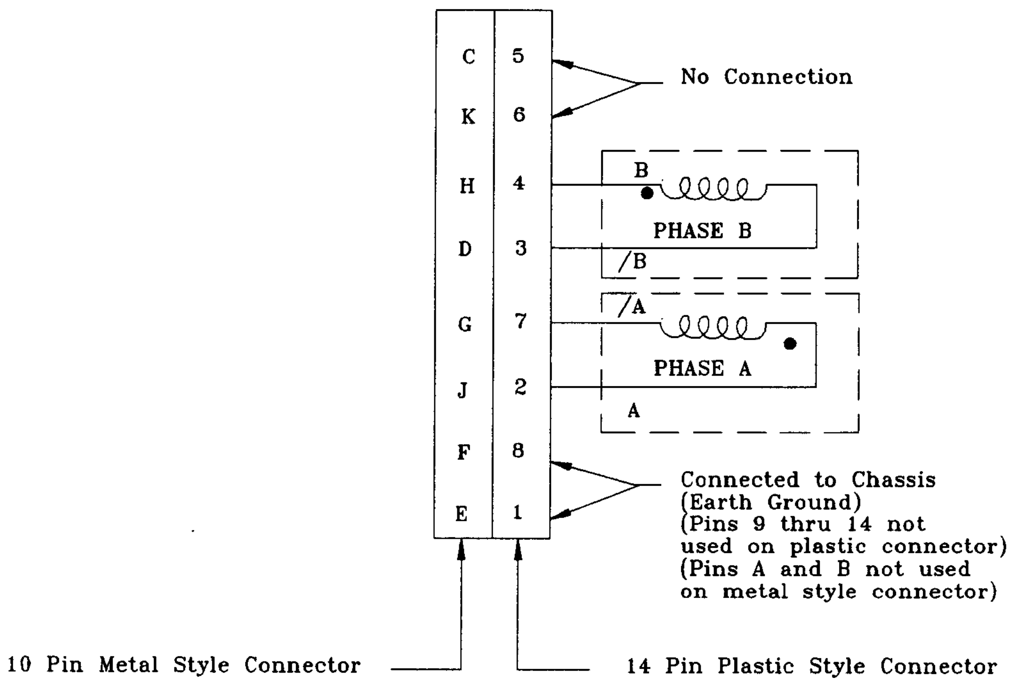


Figure 11-4: Outline of Connections for Bipolar Stepping Motor

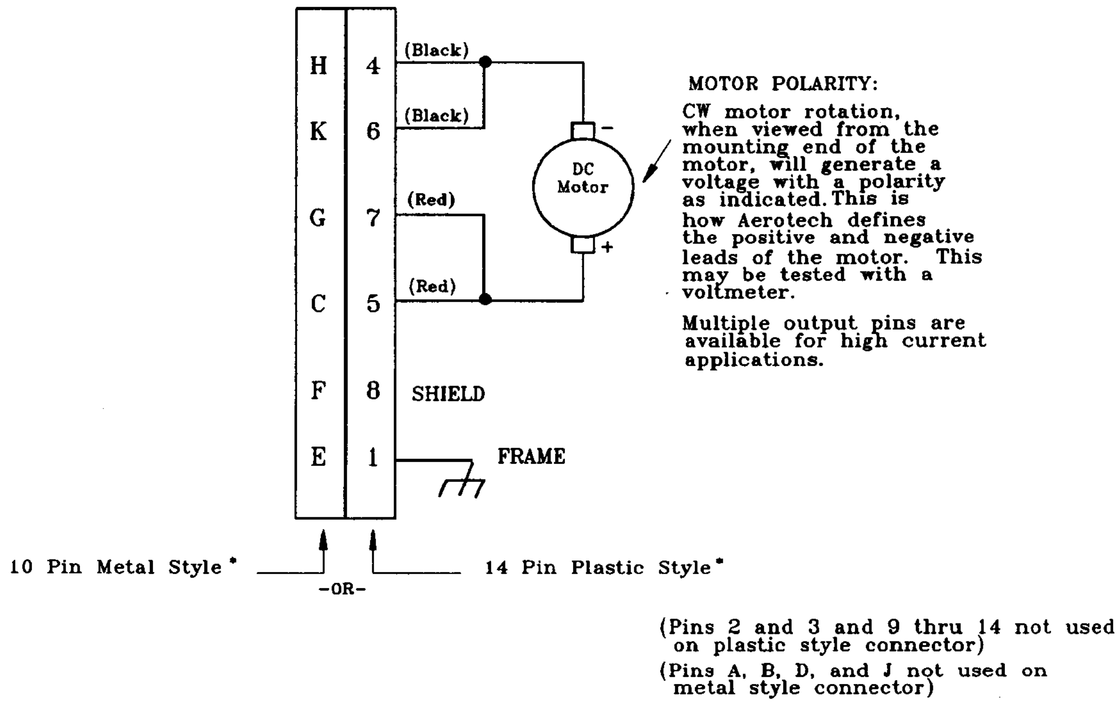


Figure 11-5: Outline of Connections for "Brush" Type DC Servo Motor

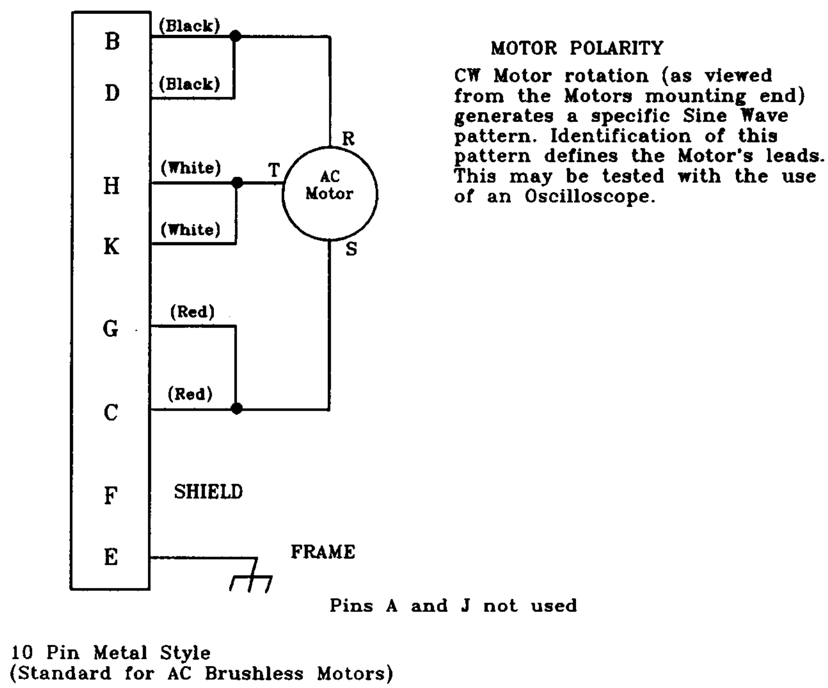


Figure 11-6: Outline of Connections for "Brushless" Type AC Motor

11-1-3 MOTOR PHASING

BRUSH TYPE DC SERVO MOTORS

The motor is commanded to turn by the Servo Amplifier. The turning of the motor generates a feedback voltage with an amplitude proportional to speed and a polarity representing direction. It is important that correct polarity is observed when the motor is connected to Aerotech equipment.

WARNING: IF THE MOTOR IS NOT PROPERLY CONNECTED, INSTABILITY OF THE SERVO DRIVE SYSTEM MAY RESULT IN A RUNAWAY CONDITION.

The polarity of the motor is in reference to Clockwise (CW) rotation of the motor. This may be determined as follows:

Connect a Voltmeter to the motor's terminals or leads. Turn the motor shaft in a Clockwise (CW) direction. A positive indication will occur while the motor shaft is being turned, if the correct polarity has been identified. If no indication or a negative indication is noted, the polarity of the motor leads has not been identified correctly and the motor connections must be reversed. Refer to Figure 11-7.

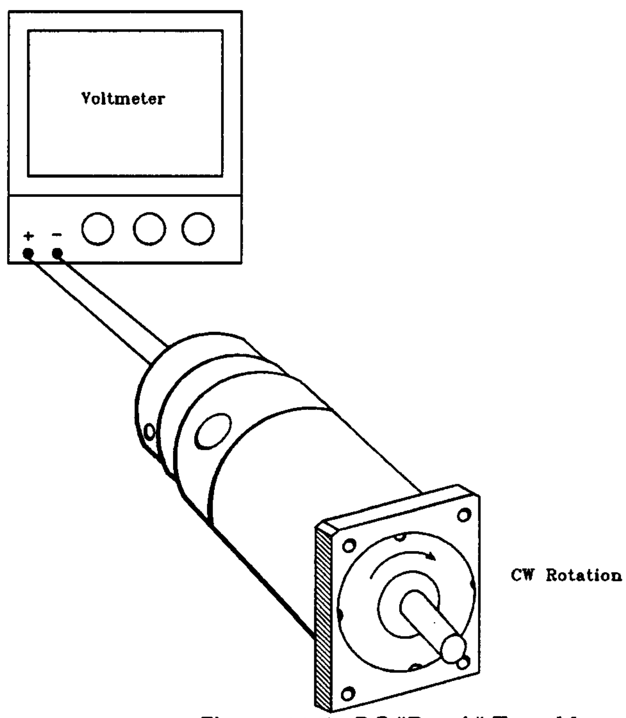


Figure 11-7: DC "Brush" Type Motor Phasing

AC BRUSHLESS TYPE SERVO MOTORS

Motor polarity must also be determined for Brushless type motors. It is important that correct polarity is observed when the motor is connected to Aerotech equipment.

WARNING: IF THE MOTOR IS NOT PROPERLY CONNECTED, INSTABILITY OF THE SERVO DRIVE SYSTEM MAY RESULT IN A RUNAWAY CONDITION.

The polarity of the motor is in reference to Clockwise (CW) rotation of the motor. This may be determined as follows:

Connect an Oscilloscope to two of the Motor's leads. Turn the motor shaft in a Clockwise (CW) direction. The Oscilloscope indication should be as illustrated in Figure 11-7 (for motors equipped with Hall sensors) or Figure 11-8 (for motors equipped with Resolvers or Encoders). If the Oscilloscope indication is not in accordance with Figure 11-8 or 11-9, or no indication is noted, the polarity of the motor leads has not been identified correctly and the motor connections must be changed.

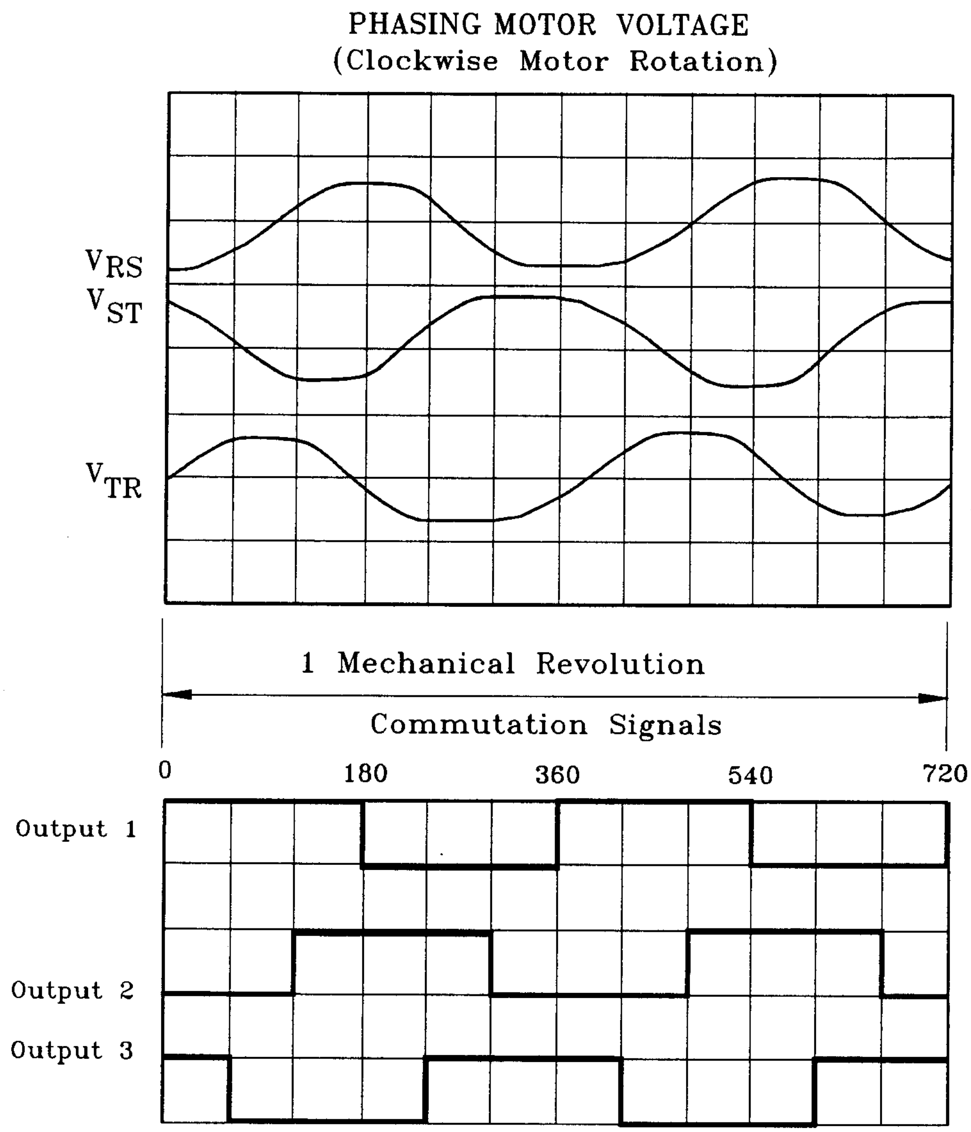


Figure 11-8: "Brushless" Motor Equipped With Hall Sensors

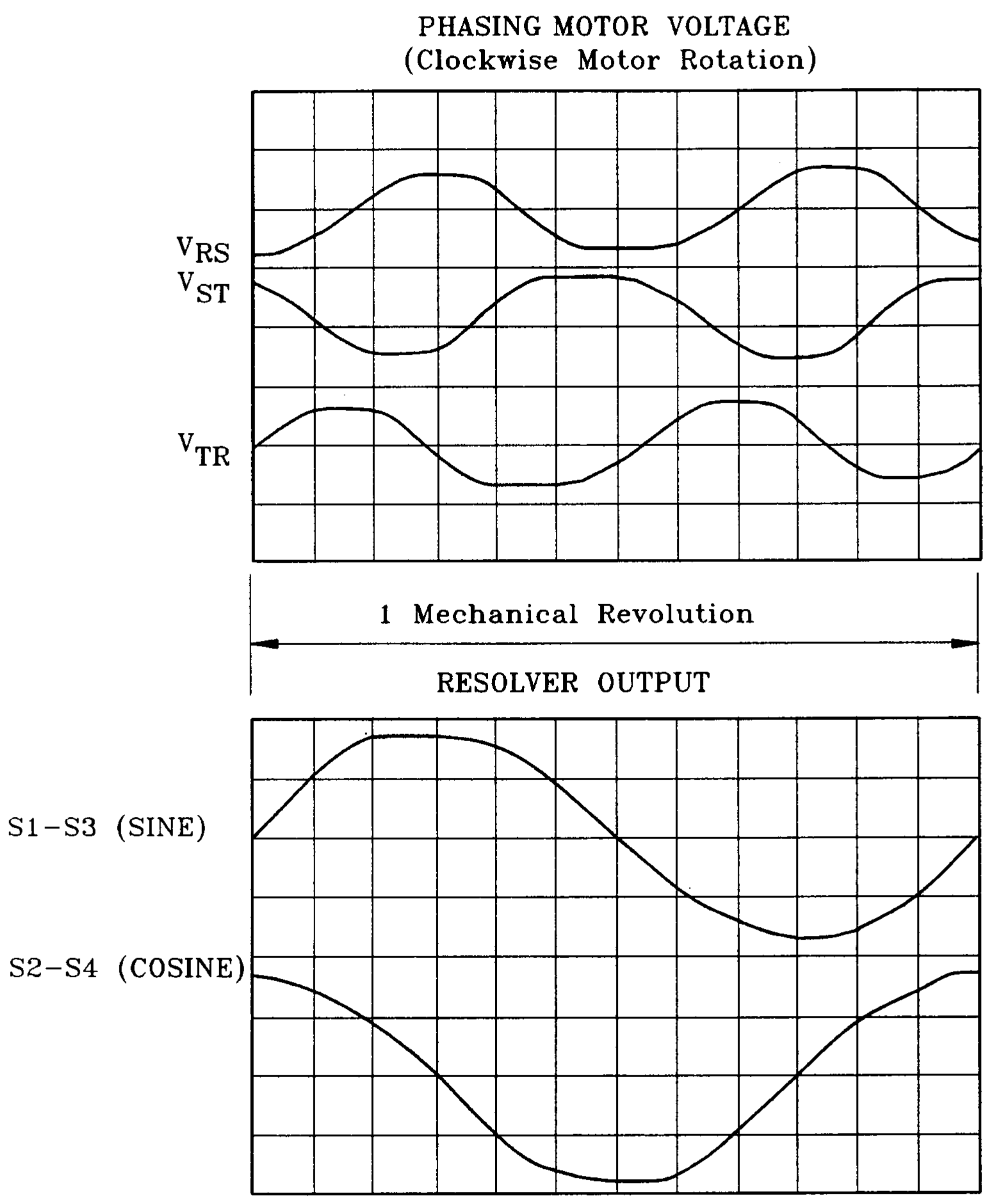
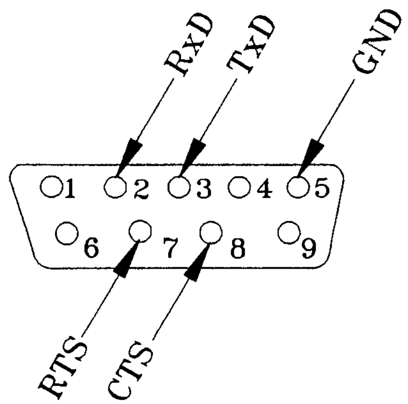


Figure 11-9: "Brushless" Motor Equipped With Resolver or Encoder

11-1-4 PORT A CONNECTOR (P30)

This port is an RS-232, +/-12V serial interface that uses the software protocol XON/XOFF or hardware signals RTS/CTS. This port is used for file transfers, remote mode, data transfers and messages. The Port A setup parameters are established in the Setup Mode, General Parameter section. The Unidex 400's PORT A is configured as a DCE.

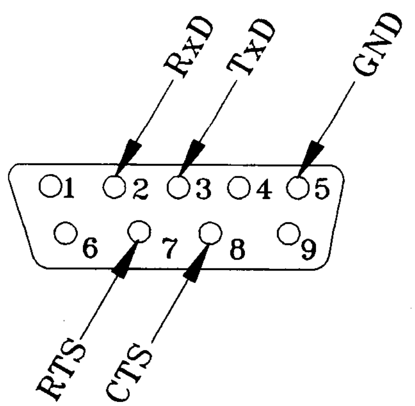


Rear Panel Connector Pin #	IBM-PC 9 Pin	IBM-PC 25 Pin	Description
1,4,6,9			(Not Used)
2	2	3	Receive Data, Unidex 400 Output, transfers data to an RS-232 device
3	3	2	Transmit Data, Unidex 400 Input, accepts input data from an RS-232 device
5	5	7	Ground
7 8	6 8 4	6 5 20	

Figure 11-10: PORT A Connector (P30)

11-1-5 TERMINAL CONNECTOR (P21)

This port is an RS-232, +/-12V serial interface that uses the software protocol XON/XOFF. This port is used to interface a Televideo 905 video display terminal when the Unidex 400 is not equipped with the Front Panel option. Protocol is fixed at a 9600 baud rate, 8 data bits, no parity, and 1 stop bit. The Unidex 400's TERMINAL Port is configured as a DCE.



The following table provides an illustration of a typical connection to a Terminal having a 25 pin, D-type, DTE connector.

TERMINAL Port Pin Number (DCE)	Terminal Pin Number (DTE)	Description
2	3	Receive Data, Unidex 400 Output, transfers data to an RS-232 device
3	2	Transmit Data, Unidex 400 Input, accepts input data from an RS-232 device
5	7	Ground
	5	Clear to Send
	6	Data Set Ready
	20	Data Terminal Ready

Figure 11-11: TERMINAL Connector (P21)

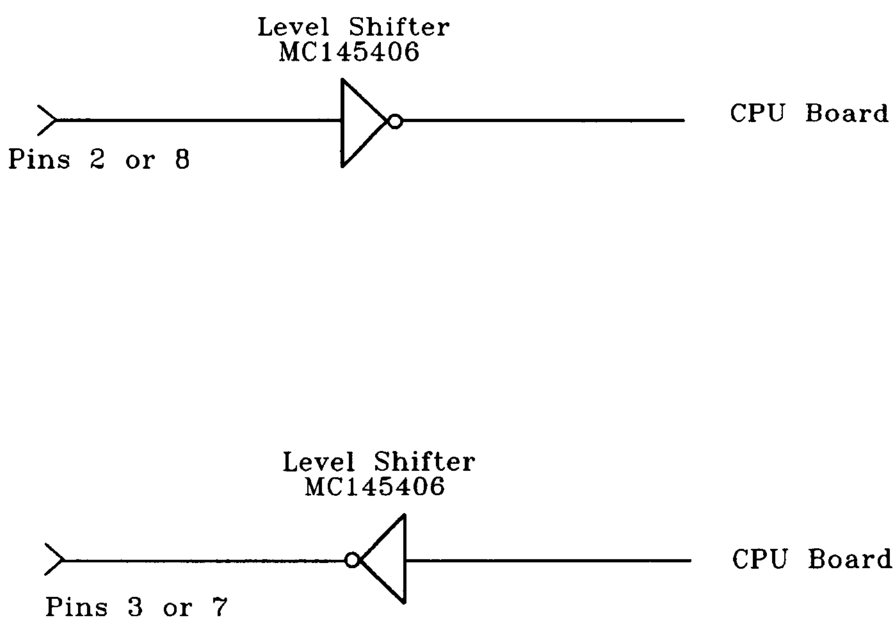
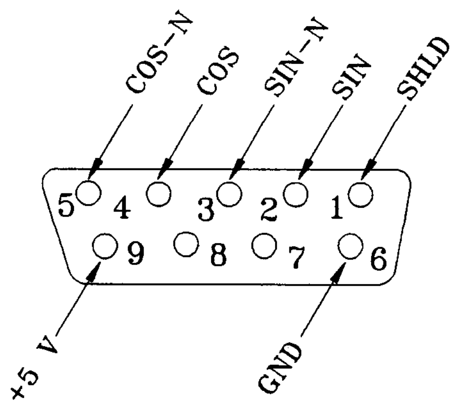


Figure 11-12: Electrical Characteristics of PORT A and TERMINAL Connectors (P30, P21)

11-1-6 WHEEL 1/WHEEL 2 CONNECTORS (P40/P41)

These ports are used to interface an optional Unidex 400 Handwheel. The signals for this connector are RS422 differential line receiver inputs.



Pin	Description
1	Shield Connection (Sent to Chassis Ground unless otherwise specified)
2	Sine Phase Input
3	Inverse Sine Phase Input
4	Cosine Phase Input
5	Inverse Cosine Input
6	Ground
7,8	(Not Used)
9	+5 V Input Common

Figure 11-13: WHEEL 1 and WHEEL 2 Connector (P40, P41)

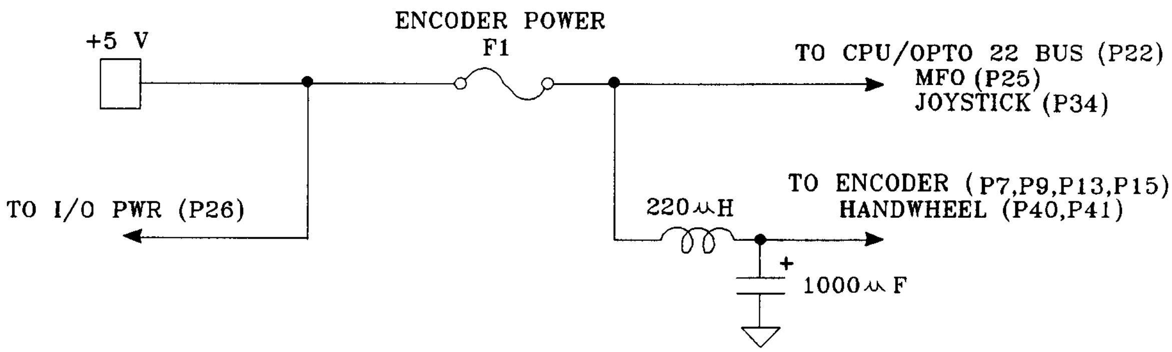
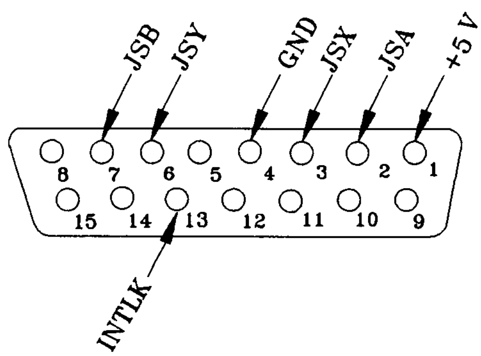


Figure 11-14 : Electrical Characteristics of WHEEL1 and WHEEL2 Connectors (P40, P41)

11-1-7 JOYSTICK CONNECTOR (P34)

The Joystick connector is used to interface an optional Unidex 400 Joystick. The Joystick's Interlock signal (Pin 13) must be sent low to indicate to the Unidex 400 an active Joystick. Figure 11-16 illustrates the electrical characteristics of this connector.



Pin	Description
1	+5V Input Common
2	Axis grouping signal from Joystick push-button "A"
3	Axis 1 Velocity Control signal (as designated by Joystick push-button "A")
4	Ground
5,8,9,10,11,12,14,15	(Not Used)
6	Axis 2 Velocity Control signal (as designated by Joystick push-button "A")
7	Joystick/Axis speed ratio signal from Joystick push- button "B"
13	Interlock signal read upon power up to indicate an active Joystick

Figure 11-15: JOYSTICK Connector (P34)

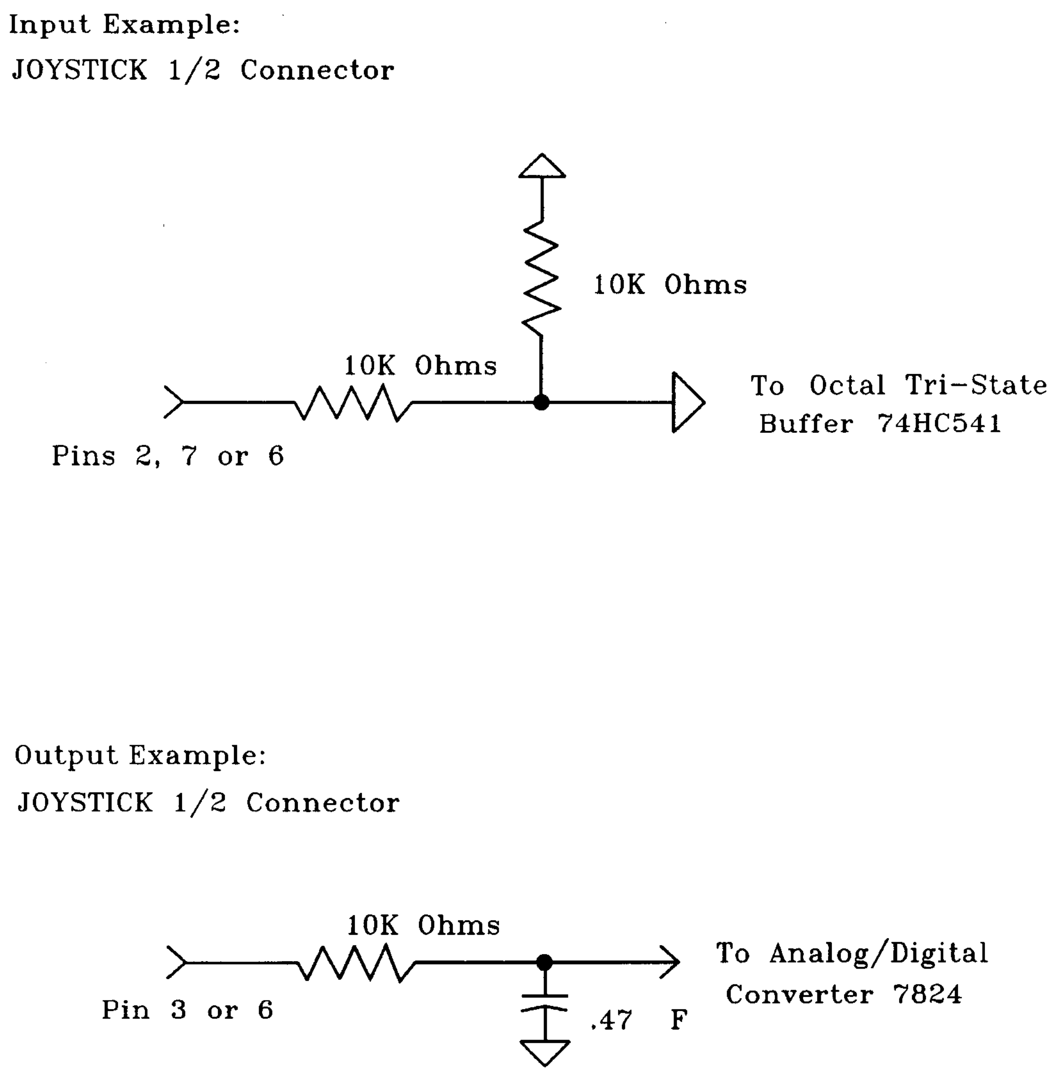
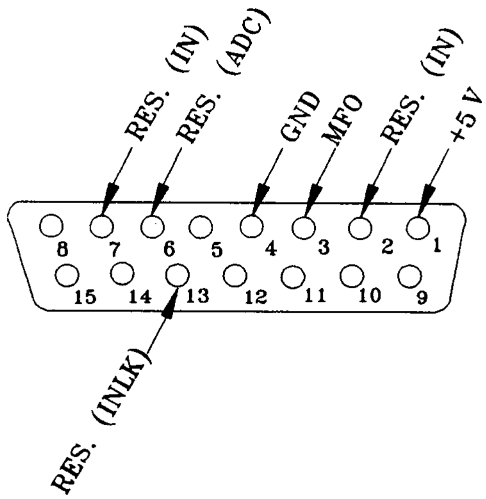


Figure 11-16 : Electrical Characteristics of the JOYSTICK Connector (P34)

11-1-8 MFO CONNECTOR (P25)

The MFO connector is used to interface to an external MFO Potentiometer. Figure 11-18 illustrates the electrical characteristics of this connector.



Pin	Description
1	+5V Input Common
2	Reserved Input
3	MFO Wiper Input
4	Ground
5,8,9,10,11,12,14,15	(Not Used)
6	Reserved A/D Converter
7	Reserved Input
13	Reserved Interlock

Figure 11-17: MFO Connector (P25)

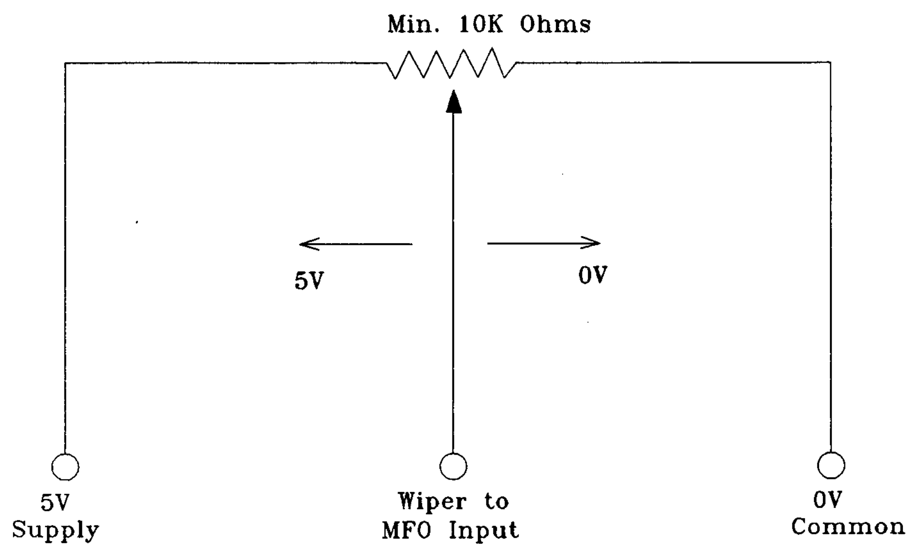


Figure 11-18: Electrical Characteristics of an External MFO Potentiometer

11-1-9: IEEE-488 INTERFACE

The IEEE-488 interface connector (P11) permits a host computer to interactively control a Unidex 400.

IEEE-488 contains 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 Bus Controller.

Details of the IEEE-488 connector are shown in Figure 11-19. Configurations for multiple device connections to the IEEE-488 Interface are illustrated in Figure 11-20.

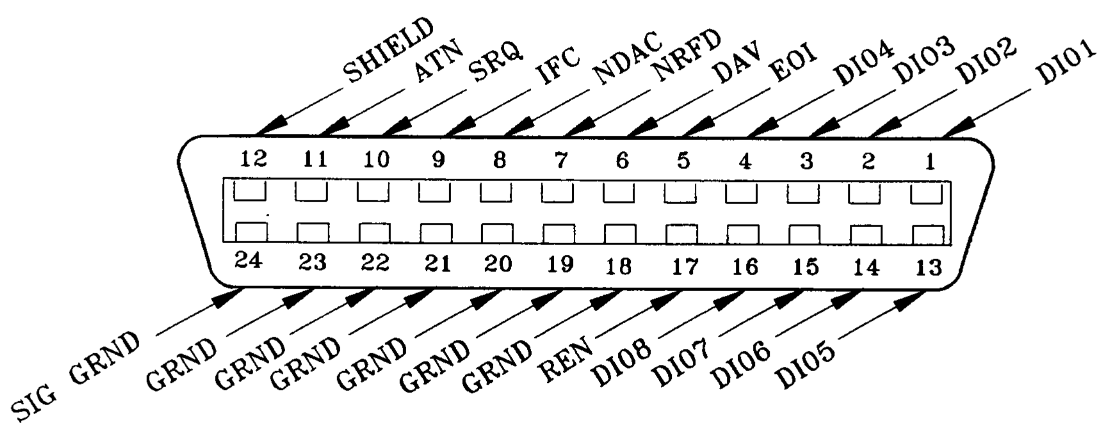


Figure 11-19: IEEE-488 Connector (P11)

PIN	DESCRIPTION
1,2,3,4 13,14,15,16	Used for sending data (ATN lines false) or bus interface messages (ATN line true)
5	End or Identify, once asserted indicates the termination of data flow. Asserted when the last data byte is placed on the Bus.
6	Data Valid, asserted by the Talker to indicate to all Listeners that data on the Bus is valid.
7	Not Ready For Data, when true, indicates to Talker that all Listeners are not ready for data.
8	Not Data Accepted, when true, indicates to the Talker that all Listeners have not accepted the data placed on the Bus.
9	Interface Clear, asserted only by the System Controller to place all devices in the unaddressed state. Devices go into the Talker idle, Listener state. If control has been passed to another device, the System Controller again becomes active by asserting IFC.
10	Service Request, asserted by a device to indicate a need for interaction with the Controller.
11	Attention, asserted true by active Controller to send Bus interface messages. When ATN is asserted, signals on the data lines are interpreted as messages. ATN asserted with EOI does a Parallel Poll. When ATN is false, data may be sent over the Bus by a designated Talker.

PIN	DESCRIPTION
12	Shield
17	Remote Enable, remotely asserted to program devices on the Bus. Any device addressed to listen while REN is true, is placed in the Remote Mode of operation.
18	Ground (Twisted Pair with DAV)
19	Ground (Twisted Pair with NRFD)
20	Ground (Twisted Pair with NDAC)
21	Ground (Twisted Pair with IFC)
22	Ground (Twisted Pair with SRQ)
23	Ground (Twisted Pair with ATN)
24	Signal Ground

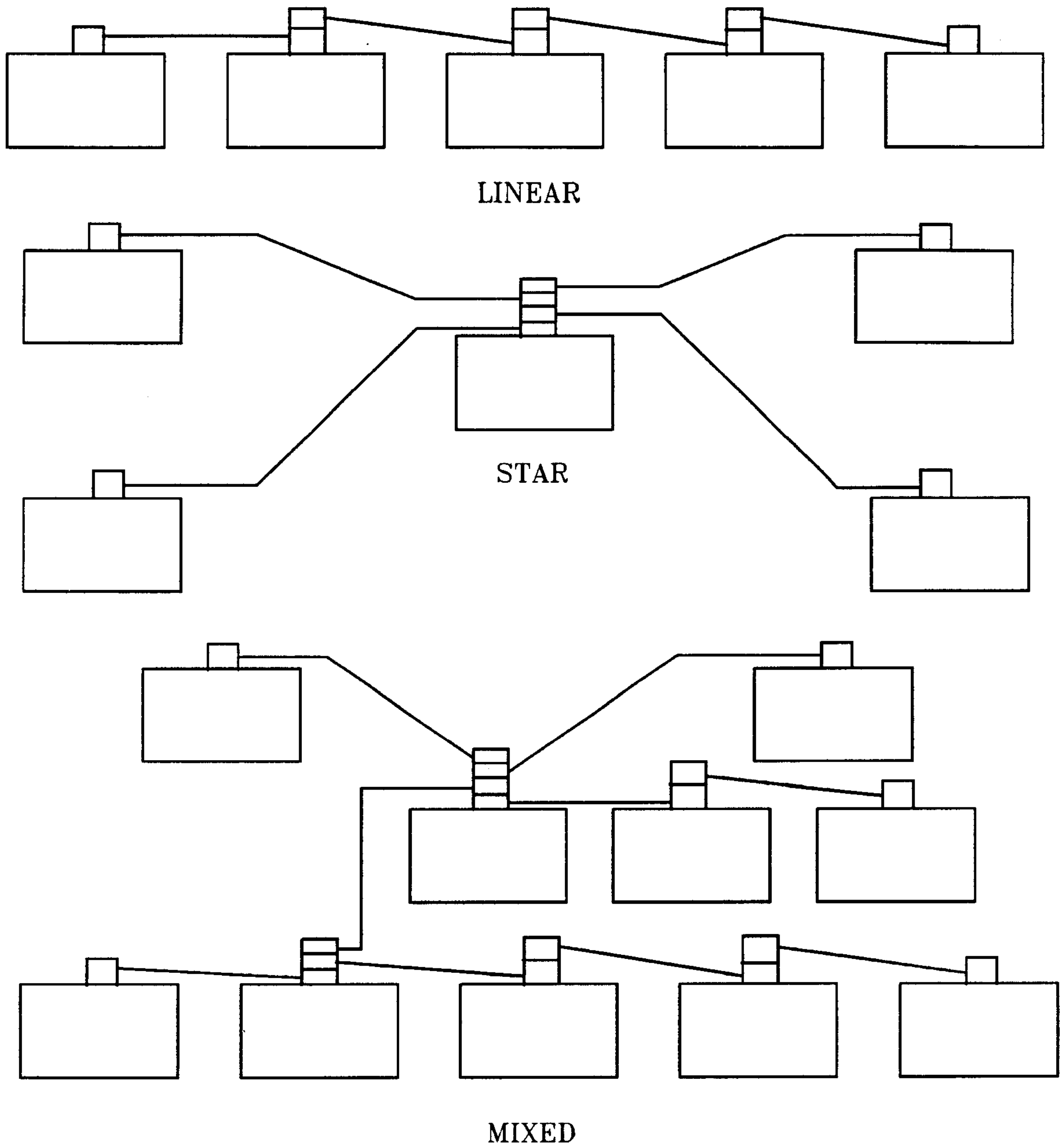
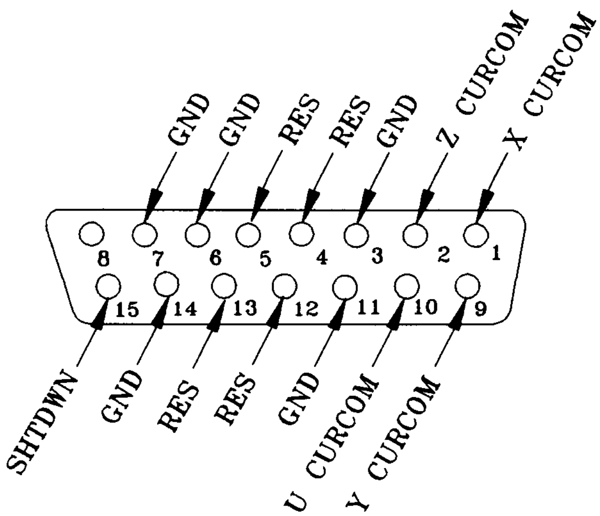


Figure 11-20: IEEE-488 Cabling Configurations

11-1-10 DRIVE INTERFACE CONNECTOR (P48)

The DRIVE INT Connector (P48) provides the current command signals for the X, Y, Z, and U axes of the Unidex 400. The signals are bipolar 8 volt signals with 4mV quantization (12 bit Digital Analog Converter). The minimum load on the signals is 2K ohms The Shutdown signal is a 7407 type Open Collector output.

Details of the DRIVE INT connector are shown in Figure 11-21. Electrical characteristics of this connector are illustrated in Figure 11-22.



PIN	DESCRIPTION
1	Output Current Command Signal for the X axis.
2	Output Current Command Signal for the Z axis.
3,6,7,11,14	Ground
4,5,12,13	Reserved
8	(Not Used)
9	Output Current Command Signal for the Y axis.
10	Output Current Command Signal for the U axis.
15	Open Collector Shutdown Output

Figure 11-21: DRIVE INT Connector (P48)

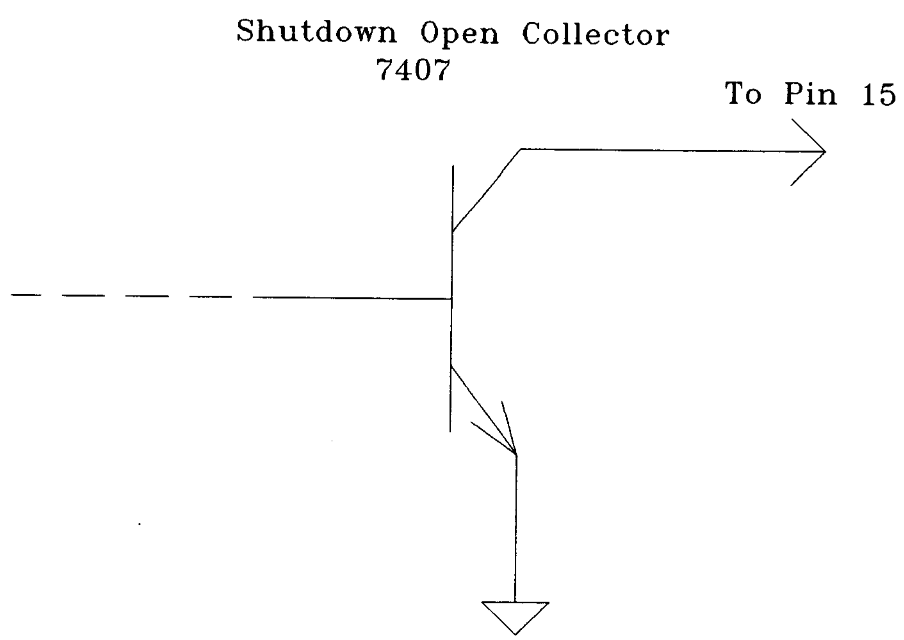


Figure 11-22: Electrical Characteristics of the DRIVE INT Connector (P48)

11-1-11 MISC I/O CONNECTOR (P23)

The Miscellaneous I/O Connector contains the interface for four User Interrupts, Brake and Brake monitor, two High Speed Opto Isolated Servo Interrupts, Position Grabbing, Emergency Stop and Fast Feedhold. Details of the MISC I/O connector are shown in Figure 11-23.

The four User Interrupts are opto isolated using a 2401-4 type passive isolator. The maximum forward current is 16mA. An internal current limiting resistance of 390 ohms is provided within the Unidex 400. An external resistance must be added when using higher voltage systems. Refer to Figure 11-24.

The Brake signal is a 7407 open collector output capable of sinking 40mA of current. The Brake monitor is a CMOS level input and is not isolated. The Servo Interrupts are isolated through a 6N136 type opto isolator. An internal current limiting resistor of 290 ohms is provided, making these signals 5 volt compatible.

Electrical characteristics of the connector are shown in Figure 11-25.

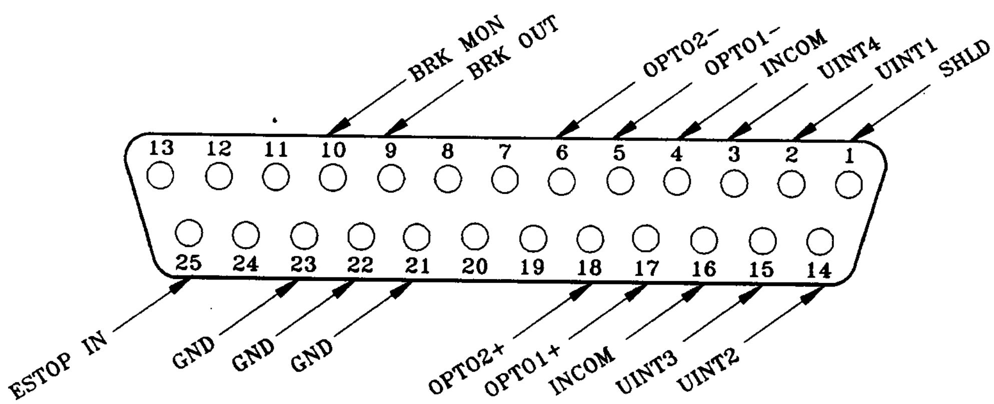
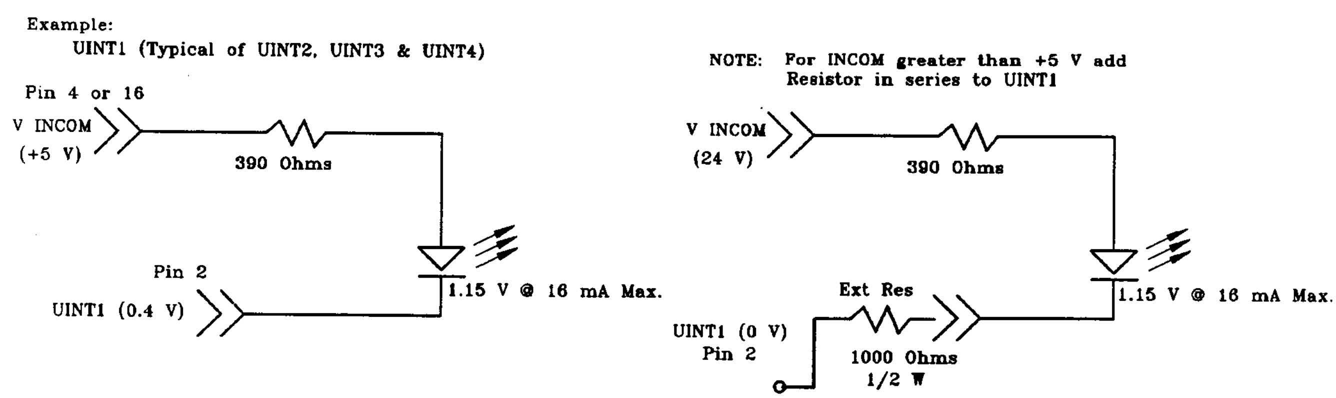


Figure 11-23: MISC I/O Connector (P23)

PIN	DESCRIPTION
1	Shield (Frame Ground)
2	User Interrupt 1 Input to CPU Board. Activates an established Interrupt routine.
3	User Interrupt 4 Input to CPU Board. Activates an established Interrupt routine.
4	Input Common, used in conjunction with High Speed Interrupt and User Interrupts (IN COM is shared by all CPU Inputs and Interrupts).
5	Fast Feedhold Negative Opto 1 Interrupt Input to DSP
6	Position Grabbing, Negative Opto 2 Interrupt Input to DSP
7,8,11,12,13,19,20,24	(Not Used)
9	DSP Output activates Brake In Fault condition.
10	Monitors Brake Interlock, provides DSP Input to activate shutdown if Brake not connected.
14	User Interrupt 2 Input to CPU Board. Activates an established Interrupt routine.
15	User Interrupt 3 Input to CPU Board. Activates an established Interrupt routine.
16	Input Common, used in conjunction with High Speed Interrupt and User Interrupts 1 and 2 (IN COM is shared by all CPU Inputs and Interrupts).
17	Fast Feedhold Positive Opto 1 Interrupt Input to DSP
18	Position Grabbing, Positive Opto 2 Interrupt Input to DSP
21,22,23	Ground
25	Emergency Stop Input. This pin is internally pulled to a high level to indicate an Emergency Stop condition. Refer to General Parameter 037.

UINT1, UINT2, UINT3, UINT4 (PS2401A4 Opto Coupler)



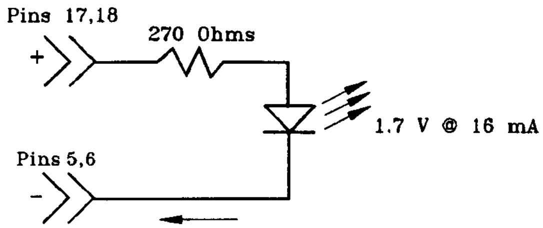
External Resistor >User V − 1.15 − 0.390 x 16)/16

Example:

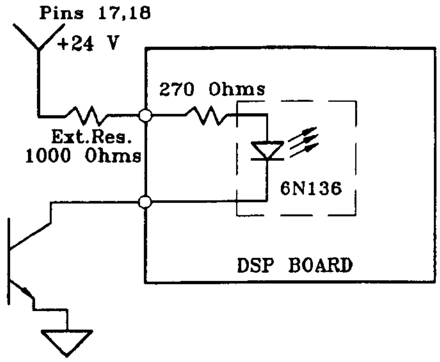
User Voltage	External Resistor Value (Ohms)
5	0
12	290 1/4 watt
24	1000 1/2 watt

Figure 11-24: Electrical Characteristics of the MISC I/O Ccnnector (P23) - Opto1&2, Brake Signals

OPTO1, OPTO2 (6N136 Opto Coupler)



Typical Application:

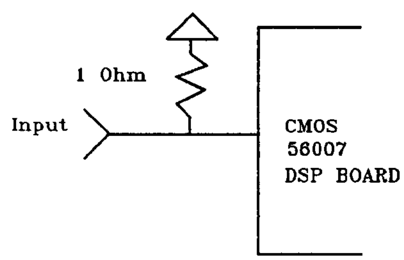


External Resistor >User V − 1.7 − 0.270 x 16)/16

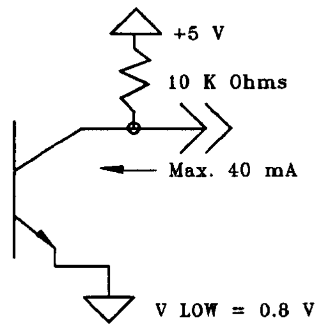
Example:

User Voltage	External Resistor Value (Ohms)
5	0
12	350 1/4 watt
24	1000 1/2 watt

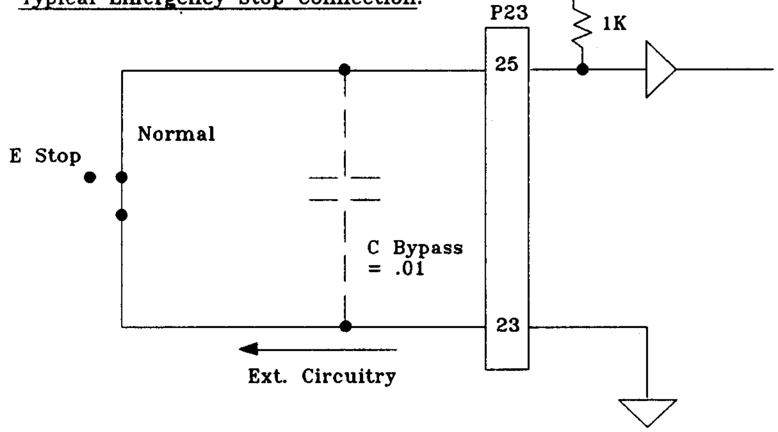
BRAKE MONITOR



BRAKE OUTPUT
(SN7407 Open Collector TTL)



Typical Emergency Stop Connection:



A Bypass Capacitor may be added in noisy environments.

Figure 11-25: Electrical Characteristics of the MISC I/O Connector (P23) - User Interrupts

11-1-12 ENCODER/LIMIT CONNECTOR

The optically isolated Limit and Home Switch Inputs are interfaced to the Unidex 400 through 25 pin D-type ENC/LMT Connectors (P7, P9, P13, and P15).

An ENC/LMT Connector is provided for each of the four axes. Details of a typical connector are shown in Figure 11-26. Electrical characteristics of a typical Limit or Home Switch Input are shown in Figure 11-27. Electrical characteristics of the sensor signals are shown in Figure 11-28.

Aerotech supplied Limit and Home Switches are of the Normally Open, Logic 0 type (Refers to Contact positions when not in Limit). If Normally Closed, Logic 1, Limit and/or Home Switches are used , the Axis Parameter "4: Home Switch Normally Open?" and /or "23: Limit Switch Normally Open", must be configured accordingly. (See Chapter 7: The Setup Mode.)

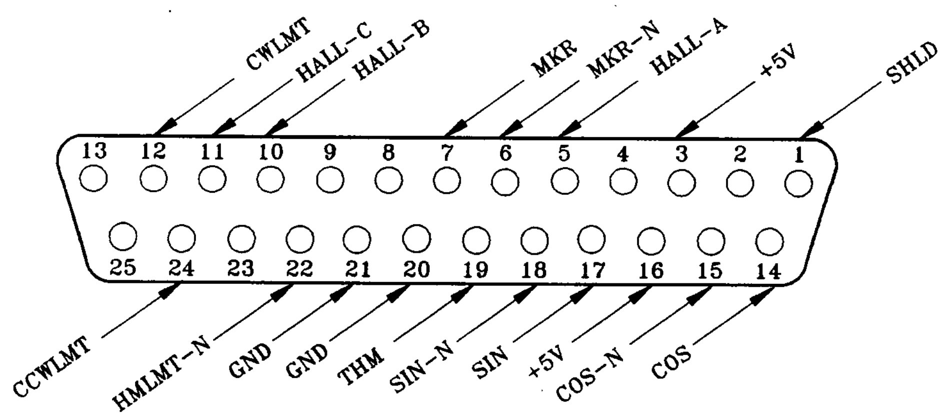


Figure 11-26: ENC/LMT Connector (P7,P9,P13,P15)

PIN	DESCRIPTION
1	Encoder Shield connection. Sent to Chassis Ground unless otherwise specified.
2,4,8,9,13,23,25	(Not Used)
3	+5 V Input Common (Fused on Rear Panel)
5	Hall Effect Switch "A" signal feedback to the DSP Board for Motor commutation (Active only when using a Brushless type Motor)
6	Inverse Marker pulse feedback from the Encoder to the DSP Board (RS-422)
7	Positive Marker pulse feedback from the Encoder to the DSP Board (RS-422)
10	Hall Effect Switch "B" signal feedback to the DSP Board for Motor commutation (Active only when using a Brushless type Motor)
11	Hall Effect Switch signal "C" feedback to the DSP Board for Motor commutation (Active only when using a Brushless type Motor)
12	CW * Limit Switch Input Signal to the DSP Board
14	Positive Cosine feedback from the Encoder to the DSP Board (RS-422)
15	Inverse Cosine feedback from the Encoder to the DSP Board (RS-422)
16	+5 V Input Common (Fused on Rear Panel)
17	Positive Sine feedback from the Encoder to the DSP Board (RS-422)
18	Inverse Sine feedback from the Encoder to the DSP Board (RS-422)
19	Motor Thermistor feedback providing motor temperature data to the R/D Board (Active only when using a Brushless type Motor)
20,21	Ground, common to all Encoder connectors
22	Home Limit Switch Input signal to the DSP Board
24	CCW * Limit Switch Input Signal to the DSP Board

* Refer to page 11-35 for Limit Switch/Direction relationship

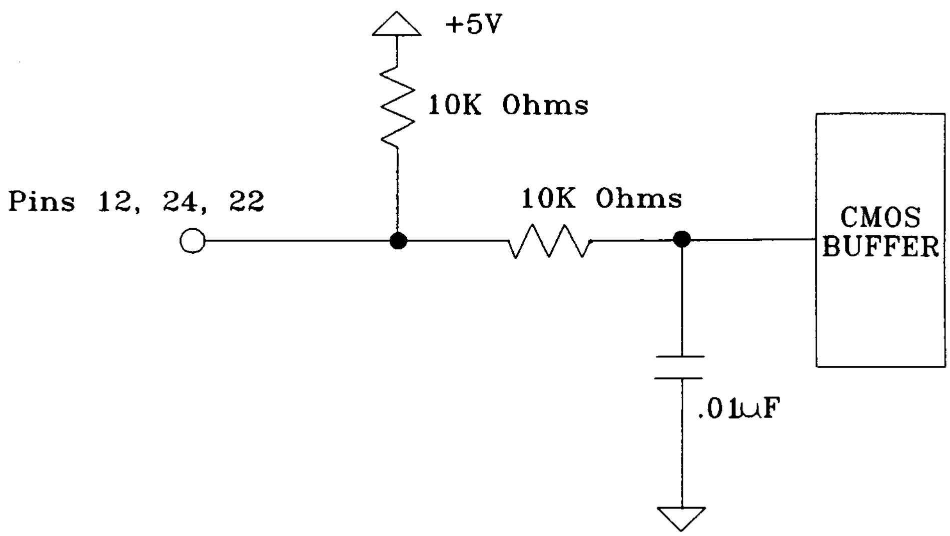


Figure 11-27: Electrical Characteristics of the Encoder and Limit Signals

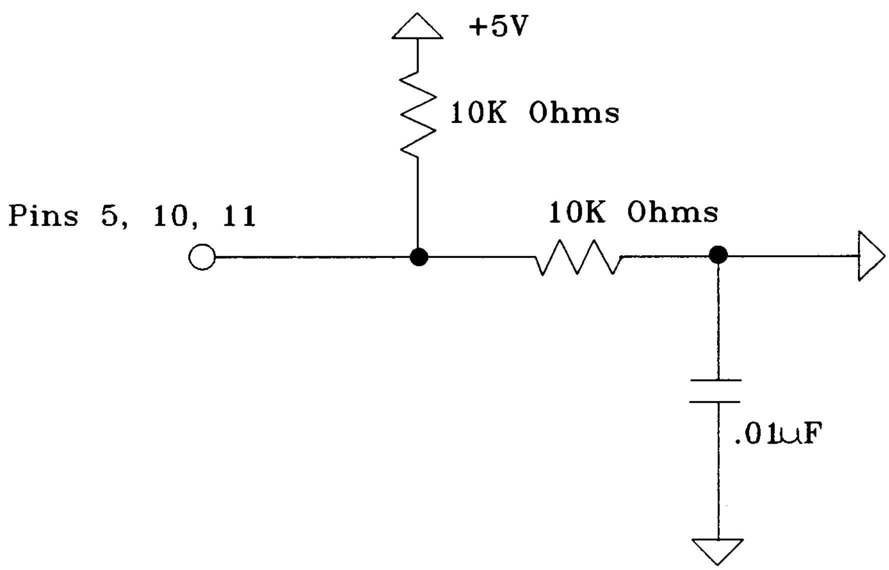


Figure 11-28: Electrical Characteristics of the Hall Effect and Thermal Sensors

LIMIT SWITCH DIRECTION

The CW and CCW Limit Switches provide travel limit protection for the Unidex 21 Drive System. Clockwise (CW) and Counterclockwise (CCW) motor rotation are referenced as viewed from the mounting flange of the motor. Aerotech has established this standard to clarify the Limit direction when one controller is interfaced to several mechanical configurations. The CW Limit stops CW motor rotation and the CCW Limit stops CCW rotation. (See Figure 11-29.)

Aerotech tables are manufactured such that the CW and CCW Limit Switches precede the rubber mechanical stops by approximately one revolution of the ballscrew. It is not recommended that the table be allowed to come into contact with the mechanical stops.

WARNING: DAMAGE TO THE EQUIPMENT MAY OCCUR IF THE TABLE IS ALLOWED TO COME INTO CONTACT WITH THE MECHANICAL STOPS.

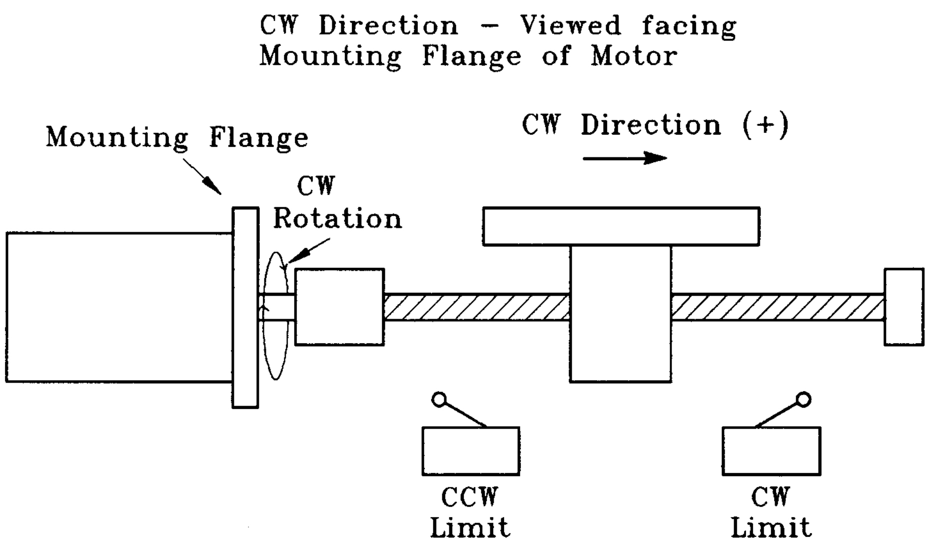


Figure 11-29: Limit Switch Direction

HOME SWITCH POSITION

Most Aerotech Control systems are equipped with a cold-start reference point, which Aerotech refers to as the Home Position. This is the Machine Home and is a constant position depending on the marker location of a rotary or linear encoder.

When a Home command is issued, the motor will turn CCW (default) or CW (optional) until Home Limit Switch activation occurs, establishing the Home reference point. (Refer to the Setup Mode for parameters related to Home Limit Switch positions.) Upon Home Limit switch activation, the motor will reverse and rotate in the opposite direction until the Marker is encountered .

For most Rotary motion stages the Home Limit Switch is an independent switch and cam, incorporated into the system specifically for the Home cycle. When using a Linear motion stage, the CW or CCW Limit Switches may double as the Home Limit Switch.

Regardless of whether the Home reference point is established by a CW, CCW or Home Limit Switch, it's optimum position should be 1/2 revolution (± 1/4 revolution) from the location that the Switch opens (for a Normally Open type switch) to the leading edge of the marker pulse. (See Figure 11-30.)

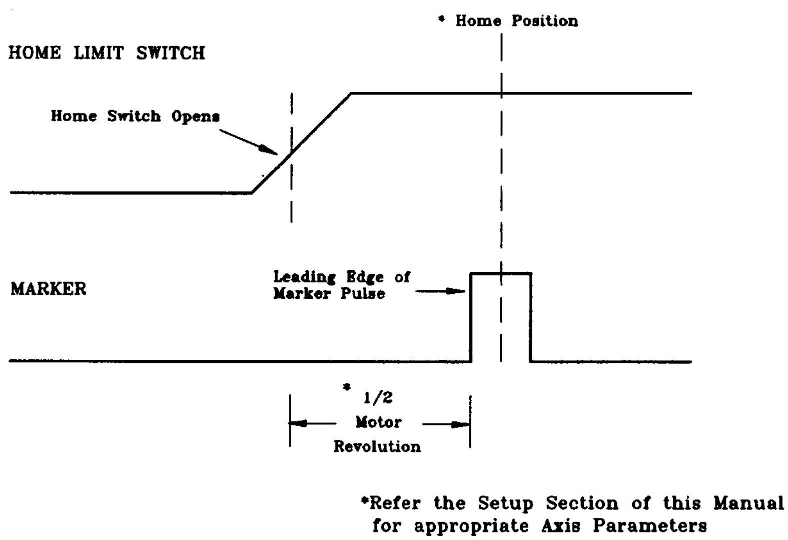


Figure 11-30: Home Switch Position

11-1-13 I/O POWER CONNECTOR (P26)

The I/O POWER connector provides an auxiliary power source for external devices such as the Aerotech LMI Board. This connector is to be used for low power applications only.

This power source should not be used to power opto-isolators because isolation will be lost. Always use an external, transformer isolated power supply when interfacing to opto-isolators.

WARNING: **THE I/O POWER CONNECTOR MUST NOT BE USED TO POWER OPTO-ISOLATED DEVICES BECAUSE ISOLATION WILL BE LOST.**

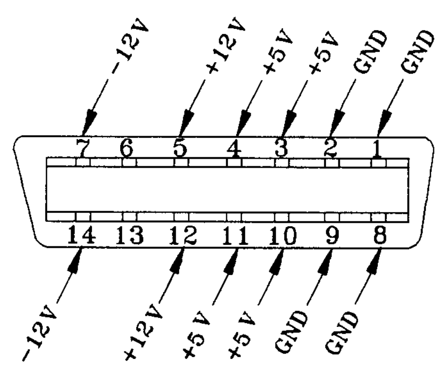


Figure 11-31: I/O POWER Connector (P26)

11-1-14: CPU/OPTO 22 BUS CONNECTOR (P22)

The CPU/OPTO 22 Connector is a 50 pin Champ connector providing 8 outputs and 12 inputs. The signals are non-opto-isolated TTL level signals, using 74F244/245 type drivers.

When used with the OPTO 22 I/O mounting rack assemblies (or equivalent) the Model OPC interconnection cable (standard 5 ft. length) provides a direct mating between the 50-pin Champ style connector and the edge connector on the OPTO board. Opto 22 Models PB8, PB16 and PB24 (or equivalent) may be utilized to provide either 8 outputs, 8 outputs/8 inputs, or 8 outputs/12 inputs with OPC interconnection. Other combinations are available with special cabling.

Details of the connector are shown in Figure 11-32.

Table 11-1 lists the standard I/O interconnection between the Unidex 21 and the I/O mounting rack assemblies. Figure 11-33 illustrates OPTO 22 I/O mounting rack assembly wiring.

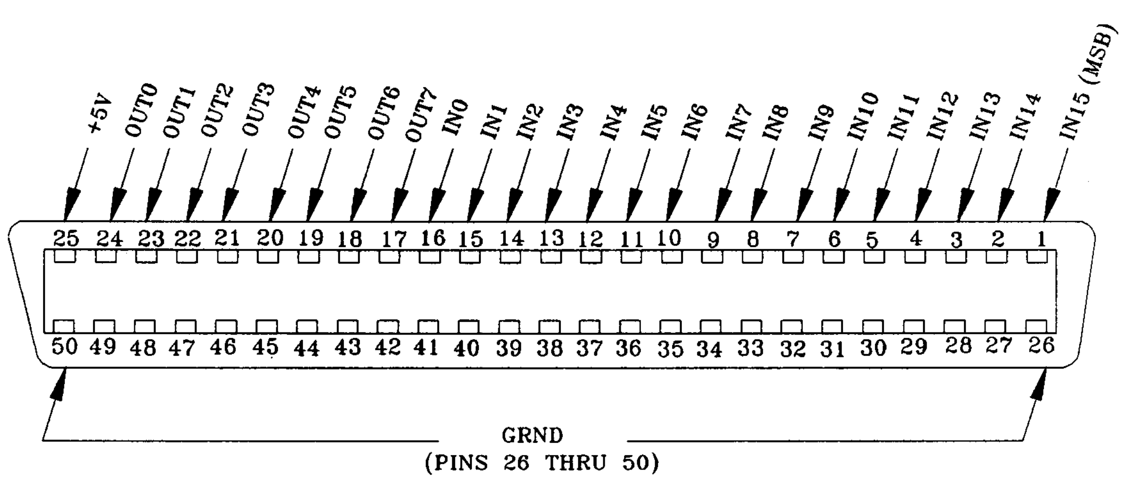


Figure 11-32: CPU/OPTO 22 Bus Connector

PIN	DESCRIPTION
1	Data Input Bit 15, Address IN15 (Most Significant Bit)
2	Data Input Bit 14, Address IN14
3	Data Input Bit 13, Address IN13
4	Data Input Bit 12, Address IN12
5	Data Input Bit 11, Address IN11
6	Data Input Bit 10, Address IN10
7	Data Input Bit 9, Address IN9
8	Data Input Bit 8, Address IN8
9	Data Input Bit 7, Address IN7
10	Data Input Bit 6, Address IN6
11	Data Input Bit 5, Address IN5
12	Data Input Bit 4, Address IN4
13	Data Input Bit 3, Address IN3
14	Data Input Bit 2, Address IN2
15	Data Input Bit 1, Address IN1
16	Data Input Bit 0, Address IN0 (Least Significant Bit)
17	Data Output Bit 7, Address OUT7 (Most Significant Bit)
18	Data Output Bit 6, Address OUT6
19	Data Output Bit 5, Address OUT5
20	Data Output Bit 4, Address OUT4
21	Data Output Bit 3, Address OUT3
22	Data Output Bit 2, Address OUT2
23	Data Output Bit 1, Address OUT1
24	Data Output Bit 0, Address OUT0 (Least Significant Bit)
25	Input Common (5V Typical)
26 thru 50	Output Common (GND. Typical)

Sample Output Command: OUT0=H,01 or OUT0=H,00

Sample Input Command: (JUMP, ENT1, IN0.EQ.H, 01)

INTERFACE CABLE ASSEMBLY (Model OPC)		PB8, PB16A, PB16C AND PB24 BOARD				
OPTO-INTERFACE (P22)	CONTROL CONNECTION (EDGE CONNECTOR ON OPTO BOARD)		MODULE POSITION	CONNECTION DESCRIPTION	TYPE OF MODULE	FIELD CONNECTION (BARRIER STRIP)
25	49			+5V internal supply		
24	47		0	OUT0	Output	1 and 2
23	45		1	OUT1	Output	3 and 4
22	43		2	OUT2	Output	5 and 6
21	41	PB8	3	OUT3	Output	7 and 8
20	39		4	OUT4	Output	9 and 10
19	37		5	OUT5	Output	11 and 12
18	35		6	OUT6	Output	13 and 14
17	33		7	OUT7	Output	15 and 16
16	31		8	IN0	Input	17 and 18
15	29		9	IN1	Input	19 and 20
14	27	PB16A	10	IN2	Input	21 and 22
13	25	AND	11	IN3	Input	23 and 24
12	23	PB16C	12	IN4	Input	25 and 26
11	21		13	IN5	Input	27 and 28
10	19		14	IN6	Input	29 and 30
9	17		15	IN7	Input	31 and 32
8	15		16	IN8	Input	33 and 34
7	13		17	IN9	Input	35 and 36
6	11		18	IN10	Input	37 and 38
5	9	PB24	19	IN11	Input	39 and 40
4	7		20	IN12	Input	41 and 42
3	5		21	IN13	Input	43 and 44
2	3		22	IN14	Input	45 and 46
1	1		23	IN15	Input	47 and 48

Pins 26 through 50 are signal common.

All even pins (2 to 50) are signal common.

WARNING: Type of module (input or output) cannot be interchanged. To do so may damage Unidex 400.

Typical Modules: IDC5, IDC5B, IAC5, IAC5A
ODC5, ODC5A, OAC5, OAC5A

Table 11-1: Interconnection Diagram of PB8, PB16A or C, or PB24 I/O Board

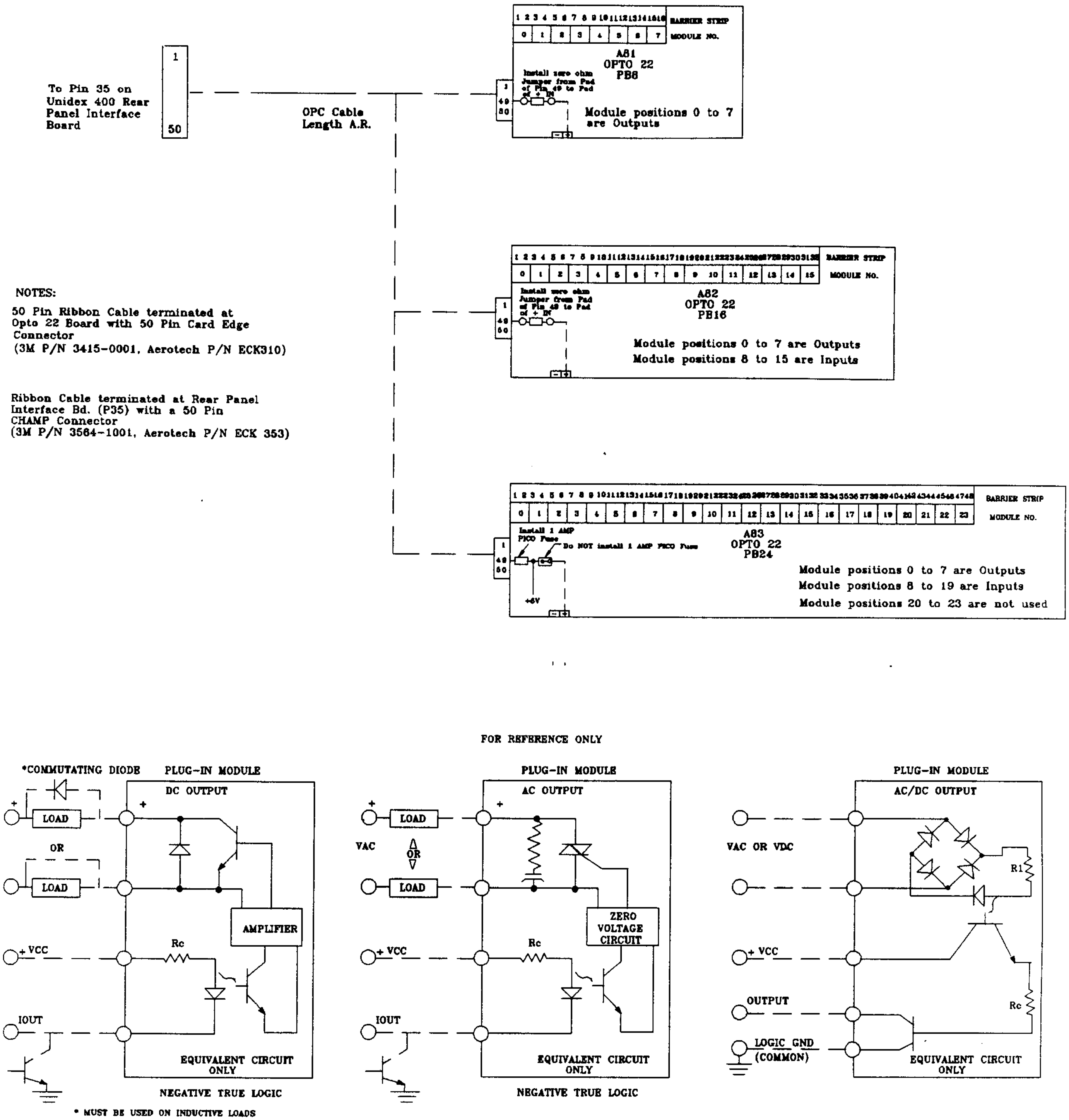
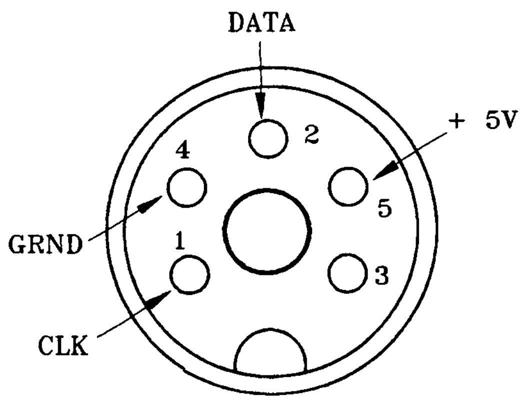


Figure 11-33: CPU I/O OPC Interconnection Diagram

11-1-15: KEYBOARD CONNECTOR

The Keyboard Connector provides an interface between the Unidex 400 and an external Keyboard. The Keyboard must be a 101-key, "Qwerty" IBM PC/AT Keyboard operated by IBM Scan Code 2 (such as Model G81-1000 HBU/04 available from Cherry Electrical Products, 3600 Sunset Ave., Waukegan, Il. 60687).

Details of the KEY BD Connector (P27) are shown in Figure 11-34. Electrical characteristics of the connector are shown in Figure 11-35.



PIN	DESCRIPTION
1	Clock
2	Data Output
3	(Not Used)
4	Ground
5	+5V Input Common

Figure 11-34: KEYBOARD Connector

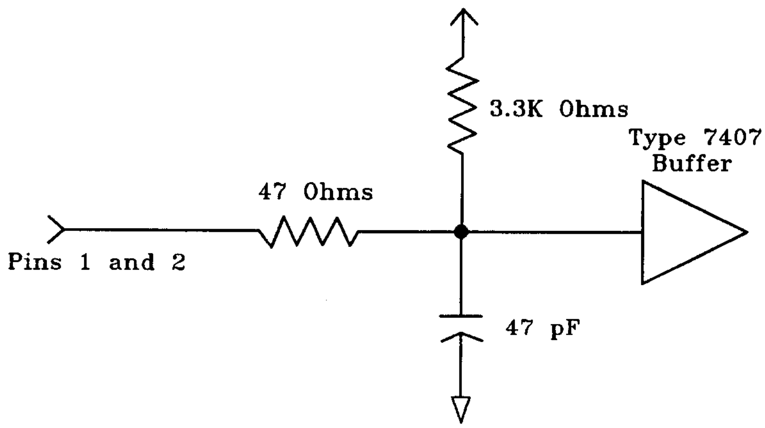


Figure 11-35: Electrical Characteristics of the KEYBOARD Connector

(THIS PAGE INTENTIONALLY LEFT BLANK)

CHAPTER 12: ERROR MESSAGES

SECTION 12-1: INTRODUCTION

During operation, the Unidex 400 may encounter error conditions. When such conditions occur, an appropriate message will appear on the display. Refer to Chapter 9 for Error Codes related to Remote operation.

The following sections contain a list of each of the Error Codes and a brief description of each.

SECTION 12-2: HARDWARE/FATAL ERRORS

These Errors may cause the system to stop processing.

Message	Source/Suggested Solution
BUS ERROR	Power Down Unidex 400, Contact Aerotech, Inc.
ADDRESS ERROR	Power Down Unidex 400, Contact Aerotech, Inc.
ILLEGAL INSTR	Power Down Unidex 400, Contact Aerotech, Inc.
ZERO DIVIDE	Power Down Unidex 400, Contact Aerotech, Inc.
LINE 1010	Power Down Unidex 400, Contact Aerotech, Inc.
LINE 1111	Power Down Unidex 400, Contact Aerotech, Inc.
UNINIT IV	Uninitialized Interrupt Power Down Unidex 400, Contact Aerotech, Inc.
SPURIOUS	Spurious Interrupt Power Down Unidex 400, Contact Aerotech, Inc.

Message	Source/Suggested Solution
LIMIT S1 MOVE OUT	A Limit Switch has been activated during a MDI, Programmed or Jog move. When in MDI or a Program, select **Abort** from the display, then press the Front Panel "**S1**" Key to move the axis out of the Limit. When in the Jog Mode, press the Front Panel "**S1**" key to move the axis out of the Limit and then continue with Jog movement.
POSITION TRAP	A Servo Loop condition has been exceeded, producing an error. This error will occur when the Servo Loop performance is undesirable (Refer to Chapter 7: The Setup Mode) or, if a component in the Drive has failed (Amplifier, Motor, Encoder or Servo Board).
VELOCITY TRAP	A Servo Loop condition has been exceeded, producing an error. This error will occur when the Servo Loop performance is undesirable (Refer to Chapter 7: The Setup Mode) or, if a component in the Drive has failed (Amplifier, Motor, Encoder or Servo Board).
INTEGRAL TRAP	A Servo Loop condition has been exceeded, producing an error. This error will occur when the Servo Loop performance is undesirable (Refer to Chapter 7: The Setup Mode) or, if a component in the Drive has failed (Amplifier, Motor, Encoder or Servo Board).
THERMAL TRAP	This error will occur when a Thermal Sensor has detected an overtemperature situation (Systems equipped with an RDP Board only).

Message	Source/Suggested Solution
FEEDRATE TRAP	A Feedrate, greater than the Top Feedrate (as established by Axis Parameter x08) has been attempted.
RESET THE SYSTEM OR S1 TO DEFAULT THEN RESET	This message will occur following a non-recoverable error such as a Trap. If this message occurs immediately following a Reset or Power Up, press "**S1**" to reload default parameters.
M ERROR	Acknowledge signal was not detected in amount of time specified by General Parameter 13.
S ERROR	Acknowledge signal was not detected in amount of time specified by General Parameter 15.
T ERROR	Acknowledge signal was not detected in amount of time specified by General Parameter 17.
BRAKE ENGAGED	Brake monitor input to Unidex 400 is in it's Logic Low State
GANTRY/PERPEN AXIS PAIR	The same axis has been specified as being Master and Slave or as Horizontal and Vertical. Reconfigure General Parameters 011 or 024.
MFO = 0, or FEEDHOLD	MFO Pot completely CCW, Feedhold or Quickstop active
AXIS LIMIT	A Limit Switch has become active while in Slew (Joystick) mode.
DSP FEEDBACK ERROR	Servo Board has detected an invalid code. Reset system, if error repeats, contact Aerotech, Inc.

SECTION 12-3: NON-FATAL ERRORS

These error messages indicate a correctable condition that prevents processing. If any of these Errors occur, correct the condition and Reset the system before continuing.

Initial Selection Menu

Message	Source/Suggested Solution
AXIS MIS-MATCH, RESET SYSTEM	Axes requested for motion do not match those activated in Axis Parameter x37, Existence Checking
NO PASSWORD PRIVILEGE	Entry has been requested into a Mode for which the current password does not have privilege
KEYBOARD ERROR - Break	Keyboard protocol error, faulty keyboard connection, wrong type keyboard.
KEYBOARD ERROR - Frame	Keyboard protocol error, faulty keyboard connection, wrong type keyboard.
KEYBOARD ERROR - Parity	Keyboard protocol error, faulty keyboard connection, wrong type keyboard.
KEYBOARD ERROR - Overrun	Keyboard protocol error, faulty keyboard connection, wrong type keyboard.

Edit Mode

Message	Source/Suggested Solution
illegal file name	An illegal filename (non-numerical). Valid filenames are 0000 - 9900.
out of memory	Out of Memory - No room to load Edit File
file not found	Requested file does not exist.
key input error, press S1	Illegal key sequence in menu mode
reserved file name	Cannot edit a reserved file.

File Mode

bad file	Invalid File Name
no file	Source File does not exist
file exist	File to be input already exists
no room	Memory Full
trans cksm	File Transfer Checksum Error
COMM (or) Prtcl	Serial Port Error during file transfer

Setup Mode

Message	Source/Suggested Solution
bad file name	File named to receive Setup Data is invalid
file not found	File named, from which to load Setup Data, not found
file already exists	File named to receive Setup Data already exists
no room in memory	Memory Full
data out of bounds	Parameter data entered is beyond limits
item out of bounds	Item Number selection not valid

Test Mode

system RAM r/w error	Read/Write Error in System RAM signifying hardware failure - Contact Aerotech, Inc.
user RAM r/w error	Read/Write Error in User RAM signifying hardware failure - Contact Aerotech, Inc.
xxxx checksum error	Checksum error in User's program number xxxx
ROM checksum error	System ROM error - Contact Aerotech, Inc.
SETUP checksum error	Setup data corrupt. Reload backup Setup file or set to default.

Machine Mode

Message	Source/Suggested Solution
format error, press S1	Format error, press Front Panel S1 Key
not found, press S1	File requested to be Run does not exist, press Front Panel S1 Key
reserved, press S1	File requested to be Run is a reserved file.
error, press S1	Miscellaneous Error, press Front Panel S1 key

SECTION 12-4: RUN TIME ERRORS

These errors indicate a correctable condition encountered during program execution. If any of these Errors occur, return to the Edit Mode, correct the condition and Rerun the program.

Message	Source/Suggested Solution
unrecognized command	
G code error	Command Syntax is not expected.
duplicate code	Command repeated within program block.
missing code	Axis move must be specified before Feedrate.
too many digits	Move or Feedrate value too large.
unrecognized variable	Variable has not been correctly defined. Refer to Chapter 3, Edit Mode, Math, File 9999 for addition of variables.
miss center points	Center Point not provided for requested Circle.

Message	Source/Suggested Solution
no feedrate	If G0 used, feedrate must be specified. If G1,G2 or G3 is used previous feedrate will be used.
no MST bus	Jumpers not properly configured. Refer to Chapter10, Hardware Information.
no I/O bus	Jumpers not properly configured. Refer to Chapter10, Hardware Information.
undefined math function	Math Syntax Error
undefined variable	Variable has not been defined. Refer to Chapter 3, Edit Mode, Math, File 9999 for addition of variables.
illegal I/O format	Incorrect Command Syntax
illegal math format	Math Syntax Error
undefined array	
data file error	File does not exist
modify system variable	System variables cannot be overwritten.
no more RAM space	File 9999 cannot be created. Not enough memory for additional variable.
no IEEE-488 option	IEEE-488 Option requested, not available

Message	Source/Suggested Solution
no more stack buffer	Memory allocated for User stack gone. Reconfigure General Parameter 010.
no repeat loop	Repeat Loop Syntax Error. Repeat Loop requested but cannot be executed.
not parts pgm file	File requested not a Parts Program. File requested is a reserved file
file not found	
label not found	Label Syntax Error
no gosub	No return from GOSUB command. Illegal placement of "RTS"
F9999 overflow	File 9999 overflow. Too many variables defined.

(THIS PAGE INTENTIONALLY LEFT BLANK)

CHAPTER 13: TROUBLESHOOTING THE UNIDEX 400

This Chapter is provided to aid the User in identification and solution of certain malfunctions.

SYMPTOM	POSSIBLE CAUSE/SOLUTION
Power switch does not illuminate when moved to the ON position	Verify line voltage (Refer to Section 11-1-1)
Power switch illuminates, fans operate, display remains blank and system does not produce initial "beep" upon power up.	Main power supply has tripped or blown a fuse. Contact an Aerotech, Inc. Customer Service Representative.
Upon power up, display shows only a small cursor in upper left corner.	Control cable not properly connected to Front Panel. Contact an Aerotech, Inc. Customer Service Representative.
Upon power up or reset, the display reads "INITIALIZE CPU".	CPU board faulty or not seated properly. Contact an Aerotech, Inc. Customer Service Representative.
Upon power up or reset, the display reads "INITIALIZE CPU" and "INITIALIZE DSP".	CPU board cannot communicate with the DSP board, one of the boards is faulty or not seated properly. Contact an Aerotech, Inc. Customer Service Representative.

SYMPTOM	POSSIBLE CAUSE/SOLUTION
Active axes not recognized (No Tracking Display).	Blown Motor Fuse. (Refer to Section 11-1-4)
	Feedback device not properly connected. (Refer to Sections 11-1-13 and 11-1-2).
	Incorrect Feedback device type specified (Refer to Section 7-7, Axis Parameter x33.)
	Rear Panel "ENCODER POWER" fuse blown. (Refer to Section 10-4)
	Failed Amplifier or Amplifier fuse. Contact an Aerotech, Inc. Customer Service Representative.
	Faulty motor brushes. Contact an Aerotech, Inc. Customer Service Representative.
Axis "run away"	Feedback device not properly connected. (Refer to Sections 11-1-13 and 11-1-2)
	Feedback device or motor connections switched between axes.
	Incorrect Feedback device type specified in Setup parameters. (Refer to Section 7-7, Axis Parameter x33)

SYMPTOM	POSSIBLE CAUSE/SOLUTION
"KEYBOARD ERROR: type"	Incorrect keyboard type or bad connection. (Refer to Section 11-1-8)
Motor oscillation while running	Servo gains set too high, retune. (Refer to Section 7-9)
Motor oscillation while in position	Servo gains set too high, retune. (Refer to Section 7-9)
"POSITION TRAP" or "INTEGRAL TRAP" or "VELOCITY TRAP" message	Servo loop not properly tuned, retune. (Refer to Section 7-9) Motor or Amplifier fuse blown. (Refer to Section 11-1-4)
"SLEW" is selected from the Machine Mode display, the display immediatly returns to the Machine Mode display.	Joystick not properly connected. (Refer to Section 11-1-8)
"BRAKE ENGAGED" message when a move is requested.	Verify Brake circuitry. (Refer to Section 11-1-12)
Home move is not in the desired direction	Change Axis parameter #002 setting. (Refer to Section 7-7)
Axis always in Limit	Limit Switch does not correspond to polarity setting. (Refer to Section 7-7, Axis Parameter x23).

SYMPTOM	POSSIBLE CAUSE/SOLUTION
Axis runs into mechanical stop	Limit Switches not properly connected. (Refer to Section 11-1-13)
	Improper feedback or motor connections. (Refer to Sections 11-1-13 and 11-1-2)
	Feedback and/or Motor connections swapped between axes.
Axis moves wrong distance	Verify that the appropriate scale factor is being used. (Refer to Section 7-7, Axis Parameter x00 and x01 and Section 7-8, General Parameter 002)
	Verify appropriate number of decimal places has been set. (Refer to Section 7-8, General Parameter 004)
	Encoder quadrature not proper. (Refer to Section 7-7, Axis Parameter x35)
No motion , but torque present on motor	MFO set to 0 (Refer to Section 7-8, General Parameter 023)
	Feedhold active (Refer to Section 6-3-3)
	Quick stop active (Refer to Section 7-8, General Parameter 030)

SYMPTOM	POSSIBLE CAUSE/SOLUTION
Intermittent motor operation	Motor brushes faulty. Contact an Aerotech, Inc. Customer Service Representative.
	Motor cable damaged, verify cable integrity.
Keyboard does not function	Verify that the Keyboard Mode is active. (**KEYB** is displayed on the second selection menu of the Edit Mode.)

(THIS PAGE INTENTIONALLY LEFT BLANK)

CHAPTER 14: MAINTENANCE

SECTION 14-1: BATTERY REPLACEMENT

Battery backup is provided for internal memory. Under normal operating conditions, battery life is approximately five years. It is suggested that the battery be changed at or about this time to prevent unexpected memory loss. The battery is located on the CPU Board (See Figures 14-1 and 14- 2). The following symptoms indicate the battery needs replaced:

Incomplete display upon power up.

SETUP ERROR, F1 TO DEFAULT, OTHERS SKIP displayed upon power up.

RAM ERROR, display upon power up.

The replacement battery must be a BCX Series BCX72 3B50 PC Size battery manufactured by Electrochem Industries, Division of Greatbatch LTD., 10,000 Wehrle Drive, Clarence, N.Y. 14031 or equivalent.

WARNING: PRIOR TO CHANGING THE BATTERY, MAKE CERTAIN ALL FILES HAVE BEEN BACKED UP.

The procedure for changing the battery is as follows:

1) Disconnect the Unidex 400 from the power source.

DANGER: TO AVOID THE POSSIBILITY OF ELECTRICAL SHOCK, MAKE CERTAIN ALL POWER HAS BEEN REMOVED FROM THE UNIDEX 400 PRIOR TO REMOVING THE FRONT PANEL.

2) Remove the Front Panel from the Unidex 400.

WARNING: STATIC DISCHARGE MAY SEVERELY DAMAGE THE CIRCUIT BOARDS. A STATIC STRAP, CONNECTED TO EARTH GROUND, MUST BE WORN BY ANYONE COMING INTO CONTACT WITH THE CIRCUIT BOARDS.

3) Verify that the cable connectors are appropriately labeled then remove the connectors from the CPU board.

4) Disconnect t he small cable leading to the Front Panel Board from the CPU board.

5) Remove the CPU board from the Unidex 400, and place the board on a static free pad.

6) Carefully clip the strap holding the battery in place. Remove the battery from the CPU board.

7) Install a new battery and hold-down strap onto the CPU board.

8) Re-install the CPU board into the Unidex 400 chassis.

9) Re-connect all previously removed CPU cables.

10) Re-install the Unidex 400's Front Panel onto the chassis.

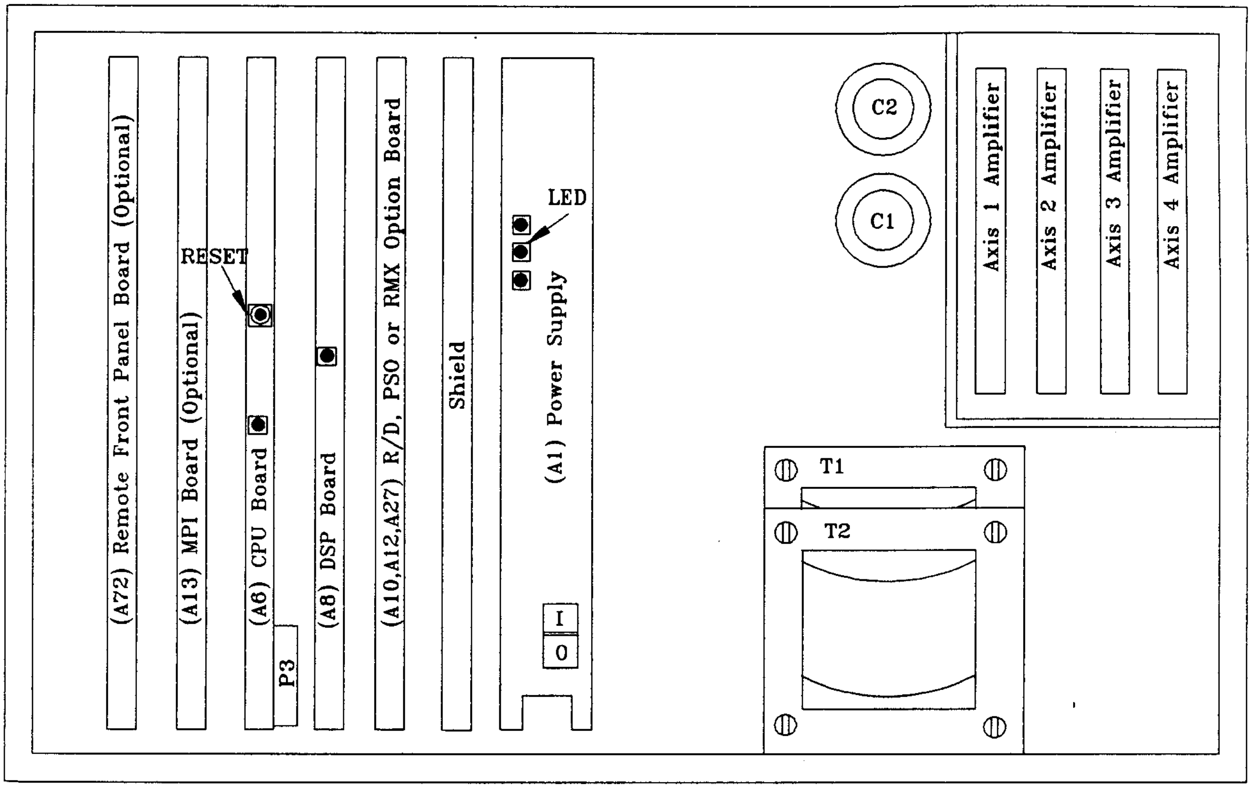


Figure 14-1: Unidex 400, Models U400B and U400R (Front View - Panel Removed)

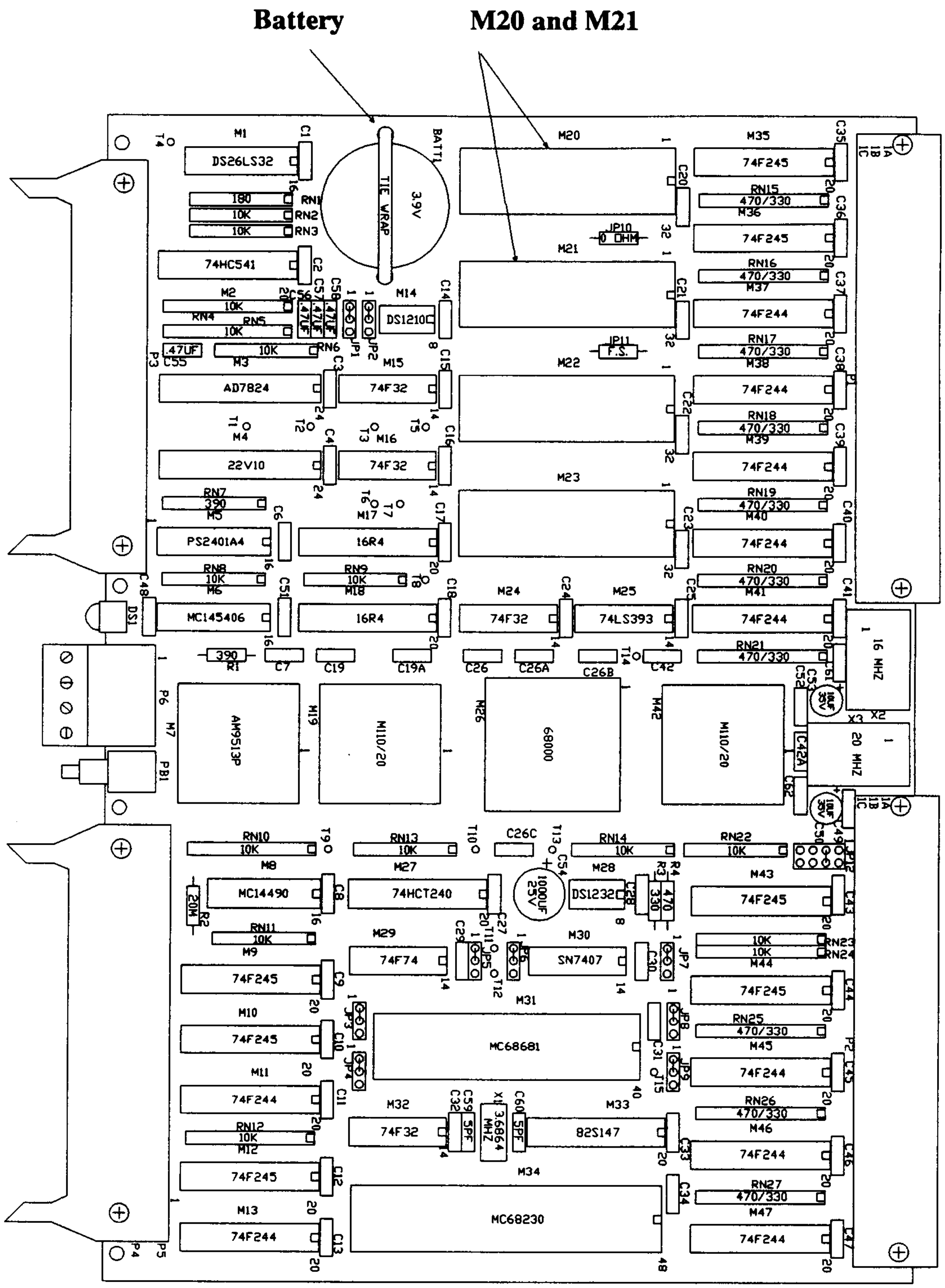


Figure 14-2: CPU Board Assembly

SECTION 14-2: SOFTWARE INSTALLATION INSTRUCTIONS

At certain times it may be necessary for Aerotech, Inc. to send the Customer updated versions of Unidex 400 software on EPROM chips. This section provides instructions for installing the new chips.

Remove the Unidex 400 circuit Boards as follows:

1) Disconnect the Unidex 400 from the power source.

DANGER: **TO AVOID THE POSSIBILITY OF ELECTRICAL SHOCK, DISCONNECT THE UNIDEX 400 FROM THE POWER SOURCE BEFORE REMOVING THE FRONT PANEL.**

2) Remove the Front Panel of the Unidex 400.

WARNING: **STATIC DISCHARGE MAY SEVERELY DAMAGE THE CIRCUIT BOARDS. A STATIC STRAP, CONNECTED TO EARTH GROUND, MUST BE WORN BY ANYONE HANDLING THE CIRCUIT BOARDS.**

3) Verify that the cable connectors are appropriately labeled then remove the connectors from the CPU and DSP Boards.

4) Disconnect one end of the small cable leading from the Front Panel Board to the CPU board.

5) Remove the CPU and DSP Boards from the Unidex 400, and place them on a static free pad.

CPU Board

Locate the EPROMs labeled M20 and M21. (Refer to Figure 14-2 for the relative location of these chips).

Carefully remove these chips from the CPU Board, noting their orientation.

Observing the same orientation, carefully insert the chips, containing the new software, into the appropriate slots. Make certain the chips are secure and that no pin damage has occurred.

DSP Board

Locate the EPROM labeled M10. (Refer to Figure 14-3 for the relative location of this chip).

Carefully remove this chip from the DSP Board, noting it's orientation.

Observing the same orientation, carefully insert the chip, containing the new software, into the appropriate slot. Make certain the chip is secure and that no pin damage has occurred.

6) Re-Install the CPU and DSP Boards in their proper locations within the card cage (Refer to Figure 14-1)and replace the Unidex 400's Front Panel.

NOTE: **It may be necessary to adjust the pins of the EPROM to insure proper fit into the socket.**

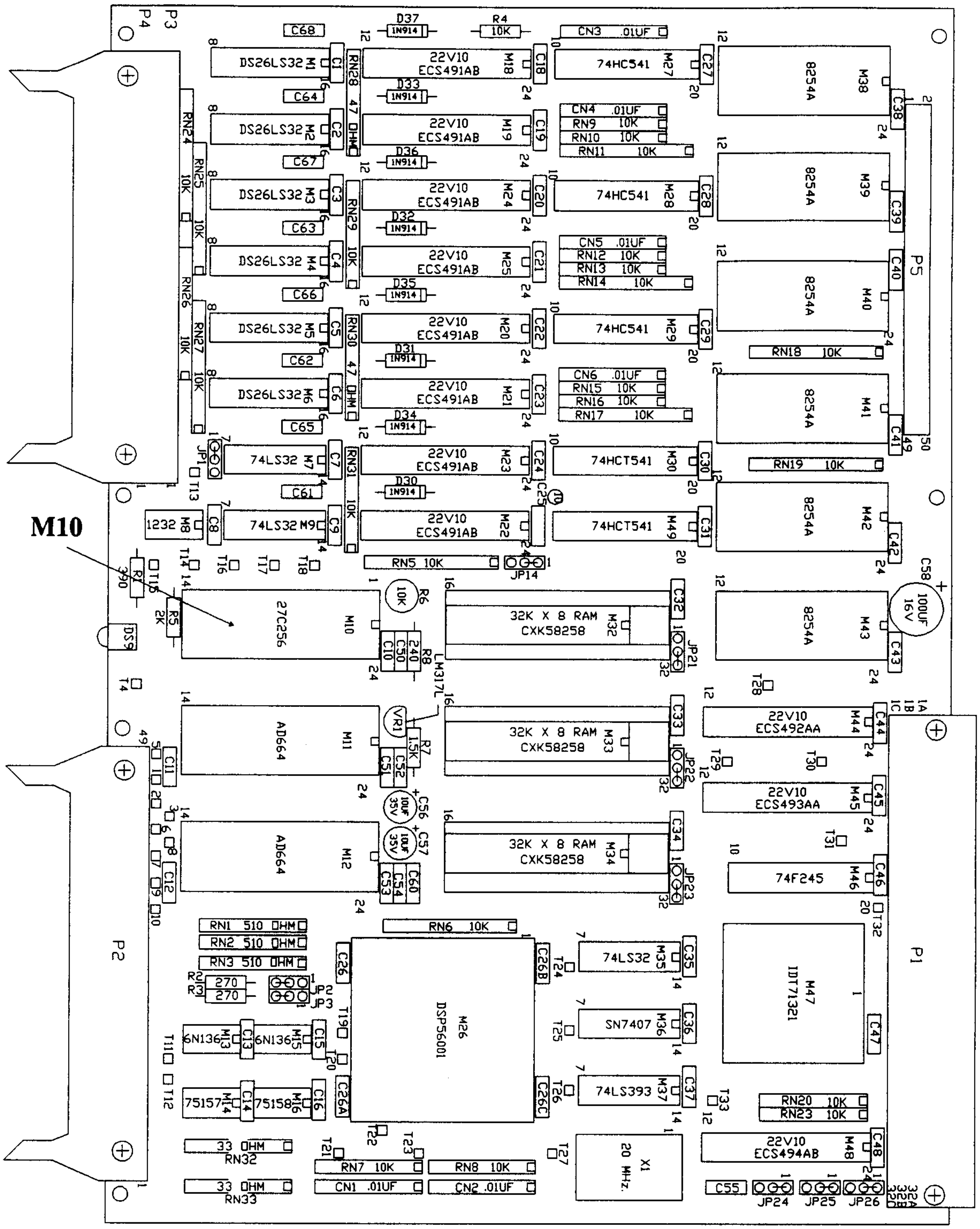


Figure 14-3: DSP Board Assembly

SECTION 14-3: SERVICE AND REPAIR

Customer repair of the equipment is limited. Control Board(s) may be removed and replaced if necessary, however, component level repair must not be attempted.

On-site service should be performed by an experienced electronic technician, preferably one trained by Aerotech.

SECTION 14-4: SHIPMENT

The procedure for shipping equipment to Aerotech, described below, pertains to warranty as well as non-warranty repairs.

1. Before returning any equipment a "Return Authorization Number" must be obtained from Aerotech. (Be prepared to give the serial number of the equipment being returned.)

2. The equipment being returned must be encased in a proper cushioning material and enclosed in a cardboard box.

Call for a "Return Authorization Number" if it is necessary to ship any equipment to the factory.

WARNING: DAMAGE TO THE EQUIPMENT DUE TO IMPROPER PACKAGING MAY VOID WARRANTY!

Aerotech Sales and Service offices are listed below. For service and information, contact the office servicing your area.

AEROTECH, INC. SALES OFFICES

WORLD HEADQUARTERS
AEROTECH ,INC.
101 Zeta Drive
Pittsburgh, Pa. 15238

Phone (412) 963-7470
FAX (412) 963-7459
TWX (710) 795-3125

AEROTECH, LTD.
Aldermaston
Berkshire RG7 4QW, England

Phone (07356) 77274
TLX 847228
FAX (07356) 5022

AEROTECH GMBH
Neumeyerstrasse 90
8500 Nuernberg 10
West Germany

Phone (0911) 521031
TLX 622474
FAX (0911) 521235

(THIS PAGE INTENTIONALLY LEFT BLANK)

CHAPTER 15: OPTIONS

This Chapter contains a description of each of the options that may be supplied with a Unidex 400 Motion Controller.

SECTION 15-1: JOYSTICK CONTROL AND DIGITIZING

The Joystick option provides the User precision manual axis positioning. The Aerotech supplied Joystick provides four axes of positioning capability. Manual Axis control is useful for delicate machine setups and position digitizing. The Joystick may also be used to make axis-pair selections, select a high or low velocity rate, choose velocity or absolute positioning modes and control program execution. The Joystick is sampled by the Unidex 400 from 8 Bit A/D's.

15-1-1 HARDWARE CONFIGURATION

The Unidex 400 accommodates only Aerotech supplied Joysticks Model Number JBV EFM 910-19 (see Figure 15-1).

WARNING: **Only Aerotech supplied Joysticks Model Number JBV EFM 910-19 may be used with the Unidex 400.**

The Joystick is interfaced to the JOYSTICK Connector (P34) on the Rear Panel of the Unidex 400. (See Chapter 11 of this manual for a detailed description of the JOYSTICK Connector.)

Before connecting the Joystick to the Unidex 400, make certain that the Joystick's Horizontal and Vertical Null Offset Indicators are located at approximately the center position. (The Null Offsets have been factory set and should need no adjustment.)

Connect the Joystick to the JOYSTICK (P34) Connector on the Rear Panel of the Unidex 400.

NOTE: **In order for the Unidex 400 to recognize the existence of a Joystick, connection must be made to the Controller prior to System Power Up or Reset.**

Power Up the Unidex 400.

NOTE: **If the Joystick is not recognized following a power up or reset, and the appropriate connections have been made, verify that the Null Offset Indicators are at approximately the center position.**

The following section provides details for operating the Joystick.

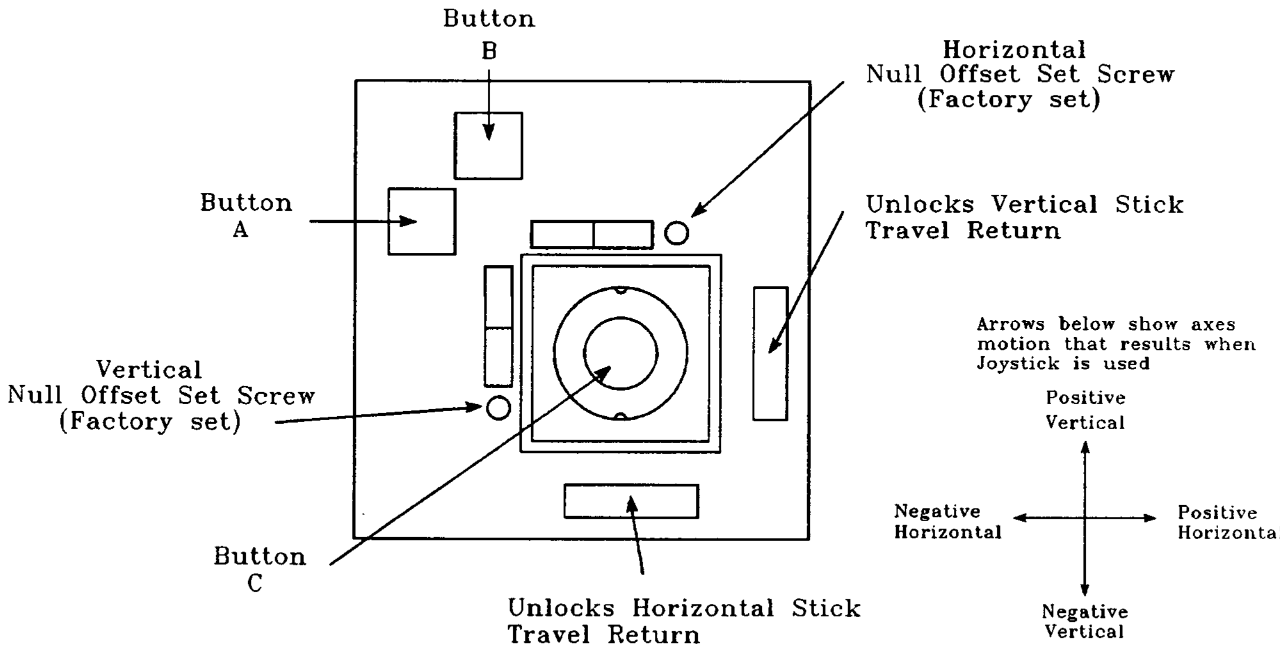


Figure 15-1: Unidex 400 Joystick - Top View

15-1-2 JOYSTICK OPERATION

The Joystick buttons, A, B, and C provide the following functions (refer to Figure 15-1 for button locations):

> **NOTE: Joystick button signals are debounced for 20 mSec. If a Joystick button is depressed for less than this amount of time, the signal will not be recognized.**

Joystick Button A

Button A is used to toggle between the horizontal (h) and vertical (v) axis pairs as established by General Parameter 008 (see the next section for an explanation of the Setup Parameters that affect Joystick operation). A maximum of eight defined axis pairs may be toggled through by the use of this button. When all defined pairs have been toggled through, the display will "wrap around" to the first defined pair. If in the Edit - Digitize Mode, the option of Block DELETE will be displayed following the last defined axis pair. When Block DELETE is active, the program block at the cursor location will be deleted by depressing Button A.

As illustrated below, the active axis pair is displayed in the status line during Joystick use.

In this example the active axes in the Machine - Slew Mode are X for horizontal stick movement, and Y for vertical stick movement.

```
X    0000.0000              Z    0000.0000          MFO  100 %
Y    0000.0000              U    0000.0000


JOY_STICK    SLEW    h-X    v-Y  p
                                 abort                        quit
```

Active axes while in the Edit - Digitize Mode.

```
    > 0: (program)


idxI        M          F          F          F         F

ins-Y/N     del/set    Varb       quit       <h-X   v-Y   h>
```

Joystick Button B

Button B is used to activate one of three modes of Joystick controlled axis motion. The three modes are Low Velocity Mode, High Velocity Mode and the Absolute Mode.

The Velocity Mode of operation coordinates the speed that the axis will move, to the amount of Joystick deflection. The Low Velocity selection enables the low speed range for Joystick deflection. Refer to the next section for an explanation of Axis Parameter x20 that determines the low speed range for the Low Velocity Mode.

The High Velocity selection enables the high speed range for Joystick deflection. Refer to the next section for an explanation of Axis Parameter x19 that determines the high speed range for the High Velocity Mode.

The Absolute Position Mode is used to coordinate the distance that the axis will move to the amount of Joystick deflection. Refer to the next section for an explanation of Axis Parameter x18 that determines the Absolute Position Mode Scale Factor.

As illustrated below, the active mode of Joystick motion is displayed as the last item in the status line while the Joystick is being used. The character will be a **p** for Absolute Positioning Mode, an **h** for High Velocity Mode or an **l** for Low Velocity Mode.

```
X   0000.0000          Z   0000.0000          MFO  100 %
Y   0000.0000          U   0000.0000

JOY_STICK   SLEW      h-x      v-y  p
                                 abort                    quit
```

```
     > 0: (program)

     idxI       M         F         F        F         F

     ins-Y/N    del/set   Varb      quit     <h-X  v-Y  h>
```

Joystick Button C

The function of Joystick Button C is dependent upon whether the Joystick is being used in the Edit - Digitize Mode or the Machine - Slew Mode. When in the Edit - Digitize Mode, Joystick Button C is used to record (digitize) the current axis location and write the location to a program block. When in the Machine - Slew Mode, Joystick Button C is used exit the Slew mode and return to the initial Machine Mode display.

Horizontal/Vertical Null Offset Switches and Set Screws

The Null Offset Switches and the accompanying set screws are used for fine tuning of the Joystick's resolution . These switches are factory set and should not be altered by the User.

Horizontal/Vertical Joystick Travel Return Switches

The Joystick Travel Return Switches are used to eliminate the Joystick's "return to center" feature. When this switch is in the "Off" position, the Joystick will remain in the position where it was released.

WARNING: WHEN THE JOYSTICK TRAVEL RETURN SWITCHES ARE IN THE "OFF" POSITION THE AXES WILL CONTINUE TO MOVE AFTER THE JOYSTICK HAS BEEN RELEASED.

15-1-3 JOYSTICK PARAMETERS

Before using a Joystick to control movement or digitize positions, certain Parameter settings must be established. The following paragraphs provide instructions for setting the applicable parameters. For a detailed description of all Parameter settings refer to Chapter 7 of this manual.

The Initial Selection Screen shown below will be displayed upon power up:

```
    UNIDEX 400                        Version x.xx

      RAM:ok    ROM:ok    SETUP:ok
      User RAM: xxxxxx

edit      file      setup      test      machine      PSWD
```

NOTE: **The PSWD option (F6) is displayed only if the Password function has been activated from the Parameter Mode.**

Enter the Setup Mode by selecting **setup** (F3), the display will be:

```
Set Up Mode



save      load      axes      general      tune      quit
```

<u>Axis Parameters</u>

Select **axes** to alter the axis parameters, the display will be:

```
x00 : METRIC SCALE FACTOR
x01 : ENGLISH SCALE FACTOR
x02 : HOME DIRECTION IS CCW ?                          0/1
 x= 1,2,3,4 for Axis X,Y,Z,U
item:                      data:
ins-Y/N    del     item          preset      quit      m
```

The following axis parameters affect Joystick use:

x18 : JOYSTICK ABS MODE SCALE FACTOR

This parameter is used to establish a "window" of axis movement for fine positioning when in the Joystick's Absolute Mode.

The scale factor setting range is 0-255. The default setting is 10, providing Joystick positioning in increments of 10 machine steps i.e. each increment that the Joystick moves, the axis will move 10 machine steps. An Aerotech, Inc. Joystick has approximately +/- 100 increments. Full deflection of a Joystick with a scale factor of 10 would equal 1000 machine steps.

x19 : JOYSTICK HIGH VELOCITY (STEPS/SEC)

Resolution ratio's vary between axes. When a Joystick move is requested, the speed/distance ratio for each affected axis must be compatible. This parameter establishes the speed at which the axis will move when a Joystick move is requested at the "High" speed setting. To derive the optimum Joystick "High" speed setting for this axis, proceed as follows:

Joystick High Velocity setting = desired velocity ÷ (screw pitch ÷ encoder steps per revolution)

NOTE: **The desired velocity must be expressed in "units per second".**

NOTE: **Compatible Units (English/Metric) must be used within this formula.**

NOTE: **When Aerotech equipment is being used, refer to Engineering Specification 12117 for the Lead/Gear Ratio and the Encoder Resolution.**

Example:

A velocity of 5 inches per second is desired when the Joystick is fully deflected. The screw pitch (lead/gear ratio) is .1 inches. The encoder resolution is 1000 steps per revolution. The setting of Axis Parameter x20 is derived as follows:

Joystick High Velocity setting = 5 ÷ (.1 ÷ 1000)

The Joystick High Velocity setting would be 50,000 steps per sec.

The setting range for this parameter is 0-99999999. The default setting is 50000 steps/second.

x20 : JOYSTICK LOW VELOCITY (STEPS/SEC)

Resolution ratio's vary between axes. When a Joystick move is requested, the speed/distance ratio for each affected axis must be compatible. This parameter establishes the speed at which the axis will move when the Joystick is fully deflected at the "Low" speed setting. To derive the optimum Joystick "Low" speed setting for this axis, proceed as follows:

Joystick Low Velocity = desired velocity ÷ (screw pitch ÷ encoder steps per revolution)

NOTE: **The desired velocity must be expressed in "units per second".**

NOTE: **Compatible Units (English/Metric) must be used within this formula.**

NOTE: **When Aerotech equipment is being used, refer to Engineering Specification 12117 for the Lead/Gear Ratio and the Encoder Resolution.**

Example:

A velocity of 1 inch per second is desired when the Joystick is fully deflected. The screw pitch (lead/gear ratio) is .2 inches. The encoder resolution is 4000 steps per revolution. The setting of Axis Parameter x20 is derived as follows:

Joystick Low Velocity setting $= 1 \div (.2 \div 4000)$

The Joystick Low Velocity setting would be 20,000 steps per sec.

The setting range for this parameter is 0-99999999. The default setting is 10000 steps/second.

General Parameters

Select **general** from the main Setup Menu to alter a General Parameter, the display will be:

The following General Parameter affects Joystick use:

000 : PASSWORD		
001 : SERIAL AUTO ECHO ?		**0/1**
002 : DEFAULT SF. ENGLISH/METRIC		**0/1**
003 : RS232 SETUP		
item:	**data:**	
ins-Y/N del item		**preset quit m**

008 : JOYSTICK PAIR

This parameter is used to setup combinations of axis pairs. Following setup, the pairs may be toggled through by use of the upper left Joystick button. A maximum of eight sets of Joystick pairs may be defined. The display will "wrap around" to the first defined pair after all defined pairs have been displayed.

Following selection of the Joystick Pair parameter, the display will be:

```
JOYSTICK SET UP NUMBER - 1

Horizontal                      Vertical
XYZU = [1000]                   XYZU = [0100]


  horiz     vert     number               preset     Quit
```

horiz Select **horiz** to move the cursor to the **Horizontal** field on the display, key in a "1" for the axis that is to be the horizontal axis for this axis pair (only one axis may be set to 1). The left and right arrow keys are used to move the cursor between the axes.

vert Select **vert** to move the cursor to the **Vertical** field on the display, key in a "1" for the axis that is to be the vertical axis for this axis pair (only one axis may be set to 1). The left and right arrow keys are used to move the cursor between the axes.

number Select **number** to step through all axis pair settings (Joystick Setup Number 1-8).

preset Select **preset** to display the default values for the Joystick pair number displayed.

quit Press the F6 key to save the displayed settings and return to the General Parameter display.

15-1-4 USING THE JOYSTICK

The Joystick provides two major functions to the Unidex 400 User. Through the Edit Mode a program can be created or altered by digitizing a desired motion path, or the Joystick may be used from within the Machine Mode to manually position an axis. The following sections provide the operating procedures for each of these functions.

NOTE: **Prior to using the Joystick, all applicable system parameters must be appropriately configured and the system must have been reset or powered down to initialize the data. (See Section 15-1-3)**

15-1-4-1 DIGITIZING

Following power up of the Unidex 400, the initial selection screen is displayed:

```
UNIDEX 400                          Version x.xx

  RAM:ok    ROM:ok    SETUP:ok
  User RAM: xxxxxx

edit     file     setup     test     machine     PSWD
```

Select **edit** to create or alter a program by digitizing.

The display will be:

```
editing & digitizing

  input file name (0 to 9900) =


ins-Y/N     del     last-E          last-R     quit
```

Enter the program number to be edited by digitizing. (Refer to Chapter 3 for operating information for the Edit Mode.)

The following Edit screen will be displayed:

```
> 0: (program)



insert      alter      delete      up      down      end      m
```

Press the "up arrow " key to view the Joystick Digitizing display.

```
> 0: (program)



D_idx      D_lin      D_cir      D_l+c      D_c+c      m
```

NOTE: **Positions digitized in Absolute values will be referenced to a previously established Software Home. If a Software Home position is required at the start of Joystick motion, the G92 command must be used to establish it. (Refer to Chapters 3 and 4 of this manual for programming information.)**

If digitizing is to be done within a previously established program, use the arrow keys to move the cursor to the desired location within the program.

Each of the digitizing selections will be described in the following sections:

D_idx

The **D_idx** selection is used to digitize positions along an index type move (refer to Chapter 3 of this manual for a description of an index type move). This selection establishes the G0 command in machine tool language.

Following selection of **D_idx**, the display will be:

```
> 0: (program)



idxI/A    M/E      F       F       F           F
ins-Y/N   del/set  Varb    quit    <h-X  v-Y   h>
```

The "up arrow" and "down arrow" keys are used to toggle through the index move fields. Joystick Button A is used to change the active axis pair.
The fields available for an index type move are:

idxI/A Digitizing of an index type move may be done in either the Absolute or Incremental Mode. To select the mode, use the arrow keys to place the cursor at this field, then select **del/set** (F2) to toggle between Incremental or Absolute modes of positioning. This selection establishes the G90 or G91 command in machine tool language. Refer to Chapter 3 of this manual for an explanation of the Absolute and Incremental modes of positioning.

M/E Digitizing of an index type move may be in either Metric or English units. To select the units, use the arrow keys to place the cursor at this field, then select **del/set** (F2) to toggle between Metric or English units. This selection establishes the G70 or G71 command in machine tool language. Refer to Chapter 3 of this manual for an explanation of the English and Metric modes.

F The Feedrate for an index type move of each of the four axes may be entered at the Feedrate locations. The order of the axes is XYZU. To enter the feedrate, use the arrow keys to place the cursor at the desired axis field, then key in the desired feedrate or select **Varb** (F3) and key in the variable number that contains the desired feedrate.

D_lin

The **D_lin** selection is used to digitize positions along a linear type move (refer to Chapter 3 of this manual for a description of a linear type move). This selection establishes the G1 command in machine tool language.

Following selection of **D_lin**, the display will be:

```
   > 0: (program)



 linI/A      M/E        F
 ins-Y/N     del/set    Varb      quit      <h-X   v-Y   h>
```

The "up arrow" and "down arrow" keys are used to toggle through the linear move fields. Joystick Button A is used to change the active axis pair.
The fields available for a linear type move are:

linI/A Digitizing of a linear type move may be done in either the Absolute or Incremental Mode. To select the mode, use the arrow keys to place the cursor at this field, then select **del/set** (F2) to toggle between Incremental or Absolute modes of positioning. This selection establishes the G90 or G91 command in machine tool language. Refer to Chapter 3 of this manual for an explanation of the Absolute and Incremental modes of positioning.

M/E Digitizing of a linear type move may be in either Metric or English units. To select the units, use the arrow keys to place the cursor at this field, then select **del/set** (F2) to toggle between Metric or English units. This selection establishes the G70 or G71 command in machine tool language. Refer to Chapter 3 of this manual for an explanation of the English and Metric modes.

F The Feedrate for a linear type move may be entered the Feedrate location. To enter the feedrate, use the arrow keys to place the cursor at this field, then key in the desired feedrate or select **Varb** (F3) and key in the variable number that contains the desired feedrate.

D_cir

The **D_cir** selection is used to digitize positions along a circular type move (refer to Chapter 3 of this manual for a description of a circular type move). This selection establishes the G2 (CW) or G3 (CCW) command in machine tool language.

Following selection of **D_cir**, the display will be:

```
> 0: (program)


cirI/A      M/E       X       X       F
ins-Y/N     del/set   Varb    quit    <h-X  v-Y  h>
```

The "up arrow" and "down arrow" keys are used to toggle through the circular move fields.

Joystick Button A is used to change the active axis pair.

The fields available for a circular type move are:

cirI/A Digitizing of a circular type move may be done in either the Absolute or Incremental Mode. To select the mode, use the arrow keys to place the cursor at this field, then select **del/set** (F2) to toggle between Incremental or Absolute modes of positioning. This selection establishes the G90 or G91 command in machine tool language. Refer to Chapter 3 of this manual for an explanation of the Absolute and Incremental modes of positioning.

M/E Digitizing of a circular type move may be in either Metric or English units. To select the units, use the arrow keys to place the cursor at this field, then select **del/set** (F2) to toggle between Metric or English units. This selection establishes the G70 or G71 command in machine tool language. Refer to Chapter 3 of this manual for an explanation of the English and Metric modes.

X Two axes must be designated to produce a circular type move. To select the first axis, use the arrow keys to place the cursor at the **X** in the third field, then select **del/set** (F2) to toggle through the available axes. To select the second axis, use the arrow keys to place the cursor at the **X** in the fourth field, then select **del/set** (F2) to toggle through the available axes.

NOTE: **It is recommended that the axes selected to do the circular move are the same axes as the Joystick"s active axes pair.**

F The Feedrate for a circular type move may be entered at the Feedrate location. To enter the feedrate, use the arrow keys to place the cursor at this field, then key in the desired feedrate or select **Varb** (F3) and key in the variable number that contains the desired feedrate.

The Joystick is used to digitize an arc or a circle as follows:

Use Joystick Button A to activate the axes pair to execute the circular move.

Use the Joystick to move the Axes from the Start Point to Point 1, press Joystick Button C to digitize (store) this point.

Use the Joystick to move the Axes from Point 1 to Point 2, press Joystick Button C to digitize (store) this point.

NOTE: **When creating a complete circle, Point 2 must be the same location as the Start Point.**

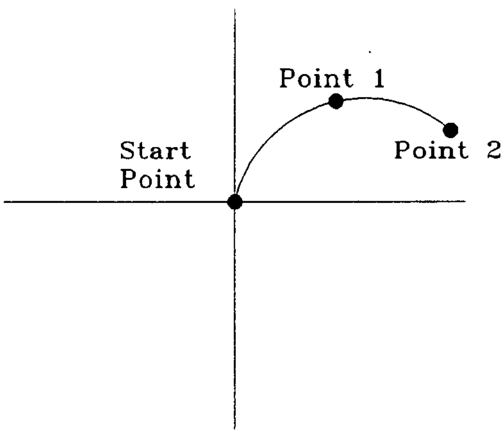


Figure 15-2: Digitizing an Arc

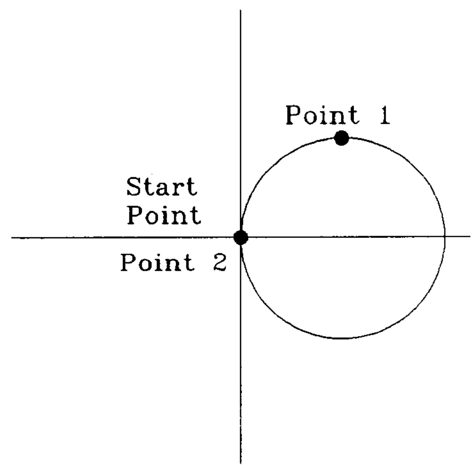


Figure 15-3: Digitizing a Circle

D_l+c

The **D_l+c** selection is used to digitize positions along both a circular type move and a linear type move within one program block (refer to Chapter 3 of this manual for a description of a linear + circular type move). This selection establishes the G1 (linear) and G2 (CW) or G3 (CCW) command in machine tool language.

Following selection of **D_l+c**, the display will be:

```
> 0: (program)


lin+cirl/A        M/E        X        X        F
ins-Y/N    del/set    Varb      quit      <h-X  v-Y  h>
```

The "up arrow" and "down arrow" keys are used to toggle through the circular move fields.

Joystick Button A is used to change the active axis pair.

The fields available for a circular and linear type move are:

lin+cirl/A Digitizing of the circular and linear type moves may be done in either the Absolute or Incremental Mode. To select the mode, use the arrow keys to place the cursor at this field, then select **del/set** (F2) to toggle between Incremental or Absolute modes of positioning. This selection establishes the G90 or G91 command in machine tool language. Refer to Chapter 3 of this manual for an explanation of the Absolute and Incremental modes of positioning.

M/E Digitizing of the circular and linear type move may be in either Metric or English units. To select the units, use the arrow keys to place the cursor at this field, then select **del/set** (F2) to toggle between Metric or English units. This selection establishes the G70 or G71 command in machine tool language. Refer to Chapter 3 of this manual for an explanation of the English and Metric modes.

X Two axes must be designated to produce the circular type move. To select the first axis, use the arrow keys to place the cursor at the **X** in the third field, then select **del/set** (F2) to toggle through the available axes. To select the second axis, use the arrow keys to place the cursor at the **X** in the fourth field, then select **del/set** (F2) to toggle through the available axes. Active axes not selected for the circular move will produce linear motion.

NOTE: **It is recommended that the axes selected to do the circular move e are the same axes as the Joystick"s active axes pair.**

NOTE: **Thee axes that are not involved in the circular move will be available to produce linear motion.**

F The Feedrate for the circular and linear type moves may be entered at the Feedrate location. To enter the feedrate, use the arrow keys to place the cursor at this field, then key in the desired feedrate or select **Varb** (F3) and key in the variable number that contains the desired feedrate.

The Circular and Linear move is digitized as follows:

Use Joystick Button A to activate the axes pair that will execute the circular move.

Use the Joystick to move the Axes from the Start Point to Point 1, press Joystick Button C to digitize (store) this point. (Refer to Figures 15-2 and 15-3.)

Use the Joystick to move the Axes from Point 1 to Point 2.

NOTE: **When creating a complete circle, Point 2 must be the same location as the Start Point.**

Use Joystick Button A to activate the axes pair that will execute the linear move.

Use the Joystick to move the Axes to the end point of the linear move.

Press Joystick Button C to digitize (store) Point 2 of the circular move and the end point of the linear move.

D_cir + cir

The **D_cir + cir** selection is used to digitize positions along two circular type moves (refer to Chapter 3 of this manual for a description of a circular type move). This selection establishes the G2 (CW) or G3 (CCW) command in machine tool language.

Following selection of **D_cir + cir**, the display will be:

```
> 0: (program)



cir+cirI/A      M/E        X        X        F
 ins-Y/N     del/set     Varb      quit     <h-X  v-Y  h>
```

The "up arrow" and "down arrow" keys are used to toggle through the circular move fields.

Joystick Button A is used to change the active axis pair.

The fields available for a circular type move are:

cir + cirI/A — Digitizing of a circular type move may be done in either the Absolute or Incremental Mode. To select the mode, use the arrow keys to place the cursor at this field, then select **del/set** (F2) to toggle between Incremental or Absolute modes of positioning. This selection establishes the G90 or G91 command in machine tool language. Refer to Chapter 3 of this manual for an explanation of the Absolute and Incremental modes of positioning.

M/E — Digitizing of a circular type move may be in either Metric or English units. To select the units, use the arrow keys to place the cursor at this field, then select **del/set** (F2) to toggle between Metric or English units. This selection establishes the G70 or G71 command in machine tool language. Refer to Chapter 3 of this manual for an explanation of the English and Metric modes.

X Two axes must be designated to produce a circular type move. To select the first axis, use the arrow keys to place the cursor at the **X** in the third field, then select **del/set** (F2) to toggle through the available axes. To select the second axis, use the arrow keys to place the cursor at the **X** in the fourth field, then select **del/set** (F2) to toggle through the available axes. Active axes not selected for the first circular move are used to produce the second circular move.

NOTE: **It is recommended that the axes selected to do the circular move e are the same axes as the Joystick"s active axes pair.**

NOTE: **The axes that are not involved in the first circular move will be available to produce the second circular motion.**

F The Feedrate for the circular type moves may be entered at the Feedrate location. To enter the feedrate, use the arrow keys to place the cursor at the desired axis field, then key in the desired feedrate or select **Varb** (F3) and key in the variable number that contains the desired feedrate.

The Circular moves are digitized as follows:

Use Joystick Button A to activate the axes pair that will execute the first circular move.

Use the Joystick to move the Axes for the first circular move from the Start Point to Point 1, (Refer to Figures 15-2 and 15-3).

Use Joystick Button A to activate the axes pair that will execute the second circular move.

Use the Joystick to move the Axes for the second circular move from the Start Point to Point 1

Press Joystick Button C to digitize (store) Point 1 of both of the circular moves.

NOTE: **When creating a complete circle, Point 2 must be the same location as the Start Point.**

Use Joystick Button A to activate the axes pair that will execute the first circular move.

Use the Joystick to move the Axes for the first circular move from the Start Point to Point 1, (Refer to Figures 15-2 and 15-3).

Use Joystick Button A to activate the axes pair that will execute the second circular move.

Use the Joystick to move the Axes for the second circular move from the Start Point to Point 1

Press Joystick Button C to digitize (store) Point 1 of both of the circular moves.

Use the Joystick to move the Axes for the second circular move from Point 1 to Point 2

NOTE: **When creating a complete circle, Point 2 must be the same location as the Start Point.**

Use Joystick Button A to activate the axes pair to execute the first circular move.

Use the Joystick to move the Axes for the first circular move from Point 1 to Point 2

Press Joystick Button C to digitize (store) Point 2 of both circular moves.

15-1-4-2 MACHINE MODE POSITIONING

The Joystick may be used from the Machine Mode to provide precision manual axes positioning.

NOTE: **Prior to using the Joystick, all applicable system parameters must be appropriately configured and the system must have been reset or powered down to initialize the data. (See Section 15-1-3.)**

Following power up of the Unidex 400, the initial selection screen is displayed:

```
   UNIDEX 400                          Version x.xx

     RAM:ok    ROM:ok    SETUP:ok
     User RAM: xxxxxx

edit      file      setup      test      machine      PSWD
```

Select **machine** to enter the Machine Mode. The following screen will be displayed:

```
X   0000.0000        Z   0000.0000          MFO  100 %
Y   0000.0000        U   0000.0000



home      jog      mdi      run      slew      quit      m
```

NOTE: **When activated from the Machine Mode, axes under Joystick control are as defined in the General Parameter 008: Joystick Pairs.**

Select **slew** to initiate Joystick positioning. As a safety feature, the Slew Mode cannot be entered if a Joystick is not connected to the Unidex 400.

NOTE: **If a Joystick is connected to the Unidex 400 following power-up, the system must be "reset" in order for the Joystick to be recognized.**

In addition to being accessible from the main Machine Mode, the Slew Mode may also be activated from the "Mdi" mode or from the User's program (G19).

Following selection of **slew**, the display will be:

```
X   0000.0000        Z   0000.0000        MFO  100 %
Y   0000.0000        U   0000.0000


JOY_STICK   SLEW     h-x      v-y  p
                                abort
```

Move the Joystick to position the axes as desired. Refer to Section 15-1-2 for details of Joystick operation.

Select **abort** (F4) to end Joystick control of the axes.

SECTION 15-2: HANDWHEEL OPTION

The Handwheel option provides the User with precision manual axis positioning, useful for delicate machine setups and positioning. The degree of precision is User controlled through Parameter settings that establish the Handwheel Increment/Machine Step ratio. Two Handwheels may be used simultaneously to control two axes, if both are activated from within the parts program.

15-2-1: HARDWARE CONFIGURATION

The Aerotech supplied Handwheel is a 100 pulse per revolution, 8830 Line Driver Output, Model MGN-10B-S8, manufactured by SUMTAK, a Division of Daido Corporation, Somerset, N.J. (See Figure 15-2).

The Handwheel(s) is interfaced to the WHEEL1 (P40) and/or WHEEL 2 (P41) Connectors at the Rear Panel of the Unidex 400. (Refer to Chapter 11 of this manual for details of the WHEEL 1 and WHEEL 2 Connectors.)

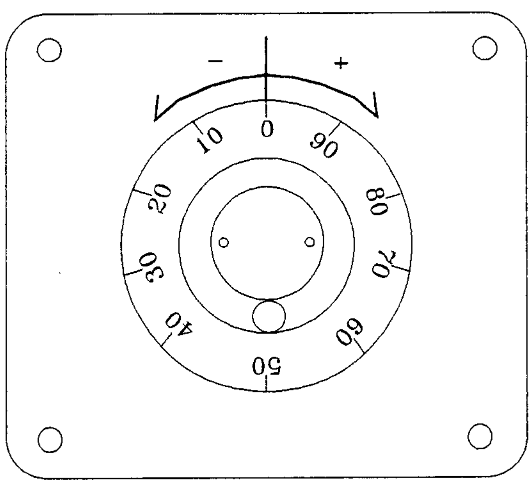


Figure 15-4: Unidex 400 Handwheel - Top View

15-2-2: SETUP

Connect the Handwheel(s) to the WHEEL 1 and/or WHEEL 2 Connectors on the Rear Panel of the Unidex 400.

NOTE: **In order for the Unidex 400 to recognize the existence of the Handwheel, connection must be made to the Controller prior to power up.**

Power up the Unidex 400. Before using a Handwheel to control movement, certain Parameter settings must be established. The paragraphs that follow provide instructions for setting the applicable parameters. For a detailed description of all Parameter settings, refer to Chapter 7 of this manual.

Axis Parameters

No Axis Parameters specifically affect Handwheel operation.

General Parameters

The following General Parameters affect Handwheel use:

018 : WHEEL 1 TYPE 0-3

This parameter is used to configure the input data format of the RS-422 Wheel 1 connector, to be in conjunction with the signal requirements of the input device.

The selections are:

0 - CW/CCW pulses
1 - CLK/DIR
2 - QUADRATURE x 1
3 - QUADRATURE x 2

The default setting is 2, for a Clock and Direction type input.

019 : WHEEL 2 TYPE 0-3

This parameter is used to configure the input data format of the RS-422 Wheel 2 connector, to be in conjunction with the signal requirements of the input device.

The selections are:

0 - CW/CCW pulses
1 - CLK/DIR
2 - QUADRATURE x 1
3 - QUADRATURE x 2

The default setting is 2, for a Clock and Direction type input.

020 : WHEEL 1 SCALE FACTOR

This parameter is used to establish a relationship between handwheel increments and machine steps of axes movement for Wheel 1 input.

machine steps = scale factor * wheel steps

The setting range is 0-255. The default setting is 10.

021 : WHEEL 2 SCALE FACTOR

This parameter is used to establish a relationship between handwheel increments/decrement setting and machine steps of axes movement for Wheel 2 input.

machine steps = scale factor * wheel steps

The setting range is 0-255. The default setting is 10.

022 : MFO ADJUSTS HANDWHEEL ? 0\1

This parameter is used to enable the MFO setting (established by the **G5/UMFO** command or the external MFO Potentiometer) to provide the Handwheel scale factor. Enabling this function provides the ability to change the Handwheel increment/machine step relationship without entering the parameter mode.

Example:

100 handwheel counts * 50% MFO setting results in 50 counts

0-No the MFO setting will not provide the Handwheel scale factor, 1-the MFO setting will provide the Handwheel scale factor.

The default setting is 1.

NOTE: **After all Parameters affecting Handwheel use have been properly configured, the System must be Reset to initialize the data.**

15-2-3: OPERATION

The Handwheel is used primarily to provide fine positioning of an axis. Handwheel positioning may be initiated through use of the HWEL command in a program or the Mdi mode, or from within the Machine Mode.

Refer to the *Unidex 21 Programming Manual* for a detailed description of the HWEL command for program or Mdi use.

NOTE: **Prior to using the Handwheel to control axis motion, all applicable system parameters must be appropriately configured and the system must have been reset or powered down to initialize the data.**

To initiate Handwheel positioning from the Machine Mode, proceed as follows:

Press the "W" key from the Machine Mode Main Menu.

Press the key of the axis to be active for Handwheel use. The selected axes name will be displayed at the lower left corner of the display.

Upon axis selection, the Disarm status will change to Armed (A).

> **NOTE:** **During Handwheel operation, the Handwheel may be disarmed at any time by depressing the "D" key.**

Input is available from one of two ports. Press the "I" key to toggle between Input ports 1 and 2.

The ratio of Handwheel increments to machine steps may be established by the use of the High/Low settings of this display.

Press the "H" key to increase the number of machine steps that is equal to each Handwheel increment. The machine step number is increased in increments as established in the Parameter Mode. (See Section 4-2 of this Chapter.) Press the "L" key to decrease the number of machine steps that is equal to each Handwheel increment. The machine step number is decreased in increments as established in the Parameter Mode. (See Section 4-2 of this Chapter.)

Upon completion of Handwheel operation, press the "D" key to disarm the Handwheel.

SECTION 15-3: POSITION SYNCHRONIZED OUTPUT OPTION (PSO)

The PSO option is a programmable, DSP based, Position Synchronized Output control card that precisely coordinates the functioning of a peripheral device with the motion initiated by the Unidex 400 Controller.

The PSO provides both digital and analog outputs. Variable output spacing, multiple pulse firing, pulse width, analog range, ramping functions, and power level adjustment are all coordinated by transducer feedback.

15-3-1: HARDWARE CONFIGURATION

JUMPERS AND SWITCHES

The PSO Card contains several jumpers and one eight position DIP Switch that must be appropriately configured prior to operation. (See Figure 15-4 for relative locations of the Switch and Jumpers.)

The PSO Card DIP Switch provides the VME Bus address and should be configured as follows:

SWITCH	POSITION	CONDITION
1	0	OFF
2	1	ON
3	0	OFF
4	1	ON
5	1	ON
6	0	OFF
7	1	ON
8	1	ON

The following is a list of the Jumpers on the PSO Card and an explanation as to their function:

Jumper	Condition	Function
JP1	1-2	High Speed Interrupt using a HCPL2601 Opto Coupler in the M9 location.
	2-3	High Speed Interrupt using a HP6N136 Opto Coupler in the M9 location. (Default)
JP2	1-2	Opto Isolated User Input using a HCPL2601 Opto Coupler in the M10 location.
	2-3	Opto Isolated User Input using a HP6N136 Opto Coupler in the M10 location. (Default)
JP3	1-2	Opto Isolated User Input using a HCPL2601 Opto Coupler in the M11 location.
	2-3	Opto Isolated User Input using a HP6N136 Opto Coupler in the M11 location. (Default)
JP4	1-2	Opto Isolated User Input using a HCPL2601 Opto Coupler in the M12 location.
	2-3	Opto Isolated User Input using a HP6N136 Opto Coupler in the M12 location. (Default)
JP5	1-2	Opto Isolated User Input using a HCPL2601 Opto Coupler in the M13 location.
	2-3	Opto Isolated User Input using a HP6N136 Opto Coupler in the M13 location. (Default)
JP6	1-2	Opto Isolated Output using either a 4N33 or HP6N136 Opto Coupler in the M14 location. (Default)
	2-3	Opto Isolated Output using a HCPL2601 Opto Coupler in the M14 location.

Jumper	Condition	Function
JP7	1-2	Opto Isolated Output using either a 4N33 or HP6N136 Opto Coupler in the M15 location. (Default)
	2-3	Opto Isolated Output using a HCPL2601 Opto Coupler in the M15 location.
JP8	1-2	Opto Isolated Output using either a 4N33 or HP6N136 Opto Coupler in the M16 location. (Default)
	2-3	Opto Isolated Output using a HCPL2601 Opto Coupler in the M16 location.
JP9	1-2	Opto Isolated Output using either a 4N33 or HP6N136 Opto Coupler in the M17 location. (Default)
	2-3	Opto Isolated Output using a HCPL2601 Opto Coupler in the M17 location.
JP10		Factory use only.
JP11		Factory use only.
JP12		Factory use only.
JP16	1-2	Internal or external Battery Back-Up.
	2-3	No Battery Back-Up. (Default)
JP17	1-2	Internal or external Battery Back-Up.
	2-3	No Battery Back-Up. (Default)
JP18	1-2	External Battery Back-Up.
	2-3	No Battery Back-Up. (Default)
JP19	1-2	Internal Battery Back-Up.
	2-3	No Battery Back-Up. (Default)
JP20	1-2	Internal or external Battery Back-Up.
	2-3	No Battery Back-Up. (Default)

Jumper	Condition	Function
JP21	1-2	Internal or external Battery Back-Up.
	2-3	No Battery Back-Up. (Default)
JP22		Factory Set
JP23		Factory Set
JP24		Factory Set
JP25		Factory Set
JP26		Factory Set
JP27	1-2	Active Low Polarity for the Main Output. (Default)
	2-3	Active High Polarity for the Main Output.
JP28	1-2	Active Low Polarity for the AUX2 Output. (Default)
	2-3	Active High Polarity for the AUX2 Output.
JP29	1-2	Active Low Polarity for the AUX3 Output. (Default)
	2-3	Active High Polarity for the AUX3 Output.
JP30	1-2	Active Low Polarity for the AUX4 Output. (Default)
	2-3	Active High Polarity for the AUX4 Output.
JP31	1-2	Provides TTL Output for the AUX4 signal using a 7407 Buffer Open Collector.
	2-3	Provides Opto Output for the AUX4 signal. (Default)
JP32	1-2	Provides TTL Output for the AUX3 signal using a 7407 Buffer Open Collector.
	2-3	Provides Opto Output for the AUX3 signal. (Default)

Jumper	Condition	Function
JP33	1-2	Provides TTL Output for the AUX2 signal using a 7407 Buffer Open Collector.
	2-3	Provides Opto Output for the AUX2 signal. (Default)
JP34	1-2	Provides TTL Output for the MAIN signal using a 7407 Buffer Open Collector.
	2-3	Provides Opto Output for the MAIN signal. (Default)

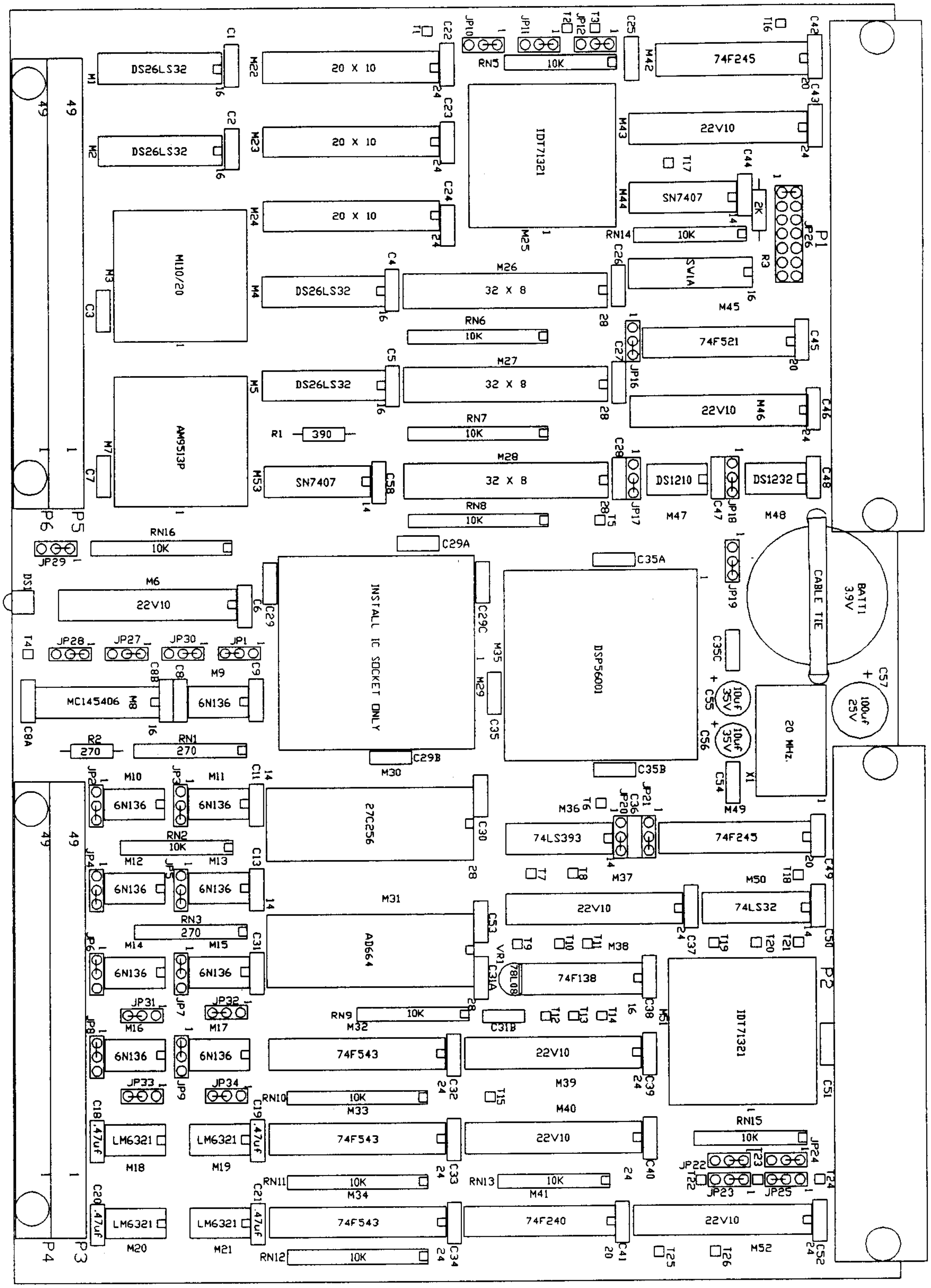
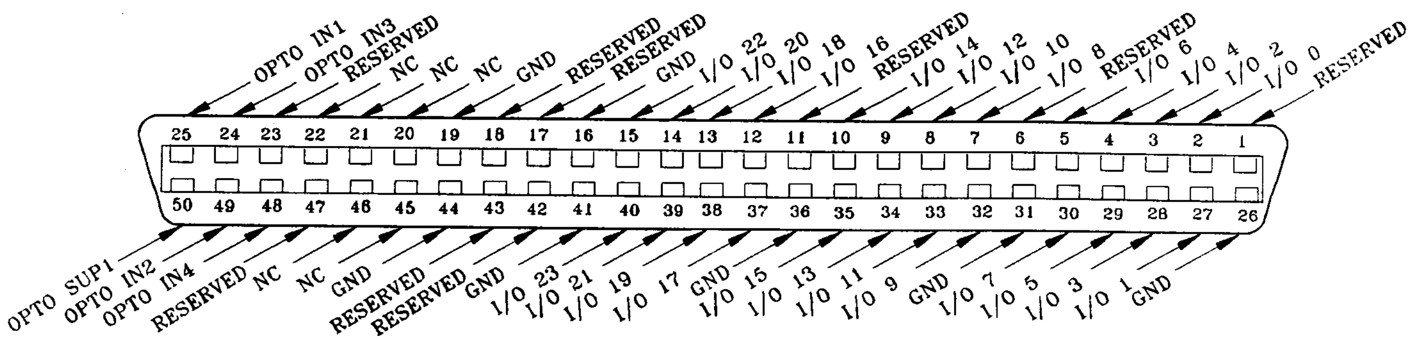


Figure 15-5: PSO Board - Jumpers and Switches

PSO INTERFACE

Interface to the PSO Card is through Connectors P51 and P52 located in the the Optional Connector Section of the Rear Panel of the Unidex 400 Controller. Details of the connectors are shown in Figures 15-6 and 15-7. Electrical characteristics of the connectors are illustrated in Figure 15-8.

J51 - CONNECTOR



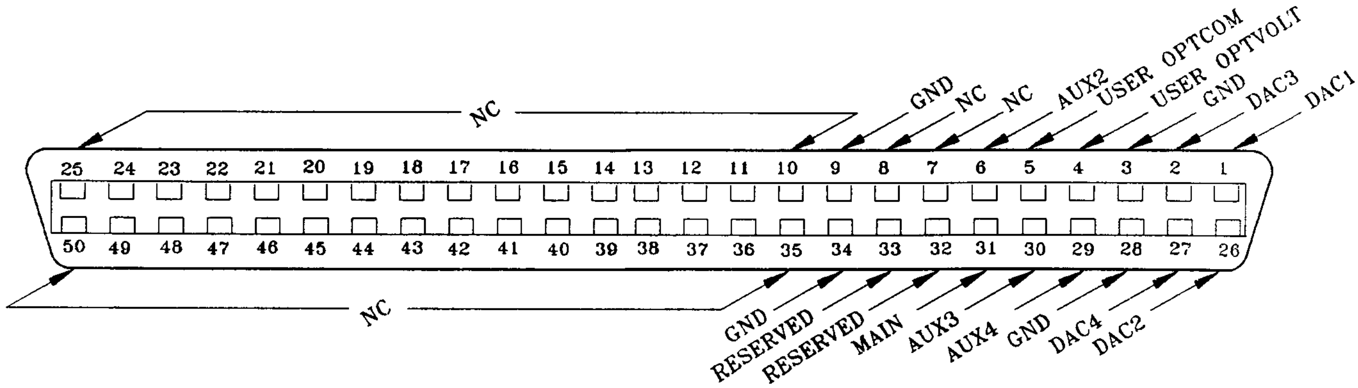
PIN	DESCRIPTION
1	Reserved
2	I/O Bit 0 (TTL) *
3	I/O Bit 2 (TTL) *
4	I/O Bit 4 (TTL) *
5	I/O Bit 6 (TTL) *
6	Reserved
7	I/O Bit 8 (TTL) *
8	I/O Bit 11 (TTL) *
9	I/O Bit 12 (TTL) *
10	I/O Bit 14 (TTL) *
11	Reserved
12	I/O Bit 16 (TTL) *
13	I/O Bit 18 (TTL) *
14	I/O Bit 20 (TTL) *
15	I/O Bit 22 (TTL) *
16	Common
17	Reserved

Figure 15-6: Unidex 400 PSO Conector (J51)

PIN	DESCRIPTION
18	Reserved
19	Common
20,21,22	Not Connected
23	Positive Opto Input
24	Opto Input 3
25	Opto Input 1
26	Common
27	I/O Bit 1 (TTL) *
28	I/O Bit 3 (TTL) *
29	I/O Bit 5 (TTL) *
30	I/O Bit 7 (TTL) *
31	Common
32	I/O Bit 9 (TTL) *
33	I/O Bit 11 (TTL) *
34	I/O Bit 13 (TTL) *
35	I/O Bit 15 (TTL) *
36	Common
37	I/O Bit 17 (TTL) *
38	I/O Bit 19 (TTL) *
39	I/O Bit 21 (TTL) *
40	I/O Bit 23 (TTL) *
41	Common
42,43	Reserved
44	Common
45,46	Not Connected
47	Reserved
48	Opto Input 4
49	Opto Input 2
50	User supplied Input Opto Voltage

* Outputs are driven by the 74F543 TTL Buffer.

J52 - CONNECTOR



PIN	DESCRIPTION
1	Digital to Analog Converter Output - 1 (+/- 10V @ 100 mA)
2	Digital to Analog Converter Output - 3 (+/- 10V @ 100 mA)
3	Common
4	User supplied Output Opto Voltage, +5V (Required with 6N or HCP only)
5	User supplied Output Opto Voltage Common
6	AUX2 Output
7,8	Not Connected
9	Common
10 thru 25	Not Connected
26	Digital to Analog Converter Output - 2 (+/- 10V @ 100 mA)
27	Digital to Analog Converter Output - 4 (+/- 10V @ 100 mA)
28	Common
29	AUX4 Output
30	AUX3 Output

Figure 15-7: Unidex 400 PSO Conector (J52)

PIN	DESCRIPTION
31	MAIN Output
32	Reserved
33	Reserved
34	Common
35 thru 50	Not Connected

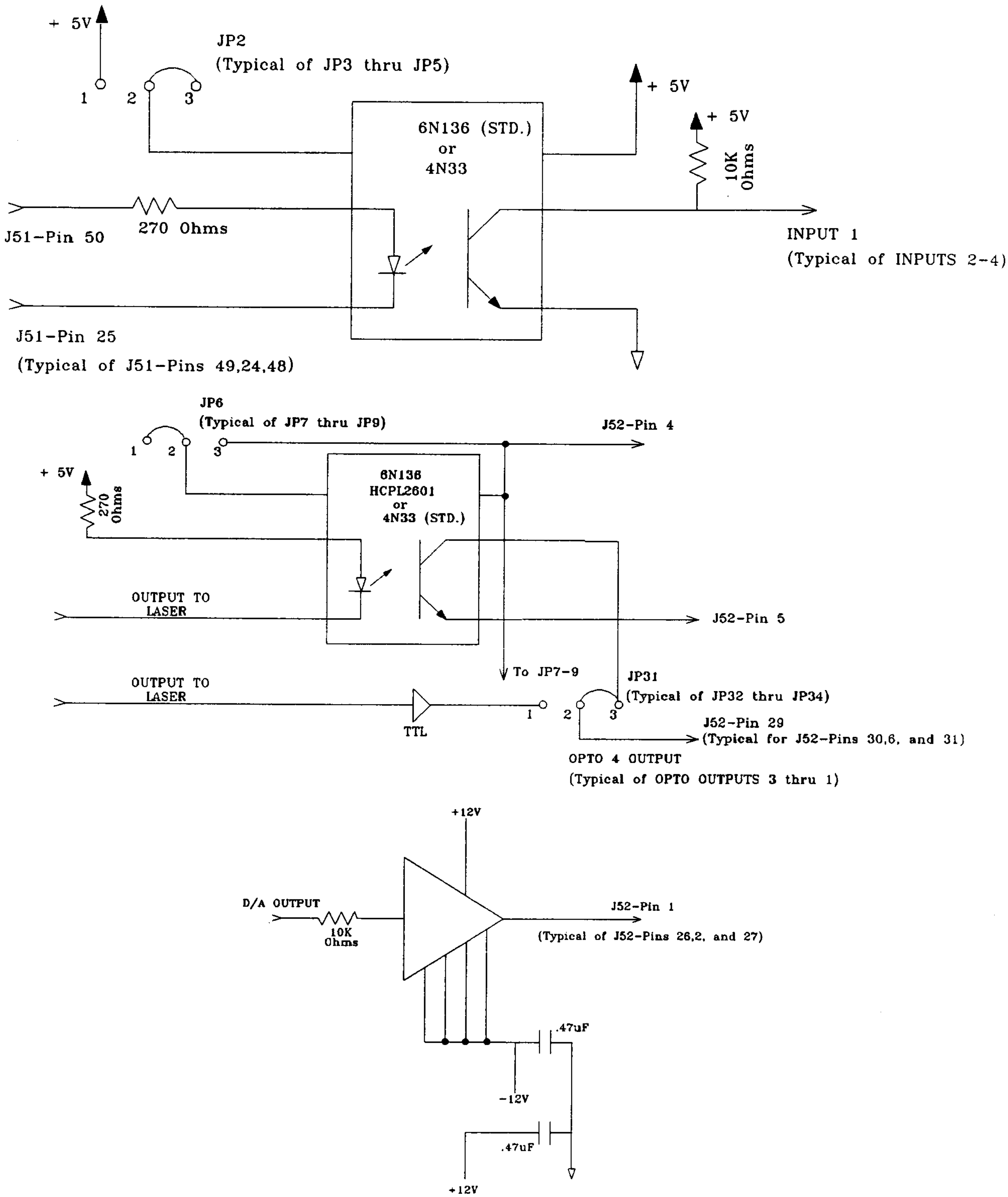


Figure 15-8: Electrical Characteristics of the PSO Interface Connectors (J51 and J52)

15-3-2: USING THE PSO OPTION

The PSO option is utilized by various programming options to coordinate the functioning of a peripheral device with motion initiated by the Unidex 400 Controller. It is most commonly used to control Laser firing. With the use of a variety of commands, the PSO Board may be instructed to activate up to four firing outputs with analog level controls and various types of single-shot or pulse train outputs.

The Position Synchronized Output Board is activated through either a parts program or from the MDI Mode. The following sections provide the commands related to PSO function:

The initial selection screen below will be displayed upon power up:

```
UNIDEX 400                              Version x.xx

   RAM:ok    ROM:ok    SETUP:ok
   User RAM: xxxxxx

edit      file      setup      test      machine      PSWD
```

NOTE: **The PSWD option (F6) is displayed only if the Password function has been activated from the Parameter Mode.**

Select **edit** to create or alter a program.

The display will be:

```
editing & digitizing

   input file name (0 to 9900) =

ins-Y/N      del      last-E            last-R      quit
```

Enter the program number to be created or edited using the PSO function. (Refer to Chapter 3 for operating information for the Edit Mode.)

The following Edit screen will be displayed:

```
> 0: (program)

insert    alter    delete    up    down    end    m
```

Select either **insert** or **alter** from the initial Edit screen, the display will be:

```
> 0: (program)

motion    I/O    mode    flow    servo    quit    m
```

Press the "down arrow" for additional Edit functions, the display will be:

```
> 0: (program)

option    dwell    MSG    MISC    INTRP    math    m
```

Select **option** from the second Edit screen, the display will be:

```
> 0: (program)

jstik    handwl    PSO                    quit
```

PSO-- > DIST

Select **PSO** from the second Option screen, the display will be:

dist	**pulse**	**fire**	**cond**	**output**	**quit**	**m**

Press the "down arrow" to view additional PSO functions.

Rtime	**Bmap**	**quit**	**m**

15-3-2-1: Firing Distance

dist Select **dist** to establish the number of machine steps or program units the axes will travel before synchronized output (firing) occurs. Refer to Example Programs 4,6,9 and 10.

The firing distance (PSO D or G80D) command is used in conjunction with the PSO F/G80 and Pulse type command PSO P/G80P

Following selection of **dist**, the display will be:

FIRING DISTANCE

ins-Y/N	**del**	**,**	**help**	**Varb**	**quit**	**m**

Select **help** to display the conditions available for establishing the firing distance.

PSO-- > DIST

The display will be:

```
FIRING DISTANCE
0,nn        : fixed inc. machine steps
1,nn,mm     : inc. from var nn, + - mm sets
2,nn,mm     : abs.  from var nn, + -mm sets
>
 ins-Y/N        del       ,       help     Varb        quit        m
```

Following is a description of each of the distance options:

0,nn Pulse Train firing will occur at the fixed incremental distance designated by "n".

Example

Enter **0,28** the program block will be:
PSO D,0,28 or G80 D,0,28

Pulse Train firing will occur at the incremental distance of 28 machine steps. No scale factor is used.

1,nn,mm Pulse Train firing will occur at the incremental distance as defined in variable number "n". Subsequent Pulse Train firing will occur at the incremental distances contained in variables +/- "m" from the initial variable.

Example

Enter **1,124,5** the program block will be:
PSO D,1,124,5 or G80 D,1,124,5

Pulse Train firing will occur at the incremental distances (in machine steps) specified by variable numbers 124, 125, 126, 127, and 128.

PSO-- > DIST

2,nn,mm Pulse Train firing will occur at the absolute distance as defined in variable number "n". Subsequent firing will occur at the absolute distances contained in variables +/- "m" from the initial variable. Current location is (0,0,0,).

Example

Enter **2,145,3** the program block will be:
PSO D,2,145,3 or G80 D,2,145,3

Pulse Train firing will occur at the absolute distances (in machine steps) specified by variable numbers 145, 146, and 147.

No scale factor is used.

The following selections may be made from any **help** menu when entering or editing a PSO condition.

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

, Select F3 to place a comma in the text.

help Select F4 (**help**) to display the available options and syntax of a PSO command. Subsequent pressing of the F4 key will scroll through additional option displays.

Varb Select F5 (**Varb**) to insert a variable number into the PSO program block. (Refer to the Edit -- > Math Mode for variable definition.)

quit Select F6 (**quit**) to abort the command and return to the initial PSO display.

Press the "down" arrow for additional PSO edit selections.

A through **F** These selections are available to enter a Hex value into a PSO program block.

Press **ENTER** to write the program block.

PSO-- > PULSE

15-3-2-2: Pulse Train

pulse Select **pulse** from the initial PSO menu to configure the Pulse Train. Refer to Example Programs 1,4,5,6,7,9 and 10.

The Pulse Train (PSO P or G80 P) command is used in conjunction with the PSO F/G80 F, PSO D/G80 D, PSO M/G80 M and the PSO C/G80 C commands.

Following selection of **pulse**, the display will be:

```
PULSE TRAIN



ins-Y/N     del     ,     help    Varb     quit     m
```

Select **help** to display the conditions available to configure the Pulse Train.

```
PULSE TRAIN
0,ww        : 1 shot pulse width ww Msec
1,l,w,t     : lead = 1 Msec, width = w Msec
              trail = t Msec
>
ins-Y/N     del     ,     help    Varb     quit     m
```

Select **help** a second time to view additional conditions that may be used to configure the Pulse Train.

The display will be:

```
PULSE TRAIN
2,l,w,t ,r,g   : lead = 1 Msec, width= w Msec
                 trail = t Msec, ramp by r Msec, with
                 g Msec gap, if g=0, gap = ramp width
>
 ins-Y/N        del        ,        help     Varb        quit       m
```

Select **help** a third time to view additional conditions that may be used to configure the Pulse Train.

The display will be:

```
PULSE TRAIN
3,nn,mm   : var nn off, nn+1 on, ... Msec
4,ww      : 1 shot pulse width ww Usec
5         : odd event enable, even disable
>
 ins-Y/N        del        ,        help     Varb        quit       m
```

Following is a description of each of the Pulse Train options:

0,ww Establishes the width of a single pulse output in milliseconds (see Figure 15-9).

Example

Enter **0,52** the program block will be:
PSO P,0,52 or G80 P,0,52

A Pulse Train is established with a pulse width of 52 milliseconds.

PSO-- > PULSE

1,l,w,t Establishes a Pulse Train with the following characteristics:
l = pulse lead in milliseconds
w = pulse width in milliseconds
t = pulse trail in milliseconds
(See Figure 15-9.)

Example

Enter **1,25,32,45** the program block will be:
PSO P,1,25,32,45 or G80 P,1,25,32,45

A Pulse Train is established with a pulse lead of 25 milliseconds, a width of 32 milliseconds and a trail of 45 milliseconds.

2,l,w,t,r,g Establishes a Pulse Train with the following characteristics:
l = pulse lead in milliseconds
w = pulse width in milliseconds
t = pulse trail in milliseconds
r = ramp up and down time in milliseconds
g = interval (gap) between ramps in milliseconds
If the interval is set at "0" the interval will be the same as the Pulse Train width (w). (See Figure 15-9.)

Example

Enter **2,12,24,28,36,47** the program block will be:
PSO P,2,12,24,28,36,47 or G80 P,2,12,24,28,36,47

A Pulse Train is established with a pulse lead of 12 milliseconds, a width of 24 milliseconds, a trail of 28 milliseconds, an up and down ramp time of 36 milliseconds and an interval of 47 milliseconds between ramps.

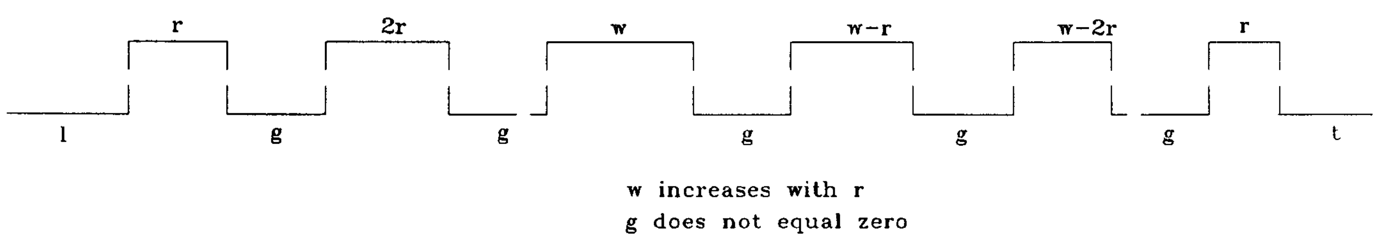


Figure 15-9: Typical Pulse Train

3,nn,,mm The Pulse Train will be configured as defined in variable number "n", "n + 1", "n + 2" etc. The pulse pattern is Variable "n" OFF, Variable "n + 1" ON, Variable "n + 2" OFF, Variable "n + 3" ON, etc. The number of consecutive variables used to produce the Pulse train is designated by "mm". (Refer to Figure 15-10.)

Example

Enter **3,143,9** the program block will be:
PSO P,3,143,9 or G80 P,3,143,9

If Variables 143 through 151 contain the following times (in ms), the pulse will be as illustrated in Figure 15-10.

Variable	Time (ms)
143	3
144	5
145	2
146	7
147	2
148	5
149	5
150	6
151	8

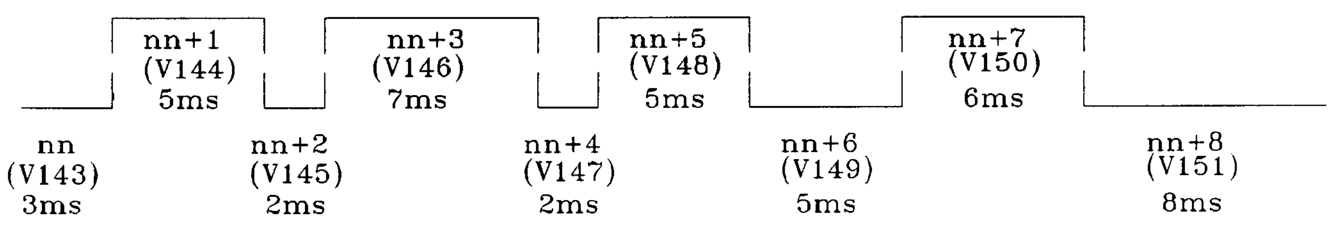


Figure 15-10: Optional Pulse Train

PSO-- > PULSE

4,ww Establishes the width, "w" of a single pulse output in micro seconds

Example

Enter **4,187** the program block will be:
PSO P,4,187 or G80 P,4,187

NOTE: **This command cannot be made to repeat continuously by using the PSO F,1/G80 F,1 command.**

A Pulse Train is established with a pulse width of 187 micro seconds.

5 Pulse output is toggled every sample distance.

Example

Enter **5** the program block will be:
PSO P,5 or G80 P,5

After the axes (designated by the PSO F/G80 F) has moved the distance specified by the PSO D/ G80 D command, the pulse output is enabled and remains enabled until the axes moves the designated distance.

15-3-2-3: Fire

fire Select **fire** from the initial PSO menu to enable or disable tracking of the specified axis position by the PSO. This command is also used to "manually" fire Pulse Trains.

Refer to Example Programs 1,4,5,6,7,8,9,10.

The Fire (PSO F or G80 F) command is used in conjunction with the PSO P/G80 P, PSO D/G80 D, and the PSO M/G80 M commands.

Following selection of **fire**, the display will be:

```
FIRING CONTROL



ins-Y/N     del     ,     help     Varb     quit     m
```

Select **help** to display the conditions available for enabling or disabling the Pulse Train Output.

```
FIRING CONTROL
0        : disable
1        : continuously firing
2,n      : fire n sets, n=0, wait till done
>
ins-Y/N     del     ,     help     Varb     quit     m
```

Select **help** a second time to view additional conditions that may be used to enable or disable the Pulse Train Output.

```
FIRING CONTROL
3,a1,a2,a3 : enable, lock at position
             of axes, 1-X, 2-Y, 3-Z, 4-U
             at most 3 axes
>
ins-Y/N     del     ,     help     Varb     quit     m
```

PSO-- > FIRE

Select **help** a third time to view additional conditions available for enabling or disabling the Pulse Train Output.

```
FIRING CONTROL
4,n,a1,a2,a3 : bit map case, n + forward
               - reverse dist, 1 output high, 0 low
5,n,a1,a2,a3 : 1 pulse train, 0 no
>
ins-Y/N      del      ,      help      Varb      quit      m
```

Following is a description of each of the Firing Control options:

0 Output Firing and Tracking disabled. (Default) May also be used to abort a previously activated PSO F,1/G80 F,1.

Example

Enter **0** the program block will be:
PSO F,0 or G80 F,0

Output Firing Pulse Train and Tracking disabled.

1 Activates the Output Firing as established by the **PSO P/G80 P** Command. The Pulse Train will continue until disabled by the **PSO F,0/G80 F,0** command. No position Tracking.

Example

Enter **1** the program block will be:
PSO F,1 or G80 F,1

Output Firing Pulse is enabled.

2,n Activates the Output Firing Pulse Train established by the PSO P/G80 P **Command. The Pulse Train will continue "n" number of times. If n = zero the Output Firing Pulse Train will not be activated until the previous Output Firing Pulse Train is complete. No position Tracking.**

Example

Enter **2,10** the program block will be:
PSO F,2,10 or G80 F,2,10

Output Firing Pulse will occur 10 times.

3,a1,a2,a3

Activates the Output Firing Pulse Train established by the **PSO P/G80 P** Command. Position Counters will "lock on" the motion of the designated axes (See NOTES). Output firing will occur at distances as established by the **PSO D/G80 D** command. A maximum of three (3) axes may be "locked on" simultaneously.

Example

Enter **3,1,2,4** the program block will be:
PSO F,3, 1,2,4 or G80 F,3,1,2,4

Activates the Output Firing Pulse Train. Position Counters will "lock on" the motion of axes X, Y, and U. Output firing will occur at distances as established by the **PSO D/G80 D** command.

4,n,a1,a2,a3

Activates the Output Firing Pulse Train established by the **PSO P/G80 P** Command, and locks the Position Counters onto the designated axes. (See NOTES). A maximum of three (3) axes may be "locked on" simultaneously. The Firing Pattern is determined by the **PSO M/G80 M** command.

If Bit = 1, the Output will go/remain high.
If Bit = 0, the Output will go/remain low.

The Pulse Output will occur at the fixed incremental distance "n". If "n" is positive, the Bit Pattern will be run in a forward direction. If "n" is negative, the bit pattern will run in reverse.

Example

Enter **4,200,1,2,4** the program block will be:
PSO F,4,200, 1,2,4 or G80 F,4,200,1,2,4

Activates the Output Firing Pulse Train. Position Counters will "lock on" the motion of axes X, Y, and U. Output firing pattern will occur as established by the **Bmap** command, for a forward incremental distance of 200 machine steps.

PSO-- > FIRE

5,n,a1,a2,a3

Activates the Output Firing Pulse Train established by the **PSO P/G80 P** Command, and locks the Position Counter on the designated axis (See NOTES). A maximum of three (3) axes may be "locked on" simultaneously. The Firing Pattern is determined by the **PSO M/G80 M** command.

If Bit = 1, the Output will be 1 Pulse Train.
If Bit = 0, there will be no Output.

The Pulse Output will occur at a fixed incremental distance "n". If "n" is positive, the Bit Pattern will be run in a forward direction. If "n" is negative, the Bit Pattern will be run in reverse.

Example

Enter **4,200,1,2,4** the program block will be:
PSO F,4,200, 1,2,4 or G80 F,4,200,1,2,4

Activates the Output Firing Pulse Train. Position Counters will "lock on" the motion of axes X, Y, and U. Output firing pattern will occur as established by the **Bmap** command, for a forward incremental distance of 200 machine steps.

NOTES:

1. Axes names are entered into the **PSO-- > Fire** command as follows:
   1 for the X axis
   2 for the Y axis
   3 for the Z axis
   4 for the U axis

   A maximum of three (3) axes may be designated simultaneously.

2. The Position Counters translate the value of the Firing Distance "n", as follows:

   For a single axis: n = Counter 1

   For two axes: $n = \sqrt{\text{Counter }1^2 + \text{Counter }2^2}$

   For three axes: $n = \sqrt{\text{Counter }1^2 + \text{Counter }2^2 + \text{Counter }3^2}$

PSO-- > COND

15-3-2-4: Conditions

cond Select **cond** from the initial PSO menu to enable or disable Position Counter Tracking, based on the Input Bus. Refer to Example Programs 2 and 9.

The Conditional (PSO C or G80 C) command is used in conjunction with the PSO F/G80 F, PSO command.

Following selection of **cond**, the display will be:

```
    INPUT CONDITION



  ins-Y/N      del      ,      help    Varb      quit      m
```

Select **help** to display the available Input Conditions.

```
    INPUT CONDITION
 0        : ignore input condition
 4,i,o    : i*8 input lines, o*8 output
            at most, 24 I/O lines
 >
  ins-Y/N      del      ,      help    Varb      quit      m
```

Select **help** a second time to display additional Input Conditions.

```
    INPUT CONDITION
 1,i,n    : enable counter when  i low (n=0)
            high (n< >0), if disable, keep counter
 2,i,n    : same as 1, except counter to 0
 >
  ins-Y/N      del      ,      help    Varb      quit      m
```

PSO-- > COND

Select **help** a third time to display additional Input Conditions.

```
    INPUT CONDITION
3,iii,ooo   : enable when input pattern
              = iii, otherwise disable, & set output
              to pattern ooo, 0 low, 1 high, 2 skip
>
 ins-Y/N       del        ,       help     Varb       quit       m
```

Following is a description of each of the Input Condition options:

0 Input signal conditions are ignored (Default).

Example

Enter **0** the program block will be:
PSO C,0 or G80 C,0

Tracking is not affected by the Input Bus.

4,i,o Configures the 24 Bit I/O Bus to determine the number of Input and Output lines assigned. (Inputs are justified to the Least Significant Bit.) Following are the available configurations of the I/O Bus:

If

i = 0, and o = 3 Bits 0-23 are configured as Outputs
i = 1, and o = 2 Bits 0-7 are configured as Inputs, Bits 8-23 are configured as Outputs
i = 2, and o = 1 Bits 0-15 are configured as Inputs, Bits 16-23 are configured as Outputs
i = 3, and o = 0 Bits 0-23 are configured as Inputs

NOTE: **The sum of the value of "i" and "o" must be equal to 3.**

Example

Enter **4,1,2** the program block will be:
PSO C,4,1,2 or G80 C,4,1,2

Bits 0-7 are configured as Inputs, Bits 8-23 are configured as Outputs.

PSO-- > COND

1,i,n Position Tracking is enabled only when an Input "i" (0 through 23) is "High" (n does not equal zero) or "Low" (n equals zero). (Refer to the PSO C,4/G80 C,4 command.) When the Position Tracking Counter is disabled, Counter data is retained.

Example

Enter **1,5,2** the program block will be:
PSO C,1,5,2 or G80 C,1,5,2

Position Tracking is enabled when Input 5 is "High". When Input 5 goes "Low", Position Tracking stops but Counter data remains.

2,i,n Position Tracking is enabled only when an Input "i" (0 through 23) is "High" (n does not equal zero) or "Low" (n equals zero). (Refer to the PSO C,4/G80 C,4 command.) When the Position Counter is disabled, Counter data is reset to zero.

Example

Enter **2,21,0** the program block will be:
PSO C,2,21,0 or G80 C,2,21,0

Position Tracking is enabled when Input 21 is "Low". When Input 21 goes "High", Position Tracking stops and Counter data is cleared.

3,iii,ooo Position Tracking is enabled only when Input Bits 0-7 are configured exactly as designated by iii, otherwise Position Tracking is disabled and the output pattern is set to the pattern established by ooo (0 = Low, 1 = High, x = Not Checked).

Example

Enter **3,xx1x0101,xxxx1100** the program block will be:
PSO C,3,3,xx1x0101,xxxx1100 or G80 C,3,3,xx1x0101,xxxx1100

Position Tracking is enabled when input signals are as follows:
Bits 0,2,5 are high, bits 1,3 are low, bits 4,6,7 are not checked.
Otherwise the function is disabled and the output signal will be:
Bits 0,1 will be set low, bits 2,3 will be set high, bits 4,5,6,7 will remain the same.

PSO-- > OUTPUT

15-3-2-5: Output

output Select **output** from the initial PSO menu to configure the four D/A outputs or Binary output bits. The Unidex 400 is standardly equipped with Bipolar DAC outputs , if Unipolar DAC outputs are desired, contact an Aerotech Service Representative for information. Refer to Example Programs 2,3, and 10.

The Conditional (PSO T or G80 T) command is used in conjunction with the PSO F/G80 F, PSO command.

Following selection of **output**, the display will be:

```
OUTPUT



ins-Y/N     del     ,     help     Varb     quit     m
```

Select **help** to display the available Output Conditions.

```
OUTPUT
0,b,n...   : output bit b set to low n = 0
             or high n < > 0,
1,h        : output pattern = Hex data  h
>
ins-Y/N     del     ,     help     Varb     quit     m
```

Select **help** a second time to display additional Output Conditions.

```
OUTPUT
2,d,n...   : bipolarD/A to +-n volts
             at most 4 D/A
3,d,n,...  : unipolar D/A to +n volts
>
ins-Y/N     del     ,     help     Varb     quit     m
```

Select **help** a third time to display additional Output Conditions.

```
OUTPUT
4,d,D,M,V.    : bipolarD/A  D = voltage when
   vel = 0, M = max volt, V = vel   at M
5,d,D,M,V     : unipolar
>
ins-Y/N      del      ,      help     Varb      quit      m
```

Following is a description of each of the Output Condition options:

0,b,n,... Establishes output bit "b" as either high or low "n".
If "n" = 0 then the output will be low.
If "n" does not equal zero then output will be high.

Example

Enter **0,2,0,3,4,1,0** the program block will be:
PSO T,0,2,0,3,4,1,0 or G80 T,0,2,0,3,4,1,0

Output bits will be configured as follows:

Bit 2 is low
Bit 3 is high
Bit 1 is low

1,h Sets outputs equal to Hex data "h".

Example

Enter **1,F** the program block will be:
PSO T,1,F or G80 T1,F

Output bits are equal to Hex "F".

PSO-- > OUTPUT

2,d,n,... This selection is used to set the output voltage when the Unidex 400 is configured with Bipolar (11 bit + 1 sign) DAC outputs (Default).

d = DAC output line (0 to 3)

n = -10V to +10V output voltage

The output voltage has a minimum step size of 4.88 mV.

NOTE: **Bipolar DACs are installed standard.**

Example

Enter **2,2,V0** the program block will be:

PSO T,2,2,V0 or G80 T,2,2,V0

Bipolar DAC output line 2 set for the voltage contained in variable 0.

3,d,n,... This selection is used to set the output voltage only when the Unidex 400 is equipped with Unipolar 12 bit DAC outputs.

d = DAC output line (0 to 3)

n = 0V to +10V output voltage

The output voltage has a minimum step size of 2.4 mV.

NOTE: **This command is not standard and should only be used with Unidex 400's that are specially equipped with Unipolar DACs.**

Example

Enter **3,0,5** the program block will be:

PSO T,3,0,5 or G80 T,3,0,5

Unipolar DAC output line 0 set for 5 volts.

PSO-- > OUTPUT

4,d,D,M,V This selection is used to provide an output voltage proportional to the velocity of the User specified axis, when the Unidex 400 is configured with Bipolar (11 bit + 1 sign) DAC outputs (Default).

d = DAC output line (0 to 3)
D = output voltage when velocity is zero (-10V to +10V)
M = maximum voltage at target velocity (-10V to +10V)
V = target velocity (-2^{23} to $+2^{23}$ (-1) machine steps per mSec.)

NOTE: Bipolar DACs are installed standard.

Example
Enter **4,3,0,10,1000** the program block will be:
PSO T,4,3,0,10,1000 or G80 T,4,3,0,10,1000

Bipolar DAC output line 3 configured for 0 volts at 0 velocity, and a velocity of 1000 machine steps per millisecond at the maximum voltage of 10 volts.

NOTE: Both the Velocity ramping mode and the Position ramping mode may be active simultaneously. Setting the the Velocity and Position mode outputs for the same line will result in the summation of the two signals at the specified output.

5,d,D,M,V This selection is used to provide an output voltage proportional to the velocity of the User specified axis, when the Unidex 400 is configured with a Unipolar 12 bit DAC output configuration.

d = DAC output line (0 to 3)
D = output voltage when velocity is zero (0V to +10V)
M = maximum voltage at target velocity (0V to +10V)
V = target velocity (-2^{23} to $+2^{23}$ (-1) machine steps per m Sec.)

NOTE: This command is not standard and should only be used with Unidex 400's that are specially equipped with Unipolar DACs.

Example
Enter **5,2,5,10,5000** the program block will be:
PSO T,5,2,5,10,5000 or G80 T,5,2,5,10,5000

Unipolar DAC output line 2 set for 5 volts at 0 velocity, and a velocity of 5000 machine steps per millisecond at the maximum voltage of 10 volts.

PSO-- > OUTPUT

6,d,D,M,P This selection is used to provide a position that is proportional to the maximum output voltage, when the Unidex 400 is configured with Bipolar (11 bit + 1 sign) DAC outputs (Default).

d = DAC output line (0 to 3)
D = output voltage when velocity is zero (-10V to +10V)
M = maximum voltage at target position (-10V to +10V)
P = target position (-2^{23} to $+2^{23}$ (-1) machine steps)

NOTE: Bipolar DACs are installed standard.

Example

Enter **6,2,0,10,500** the program block will be:
PSO T,6,2,0,10,500 or G80 T,6,2,0,10,500

Bipolar DAC output line 2 set for 0 volts at 0 velocity, a move of 500 machine steps will initiate a maximum output of 10 volts.

NOTE: Both the Velocity ramping mode and the Position ramping mode may be active simultaneously. Setting the the Velocity and Position mode outputs for the same line will result in the summation of the two signals at the specified output.

7,d,D,M,P This selection is used to provide a position that is proportional to the maximum output voltage, when the Unidex 400 is configured with a Unipolar 12 bit DAC output.

d = DAC output line (0 to 3)
D = output voltage when velocity is zero (0V to +10V)
M = maximum voltage at target position (0V to +10V)
P = target position (-2^{23} to $+2^{23}$ (-1) machine steps)

NOTE: This command is not standard and should only be used with Unidex 400's that are specially equipped with Unipolar DACs.

Example

Enter **7,1,5,10,2000** the program block will be:
PSO T,7,1,5,10,2000 or G80 T,7,1,5,10,2000

Unipolar DAC output line 1 set for 5 volts at 0 velocity, a move of 2000 machine steps will initiate a maximum output of 10 volts.

15-3-2-6: Rtime

Rtime Select **Rtime** from the initial PSO menu to configure the PSO Board's Position Counter.

The Conditional (PSO R or G80 R) command is used in conjunction with PSO P/G80 P, PSO F/G80 F, PSO D/G80 D, and the PSO M/G80 M commands.

Following selection of **Rtime**, the display will be:

```
REAL-TIME CONTROL



ins-Y/N     del     ,     help     Varb     quit     m
```

Select **help** to display the available Real-Time Position Counter Conditions.

```
    REAL-TIME CONTROL
0       : clear all  real time control
          re-enable counters
          see case 1 & 3
>
 ins-Y/N     del     ,     help     Varb     quit     m
```

Select **help** a second time to display additional Real-Time Position Counter Conditions.

```
    REAL-TIME CONTROL
1       : stop counter, keep counter data
2,i,n   : same as 1, trig by interrupt  i
          abort case,  n = 0  disable,  < >0  enable
>
 ins-Y/N     del     ,     help     Varb     quit     m
```

PSO-- > REAL TIME

Select **help** a third time to display additional Real-Time Position Counter Conditions.

```
REAL-TIME CONTROL
3      : stop counter, keep counter data
4,i,n  : same as 3, trig by interrupt i
         abort case, n = 0 disable, < >0 enable
>
ins-Y/N     del     ,     help     Varb     quit     m
```

Following is a description of each of the Real-Time Position Counter Condition options:

0 This selection is used to clear previously established Real-Time Position Counter Conditions. PSO Tracking will resume, the counter will have retained or cleared the previous axis position as specified by the stop PSO Tracking command that had been used.

NOTE: PSO R,2 and PSO R,4 are modal commands and as such will remain in effect until cleared by entering the command with "n" equal to zero.

Example

Enter **0** the program block will be:

PSO R,0 or G80 R,0

PSO Tracking will resume, counter configuration will be in accordance with that established by PSO R,1 or PSO R,3.

PSO-- > REAL TIME

1 This selection is used to stop PSO Tracking of an axis. The current axis position is retained in the counter.

Example

Enter **1** the program block will be:
PSO R,1 or G80 R,1

PSO Tracking will be disabled , the current axis position will remain in the counter.

2,i,n This selection is used to designate an Interrupt signal that will stop PSO Tracking of an axis. The current axis position is retained in the counter.

i = Interrupt signal 1, 2, 3 or 4

NOTE: PSO R,2 and PSO R,4 are valid only for Interrupt signals that abort the current program block and immediately process the designated subroutine. (Refer to Chapter 3, Pages 3-76 through 3-78 or Pages 3-85 through 3-87 for information concerning Interrupt - abort.)

If "n" does not equal zero,
Interrupt "i" will be used to generate a PSO R,1 command.

If "n" equals zero,
The previous PSO R,2,i,1 command is cleared.

Example

Enter **2,1,1** the program block will be:
PSO R,2,1,1 or G80 R,2,1,1

PSO Tracking will be disabled upon receipt of a signal through Interrupt 1, the current axis position will remain in the counter.

PSO-- > REAL TIME

3 This selection is used to stop PSO Tracking of an axis. The current axis position is cleared from the counter.

Example

Enter **3** the program block will be:
PSO R,3 or G80 R,1

PSO Tracking will be disabled, the counter is cleared of previous position data.

4,i,n This selection is used to designate an Interrupt signal that will stop PSO Tracking of an axis. The axis position data is cleared from the counter.

i = Interrupt signal 1, 2, 3 or 4

NOTE: PSO R,2 and PSO R,4 are valid only for Interrupt signals that abort the current program block and immediately process the designated subroutine. (Refer to Chapter 3, Pages 3-76 through 3-78 or Pages 3-85 through 3-87 for information concerning Interrupt - abort.)

If "n" does not equal zero,
Interrupt "i" will be used to generate a PSO R,3 command.

If "n" equals zero,
The previous PSO R,4,i,1 command is cleared.

Example

Enter **4,3,1** the program block will be:
PSO R,4,3,1 or G80 R,4,3,1

PSO Tracking will be disabled upon receipt of a signal through Interrupt 3, the counter is cleared of previous position data.

15-3-2-7: Bmap

Bmap Select **Bmap** from the initial PSO menu to establish a condition such that Pulse Train Output occurs in accordance with a bit pattern located in a previously established variable string. Refer to Example Program 8.

The Bit Mapping Command (PSO M or G80 M) command is used in conjunction with PSO P/G80 P, PSO F,4/G80 F,4, and the PSO R/G80 R commands.

Following selection of **Bmap**, the display will be:

```
BIT MAPPING



ins-Y/N     del     ,     help    Varb     quit     m
```

Select **help** to display the available Bmap Conditions.

```
BIT MAPPING
0 ,nn,mm     : from var nn, + -mm sets


>
ins-Y/N     del     ,     help    Varb     quit     m
```

PSO-- > BIT MAPPING

Following is a description of the Bit Mapping configuration selection:

0 ,nn,mm This selection is used to identify the variables that contain the bit patterns to be used in mapping.

Bit mapping will begin with the bit pattern defined in variable number "n" and continue "m" number of bytes.

Example

Enter **0,3.40** the program block will be:
PSO M,0,3,40 or G80 M,0,3,40

Bit Mapping will start with the byte defined in variable 3 and continue for 40 bytes (10 consecutive variables total).

15-3-3 PSO EXAMPLE PROGRAMS

Example 1: Pulse Train specifications defined in variables.

Machine Tool	Menu Driven	Comments
G30 ARRAY PULSE TRAIN EXAMPLE		;program title
G97 V0=1	V0=1	;variable 0 contains a pulse off time of 1 mSec.
G97 V1=2	V1=2	;variable 1 contains a pulse on time of 2 mSec.
G97 V2=3	V2=3	;variable 2 contains a pulse off time of 3 mSec.
G97 V3=4	V3=4	;variable 3 contains a pulse on time of 4 mSec.
G97 V4=5	V4=5	;variable 4 contains a pulse off time of 5 mSec.
G80 P,3,0,5	PSO P,3,0,5	;start pulse train with variable 0 and continue for five elements
G80 F,1	PSO F,1	;fire continuously

Example 2: Set all PSO data bits as outputs and output (Hex) data.

Machine Tool	Menu Driven	Comments
G30 BINARY OUTPUT BUS EXAMPLE		;program title
G97 V0=H,0	V0=H,0	;variable 0 contains Hex 0
G80 C,4,0,3	PSO C,4,0,3	;all bits set as outputs
N100	Label 100	;label
G80 T,1,V0	PSO T,1,V0	;set outputs to Hex data in variable 0
G97 V0=V0.ADD4.H,1	V0=V0.ADD4.H,1	;next data
G60 L100	GOTO LABEL 100	;jump to label "100"

Example 3: Ramp all DAC's from -10 to +10 Volts

Machine Tool	Menu Driven	Comments
G30 BIPOLAR DAC OUTPUT EXAMPLE		;program title
G97 V0=-10	V0=-10	;variable 0 contains starting (lowest) voltage
N100	Label 100	;label
G97 V0=V0+.1	V0=V0+.1	;increment variable 0 value by 1/10 volt
G21 PSO DAC OUTPUT VOLTAGE #V0		;display message
G80 T,2,0,V0,1,V0,2,V0,3,V0		;set all Bipolar DACs to voltage of variable 0
G62 L100 C0<10.		;jump to label 100 if variable 0 is less than 10.
G97 V0=-10.	V0=-10.	;restart
G60 L100	GOTO LABEL 100	;jump to label "100"

Example 4: Fixed firing distance and pulse width tracking.

Machine Tool	Menu Driven	Comments
G30 SINGLE AXIS TRACKING EXAMPLE		;program title
G80 D,0,1000	PSO D,0,1000	;fire every 1000 machine steps
G80 P,0,3	PSO P,0,3	;3 mSec pulse width
G80 F,3,1	PSO F,3,1	;track on X axis
G10 X1 F500.	Free: cont X1 F500	;continuous CW freerun of the X axis at 500 units/min.

Example 5: Fire "n" number of User defined Pulse Trains.

Machine Tool	Menu Driven	Comments
G30 ONE SHOT PULSE TRAIN		;program title
G80 P,1,1,3,1	PSO P,1,1,3,1	;define pulse train, 1mSec lead, 3 mSec pulse width, 1 mSec trail
G80 F,2,1000	PSO F,2,1000	;fire 1000 pulse trains

Example 6: Use of arrayed firing distance.

Machine Tool	Menu Driven	Comments
G30 ARRAY FIRING DISTANCE		;program title
G97 V0=100	V0=100	;variable 0 contains first firing distance
G97 V1=200	V1=200	;variable 1 contains second firing distance
G97 V2=300	V2=300	;variable 2 contains third firing distance
G97 V3=400	V3=400	;variable 3 contains fourth firing distance
G80 P,0,1	PSO P,0,1	;1 mSec pulse width
G80 D,1,0,4	PSO D,1,0,4	;firing distance contained in V0, 4 elements (V1,V2,V3)
G80 F,3,1	PSO F,3,1	;enable firing, track on X axis
G10 X1 F100.	Free: cont X1 F100	;continuous CW freerun of the X axis at 100 units/min.

Example 7: Tracking on a circular contour.

Machine Tool	Menu Driven	Comments
G30 TWO AXIS TRACKING EXAMPLE		;program title
G80 P,4,100	PSO P,4,100	;1 shot pulse width of 100 User units
G1 F100.	linear F100.	;linear move with feedrate 100. unit/min.
G8 G1 X1.	vel linear X1.	;enables velocity profiling for linear contouring
G80 F,3,1,2	PSO F,3,1,2	;enable firing, track on X and Y axes
G8 G2 X0 Y0 I0 J-1.	vel cw X0 Y0 I0 J-1.	;enables velocity profiling for CW circular contour
G80 F,0	PSO F,0	;disable firing and tracking
G1 X1.	linear X1.	;linear move of X axis, 1. unit/min.

Example 8: Bit map firing example.

Machine Tool	Menu Driven	Comments
G30 BIT MAP FIRING		;program title
G97 V0=H,AAAAAAAA	V0=H,AAAAAAAA	;variables 0-9 define the firing toggle sequence. Variables must be defined as integers. Each variable is 32 bits (4 bytes)
G97 V0=H,AAAAAAAA	V0=H,AAAAAAAA	
G97 V1=H,0	V1=H,0	
G97 V2=H,AAAAAAAA	V2=H,AAAAAAAA	
G97 V3=H,0	V3=H,0	
G97 V4=H,AAAAAAAA	V4=H,AAAAAAAA	
G97 V5=H,0	V0=H,0	
G97 V6=H,AAAAAAAA	V6=H,AAAAAAAA	
G97 V7=H,0	V0=H,0	
G97 V8=H,AAAAAAAA	V8=H,AAAAAAAA	
G97 V9=H,0	V9=H,0	
N100	Label 100	;label
G80 M,0,0,40	PSO M,0,0,40	;set bit map pattern, beginning with V0, +40 bytes
G80 F,4,100,1	PSO F,4,100,1	;track on X axis, firing distance is 100 machine steps
G1 X32. F600	linear X32. F600	;linear contour move of the X axis
G4 F100	Dwell 100 ms	;system pause for 100mSec.
G80 M,0,0,40	PSO M,0,0,40	;set bit map pattern, beginning with V0, +40 bytes
G80 F,4,-100,1	PSO F,4,-100,1	;reverse bit patterns
G1 X-32.	linear X-32.	;return X axis
G4 F100	Dwell 100 ms	;system pause for 100mSec.
G60 L100	GOTO LABEL 100	;jump to label "100"

Example 9: Conditional firing

Machine Tool	Menu Driven	Comments
G30 CONDITIONAL FIRING		;program title
G80 C,1,0,0	PSO C,1,0,0	;position tracking enabled when input 0 is low
G80 D,0,1000	PSO D,0,1000	;fire every 1000 machine steps
G80 P,0,1	PSO P,0,1	;1 mSec pulse width
G80 F,3,1	PSO F,3,1	;track on X axis
G1 X1. F500.	linear X1 F500	;linear contour move of the X axis

Example 10: Analog Output - Velocity Ramping - Bipolar DAC

Machine Tool	Menu Driven	Comments
G30 BIPOLAR DAC VELOCITY RAMPING EXAMPLE		;program title
G97 V0=300	V0=300	;variable 0 contains first firing distance
G97 V1=700	V1=700	;variable 1 contains second firing distance
G80 T,4,0,3,7,125	PSO T,4,0,3,7,125	;bipolar velocity mode, output D/A 0, 3 volt offset, 7 volt output at 125 machine steps/mSec
G80 P,0,1	PSO P,0,1	;1 mSec pulse width
G80 D,1,0,2	PSO D,1,0,2	;incremental firing distance start with distance in variable 0, 2 elements
G80 F3,X,Y	PSO F3,X,Y	;enable firing, track vector velocity in the X/Y plane

SECTION 15-4: RESOLUTION MULTIPLIER OPTION (RMX)

The RMX Board is a factory installed option that provides resolution multiplication by increasing the effective line count of an incremental encoder by a factor of 256.

15-4-1: HARDWARE CONFIGURATION

The RMX Board is mounted directly to the Unidex 400's DSP Board. A Reset Switch and LED are located on the front edge of the board. The Reset Switch should not need to be pressed under normal operating conditions. The LED will come on for approximately .5 seconds following a power up or reset.

JUMPERS AND SWITCHES

The RMX Board contains several jumpers that have been factory set prior to shipment. The following list of Jumpers is provided for reference only. (Refer to Figure 15-9 for relative locations of the RMX Board components.)

Jumper	Condition	Function
JP1		Not Applicable
JP2	1-2	Configures Limit Polarity as active High.
	2-3	Configures Limit Polarity as active Low. (Default)
JP3	1-2	Configures Limit Polarity as active High.
	2-3	Configures Limit Polarity as active Low. (Default)
JP4	1-2	Configures Limit Polarity as active High.
	2-3	Configures Limit Polarity as active Low. (Default)
JP5	1-2	Configures Limit Polarity as active High.
	2-3	Configures Limit Polarity as active Low. (Default)

Jumper	Condition	Function
JP16	1-2	Developmental use only. (Default)
	2-3	Developmental use only.
JP17	1-2	Enables Watchdog Timer.
	2-3	Disables Watchdog Timer. (Default)
JP18	1-2	Disables Battery Backup.
	2-3	Enables Battery Backup. (Default)
JP20	1-2	Disables Battery Backup.
	2-3	Enables Battery Backup. (Default)
JP21	1-2	Disables Battery Backup.
	2-3	Enables Battery Backup. (Default)
JP22	1-2	Disables Battery Backup.
	2-3	Enables Battery Backup. (Default)
JP23	1-2	Disables Battery Backup.
	2-3	Enables Battery Backup. (Default)

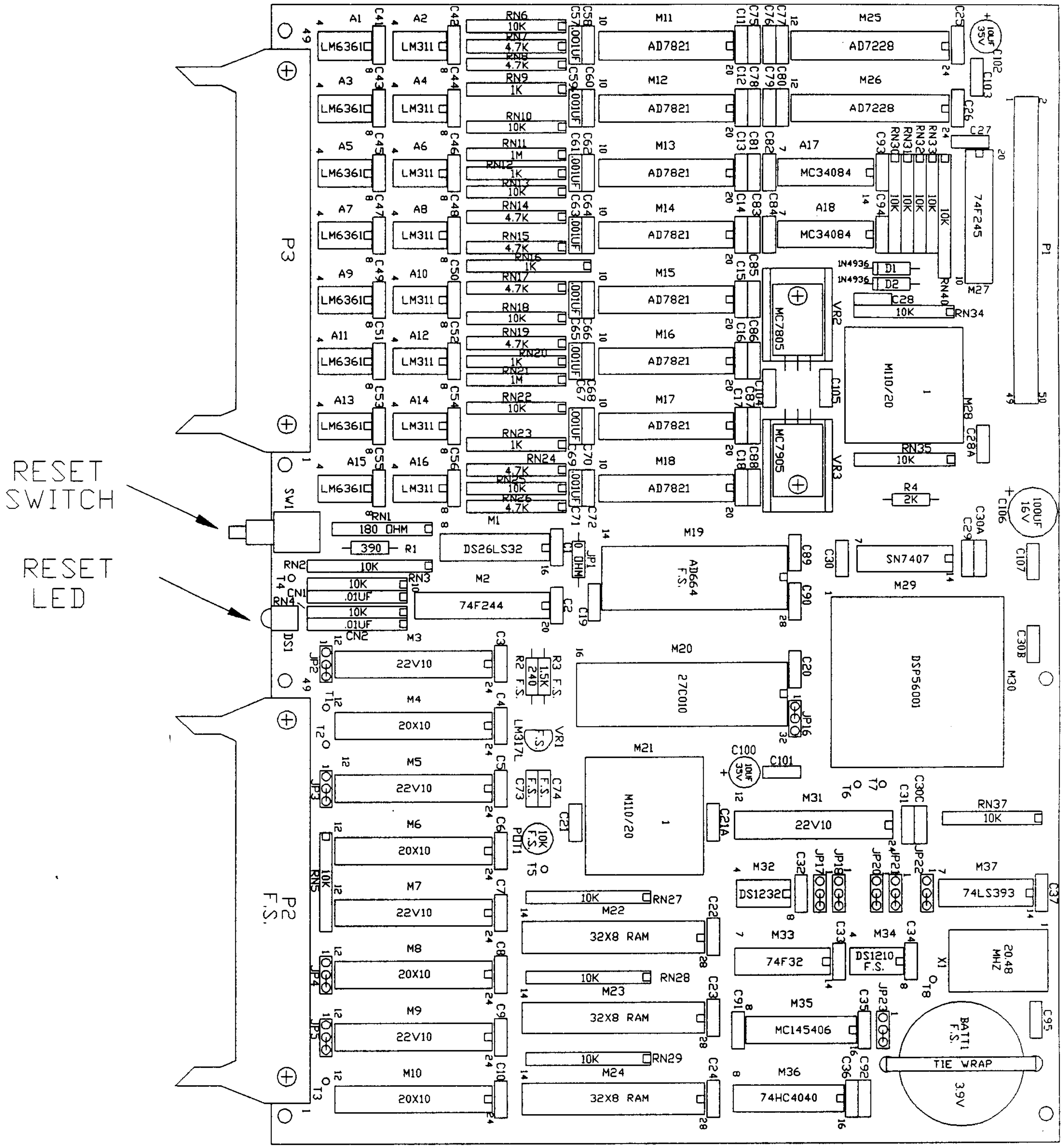


Figure 15-9: Unidex 400 RMX Board Assembly

RMX INTERFACE

The RMX Board is internally interfaced to the Unidex 400's DSP Board. The RMX Board's P1 Connector connects directly to the DSP's P5 mating connector. This connection provides the +5V, +12V and -12V power requirements of the RMX Board. The Encoder/Limit ribbon cable interfaced to the DSP Board's P3 connector has been extended to make a parallel connection to the RMX Board's P3 connector. This connector carries the signals designated below:

Pin	Signal	Pin	Signal
1	Limit 1+	2	Limit 1-
3	No Connection	4	Ground
5	Sine 1	6	Sine 1-N
7	Cosine	8	Cosine 1-N
9	Marker 1	10	Marker 1-N
11	Ground	12	Ground
13	Limit 2+	14	Limit 2-
15	No Connection	16	Ground
17	Sine 2	18	Sine 2-N
19	Cosine 2	20	Cosine 2-N
21	Marker 2	22	Marker 2-N
23	Ground	24	Ground
25	Limit 3+	26	Limit 3-
27	No Connection	28	Ground
29	Sine 3	30	Sine 3-N
31	Cosine 3	32	Cosine 3-N
33	Marker 3	34	Marker 3-N
35	Ground	36	Ground
37	Limit 4+	38	Limit 4-
39	No Connection	40	Ground
41	Sine 4	42	Sine 4-N
43	Cosine 4	44	Cosine 4-N
45	Marker 4	46	Marker 4-N
47	Ground	48	Ground
49	No Connection	50	No Connection

15-4-2: USING THE RMX OPTION

The RMX option requires no User interface during operation, however, prior to using the RMX option the system must be configured to recognize RMX input for the specified axes. The following paragraphs provide instructions for setting the applicable parameters. For a detailed description of all Parameter settings refer to Chapter 7 of this manual.

The Initial Selection Screen shown below will be displayed upon power up:

```
    UNIDEX 400                       Version x.xx

      RAM:ok    ROM:ok    SETUP:ok
      User RAM: xxxxxx

edit      file      setup      test      machine      PSWD
```

NOTE: **The PSWD option (F6) is displayed only if the Password function has been activated from the Parameter Mode.**

Enter the Setup Mode by selecting **setup** (F3), the display will be:

```
    Set Up Mode



    save      load      axes      general      tune      quit
```

Axis Parameters

Select **axes** to alter the axis parameters, the display will be:

```
x00 : METRIC SCALE FACTOR
x01 : ENGLISH SCALE FACTOR
x02 : HOME DIRECTION IS CCW ?                         0/1
 x= 1,2,3,4 for Axis X,Y,Z,U
item:                         data:
ins-Y/N     del     item            preset      quit      m
```

The following axis parameters must be configured appropriately prior to using the RMX option.

x33 : FEEDBACK DEVICE TYPE

This parameter is used to specify the type of device that the Feedback Servo Loop will use. The options are; 0-Encoder, 1-Resolver, 2-Resolution Multiplier Board, 3-Open Loop (no feedback).

If the RMX option is included in the Unidex 400 being used, make certain this parameter is set to "2".

x35 : BRUSHLESS COMMUTATION FACTOR

This parameter is used only when a brushless type motor is being used. The commutation factor is derived as follows:

> If the feedback device is an encoder or a RMX Board, the commutation factor is the number of machine steps per electrical cycle.
>
> If the feedback device is a resolver, the commutation factor is the number of electrical cycles per revolution.

If the RMX option is included in the Unidex 400 being used, enter the number of machine steps per electrical cycle into this parameter.

NOTE: **Prior to using the Unidex 400 with the RMX option, all applicable system parameters must be appropriately configured and the system must have been reset or powered down to initialize the data.**

SECTION 15-5: RESOLVER TO DIGITAL (R/D) OPTION BOARD

The R/D option board provides an interface for up to four axes of Resolver or Inductosyn transducer feedback devices. These devices generate sinusoidal outputs as a function of their orientation which the R/D board converts into a parallel digital word. The resolution of the conversion is factory set for either 10, 12, 14, or 16 bits per resolver/inductosyn cycle.

15-5-1: HARDWARE CONFIGURATION

The optional R/D Board is mounted directly to the Unidex 400's DSP Board via a 50 pin header. In the event that servicing the R/D card is necessary, both cards must be removed from the system as a unit and carefully separated.

JUMPERS AND SWITCHES

The R/D Board contains several jumpers that have been factory set for the type of transducer being used (i.e. resolver or inductosyn). The following list of Jumpers is provided for reference only. (Refer to Figure 15-10 for relative locations of the R/D Board components.)

Jumper	Condition	Function
JP1	1-2	Configures U axis reference frequency for resolver feedback (Default)
	2-3	Configures U axis reference frequency for inductosyn feedback
JP2	1-2	Configures Z axis reference frequency for resolver feedback (Default)
	2-3	Configures Z axis reference frequency for inductosyn feedback
JP3	1-2	Configures Y axis reference frequency for resolver feedback (Default)
	2-3	Configures Y axis reference frequency for inductosyn feedback
JP4	1-2	Configures X axis reference frequency for resolver feedback (Default)
	2-3	Configures X axis reference frequency for inductosyn feedback
JP5 through 9		Factory use only

The R/D Board also contains two potentiometers designated R51 and R52. Potentiometer R51 is used to compensate for variability in peak amplitude between feedback devices by increasing or decreasing the amplitude of the reference frequency. This potentiometer is adjusted to provide 2 volts peak amplitude on the R/D converter. These signals may be monitored at pins 6,7,10,11,13,14,15 or 16 of the following differential line receivers:

A16/A15 - Cos/Sin Channel 1
A14/A13 - Cos/Sin Channel 2
A11/A12 - Cos/Sin Channel 3
A7/A6 - Cos/Sin Channel 4

Potentiometer R52 is used to increase or decrease the amount of phase shift to the reference frequency when inductosyn feedback is used.

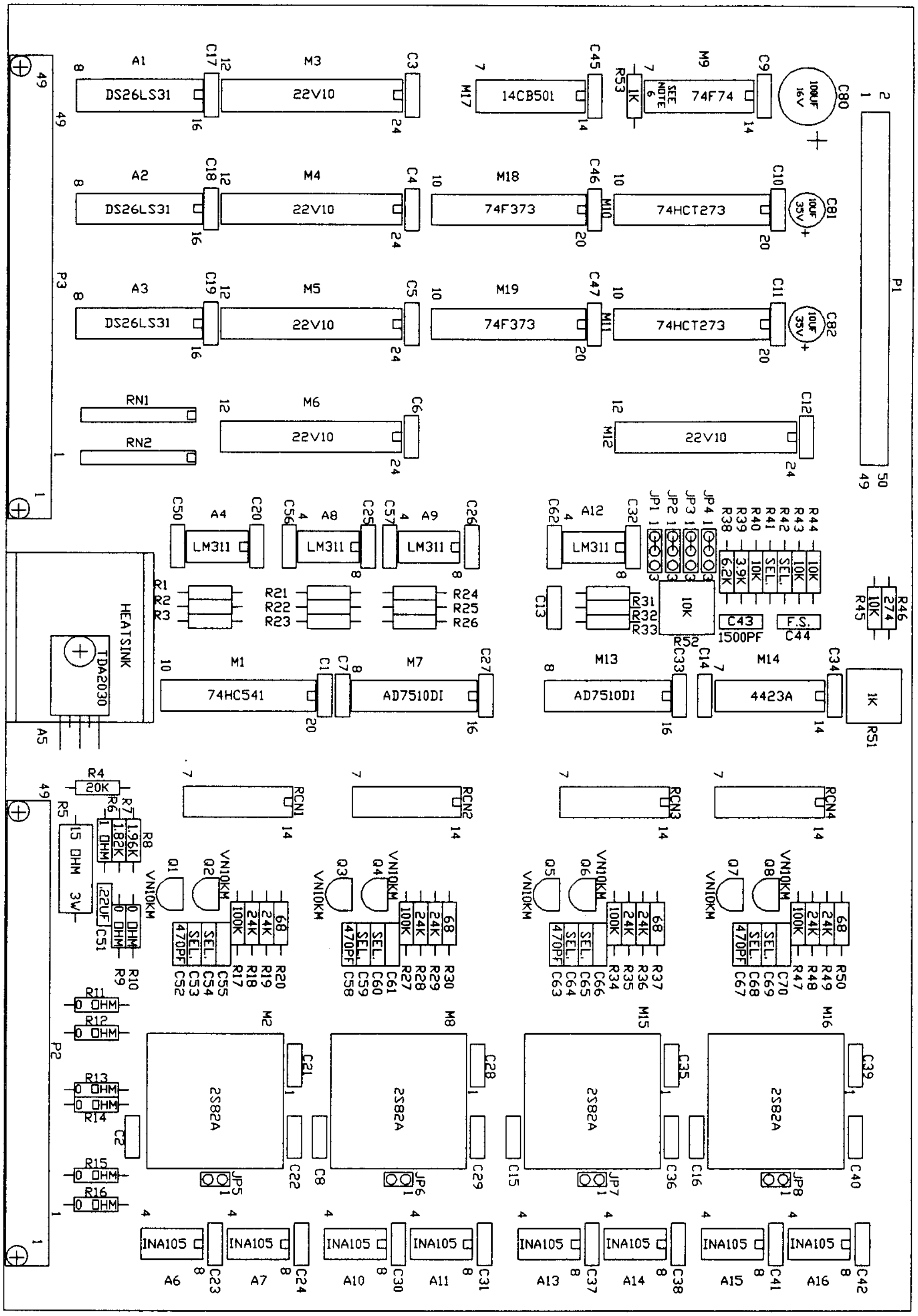


Figure 15-10: R/D Board Assembly

R/D INTERFACE

The R/D Board has two 50 pin ribbon cable connectors located on it's front edge. The bottom connector, labeled P2, contains the Resolver/Inductosyn interface signals. The top connector, labeled P3, provides an optional quadrature square wave output for each of the four R/D channels (Note: The R/D Board must be factory configured to enable the quadrature output function.

These connectors carry the signals listed below:

P2 Connector

Pin	Signal
1	-Cosine 4
2	Reference Ground Channel 4
3	+Cosine 4
4	Ground
5	Sine Shield Channel 4
6	Reference Channel 4
7	-Sine 4
8	Cosine Shield Channel 4
9	+Sine 4
10	No Connection
11	No Connection
12	No Connection
13	-Cosine 3
14	Reference Ground Channel 3
15	+Cosine 3
16	Ground
17	Sine Shield Channel 3
18	Reference Channel 3
19	-Sine 3
20	Cosine Shield Channel 3
21	+Sine 3
22	No Connection
23	No Connection
24	No Connection

Pin	Signal
25	-Cosine 2
26	Reference Ground Channel 2
27	+Cosine 2
28	Ground
29	Sine Shield Channel 2
30	Reference Channel 2
31	-Sine 2
32	Cosine Shield Channel 2
33	+Sine 2
34	No Connection
35	No Connection
36	No Connection
37	-Cosine 1
38	Reference Ground Channel 1
39	+Cosine
40	Ground
41	Sine Shield Channel 1
42	Reference Channel 1
43	-Sine 1
44	Cosine Shield Channel 1
45	+Sine 1
46	No Connection
47	No Connection
48	No Connection
49	No Connection
50	Ground

P3 Connector

Pin	Signal
1	No Connection
2	No Connection
3	No Connection
4	Ground
5	+Sine 1
6	-Sine 1
7	+Cosine 1
8	-Cosine 1
9	+Marker 1
10	- Marker 1
11	Ground
12	Ground
13	No Connection
14	No Connection
15	No Connection
16	Ground
17	+Sine 2
18	-Sine 2
19	+Cosine 2
20	-Cosine 2
21	+ Marker 2
22	- Marker 2
23	Ground
24	Ground
25	No Connection
26	No Connection
27	No Connection
28	Ground
29	+ Sine 3
30	- Sine 3

Pin	Signal
31	+ Cosine 3
32	- Cosine 3
33	+ Marker 3
34	- Marker 3
35	Ground
36	Ground
37	No Connection
38	No Connection
39	No Connection
40	Ground
41	+ Sine 4
42	- Sine 4
43	+ Cosine 4
44	- Cosine 4
45	+ Marker 4
46	- Marker 4
47	Ground
48	Ground
49	No Connection
50	No Connection

15-4-2: USING THE R/D OPTION

The R/D option requires no User interface during operation, however, prior to using the R/D option the system must be configured to recognize R/D input for the specified axes. The following paragraphs provide instructions for setting the applicable parameters. For a detailed description of all Parameter settings refer to Chapter 7 of this manual.

The Initial Selection Screen shown below will be displayed upon power up:

```
UNIDEX 400                              Version x.xx

   RAM:ok    ROM:ok    SETUP:ok
   User RAM: xxxxxx

edit     file     setup     test     machine     PSWD
```

Enter the Setup Mode by selecting **setup** (F3), the display will be:

```
Set Up Mode



save     load     axes     general     tune     quit
```

Axis Parameters

Select **axes** to alter the axis parameters, the display will be:

```
x00 : METRIC SCALE FACTOR
x01 : ENGLISH SCALE FACTOR
x02 : HOME DIRECTION IS CCW ?                        0/1
 x= 1,2,3,4 for Axis X,Y,Z,U
item:                          data:
ins-Y/N     del     item            preset     quit     m
```

The following axis parameters must be configured appropriately prior to using the R/D option.

x33 : FEEDBACK DEVICE TYPE

This parameter is used to specify the type of device that the Feedback Servo Loop will use. The options are; 0-Encoder, 1-Resolver, 2-Resolution Multiplier Board, 3-Open Loop (no feedback).

If the R/D option is included in the Unidex 400 being used, make certain this parameter is set to "1".

x35 : BRUSHLESS COMMUTATION FACTOR

This parameter is used only when a brushless type motor is being used. The commutation factor is derived as follows:

> If the feedback device is an encoder or a RMX Board, the commutation factor is the number of machine steps per electrical cycle.
>
> If the feedback device is a resolver, the commutation factor is the number of electrical cycles per revolution.

If the R/D option is included in the Unidex 400 being used, enter the number of electrical cycles per revolution cycle into this parameter.

NOTE: **Prior to using the Unidex 400 with the R/D option, all applicable system parameters must be appropriately configured and the system must have been reset or powered down to initialize the data.**

SECTION 15-6: MPI INTERFACE CARD OPTION

The Aerotech MPI (GPIB) optional interface card may be configured to supply three interface options. These options are; the Opto 22 ™ PAMUX ™ I/O Bus, the IEEE-488 Communications card and a combination IEEE-488 and PAMUX ™ I/O Bus interface.

15-5-1: HARDWARE CONFIGURATION

The optional MPI Board (labeled A13) is connected to the Unidex 400's CPU Board through the VME interface. Various internal wiring and connector additions are necessary to install this option.

JUMPERS AND SWITCHES

The MPI Board contains several jumpers that have been factory set in accordance with the board's use. The following list of Jumpers is provided for reference only. (Refer to Figure 15-11 for relative locations of the MPI Board components.)

Jumper-In	Jumper-Out	IEEE-488 VME Interrupt Level
JP1	JP2 thru 6	Interrupt Level 1
JP2	JP1, JP3 thru JP6	Interrupt Level 2 (Default)
JP3	JP1,JP2, JP4 thru JP6	Interrupt Level 3
JP4	JP1 thru JP3, JP5,JP6	Interrupt Level 4
JP5	JP1 thru JP4,JP6	Interrupt Level 5
JP6	JP1 thru JP5	Interrupt Level 6

Jumper-In	Jumper-Out	Motorola I/O Interrupt Level
JP7	JP8 thru 12	Interrupt Level 1
JP8	JP7, JP9 thru JP12	Interrupt Level 2 (Default)
JP9	JP7,JP8,JP10 thru JP12	Interrupt Level 3
JP10	JP7 thru JP9, JP11,JP12	Interrupt Level 4
JP11	JP7 thru JP10,JP12	Interrupt Level 5
JP12	JP7 thru JP11	Interrupt Level 6

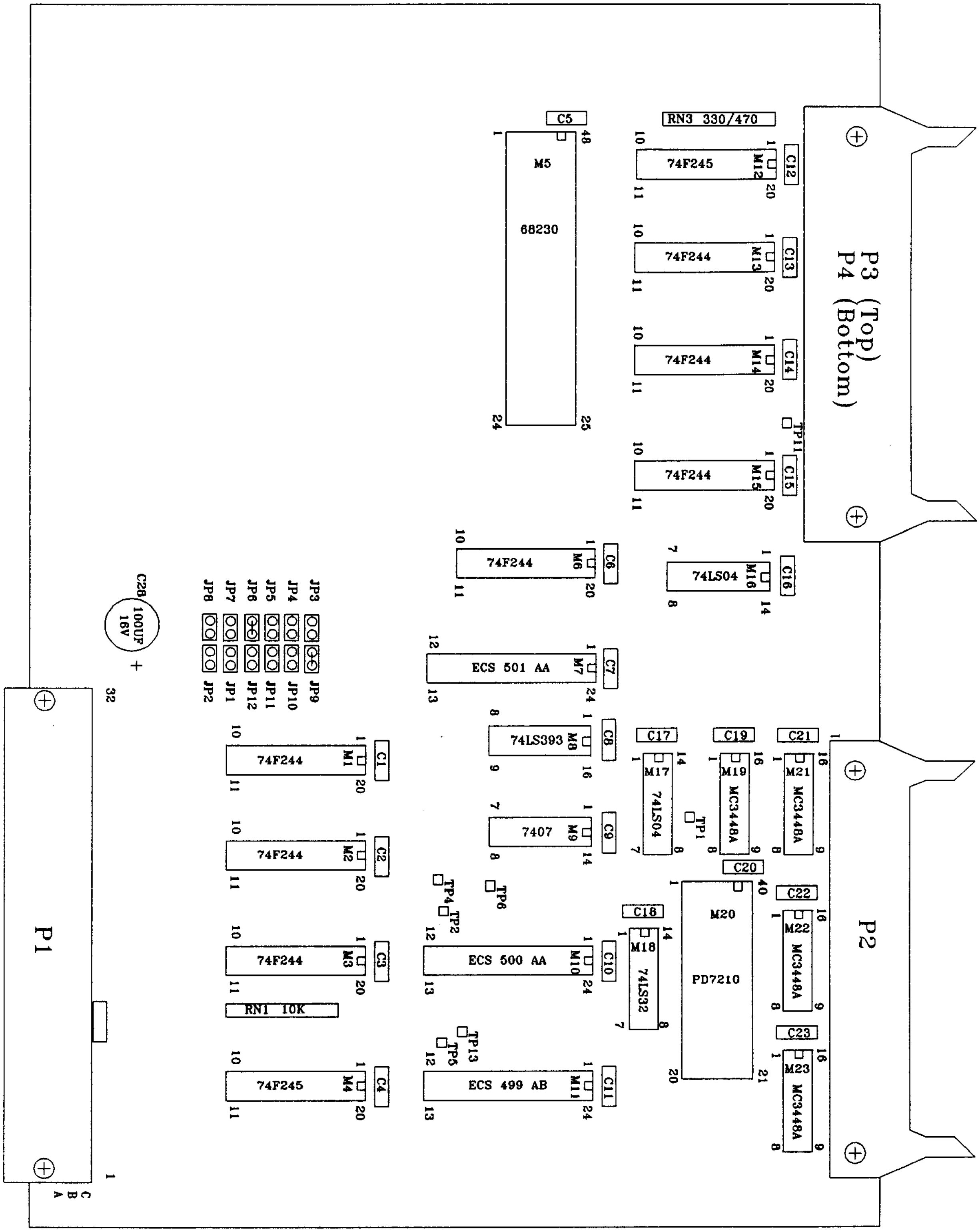
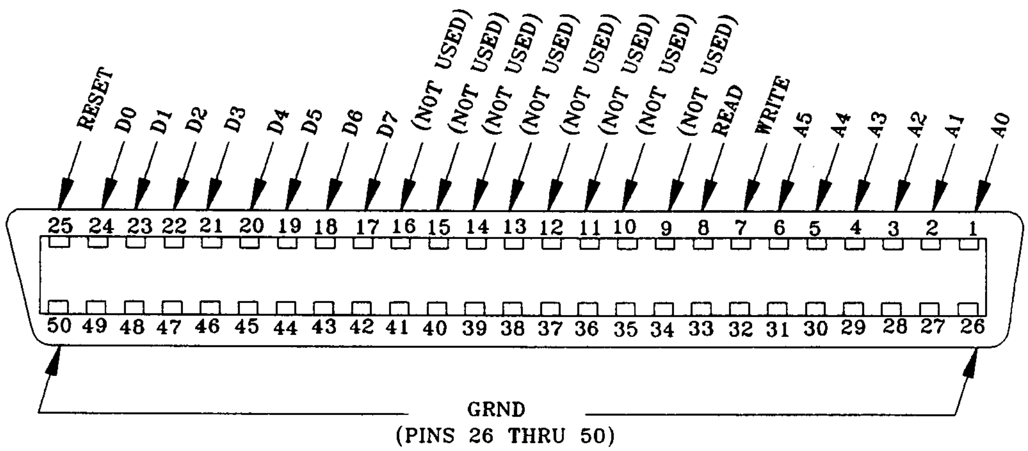


Figure 15-11: MPI Board Assembly

MPI INTERFACE

The MPI Board contains three interface busses; the PAMUX™ Digital Brain Board Interface, the IEEE-488/GPIB Interface and the Motorola I/O Interface (not used). A pin-out for the PAMUX™ interface follows. Refer to Chapter 11 of this manual for a description of the IEEE-488/GPIB connector.

The PAMUX Interface is designed to interface directly to the PAMUX™ B5 Digital Brain Board (refer to the PAMUX™ B5 Brain Board Manual for details). It contains six address lines, eight data lines and three control signals. This connector is labeled "P10" and is located in the Option section of the Unidex 400's Rear Panel.



Pin	Signal
1	Address
2	Address
3	Address
4	Address
5	Address
6	Address
7	Control
8	Control
9 thru 16	Not Used
17	Data
18	Data
19	Data
20	Data

Pin	Signal
21	Data
22	Data
23	Data
24	Data
25	Control
26 thru 50	Ground

15-5-2: USING THE MPI OPTION

User interface with the MPI Board option is dependent upon it's function. Following is a description of the two functions.

PAMUX USE

The PAMUX™ Bus contains 64 addressable locations. These are accessed by the Unidex 400 through reserved variables V9910 though V9973. V9910 is the lowest address (0) and V9973 is the highest address(63).

When reading or writing to the the MPI Board, all data is considered integer, not floating point. The FTB and BTF functions may be used to convert data types if necessary. All numeric values that are used in programming the Unidex 400 are considered floating point unless preceded by a "H," or "H:" (H = Hex) depending on the case. "H:" is used to display a variable in Hex format, all other cases use "H,".

Attempts to read or write to the PAMUX™ interface without an MPI Board in the system will result in a "BUS ERROR" message and the system will require resetting. The command "G97 V9910=H,01" will write binary 00000001 to the data bits D7 through D0. The command "G97 V0=V9910" will store the binary data to the variable V0.

Command Example:

G97 V9910=H,01	;write binary 00000001 to the data line D7 thru D0 at address 0
G97 V0=V9910	;read address 0 of PAMUX interface and store in variable 0.

The following Jumper configuration is required for the PAMUX™ B5 Brain Board.

```
0 1 2 3 4 5 6 7 8
X X X X X 1 1 1 0
```

- 8 — reset has no affect on watchdog
- 7 — active high reset
- 5 6 — disable watchdog timer
- 0 1 2 3 4 — User selectable address

PAMUX™ Base Address Jumper Configuration	Unidex 400 Low Byte Access	Unidex 400 High Byte Access
0 0 0 0 0	V9910	V9911
0 0 0 0 1	V9912	V9913
0 0 0 1 0	V9914	V9915
0 0 0 1 1	V9916	V9917
0 0 1 0 0	V9918	V9919
0 0 1 0 1	V9920	V9921
0 0 1 1 0	V9922	V9923
0 0 1 1 1	V9924	V9925
0 1 0 0 0	V9926	V9927
0 1 0 0 1	V9928	V9929
0 1 0 1 0	V9930	V9931
0 1 0 1 1	V9932	V9933
0 1 1 0 0	V9934	V9935
0 1 1 0 1	V9936	V9937
0 1 1 1 0	V9938	V9939
0 1 1 1 1	V9940	V9941
1 0 0 0 0	V9942	V9943
1 0 0 0 1	V9944	V9945
1 0 0 1 0	V9946	V9947
1 0 0 1 1	V9948	V9949

PAMUX™ Base Address Jumper Configuration	Unidex 400 Low Byte Access	Unidex 400 High Byte Access
1 0 1 0 0	V9950	V9951
1 0 1 0 1	V9952	V9953
1 0 1 1 0	V9954	V9955
1 0 1 1 1	V9956	V9957
1 1 0 0 0	V9958	V9959
1 1 0 0 1	V9960	V9961
1 1 0 1 0	V9962	V9963
1 1 0 1 1	V9964	V9965
1 1 1 0 0	V9966	V9967
1 1 1 0 1	V9968	V9969
1 1 1 1 0	V9970	V9971
1 1 1 1 1	V9972	V9973

1 = Jumper Installed
0 = Jumper Removed
X = User Selectable

PAMUX PROGRAM EXAMPLES

Example 1: Set one of 16 single output bits at a time.

Machine Tool	Menu Driven	Comments
G30 PB16 PAMUX WRITE PROGRAM		;program title
G97 V1=H,01	V1=H,01	;system variable 1 set to Hex 1
N1	label 1	;define label 1
G97 V9910=V1	V9910=V1	;send variable 1 to PAMUX address 0
G4 F500	dwell 500 ms	;dwell for 500 ms
G97 V1=V1.LSL4.H,1	V1=V1.LSL4.H,1	;shift active bit 1 place to the left
G62 L2 C1=H,100	GOTO LABEL 2 C1=H,100	;goes to label 2 if first 8 have been sent
G60 L1	GOTO LABEL 1	;go to label 1
N2	label 2	;define label 2
G97 V9910=H,0	V9910=H,0	;clear all 8 bits of PAMUX address 0
G97 V1=H,1	V1=H,1	;system variable 1 set to Hex 1
N3	label 3	;define label 3
G97 V9911=V1	V9911=V1	;write variable 1 to PAMUX address 1
G4 F500	dwell 500 ms	;dwell for 500 ms
G97 V1=V1.LSL4.H,1	V1=V1.LSL4.H,1	;shift active bit 1 place to the left
G62 L4 C1=H,100	GOTO LABEL 4 C1=H,100	;goes to label 4 if first 8 have been sent
G60 L3	GOTO LABEL 3	;go to label 3
N4	label 4	;define label 4
G97 V9911=H,0	V9911=H,0	;clear all 8 bits of PAMUX address 1
M2	end prog	;program done

Example 2: Read PAMUX address 1 and display.

Machine Tool	Menu Driven	Comments
G30 HIGH BYTE PAMUX READ PROGRAM		;program title
N1	label 1	;define label 1
G97 V0=V9911	V0=V9911	;read PAMUX address 1
G97 V1=BTF(V0)	V1=BTF(V0)	;convert to floating format
G21 #H:V0 #V0	MESG #H:V0 #V0	;display data in Hex and Decimal formats
G4 F200	dwell 200 ms	;dwell for 200 ms
G60 L1	GOTO LABEL 1	;go to label 1

Example 3: Dwell until input condition is met.

Machine Tool	Menu Driven	Comments
G30 CONDITIONAL INPUT PROGRAM		;program title
N1	label 1	;define label 1
G97 V0=V9911	V0=V9911	;read input state
G62 L1 C0< >H,3A	GOTO LABEL 1 C0< >H,3A	;loop until bits 1,3,4,5 are set and 0,2,6,7 are cleared
G97 V9910=H,11	V9910=H,11	;set output bits 0 and 4, clear others

...User Program

IEEE-488/GPIB USE

When the MPI Board is configured as a GPIB Bus it may be used for remote control, and transfer of files and data to/from the Unidex 400. Before using the GPIB Bus for Remote Operation or file transfer the following general parameters must be appropriately established. Refer to Chapter 11 for Connector details.

032 : IEEE 488 SETUP (GPIB)

This parameter is used to configure a variety of IEEE-488 settings. Following selection of this parameter, the display will be:

```
0 : GPIB ADDRESSING MODE
    - 0  talk only        - 1  listen only
    - 2  talker/listener
1 : DEVICE ADDRESS  (0-31)
item : _                  data :
 ins-Y/N     del      item      data      preset      quit    m
```

The following selections are available for viewing or altering IEEE-488 settings:

ins-Y/N Press the F1 key to toggle character insert ON (Y) or OFF (N). When ON, characters are inserted without overwriting existing text.

del Press the F2 key to delete the character at the cursor position.

item Select **item** to move the cursor to the location on the display that a general parameter number may be entered.

Press **ENTER** to accept the item number and to move the cursor to the **data** location on the display.

The current setting will be displayed.

Key in the new parameter setting and then press **ENTER** to accept the new setting. This setting will remain in effect until overwritten by a new setting or the default value is recalled by use of the **preset** selection.

NOTE: **Following the configuration of IEEE-488, the Unidex 400 must be Reset to initialize the values.**

preset Select **preset** to display the default value of the parameter setting. Press **ENTER** to accept the default setting as the parameter value.

quit Press the F6 key to abort the command and return to the initial setup menu.

Press the "down" arrow for additional IEEE-488 settings. Press the "Left" and "Right" arrows to move the cursor in these directions.

The following section provides an explanation of each of the IEEE-488 settings.

032 : 0 : GPIB ADDRESSING MODE

This setting is used to establish the addressing mode of the Unidex 400 on the IEEE-488 Bus. The address may be configured as one of the following:

0 - talk only the Unidex 400 should be configured as a "talker" when it is needed only to send data.

1 - listener the Unidex 400 should be configured as a "listener" when it is needed only to receive data.

2 - talker/listener the Unidex 400 will respond to the GPIB Controller, both sending and receiving data.

The default setting is 2, the Unidex 400 is both a "Listener" and a "Talker".

032 : 1 : DEVICE ADDRESS (0-31)

This setting is used to configure the address of the Unidex 400 on the Bus. Care should be taken that the address allocated to the Unidex 400 by this parameter is not already set for another device.

The setting range is 0-31. The default setting is 2.

032 : 2 : PARALLEL POLL RESPONSE (BIT)

This parameter determines the bit number that the Unidex 400 will set when the Controller does a parallel poll of the devices on the Bus. The parallel poll response may be disabled by setting this parameter to zero. If this parameter is set between 9 and 16 the Unidex 400 will respond with a reverse polarity parallel poll.

Example:

If set to 1 : Unidex 400 will set DI01 to a logic high during a parallel poll.

If set to 9 : Unidex 400 will set DI01 to a logic low during a parallel poll.

If set to 0 : Unidex 400 will not drive any line during a parallel poll.

The setting range is 0-16. The default setting is 1.

032 : 3 : EOS CHARACTER (0-255)

The End of String character indicates the end of a block of data. This parameter is typically set to the line-feed character (10 decimal). The Unidex 400 will send the EOS character when returning strings of data.

The Unidex 400 does not use the EOS character to stop data during the file transfer mode since the line feed character is used to delimit program lines. During file transfer, the Unidex 400 uses only the User specified End of File (EOF) and End All File (EAF) characters to separate files and terminate transfer.

The EOS character may be any value between 0 and 255. This also determines the size of the EOS character, 7 or 8 bits.

The Unidex 400 expects a <CR> (13 dec.) as a terminating character for the data received. (Refer to general parameter 006).

The setting range is 0-255. The default setting is 10.

032 : 4 : TERMINATE WRITE EOS-0/EOI-1 0/1

The Unidex 400 will send either the End of String (EOS) character or send an End or Identify (EOI) character along with the EOS character to terminate data transfer. During string input to the Unidex 400, only <CR> may be used to terminate the string. String input occurs during use of Remote mode "2" as a command string or as a message using the **G17** or **G19** commands.

The default setting is 0, only the EOS character will terminate data transfer.

The following display is provided for information only:

```
Command Input:  U400 will wait for Line
                Feed.
File Transfer:  U400 will use EOF/EAF
                only. See Gen Par  6,7
item : _                 data :
 ins-Y/N    del      item          preset     quit    m
```

REMOTE OPERATION

The Unidex 400 may be operated in two GPIB Remote Modes, Mode 2 and Mode 4. The Remote Mode is entered by sending the "Set Remote Mode" command 1D Hex, 29 Dec) followed by the ASCII character for the mode. Modes "1" and "3" are RS-232 modes. Refer to Chapter 9 for additional information concerning Remote Operation of the Unidex 400.

Any Mode may be enabled from the GPIB Bus. While in one of the GPIB Modes, data input from a terminal, Front Panel or Serial Port A is ignored. The only exception is when a program command specifically requests data from another source.

MODE 4

Mode 4 is similar to the Local Mode except that the keystrokes are initiated from the GPIB Bus. All ASCII characters are valid in this mode.

MODE 4 COMMAND TABLE

Unidex 400 Key	Decimal
F1	1
F2	2
F3	3
F4	4
F5	5
F6	6
up arrow	7
dn arrow	8
lf arrow	9
rt arrow	10
"0"-"9"	48-57
+/-	21
.	22
Enter	23
c-start	24
feed hold	25
s1	26
s2	27
s3	28
RESET	31,31

MODE 2

Mode 2 utilizes the immediate commands listed below. These commands should be input to the Unidex 400 terminated by a <CR>. The Unidex 400 will then return some number of bytes in either ASCII or Binary format, depending on the command. When all data has been sent to the controller, the Unidex 400 will send EOS or EOS/EOI, depending on the configuration of General Parameter 032:4.

MODE 2 COMMAND TABLE

PD	Print Directory of all files in memory
Pnn	Print Program Number nn
PS	Print Status Bytes
PX/PY/PZ/PU	Print axis X,Y,Z,U commanded position
Px/Py/Pz/Pu	Print axis X,Y,Z,U actual encoder position
PA	Print all programs in memory
Q	Query (serial poll), see Byte 1 of status
C	Reset System
F	Insert block number when printing program
G	Cancel block number when printing (default)
M	Status/Position format set to Binary (default)
N	Status/Position format set to ASCII

PD : Print Directory

Unidex 400 returns a list of all program numbers in memory. Memory status information is also printed. Unidex 400 inserts a <CR> <LF> following every 80 bytes of information sent. It uses these characters to delimit the file listings from the memory information. Unidex 400 will terminate the transmission with EOS or EOS/EOI.

Pnn : Print File #nn

The Unidex 400 sends the ASCII file "nn" to the GPIB Bus. A <CR> <LF> is also sent since these characters are used to delimit the lines of the program. The Unidex 400 will terminate the transmission with EOS or EOS/EOI.

PS : Print System Status

The system status is a group of thirteen bytes that represent the current state of the Unidex 400 and are returned in response to the "PS" command. Refer to Chapter 9, Pages 9-6 through 9-12 for a list of the System Status Codes. Note that Byte 1 is the same as Serial Poll. System Status Bits labeled with a "*" are not implemented at this time. The Unidex 400 will request service from the controller then send the error code. A <CR> <LF> is sent to delimit each status byte in the ASCII Mode. In the Binary Mode the bytes are sent directly (refer to the "M" and "N" commands). Transmission is terminated by a EOS or EOS/EOI.

PX/PY/PZ/PU : Print Commanded Position

These commands return the commanded position of the X,Y,Z and U axes respectively in machine units. This number is the actual 32 bit position register of the given axis. Its Binary value will be returned either in ASCII format or as four Binary bytes (refer to the "M" and "N" commands). Transmission is terminated by a EOS or EOS/EOI.

PX/PY/PZ/PU : Print Encoder Position

Same as above except position is as read from the position feedback device.

PA : Print All Programs In Memory

The Unidex 400 will send all files in memory to the GPIB in ASCII format. No special characters are used to delimit the files. Transmission is terminated by a EOS or EOS/EOI.

Q : Query

The Unidex 400 will send the Serial Poll byte to the GPIB. It may be in either ASCII or Binary format (refer to the "M" and "N" commands). Transmission is terminated by a EOS or EOS/EOI.

C : Reset System

This command performs a hardware reset on the Unidex 400. The Unidex 400 will no longer be in the Remote Mode following use of this command.

F : Print Block Number of Program

The Unidex 400 inserts the line numbers of each program transmitted to the GPIB Bus. (Refer to the PA and Pnn commands.)

G : Block Number Off (Default)

Cancels block numbering during program printing.

GPIB FILE TRANSFER

The Unidex 400 can receive files or transmit files over the GPIB Bus. The file output mode utilizes User defined special characters to delimit files. End of File (EOF) and End All Files (EAF) are used to signal the end of each file and the end of all file transmission for multi-file transmission. For single file transfers, EOF and EAF characters are sent following tranfer of the file data. EOI may be sent with the last character (EAF) during file transfer if the GPIB parameter 032:4 is set to EOI (1). (Refer to Chapter 7 of this manual for parameter information.)

Output Files

The Unidex 400 can transmit one or all files contained in its memory to the GPIB Bus. All data except for EOF and EAF characters are in ASCII format. The format for data transfer is as follows:

1. Check for files. If there are no files the Unidex 400 will set SRQ and Feedback file error code 32H (50 dec.)

2. Send file header and file name.

3. Send CR and "%" to signify the start of file data.

4. Send all file data.

5. Send End of File character (EOF).

6. If Single file Mode or All Files are transferred, send End All File (EAF). Otherwise goto line 2.

Input Files

The Unidex 400 can accept one or more files from the GPIB Bus. If multiple files are input, the file names are entered into the data stream. The file name is the last four ASCII characters between 0-9 that precedes the "%" character.

If a single file is input, its name should be placed in the name field of the file input menu. This is often performed by the remote device (Mode 4). The format for file input is as follows:

1. Scan the input stream for the "%" character or EAF. The last four ASCII characters will be the file name in the multi-file mode.

2. Check the file name against the current files in memory. If the file already exists, feedback an error code, otherwise continue.

3. Check memory, if no memory remains, an error will be returned.

4. Input all data to the Unidex 400 User file location until the EOF character is received. Memory limits are checked after each byte is received.

5. If single file is sent, wait for End All File character. If multiple files are sent, goto 1.

NOTE: **The Unidex 400 will display the message "DONE" when file transfer is complete.**

NOTE: **If a remote error should occur the Unidex 400 will request service by setting the SRQ bit in the Series Poll Status byte. The Unidex 400 will then feedback a one byte error code followed by EOS or EOS/EOI.**

File Error Codes:

30H : Illegal file mode key input
31H : File format error
32H : No File
33H : File already exists
34H : Memory full
35H : RS232 transmit checksum error
36H : RS232 protocol error (frame, parity, overrun, break, stop)
37H : GPIB bus error

Refer to Chapter 9 of this manual for a complete Error Code listing.

Service Request

When using Remote Modes 2 or 4, the Unidex 400 may request service (SRQ) from the controller. In the event of an error, the Unidex 400 waits for the controller to poll it before doing any further processing. At the completion of the polling, the Unidex 400 returns a one or two byte error code (excluding the serial poll byte) depending on the error type. The Unidex 400 terminates with either EOS or EOS/EOI, depending on the configuration of General Parameter 032:4. (Refer to Chapter 7 of this manual for parameter information.

Recoverable errors (MINOR errors) return a single byte. This byte is an error identification code. Examples of recoverable errors are; illegal key entries, file check errors, command format errors, axis in limit, etc.

Non-recoverable errors (MAJOR errors) return two bytes of error information. The first byte is used to signal that the second byte is a major error. The second byte identifies the type of error. Major errors usually indicate failure of a system component requiring replacement. Contact Aerotech, Inc. in the event of a Major error.

The Unidex 400 may also request service from the controller after certain events have been completed. These events are not error conditions so the error bit is not set, the Unidex 400 does not wait wait for the controller to respond to these Service Requests Examples of this type of Service Requests are:

1. A Program has finished execution from the AUTO Mode.
2. An immediate command has finished execution from the MDI Mode.
3. One of the following run-time front panel keys has been depressed; ABORT, STOP or QUIT.

NOTE: **The Serial Poll Status Byte is the same as byte one of the System Status when in Mode 2.**

APPENDIX A: CORNER ROUNDING VELOCITY PROFILING

CORNER ROUNDING (G23) AND NON-CORNER ROUNDING (G24) MODES

An insertion of the code "G23" within a motion command statement (eg., G8 G1 X100.0 Y200.0) or on a seperate line, sets the Unidex 400 for the "corner rounding" mode. The definition of "corner rounding" as applied to the Unidex 400is as follows.

Set the trajectory generator of the Unidex 400 to execute all subsquent motion commands (either linear or circular commands, in a single or multi-axis configuration) at the commanded feedrate. Depending upon the setting of General Parameters 40 or 41, the Unidex 400 may either utilize the Ramptime parameter (General Parameter 5) or the **G6** (**SERVO – > RAMP**) command setting and/or the **G32** (**MODE – > FILT**) c ommand for acceleration/deceleration..

Disable the "motion complete flag" in the servo control loops of each of the enabled axis's. This effectively tells the trajectory generator to "not wait" for the given axis's to stablize on the final commanded position of the just completed motion command before sending the next motion command(s) to the servo loops.

An insertion of the code **G24** instead of the code **G23** "inverts" the meaning of the two statements for the **G23** mode. In other words with **G24** inserted, the "motion complete" function is enabled insuring that the commanded position of the currently executing motion command is completely executed by the servo loops before the next command(s) is sent down by the trajectory generator.

Also it is possible to utilize both the **G6** (**SERVO – > RAMP**) or **G32** (**MODE – > FILT**) commands (or their related parameter settings) in order to optimize the trajectory command execution and provide similar performance to "following error" based motion controllers. Refer to related commands and parameters for further information.

Note that the **G23** and **G24** commands are system modal. In other words, once the command **G23** is encountered either in the execution of a program or the system "MDI" (manual) mode, it stays in effect until changed by a **G24** command or a system reset.

PROFILING (G8) AND NON-PROFILING (G9) MODES

An insertion of the **G8** command enables the velocity profiling mode. Unlike the **G23/G24** commands, which are modal, this command must be inserted within each motion command in order for it to be recognized for the given motion command (eg. G8 G1 X100.0 Y100.0). The definition of "profiling" as applied to the Unidex 400 is as follows.

Each motion command processed by the trajectory generator is setup to link the moves together such that constant vector velocity is maintained between commanded motion blocks or, if the Feedrate is changed, the system will accelerate/decelerate to the next velocity utilizing either the **G6 (SERVO – > RAMP)** command and/or the **G32 (MODE – > FILT)** command.

If the motion command containing this code is the first motion command within a string of other **G8** motion commands (eg., if it is the first command after a previously executed **G9** command), the vector velocity of this command will begin at zero speed. Subsequent **G8** commands encountered after this first **G8** command will be executed beginning at the vector velocity reached by the previous **G8** command.

If the **G24** mode is enabled, each new vector velocity specified by **G8** commands will be "ramped" to that specified velocity starting at "attained" velocity of the previous **G8** command (the ramp time being specified by the **G6 (SERVO – > RAMP)** command, eg., (G6,100). Note that for this particular mode of operation, it is important that the vector velocity "direction" be maintained between ending of one **G8** command to the beginning of the next **G8** or ending **G9** command inorder for the desired position trajectory to be maintained. If velocity vector direction needs to be changed, it MUST be accomplished through the programming of an "arc" or circle segment. This is the only way to ensure tangency.

If the G23 mode is enabled, and General Parameter 41 is set to NO, each new vector velocity specified by G8 commands will be instntaneously executed at the specified velocity. In the mode the **G6 (SERVO – > RAMP)** command is is ignored by the trajectory generator. This means that when it is desired to change vector velocity directions between G8 programmed segments, the programmer need specify only new "linear" command segments (e.g., G8 G1 X100.0 Y100.0) in order to change direction . The acceleration/deceleration can only be provided by the digital filter setting, **G32 (MODE – > FILT)** or General Parameter 40."

PROFILING (G8) AND NON-PROFILING (G9) MODES (CON'T)

An insertion of the **G9** command will terminate the velocity profiling mode and essentially puts the trajectory generator in the "point to point" position mode. The motion command executed under this mode is "guareenteed" to feed the servo control loop(s) the final or ending position of the specified command. Note that this command is "inferred" (eg., considered the default command) in the **G8/G9** mode of operation. The motion command "G1 X100.0 Y100.0" is automaticly reconized by the trajectory generator software as being the command "G9 G1 X100.0 Y100.0". The definition of "non-profiling" as applied to the Unidex 400 is as follows.

> Each G9 command processed by the trajectory generator is setup to execute based on the priority of attaining the final commanded position. The velocity profile will be calculated in such a fashion as to insure that the position generated by the trajectory generator is gaureeteed to be obtained at a final axis velocity of zero.
>
> A G9 command NOT preceeded by a G8 command has a beginning ending vector velocity of zero.
>
> A G9 command preceeded by a G8 command has a beginning vector velocity equal to the ending velocity of the previous G8.

In the "following error" mode, an inheherent "exponential" ramp function is added to each of the servo axis, effectively "smoothing" out sudden changes in velocity commands. The effect is illustrated in Figure A-1.

PROFILING (G8) AND NON-PROFILING (G9) MODES (CON'T)

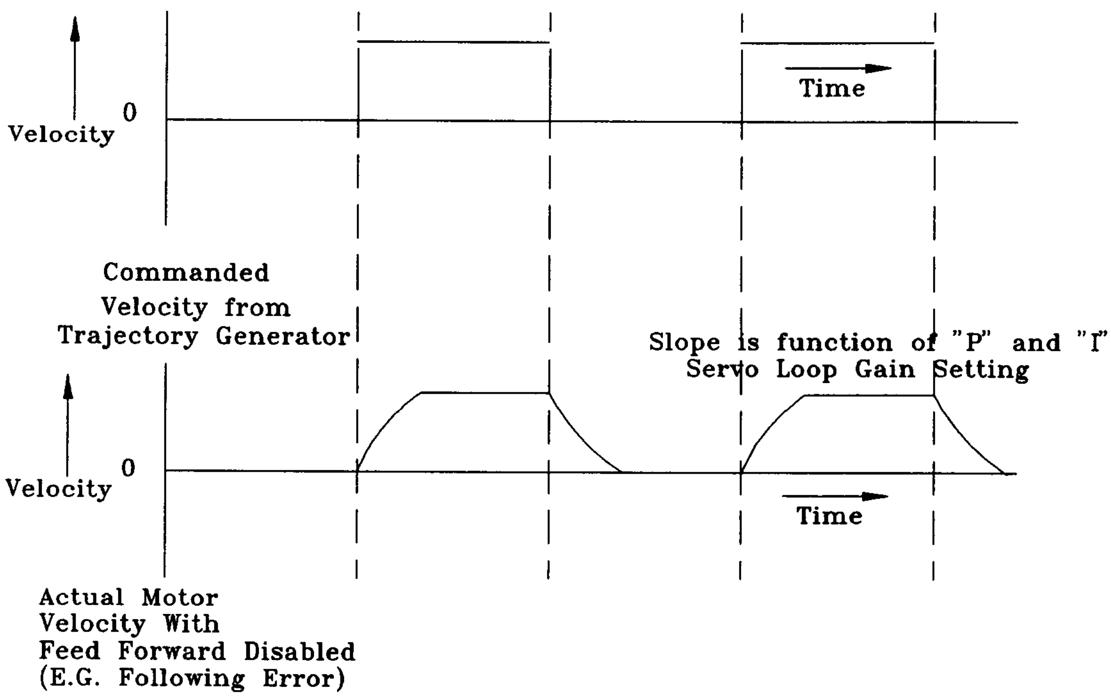


Figure A-1: Effects of G23 Mode on Trajectory Generators with Following Error Enabled (KF1 = 0)

This effect may also be obtained using the **G32 (MODE – > FILT)** command. or the General Parameter 40. In this mode of operation the exponential ramp is applied to the velocity command. Note that following error does not need to be enabled in this mode.

PROFILING (G8) AND NON-PROFILING (G9) MODES (CON'T)

Combining the **G8** function with **G23**, and enabling "following error" for the servo loops allows the position profiles such as the one depicted in Figure A-2, to be executed.

Refering to Figure A-2, take note that the joining points of the **G8** command segments (which are strings of arc and linear commands) are not tangent to each other. When operating in the **G23** mode, segment tangency is not required. In addition, velocities for each segment can be changed at will (changing of velocitys can not be made appearent in this diagram) without significantly changing the intended profile.

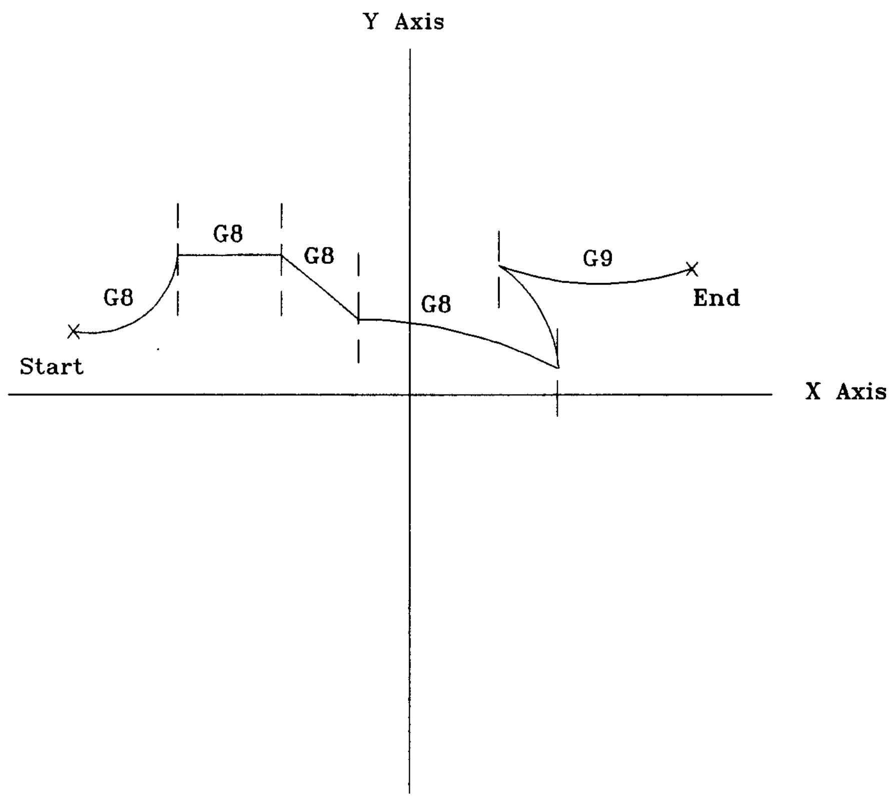


Figure A-2: Typical Two Axis Position Profile Using G8, G23 with Following Error or Filter Enabled

PROFILING (G8) AND NON-PROFILING (G9) MODES (CON'T)

Combining the **G8** function with **G24** allows position profiles such as the one depicted in Figure A-3 to be executed. Note that for this particular mode of operation, allowing following error to be enabled or disabled produces negligible effects on the motion profile unless high vector velocities are commanded.

Refering to Figure A-3, notice that the joining points of each segment are tangent. Tangency between all **G8** and **G9** segments is a requirement for **G24** operation. As in the case of the **G23** operation described above, segment velocities can be changed at will. In addition, ramptime can be changed from segment to segment.

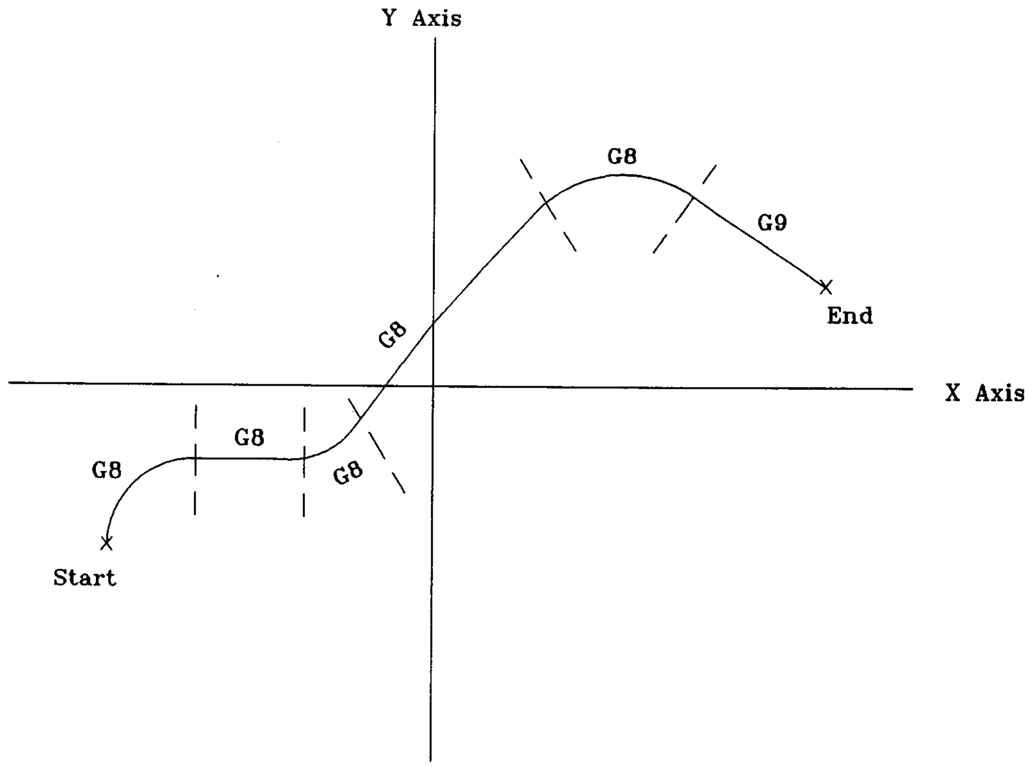


Figure A-3: Typical Two Axis Position Profile Using G8 and G24 Modes

(THIS PAGE INTENTIONALLY LEFT BLANK)

INDEX

A

B

C

D

E

T

U

V

W