UNIDEX™1 DC SERVO MOTION CONTROLLER USER'S MANUAL

PN: EDU108

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

Rel 1.0, April, 1989

DISCLAIMER:

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

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

NOTICE:

The information presented in this manual provides a detailed explanation of the Unidex 1 DC Servo Motor Controller. Detailed information pertaining to Unidex 1 programming as well as motor, input power, and external control interconnections, is provided in this manual.

If you purchase the optional support software package, you will receive the *Unidex 1 Interactive Control Software (SSP2) Manual* as well.

Table of Contents

CHAPTER 1: INTRODUCTION

SECTION 1-1 GENERAL DESCRIPTION

Unidex 1 is a single axis microprocessor-based motion controller. It is available in two packages: a low power 6U Euro-cardrack mount, and a low power panel/desktop mount. Both configurations are shown in figure 1-1. Table 1-1 lists the standard motor/drive part numbers for the associated drive configurations of figure 1-1. Also listed in table 1-1 is the specific motor and drive part number of the given configuration to provide a reference for other Aerotech stepping drive literature. Figure 1-2 illustrates peak torque vs. speed characteristics for the motor/drive configurations of table 1-1.

The Unidex 1 is controlled through the Computer Interface port (Port-A) exclusively. Port A provides two methods of control: (1) the RS-232 command mode and (2) a special menu-assisted control mode. The menu-assisted control is provided by an optional hand-held terminal (model TFX). (This hand-held terminal is described in detail in chapter 4.)

SECTION 1-2 UNIDEX 1 FEATURES

Program Capacity

- Capacity to store approximately 700 1-axis moves
- Stores up to 99 randomly accessible programs

Programmable Accel/decel Parameters

- Accel/decel ramp time = 50 - 6500 milliseconds
- Start/stop feedrate = 1 - 250000 steps/second

Rs-232 Command Mode Features

- Setup mode for modal parameters
- Immediate command execution
- Downloading for up to 99 programs
- Program-run in Auto-run or Block-run mode
- Print - Directory, Axis Position, Single Program, Entire Memory, System Status
- Interactive control capability (Service Request Mode)

Hand-held Terminal (model Tfx) Features (optional)

- Special menu-assist screens
- Setup mode for modal parameters
- Immediate command execution
- Enter and edit up to 99 programs
- Program run in Auto-run or Block-run mode
- Print: Directory, Program, Axis Position, System Status
- Special operations: Home, Index, Stop Axis
- Tracking display: Continuously displays Mode of Operation, Position, Direction of Travel, and Marker Status while the motor is running

Daisy Chain Operation

- Up to 30 Unidex 1s can be connected and accessed at one time
- Address selectable

Auto-boot Program

- Select any of 99 programs for automatic execution upon power up

Key Program Mode Features

- Conditional program label and subroutine branching through logic input status
- Interrupt program label and subroutine branching through logic input status
- Programmable position boundaries (limits) in program run mode

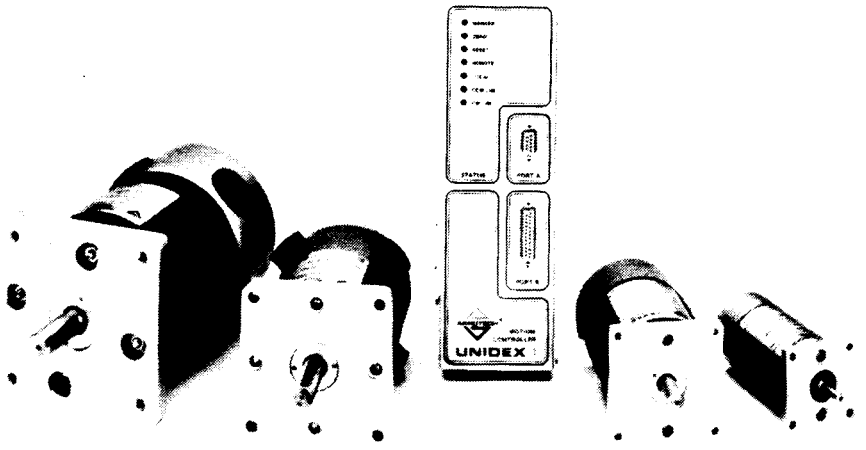


Figure 1-1: Unidex 1 DC Servo Motion Control Family of DC Motor Control

SERVO DRIVER MODELS

Model		U1SA	U1SB	U1SC	U1SD	U1SE	U1SF	U1SG	U1SH	U1SJ
Continuous Torque	oz-in	17	35	50	75	135	210	410	580	800
	N-m	0,12	0,25	0,35	0,53	0,95	1,5	2,9	4,1	5,6
Peak Torque	oz-in	82	170	186	280	490	700	1410	1300	1600
	N-m	0,58	1,2	1,3	2,0	3,5	4,9	10,0	9,2	11,3
Speed (max)	rpm	6500	5000	4200	5000	3600	2500	2750	2750	2750
	rps	108	83.3	70	83.3	60	41.7	45.8	45.8	40
Rated Power	Watts	60	90	115	140	200	220	340	600	780
Rotor Inertia	oz-in-sec^2	.004	.005	.008	.023	.052	0.130	0.180	0.26	0.37
	kg-m^2	0,03x10-3	0,04x10-3	0,06x10-3	0,16x10-3	0,36x10-3	0,92x10-3	1,3x10 -3	1,8x10 -3	2,6x10 -3
Motor										
Model		1017LT	1035LT	1050LT	1075LT	1135LT	1210LT	1410LT	1580LT	1960LT
Type		DC permanent magnet, less tachometer								
Driver Amplifier										
Type		20KHz pwm, MOSFET bridge								
Bus	VDC	40	40	40	80	80	80	160	160	160
Recommended Output Fuse Rating	Amps	4	4	5	5	5	6.25	6.25	8	10
Peak Output Current	Amps	20	20	20	20	20	20	20	20	20
Protection										
Short Circuit		Phase-to-phase, Phase-to-ground								
Low AC Protection		AC less than 95VAC, recovery without power reset								
Weights										
Motor	lbs	2.7	3.5	4.5	6.3	9.1	10	14.3	26	38
	kg	1,2	1,6	2,0	2,9	4,1	4,5	6.5	11,8	17.3
Total (Motor, Controller, Cable, Mn)	lbs	18.7	19.5	20.5	22.3	25,1	26.0	25.3	37	49
	kg	8,5	8,9	9,3	10,1	11,4	11,8	11,5	16.8	22,3
Motor Armature Inductance (LA)	mH	.7	2.0	2.5	2.7	3.1	4.9	15.8	2.3	4.5
Motor Armature Resistance (RA)	ohms	.47	.9	.7	1	1.4	1.8	1.97	1.1	.5

Table 1-1: Unidex 1 Motion Controller DC Servo Specifications

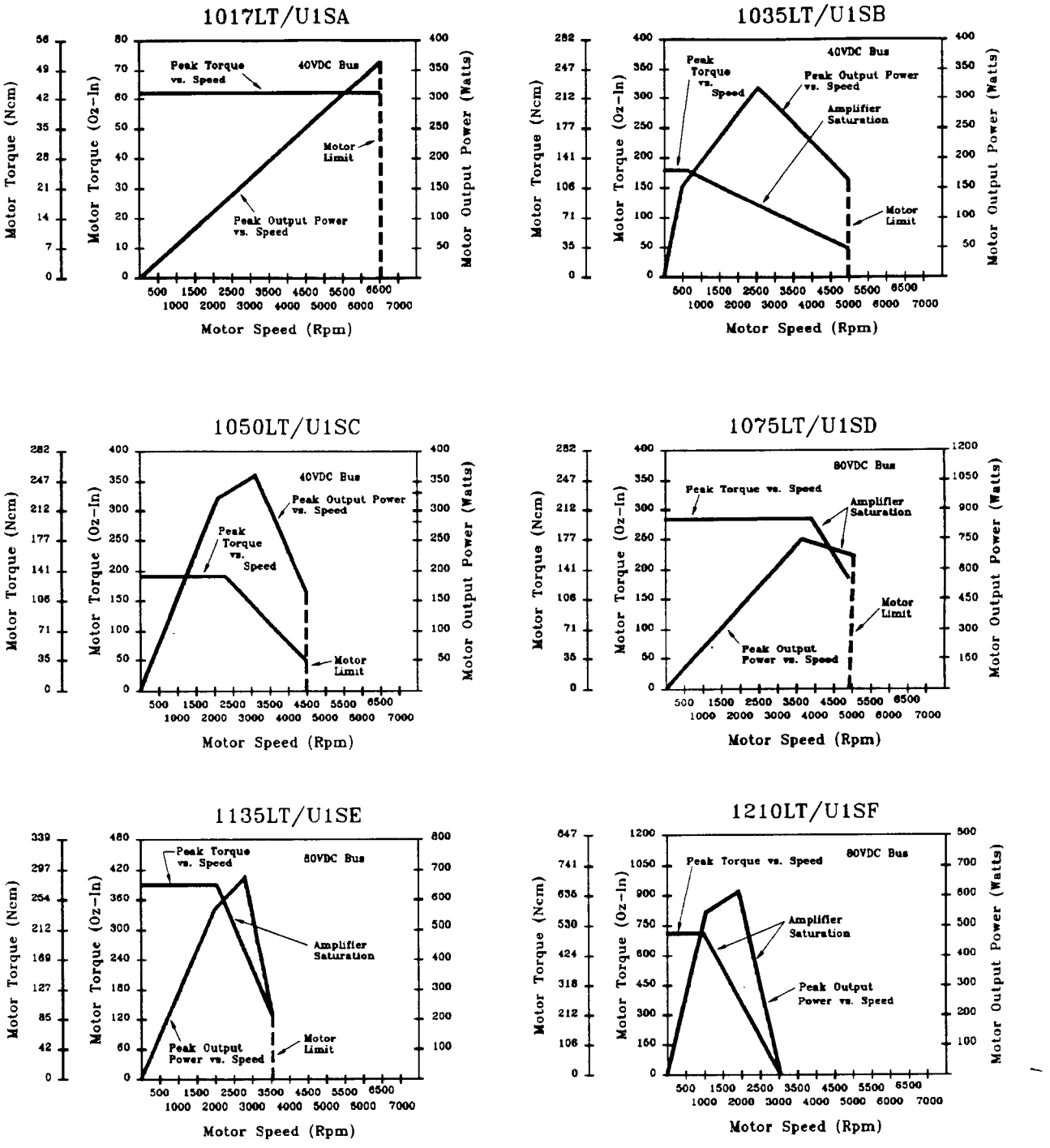
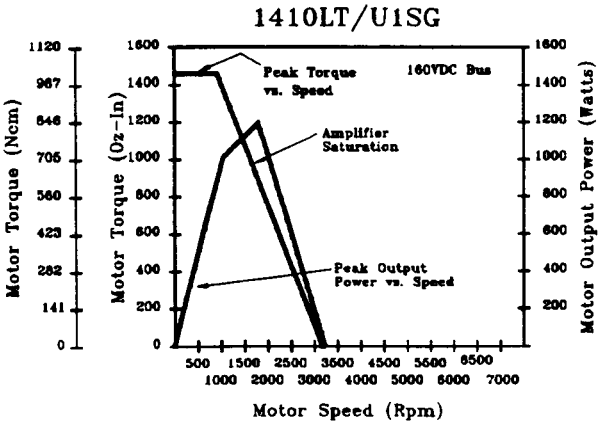
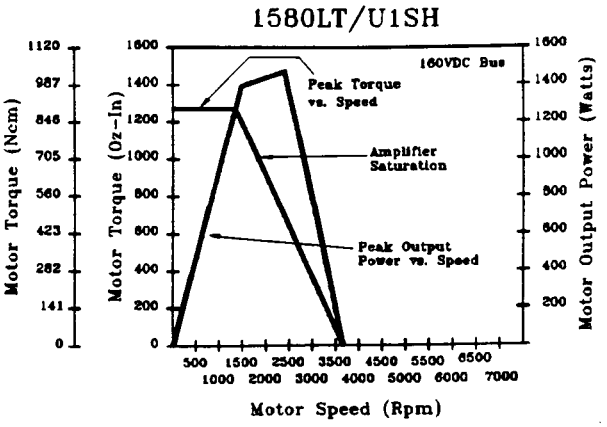


Figure 1-2: Peak Torque vs. Speed for Unidex 1 Models





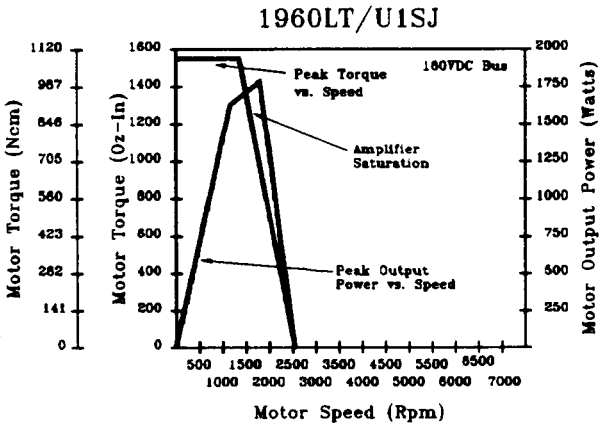


Figure 1-2: Continued

CHAPTER 2: POWERING UP

SECTION 2-1 POWERING UP

Before applying power to the Unidex 1, review all information outlined in chapter 8 (System Layout and External Connections). **To avoid damaging your Unidex 1, make certain that the correct AC input voltage has been chosen.** (See "Unit Specification Label" of figure 8-1, 8-2A and 8-3.)

After voltage is applied to Unidex 1, one or more LEDs will light. The Reset LED may light briefly (less than a few seconds), but will turn off if the unit is operating correctly. If the Auto-Boot program is present, it will be executed at this time (see section 2-5).

SECTION 2-2 INITIALIZATION

Unidex 1 can operate in one of two modes: the Computer Interface mode (chapter 5), or the optional Hand Held Terminal mode (chapter 4). (A third operating mode utilizes an optional RS-232 Thumbwheel Programmer (TP), or Thumbwheel Indexer (TI) option—see chapter 4.) Both are activated in a similar fashion, which is described below. This description is for a single unit. Activating more than one Unidex 1 (daisy-chain mode) is described in the following section. All Unidex 1s must be in the inactive state before an activation command is given. If there are any active units, a deactivation command should be sent first (see section 2-3).

Mode	Activation Command
Computer Interface	>> nn <CR>
Hand Held Terminal	## nn <ENTER>
TP Thumbwheel Control	<<
TI Thumbwheel Control	@

NOTE: TP and TI Controllers generate these codes automatically. See chapter 4 for details.

NOTE: "nn" denotes the device address number.

NOTE: *When all Unidex 1s are inactive, all characters sent from a host computer or the optional Hand Held Terminal will be echoed back.* The optional Hand Held Terminal will display these characters and will appear as a double character on the screen. For example, "####0011" will appear to activate device #1.

Consider the echo character as a check on the communication loop (all Unidex 1s are receiving the command) and not as part of the command.

To activate the Computer Interface send two "greater than" signs (>>), followed by the device number of Unidex 1 (shown in the above example as nn). If the device number is unknown, it can be checked and set (described in chapter 3, Set Up). Follow the device number by a <CR> (carriage return). (The default address is 01.)

NOTE: Digit codes 00 and 01 should not be selected as device numbers for daisy chained Unidex 1s because of the possible interference with the set up of daisy chain operations (00 is used for Setup operations and 01 is the default address).

After the two digit device number has been entered, a <CR> must be sent. Following the <CR>, a short wait is necessary before sending another character. This wait time is needed to allow the Unidex 1

to configure itself as a talker. The wait time is dependent upon the baud rate and can be calculated using the following formula.

Time (seconds) = 12/Baud rate

For example, the wait time required for 9600 baudrate would be calculated as follows:

T = 12/9600 = .00125 seconds (or 1.25 milliseconds)

(Wait time must be equal to or greater than .00125 seconds for 9600 baud rate.)

The Unidex 1 will now be active and ready to receive a command. The Computer Interface will give no immediate response unless requested. A print command (section 5-7) can be used to verify activation. The optional Hand Held Terminal will respond immediately with a Menu Screen if only one unit has been activated. (Refer to chapter 4 for details on the TI and TP Hand Held Terminals, and chapter 5 for details on Computer Interface.)

SECTION 2-3 DEACTIVATING UNIDEX 1

To deactivate a Unidex 1 unit, send the <CTRL> B character (hex code 02). This will deactivate all active Unidex 1s and put them in the echo mode. Following the <CTRL>B, a short wait is required before sending another character (see section 2- 2).

SECTION 2-4 MULTI-AXIS (DAISY CHAIN) MODE

It is recommended that the Computer Interface mode be utilized if commands will be issued simultaneously to more than one Unidex 1.

To activate more than one axis at a time, enter a comma or space between each two-digit device code, and enter a <CR> after the last device number. To activate (in the Computer Interface mode) units 02, 05, 11, 21 and 30, enter the following code:

02,05,21,30,11 <CR>

It is not required that device numbers be entered in numerical sequence.

NOTE: **A deactivation command (<CTRL> B) should be sent preceding the activation command, if there is any active Unidex 1 presently on line.**

It is not recommended that the Hand Held Terminal mode be used if more than one Unidex 1 is active (except for the Trigger operation), since the Hand Held Terminal mode does not support interactive control. If more than one Unidex 1 is active at a time, information will not be sent to the Hand Held Terminal from any of the units.

See section 5-1 for "Daisy Chain" Specifications for Unidex 1.

SECTION 2-5 AUTO BOOT PROGRAM OPERATION

After the Unidex 1 has been powered up, the Auto Boot program will be executed. The Auto Boot program can be any of the 99 possible programs in memory, and may be selected through the Setup mode (chapter 3). If a program does not exist for the Auto Boot program number, the Unidex 1 will be ready to be activated and receive a command once powered up.

CHAPTER 3: SETUP

SECTION 3-1 DESCRIPTION

The setup mode allows for the Setup of certain Unidex 1 system features. These include:

- Device Address
- Baud Rate
- Stop Bits
- Parity
- Word Length
- Accel/Decel Ramp Time
- Start/Stop Frequency
- Output Active Level
- Boot Program #
- Load Default Values
- Hand Held Terminal Initialization
- PID Gain Selection (position control loop)
- Home direction
- Home limit speed
- Thumbwheel Indexer Index Speed

SECTION 3-2 SELECTING SETUP MODE

To enter the Setup mode, the Unidex 1 must be powered down and a connection must be made between the Setup input (pin 6 of the

input/output connector, Port B; see section 8-6) and signal common (pin 8 of the input/output connector).

The Setup mode requires the TFX hand held terminal or the Communication Interface device to be set to the following settings (even if previously set up to a different value):

- Baud Rate: 9600
- Stop Bits: 1
- Word Length: 7
- Parity: Even

The setup mode always operates in the above configuration.

See section 3-4 if the TFX hand held terminal must be initialized to the above settings.

With the jumper between pin 6 and 8 of Port B, power up and activate Unidex 1 by entering one of the following commands. **(This cannot be done with Port A wired in daisy chain.)**

Hand Held Terminal: ## 00 <ENTER>
Computer Interface: > > 00 <CR>

Note the TFX hand held terminal mode will display the following screen when activated:

1 DISPLAY SETUP
2 ALTER SETUP

The RS-232 Computer Interface mode will give no indication of being in the Setup Mode unless the host device sends a print setup command (**PV<CR>**).

SECTION 3-3 HAND HELD TERMINAL (TFX) SETUP

The hand held Terminal setup mode allows parameters to be read or changed. It also provides a method of initializing the TFX hand held terminal. (Refer to the next three sections for information regarding this mode.)

SECTION 3-4 HAND HELD TERMINAL (TFX)

A special hand held terminal Initialization mode has been provided to automatically initialize the TFX hand held terminal to the Setup Communications protocol of the Unidex 1 (9600 Baud, see section 3-2). This mode will set the hand held terminal port to the Unidex 1 communication requirements automatically. Once it is initialized, it should never require re-initialization. However, certain abnormal transient conditions may accidentally alter the internal hand held terminal memory. *To initialize the hand held terminal, power down Unidex 1. Connect set up pin (pin 6 of I/O, Port B connector) to signal common pin (pin 8 of I/O, Port B connector) with a removable jumper plug.* The hand held terminal must be connected to the Port A connector (it cannot be in a daisy chain configuration). Power up Unidex 1 and disconnect the Setup to Signal Common jumper after power has been applied for at least 10 seconds. The hand held terminal will start displaying characters while initializing. When the hand held terminal has been initialized, it will display the following screen. (Initialization time is less than 30 seconds.)

TERMINAL SET
9600 BAUD, EVEN

NOTE: Do not turn off power until the "Terminal Set" screen is present. The Unidex 1 must be powered down to exit this mode.

SECTION 3-5 HAND HELD TERMINAL DISPLAY SETUP

After the hand held terminal has been placed in the Setup Mode and activated (as described previously in section 3-2) the hand held terminal will display:

1 DISPLAY SETUP
2 ALTER SETUP

To display the Setup, press the F1 key. Pressing the F2 key will enable a setup parameter to be changed (see section 3-6, Hand Held Terminal, Alter Setup).

Every time the F1 key is pressed, a Setup parameter will be displayed. When the sequence has displayed the last setup code, the display will move back to the first Setup code. If the F5 key is pressed, the display will go back to the Display or Alter select screen. A description of each of the codes can be found in section 3-9.

SECTION 3-6 HAND HELD TERMINAL ALTER SETUP

The hand held terminal must be in the Setup mode and be active to alter the Setup. When the Setup mode is active, the following Display/Alter screen will be displayed:

1 DISPLAY SETUP
2 ALTER SETUP

To Display Setup, press the F1 key. To Alter Setup, press the F2 key.

If the Alter mode was selected, the following screen will appear:

ENTER SETUP CODE

To change a Setup code, enter the code directly into the keyboard as described in the Setup code description (see section 3-9). After code has been entered, *an * (End Of Block) character* must be entered. After the * character has been entered, the Setup code will be updated.

During the process of entering the setup code, the <BACK-SPACE> key will delete one character each time it is pressed, beginning with the last one entered. The changed Setup code can be verified by using the Display command. *To exit the Setup Mode, first power down the Unidex 1, and then remove the setup jumper (6-8 of I/O connector, Port B).*

SECTION 3-7 COMPUTER INTERFACE, PRINT SETUP

To print the Setup codes, Unidex 1 must be active and in the Setup mode. This is done by entering **> > 00 <CR>** (see Computer Interface mode, chapter 5). Unidex 1 will echo back characters which must be cleared from the computer's input buffer.

The Computer Interface Print command is PV <CR>. Upon receiving the PV command, Unidex 1 will send Setup codes and their values will be sent to the host device. A description of all of the setup codes is given in section 3-9.

SECTION 3-8 COMPUTER INTERFACE, ALTER SETUP

To alter the Setup codes in the Computer Interface mode, Unidex 1 must be active and in the Setup mode by sending **> > 00 <CR>**. To change a Setup code, enter the complete setup code followed by an "*" character. A description of all setup codes can be found in section 3-9. For example:

BR 14 *

The previous example is a baud rate setup code. The "*" character will cause the setup code to be updated. However, the given code will not be activated until both power and the set-up jumper are removed.

SECTION 3-9 SETUP CODE DESCRIPTIONS

This section describes each of the setup codes. These codes are valid in both the Hand Held Terminal and Computer Interface modes.

A. BAUD RATE

The baud rate setup command is used to set the communication baud rate for normal operating mode. (This does not affect the current Setup mode communication protocol. See section 3-2.) For example:

BR 14 *

This command will set the baud rate to 9600 baud (code 14, as shown next). The following codes set the following baud rates. (Hand held terminal operation uses 9600 baud rate.)

BAUD RATE CODES

Code	Baud	Code	Baud
01	50	09	1800
02	75	10	2400
03	109.92	11	3600
04	134.58	12	4800
05	150	13	7200
06	300	14	9600
07	600		
08	1200		

NOTE: Any code other than those listed above are invalid.

B. WORD LENGTH

The communication word length can be either 7 or 8 bits. Hand held terminal (TFX) operation requires a word length of 7:

WL 7 *

The above command sets the word length to 7 bits.

C. STOP BITS

The communication stop bit can be selected for either 1 or 2 bits. The hand held terminal (TFX) operation requires 1 stop bit:

SB 1 *

The above command sets the stop bit to 1.

D. PARITY

The parity command allows a parity of even (E), odd (O) or no parity (N) to be selected. Even parity is typically used for hand held terminal (TFX) operation.

PY E *

The above command will set the parity to Even.

E. DEVICE ADDRESS

The device address command sets the address which will activate Unidex 1. The device address must be a number from 0 to 31. (It is recommended, however, that 0 and 1 not be used as addresses, since 0 is used in the Setup mode, and 1 is the default address.) For example:

DA 24 *

The above example command will set the Unidex 1 to address 24.

F. BOOT PROGRAM

The boot program command sets the program number (from 1 to 99) to be executed on power up. **(Enter a boot program number of 00 to disable the boot program feature.)**

BP 49 *

The above example will set program 49 to execute when Unidex 1 is powered up.

G. OUTPUT ACTIVE STATE

The output active state command allows the Unidex 1 to be setup for an active high (H) or an active low (L) state on opto-coupled outputs 01 through 04 (refer to section 8-5 C).

Outputs set for Active High will produce:

Output a "0" = Transistor On (see section 8-5 C)
Output a "1" = Transistor Off

Outputs set for Active Low will produce:

Output a "0" = Transistor Off
Output a "1" = Transistor On

The command **OT H *** will set the output active state to active high.

H. ACCEL/DECEL

The ramp time command sets the Accel/Decel time in milliseconds. Ramp time must be between 0 and 6500.

AD 250 *

The above command will set the ramp time to 250 milliseconds.

I. START/STOP

The start/stop code sets the start/stop frequency for ramped X moves. The start/stop frequency is entered in steps/sec and must be between 2 and 250000. Once entered into Unidex 1, the start/stop frequency will be converted to the nearest usable frequency.

SS 10000 *

The above example would set the start/stop frequency for 10,000 steps/sec.

Due to the Unidex 1's 1 mS feedrate accuracy, the SS frequency will be automatically rounded to the nearest achievable feedrate. Displaying the setup commands show the actual value.

J. D GAIN

The D (derivative) gain code sets the D gain value for the PID loop (see section 3-10 for details). The D gain must be between 0 and 99. The new D gain will become effective following the next power reset. For example:

DG 53 *

The above command will set the D gain value to 53.

K. P GAIN

The P (proportional) gain code sets the P gain value for the PID loop (see section 3-10 for details). The P gain must be between 0 and 99. The new P gain will become effective following the next power reset. For example:

PG 32 *

The above command will set the P gain value to 32.

L. I GAIN

The I (integral) gain code sets the I gain value for the PID loop (see section 3-10 for details). The I gain must be between 0 and 99. The new I gain will become effective following the next power reset. For example:

IG 05 *

The above command will set the I gain value to 5.

M. HOME DIRECTION

The home direction can be configured to go into the CW limit (+) or CCW limit (-) direction.

HD - * or HD + *

The first command will cause a - or CCW limit direction when homing. The second command will cause a + or CW limit direction when homing. The first "marker" input pulse encountered after the Unidex 1 is reversed from the CW or CCW Limit direction, is the home reference position.

N. HOME SPEED

The speed at which the motor will move when approaching the limit while homing is selectable between 2 and 250,000 steps/second.

HS 8000 *

The above command will set the home limit speed to 8000 steps/second.

O. THUMBWHEEL INDEXER INDEX SPEED

The default Thumbwheel Indexer index speed is selectable between 2 and 250,000 steps/second.

TS 10000 *

The above command will set the Thumbwheel Indexer index speed to 10000 steps/second.

P. LOAD DEFAULT

The load default command will set the setup mode to the following. (The commands listed below will not be displayed.)

LD *

COMMAND	SETTING
BR 14	Baud rate = 9600
WL 7	Word length = 7
SB 1	Stop bits = 1
PY E	Parity = Even
DA 01	Device Address = 1
BP 00	Boot program = Disabled
OT L	Active low
AD 250	Accel/Decel = 250
SS 8000	Start/Stop = 8000
DG 50	D gain = 12
PG 30	P gain = 10
IG 05	I gain = 5
HD -	Home limit direction = - (CCW)
HS 8000	Home limit speed = 8000
TS 10000	Thumbwheel Indexer index speed = 10000

The LD command offers a fast and easy means of resetting Unidex 1 to the factory default setup parameters.

Q. PRINT SETUP VALUES

The print setup command will display all setup parameters currently set. For this command a Carriage Return must follow the command. This command is not applicable to the hand held terminal (TFX) interface.

PV <CR>

DISPLAY FORMAT

BR nn
WL n
SB n
PY n
DA nn
BP nn
OT n
AD nnnn
SS nnnnnnn
DG nn
PG nn
IG nn
HD n
HS nnnnnnn
TS nnnnnnn

This concludes the explanation of the setup. *Be sure that the jumper between pin 6 and 8 of Port B is removed after setup.*

Remember, the setup procedures outlined in this chapter will not operate if the communication to Port A is configured for the daisy chain mode and the communication parameters are not the default values. The RS-232 terminal, computer or optional hand held terminal (TFX) must be directly linked to Port A.

SECTION 3-10 PID GAIN LOOP ADJUSTMENTS

In section 3-9, *Setup Code Descriptions*, the initialization of gain parameters PG, IG and DG was discussed.

These parameters are provided solely for the adjustment of Position and Velocity Loop Gain in the PID loop network.

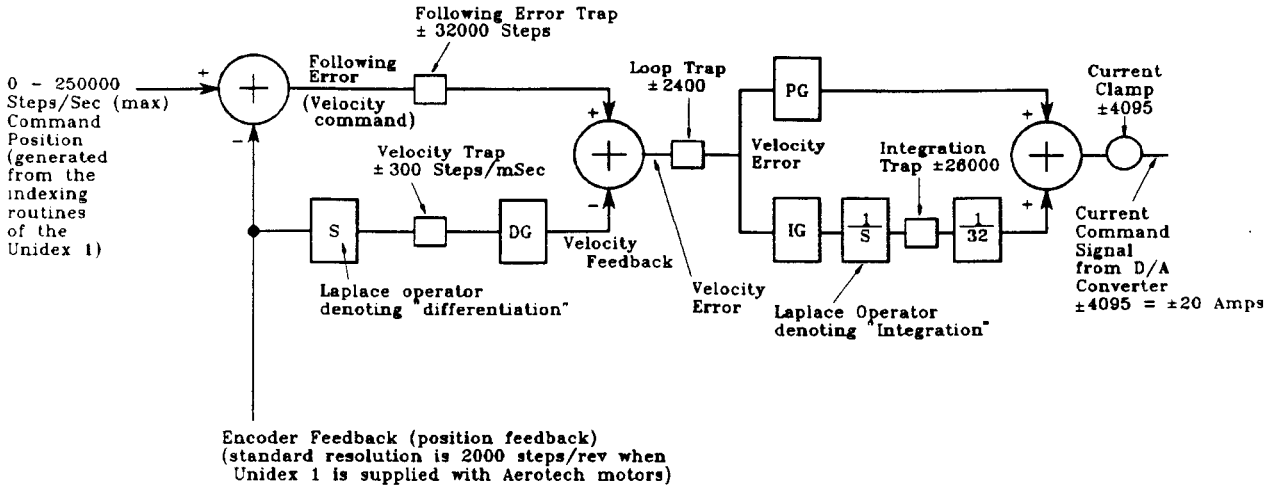
The PID control loop is a system software routine that runs "in the background" of the indexing (command software) of the Unidex 1. Every one millisecond (1 mSec), the indexing control software of the Unidex 1 is automatically interrupted and a PID control routine is requested.

This routine samples the position "change" from the last sampling time (the previous 1 mSec sample) and performs computations for determining present position and velocity. The present position and velocity information is then combined with the commanded position and velocity (from the indexer) through the PID network, to produce a motor torque command signal.

This torque signal is in the form of a current command signal generated from a D/A converter connected to the Unidex 1 microprocessor control circuit. The current command signal is used as an input signal to the power amplifier of the Unidex 1. A block diagram of the PID control loop algorithm for the Unidex 1 is shown in the following illustration. On close examination this diagram, you may notice that the Unidex 1 PID control loop is not arranged in the "standard" PID form, where proportional term "PG", the integral term "IG * 1/S" and the derivative term "DG * S" usually derive their inputs directly from the "position error". The reason for this is as follows.

Since the Unidex 1 DC controller is specifically designed for high performance "point to point" motion control, "zero" following error (see note) during the motion interval is not required for accurate positioning. Therefore, with the PID loop arrangement previously

shown, following error exists at all times during the motion interval. The amount of following error depends on the encoded resolution and the gain setting of parameter "DG".



Block Diagram of PID Control Loop for the Unidex 1 DC Controller

NOTE: Following error is the instantaneous difference between commanded position and actual position.

It is important that you notice the meaning of the term "following error". It is not to be confused with "position error" that occurs at the end of motion. Position error, which is the difference between the final commanded position and actual position, will always be zero provided the gain parameter IG is set to the value of one or greater.

Generation of following error in a closed loop position control system has one distinct advantage in point-to-point motion control. This is the elimination of the need for acceleration/ deceleration control on the command input for low to mid-range command feedrates. Since following error contributes to a "lag" between the system command and feedback, step-wise (or unit step) command feedrate inputs can be applied to the control loop without the generation of position over-

shoot at the end of the motion. Removing accel/decel from the indexing commands of the Unidex 1 (see section 3-9 and section 6-5) greatly reduces "block-to-block" motion command overhead, and provides faster command through-put.

Various "traps" (i.e., loop boundary checks) are programmed into the Unidex 1 PID loop control algorithm to check for such abnormalities as loop oscillation (caused by maladjusted gain parameters PG, IG, DG) or loss of feedback encoder. A brief discussion of these are as follows (referring to the PID block diagram previously illustrated).

Velocity Trap — If the "change" in feedback counts exceeds ±300 counts between successive sample periods (1mSec), a trap will occur. ("Trap" is defined after this listing.)

Following-Error Trap — If the following error (i.e., the summation of the command minus the encoder feedback step count) exceeds ±32000 steps, a trap will occur.

Loop Trap — If the difference between the following error and the velocity feedback exceeds the value of ±2400 counts, a trap will occur.

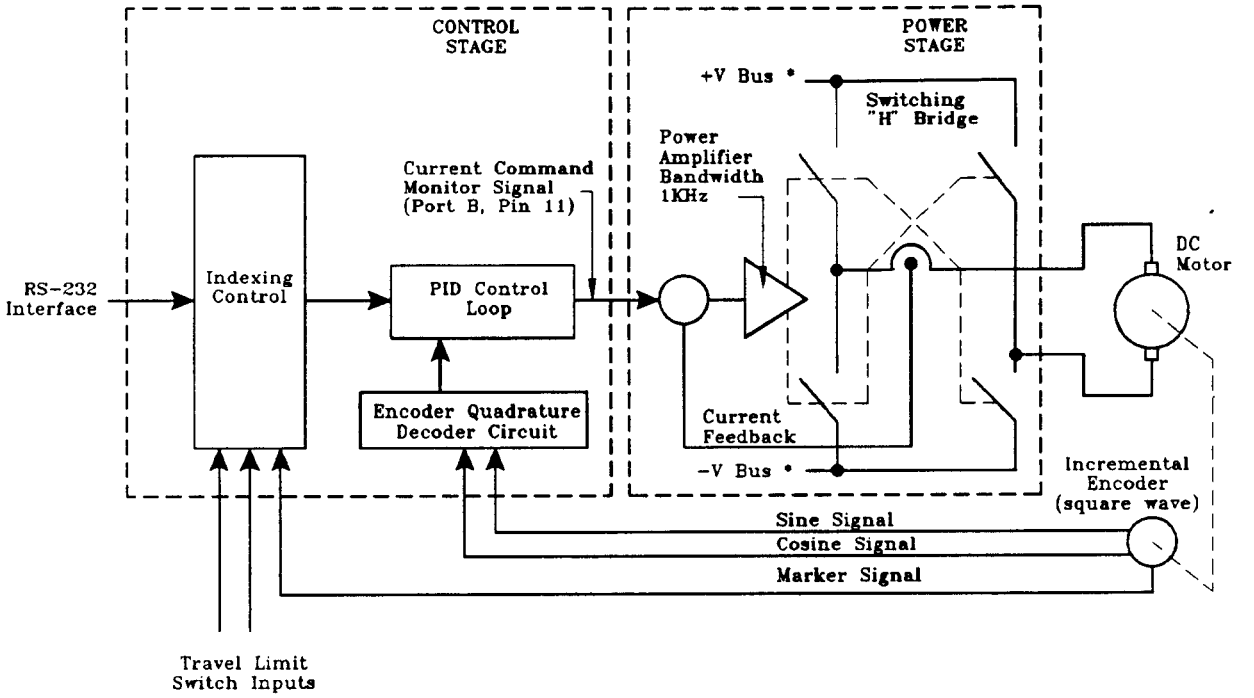
Integration Trap — If the integrator (1/S) exceeds ±26000 counts, a trap will occur.

Current Clamp — If the sum of the proportional element (velocity error * PG) and the integral element (velocity error * 1/S * 1/32) exceeds ±4095 counts (this value is equivalent to ±20 amps generated by the Unidex 1 DC power amplifier), the value of ±4095 will be maintained as the maximum "clamped" output value and no trap will occur.

A "trap" is defined as the action taken by the Unidex 1 loop algorithm when an "out of bounds" condition has been detected by the PID control loop. When a trap occurs, the loop algorithm immediately resets the current command output signal to zero. This reset condition is maintained until the user applies either a power-up reset, or a "soft" reset via a "CTRL-A" sequence through the RS-232 communications port. When using the TFX hand held terminal (chapter 4), the following will appear on the display screen in the event of a loop trap condition.

PSN LOOP ERROR
RESET TO ENABLE

The block diagram below shows the PID loop control as it would appear functionally, within the Unidex 1 DC system.



* +V/-V Bus: 40 VDC for U1SA through U1SC
80 VDC for U1SD through U1SF
160 VDC for U1SG through U1SJ

Block Diagram of Basic Unidex 1 DC Control System

The mathematical derivation of the PID loop is here for those users interested in deriving a precise control model for the given load/motor application. The block diagram of the PID control loop, in addition to the selected values of PG, IG and DG, should provide enough information to mathematically model the given system.

For those using the Unidex 1 DC with an Aerotech DC motor, table 1-1 and figure 1-2 in chapter 1 should provide enough information to enable you to model the given Unidex 1 DC/Motor combination.

A table (shown below) has been tabulated to provide a quick reference for what would be considered the optimum setting of loop gain parameters PG, IG and DG for given load inertias. In this table, load frictions are considered to be nominal, relative to the inertia specified, and are automatically factored into the gain parameters.

In this table, "ball-screw" driven loads (with approximately 4mm pitch) are assumed. Reflected load inertia and friction are minimal, and are factored into the motor and ball-screw friction and inertia.

Motor/Screw Inertia (oz-in-sec^2)	Viscous Motor/ Screw Friction (oz-in/Krpm)	Aerotech Motor Used	Nominal PG Setting	Nominal IG Setting	Nominal DG Setting	Min. Accel/ Decel Setting (mSec)	Recommended Start/Stop Setting (Steps/Sec)
.005	8	1017LT	9	24	12	50	8000
.007	8	1035LT	10	23	14	50	8000
.009	10	1050LT	10	23	15	50	8000
.025	12	1075LT	11	23	15	60	8000
.060	14	1135LT	11	23	18	70	8000
.145	17	1210LT	11	22	22	80	8000
.200	21	1410LT	10	21	15	100	8000
.310	30	1580LT	11	21	25	100	8000
.450	50	1960LT	11	20	28	100	8000

If loop gain values PG, IG and DG are maladjusted with respect to the system inertia present, position "overshoot" may occur at the end of the command motion. The appropriate adjustment of acceleration/deceleration greatly reduces the undesirable effects of overshoot. However, increasing accel/decel ramp time (see chapter 6) decreases system positioning time.

To optimally adjust loop gain, parameters PG, IG and DG in systems where the load inertia and friction are not known or are hard to determine, a signal is provided on one of the Unidex 1 DC control ports for monitoring this current command signal to the power stage portion of the Unidex 1 DC. (See the previous block diagram of the basic Unidex 1 DC control system.)

The current command monitor signal described previously is located on port B, pin 11 (see section 8-9). The diagrams shown next (figure 3-1) provide illustrations of: a signal displaying extreme end-of-motion position overshoot, a signal displaying minimal overshoot, and a signal displaying no overshoot.

The following is the Unidex 1 program used to generate the waveforms shown in figure 3-1. Note that this program uses "free run" commands "R+" and "R-" to generate motion. These commands are used, instead of the traditional index command "Xdddddd" (for example, "X 100000"). This is to insure a smooth accel/decel trajectory for taking measurements. Since the free run commands "R+" and "R-" *do not* specifically control distance (as does the Xdddd command), the motor should *NOT* be connected to the load during these measurements.

X F 50000 R+ *	Freerun + direction
DW 1 *	.1 Sec. dwell
SX *	Stop freerun
DW 5 *	.5 Sec. dwell
X F 50000 R- *	Freerun - direction
DW 1 *	.1 Sec. dwell
SX *	Stop freerun
DW 5 *	.5 Sec. dwell
RP *	Repeat program

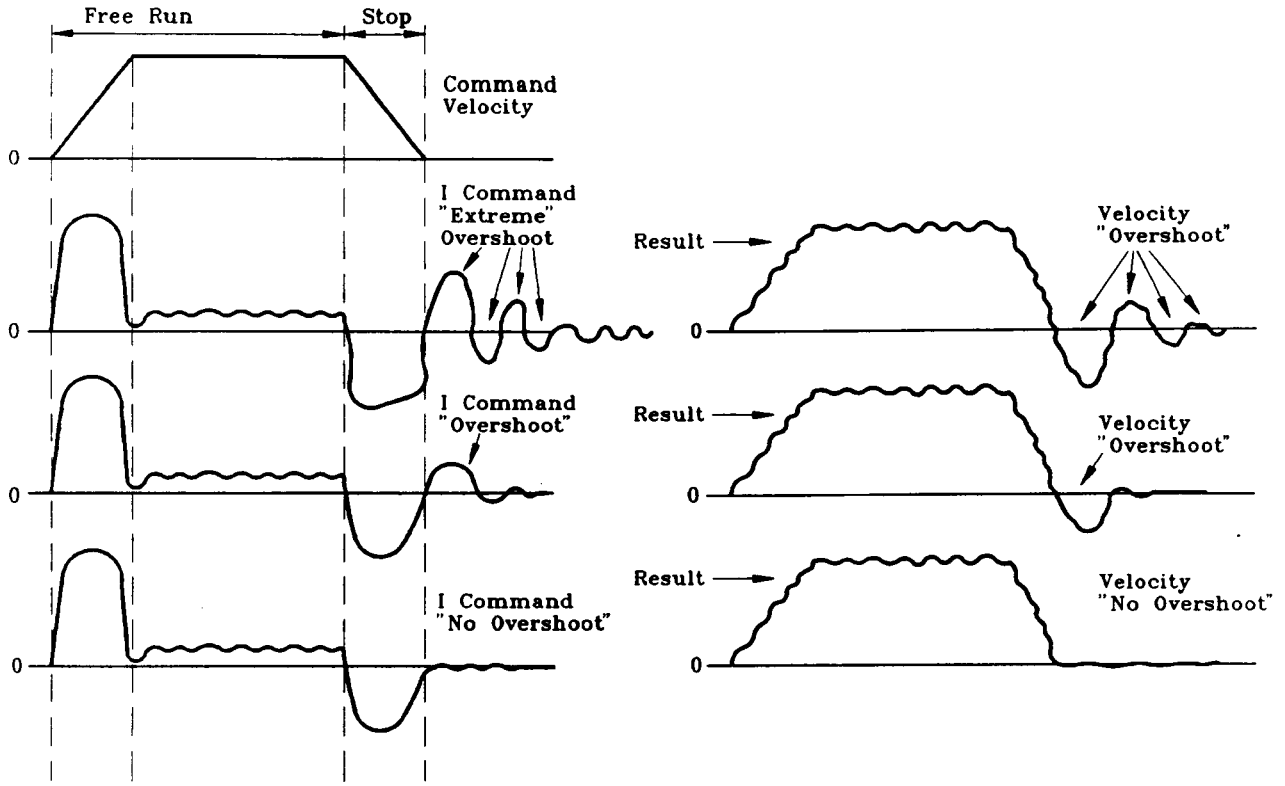


Figure 3-1: Various Characteristics of the Current Command Signal (Port B, Pin 11) based on Gain Setting PG,IG,DG

If the load characteristics (i.e., inertia and friction) are not known, the following "rule of thumb" may be used for the adjustment of gain parameters PG, IG and DG.

1. Set the integral gain term IG to its minimum value of 1. This minimum value will guarantee zero position error at the end of a commanded move. However, the time it takes to attain the absolute final position will be greatly lengthened with respect to the optimum setting.

2. Set PG to the value of 10 if the encoder feedback resolution is 2000 steps/rev (the standard resolution when the Unidex 1 is supplied with an Aerotech motor). Half or double this setting, if using a 4000 or 1000 step/rev encoder, respectively.

3. Set DG to a value of 40 if the encoder feedback resolution used is 2000 step/rev. Half this value if the encoder used is 4000 steps/rev. Double it if the encoder used is 1000 steps/rev.

4. Set the accel/decel parameter of the Unidex 1 to 50mSec, the feedback parameter to 100,000 steps/rev, and the distance to 400,000 steps. Again, half the values of feedrate and distance when using a 4000 steps/rev encoder, and double these values when using a 1000 steps/rev encoder. (Use the manual indexing modes discussed in section 4-5, 4-10 or 5-8 of this manual.)

5. Execute the motion parameter described above while monitoring port B, pin 11, with an oscilloscope. Slowly decrease the value of DG (by the amount of 3), until the wave form of diagram A, begins to appear.

6. Slowly increase parameter IG (by the amount of 3), until oscillation appears on the current command signal at pin 11. Then decrease this value accordingly, to stabilize this signal.

CHAPTER 4: UNIDEX 1 OPTIONS

Aerotech provides three control interface options for the Unidex 1 controller. These are the hand held terminal (TFX), the thumbwheel (TP) option, and the Thumbwheel Indexer option (TI). These options are outlined in figures 4-1, 4-2 and 4-3 respectively.

The hand held terminal (TFX) option provides the user with the ability to enter manual and program command instructions to the Unidex 1 directly, eliminating the need for a bulky RS-232 data terminal or computer interface.

The thumbwheel (TP) option is supplied for those users that require a very simple means of commanding program execution for the Unidex 1. This option is geared to factory automation, where program motion has already been pre-defined.

The thumbwheel indexer (TI) option is supplied for applications involving variable distance/feedrate motion commands under manual control. This option is similar to that of the TFX option when operated in the manual indexing mode, except that this option provides for a more simplified user-machine interface.

SECTION 4-1 HAND HELD TERMINAL (TFX)

The hand held terminal (TFX) (figure 4-1) will be referred to as the "hand held terminal" throughout the remainder of this text. The hand held terminal is supplied with a cable that is designed to plug into the

communication interface connector, Port A, of the Unidex 1 (see chapter 5). No additional interconnect wiring is required by the user. The hand held terminal receives its power directly from Unidex 1.

NOTE: Never connect or disconnect the hand held terminal from the Unidex 1 while power is applied.

It should be noted that the hand held terminal cable can be used to "daisy chain" 2 to 30 Unidex 1 controllers to one hand held terminal (see figure 5-3). Pin out definitions for the hand held terminal are shown in figure 5-3.

The hand held terminal will perform a self check when you first power up. This process takes approximately 5 seconds, after which the screen will be blank, except for the blinking cursor. If a blinking cursor does not appear, the hand held terminal may need to be initialized (see section 3-4).

NOTE 1: If you need to reset all Unidex 1s ("daisy chain" mode) on the bus through the hand held terminal, enter "<CTRL> D".

NOTE 2: A **Soft Reset** is also available and will reset only the active Unidex 1(s). It is initiated by entering "<CTRL> A" through the hand held terminal. This reset will stop motion (DECEL to stop) and program execution in all active Unidex 1s. All units that have responded to this reset will still be active. The following screen will be displayed:

1 EDIT 2 DISPLAY
3 RUN 4 ETC

(The above screen not displayed if more than one axis is active.)

NOTE 3: The escape character, <ESC>, should never be used, since certain Escape sequence codes will lock up the hand held terminal and will require it to be re-initialized (section 3-4).

To activate Unidex 1 through the hand held terminal after power up or a hand held terminal reset (note 1), you must enter:

nn <ENTER>

The "nn" designates the appropriate Unidex 1 device number.

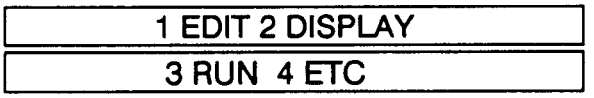
NOTE 4: If hand held terminal is used in the "daisy chain" mode, the characters entered will be "echoed" in duplicate until the given device is activated. (For example: #### nnnn).

NOTE 5: You must press the <SHIFT> key to make the above entry. Notice the cursor becomes larger when the upper case is active. Function keys are considered upper case. **If pressing a Function Key (F1 to F5) gives no result, check the cursor. You may be in the lower case mode.**

To enter a "<CTRL> A", "<CTRL> B" or "<CTRL> D", press the control key. The cursor will now shown [ct]. Then press A, B or D.

NOTE: Some models require the <CTRL> key to be held down while pressing A, B or D key.

After activating Unidex 1, you will see the following display:



Press function key F1 for **Edit**, F2 for **Display**, F3 for **Run** and F4 for **Etc.** The following sections will explain each of these modes.

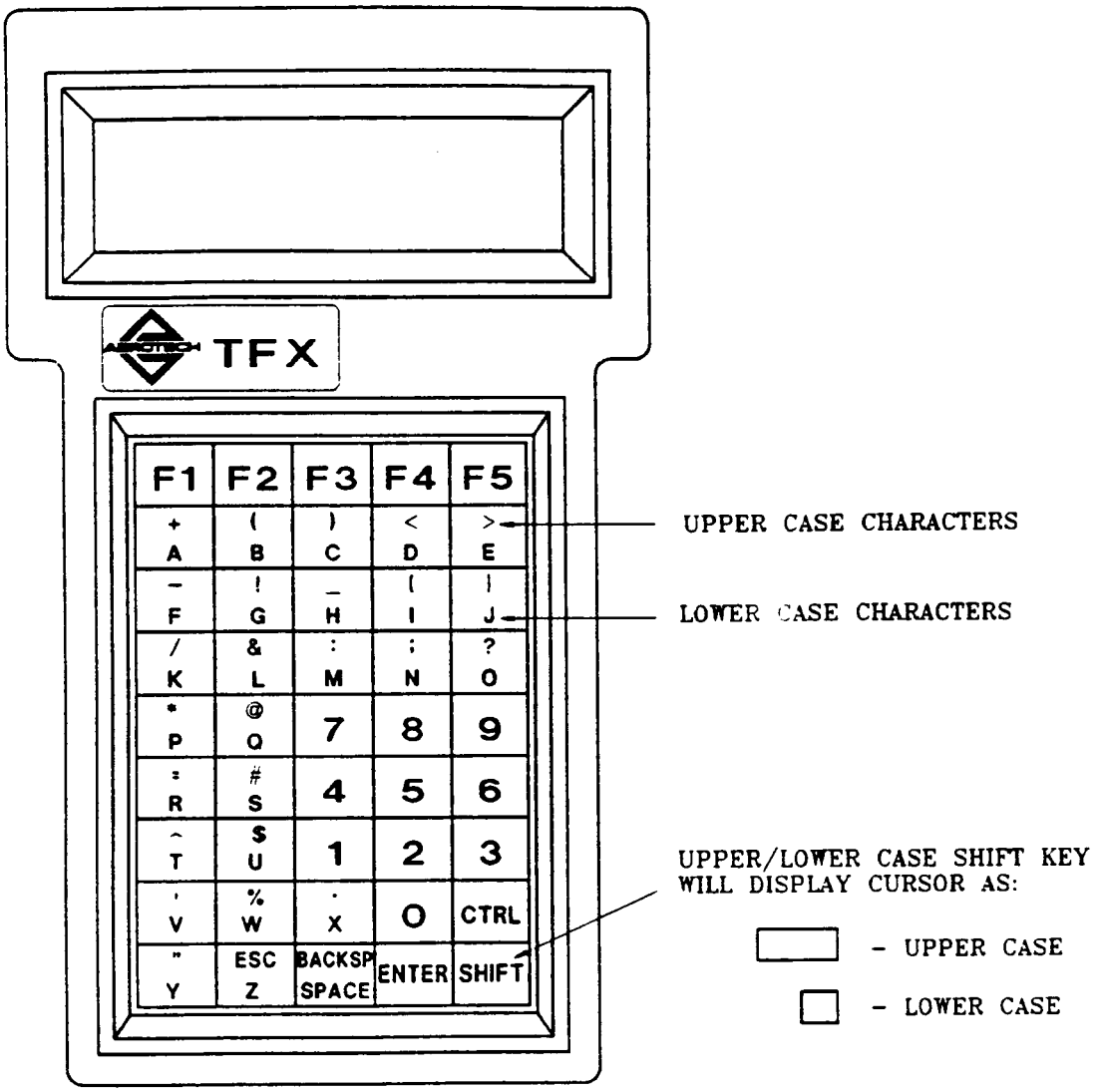


Figure 4-1: Outline of Hand Held Terminal Option for Unidex 1

SECTION 4-2 EDIT

Press F1, **Edit**, from the main menu screen, to enter the editing mode. You will see:

1 PRG 2 ALTER
3 DELETE 5 RTN

Press F1, **Program**, to input a new program, F2, **Alter**, to edit an existing one, F3, **Delete**, to delete a program, and F5, **Return**, to return to the previous menu.

NOTE: Sometimes F5 is not listed on the menu screens and yet still performs the function of returning you to the previous menu.

The following subsections will describe each Edit menu.

A. INPUT PROGRAM

Press F1, **Program**, to see:

INPUT PROGRAM
00 ENTER

Enter the new program number. You will then see:

INPUT BLOCK ?
1 CONTINUE 5 END

Press F1, **Continue**, to see:

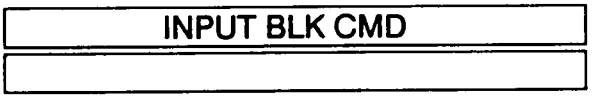
You may now begin to enter commands. (See chapter 5 for a summary of Unidex 1 program commands.)

B. ALTER PROGRAM

Once several blocks of commands are entered, you may find it necessary to step through the program in order to edit it. Press F5, **End**, of the Input Program mode to get back to the main edit menu. Then press F2, **Alter**, to see:

Enter the program number to see:

Press **F1** to display the program commands. Continue to press F1 to view subsequent command blocks, (**F2** to go back over previous blocks). When you get to a point where more commands must be inserted, press F5 to return to the Alter menu. Then press F3, **Insert**, to insert a block of commands. It will be inserted **before** the block where F5 was pressed. Pressing F3 will show:



You may now insert a command block.

NOTE: F5 is not valid at this time. To exit this situation without entering a command, enter an End-Of-Block character (*). You will be taken back to your program. If you enter an invalid command, you will see:

FORMAT WARNING
1 CONTINUE

Press F1 to re-enter the program.

Another way to insert commands is to press F2, **Get Block**, of the Alter Program menu. Press F5 to return to the Alter menu, then press F2, **Get Block**, to see:

ENTER BLOCK #
0000 ENTER

Enter a block number, such as 2. That program block will be displayed. You cannot directly change the block by overwriting it. Instead you must press F5 to see the Alter Program menu again:

1 DSP 2 GET BLK
3 INS 4 DEL 5 BK

You may now press F3, **Insert**, to Insert a new block of program commands before the program block just viewed. Press F4, **Delete**, to delete that program block. Press F2, **Get Block**, to go to a certain block. Press F1, **Display**, to re-enter the program, or F5, **Back**, to go back to the main edit menu.

Remember, once in the program by pressing any Alter menu function (except, of course, F5) you may step through your program by pressing **F1 and F2**, even though these functions are not displayed on the screen at the time. Use these functions to check your program while editing.

C. DELETE PROGRAM

Press F5, **Back**, to go back to the main edit menu. You will see:

1 PGM 2 ALTER
3 DELETE 5 RTN

Press F3, **Delete**. The display will show:

1 CLEAR MEMORY
2 DEL PGM 5 RTN

Press F5, **Return**, to return to the main edit menu.

Press F2, **Delete Program**, to see:

INPUT PROGRAM
00 ENTER

Enter a program number. That program will be deleted from Unidex 1 memory and the hand held terminal will again display the main edit menu.

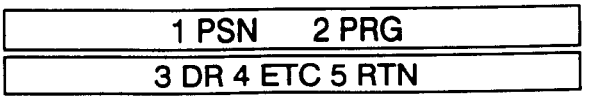
Press F1, **Clear Memory**, to clear all programs from memory. Remember, **pressing F1 when in the Delete Menu deletes all programs.**

D. RETURN

Pressing F5, **Return**, of the main edit menu will return you to the main menu.

SECTION 4-3 DISPLAY

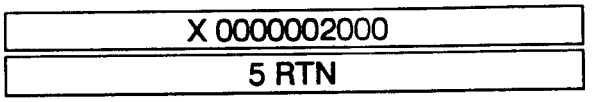
Press F2, **Display**, of the main menu screen to see:



The above menu will display: axis position, a program, the directory, input status, output status and setup parameters. Each will be described in a following subsection.

A. POSITION DISPLAY

Once in the display mode, press F1, **Position**, to see a display such as:



To maintain consistency with the Aerotech Unidex 11 Series controllers, the axis label (X) is allocated to this position. Since Unidex 1 is a one- axis system, X will always be displayed.

Press F5, **Return**, to return to the main Display menu.

B. PROGRAM DISPLAY

Press F2, **Program**, to see:

INPUT PROGRAM
00 ENTER

Once you enter any program number which exists in memory, you will see:

PRESS F1 TO CONT
PRESS F2 TO EXIT

Press F1 to step through the program. Each time F1 is pressed, a subsequent block will appear. (Note that the program editing can only be accomplished in the edit mode. See section 4-2).

Once you come to End-of-Program, press F5 to return to the main display.

C. DIRECTORY DISPLAY

Press F3, **Directory**, to see:

PRESS F1 TO CONT
PRESS F5 TO EXIT

If any program is residing in memory, pressing F1, **Continue**, will give the program number and how many bytes of memory it uses. For example:

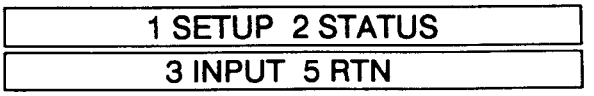
PROGRAM # 01
00035 BYTES

Press F1 again to see each subsequent program. When the last program has been displayed, the remaining bytes in memory that are free will be given. For example:

Press F5 to return to the main Display menu.

D. ETC

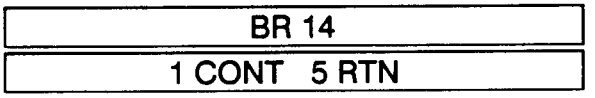
Press F4, **Etc**, to see:



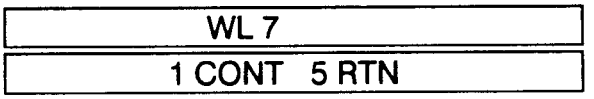
1. Setup Display

Press F1, **Setup**, to see the setup parameters chosen. (For detailed information on the Setup parameters, see chapter 3, Setup.)

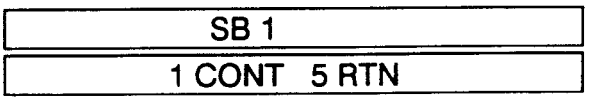
The first setup displayed is the Baud Rate:



Press F1 to continue. You will see Word Length:



Press F1 to see Stop Bits:



Press F1 again for Odd, Even or No Parity:

PY E
1 CONT 5 RTN

The subsequent screens describe the following:

Device address (1 through 30):

DA 01
1 CONT 5 RTN

Boot Program (0 through 99):

BP 01
1 CONT 5 RTN

Output Active State (high or low):

OT L
1 CONT 5 RTN

Accel/decel time (0-6500 mS) (see chapter 6):

AD 0250
1 CONT 5 RTN

Start/Stop frequency (default speed if move is too short for accel/decel rate; see section 6-2 and 6-5).

SS 0008000
1 CONT 5 RTN

D gain (0 through 99):

DG 50
1 CONT 5RTN

P gain (0 through 99):

PG 30
1CONT 5RTN

I gain (0 through 99):

IG 30
1CONT 5RTN

Home limit direction (+ or -):

HD-
1CONT 5RTN

Home limit speed:

HS 0008000
1CONT 5RTN

Thumbwheel Indexer index feedrate:

TS010000
1CONT 5RTN

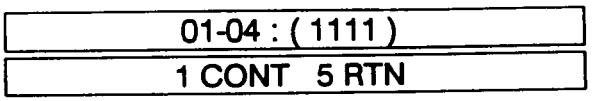
If F1 is pressed again, the display will "wrap around" to the Baud Rate again.

Press F5 at any time to return to the ETC display.

2. *Status Display*

Press F2, **Status**, to see the present Unidex 1 status. Status codes, described in chapter 10 (Troubleshooting) will be displayed if valid. Press F1 to continue and F5 to return.

The output status (high or low) is the next to the last status screen and is always displayed:



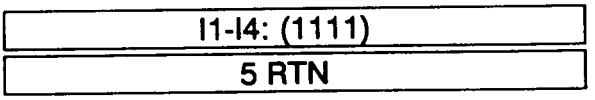
Press F5 to return or F1 for the next screen.

The software version is the last status screen and is always displayed:

(where xx represents the software level).

3. *Input Display*

Press F3, **Input**, to display the status of the inputs (high or low):



Press F5 once to return to the Etc menu, and again to return to the main Display menu.

SECTION 4-4 RUN MODE

Press F3, **Run**, from the main menu, to see:

1 IMD 2 AUTO
2 BLOCK 5 RTN

The run mode allows you to execute a program in the Auto or Block mode, or execute a string of commands in the Immediate mode.

Each of these modes will be covered in one of the following subsections.

A. IMMEDIATE MODE

Press F1, **Immediate**, to see:

INPUT BLK CMD

Enter a block of commands. If, for example, **HX*** were entered, the screen would display:

AXIS IN HOME
CYCLE

When Unidex 1 is finished executing the command, you will see the main Run menu displayed again.

B. AUTO MODE

Press F2, **Auto**, of the Run menu. You will see:

INPUT PROGRAM
00 ENTER

Once the program number has been entered, you will see:

WAIT FOR TRIGGER
1 YES 2 NO

If you want Unidex 1 to suspend execution of the program until it receives a trigger, press F1, **Yes**. You will see:

WAIT FOR TRIGGER
PRESS T

Press T to begin program execution. The tracking display will be displayed as they execute. When finished, the main Run menu will be displayed again.

If no trigger is required, press F2, **No**, when in the Auto mode. As soon as F2 is pressed, the program will execute, displaying the tracking display as it runs. When execution is finished, the main Run mode will be displayed again. However, at this point, multiple Unidex 1s may be enabled to listen (see section 2-4), so that they may all begin to execute the same program simultaneously when T is pressed.

C. BLOCK MODE

Press F3, **Block**, of the Run menu. You will see:

INPUT PROGRAM
00 ENTER

Once your program number and < ENTER > have been pressed, the first command block will execute. You will then see:

PRESS F1 TO CONT
PRESS F5 TO EXIT

Press F1, **Continue**, to execute the next block, or F5, **Exit**, to exit the program and re-enter the main Run menu.

You will continue to receive the above message as long as there are blocks to execute. When the end of the program is reached, the main Run menu will again be displayed.

Press F5, **Return**, to return to the Main menu.

SECTION 4-5 ADDITIONAL MODES

Press F4, **Etc**, of the Main menu to see:

1 STOP 2 INDEX
3 HM 4 RMT 5 RTN

The following subsections will describe each of the "ETC" menus.

A. STOP

Press F1, **Stop**, to stop the axis motion. When this mode can be accessed, pressing F1 will halt the axis move. (This command is useful in resetting a "free-run" command.)

B. INDEX

Press F2, **Index**, to see:

1 EXECUTE 2 INC
3 FD 4 DST 5 RTN

Once in the Index mode, you may set a feedrate (F3) and distance (F4) at which to execute (F1) your move. The move may be made in the Incremental or Absolute mode (F2). The details of these modes will be described in the following subsections.

NOTE: Press F2, **Incremental**, to switch to the Absolute mode. These two modes toggle, but the default mode is Incremental. Notice the difference in the Index menu display when the Absolute mode is active. You will now see:

1 EXECUTE 2 ABS
3 FD 4 POS 5 RTN

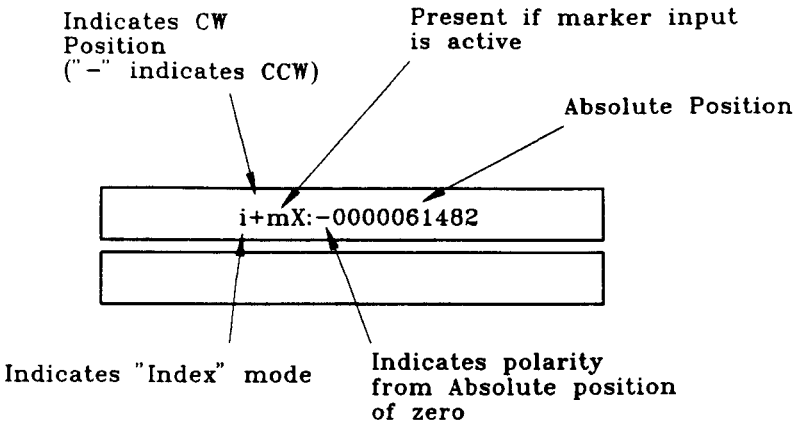
F4 changes to remind you that in the Absolute mode, data entered is not distance, but a *specific position* the axis is to reach.

1. Execute

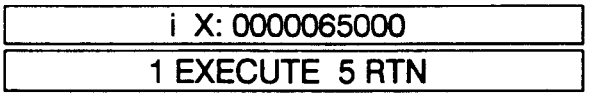
Press F1, **Execute**, to make the axis index (move a specific distance at a specific feedrate). This distance and feedrate is set via F3, **Feedrate**, and F4, **Distance**. If you do not enter new values, the last

ones entered will remain in effect. (They will stay the same even after a reset, or powering down and then up again.)

If you press F1, **Execute**, you will see a display similar to the following example.



Once execution of the move is complete, you will see a display such as:



Press F1, **Execute**, to re-execute the same index. Press F5, **Return**, to go back to the main Etc menu.

NOTE: You see the previous screen whether you are in the absolute or incremental mode. However, in the absolute mode, once a position is attained (such as 65000 in the above sample screen) pressing F1, **Execute**, will do nothing (the position has already been reached). To continue indexing, either change the position data in F4, **Position**, or change to the incremental mode by pressing F2.

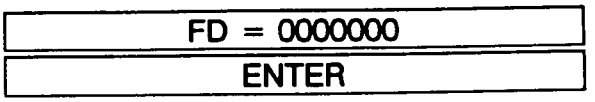
2. *Absolute/Incremental mode*

With F2, **Incremental/Absolute**, you may toggle between these two modes. As mentioned above, in the incremental mode, distance (F4) is how many steps the axis is to move from its present position.

Absolute is a specific position. Once it is reached, executing the same distance (position) will do nothing.

3. *Feedrate*

Press F3, Feedrate, to see a screen similar to:



When an index is executed, the axis will move at a feedrate of 10000 steps/second.

4. *Distance or Position*

Press F4, **Distance**, (when in the Incremental mode), to see a screen similar to:

When an index is executed in the incremental mode, the axis will move the distance specified (20000 steps in the above example) from the current position.

Press F4, **Position**, (when in the absolute mode), to see a screen similar to:

POS =-0000020000
ENTER

When an index is executed in the absolute mode, the axis will move to the absolute position (-20000 steps in the previous example).

C. HOME

Pressing F3, **Home**, of the Etc menu, will send the axis home. Press F3 to see:

AXIS IN HOME
CYCLE

Once home is reached, you will see:

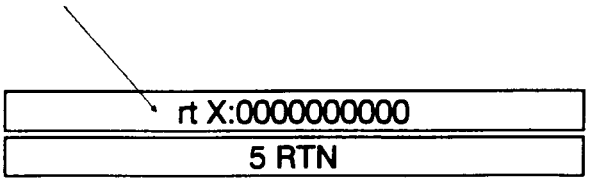
AXIS AT HOME
1 HM 5 RTN

Press F1 to start another home cycle. Press F5 to return to the main menu.

D. REMOTE

Press F4, **Remote**, to see a screen such as:

Indicates remote mode



The remote display monitors any pulses coming in from the external clock and direction inputs (input/output connector – see section 8-8, subsection F).

SECTION 4-6 THUMBWHEEL (TP) INITIALIZATION

The Thumbwheel (TP) (figure 4-2) is supplied with a molded interconnect cable that is designed to plug directly into the communications interface connector port A of the Unidex 1 (see section 8-10). No additional interconnect wiring is required by the user. The thumbwheel (TP) derives its power directly from the Unidex 1.

NOTE: Never connect or disconnect the Thumbwheel (TP) from the Unidex 1 while power is applied.

Unlike the hand-held terminal options (section 4-1), the Thumbwheel (TP) option *cannot* be "Daisy-chained" to other Unidex 1 controllers.

The Unidex 1 Thumbwheel (TP) Option will call and run any of 99 possible programs in Unidex 1 memory. The programs must have pre-

viously been entered into Unidex 1 (as described in this chapter and in chapter 5). Upon completion of the program, Unidex 1 will respond with a READY LED signal. If the program called has not been run completely or an ERROR has occurred, Unidex 1 will respond with an error LED signal.

SECTION 4-7 THUMBWHEEL (TP) INTERFACE

The Unidex 1 Thumbwheel (TP) will connect directly to Unidex 1 Port A (9 pin "D") connector. The Unidex 1 Thumbwheel option can only be used with a single Unidex 1 and will not perform in a multi-axis (Daisy-chain) configuration. The Unidex 1 Thumbwheel option requires that Unidex 1 be set to the following specifications (see section 3-9 for a description of these codes):

BAUD RATE	**: 9600**
STOP BITS	**: 1**
WORD LENGTH	**: 7**
PARITY	**: EVEN**

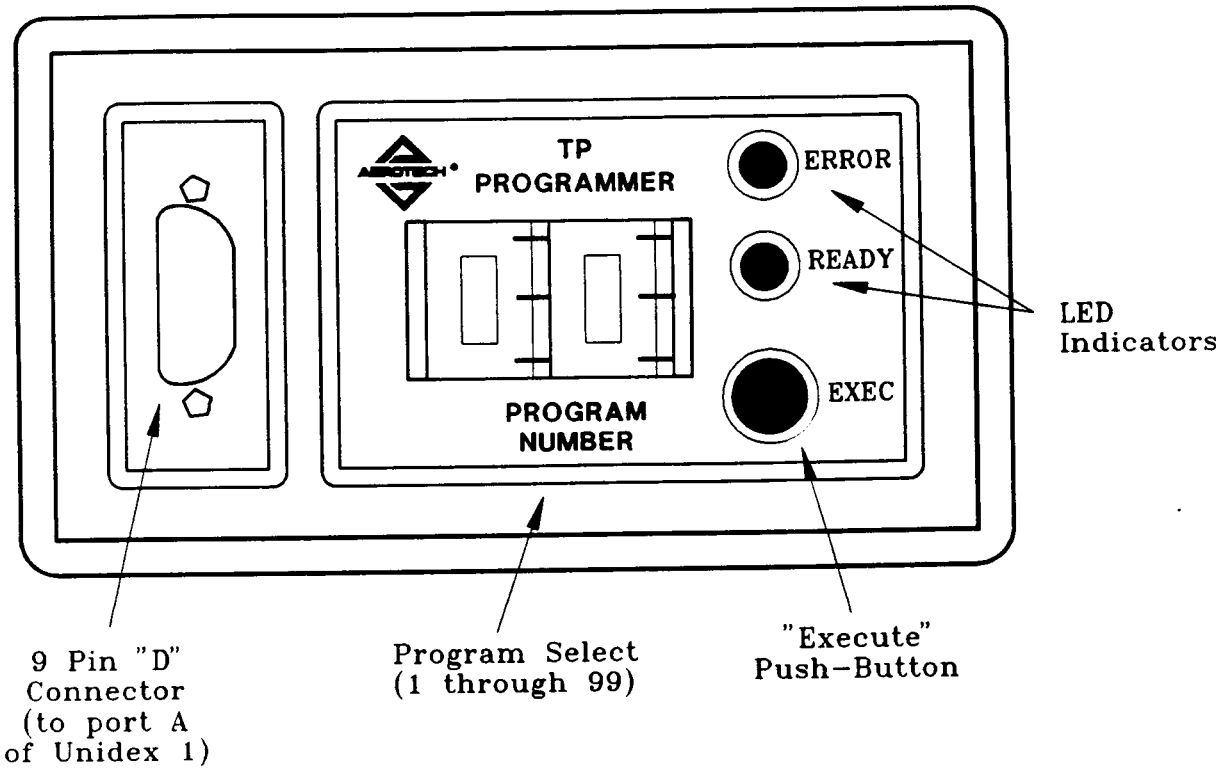


Figure 4-2: Thumbwheel (TP) Option

SECTION 4-8 THUMBWHEEL (TP) POWER UP

The Unidex 1 and Unidex 1 Thumbwheel (TP) option must be connected before power up to insure that the Thumbwheel (TP) option will be recognized. When the Unidex 1 is powered up, the READY LED should be on and the ERROR LED should be off. To check the Thumbwheel (TP) option, set the thumbwheels to 00 (Program 00 *cannot* exist) and press the EXECUTE push-button. Upon pressing the push-button, both the READY and ERROR LEDs will be on. If both LEDs are not on, the problem may be any of the following:

1. Unidex 1 and Unidex 1 Thumbwheel (TP) are not powered up properly.

2. Check interface cable between Unidex 1 and Unidex 1 Thumbwheel (TP).

3. Check communication parameters (see section 3-9).

SECTION 4-9 THUMBWHEEL (TP) OPERATION

To execute a program using the Unidex 1 Thumbwheel (TP) option, set the thumbwheels for the desired program and press the EXECUTE push-button. The program execution will begin immediately and will be indicated by both the READY and ERROR LEDs being off. When the program has completed and there are no errors, a ready signal will be sent and the READY LED signal will re-light. If the program does not exist or an error occurs during execution of the program, the program execution will be terminated and a error signal will be sent. The error signal will cause the READY and ERROR LEDs to be turned on simultaneously.

The EXECUTE push-button should not be pressed while both LEDs are off, since it may clear the Unidex 1 response signal and

result in the the ERROR and READY LED status not being set. Pressing the EXECUTE push-button while the program is being executed (READY and ERROR LEDs off) may also cause brief interruptions in the program execution. If EXECUTE is pressed at the same time that a Unidex 1 response signal is being sent back, the displayed LED status may be cleared or incorrect. *A questionable or missing LED status cannot be recovered.* Commands from the Unidex 1 Thumbwheel (TP) are not accepted during program execution and cannot be used to terminate any programs in progress.

SECTION 4-10 THUMBWHEEL INDEXER (TI)

The thumbwheel indexer (TI) shown in figure 4-3, is supplied with a molded interconnect cable that is designed to plug directly into the communications interface connector port A of the Unidex 1 (see section 8-10). No additional interconnect wiring is required by the user. The thumbwheel (TP) derives its power directly from the Unidex 1.

NOTE: Never connect or disconnect the thumbwheel (TP) from the Unidex 1 while power is applied.

Unlike the hand-held terminal options (section 4-1), the thumbwheel indexer (TI) option *cannot* be "daisy-chained" to other Unidex 1 controllers.

The Unidex 1 thumbwheel indexer option is similar to the thumbwheel (TP) option discussed in sections 4-6 through 4-9, except that it provides a much greater level of manual motion control. The TI option allows not only the selection of program execution, as with the TP option, but also the selection of 9 additional levels of motion commands and command modes. These are as follows:

Mode 0/ Home	Send axis to home position. (Similar to RS-232 command "home"; see section 5-8 D.)
Mode 1/ Index	Move axis the prescribed distance as set by "data" thumbwheel settings (limited to 7 digits). The feedrate can be set by Mode 2. The power-up feedrate defaults to the feedrate set in the Thumbwheel Indexer index speed command (see section 3-9 O). (Similar to RS-232 command "Index"; see section 5-8 B.)
Mode 2/ Feedrate	Set axis feedrate to be used with all subsequent indexes. (Similar to RS-232 command "Index"; see section 5-8 B.)
Mode 3/ Accel/Decel	Set axis accel/decel rate to be used with all subsequent indexes. (Similar to RS-232 command "Accel/Decel"; see section 5-8 AA.)
Mode 4/ Absolute/ Incremental	Set index mode so that index length set in "data" thumbwheel is relative to preset point (absolute) or with respect to last move (incremental). (Similar to RS-232 command "Inc/Abs"; see section 5-8 P.)
Mode 5/ Load Position	Set axis position tracking register (limited to 7 digits). (Similar to RS-232 command "Load Position Register"; see section 5-8 O.)
Mode 6/ Run Program	Execute specific Unidex 1 program (1 through 99). (Similar to RS-232 command "Auto Run"; see section 5-7 B.)
Mode 7/ Output	Set the four output control signals to 1 (true) or 0 (false) as designated by the first four digit positions of the "data" thumbwheel. Values 2 through 9 for the given digit specify the "don't care" state. (Similar to RS-232 command "Output State"; see section 5-8E.)

Mode 8/ Slew — Slew (jog) axis at specified feedrate (±0 to 250,000 steps/sec) as designated by the data thumbwheels. The sign indicates direction (+ for CW, - for CCW). The "Execute" push- button controls the slew (or jog) process.

Mode 9/ Reset — Reset the Unidex 1 (equivalent to a Ctrl-A "soft reset"). The "execute" push-button performs the reset. (Similar to RS-232 command "Reset"; see section 5-7 AA .)

For the above mode definitions, modes 1, 2, 3, 5, 6, and 8 require the respective numerical parameter of the given command be set up in the "data" entry thumbwheel. Standard numerical limits for the given command apply, i.e., accel/decel has a value limitation of between 0 through 6500 mSec, etc.

For modes 0 and 9, data entry on the "data" thumbwheel are not applicable. Depress the "Execute" push-button to execute these commands.

For mode 7 (output), the first four (least significant) data thumbwheel switches are used (digit 1 for output 1, digit 2 for output 2, etc.). With these four digits, only value 0 (false) and 1 (true) are applicable as set or reset indicators. Digit values 2 through 9 signify "don't care" for the given output.

For mode 4, the "-" data digit selects the absolute mode and the " + " data digit selects the incremental mode. All other data digits are ignored by this mode.

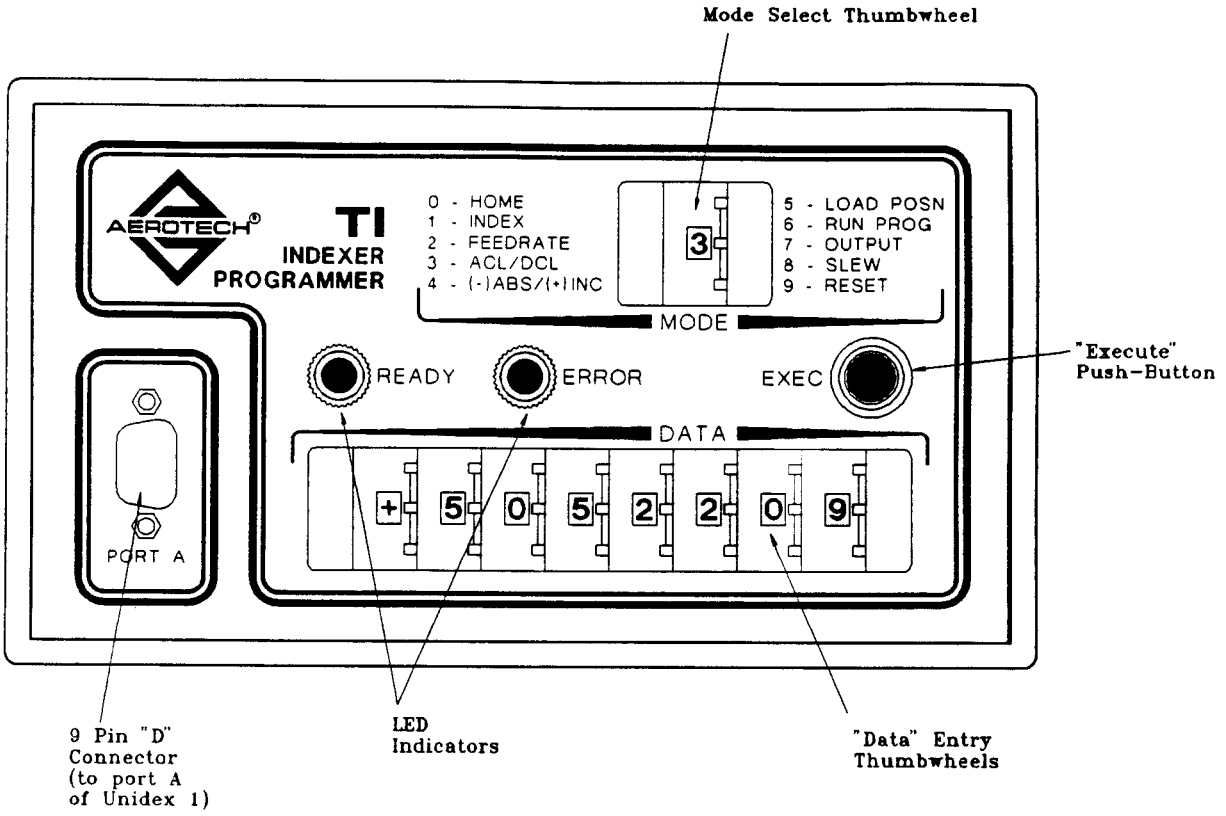


Figure 4-3: Thumbwheel Indexer (TI) Option

SECTION 4-11 THUMBWHEEL INDEXER (TI)

The Unidex 1 Thumbwheel Indexer (TI) will connect directly to Unidex 1 port A (9 pin "D" connector). This Unidex 1 option can only be used with a single Unidex 1 and will not perform in a multi-axis (daisy-chain) configuration. The Unidex 1 Thumbwheel Indexer option requires that Unidex 1 be set to the following specifications (see section 3-9 for a description of these codes):

BAUD RATE	**: 9600**
STOP BITS	**: 1**
WORD LENGTH	**: 7**
PARITY	**: EVEN**

SECTION 4-12 THUMBWHEEL INDEXER POWER-UP

The Thumbwheel TI option is similar to the TP option in operation. The "Ready" and "Error" LED's indicate "Execution Complete" and/or Error conditions. Review sections 4-8 and 4-9 for more information.

CHAPTER 5: COMPUTER INTERFACE (PORT A)

The Computer Interface (i.e., RS-232C daisy chain) option for the Unidex 1 makes it possible to control Unidex 1 from a host device (CRT terminal, personal computer, etc.) via serial port A. A simple command sequence to the Computer Interface gains control of the device. The host may, from this point on, perform different tasks using Unidex 1. These tasks include executing command block(s) in the immediate mode, downloading a program into the user memory, running a program from the memory in auto-run or block-run mode, and reading the axis position, system statuses, directory, program or the entire memory.

Unidex 1 may also be set up for interactive control. In this mode, Unidex 1 sends a service request after executing a program or a command block, or if there is an error condition. Subsequently, the host device is required to poll Unidex 1 before proceeding further.

SECTION 5-1 REQUIRED HARDWARE

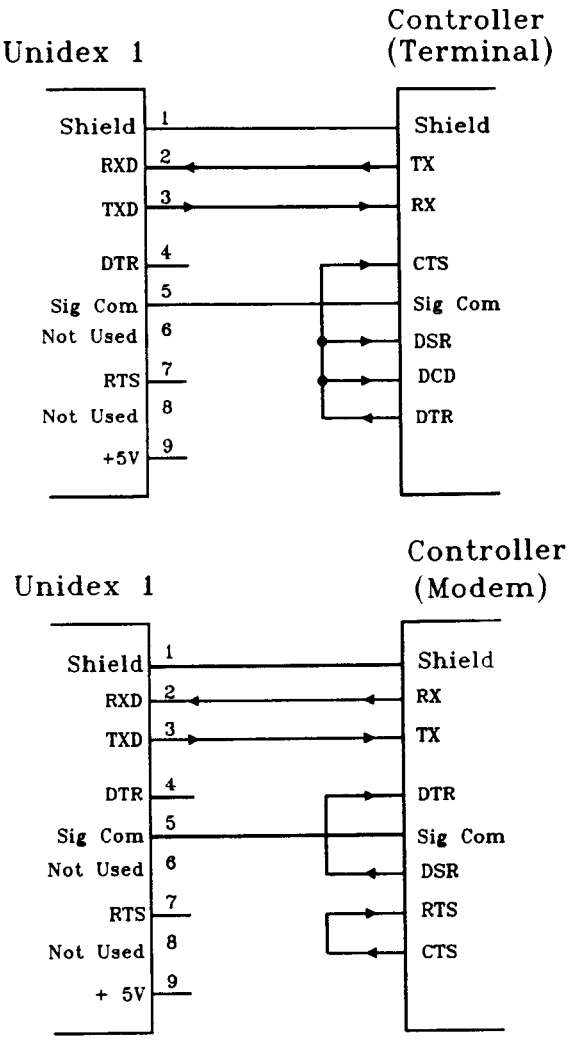
The Port A Computer Interface port (see also section 8-10) is a 9 pin "D" type female connector implementing the following signal lines:

PIN 1	Shield
PIN 2	Receive data (RX)
PIN 3	Send Data (TX)
PIN 4	Data Terminal Ready (DTR)
PIN 5	Signal Common (SG)
PIN 6	No Connection
PIN 7	Request To Send (RTS)

PIN 8	No Connection
PIN 9	+5V

Figures 5-1, 5-2 and 5-3 show three examples of Communication Interfaces. The first two show interface connections required for RS-232 devices. Figure 5-3 shows a multi-axis Hand Held Terminal (Hand Held Terminal is explained in chapter 4) interface example.

The daisy chain configuration can be extended to 30 Unidex 1s. To add more units, repeat Unidex 1 connections as shown in figures 5-2 and 5-3.



NOTE: RS-232C Interface cable must be shielded

Figure 5-1: Single-Axis RS-232 Interface

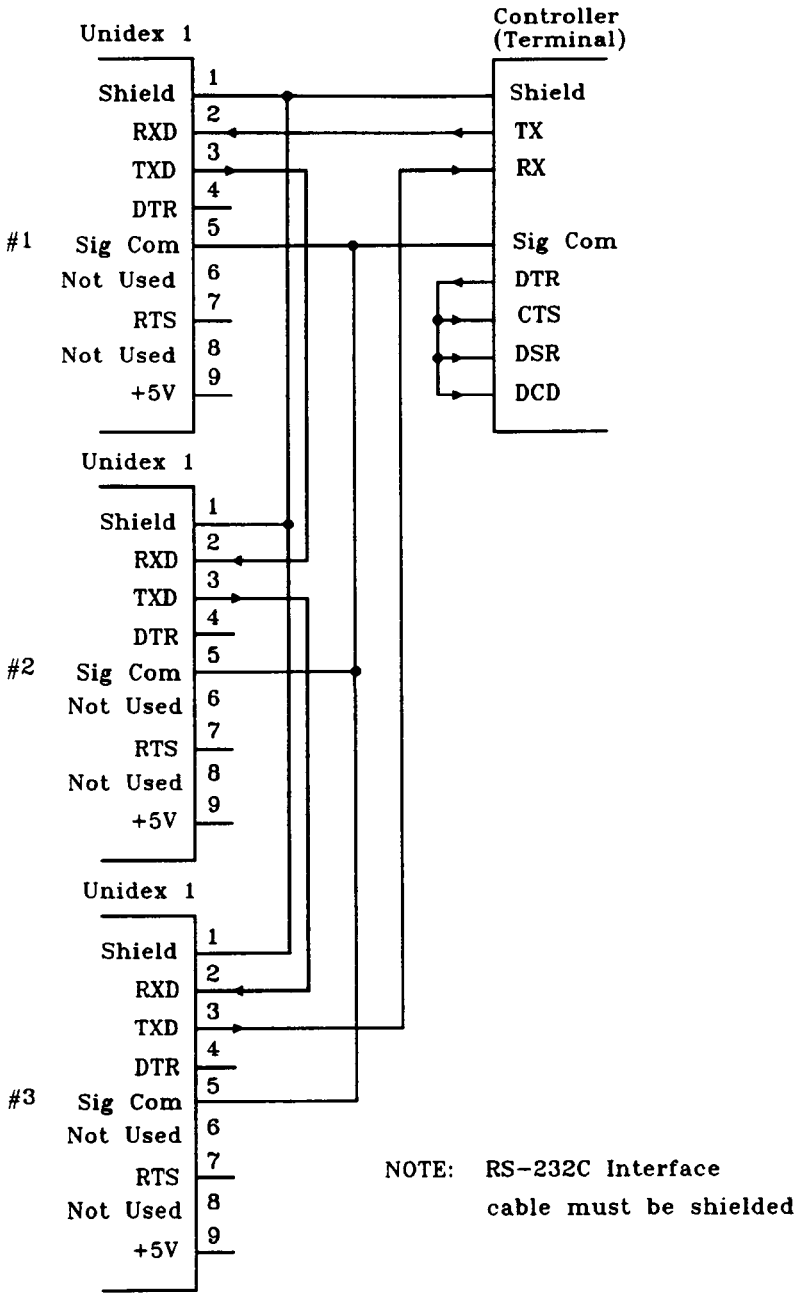


Figure 5-2: Multi-Axis (Daisy Chain) RS-232 Interface

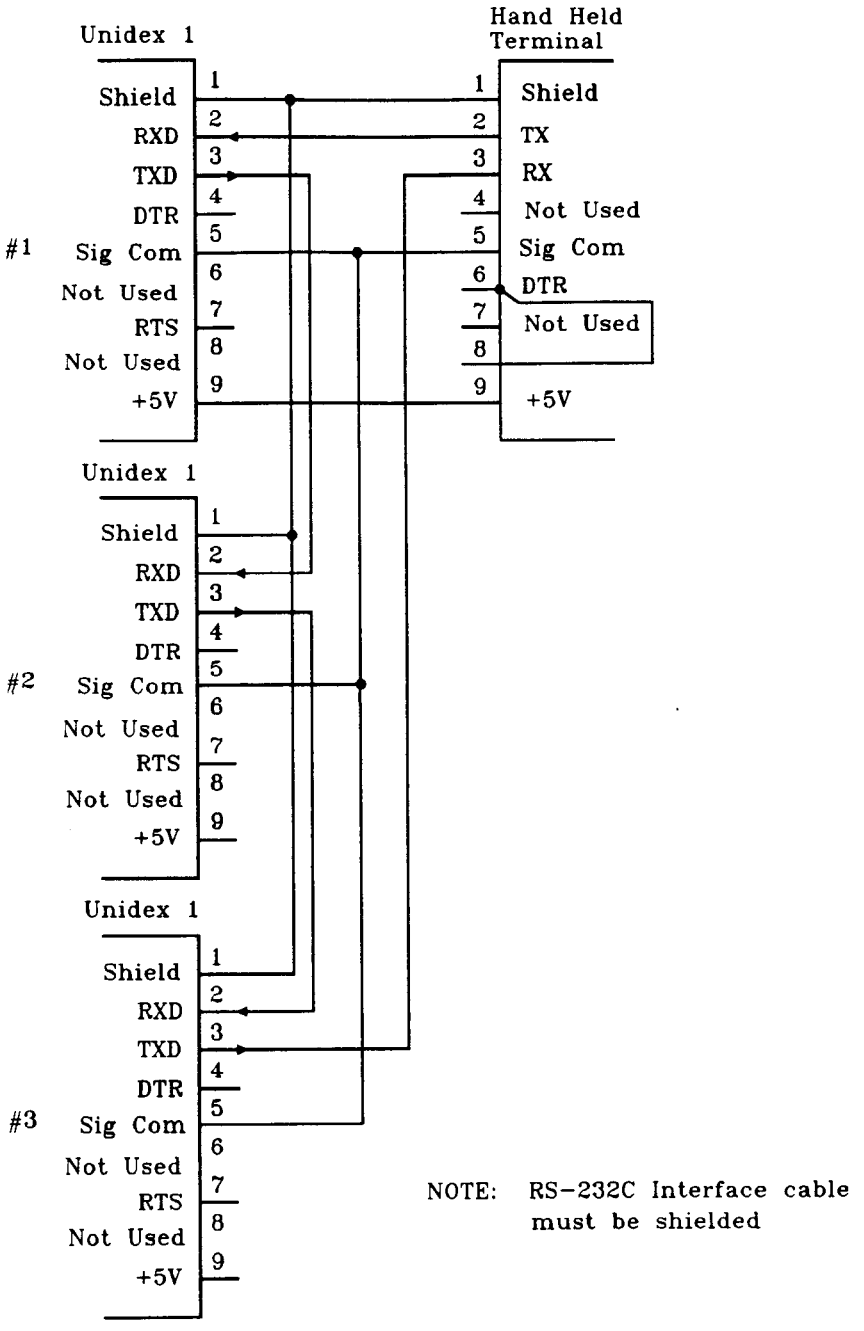


Figure 5-3: Multi-Axis (Daisy Chain) Hand Held Terminal Interface

SECTION 5-2 SETTING UP COMPUTER INTERFACE

It is required that certain communication parameters of the sending device match those of the receiving device to enable data to be transferred. These parameters determine the format for communication. There are four parameters that must be set up in Unidex 1. They are:

Baud Rate — This relates to the rate of transfer of data. The value represents the "number of bits per second" of transfer. Unidex 1 may be set up for a baud rate of from 50 to 9600. (Settings include 50, 75, 109.92, 134.58, 150, 300, 600, 1200, 1800, 2400, 3600, 4800, 7200 and 9600.)

Character Length — Each byte of data is encoded in either 7 or 8 bit length.

Number of Stop Bits — The end of each byte is indicated by the stop bit sequence. This is also the minimum time required by the receiver to start looking for the next byte after the 7 or 8 bits of the current byte are received. Number of stop bits may be 1 or 2.

Parity — The parity bit is an extra bit added to the character (in addition to the 7 or 8 bits described above) so as to make the number of "1" bits either even or odd, thus adding an extra check for accuracy of data. Parity may also be disabled so that no parity bit is added. Unidex 1 may be set up for ODD or EVEN or DISABLED parity.

Unidex 1 defaults to the following values upon power-up if there is no battery backed user memory or if the "Load Default" operation is performed.

Baud Rate	: 9600
Character Length	: 7 Bits
No. of Stop Bits	: 1
Parity	: Even

To change the above values, Unidex 1 has to be put in the Set-up Mode as described in chapter 3.

SECTION 5-3 COMPUTER INTERFACE (PORT A)

Unidex 1 is ready for communication at power up if the Computer Interface format has been correctly set up or if the default values are already the required format.

The host device must now send the "attention" command to Unidex 1. This consists of the character string > > followed by the 2 digit device number and <CR> (see section 3-2 for more information). The two > > signs must be consecutive. Unidex 1, upon receiving the "attention" command, will accept and execute any valid command.

The attention command for the hand held terminal mode is "## nn", as explained in chapter 4. (The ## characters should not be used for the operation described in this chapter.) Unidex 1 will echo back characters which must be cleared from the computer's input buffer.

SECTION 5-4 TYPES OF COMMANDS

Commands sent to Unidex 1 via the Computer Interface may be classified into two types, System commands and Program commands. Tables 5-1 and 5-2, which follow, respectively summarize these two command sets.

Unidex 1 System Command Summary

NOTE: System commands must be entered as UPPER CASE letters.

>>nn<CR>	: Activate RS-232C interface (nn = Device Address)
A nn <CR><LF> *	: Run program #nn in Auto mode (nn = 0 to 99)

B nn <CR>	: Run program #nn in Block mode (subsequent <CR> will execute successive program blocks)
C <CR>	: Reset Unidex 1 (to power up conditions)
D <CR>	: Cancel Remote mode
E nn *	: Begin downloading program #nn. Existing program #nn will be deleted automatically
E $ nn *	: Delete program # nn
E $ 00 *	: Clear program memory (all programs)
F <CR>	: Insert block (line) numbers when printing programs (for editing purposes)
G <CR>	: Cancel block (line) number printing (default)
H <CR>	: Put Unidex 1 in Hold mode (Trigger required to execute programs). Hold mode cancelled by O <CR> <LF>
I (string)* <CR>	: Execute program block (string) in the Immediate mode ("string" is any valid motion program command)
J <CR>	: Set up Unidex 1 to send Service Request after execution
K <CR>	: Cancel set up to send SRQ (default)
M <CR>	: Set up to transmit status in binary format (default)
N <CR>	: Set up to transmit status in Hex-ASCII format
O <CR>	: Cancel Hold mode (default)
PX <CR>	: Print X axis position register value
PD <CR>	: Print directory listing
Pnn <CR>	: Print program #nn
P00 <CR>	: Print all programs in memory
PS <CR>	: Print status bytes
PF <CR>	: Print software level
Q <CR>	: Query (serial poll); Unidex 1 returns a byte
R <CR>	: Enable Remote mode from host controller
T <CR>	: Trigger to start program execution
<DEL> (hex 7F) or <CTRL> D(hex 04)	: Hardware reset Unidex 1
<CTRL> A(hex 01)	: Reset (cancel the in-progress motion of Unidex 1)
<CTRL> B(hex 02)	: Deactivate all Unidex 1s
<CTRL> D(hex 04)	: Hardware reset all Unidex 1s
<CTRL> Q (hex 11)	: Start transmission after <CTRL> S is received
<CTRL> S (hex 13)	: Stop transmission until <CTRL> Q is received

* The line feed **<LF>** is optional and is not required to execute any command.

Unidex 1 Program Command Summary

* or /	: End of block (terminates the block commands listed below)
X Ffffffff Dddddddddddd* (see note 1)	: X axis move at feedrate ffffff steps/sec a distance of ddddddddddd steps, CW (v="+") or CCW (v="-")
X F ffffff Rv(-) *	: X axis free run at feedrate ffffff steps/sec, CW (v="+") or CCW (v="-")
DW nnn.n *	: Dwell nnn.n seconds
H X *	: X axis home
OT 10XX *	: Output O1, O2, O3, O4 (Opto-coupler outputs) 1: True 0: False X: Don't care
IT X001 *	: Wait until input state I1,I2,I3,I4 (Opto-coupler inputs) matches X001
OS 0011 *	: OUT/STOP state (drive outputs to 0011 on activation of feedhold input)
OR XXXX *	: OUT/RUN state (drive output to XXXX on deactivation of feedhold input)
RS nnnn * (see note 2)	: Repeat loop start nnnn times
RE * (note 2)	: Repeat loop end
RC 10X0 * (note 2)	: End Repeat loop on input condition 10X0
RX *	: Start free run axis X after Stop (S X)
SX *	: Stop free run axis X
RP * (note 2)	: Repeat program
LXvddddddddddd *	: Load position register X with ddddddddddd, v = + or -
IN *	: Set Incremental mode position tracking
AB *	: Set Absolute mode position tracking
LB nn * (note 2)	: Label # nn (assign block to label #nn)
GT nn * (note 2)	: GoTo label #nn
GS nn * (note 2)	: GoSub label #nn
SR * (note 2)	: Subroutine Return
PS * (note 2)	: Program Stop (end of program execution)
CT nn I 10X0 * (note 2)	: GoTo label #nn if condition input state is 10X0, else continue
CS nn I 1XXX * (note 2)	: GoSub label #nn if input state is 1XXX, else continue
RI 0X10 *	: Go to remote mode if input conditions match (stay in remote if inputs remain the same)
MT nn * (note 2)	: GoTo label #nn on marker
MS nn * (note 2)	: GoSub label #nn on marker
CO *	: Corner rounding
NC*	:Non-corner rounding
BI nn I XXX1 * (note 2)	: Branch to label #nn on interrupt input condition XXX1
SI nn I X1X0 * (note 2)	: GoSub label #nn on interrupt input condition X1X0
EI * (note 2)	: Enable interrupt
DI * (note 2)	: Disable interrupt

AD nnnn *	: Accel/Decel ramp time in milliseconds
LP ddddddddd * (note 2)	: Load positive limit with ddddddddd
LM ddddddddd * (note 2)	: Load minus limit with ddddddddd
EP * (note 2)	: Enable positive limit
EM * (note 2)	: Enable minus limit
EL * (note 2)	: Enable both limits
DP * (note 2)	: Disable positive limit
DM * (note 2)	: Disable minus limit
DL * (note 2)	: Disable both limits
;	: Program comment may begin after ; (comments terminated by <CR>)
%	: End edit (downloading)

NOTE 1: "X" axis designated to maintain consistency with the Unidex 11 series Controllers.

NOTE 2: Command is not valid in the Immediate mode.

Following is a brief description of both command types.

SYSTEM COMMANDS

These commands interact with Unidex 1 as a device and perform operations such as resetting Unidex 1, printing a program, printing position values, running a program, downloading a program, transferring status byte information from Unidex 1, etc. Each system command establishes a mode of operation once it is received by Unidex 1. A detailed description of these commands is listed in section 5-7.

PROGRAM COMMANDS

These are the user program blocks in a motion program that Unidex 1 executes when running a program in the auto or block mode. Program commands are valid only if entered in the immediate or edit mode. A detailed description of these commands is listed in section 5-8.

SECTION 5-5 SERVICE REQUEST AND SERIAL POLL

Service request is an important concept in device control when there is a controller (host computer) as the master and the controlled device (such as a printer or Unidex 1) as the slave. The purpose of Service Request is for the slave device to catch the attention of the master controller. Typically, a controller has more than one device being controlled by it and it would be very inefficient for the master controller to continually read the statuses of all the devices to check for error states. The controller (slave) device therefore has the capacity to send a signal (service request) to the master controller whenever it requires the attention of the master. The reason for requesting service may be to signal an error condition or to signal the completion of a task.

Unidex 1 implements a service request by sending a predetermined byte of data followed by a < CR >. The controller (master) may be set up to be interrupted by this data byte whereupon it must take a necessary action. *Service requests are only sent if the Unidex 1 is the only active unit.* Otherwise, it will wait until it is the only active unit.

The minimum necessary action that the controller must take once Unidex 1 has sent the service request signal is to poll Unidex 1 by sending the query (serial poll) command Q < CR >. Unidex 1 waits until the query command is received, and will not respond to any other system command until this is done. The purpose of the query command Q < CR > is to transfer a status byte from Unidex 1 to the controller. The 8 bits of this status byte represent different internal states of Unidex 1. Serial polling may be done any time the Computer Interface is active, not necessarily only after a service request.

The status byte may be analyzed by the controller to determine the cause of the service request. Each of the bits in the status byte is described as follows:

	Zero	One
BIT 0	Incremental mode	Absolute mode
BIT 1	Not running a program	Running a program
BIT 2	Block run mode	Auto run mode
BIT 3	Note used	Not used
BIT 4	Communication disabled	Communication enabled
BIT 5	Inactive - Not executing a command in immediate mode	Active - Executing a command in immediate mode
BIT 6	No service request signal sent	Service request signal sent, waiting for Q
BIT 7	No errors detected	Error detected

Unidex 1 may be put into the Service Request Mode by the system command J <CR>. The default Service Request data byte sent by Unidex 1 defaults to %. Unidex 1 sends % **<CR>** as a Service Request signal.

This byte may also be programmed by the user by entering a character immediately after J. This entry becomes the new Service Request character. For example, if **J! <CR>** were entered, the Service Request character would become ! and Unidex 1 would send ! **<CR>** as the Service Request signal.

NOTE: The characters >, < and # should not be used, since they are activation characters.

The Service Request Mode may be cancelled by sending the system command **K <CR>**. In this mode, Unidex 1 will not send a Service Request signal for any reason. This is the default mode. In this mode,

to determine if an immediate command or a program has been completed, a serial poll (Query) may be done as explained on the previous page and the status byte analyzed (bit 1 and bit 5). When these bits are clear (zero), Unidex 1 is ready to accept the next command.

When in the Service Request Mode, Unidex 1 sends a Service Request (SRQ) character under the following conditions:

1. When an Immediate command execution is complete.
2. When a program is completely executed in the Auto Run mode.
3. When a block is executed in the Block Run mode.
4. When a run time error condition is generated and the program is aborted.
5. When an axis limit is activated.
6. At the end of a program download operation, if an error was generated while downloading. (The SRQ character is sent by Unidex 1 after the "%" that ends the downloading of the program.)
7. When it is requested that a nonexistent program be printed. (If Pnn <CR>, Ann <CR> or Bnn <CR> is sent to Unidex 1 and program number "nn" does not exist, Unidex 1 will send the SRQ character.)

NOTE: For more information on error bytes, see section 5-6.

SECTION 5-6 ERROR CODES

An error condition may be detected by the host computer by checking the most significant bit (bit 7) of the serial poll status byte (section 5-5). If this bit is set to "1", an error has occurred.

To further determine the type of error, the system status may be read by the host by sending the command **PS <CR>**. Unidex 1 will send back 9 bytes followed by <CR><LF>. These 9 bytes represent a complete status report of Unidex 1. They are described as follows:

BYTE 1 Same as Serial poll status byte described in section 5-5.

BYTE 2 EDITOR ERROR STATUS

	Zero	One
BIT 0	No illegal character during download	Illegal character during download (illegal command code)
BIT 1	Memory not full during download	Memory full during download
BIT 2	No user memory checksum error	Checksum error during download of program
BIT 3	No command format error	Command format error
BIT 4	No memory repair	Memory repair
BIT 5 - 7	Not used	Not used

NOTE: If one of these errors is generated during download operation, Unidex 1 will send an SRQ (Service Request) character if in the Service Request Mode. It is recommended that the user then edit and correct that program.

BYTE 3 RUNTIME ERROR STATUS 1

	Zero	One
BIT 0	Axis not in hardware limit	Axis in hardware limit
BIT 1	No axis software limit	Axis software limit
BIT 2	No position loop error (PID Loop)	Position loop error (PID Loop)
BIT 3	No illegal program exit	Illegal program exit
BIT 4	No illegal byte in memory	Illegal byte in memory
BIT 5	Program number valid	Invalid program called out for run (does not exist)
BIT 6	Memory not clear	No programs in memory (memory clear)
BIT 7	No user memory checksum error	User memory checksum error

BYTE 4 RUNTIME ERROR STATUS 2

	Zero	One
BIT 0	Not used	Not used
BIT 1	No "Repeat Loop End Invalid" error	"Repeat Loop End" error (no Repeat Loop Start command)
BIT 2	No "Repeat Loop Incomplete" error	Repeat loop incomplete (no repeat loop end command)
BIT 3	Eight nested repeat loops not exceeded	Eight nested repeat loops exceeded
BIT 4	No "Return from Subroutine Invalid" error	"Return from Subroutine" invalid (no GoSub command)
BIT 5	No "Incomplete subroutine" error	Incomplete subroutine (no Sub Return command)
BIT 6	Eight nested subroutines not exceeded	Eight nested subroutines exceeded
BIT 7	No "Missing Label" error	Missing label

BYTE 5 COMMUNICATION STATUS 1

	Zero	One
BIT 0	Not used	Not used
BIT 1	Not used	Not used
BIT 2	Not used	Not used
BIT 3	Not used	Not used
BIT 4	Not used	Not used
BIT 5	No serial poll initiated	Serial poll initiated by sending Q. Wait
BIT 6	Not in SRQ mode	In SRQ mode
BIT 7	Not in Hold mode	In Hold mode

BYTE 6 COMMUNICATION STATUS 2

	Zero	One
BIT 0	Not used	Not used
BIT 1	Not used	Not used
BIT 2	In Hold mode, but no Trigger command received	In Hold mode, and Trigger command received
BIT 3	Unidex 1 "receive buffer" not full	Unidex 1 "receive buffer" full
BIT 4	X on received during transmit	X off received during transmit
BIT 5	Not in program download mode	In program download mode
BIT 6	Status bytes printed in binary format	Status bytes printed in Hex-ASCII format
BIT 7	Not used	

BYTE 7 AXIS MOTION STATUS

	Zero	One
BIT 0	X axis not moving	X axis moving
BIT 1 to BIT 7	Not used	Not used

BYTE 8 FREE RUN MODE STATUS

	Zero	One
BIT 0	X axis not in free run mode	X axis in free run mode
BIT 1 to BIT 7	Not used	Not used

BYTE 9 I/O STATUS

	Zero	One
BIT 0	Input 1 is a 0 (I1)	Input 1 is a 1 (I1)
BIT 1	Input 2 is a 0 (I2)	Input 2 is a 1 (I2)
BIT 2	Input 3 is a 0 (I3)	Input 3 is a 1 (I3)
BIT 3	Input 4 is a 0 (I4)	Input 4 is a 1 (I4)
BIT 4	Output 1 is a 0 (O1)	Output 1 is a 1 (O1)
BIT 5	Output 2 is a 0 (O2)	Output 2 is a 1 (O2)
BIT 6	Output 3 is a 0 (O3)	Output 3 is a 1 (O3)
BIT 7	Output 4 is a 0 (O4)	Output 4 is a 1 (O4)

SECTION 5-7 SYSTEM COMMANDS

System commands interact with Unidex 1 as a device and perform operations such as resetting Unidex 1, printing a program, printing position values, running a program, downloading a program, transferring status byte information from Unidex 1, etc. Commands requiring Unidex 1 to send back information will be ignored if more than one Unidex 1 is active. Each system command establishes a mode of operation once it is received by Unidex 1. Each system command must be entered as an **upper case letter.**

A. GETTING UNIDEX 1'S ATTENTION

The system command required from the host device to get Unidex 1's attention is two consecutive >, the two-digit device number and <CR>. For example:

>>01 <CR>

B. AUTO RUN MODE

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

To run a program in the Auto Run Mode, send **A**, the program number "nn", and <CR>. For example:

A 10 <CR>

If in Service Request mode (see section 5-5), once the program has been executed, Unidex 1 will send the Service Request character and wait for a serial poll. After the serial poll, you may execute the same program again by sending another <CR>. To run a different program, send **A nn <CR>** again.

C. BLOCK RUN MODE

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

B 10 <CR>

If in the Service Request mode, Unidex 1 will send the SRQ character after each block has been executed. If this is the case, the host must serial poll Unidex 1 after the execution of each block.

After the execution of the first command block and the serial poll, send a <CR> to execute the next block. Bit 1 of the status byte (section 5-5) may be checked to detect completion of the program.

The print status (PS) and print position (PX) commands can be used in the Block Run Mode after the block had been executed. To quit Block Run Mode without going through the entire program, use <CTRL>A (hex 01).

NOTE: During Interrupt operations, the interrupts will cause some blocks to terminate or be exited while motion is still in progress. An example of such a block is **HX***.

D. REMOTE RESET

Sending the command **C** followed by **< CR >** resets Unidex 1. This will take it back to power up conditions. For example:

C <CR>

This will stop motion and program execution in all active Unidex 1s. All units that have responded to this reset will need to be reactivated before they will accept a command again.

E. DISABLING REMOTE MODE

The system command **D <CR>** will disable the Remote Mode and return control to the host. (The position registers are updated with the absolute position values before returning control.)

When the Remote Mode is active, Unidex 1 will only recognize the commands "D", "Print X", "Print Status" and the Serial Poll command. All other system commands will be ignored.

F. DOWNLOADING A PROGRAM FROM HOST

The **E** command, followed by a program number ("nn") and an end-of-block character (* or /), will put Unidex 1 into the edit mode and set it up to enter the program commands into program "nn" in the user memory. If an existing program with the same number already resides in Unidex 1, it will be deleted automatically when the new program "nn" is downloaded. At the end of the program, a **%** must be sent to terminate the editor, and return to the system command level.

G. DELETING A PROGRAM

In order to delete a program from the Unidex 1 user memory, send the command **E**, followed by the character **$**, the program number "nn" and an end-of-block character, either * or /. The following example would erase program 10:

E $ 10 / (or *)

H. DELETING ALL PROGRAMS FROM USER MEMORY

In order to delete **all** programs from the Unidex 1 user memory, send the command **E**, followed by the character "$", two zeros ("00") and an end-of-block character, either * or /. Sending **E $ 00 /** would erase all programs.

I. BLOCK NUMBERING

If you want programs to be printed with block numbers, send **F <CR>**. Block numbering may make editing the program easier.

After this command is sent to Unidex 1, any program printed will contain block numbers.

J. BLOCK NUMBERING CANCEL

In order to cancel block numbering when a program is printed, as established in the above subsection, send the command **G** along with <CR>. After sending **G <CR>** to Unidex 1, programs will be printed without block numbering.

System commands F and G do not change the system set up feature stored in the battery backed memory (see section 5-2).

K. HOLD

The command to "hold" the execution of a command string or an entire program is established by the **H** command and a <CR>. Sending **H <CR>** will cause Unidex 1 to suspend execution of any Immediate, Auto or Block commands which may follow it. This is useful when synchronizing axis motion to some other action. Unidex 1 will only execute the commands when it receives a "T" (for trigger) command. For example:

```
H <CR>
A 20 <CR>          Program #20 held (not executed)
T <CR>             Program #20 triggered (executed)
```

L. IMMEDIATE MODE

The **I** command, followed by motion program commands, an end-of-block character (* or /) and a <CR>, allows a motion command to be executed immediately instead of being entered as a motion program. (For an explanation of the motion program commands, see section 5-8. A summarization of these commands is given earlier in this chapter.) Each block of immediate commands must begin with an I. For example:

I X F10000 D20000 * <CR>

The above immediate command will send the X axis 20000 steps at a feedrate of 10000 steps per second (or whatever units might be set in the system). If in SRQ mode, Unidex 1 will send a Service Request and wait for a serial poll after the command is executed. After being polled, Unidex 1 is ready to execute another block of commands.

M. SERVICE REQUEST SET UP

In order to establish the Service Request mode, send the **J** command, followed by <CR>. After the SRQ mode has been established via the J command, Unidex 1 will send the SRQ signal % **<CR>** under conditions described in section 5-5. It will then wait until it receives a serial poll from the host device before executing any further commands. For more detailed information on SRQ, see section 5-5.

The SRQ character may be programmed for something other than % by sending the character after the J command. For example:

J ! <CR>

This command sets the SRQ character to "!". Keep in mind a service request will only be sent by Unidex 1 if it is the only active unit. Otherwise, it will wait until this is the case.

N. SERVICE REQUEST CANCEL

In order to cancel the Service Request (SRQ) mode set up by the J command (see above), send a **K <CR>**. (This is the default status.)

O. STATUS BYTE IN BINARY FORMAT

Sending the command **M <CR>** establishes the format of the status bytes as binary upon transmission. Transmission of the status bytes in binary format is the default status.

P. STATUS BYTES IN HEX-ASCII FORMAT

To establish the status bytes in the Hex-ASCII format upon transmission, send **N <CR>**. In this mode, each status byte will be sent as two bytes. Each byte will be the equivalent ASCII character of each hexadecimal byte.

Q. HOLD MODE CANCEL

To cancel Hold mode, send the command "O" along with <CR>. After sending **O <CR>**, programs will no longer require a trigger (T command).

R. PRINT AXIS POSITION

1. Print X Axis Position (PX)

In order to print the X axis position register, send **PX <CR>**. The axis position is sent in the following format:

(<Space> or <->) <10 Digits> <CR>

Positive Negative

S. PRINTING DIRECTORY LISTING

To get a listing of the programs in the Unidex 1 directory, send **PD <CR>**. Bytes of memory remaining in Unidex 1 will be listed too.

T. PRINTING A PROGRAM

To have one program printed, send the command **P**, the program number "nn", and <CR>. For example:

P10 <CR>

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

U. PRINTING ALL PROGRAMS

To have all programs printed, send the command "**P**", two zeros (00) and <CR>. Sending **P00 <CR>** will cause all programs in memory to be printed.

V. PRINTING STATUS BYTES

To have the status bytes listed in section 5-6 printed, send **PS <CR>**.

W. PRINTING VERSION OF SOFTWARE

To print which version of software you have, send the command **PF**.

If using a computer interface, the reply will be sent in the following format:

SOFTWARE UDI_xx
(where xx represents the software level)

X. QUERY (SERIAL POLL)

The host device may serial poll (Query) Unidex 1 by sending **Q <CR>**. In response to a query, Unidex 1 returns its status (see section 5-5).

Y. REMOTE MODE

The system command **R <CR>** will enable Unidex 1 to be driven via the auxiliary clock and direction controls. The host controller may then signal an external device to take control of Unidex 1. The PX and PS commands may be used while in the remote mode.

Unidex 1 keeps track of the axis position during external control.

Z. TRIGGER

To execute the program that is suspended with a Hold command (H), send **T <CR>**. In multiple axis operation, this will allow the simultaneous program execution of multiple Unidex 1s.

AA. RESETTING UNIDEX 1

To Reset Unidex 1, you may send the hexadecimal number 7F or FF. Either is the ASCII code for <DEL>. You may also send the **<CTRL> D** character (hex code 04) to reset Unidex 1. Any of these resets will take Unidex 1 back to power up conditions and will reset all of the units that are connected in daisy chain, even if they are not in the active mode.

The soft reset will reset only the active Unidex 1s (in-progress motion) and is initiated by sending the **<CTRL> A** character (hex code 01). This reset will stop motion (decel to stop) and program execution in all active Unidex 1s. All units that have responded to this reset will still be active and ready to accept a command.

The command <CTRL>B (hex code 02) will deactivate all active Unidex 1s on the daisy chain line.

BB. XON/XOFF TRANSMISSION PROTOCOL

The XON/XOFF protocol regulates the transfer of information between two devices. This is required for the reliable transfer of information. The XOFF character <CTRL>S (hex 13) instructs a device to halt transmission. The XON character <CTRL>Q (hex 11) instructs a device to resume transmission.

Unidex 1 responds to XON/XOFF protocol at all times when it is active and operating from within the computer interface mode. This implies that Unidex 1 will suspend transmission to the user upon receipt of the XOFF character. Transmission will resume upon receipt of the XON character. Unidex 1 may also send XON/XOFF characters to request transmission be suspended until it is ready to accept more commands.

SECTION 5-8 MOTION PROGRAM COMMANDS

These are the user program blocks in a motion program that Unidex 1 executes when running a program in the auto run or block run mode. These commands are valid only if entered in the immediate or edit mode.

A. END OF BLOCK

An end-of-block terminates a block of a program. It may be one of two characters, * or /. For example:

X F10000 D150000 * (or /)

B. AXIS MOTION COMMANDS

The axis must be specified by an axis command (X). (The X axis designation is used to maintain consistency with the Unidex 11 Series of controllers.)

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

The distance which it is to travel (or the position it is to reach, if in the absolute mode) is specified with a distance command (D).

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

X F10000 D-150000 *

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

NOTE: When programming via Computer Interface communication (Port A), the feedrate need only be entered with the first index block. After that, the feedrate need be entered only if it is to be changed.

1. Axis Free-Run

The axis is commanded to free-run by the command R and a "+" or "-" sign to signify CW (+) or CCW (-). For example:

X F10000 R+ *

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

C. DWELL

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

DW 10 *
DW 1.0 *

Either of the above commands will cause a 1 second dwell within your program.

D. HOME

Send the axis Home with command H followed by the axis and an end-of-block character. The command **H X *** will send the axis home.

E. OUTPUT STATUS

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

OT 10XX *

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

F. INPUT STATUS

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

IT X001 *

The above command states that the program should wait until I2 is a 0, I3 is a 0 and I4 is a 1. The state of input I1 does not matter since it is programmed as a "don't care". (See section 8-6 for input signal specifications.)

G. OUT/STOP STATE

To put out values to the outputs when the program is stopped via a feedhold, enter the command "OS" followed by the desired values and an end-of-block character. For example:

OS 0011 *

When the program is stopped, a zero will be output to O1 and O2, and a 1 will be output to O3 and O4. (See section 8-8 for output signal specifications.)

H. OUT/RUN

To output values to the outputs when the program is allowed to run again, release the feedhold key (after it has been pressed), and program "OR", followed by the desired values and an end-of-block. For example:

OR XXXX *

In the above example, when the program run is allowed to continue, the outputs will remain unchanged (all have been programmed as "don't care").

I. REPEAT LOOP START

The command to start a repeat loop in your program and the number of times the loop executes is established with a "RS" command, followed by the number of times to repeat and an end-of- block (9999 maximum). For example:

RS 8 *

The above example marks the beginning of the repeat loop and says it will repeat 8 times. A maximum of eight nested loop commands are permitted. An unlimited number of un-nested loop commands are permitted.

J. REPEAT LOOP END

To mark the end of the repeat loop (started as discussed in the above subsection), program command **RE ***.

Repeat loops may be nested eight levels deep.

K. CONDITIONAL REPEAT LOOP END

To end the repeat loop based on input conditions, prior to completing the specified number of loops, program **RC** followed by the required input state and an end-of-block. For example:

RC 10XX *

The above example states that the repeat loop will end when the inputs are as follows: I1 is a 1, I2 is a 0. I3 and I4 are "don't cares" and have no control over the program block.

L. STARTING FREE RUN AFTER STOP FREE RUN

After a free run has been stopped (discussed in the next subsection), programming an **RX *** will start the axis again.

M. STOP AXIS

To stop the axis, program **SX ***.

N. REPEAT PROGRAM

To repeat the entire program from the start, enter command "RP" and an end-of-block. Remember, any commands following **RP *** within your program will not be executed.

O. LOAD POSITION REGISTER

You may load the axis position register with an **LX** command, followed by a distance and an end-of-block. The value, which is in system units, may be a positive or negative number or may be a zero. For example:

LX 0 *

In the above example, the position register is loaded with zeros. This command may be used to establish an absolute reference position. The axis may be moved to this reference position by programming, in the absolute mode, an Index block with distance values equal to the reference position.

P. INCREMENTAL MODE (IN) / ABSOLUTE MODE

In the incremental mode, (established with a **IN ***), a distance command tells Unidex 1 how much further to move the axis. For example:

X F10000 D1000 *

The above command will move the axis 1000 steps in the positive direction each time it is executed.

In the absolute mode (established by entering **AB ***) a distance command is an absolute position. For example:

X F10000 D1000 *

When in the absolute mode, the above example tells Unidex 1 to send the axis to the position 1000. Once there, the re-execution of the above command will not move the axis any further since it is already at the position commanded.

Q. CORNER ROUNDING

In corner rounding mode, Unidex 1 does not wait for the in-position before beginning the next move. For corner rounding, program:

CR *

R. NON-CORNER ROUNDING MODE

In non-corner rounding mode, Unidex 1 waits for the in-position before beginning the next move. For non-corner rounding, program:

NC *

S. LABEL

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

LB 55 *

The example label command tells Unidex 1 to direct the program execution to the block labeled 55 when a GoSub or GoTo is encountered.

T. GOTO

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

GT 20 *

The above command tells Unidex 1 to continue program execution at label 20.

U. GOSUB

The command "GS", followed by the block label number and an end-of-block, tells Unidex 1 to execute a subroutine at label #nn. For example:

GS 15 *

The subroutine to be executed is located at label #15. A maximum of 8 nested subroutines are permitted. An unlimited number of unnested subroutines are permitted.

V. SUBROUTINE RETURN

This command causes Unidex 1 to return from the subroutine execution to the program block immediately after the **GS nn** block that called the subroutine. Every subroutine should end with an **SR ***.

Subroutines may be nested to 8 levels deep.

W. PROGRAM STOP

Program stop marks the place in the program at which program execution ends. Subroutines should be placed after the **PS *** block.

X. CONDITIONAL GOTO

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

CT 22 I 10X0 *

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

Y. CONDITIONAL GOSUB

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

CS 33 I 110X

The above command tells Unidex 1 to go to the subroutine labeled "33" when the value of I1 is 1, I2 is 1, and I3 is 0. If these input conditions do not exist, continue with the next program block. If conditions are met, the subroutine will execute, then return to execute the next instruction.

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

Z. REMOTE MODE

The command to send Unidex 1 into the remote mode if input conditions are met is :

RI 0X10 *

The unit will go into remote when the input conditions match those programmed, and will remain there as long as the input conditions stay the same.

AA. ACCELERATION/DECELERATION RAMP TIME

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

AD 250 *

In the above example, the acceleration ramp time as well as the deceleration ramp time will be 250 milliseconds. The maximum value is 9999.

(For more information on Accel/Decel, see chapter 6.)

BB. BRANCH ON MARKER

The MT command will cause the program to go to a specified block if the marker is present (1) or continue with the next block if the marker is not present (0).

MT 13 *

The above command tells the Unidex 1 to go to the program block labeled 13 if the marker is present (1). If the marker is not present (0), the program will continue with the next block.

NOTE: Due to possible motor jitter, this command may not be usable.

CC. GOSUB ON MARKER

The MS command will cause the program to go to the subroutine specified if the marker is present (1), or continue with the next block if the marker is not present. The subroutine will return program execution to the next instruction upon completion (with Subroutine Return SR.)

MS 17 *

The above command tells the Unidex 1 to go to the subroutine labeled 17 if the marker is present (1).

NOTE: Due to possible motor jitter, this command may not be usable.

DD. ENABLE INTERRUPT

The EI command will enable interrupt capability. Either a GoTo (BI) or GoSub (SI) on Interrupt must have previously been programmed. Interrupts are monitored as often as possible. However, certain program blocks require longer periods of time in which interrupts must be present (Example: Move blocks). Interrupt conditions present and stable for over 60 milliseconds, should prevent missing interrupts.

The command **EI *** would enable the interrupt command and is valid until a disable interrupt block is executed or an interrupt occurs.

EE. DISABLE INTERRUPT

The DI command will disable interrupts, if already enabled. The **DI*** command disables the interrupt.

FF. BRANCH ON INTERRUPT

The branch on interrupt command sets the conditions for an interrupt. The interrupt branch block is specified by the label portion and the conditions for interrupt are specified by the I portion (same as Conditional GoTo). An Enable Interrupt command is required to enable the interrupt.

BI 31 I XXX1 *

When the interrupt is enabled, the inputs will be monitored and compared with the I conditions. If the I conditions are met, the program will go to block label 31. After an interrupt occurs, the interrupt is disabled and can be enabled using the "Enable Interrupt" command. Special action is taken if an interrupt occurs while executing any of the following blocks.

Index, Home:	Interrupt will branch to a block with motion still in operation. However, if any type of axis move is attempted before the axis has completed its move, the present move will be stopped and the new move will be executed immediately.
Dwell:	Dwell is terminated
Wait on Input:	Input is terminated
Remote on Input:	Remote is terminated

GG. GOSUB ON INTERRUPT

The GoSub on Interrupt command (SI) sets the conditions for an interrupt. For example:

SI 24 I X1X0 *

The interrupt subroutine block is specified by the label portion and the interrupt conditions by the I portion (same as conditional GoSub). An Enable Interrupt command is required to enable the interrupt. Once interrupted, the interrupt cannot be enabled again until the return from Interrupt (Return from Subroutine) occurs. Since the interrupts are disabled when an interrupt occurs, an "Enable Interrupt" is required to re-enable the interrupt. This Enable Interrupt may be placed inside the Interrupt Subroutine itself. However, the interrupts will not be enabled until the Return from Subroutine is complete.

When the interrupt is enabled, the inputs will be monitored and compared with the I conditions. If the I conditions are met, the program will go to subroutine label (label 24 in the previous example). Special action is taken for interrupts that occur during the execution of the following blocks.

Index, Home:	Interrupt will branch to block with motion still in operation. Note that moves (index or home moves) initiated in the interrupt subroutine will cause previously initiated indexes and homes to be aborted if still in progress. However, if the interrupt subroutine contains no motion commands, the interrupted move (index or home) will continue while the interrupt is serviced. The Return from Subroutine will occur when the move is completed.
Dwell:	Dwell is terminated.
Wait on Input:	Wait on Input is terminated. Program will return to next block.
Remote on Input:	Remote is terminated. Program will return to next block.

HH. SET POSITIVE LIMIT

To load the positive limit absolute position register, enter command **LP** followed by the (CW) limit position and an end-of-block. The value is in system steps (maximum 10 digits) and may be a positive, negative or zero number. For example:

LP -10000 *

In the above example, the (CW) limit position is loaded with the position value -10000. When this limit is enabled by either an EP (Enable Position Limit) or an EL (Enable Limits) command, moves which would or could potentially result in a **more positive** position than the limit value specified above, will not be executed. Values for the LP command can be positive (+) or minus (-). If a limit is encountered, the program will be terminated. The Unidex 1 checks for a potential limit overrun *BEFORE* actually making a given move. This command applies only to auto (or block) executed programs.

II. SET MINUS LIMIT

To load the negative limit absolute position register, enter the command **LM** followed by the (CCW) limit position and an end-of-block. The value is in system steps (maximum 10 digits) and may be a positive, negative or zero number. For example:

LM -50000 *

In the above example, the minus (CCW) limit position is loaded with the position value -50000. When this limit is enabled by either an **EM** (Enable Minus Limit) or an **EL** (Enable Limits) command, moves which would or could potentially result in a **more minus** position than the limit value specified above, will not be executed. If a limit is en-

countered, the program will be terminated. The Unidex 1 checks for a potential limit overrun *BEFORE* actually making a given move. This command applies only to auto (or block) executed programs.

JJ. ENABLE POSITIVE LIMIT

To enable the positive (CW) limit, enter the command **EP**, followed by an End Of Block. For example:

EP *

In the above example, the set positive (CW) limit will be enabled. This block will check for a present or possible positive limit condition. If any limit condition exists, the program will be terminated and the motion will be stopped. All moves following this block will be checked before the moves are executed. Positive free run and home commands **will not be started** and the program will be terminated. Moves which will exceed the positive limit will cause program termination and will not be executed. Either a **DP** (disable positive limit) or a **DL** (disable limits) command will disable the positive limit checks.

KK. ENABLE MINUS LIMIT

To enable the minus (CCW) limit, enter the command **EM**, followed by an end-of-block. For example:

EM *

In the above example, the previously set minus (CCW) limit will be enabled. This block will check for a present or potential minus limit

condition. If any limit condition exists, the program will be terminated and the motion will be stopped. All moves following this block will be checked before the moves are executed. Minus free run and home commands will not be started and will cause the program to terminate. Any move which will exceed the minus limit **will not be started** and will cause the program to terminate. Either a **DM** (disable minus limit) or a **DL** (disable limits) command will disable the minus limit checks.

LL. ENABLE LIMITS

To enable both the positive (CW) and minus (CCW) limits, enter the command **EL** followed by an end-of-block. For example:

EL *

In the above example, the previously set positive and minus limits will be enabled. This block will check for a present or potential limit condition. If any limit condition exists, the program will be terminated and the motion stopped. All moves following this block will be checked for a limit condition before the moves are executed. All free run and home commands **will not be started** and will result in the termination of the program. Moves which will exceed the positive or minus limits will not be executed and will cause the program to be terminated. A **DL** (disable limit) command will disable all limit checks. A **DP** (disable positive limit) or **DM** (disable minus limit) command can be used to disable one or both of the limits.

MM. DISABLE POSITIVE LIMITS

To disable the positive limit check, enter the command **DP** followed by an end-of-block. For example:

DP *

In the above example, the positive limit check will be disabled. The minus limit enable will not be effected.

NN. DISABLE MINUS LIMITS

To disable the minus limit check, enter the command **DM**, followed by an end-of-block. For example:

DM *

In the above example, the minus limit check will be disabled. The positive limit enable will not be effected.

OO. DISABLE LIMITS

To disable the positive and minus limit check, enter the command **DL** followed by an end-of-block. For example:

DL*

In the above example, both the positive and minus limit checks will be disabled.

PP. END EDIT

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

When downloading a program, this character takes Unidex 1 out of the edit mode and back to the system command mode after the download.

QQ. PROGRAM COMMENTS

Comments may be placed within a program by preceding them with a semicolon (;). Except for the characters listed below, anything following the semicolon will be ignored and will not be stored in Unidex 1's memory. The command field is terminated by a <CR> or a <LF>.

The characters that may not be used in the comments are:

1. **#, < or >**
2. Control codes **<CTRL>A, <CTRL>B, <CTRL>D**
3. Hex code **7F** or **FF**.

SECTION 5-9 COMPUTER INTERFACE COMMANDS

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

Example Immediate Commands

```
>><device number> <CR>      ; Interface active
I H X * <CR>                ; Send home axis
I X F10000 D10000 * <CR>    ; Axis move
I X F100 D1000 * <CR>       ; Axis move
```

Example Motion Program

```
>><device number> <CR>      ; Interface active
E 01 *                      ; Select program 1
H X *                       ; Send X axis home
X F10000 D10000 *           ; X axis move
DW .2 *                     ; Dwell for 2/10 second
%                           ; End edit mode
```

Send **F <CR>** to set Unidex 1 to the block number printing mode. The command **P01 <CR>** will now cause program #1 to be printed with block numbers.

Command **G <CR>** will cancel the numbering command.

The command to run program #1 in the block mode is **B01 <CR>**. For each successive block send another **<CR>**.

To run program #1 in the auto run mode, send **A01 <CR>**.

To delete program #1, send **E $ 01 ***. (Check your directory with a **PD** command to verify that program #1 has been deleted.)

The position register may be read by sending **PX <CR>.**

Send **J <CR>** to put Unidex 1 in the Service Request mode. (You may enter a character of your own choosing or use the default Service Request character, %.) When this character is sent by Unidex 1, acknowledge it with a **Q <CR>** (query) before continuing. If running the program in the auto mode, the Service Request will come after the program execution. If running a program block by block, it will follow each block.

Send command **K <CR>** to cancel the Service Request mode.

CHAPTER 6: PROGRAMMABLE ACCEL/DECEL

Programmable Accel/Decel increases the performance of the motion control system by providing the ability to achieve higher motion speeds. The ramp time (the time to attain programmed feedrate) is programmable from 50 milliseconds to 6500 milliseconds. The start/stop feedrate is also programmable. (The start/stop setting is used as a default speed for very short moves where accel/decel cannot be used.)

Once the user has set (setup mode) the desired ramp time and the start/stop feedrate for acceleration/deceleration, the values are stored in the battery backed memory. The system default values for these parameters are:

Accel/Decel Ramp Time:	**250 Milliseconds**
Start/Stop Feedrate:	**500 Steps/Second**

These parameters are modal, which is to say they stay in effect in both the (auto or block) program run and immediate modes of Unidex 1. However, Unidex 1 can be programmed (in the auto or block mode) to change the accel/decel ramp time during program execution. (Start/stop feedrate can only be changed in the Set Up mode, see chapter 3.) If the accel/decel ramp time is changed during program control, the last value set becomes the new modal accel/decel parameter.

SECTION 6-1 MANUAL PROGRAMMING OF RAMP TIME

Acceleration and deceleration parameters are programmed in the Set up mode (chapter 3). (See chapter 4 for the hand held terminal operation and chapter 5 for the Computer Interface operation.)

The required ramp time may now be programmed. This value may be from 50 to 6500. Any value below 50 will be entered in the system as zero and this will turn acceleration and deceleration OFF.

User memory (battery backed) will be updated as a result of manually programming the ramp time.

SECTION 6-2 PROGRAMMING START/STOP FEEDRATE

The start/stop speed is also programmed in the set up mode.

Feedrate values from 1 to 250000 units/sec may be entered here (in this case, 0 to 250000 steps/sec). Start/stop speed is the speed to which the axis defaults when a programmed move is too short to implement acceleration and deceleration (see section 6-5). If the time for the move is less than 16.384 milliseconds, acceleration/ deceleration is temporarily turned off and the move is executed at the start/stop feedrate.

SECTION 6-3 RAMP TIME PROGRAMMING IN PROGRAM

The motion command for entering the ramp time when downloading a program is:

AD nnnn *

where "nnnn" is the ramp time in milliseconds. To turn off acceleration/deceleration, program 0 milliseconds.

The ramp time may be programmed as many times as required within a program. At the end of program execution, the value of the last programmed ramp time will be the new value to remain as the last programmed value. If programmed manually in setup before running a program, this value will be effective at the start of the program. The last ramp-time block executed from a program becomes modal to the system, but is **NOT** retained in the battery-backed user memory after power down.

SECTION 6-4 WHEN ACCEL/DECEL ISN'T IMPLEMENTED

Accel/decel is not implemented under the following conditions:

1. Ramp time programmed is less than 50 mSec.
2. Feedrate for the move is less than 16 steps/second.
3. Total time for the move is less than 16.384 mSecs.

SECTION 6-5 ACCEL/DECEL IN OPERATION

Acceleration and deceleration velocity profiles in Unidex 1 are achieved by updating the clock rate from the indexer at fixed intervals, pre-computed from the programmed ramp time. The minimum clock-rate-update interval is 1 mSec. and the maximum number of updates is 250. If the ramp time programmed is 250 mSecs., the controller increments the clock rate from 0 to the programmed feedrate in 250 steps of a 1 mSec. interval. For ramp times less than 250 mSecs. but greater than 125 mSecs., the number of updates is 125 and the interval is accordingly computed. For ramp times between 125 and 50 mSecs., the number of updates is 50. The following table illustrates this:

Ramp Time	Number of Updates	Update Intervals (mS)
5000	250	20.0
400	250	1.6
200	125	1.6
80	50	1.6
30	*No Accel/Decel*	

In accel/decel mode, the clockrate is updated linearly. In the following description, N stands for the number of updates computed from the programmed ramp time as explained above, Fn represents feedrate at interval number n, and Fp represents programmed feedrate.

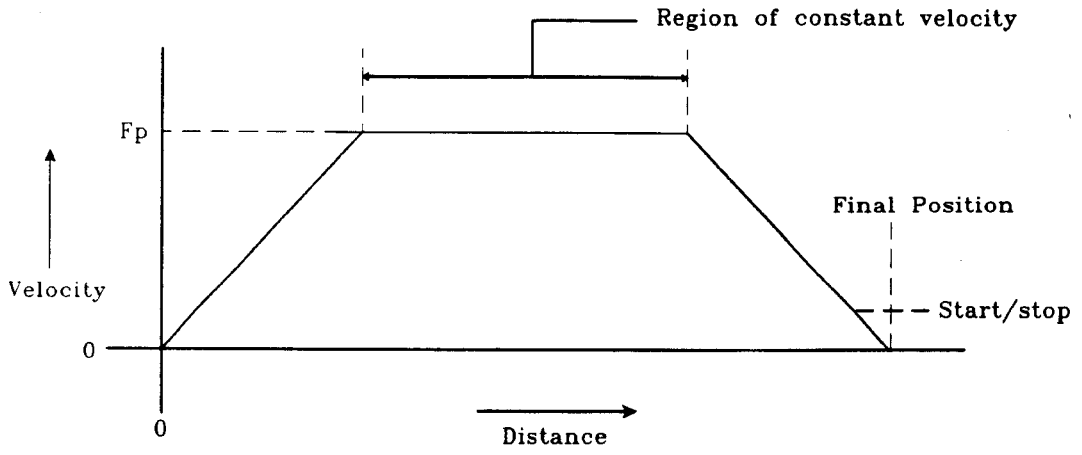
The feedrate as a function of an update interval number during acceleration is shown below.

$$F_n = (F_p/N) * n \qquad n < N$$

$$F_n = F_p \qquad n >= N \qquad 6.1$$

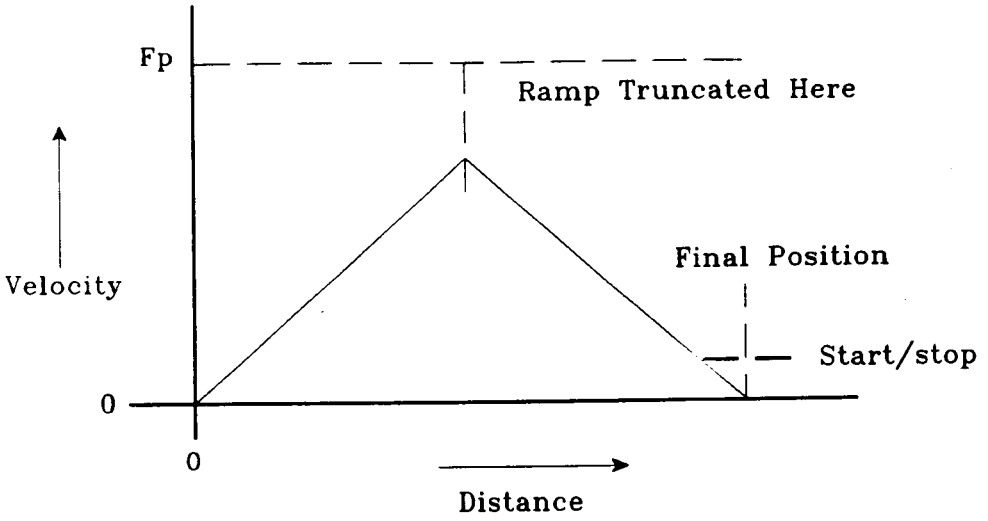
In the accel/decel mode, when the programmed move is longer than the ramp time, a trapezoidal velocity profile is achieved. When the move is shorter, (but greater than 16.384 mSec.), a triangular velocity profile results and the axis does not attain the programmed feedrate.

Figure 6-1 and 6-2 illustrate some examples of linear ramping in both full and truncated profiling modes.



A) Linear Ramping

Figure 6-1: Full Ramp Profiles For Linear Ramping



A) Linear ramping

Figure 6-2: Truncated Ramp Profiles For Linear Ramping

SECTION 6-6 RAMPING LIMITATIONS

The system hardware imposes certain restrictions on the performance of the programmable accel/decel. The maximum axis feedrate with accel/decel is 250000 steps/sec. The period resolution of the clock pulses from the indexer is 4 microseconds. To compensate for this, the number of update intervals (ramp time) is modified. Example:

Programmed feedrate:	100000 Units/Sec.
Required system clock period:	10 μSec. (reciprocal of feedrate)
Number of updates (N):	250
Closest feedrate of which the system is capable:	8 μSec or 125000 Units/Sec.
Modified number of updates (N):	250 * 8/10 = 200
Modified ramp time:	Ramp Time * 8/10
Feedrate at end of Accel:	125000 * 8/10 = 100000 Units/Sec.

For the above example, the modified number of update intervals is 200.

NOTE: When decelerating at the end of a move, the feedrate "levels off" at the programmed start-stop value in order to mask any nonlinearities of the system. If desired, the user may set up his system for optimum performance by changing the stop/start feedrate parameter.

SECTION 6-8 OTHER LIMITATIONS

Accel/decel control in Unidex 1 is implemented with individual hardware VCO (voltage control oscillator) circuitry. Since these VCOs are analog drivers, and operate asynchronously with respect to the indexing circuitry of Unidex 1, the actual programmed velocity when running in the accel/decel mode may be off as much as 5%.

Hardware and software limitations prevent Unidex 1 from providing the ideal ramp profiles shown in figures 6-1 and 6-2. In some cases the final position is reached before the velocity reaches the start/stop frequency, which results in overshoot of the motor. Also, some moves may run at start/stop velocities for longer lengths of time than others.

These limitations are not considered flaws in Unidex 1 performance, since the basic design criterion is one of point to point motion control.

CHAPTER 7: APPLICATION PROGRAM

This section provides a sample program (figure 7-2) to demonstrate some of the programming capabilities of Unidex 1. Included also are an application summary and a flow chart (figure 7-1).

See section 5-8 for a description of commands listed in figure 7-2.

SECTION 7-1 SAMPLE PROGRAM DESCRIPTION

Application

A manufacturer of precision widgets wants to automatically inspect them at the output point of the widget production line.

Solution

A single-axis Unidex 1 motion controller is used to control the motion of a linear stage to position the widgets for inspection and fault processing. The controller, through its discrete inputs and outputs, also controls the other devices used in the inspection process. The process progresses basically as follows:

a) Initially, the axis is sent home, all outputs are initialized and an initial reference, or starting point, is established.

b) Normal inspection operation commences when the presence of a widget is sensed. It is then moved to the inspection/fault processing position.

c) If, during positioning, the product sensor senses that the widget has been skewed, an indicator will be lit and the process will stop.

d) A subroutine (#50), is used to process the faulty widgets. One hundred tries are attempted, but if the fault is still present, the widget is ejected.

The flow chart and program code for the example are provided on the following pages. Note the power of the Unidex 1 programming language exhibited by its ability to branch on interrupt, conditionally branch and repeat loops.

Example Program Flow Chart

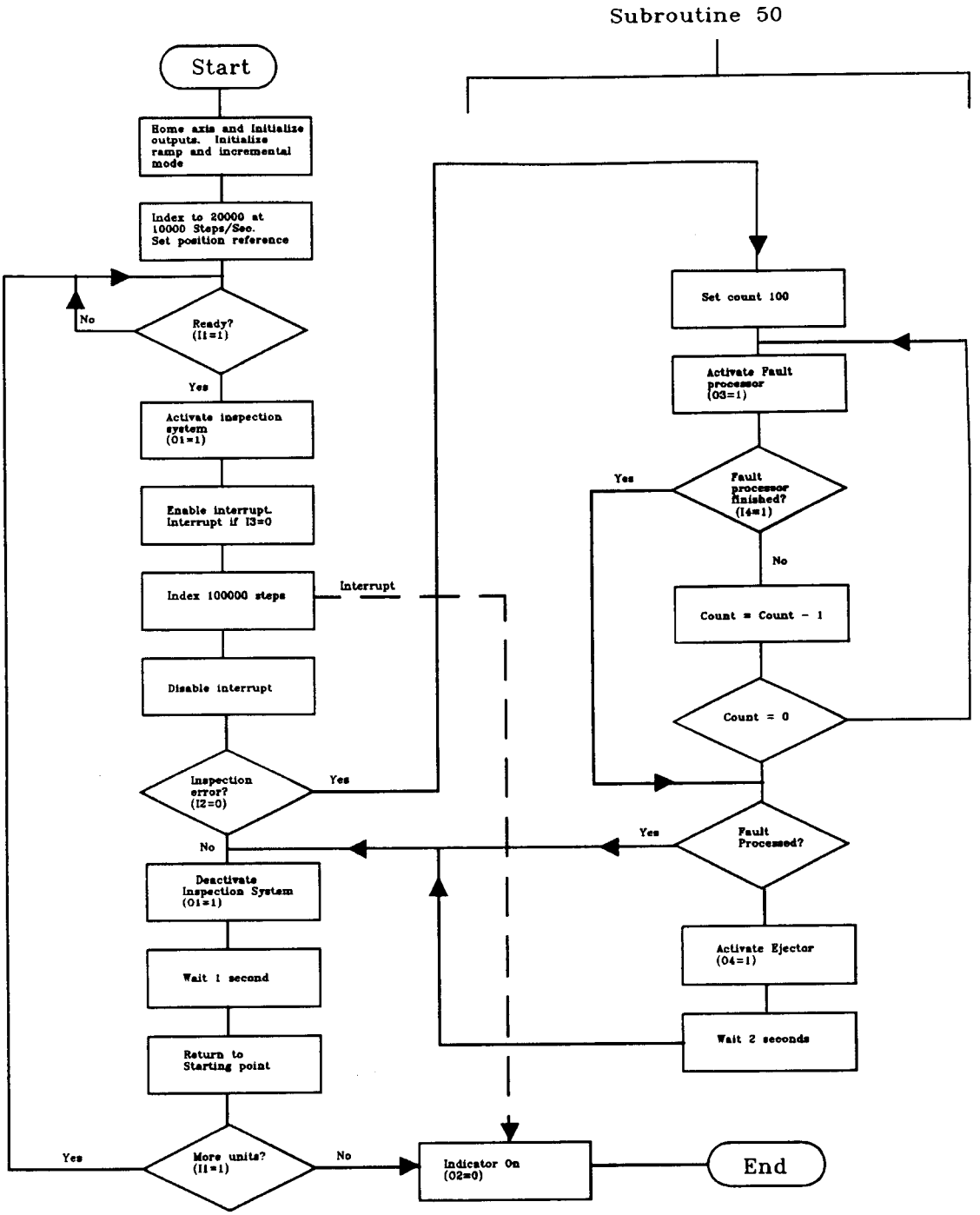


Figure 7-1: FLow Chart of Example Program Shown in Figure 7-2

Example Program
-UNIDEX 1 CODE-

```
HX *              ; Home axis
OT 1111 *         Initialize outputs to 1
AD 100 *          ; Accel/decel time = 100 mS
IN *              Incremental mode
XF10000 D20000    ; Home offset
LX 0 *            ; Set position to zero
LB 90 *           ; Label
IN *              ; Incremental mode
IT 1XXX *         ; Wait for input I1 = 1
OT 0XXX *         ; Output O1 = 0
EI *              ; Enable interrupt
BI 10 I XX0X *    ; Branch to label 10 if I3 = 0
X D 100000 *      ; Index 100,000 steps
DI *              ; Disable interrupt
CS 50 I X0XX *    ; If input I2 = 0, do subroutine 50
OT 1XXX *         ; Output O1 = 1
DW 10 *           ; 1 second wait
AB *              ; Absolute mode
X D 0 *           ; Index back to zero
CT 90 I 1XXX *    ; If input I1 = 1, repeat process
LB 10 *           ; Label 10
OT X0XX *         ; Else output O2 = 0
PS *              ; End of program

LB 50 *           ; Subroutine 50
RS 100 *          ; Repeat 100 times or until I4 = 1
OT XX0X *         ; Output O3 = 0
DW 01 *           ; Dwell 0.1 second
OT XX1X *         ; Output O3 = 1
RC XXX1 *         ; If input I4 = 1, quit loop
CT 60 I XXX1 *    ; If I4 = 1, go to 60
OT XXX0 *         ; Output O4 = 0
```

```
DW 05 *          ; Dwell 0.5 second
OT XXX1 *        ; Output O4 = 1
DW 20 *          ; Wait 2 seconds
LB 60 *          ; Label 60
SR *             ; Return from subroutine
```

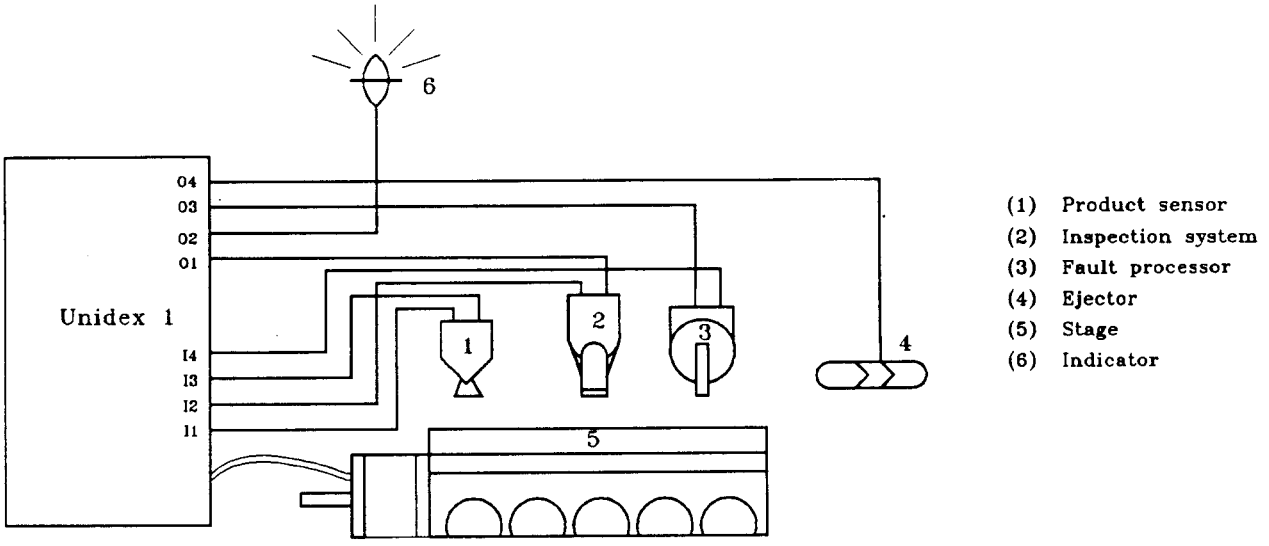


Figure 7-2: Example Program for Unidex 1

CHAPTER 8: SYSTEM LAYOUT/EXT. CONNECTIONS

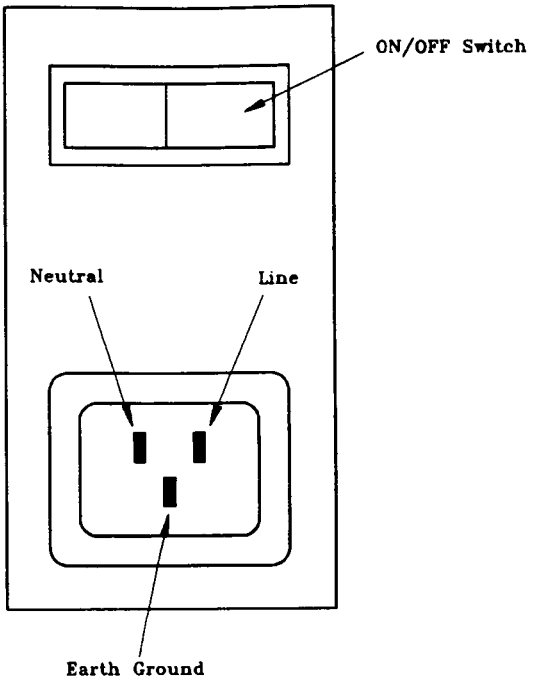
Unidex 1 DC Controller is supplied in three output voltage ranges: 40 VDC, 80 VDC, and 160 VDC, depending on the motor used (see table 1-1). The package configuration is the same for all three models.

SECTION 8-1 INPUT POWER CONNECTIONS

The Unidex 1 U1SA through U1SF packages (see table 1-1) can be factory wired for either a 115 VAC 60 Hz or 230 VAC 50/60 Hz input. Unidex 1 models U1SG, U1SH and U1SJ can only be supplied with 115 VAC 50/60 Hz input power connections. Figure 8-1 shows an outline of the connector layout for the Unidex 1 Family of Controllers.

CAUTION: Note specification on input power label located on the bottom of the chassis, below the input power connector, BEFORE APPLYING POWER. (See figure 8-1)

The outline of the input power receptacle is shown in the following illustration.



NOTE: Mating cable is supplied with unit.

Refer to Table 1-1 for current rating.

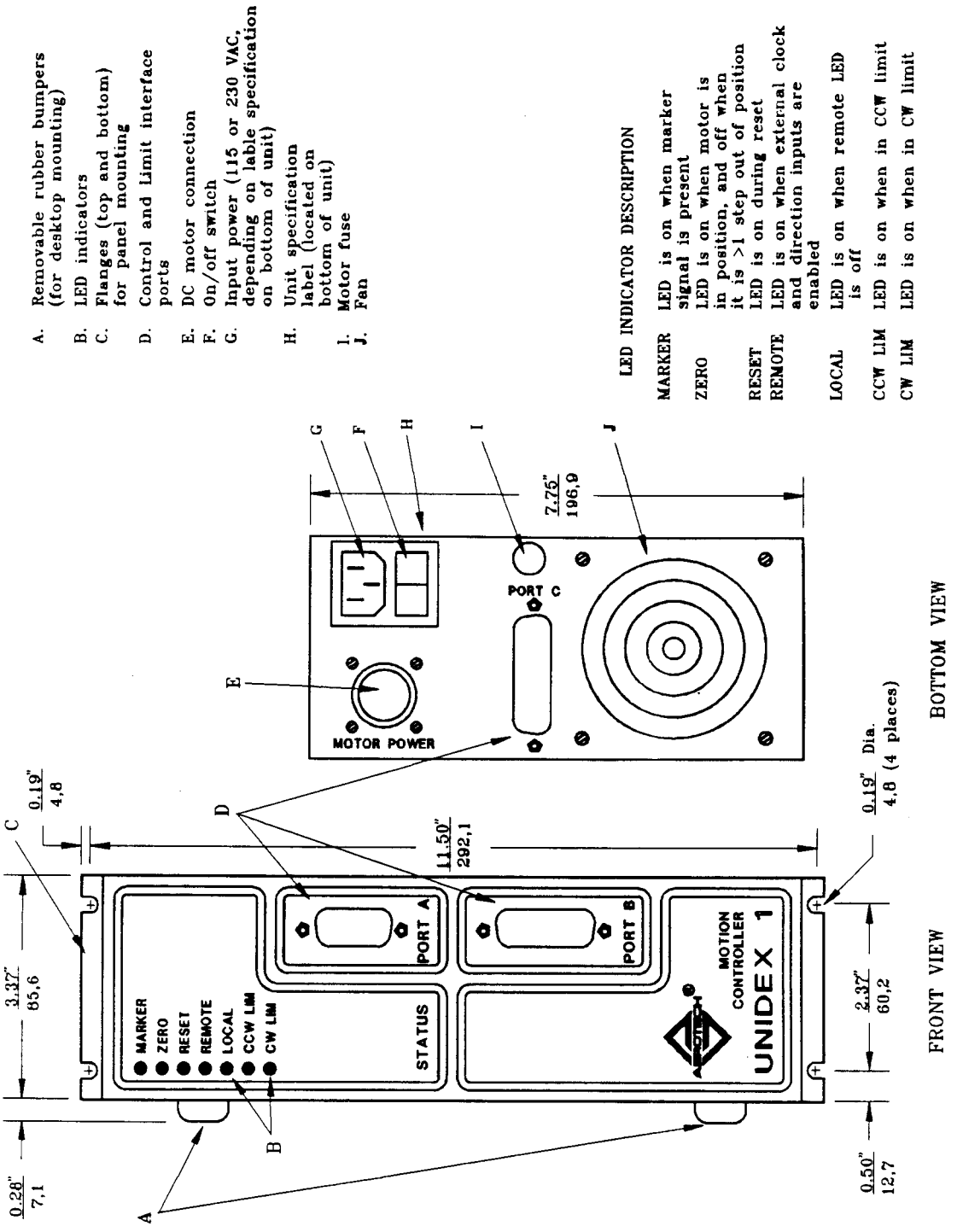


Figure 8-1: Unidex 1 Models U1SA through U1SJ Package Outline

SECTION 8-2 INPUT POWER CONNECTIONS

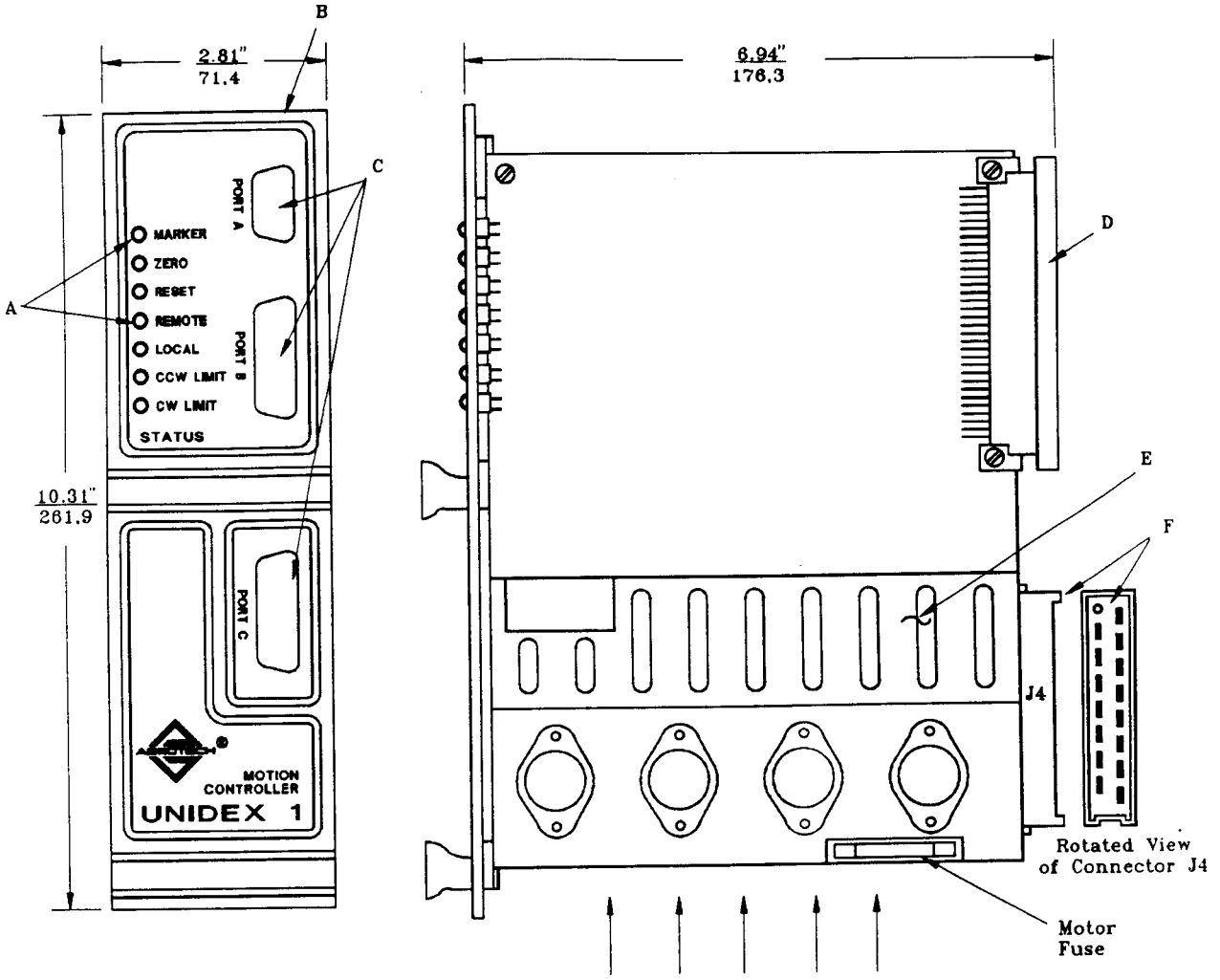
(Special 6U Eurocard Format)

The Unidex 1 DC 6U package can be field wired for DC Bus voltages of 0 through 80VDC for models U1SA through U1SF when these units are supplied without enclosures. Models U1SG through U1SJ can be supplied with bus voltages of 0 through 160VDC when these units are supplied without enclosures. Separate +5 and ±12 VDC control supplies are required for all versions.

Figure 8-2A shows an outline (as well as motor power connections) of the 6U version of Unidex 1 DC. AC power is connected to the power connector J3. A mating connector for J3 (shipped with the unit) is outlined in figure 8-2B.

WARNING: The special 6U Eurocard Unidex 1 DC configurations are intended for a totally enclosed cardrack mounting only. Lethal voltage is present on exposed portions of PC board when power is applied. Aerotech Does Not Assume Liability For Inadequately Housed Units.

Refer to table 1-1 for motor output current specification.



NOTE: User supplied fan cooling required here (30CFM minimum)

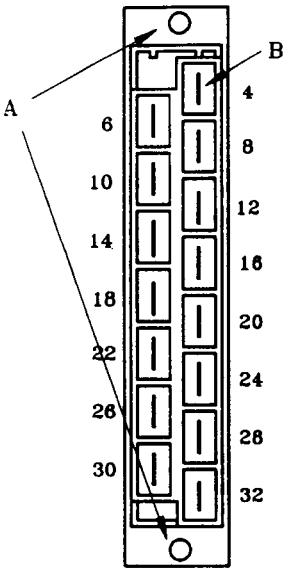
LED INDICATOR DESCRIPTION

MARKER	LED is on when marker signal is present
ZERO	LED is on when motor is in position
RESET	LED is on during reset
REMOTE	LED is on when external clock and direction inputs are enabled
LOCAL	LED is on when remote LED is off
CCW LIM	LED is on when in CCW limit
CW LIM	LED is on when in CW limit

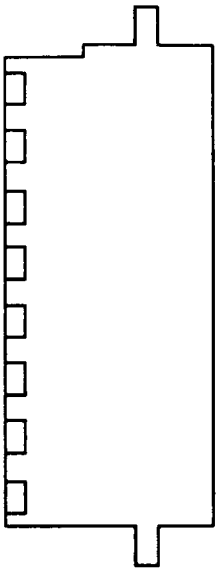
A. LED indicators
B. 6U, 14 TE front panel
C. Control and limit interface ports
D. Board interconnect cables
E. Unit specification label here
F. Connector J1 for input DC bus power and +5/ ± 12 VDC supplies (see figure 8-2B)

Figure 8-2A: Special 6U Eurocard Version of Unidex 1 DC

A. Mounting holes as per DIN 41612 specifications
B. .25" quick-connect lugs (typical)



BACK VIEW
OF CONNECTOR



SIDE VIEW
OF CONNECTOR

PIN 30 } PIN 26 }	Motor connections (16 AWG min)
PIN 28*	+ DC Bus
PIN 32*	- DC Bus (+DC Bus return)
PIN 6	+5 VDC supply (+130mA minimum)
PIN 20	+12 VDC supply (+50mA minimum)
PIN 22	-12 VDC supply (-40mA minimum)
PIN 24	Signal common (for +5/±12 supplies)
PIN 4	Heatsink (to shield)

(All other connections are reserved for future use.)

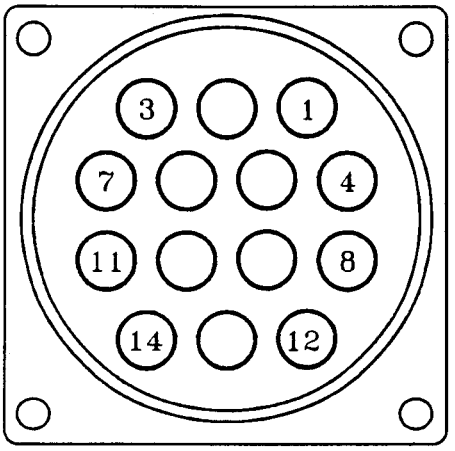
* IMPORTANT! +/- DC Bus power connections must be twisted pair with a length of no greater than 12 inches from DC bus filter capacitor. Filter capacitor must be at least 3000 μF, computer grade. Twisted pair should be no less than 14 AWG gauge.

Figure 8-2B: Quick-connect Mating Connector for connector J4 (shown in figure 8-2A)

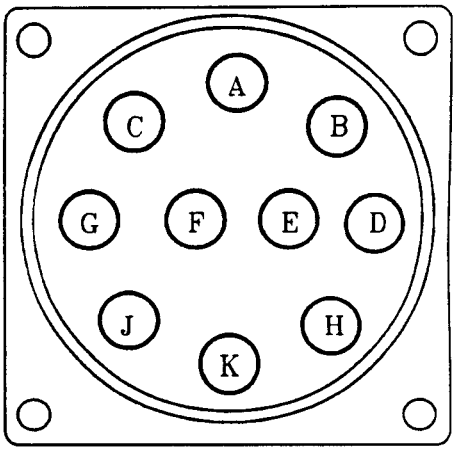
SECTION 8-3 MOTOR POWER CONNECTIONS

Model U1SA through U1SJ

A diagram of the motor power receptacle for the U1SA through U1SJ configurations (see figure 8-1) is shown below:



14 Pin Plastic Style
Models U1SA - U1SF



10 Pin Metal Style
Models U1SG - U1SJ

The connector can be of either two types. As shown above, a plastic style for models U1SA through U1SF, and a metal style for models U1SG through U1SJ.

Figure 8-3 shows the pin-out specifications for both connectors.

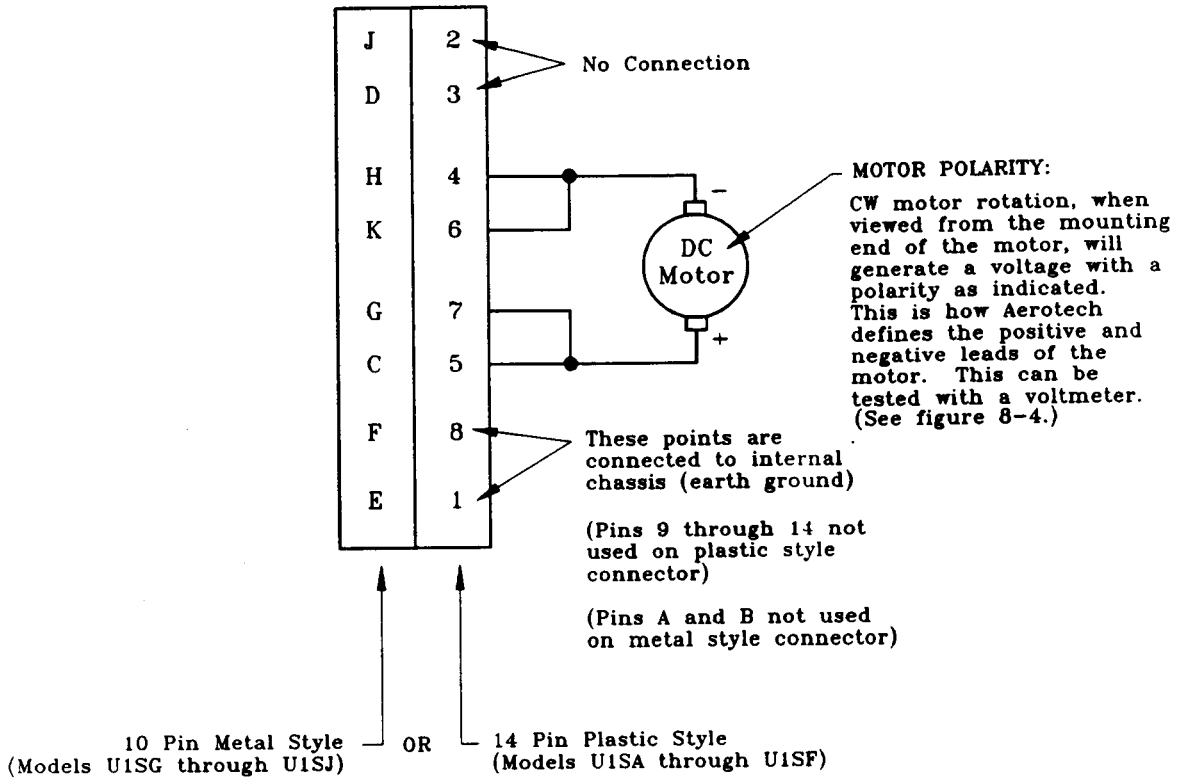


Figure 8-3: Unidex 1 DC Motor Connection Specifications for Models U1SA through U1SJ

Place a voltmeter's leads on the motor terminals (or leads) and rotate the motor CW. If the leads are connected properly, you should see positive deflection. (CW rotation is always as viewed from the mounting flange of the motor.)

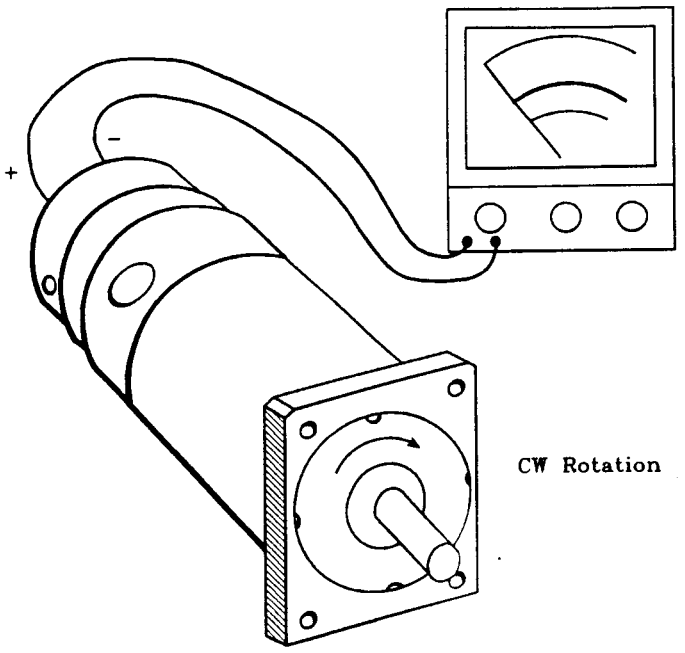
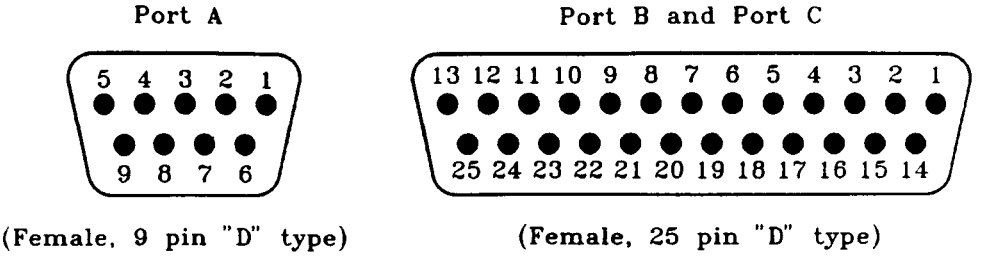


Figure 8-4: Motor Phasing

SECTION 8-4 CONTROL CONNECTIONS, PORT A, B & C

(For All Units)

There are three control connectors on Unidex 1 (labeled Port A, Port B and Port C). The location of each of these connectors is shown in figure 8-1 (Desktop/panel mount chassis version), and figure 8-2A (6U version). A description of each connector and its mate is given in figure 8-5.



NOTE: Outline drawings refer to Unidex 1 RECEPTACLES ONLY (mate of each is mirror image, as described below).

Mating connector for 9 pin "female" "D" connector. (Computer Interface (Port A))	Molded cable type "male" Belden, No. 49902. Solder pot type "male" TRW-CINCH, No. DEM-9P. Ribbon connector type "male" TRW-CINCH, No. FC-9P.
Mating connector for 25 pin "female" "D" connectors. (Input/Output (Port B) Limits (Port C))	Molded cable type "male" Belden, No. 49670. Solder pot type "male" TRW-CINCH, No. DBM-25P. Ribbon connector type "male" TRW-CINCH, No. FC-25P.

Figure 8-5: Outline and Mating Connector Description For Unidex 1 Control Connectors (port A, port B and port C)

SECTION 8-5 ENCODER/LIMITS CONNECTOR (PORT C)

The Encoder/Limit Connector (Port C) provides for the termination of the basic control interface signals between the DC motor and Unidex 1. Limit connections to this port are optically isolated. Motor travel limits and encoder signals are terminated at the limits connector. The pinouts for this connector are as follows (see figure 8-5 for receptacle outline.)

ENCODER/LIMITS CONNECTOR (PORT C)

Pin	Description
1	Shield
3,4,16	+5VDC
5	Limit +
6	/M
7	M
8, 9, 20, 21	Com
12	CW /Lmt
14	Cos
15	/Cos
17	Sin
18	/Sin
22	Home /Lmt
24	CCW /Lmt

(All other pins are spare connections.)

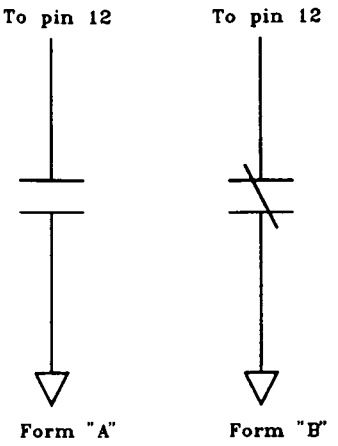
NOTE: Encoder must be "square" wave, differential line driver type.

Form "A" limit connections (normally open) are shipped standard, and are connected at pins 12, 24 and 22.

Form "B" contacts (normally closed) can also be accommodated for all three limit inputs. Since this configuration is optional, jumper JP1 on the main PC board must be reconfigured. The jumper definition is as follows:

(Standard) Limits, Normally Open - Jumper JP1, 1 to 2, Remove JP1, 2 to 3.
(Optional) Limits, Normally Closed - Jumper JP1, 2 to 3, Remove JP1, 1 to 2.

An example of the normally open versus the normally closed philosophy follows (CW Limit used as example):



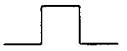
Contact positions are shown when not in limit

The CW and CCW limits provide travel limit protection for the drive system. It should be noted that the terms CW and CCW rotation are relative to Aerotech motors shipped with the Unidex 1. Direction of rotation is directly related to the DC motor wiring and the encoder wiring. Wiring on a user-supplied motor may require manipulation to provide the correct rotation, CW or CCW, as referred to within this manual.

The home limit switch is used to reference the home position (see section 5-8 for home command). When a home command is issued, the motor will turn CCW (standard) or CW (optional) until a "home limit switch" activation occurs. (See chapter 3 for setup of home direction and home speed). Upon "home limit switch" activation, the motor will then reverse and rotate in the opposite direction until the switch deactivates.

When the home switch reopens, rotation will continue until an active "Marker" indication is present at pins 6 and 7. At this time, motion will cease.

If interfacing to a standard optical encoder that has a once-per-revolution "marker" pulse, the "marker" input at pin 7 is active high:



and the "/marker" input at pin 6 is active low:

NOTE: All encoders supplied by Aerotech have once-per-revolution "marker" and "/marker" signals.

If using an encoder that does not have a "marker" signal, pin 7 should be tied high (+5V) and pin 6 should be tied low (common). It is important to note that the "marker" is ignored until the closure of the home switch during a home cycle. This allows the use of a "once per rev" marker pulse to be referenced to the home limit switch.

For most rotary motion stages, the "home limit switch" referenced above is an independent switch incorporated specifically for the "home" cycle. For linear motion stages, the "home limit switch" could be an independent switch as well. However, in most cases, the "CCW or CW limit switches" perform double duty, and act as the "home limit switch". Note that the process of putting the "home limit switch" input in parallel with the "CCW" or "CW limit switch" input is standardly done at the motor/stage, external to the Unidex 1 (reference chapter 9 and figure 9-2).

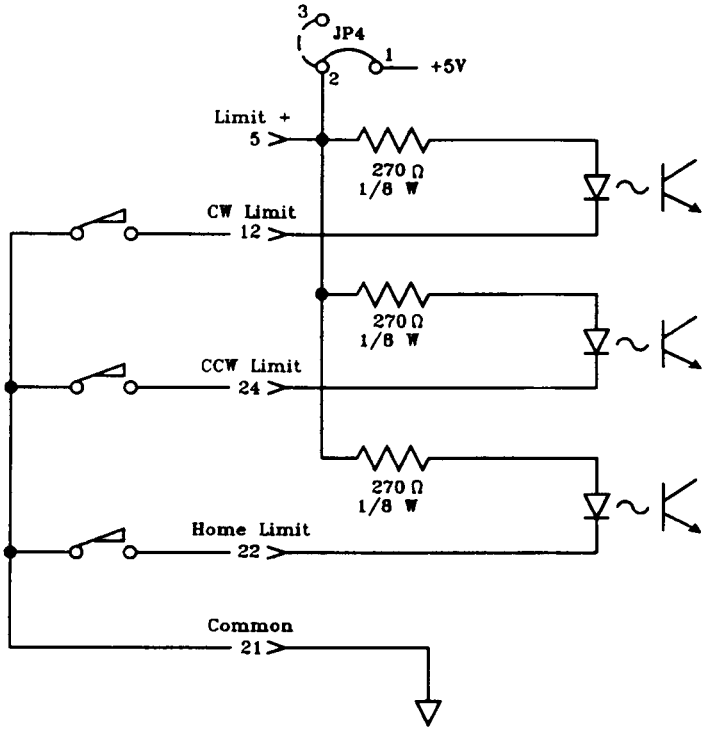
A. LIMIT INPUTS

The CW limit, CCW limit and home limit inputs are optically isolated. The inputs may be configured to use the internal +5V supply

(standard) or an external supply by the selection of an on-board jumper. Specifications for these inputs are the same as the opto isolated inputs (see section 8-6).

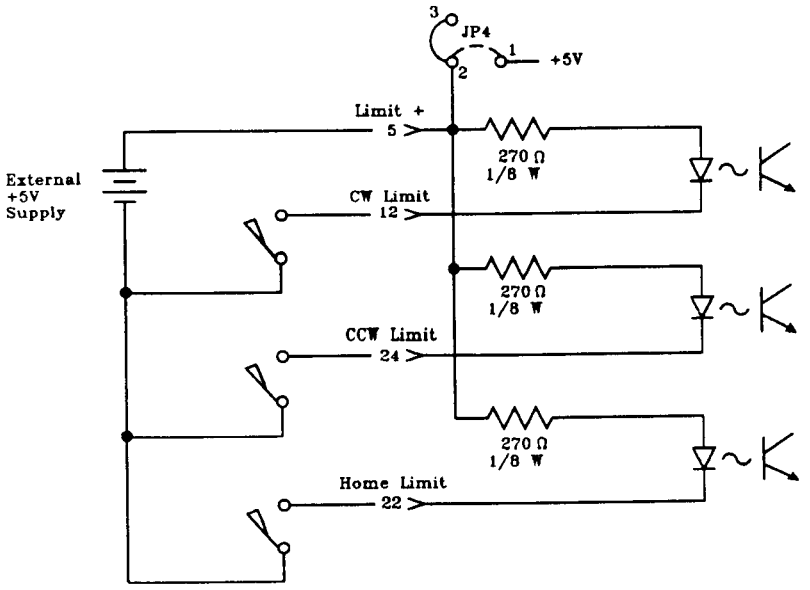
Following are typical examples. One of an internally powered limit application, and one of an externally powered limit application.

A. INTERNAL SUPPLY LIMIT CONFIGURATION
(Standard factory-shipped configuration)



NOTE: Remove JP4, 2-3 and jumper JP4, 1-2 for internal +5V supply. (This configuration is not isolated.)

B. EXTERNAL SUPPLY LIMIT CONFIGURATION
(Isolated-Consult factory)



NOTE: Remove JP4, 1-2 and Jumper JP4, 2-3 for external +5V supply. (This configuration is isolated.)

The presence of a visual indication for acknowledge CW limit, CCW limit and marker can be made by viewing the LED indicators on the front panel of the Unidex 1 (see figures 8-1 and 8-2). The optional hand held terminal also provides an indication of the presence of a limit or marker (see chapter 4).

B. ENCODER INPUTS

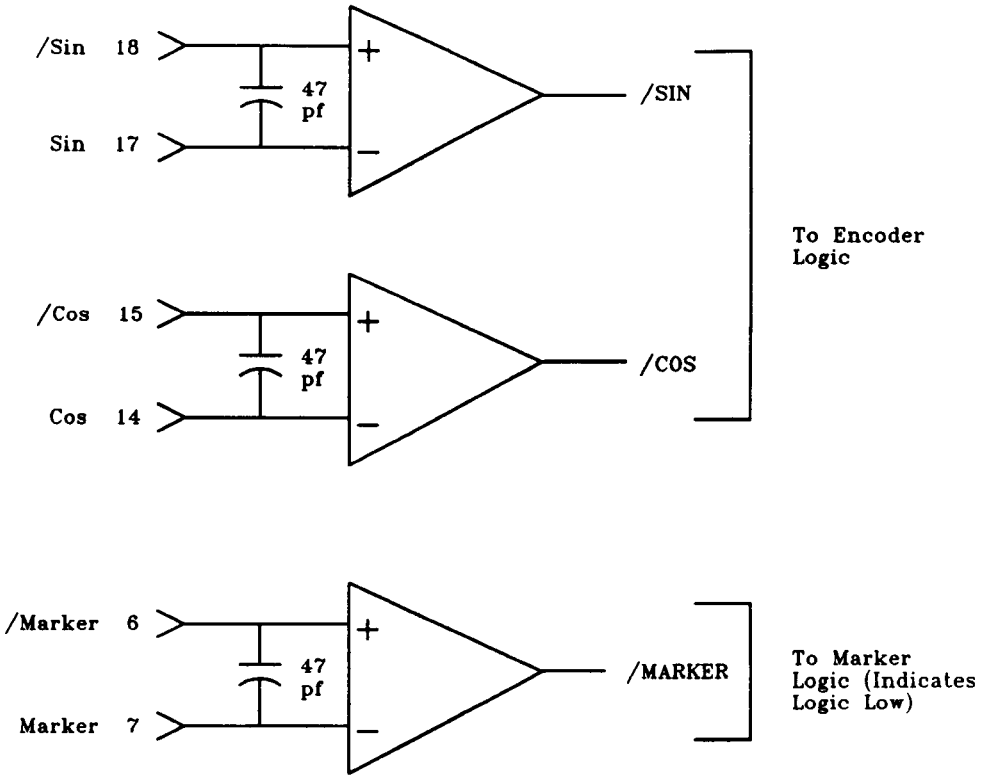
The encoder inputs are the same as the marker inputs. These inputs are the Sin, Cos and Marker inputs. The encoder must be a square wave type with line driver outputs. Logic levels for the encoder inputs are as follows:

Logic 0≤ .8 volts
Logic 1≥ 2 volts

NOTE: Encoder voltage must not exceed ±15volts.

Encoder phasing must be 90° ± 20° at 50kHz (minimum). (See figure 8-6 for encoder signal definition.) The encoder input configuration is shown in the next illustration.

"Times Four" encoder multiplication is performed internally in the Unidex 1. That is, the encoder frequency (resolution) is multiplied by four as the effective feedback resolution. Encoder resolutions of 200 lines or less are **not** recommended.



AM26LS32

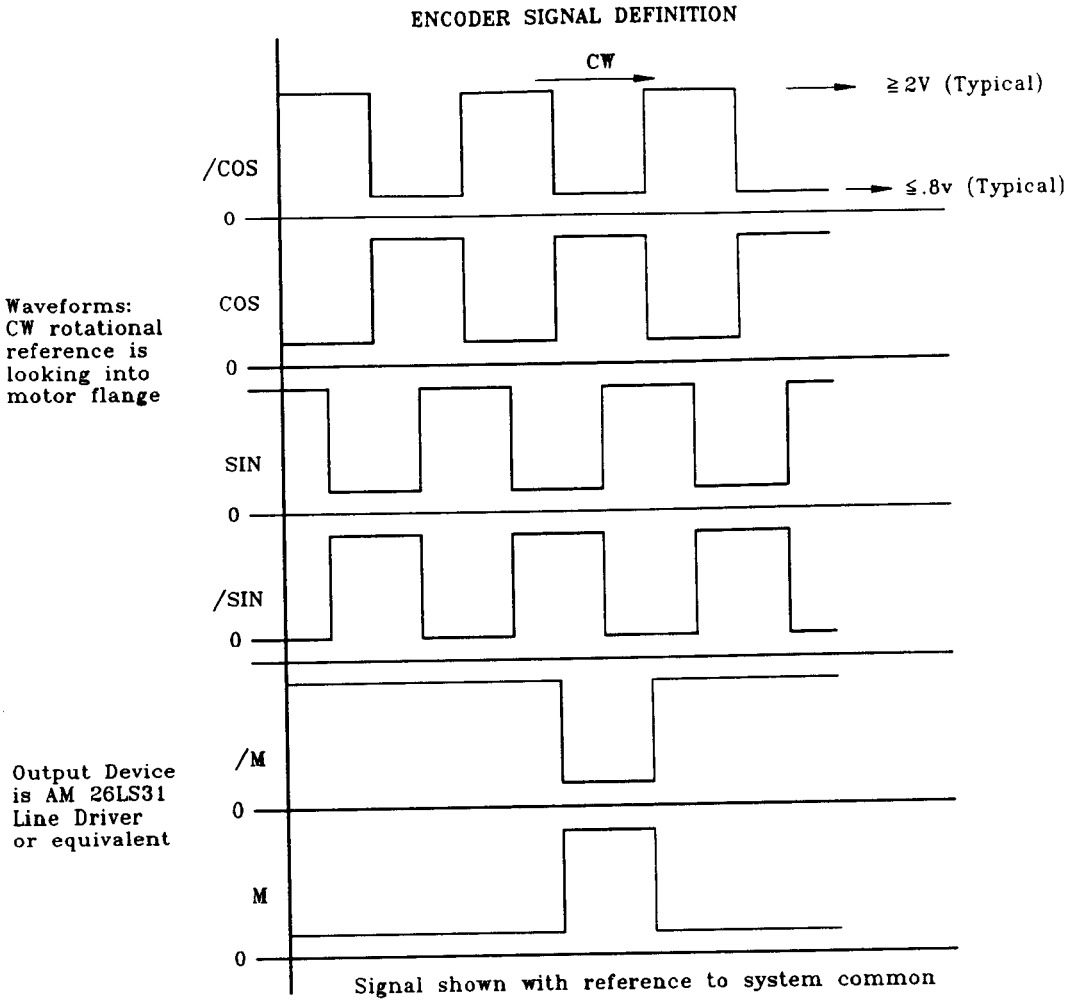


Figure 8-6: Encoder Signal Definition

SECTION 8-6 INPUT/OUTPUT CONNECTOR (PORT B)

This receptacle provides the means of terminating inputs (I1, I2, I3, I4) and outputs (O1, O2, O3, O4) and auxiliary control inputs.

The pinouts for the Port B receptacle are as follows (see figure 8-5 for outline drawing).

INPUT/OUTPUT CONNECTOR (PORT B)

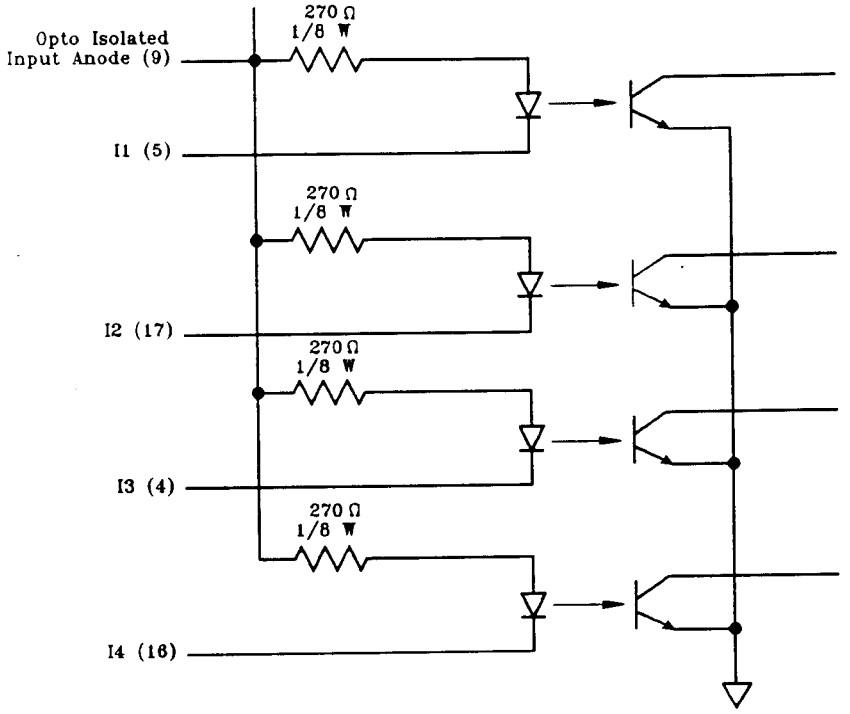
Pin	Description
1	Fault (Opto Isolated, Output)
2	Output 03 (Opto Isolated)
3	Output 01 (Opto Isolated)
4	Input I3 (Opto Isolated)
5	Input I1 (Opto Isolated)
6	/Setup
7	/Ext Clk (Opto Isolated, Input)
8	Common
9	Opto Isolated Input Anode
10	Spare
11	Current Command monitor signal (see section 3-10)
12	Spare
13	Shield
14	Output 04 (Opto Isolated)
15	Output O2 (Opto Isolated)
16	Input I4 (Opto Isolated)
17	Input I2 (Opto Isolated)
18	/Feedhold (Opto Isolated, Input) Stnd *
19	Ext Dir (Opto Isolated, Input)
20	+ 5 VDC
21	Opto Isolated Output Emitter
22	Spare
23	Spare
24	Spare
25	Spare

* Or /Reset (Opto Isolated, Input) - Optional. Consult factory.

Shield termination is made through the two strain relief screws on the connector.

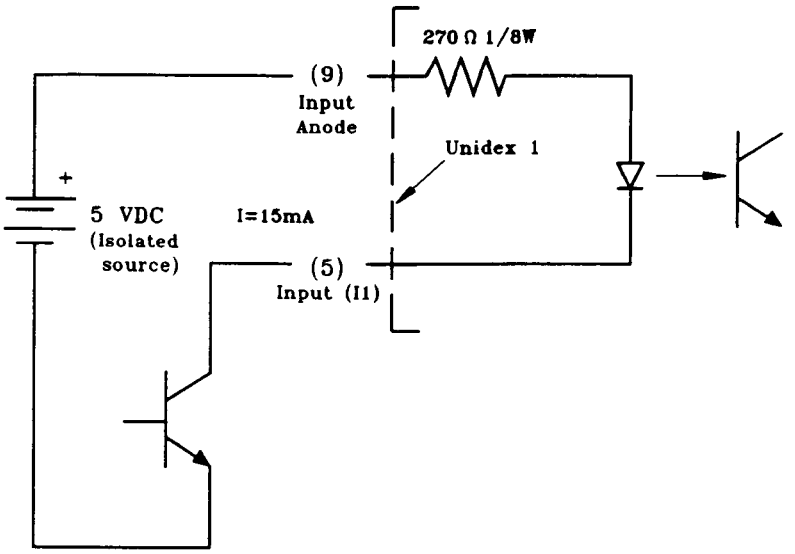
A. INPUT (I1, I2, I3, I4) CONFIGURATION

The four condition inputs (see section 5-8 for conditional input description) are Opto Isolated as shown in the following configuration.



NOTE: Input Anode (pin 9) is also used as a return for the Opto Isolated Ext. Clk., Ext. Dir. and Feedhold inputs.

B. TYPICAL ISOLATION INPUT APPLICATION EXAMPLE



OPTO ISOLATED INPUT SPECIFICATIONS

Maximum input current (I1, I2, I3, I4):	20mA
Maximum voltage (Input anode to any input):	6V
Activation voltage range (Input anode to any input):	4V - 6V
Maximum reverse voltage (Input anode more negative than any input):	6V

The opto isolated inputs may be made to operate at up to 30 volts by adding resistance in series with the input (I1 - I4) lines. To determine the resistance required, use the following formulas:

$$R_{(ohms)} = \frac{[(Voltage) - 5V]}{.015}$$

$$W_{(watts)} = .015 \times (Voltage - 5V)$$

Example: Selecting a resistor value for an operating voltage of 12 volts:

1) $$R = \frac{[(Voltage) - 5V]}{.015}$$

$$R = \frac{(12 - 5)}{.015}$$

$R = 467$ Ohms (Select nearest value)

(Use 470 Ohm resistor.)

2) $W = .015 \times (Voltage - 5V)$

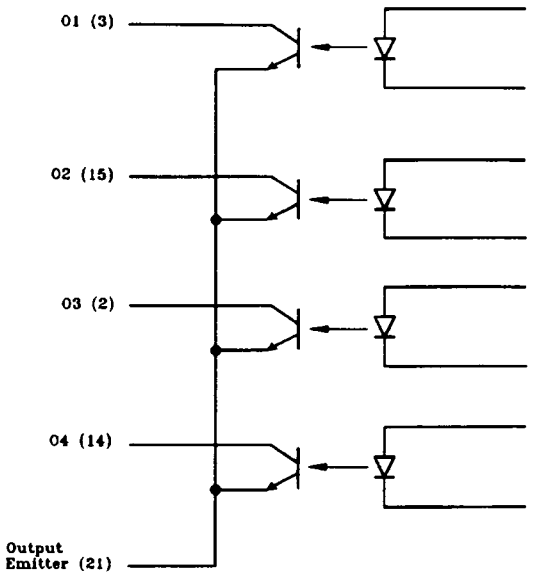
$W = .015 \times 7$

$W = .105$ Watt

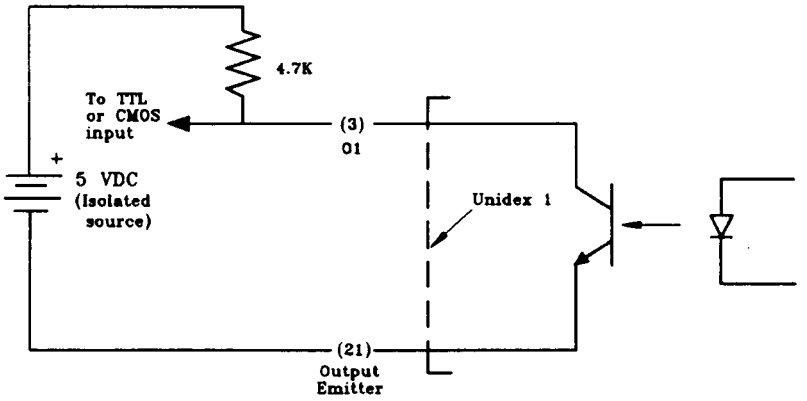
(Use 1/4 Watt or 1/2 Watt resistor.)

C. OUTPUT (O1, O2, O3, O4) CONFIGURATION

The four programmable outputs (see section 5-8 for Programmable Output description) are opto isolated with the following configuration:



Outputs are capable of driving one TTL load. A typical application is shown below:



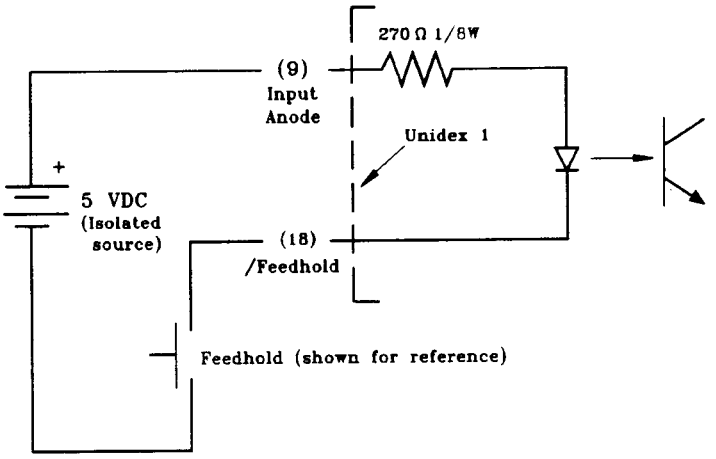
OUTPUT SPECIFICATIONS

Maximum voltage "Off State" (Collector to Emitter):	30V
Maximum reverse voltage:	-7V
Maximum collector saturation voltage (I_C = 2mA):	.5V
Maximum power dissipation:	150mW

See section 3-9 G for Output Active State setup.

D. FEEDHOLD CONFIGURATION

The feedhold input (see section 5-8 for commands associated with feedhold) is an opto isolated input with the following configuration:



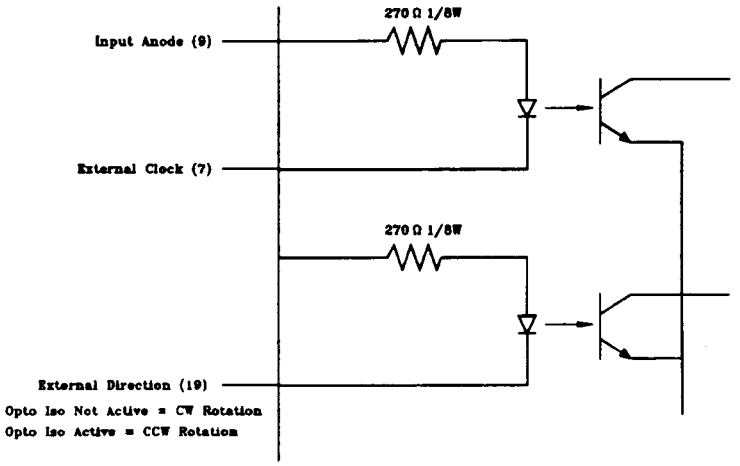
NOTE: Input common (Pin 9) is used for all opto isolated inputs.

The feedhold input specifications is the same as the opto isolated inputs. (See subsection B of this section for specifications.)

The feedhold input is designed to bring to a controlled stop, any of the 99 programs executed by the Unidex 1 indexer. The motion is allowed to continue when the switch is reopened.

E. EXTERNAL CLOCK AND DIRECTION

The external clock and direction inputs allow external clock and direction commands to be applied to the Unidex 1 controller. The Unidex 1 will keep track of these pulses, updating its internal position register. To operate in this mode, the Unidex 1 must be configured for "remote" operation. The clock and direction signal inputs are opto-isolated as shown below.



Because of opto-isolation, the maximum clock frequency is limited to about 5KHz (consult Aerotech if faster clock rates are required).

F. SETUP INTERFACE CONNECTIONS

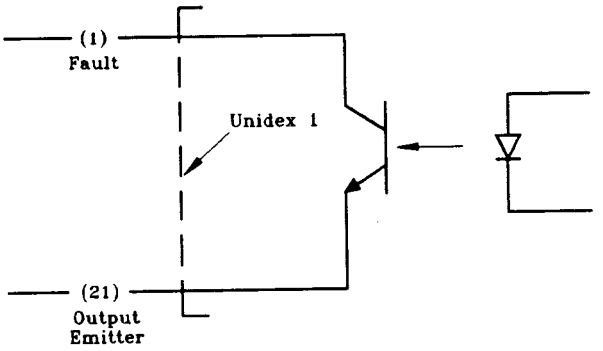
The setup line is used to put Unidex 1 in the setup mode (see chapter 3). For setup operation, connect the setup input to common. *At any other time the setup line should be left disconnected.*

PORT B

Pin 6	/Setup
Pin 8	Signal Common

G. FAULT INTERFACE CONNECTIONS

The Fault Interface is an optically isolated output used to indicate that Unidex 1 is not ready. The not ready state is indicated when the fault output transistor is off. When Unidex 1 is in the ready state the fault output transistor will be on. Specifications for the Fault interface are the same as the output specification (see subsection C of section 8-6). The fault output configuration is shown below:



SECTION 8-7 COMPUTER INTERFACE (PORT A)

This receptacle (Port A) provides for the termination of the Computer Interface. Following is the pinout listing for the RS- 232 receptacle. (See chapter 5 for details.)

COMPUTER INTERFACE (PORT A)

PIN	DESCRIPTION
1	Shield
2	Received data (RXD)
3	Transmitted data (TXD)
4	Data terminal ready (DTR)
5	Signal common
6	No Connection
7	Request to send (RTS)
8	No Connection
9	+5V (External +5V limited to -30mA max)

CHAPTER 9: MOTOR CONNECTIONS

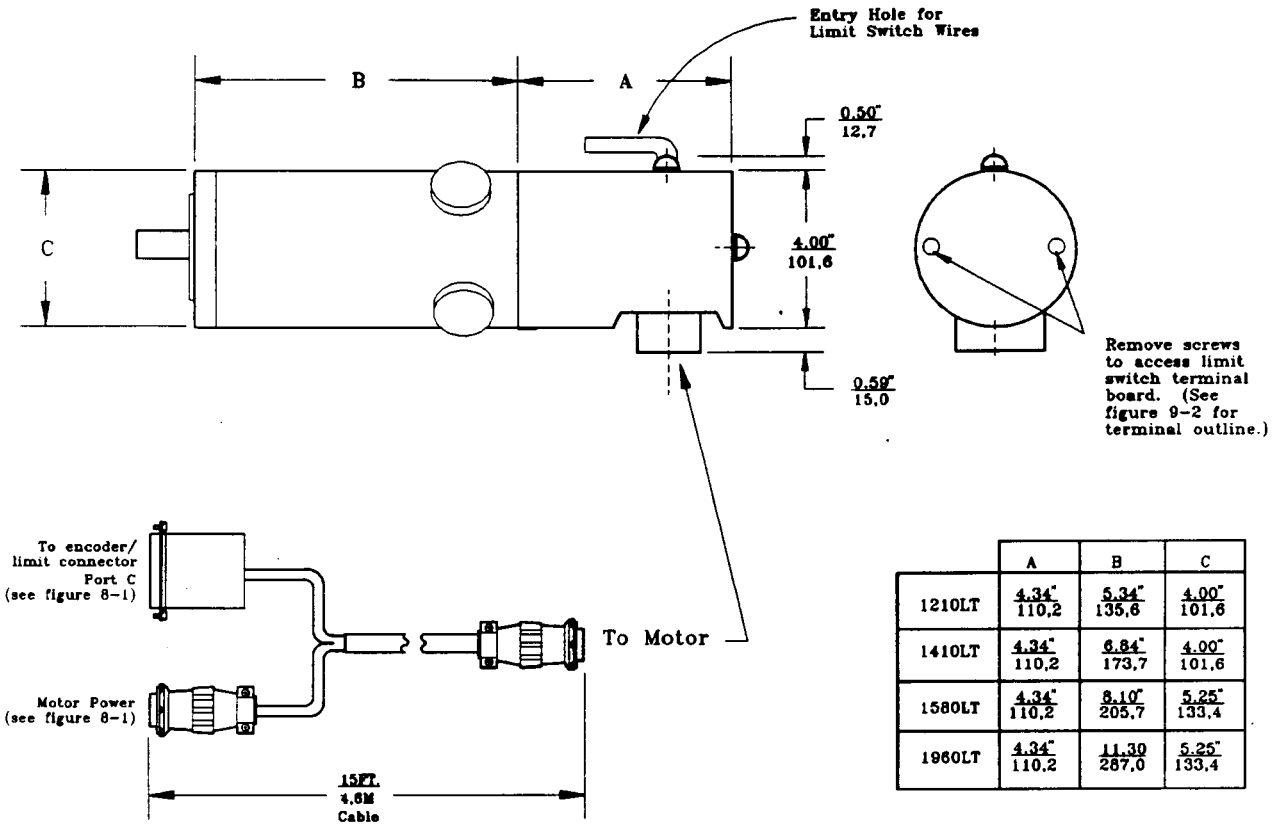
SECTION 9-1 MOTOR SPECIFICATIONS

This section defines the mechanical specifications for motor types 1017LT through 1960LT, listed in table 1-1 of this manual.

Mechanical specifications for these motors are shown in figure 9- 1. These specifications define general dimensions and are intended only for reference. (Consult *Aerotech Motion Control Product Guide* for detailed mechanical information regarding these motors.)

Figure 9-2 outlines the limit switch Terminal Board connections. Note that each motor contains a small terminal board interface, concealed by a back cover plate. The user can connect travel limit switch connections ("CW" and "CCW" limit switches) and the "home" reference limit switch connections to this terminal board. The portion of the motor cable going back to "Port C" on the Unidex 1 provides the return path for the limit switch connections.

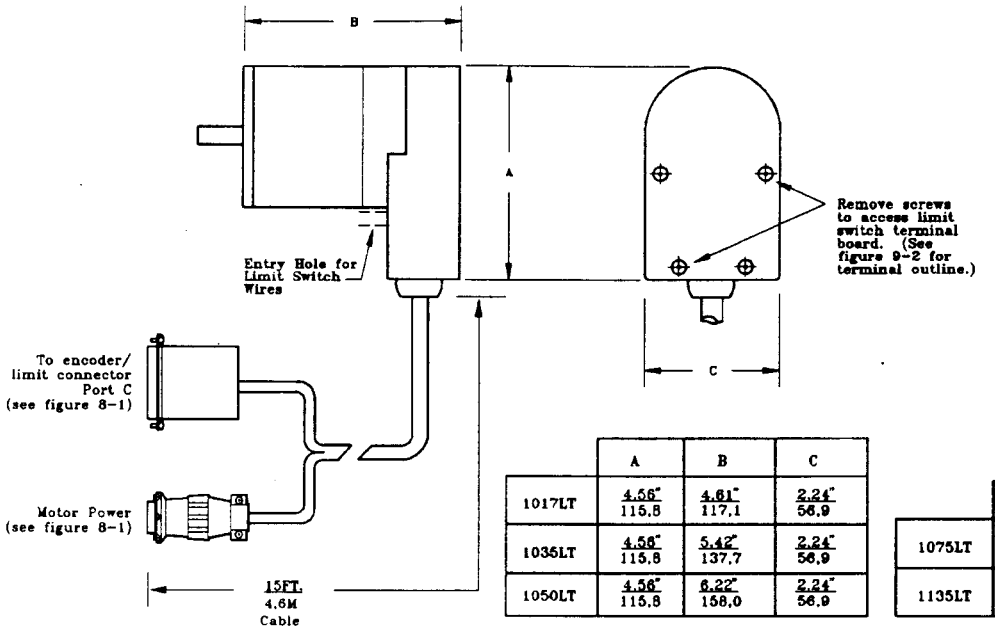
MS01 - 1210LT, 1410LT, 1580LT and 1960LT Motors



	A	B	C
1210LT	4.34" 110,2	5.34" 135,6	4.00" 101,6
1410LT	4.34" 110,2	6.84" 173,7	4.00" 101,6
1580LT	4.34" 110,2	8.10" 205,7	5.25" 133,4
1960LT	4.34" 110,2	11.30 287,0	5.25" 133,4

Figure 9-1: Motor Specifications

DC2 - 1017LT, 1035LT, 1050LT Motors
DC3 - 1075LT, 1135LT Motors



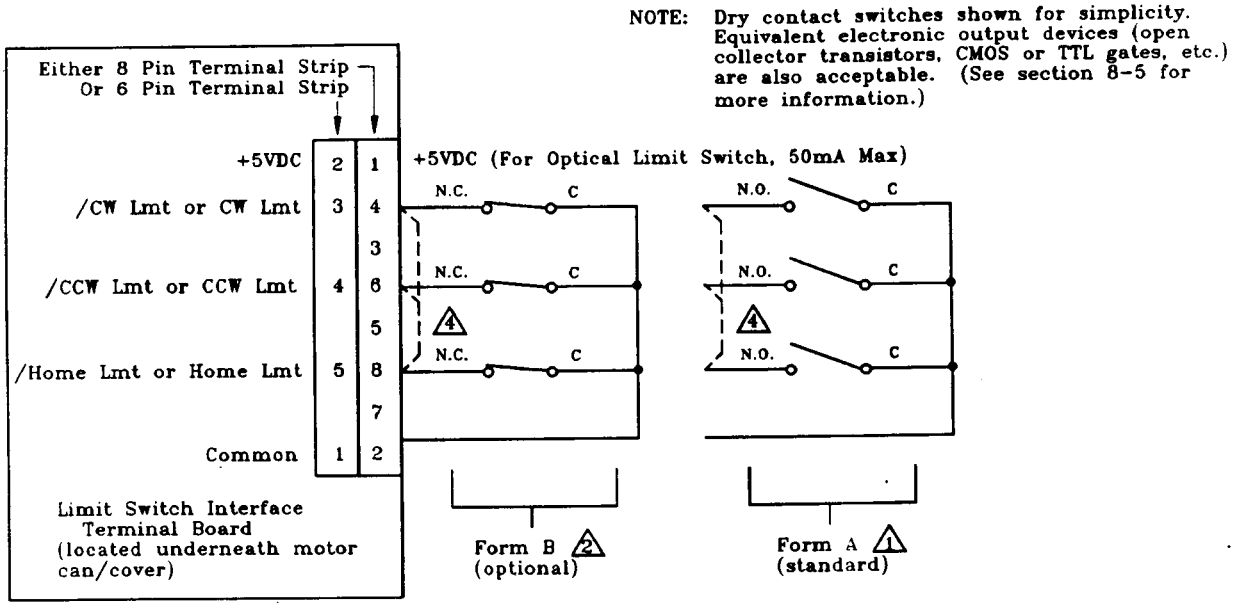
	A	B	C
1017LT	4.56" 115,8	4.61" 117,1	2.24" 56,9
1035LT	4.56" 115,8	5.42" 137,7	2.24" 56,9
1050LT	4.56" 115,8	6.22" 158,0	2.24" 56,9

DC2

	A	B	C
1075LT	5.25" 133,3	6.23" 158,3	3.24" 82,3
1135LT	5.25" 133,3	7.73" 196,4	3.24" 82,3

DC3

Figure 9-1: Continued



1 Form A, normally open (switch closes when in limit). Standard configuration unless otherwise specified.

2 Form B, normally closed (switch opens when in limit). Optional configuration.

3 All Unidex 1 Controllers are configured to accept (require) home limit inputs. This input may come from an independent switch (cam option on rotary stages) or may be paralleled from the CCW (standard) or CW (optional) limit switches (on linear stages).

Figure 9-2: Limit Switch Terminal Board Definitions

CHAPTER 10: TROUBLESHOOTING

Troubleshooting the Unidex 1 consists of reviewing the check list of possible software malfunctions, such as error status codes. The codes for the RS-232 Computer Interface mode are listed in section 5-6. For the hand held terminal mode, they are listed in section 4-3D, *Status Display*, which describes the procedure for polling error status messages. Both modes of polling are referenced in section 10-1, which follows. Section 10-2 lists possible hardware malfunctions associated with the Unidex 1.

WARNING: Troubleshooting and repair will be limited only to solutions listed in sections 10-1 and 10-2. Any attempt to repair the electronics without the supervision of an Aerotech trained field representative may void the warranty.

SECTION 10-1 SOFTWARE MALFUNCTIONS, CODES

POWER UP

Status Code	Possible Cause/Solution
System check fail: User memory checksum error. (Status Byte 2, Bit 4) *	A check sum verification is performed on the RAM upon power up. At the end of program editing or manual mode parameter changes, Unidex 1 modifies the checksum register. When Unidex 1 is powered up again, a sum of all bytes in the RAM is done. This sum is then compared to the check sum register. If they do not match, one or more bytes in RAM has been altered. **Note that a checksum error may not necessarily mean a damaged RAM. In some instances, battery backed RAM may be altered by a system software malfunction or a power failure during editing.**

* Information contained within parentheses is the status-code polling associated with the Computer Interface Mode (chapter 5). Information outside of parentheses is the status code description associated with the Hand Held Terminal Mode (chapter 4).

PROGRAM EDITING

Status Code	Possible Cause/Solution
Blk Not saved, Mem full. (Status byte 2, bit 1)	If an attempt to enter an additional block of statements is made after an End of Memory warning is encountered, and the block size is larger than the available remaining memory space, a "Block not saved" status will occur. Note that some block sizes are larger than others, depending on the statement being entered, so it is possible that small blocks (such as GoTo, GoSub,etc.) may be entered without being truncated. If the Memory Full warning is encountered after entering a block which exceeds the amount of memory available, the block will be ignored.
Memory Altered. (Status byte 2, bit 4)	When an existing program is accessed in the edit mode, Unidex 1 performs a checksum on the entire contents of program memory. Unidex 1 also evaluates the statements of each program block as they are pulled for editing purposes. Each

	statement is analyzed for the proper format. If in either of these two cases, a Memory Altered status occurs, Unidex 1 automatically performs a Memory Repair™ on the program which has been accessed. If undefined statements of a given block of the program are encountered, the entire block containing these statements is deleted. What is left is a program with missing blocks. The user need only scan this program and replace the missing blocks. Memory Repair™ is a very important feature because it reduces the possibility of Unidex 1 "locking up" if a faulty program is executed. A Memory Altered warning can only exist due to faulty RAM or some unrecoverable transient power glitch on the input power supply. In both cases, the occurrence is rare.
Format Warning. (Status byte 2, bit 3)	A character was entered that did not conform to the required format during a command or program block entry.

RUN MODE

Status Code	Possible Cause/Solution
Illegal byte in memory. (Status byte 3, bit 4)	If, during manual or program execution, a user memory byte cannot be identified by Unidex 1, an illegal byte in memory status will occur. This detection mechanism monitors on a block-by-block basis, unlike the checksum and block verification mechanisms described in the previous sections on Power Up and the Editing Mode. These mechanisms "scan" the entire user memory before execution takes place.
Exceeded 8 repeat loops. (Status byte 4, bit 3)	Unidex 1 programming mode allows only a maximum of eight "nested" repeat loops (i.e., a loop within a loop, eight times). A possible user-stack overload condition exists if the level of loop nesting exceeds eight levels. Remember, an unlimited amount of unnested repeat loops are allowed. The "Exceeded 8 Repeat Loops" error is detected during program execution.

Invalid repeat loop end. (Status byte 4, bit 1)	This status declares that a repeat loop "End Repeat" statement was encountered without a preceding "Start Repeat" statement. This error is detected during program execution.
Incomplete repeat loops. (Status byte 4, bit 2)	This status declares that a "Start Repeat" statement was encountered without a following "End Repeat" statement. This error is detected at the End of Program execution.
Incomplete subroutines. (Status byte 4, bit 5)	This error status is similar to an "Incomplete Repeat Loop" error status in that a "GoSub" statement was detected without a following "Subroutine Return" statement. This error is detected at the End of Program execution.
Missing Label. (Status byte 4, bit 7)	This status declares that a program label does not exist for a label number specified in a given GoTo, GoSub, Conditional GoTo, Conditional GoSub or similar statement. This error is detected during program execution.
Exceeded 8 subroutines. (Status byte 4, bit 6)	This error status is similar to an "Exceeded 8 Repeat Loops" error status in that a

	maximum number of 8 "nested" subroutines has been exceeded. A possible user stack overload exists if the level of subroutine nesting exceeds eight levels. Remember, up to 99 unnested subroutines are allowed (i.e., up to 99 available labels per program). This problem is detected during program execution.
Invalid return-from-sub. (Status byte 4, bit 4)	This status declares that a "Subroutine Return" statement exists without a corresponding "GoSub" statement. This error is detected during program execution.
No programs in memory. (Status byte 3, bit 6)	This status declares that no programs exist in user memory as indicated by the directory command.
Invalid program #. (Status byte 3, bit 5)	This status declares that a specific program number does not exist in user memory or it cannot be used for the requested function.
Axis in limit. (Status byte 3, bit 0)	If a CW or CCW limit is encountered during manual, block or auto run modes, an "Axis in Limit" status will occur. Motion will be

	stopped and program execution will be terminated.
Position loop error. Reset to enable. (Status byte 3, bit 2.)	A position loop (PID loop) trap has been executed (see section 3-10). Maladjusted loop gains or a faulty encoder usually causes a "trap". Motor/encoder not phased properly can cause a "trap" as well.
Illegal Program Exit. (Status byte 3, bit 3)	This status declares that a program has been exited during an index or home operation. (The move is stopped when the program is exited.)
Program Limit. (Status byte 3, bit 1)	A program limit has occurred during program execution. Program has been terminated and axis has stopped.

SECTION 10-2 HARDWARE MALFUNCTIONS

Symptom	Possible Cause/Solution
"Reset" LED blinks on and off after power is applied.	A system RAM Read/ Write error caused by a memory byte failure somewhere within the 8K RAM chip of the Unidex 1 control board. On power up, Unidex 1 checks all byte locations of the RAM by

	writing a value back for verification. The user area (program area) is not tested.
"Reset" LED stays on continously after power is applied.	Input AC voltage supplies are below minimum operating level (see table 1-1).
	Short circuit condition in DC motor wiring (see section 8-1).

CHAPTER 11: SERVICE AND REPAIR

General repair of equipment consists entirely of solutions listed in chapter 10, *Troubleshooting.*

If under warranty, repairs of defective electrical components should not be attempted, *since to do so would void the entire warranty. If necessary, any on-site service should be performed by an experienced Aerotech-trained electronics technician*

SECTION 11-1 SHIPMENT

The procedure for shipping equipment back to Aerotech, which is described below, pertains to warranty as well as non-warranty repairs.

1. Before shipping any equipment back to Aerotech, the person making the return must call ahead for a "*Return Authorization Number*". *(Have your serial number on hand when calling.)*

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 part to the factory.

Warning: Damage due to improper packaging voids warranty!

Aerotech Sales and Service offices are listed on the following pages. 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 NORTHEAST
Executive Suite 120
270 Farmington Avenue
Farmington, CT 06032
Phone (203) 673-3330
or (203) 673-2503
FAX (203) 674-1536

MA, CT, VT, ME, RI, NH, eastern NY

AEROTECH CENTRAL-WEST

26791 Lake Vue Drive #8
Perrysburg, OH 43551
Phone (419) 874-3990
FAX (419) 874-4280
MI, IN, KY, western OH

AEROTECH MIDWEST
PO Box 625
Dundee, IL 60118
Phone (312) 428-5440
FAX (312) 428-5471
IL, MO, KS, WI, MN, ND, SD, IA, NE

AEROTECH, CENTRAL-EAST
856 Cottonwood Drive
Monroeville, PA 15146

Phone (412) 373-4160
FAX (412) 373-4163
WV, western PA, western NY, eastern OH

AEROTECH SOUTHWEST
6001 Village Glen Drive
#3101
Dallas, TX 75206
Phone (214) 987-4556
FAX (214) 987-4706
TX, OK, LA, AR, CO, UT,MT, WY, ID, NM

AEROTECH NORTHWEST

444 Castro Street
Suite 400
Mountain View, CA 94041
Phone (415) 967-4996
FAX (415) 967-4998
northern CA, OR, WA, NV

AEROTECH WEST

Suite 217
7002 Moody Street
La Palma, CA 90623
Phone (213) 860-7470
FAX (213) 860-4639
AZ, southern CA

AEROTECH MID-ATLANTIC
521 Kingwood Road
King of Prussia, PA 19406

Phone (215) 265-6446
FAX (215) 265-3566
MD, DC, DE, NJ, northern VA, eastern PA

AEROTECH SOUTH ATLANTIC
8804 Lomas Court
Raleigh, NC 27615
Phone (919) 848-1965
FAX (919) 848-3393
NC, TN, southern VA, AL, FL, GA, SC, MS

INTERNATIONAL SALES OFFICES

AEROTECH LTD.
3 Jupiter House, Calleva Park
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

AEROTECH AUSTRALASIA
224 Carr Street
Suite 7
Leederville 6007
Western Australia
Phone (619) 227-6772
FAX (619) 227-6670

INTERNATIONAL REPRESENTATIVES

BRASITEC
Rua Americo Brasiliense, 2069
Chacara Santo Antonio
Cep 04715 - Sao Paulo - SP
Brazil
Phone (5511) 523 4044
TLX 1130691 BRTC
FAX (5511) 521 0221

Y. BEN MOSHE
PO Box 18125
Tel Aviv 61181
Israel
Phone (9723) 7515007
or (9723) 7513268
TLX 342436 BMS IL
FAX (9723) 727319

SIMCO
208, First Floor
Ajmeri Gate
New Delhi 110 006 India
Phone 730206
TLX 031-62176 HARS IN
FAX (9111) 510697

OPTIKON CORPORATION LTD.

410 Conestogo Road
Waterloo, Ontario
Canada, N2L 4E2
Phone 519-885-2551
FAX 519-885-4712

DONG DO TRADING CO. LTD.
Rm 903, Kwang Sung Bldg.
831-47 Youksamdong
Kangnam-Ku, Seoul, Korea
Phone (822) 556-2292
FAX (822) 556-2902
TLX 29734 DONG DO

TOKYO INSTRUMENTS INC.

Asahi-Seimei Bldg.
6-8-10 Nishikasai
Edogawa-Ku
Tokyo 134 Japan
Phone (813) 686-4711
FAX (813) 686 0831

HISCO (MALAYSIA) SDN.BHD.

1 Lorong SS13/6A
Subang Jaya Indust. Estate
47500 Petaling Jaya
Selanger, Malaysia
Phone (603) 733-4236
FAX (603) 733-6281
TLX 36226 HISCO MA

HADLAND PHOTONICS PTY. LTD.
19A Hampshire Road
Glen Waverley
Victoria, 3150, Australia
Phone (613) 560 2366
FAX (613) 560 8402
TLX AA 35670

Index

Addendum One

UNIDEX 1 TEST PROGRAM

Unidex 1 Test Program

This program can be run on any IBM compatible computer. The program was written using Microsoft QuickBASIC 4.0 software. This program is available on disk in 2 versions. The one version requires the QuickBASIC software (UN-1TEST.BAS). The other version may be executed from DOS (UN-1TEST.EXE). There is a batch file to load all needed files to run this program from DOS. To run the Setup file, run "U1SETUP.BAT" from the floppy disk drive.

This program allows you to become familiar with the Aerotech Unidex 1 controller using the RS-232 communications port. If you have not read the manual on the Unidex 1 controller, it is advisable to get acquainted with the manual at this time.

The test program has 3 communication functions for the Unidex 1 controller. The first function allows you to change the parameters in the "SETUP" mode of the Unidex 1 controller. The program prompts for the communication parameters, such as Baud Rate, Parity, Bit Size, and Stop Bits. Other setup parameters can be changed, after the Unidex 1 controller has been powered with the setup jumper installed in Port B. The program will instruct you on the proper procedure for preparing the Unidex 1 controller for altering the setup parameters. In the program, there are Help screens available to assist you in programming the setup parameters. If the Default parameters are being used, then the program will proceed with the next function after selecting the default values.

The second function is the immediate mode. You must enable the communication port by typing "> >" and the address number of the Unidex 1. Since the default address is "01", you would, for example, type "> >01" and <ENTER>. To reset the Unidex 1 controller, use [Ctrl]-D and <ENTER>. The reset LED will blink momentarily. After this reset command, you must enable the port again. All system and programming commands can be output in the immediate mode. It is recommended that after enabling the port, that the command "N" is output, so that the serial poll status byte requested by the "Q" com-

mand and the "PS" print status bytes command will be output by the Unidex 1 controller in hexadecimal format.

NOTE: **The "PS" command (Print Status Bytes) and the "PS *" command (Program Stop) are not the same commands. "PS" is a system command, while the "PS *" is a program command. Refer to manual for additional information on these commands.**

There are commands to assist you while in the immediate mode. Two of the commands are for help screens. "HS" command will display help screens for the Unidex 1 system commands, while "HP" will display the program commands. The command "EX" will exit you from the immediate mode. All commands will be executed when the <ENTER> key is pressed.

The final function of this program is to transfer files from the Unidex 1 controller to either a disk file or the parallel printer (LPT1). Files can be input from disk and output to the Unidex 1 controller. When the file is transferred from the Unidex 1 to the disk or printer, all of the output of the Unidex 1 will be stored in the file or printed on the printer. You can edit the program by using any text editor that is available.

Programs stored on disk that were created by a text editor do not require any of the data in the header or the termination characters at the end of the text file. This test program will handle the necessary output to the Unidex 1.

You can store the Unidex 1 program under any valid filename and extension. This program will permit you to copy from file to a new filename on the disk. A disk file can be output directly to the printer. After any file transfer has been completed, the program will return to the main selection screen. You must remember the names of the files on disk, since this program does not have the Display Directory Command, "DIR".

To exit the program, either press "3" when in the main selection screen, or press [Ctrl]-Break. If the program hangs up, which can hap-

pen if the "*" is not output after a program command, you may need to reset the host computer.

```
'   UNIDEX 1 TEST PROGRAM by Edward R. Burk   MAY 1988
CLS
CR$ = CHR$(13)
GOSUB 8000
SETUP$ = "COM1:9600,E,7,1"          'DEFAULT PARAMETERS ON UNIDEX 1
LOCATE 8, 1
PRINT "Are you using default communication values: 9600,E,7,1? (Yes or N
GOSUB 8200
IF A$ = "Y" THEN 100
IF A$ = "N" THEN 10

'  GET NEW BAUD RATE

10 LOCATE 10, 1
20 PRINT "Enter baud rate 110,150,300,600,1200,2400,4800, or 9600
LOCATE 10, 57
INPUT BR$
IF LEN(BR$) < 3 OR LEN(BR$) > 4 THEN 10
FOR I = 1 TO LEN(BR$)
IF ASC(MID$(BR$, I, 1)) < 48 OR ASC(MID$(BR$, I, 1)) > 57 THEN 10
NEXT I
N = VAL(BR$)
SELECT CASE N
CASE 110, 150, 300, 600, 1200, 2400, 4800, 9600
GOTO 30
CASE ELSE
GOTO 10
END SELECT
30 SETUP$ = "COM1:" + BR$

' GET PARITY TYPE

40 LOCATE 12, 1
PRINT "Enter parity type:  (E)ven   (O)dd   (N)o parity
DO
DO
P$ = INKEY$
LOOP WHILE P$ = ""
P$ = UCASE$(P$)
IF P$ = "E" OR P$ = "O" OR P$ = "N" THEN EXIT DO ELSE GOTO 40
LOOP
50 SETUP$ = SETUP$ + "," + P$

' GET NUMBER OF DATA BITS

60 LOCATE 14, 1
PRINT "Enter number of data bits  7 or 8 ?       "
DO
N$ = INKEY$
IF N$ = "7" OR N$ = "8" THEN 70
LOOP
GOTO 60
70 SETUP$ = SETUP$ + "," + N$

' GET NUMBER OF STOP BITS
```

```
80 LOCATE 16, 1
PRINT "Enter number of stop bits  1 or 2 ?       "
DO
S$ = INKEY$
IF S$ = "1" OR S$ = "2" THEN 90
LOOP
90 SETUP$ = SETUP$ + "," + S$
100 SETUP$ = SETUP$ + ",RB17000,OP5000"          ' SET TIMEOUT FOR 5 SECO
                                                 ' SET RECEIVE BUFFER TO

IF A$ = "Y" THEN GOTO 110
LOCATE 19, 1
PRINT "Does the UNIDEX 1 need to be set for communication parameters"
PRINT "other than the default parameters ?  (Yes or No)"
GOSUB 8200
IF A$ = "Y" THEN GOSUB 7000

110 OPEN SETUP$ FOR RANDOM AS #1        'OPEN COM1 PORT.

120 GOSUB 8000                          'SELECT MODE OF OPERATION.
LOCATE 8, 20
PRINT "(1) Immediate Mode"
LOCATE 10, 20
PRINT "(2) File Mode"
LOCATE 12, 20
PRINT "(3) End Program"
LOCATE 24, 1
PRINT "Select? ";
DO
A$ = INKEY$
IF A$ = "1" THEN 200                    'DIRECT COMMANDS TO UNIDEX 1
IF A$ = "2" THEN 500                    'TRANSFER OF FILES TO DISK OR PRINTER
IF A$ = "3" THEN 600                    'EXIT PROGRAM
LOOP
200 CLS
GOSUB 8000
LOCATE 8, 33
PRINT "Immediate Mode"
LOCATE 10, 5
PRINT "   Immediate mode allows you to output any system or program comm

PRINT "   the UNIDEX 1 control.  You are limited to 255 characters per l

PRINT "   need help, type HS for System commands and HP for Programming

PRINT "   If you use  'P'  print commands such as PX * then the X axis p

PRINT "   be displayed.  Use File Mode to store and retrieve programs fr

PRINT "   File Mode is also use to print programs to your LPT1 port.  Us

PRINT "   commands to redirect I/O from LPT1 to COM1.  Type EX to exit t

Time = 5                                'TIME IN SECONDS TO CLEAR COMMAND.
210 GOSUB 8100
```

```
215 GOSUB 8000
LOCATE 8, 33
PRINT "Immediate Mode"
LOCATE 20, 1
INPUT "Command"; A$
IF LEN(A$) = 1 THEN 220
IF LEFT$(A$, 2) = "EX" THEN 120
IF LEFT$(A$, 2) = "HS" THEN 225
IF LEFT$(A$, 2) = "HP" THEN 225

220 PRINT #1, A$
DELAY = .05                          ' DELAY TO ALLOW COMMAND TO BE EXECUTED.
225 SELECT CASE A$

CASE "HS", "HP"                      ' DISPLAY HELP SCREENS.
GOTO 300

CASE ">>00" TO ">>31"                ' CHECK FOR DEVICE ADDRESSING.
A$ = INPUT$(5, #1)
GOTO 215

CASE "Q"                             ' SERIAL POLL OF UNIDEX 1.
DO
A$ = INPUT$(1, #1)
PRINT A$;
IF A$ = CHR$(10) THEN 210
LOOP

CASE "PX", "PX*"                     ' READ X POSITION.
A$ = INPUT$(13, #1)
PRINT "X AXIS POSITION IS "; A$
GOTO 210

CASE "PS*"                           ' PROGRAM STOP COMMAND.
GOTO 210

CASE "PS"                            ' READ STATUS BYTES.
PRINT "STATUS = ";
DO
A$ = INPUT$(1, #1)
PRINT A$;
IF A$ = CHR$(10) THEN 210
LOOP


CASE "PD", "P00" TO "P99"            ' PRINT TO SCREEN OF DIRECTORY OR PROGRAM
250 CLS
GOSUB 8000
LOCATE 8, 1
DO
A$ = INPUT$(1, #1)
IF A$ = CHR$(3) THEN 210
IF A$ = "%" THEN 250
PRINT A$;
IF A$ = CHR$(10) THEN GOSUB 8300
```

```
LOOP

CASE ELSE
END SELECT

GOSUB 8300
GOTO 210

300 IF A$ = "HP" THEN 310
H$ = "UN1SYS"
HELP = 5
N = 1
GOTO 320
310 H$ = "UN1PRG"
HELP = 6
N = 1
320 CLS
HELP$ = H$ + RIGHT$(STR$(N), 1) + ".HLP"
ON ERROR GOTO 400
OPEN HELP$ FOR INPUT AS #2
330 HP$ = INPUT$(1, #2)
IF HP$ = CHR$(10) THEN 330
IF HP$ = CHR$(27) THEN 340
PRINT HP$;
GOTO 330
340 A$ = INPUT$(2, #2)
CLOSE #2
LOCATE 23, 1
PRINT "PRESS KEY:    ESC- exit    UP ARROW- next screen    DOWN ARROW- p
DO
HP$ = INKEY$
IF HP$ = CHR$(27) THEN 215
IF HP$ = CHR$(0) + CHR$(72) THEN 350
IF HP$ = CHR$(0) + CHR$(80) THEN 360
LOOP
350 N = N + 1
IF N > HELP THEN N = 1
GOTO 320
360 N = N - 1
IF N < 1 THEN N = 1
GOTO 320
400 LOCATE 23, 1
PRINT "Help screens are not available."
DELAY = 5
GOSUB 8300
RESUME 120



500 CLS
GOSUB 8000
LOCATE 8, 35
PRINT "File Mode"
LOCATE 12, 20
PRINT "(1) Output file to Printer"
```

```
LOCATE 14, 20
PRINT "(2) Output file to Disk"
LOCATE 16, 20
PRINT "(3) Output file to UNIDEX 1"
LOCATE 23, 1
PRINT "Select";

DO
B$ = INKEY$
IF B$ = "1" OR B$ = "2" OR B$ = "3" THEN 510
LOOP

510 SELECT CASE B$
CASE "1"
OUTPUT$ = "Printer"

CASE "2"
OUTPUT$ = "Disk"

CASE "3"
OUTPUT$ = "UNIDEX 1"

CASE ELSE
END SELECT

CLS
GOSUB 8000
LOCATE 8, 35
PRINT "File Mode"
LOCATE 12, 20
PRINT "(1) Input from Disk"
LOCATE 14, 20
PRINT "(2) Input from UNIDEX 1"
LOCATE 23, 1
PRINT "SELECT"

DO
B$ = INKEY$
IF B$ = "1" OR B$ = "2" THEN 520
LOOP

520
SELECT CASE B$
CASE "1"
INPPUT$ = "Disk"
CASE "2"
INPPUT$ = "UNIDEX 1"
CASE ELSE
END SELECT

CLS
GOSUB 8000
LOCATE 8, 35
PRINT "File Mode"
LOCATE 12, 5
```

```
PRINT "OUTPUT- ";
SELECT CASE OUTPUT$
CASE "Disk"
GOSUB 1000
OUTFILE$ = DISKFILE$
CASE "UNIDEX 1"
GOSUB 1100
OUTFILE$ = UN1PRG$
CASE ELSE
END SELECT
LOCATE 16, 5
PRINT "INPUT- ";
SELECT CASE INPPUT$
CASE "Disk"
GOSUB 1000
INFILE$ = DISKFILE$
CASE "UNIDEX 1"
GOSUB 1100
INFILE$ = UN1PRG$
CASE ELSE
END SELECT
LOCATE 20, 5
PRINT "Input from  "; INPPUT$; ":  "; INFILE$; "    Output to    "; OUTPU
LOCATE 22, 5
PRINT "Has the UNIDEX 1 addressed? (Y)es or (N)o"
GOSUB 8200
IF A$ = "N" THEN 120


CLS                                    ' Input of File.
GOSUB 8000
LOCATE 8, 35
PRINT "File Mode"
Time = 0
LOCATE 12, 1
PRINT "INPUT OF "; INFILE$
SELECT CASE INPPUT$

CASE "Disk"                              ' Input of Disk File

ON ERROR GOTO 537
OPEN INFILE$ FOR INPUT AS #2
OPEN "TTTTEMP.TXT" FOR OUTPUT AS #3

530 TEMP$ = INPUT$(1, #2)
PRINT #3, TEMP$;
PRINT TEMP$;
IF TEMP$ = CHR$(3) THEN 535
GOTO 530
535 DELAY = 2
CLOSE #3, #2
GOSUB 8100
GOTO 550

537 IF ERR = 62 THEN 538
```

```
LOCATE 23, 1
PRINT "File does not exist"
DELAY = 5
GOSUB 8300
RESUME 120
538 PRINT #3, "%"; CHR$(13); CHR$(10);
PRINT #3, CHR$(3);
RESUME 535


CASE "UNIDEX 1"                          ' Input of UNIDEX 1 Program

OPEN "TTTTEMP.TXT" FOR OUTPUT AS #2
PRINT #1, UN1PRG$; CHR$(13)
540 TEMP$ = INPUT$(1, #1)
PRINT #2, TEMP$;
PRINT TEMP$;
IF TEMP$ = CHR$(3) THEN 545
GOTO 540
545 CLOSE #2
GOSUB 8100

CASE ELSE
END SELECT

550 CLS                                  'Output of File
GOSUB 8000
LOCATE 8, 35
PRINT "File Mode"
LOCATE 12, 1
PRINT "OUTPUT OF "; OUTFILE$
SELECT CASE OUTPUT$

CASE "Disk"                              ' Output of File to Disk

OPEN "TTTTEMP.TXT" FOR INPUT AS #2
OPEN OUTFILE$ FOR OUTPUT AS #3
560 TEMP$ = INPUT$(1, #2)
PRINT #3, TEMP$;
PRINT TEMP$;
IF TEMP$ = CHR$(3) THEN 565
GOTO 560
565 CLOSE #3, #2
GOSUB 8100
GOTO 590

CASE "UNIDEX 1"                          ' Output of Program to UNIDEX 1

OPEN "TTTTEMP.TXT" FOR INPUT AS #2
PRINT #1, "E"; RIGHT$(OUTFILE$, 2); "*"; CHR$(13)
570 TEMP$ = INPUT$(1, #2)
PRINT #1, TEMP$;
PRINT TEMP$;
IF TEMP$ = CHR$(3) THEN 575
```

```
GOTO 570
575 CLOSE #2
GOSUB 8100
GOTO 590

CASE "Printer"                              ' Output to Printer

OPEN "TTTTEMP.TXT" FOR INPUT AS #2
580 TEMP$ = INPUT$(1, #2)
LPRINT TEMP$;
PRINT TEMP$;
IF TEMP$ = CHR$(3) THEN 585
GOTO 580
585 CLOSE #2
DELAY = 2
GOSUB 8300

CASE ELSE
END SELECT

590 KILL "TTTTEMP.TXT"
GOTO 120

600 PRINT #1, CHR$(127)
CLOSE #1
END

1000 INPUT "File Name.Ext "; DISKFILE$
RETURN

1100 INPUT "UNIDEX 1 Program number "; UN1PRG$
UN1PRG$ = "0" + UN1PRG$
UN1PRG$ = RIGHT$(UN1PRG$, 2)
SELECT CASE UN1PRG$
CASE "00" TO "99"
UN1PRG$ = "P" + UN1PRG$
CASE ELSE
LOCATE 16, 5
PRINT "                                                    "
LOCATE 16, 5
GOTO 1100
END SELECT
RETURN

7000 '   SUBROUTINE TO CHANGE COMMUNICATION PARAMETERS
CLS
GOSUB 8000
LOCATE 7, 18
PRINT "PROCEDURE to change COMMUNICATION PARAMETERS"
LOCATE 10, 1
PRINT "          STEP 1:    With Unidex 1 powered down, remove the TFX c
PRINT "                     from Port A."
PRINT ""
PRINT "          STEP 2:    Connect the RS232 cable from your computer t
PRINT "                     Unidex 1 Port A"
```

```
PRINT ""
PRINT "          STEP 3:    Connect a jumper between pin 6 and pin 8 on
PRINT "                     I/O connector Port B."
PRINT ""
PRINT "          STEP 4:    Power up the Unidex 1 and press any key to"
PRINT "                     continue."
Time = 0
GOSUB 8100
CLS
GOSUB 8000
LOCATE 7, 32
PRINT "SETUP PARAMETERS"
OPEN "COM1:9600,E,7,1" FOR RANDOM AS #1
PRINT #1, ">>00"; CR$;
INPUT #1, A$
7010 PRINT #1, "PV"; CR$;
LOCATE 7, 15
PRINT "ORGINAL"
FOR I = 1 TO 11
INPUT #1, A$
LOCATE 9 + I, 15
PRINT A$;
NEXT I
GOSUB 8100
SELECT CASE N
CASE 110
A$ = "BR 03*"
CASE 150
A$ = "BR 05*"
CASE 300
A$ = "BR 06*"
CASE 600
A$ = "BR 07*"
CASE 1200
A$ = "BR 08*"
CASE 2400
A$ = "BR 10*"
CASE 4800
A$ = "BR 12*"
CASE 9600
A$ = "BR 14*"
CASE ELSE
A$ = "BR 14*"
END SELECT
PRINT #1, A$; CR$;
IF N$ = "7" THEN A$ = "WL 7*" ELSE A$ = "WL 8*"
PRINT #1, A$; CR$;
IF S$ = "1" THEN A$ = "SB 1*" ELSE A$ = "SB 2*"
PRINT #1, A$; CR$;
A$ = "PY " + LEFT$(P$, 1) + "*"
PRINT #1, A$; CR$;
FOR I = 1 TO 4
LOCATE 19 + I, 1
PRINT "
NEXT I
```

```
7100 LOCATE 22, 1
PRINT "Enter any other parameter changes
PRINT "Type E to exit or H for help"
LOCATE 22, 35
INPUT A$
IF UCASE$(LEFT$(A$, 1)) = "E" THEN 7110
IF UCASE$(LEFT$(A$, 1)) = "H" THEN GOSUB 7500
IF LEN(A$) < 4 OR LEN(A$) > 11 THEN 7100
PRINT #1, A$; CR$:
GOTO 7100
7110 CLS
GOSUB 8000
CLOSE #1
LOCATE 10, 1
PRINT "           STEP 5:     To activate parameter changes , power"
PRINT "                       down the Unidex 1."
LOCATE 13, 1
PRINT "           STEP 6:     Remove the jumper between pin 6 and"
PRINT "                       pin 8 of Port B of the Unidex 1."
LOCATE 16, 1
PRINT "           STEP 7:     Apply power to Unidex 1 and press a key"
PRINT "                       to continue."
LOCATE 20, 1
PRINT "Perform steps 5 and 6 before continuing this program."
GOSUB 8100
RETURN

7500 CLS
LOCATE 1, 20
PRINT "UNIDEX 1      HELP     setup  parameters"
LOCATE 3, 1
PRINT "        E. ACCEL/DECEL    (AD)    AD nnnn*       nnnn = 0 TO 9999
PRINT ""
PRINT "        F. DEVICE ADDRESS (DA)    DA nn*           nn = 2 TO 31"
PRINT "           Device address 0 and 1 not recommmend since 0 is used"
PRINT "           in setup mode and 1 is the default address."
PRINT ""
PRINT "        G. BOOT PROGRAM   (BP)    BP nn*           nn = 1 TO 99"
PRINT "           If nn = 00 then BOOT PROGRAM is disabled."
PRINT ""
PRINT "        H. START/STOP     (SS)    SS nnnnnn*       nnnnnn = 1 TO 1
PRINT ""
PRINT "        I. RAMP PROFILE   (RP)    RP x*      x = L (linear)  P (pa
PRINT ""
PRINT "        J. OUTPUT STATE   (OT)    OT x*      x = H (active high)
PRINT ""
PRINT "        K. MOTOR CURRENT  (EH)    EH x*      x = Y (YES) or N (NO)
PRINT ""
PRINT "        L. LOAD DEFAULT   (LD)    LD *       loads default values"
GOSUB 8100
RETURN


8000 CLS
LOCATE 4, 28
```

```
PRINT "UNIDEX 1 TEST PROGRAM"
LOCATE 5, 28
PRINT "----------------------"
RETURN

8100 LOCATE 23, 1
PRINT "Press any key to continue..."
T = 1
DO
T = T + 1
IF Time = 0 THEN T = 1
IF T = Time * 1000 THEN 8110
LOOP WHILE INKEY$ = ""
8110 RETURN

8200 DO
A$ = INKEY$
A$ = UCASE$(A$)
IF A$ = "Y" OR A$ = "N" THEN RETURN
LOOP

8300 FOR I = 1 TO DELAY * 1000
NEXT I
RETURN
```