Whedco

USER'S MANUAL

NSCL-ELECTRONIC

IMC Series
Intelligent Stepping Motor Controllers: A-Version

For Miniball Controller

IHC SERIES

INTELLIGENT STEPPING MOTOR CONTROLLERS

USER'S MANUAL

MODELS

IMC-103X-X-A
IMC-105X-X-A
IMC-113X-X-A
IMC-115X-X-A

Price

$10.00

Whedco Incorporated
Ann Arbor, Michigan U.S.A.

Pub.# 0029
Revision 0

Since this equipment can be applied in many diverse situations and in conjunction with equipment from other vendors, the user and those responsible for specifying this equipment must determine for themselves its suitability for the use intended. In no event, shall Whedco Incorporated be liable for loss of use, profit or consequential damages, or damage to other equipment, resulting from the use of this equipment.

The figures and examples in this manual are designed to demonstrate general concepts for the installation and maintenance of this equipment. The users should always verify interconnection requirements to and for other equipment as well as confirm installation and maintenance requirements for the specific application. In no case, shall Whedco Incorporated be liable for actual use based on the guidelines mentioned herein.

TABLE OF CONTENTS

TABLE OF CONTENTS
(cont.)

LIST OF **FIGURES**

LIST OF TABLES

LIST OF TABLES
(cont.)

SECTION 1.0

GENERAL DESCRIPTION

1.1 INTRODUCTION

The Whedco IMC Series Intelligent Stepping Motor Controllers provide single-device control of four-phase stepping motors. The completely self-contained unit includes the intelligence and power elements required for profile execution of point-to-point moves using motors with current requirements up to 5.0 Amperes per phase. An ASCII command set provides sasy profile programming and execution via any RS-232C, RS-422. or RS-485 device. Once programmed, the controller will execute a specified profile without connection to a serial device. Motion profile delimiters support definition of complex profiles in response to dwell times, position, triggers, and speed. Contiguous accels or decels can be specified for soft starts or stops. Up to eight units can be connected to one serial port for simple control of multi-axis systems.

In addition to a high performance bipolar chopper drive, the IMC Intelligent Motor Controllers provide computer control of the profile generation functions incorporating many features to enhance system performance. The unit is equipped with inputs for home, overtravel, two triggers, and an incremental encoder. Resident software handles all routines for implementation of these inputs at the discretion of the user. Upon power up, the state of the motor windings is returned to its last active condition so that holding torque is restored without moving the motor shaft if its location has not changed during power off. Microstepping is also available.

1.2 FEATURES

Self-contained unit including ramping profile generator, translator/ driver, and power supply.

Serial communication protocol compatible with RS-232C, RS-422, or RS-485 provides for easy programming and control of up to eight units from one serial port of a supervisory controller such as a programmable logic controller or a personal computer.

Battery-backed memory eliminates the need to reload profile program on power-up.

Stand-alone capability.

Incremental encoder interface available for position verification and correction.

Parallel inputs allow execution of up to six profiles using dedicated inputs and up to 31 using encoded lines. Serial link allows execution of up to 63 profiles.

State of motor windings saved on power down and restored on power up insuring that motors not moved during power off will not move on power up.

Two set point outputs-are available and can be programmed to turn on and off for up to 50 ranges.

Profile behavior can be programmed to respond to dwell times, position, speed, or triggers,

Contiguous accels and/or decels can be specified to achieve soft starts and stops.

Complex profiles can be defined and stored for subsequent execution using a single command.

Optional microstepping capability available.

Diagnostic mode to help in system installation.

SECTION 2.0

INSTALLATION

2.1 MOUNTING

2.1.1 Panel Mounting.

Whedco IMC controllers may be mounted vertically in enclosures using the four slotted holes in the controller end plates. The mounting panel or surface should be tapped to accept **#10-32** UNF machine screws, and locking hardware used. For optimum convection cooling, mount the unit with the heat sink fins in a vertical position as shown in Figure 2-1. Whedco does not recommend this mounting method in air-tight enclosures where the temperature of the inside air will exceed 55 degrees centigrade; in this case, through-wall heat sink mounting is recommended as discussed in Section 2.1.1. Refer to Appendix A, Figure A-l for unit dimensions.

Figure 2-1 Panel Mounting

(This Figure Intentionally Blank)

2.1.2 Through-Wall Heat Sink **Mounting**.

This mounting method is preferred when the temperature of the inside air may exceed 55 degrees centigrade or where other heat generating devices are in close proximity. Using Figure 2-3 as a guide, cut out the enclosure wall to the dimensions indicated. Drill ten 0.156" diameter holes in the locations indicated by the layout. **Mount** the unit in a vertical position to allow adequate air flow. If you desire to seal the controller heat sink to the enclosure wall, apply a thin bead of RTV sealant around the heat sink flange. Use **#6-32 x 1/2 UNC** screws to mount the controller using the tapped holes provided in the heat sink. Figure 2-2 illustrates this mounting configuration.

Figure 2-2 Through-Wall Heat Sink Mounting

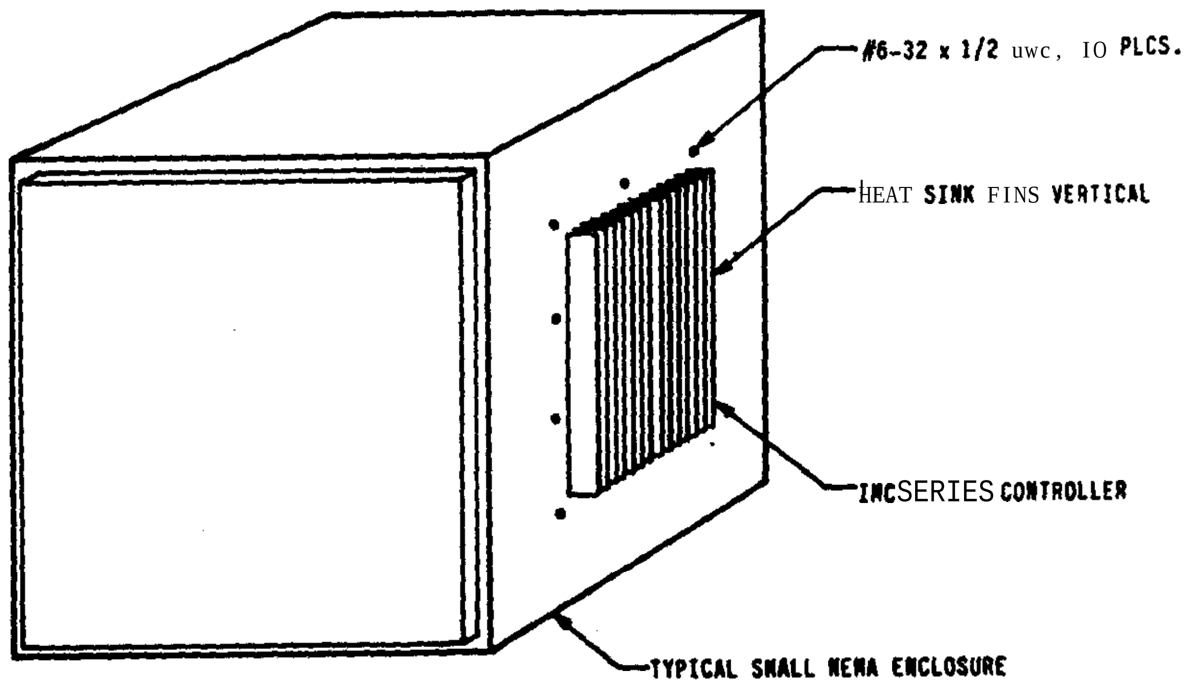


Figure 2-3 Layout for Through-Wall Mounting

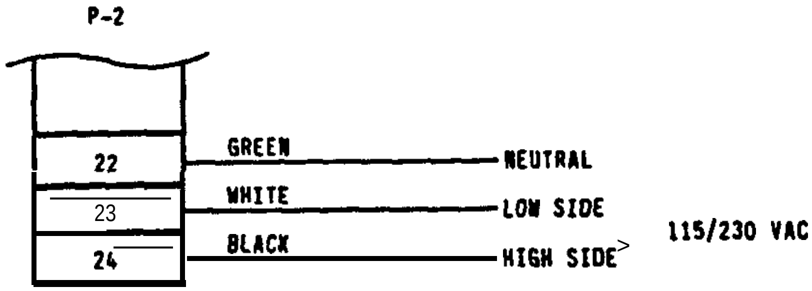
(This Figure Intentionally Blank)

2.2 CONNECTIONS

2.2.1 Power Input.

The IMC Controllers operate at 115 VAC or 230 VAC depending on the model selected. Appendix D. Module Configuration, explains the model numbering system used to identify the line voltage input. Verify the proper input voltage by checking the model number; severe damage could result from applying improper voltage. Figure 2-4 illustrates power line wiring.

Figure 2-4 Wiring to Power Line



2.2.2 Serial I/O.

The IMC Series Controllers are compatible with the RS-232C and RS-422 serial communication standards. The controllers can also operate on a party line as outlined in the RS-485 standard. However, if the party line approach is used, the echo mode of operation is not allowed (see Section 3.1). When using RS-232C, up to four units can be connected to one serial port. When using RS-422 or RS-485, up to eight units can be connected to one serial port.

If the Common Device Address will be used, the SYNC line of each controll connected to the same serial port must be tied together (see Section 2.3.3). To hook-up the serial link, refer to the figure on the next page appropriate to the communication protocol chosen.

The controllers transmit seven bits per character with odd parity, one "start" bit, and one "stop" bit for a total of 10 bits per character. Th serial device which will communicate with the controller must conform to this format.

Figure 2-5 Serial Connections for **RS-232C**

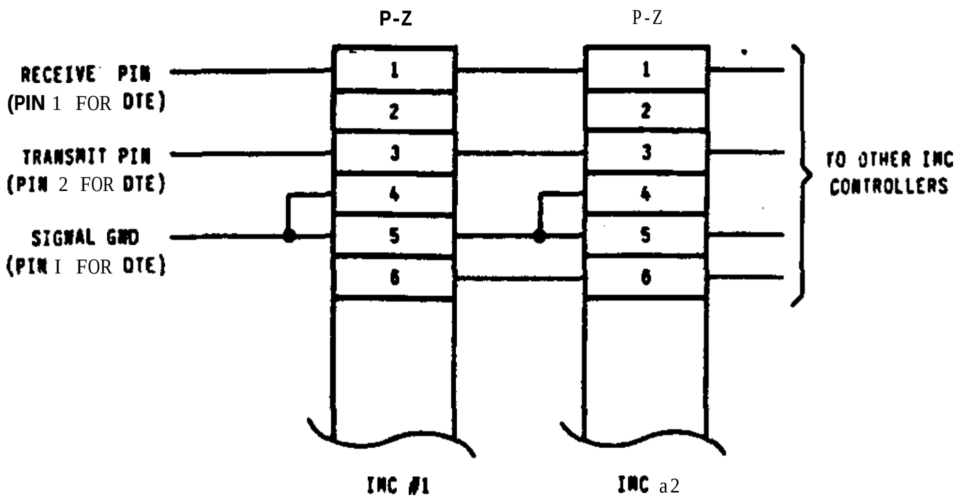


Figure **2-6** Serial Connections **for** RS-[redacted]2

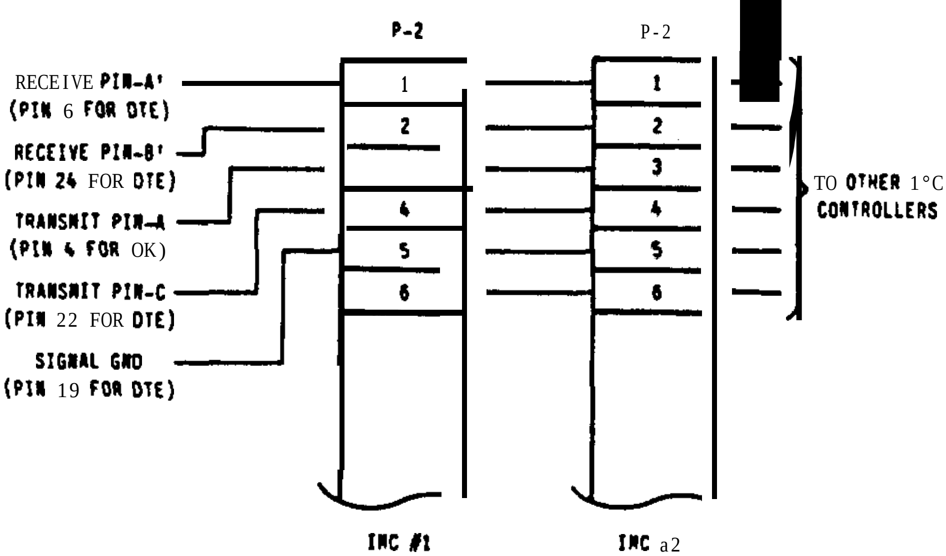
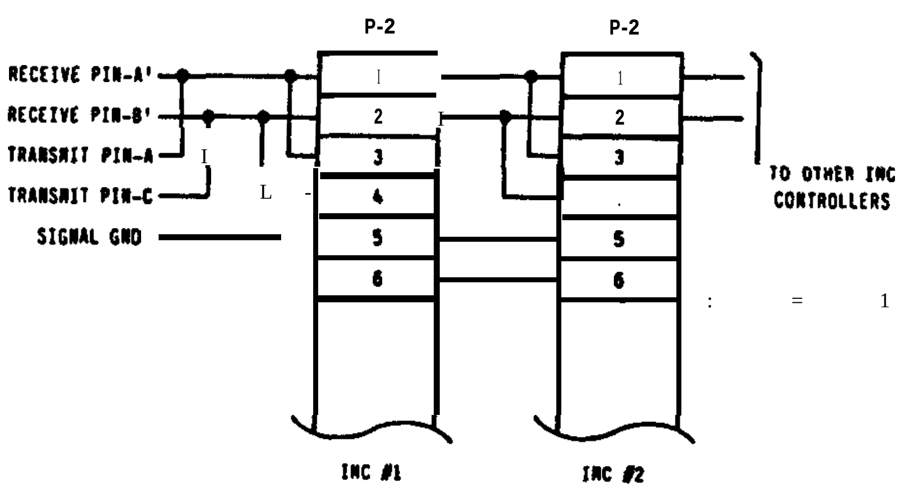


Figure 2-7 Serial Connections for RS-485 (party line)



2.2.3 **Incremental** Encoder.

The IMC controller can use position feedback from an incremental encoder with either a sine or a square wave quadrature output. The controller provides input lines for single-ended or differential output encoders as well as a power output line for 5 and 12 volt encoders. 15 volt encoders will require an external power supply. The index line can be optionally connected provided the encoder is so equipped. To connect the encoder, simply follow the diagrams given in Figures 2-8 and 2-9.

When an encoder is used, the "fwd" and "rev" LEDs indicate the encoder direction. If the encoder direction does not match the motor direction indicated by the FWD or REV motion command, the controller will fault when a motion command is initiated. To solve this problem, either switch the A and B channel connections to change the encoder direction or switch the connections to one of the motor windings to change the motor direction.

Figure 2-8 Interfacing to a Single Ended Encoder

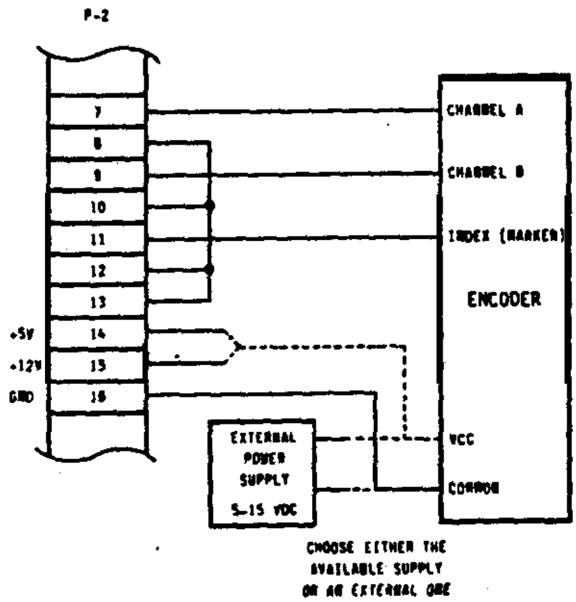
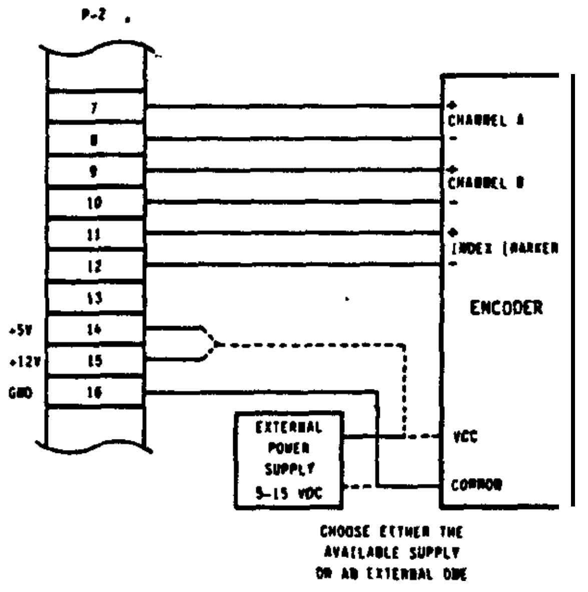


Figure 2-9 Interfacing to a Differential Encoder



2.2.4 Motor Drive Outputs

IHC Series Stepping Motor Controllers drive four or six lead motors from most manufacturers. Figure **2-10** illustrates connection to four lead motors, **Figure 2-11** for six leads.

Figure **2-10** Wiring to **4-lead** Motors

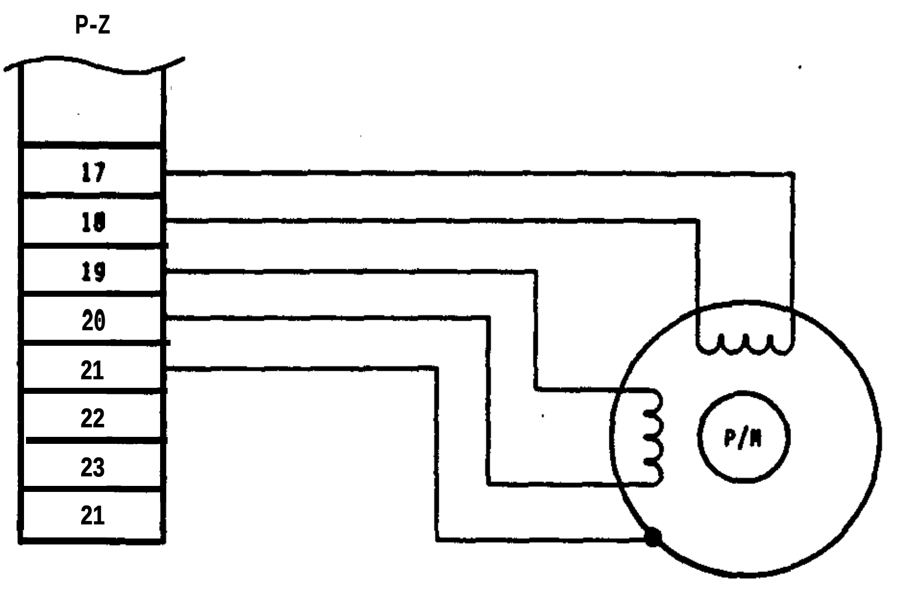
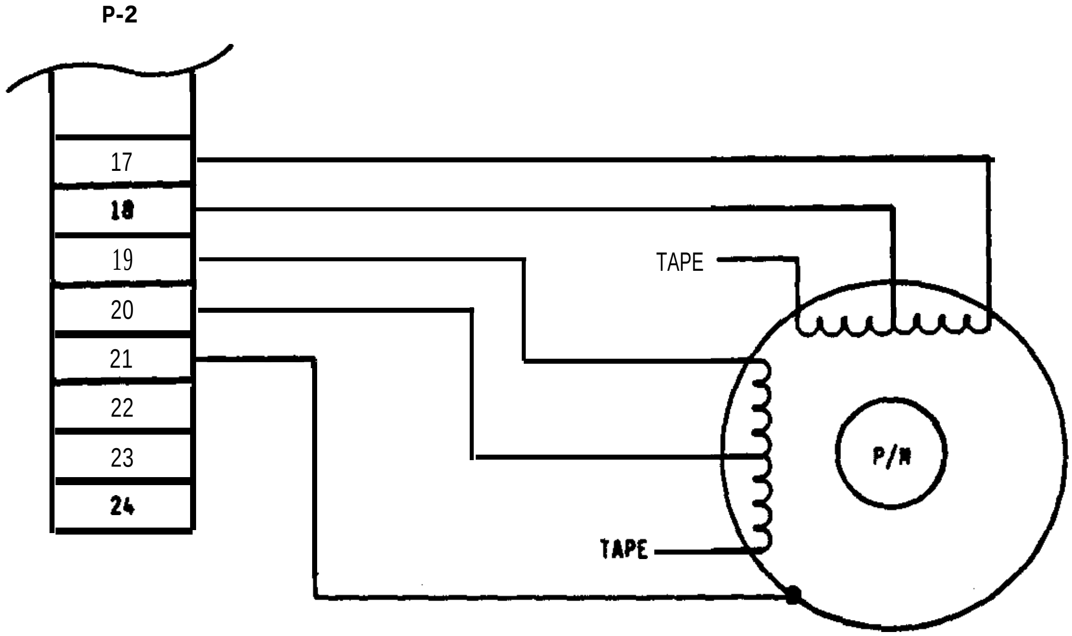


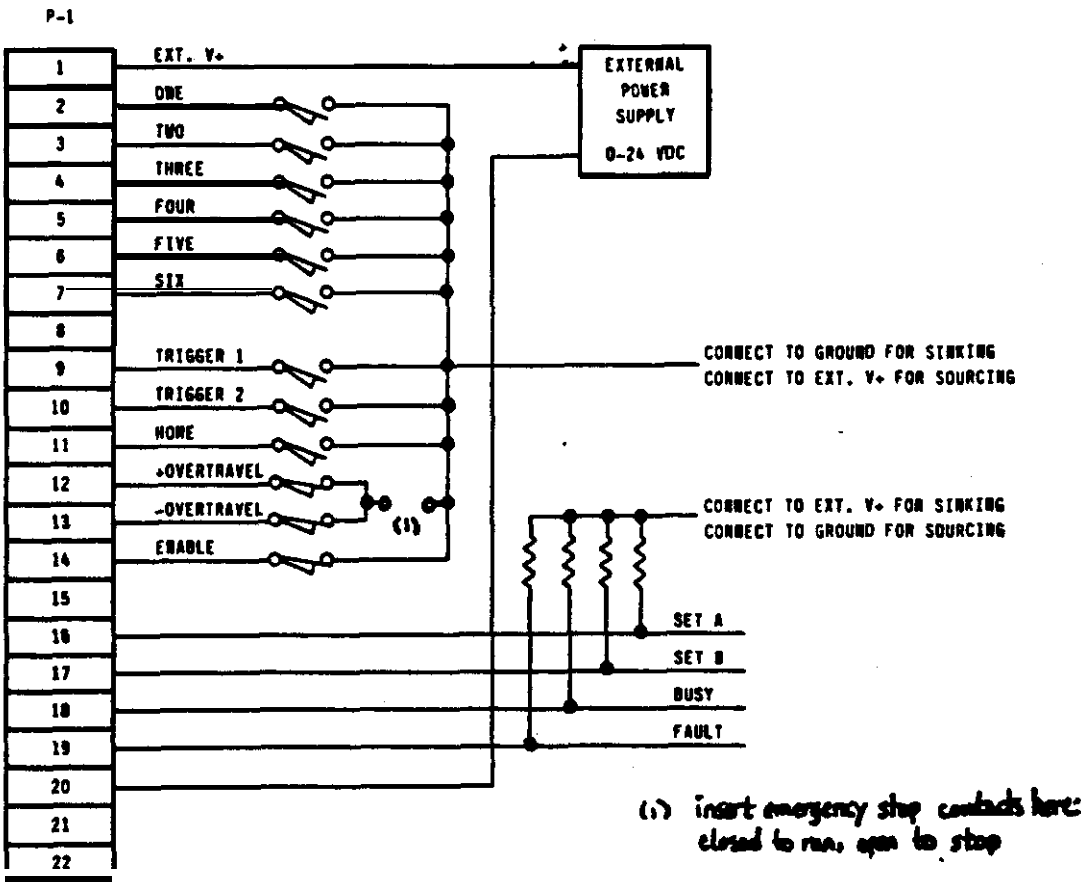
Figure 2-11 Wiring to **6-lead** Motors



2.2.5 Parallel I/O.

The IMC controller can have either sourcing or sinking outputs. The active state for the inputs and outputs on sourcing units corresponds to a high voltage level except for the **overtravel** inputs for which it is low. The active state for the inputs and outputs on sinking units corresponds to a low voltage level except for the **overtravel** inputs for which it is high. For either type of unit, the profile lines and the trigger inputs are sensitive to the inactive to active edge. While the home, overtravel and enable inputs are level sensitive. See Appendix A, Section.A.2, for a pin out of the I/O connector. The **overtravel** inputs **must** be inactive for the unit to work. If overtravel switches are not used, the inputs should be connected to V+ for sourcing units or grounded for sinking units. An emergency stop function can be implemented by connecting to the inputs through a set of contacts. See Figure 2-12 for typical connection details. he output lines can withstand current up to **200mA.** The input lines have a 2K resistor to ground **for** sourcing units or to V+ for sinking units. All the inputs and outputs can withstand voltages up to 24 **vdc.**

Figure 2-12 Connection of I/O Lines



2.3 FIELD ADJUSTMENTS

The IMC Controllers allow field adjustment of the factory settings for step size. motor current, serial address, communication protocol, and position feedback. Switches for changing these settings are **accessible** via the front panel by removing the protective metal plate. Make these adjustments only when power is not applied to the unit.

2.3.1 Step Size,

All IMC Stepping Motor Controllers operate in full or half-step mode. Controllers equipped with optional microstepping capability can **operate** at nominal fractional step sizes to **1/32** of a full step angle. Refer to Appendix D, Module Configuration, to confirm if the unit is equipped with microstepping. To adjust the step size. find D.I.P. switch **SW1** which is the uppermost switch behind the plate. Refer to Table 2-1 to set the desired step size. Note that only positions 1, 2, and 3 set step sire.

Table 2-1 Step Size D.I.P. Switch Settings (SW1)

Step Size	Models Available	Switch Positions 1 2 3	Nominal Step Angle*
Full Step	all	L L L	1.800 degrees
Half Step	all	R L L	0.900 degrees
1/4 Step	microstep	L R L	0.450 degrees
1/8 Step	microstep	R R L	0.225 degrees
1/16 Step	microstep	L L R	0.113 degrees
1/32 Step	microstep	R L R	0.057 degrees

"R" = right (open)
"L" = left (closed)

*Note: Step angle based on 200 step per revolution motor. Instantaneous step angle is motor and load dependent.

2.3.2 Motor Current.

For three amp units, the output current to the motor is adjustable in 0.20 Amp increments from 0.20 to 3.0 Amps, and for five amp units, in 0.33 Amp increments from 0.33 to 5.0 Amps. Refer to Appendix D. Module Configuration, to confirm maximum current output of the unit. The motor current is set with the last four positions of D.I.P. switch **SW1.** Refer to Table 2-2 to select the switch setting for the required motor current.

Table 2-2 Motor Current Switch Settings (SW1)

Switch Positions 5 6 7 8	3 Amp Units Current	5 Amp Units Current
R R R R	0.00 Amps*	0.00 Amps*
L R R R	0.20 Amps	0.33 Amps
R L R R	0.40 Amps	0.67 Amps
L L R R	0.60 Amps	1.00 Amps
R R L R	0.80 Amps	1.33 Amps
L R L R	1.00 Amps	1.67 Amps
R L L R	1.20 Amps	2.00 Amps
L L L R	1.40 Amps	2.33 Amps
R R R L	1.60 Amps	2.67 Amps
L R R L	1.80 Amps	3.00 Amps
R L R L	2.00 Amps	3.33 Amps
L L R L	2.20 Amps	3.67 Amps
R R L L	2.40 Amps	4.00 Amps
L R L L	2.60 Amps	4.33 Amps
R L L L	2.80 Amps	4.67 Amps
L L L L	3.00 Amps	5.00 Amps

"R" = right (open)
"L" = left (closed)

*Note: **Whedco recommends** the unit not be set to zero current.

2.3.3 Address Selection.

The address of the unit is set by the first three positions on D.I.P. switch **SW2.** The **address can** be set to **any** number in the range 0 through 7. To set the address, locate SW2 which is the lower-most switch behind the metal plate and refer to Table 2-3 to select the proper switch setting. Note that only switch positions 1, 2, and 3 sst the address.

Table 2-3 Address D.I.P. Switch Settings (SW2)

Unit Address	Positions 1 2 3
0	L L L
1	L L R
2	L R L
3	L R R
4	R L L
5	R L R
6	R R L
7	R R R

"R" = right (open)
"L" = left (closed)

If you wish to send identical information to multiple units on the **same** serial port, you **may** do so by sending **data** to the Common Device Address which is eight. However, all units cannot send back an acknowledgement simultaneously since this would **cause a** conflict on the line. Therefore, in this **case,** one unit must be selected as the "master" which will acknowledge **for** all units. Switch position 4 is used to select the master; the unit configured **as** the master should have this switch position to the right (open). All other units on the **same** serial port must be non-masters (position 4 must be to the left (closed)). **The** master must be able to know when all units connected to the serial port sre ready to acknowledge. **This** is accomplished by wiring the SYNC lines of all the units together.

2.3.4 Communication Protocol.

The IMC controllers can be configured to communicate in several ways. A separate position on D.I.P. switch SW2 selects each parameter. Table 2-4 summarizes the various options and switch positions.

Table 2-4 Communication Protocol D.I.P. Switch Settings (SW2)

Switch Position	Option	Setting with switch left (closed)	Setting with switch right (open)
5	communication format (1)	non-echo	echo
6	baud rata	1200	9600
7	serial standard	RS-422	RS-232C
8	profile lines	non-encoded	encoded

(1) See Section 3.1.

2.3.5 Encoder Feedback.

Position 4 on D.I.P. switch SW1 selects encoder feedback. If an encoder is used, position 4 must be to the right (open); if not used it must be to the left (closed). The encoder line density can be multiplied by 1. 2, or 4 to produce the actual position pulses used for feedback information. To set the position pulse multiplier, refer to Table 2-5 and use positions 9 and 10 of D.I.P. switch SW2.

Table 2-5 Position Pulse Multiplier D.I.P. Switch Settings (SW2)

Switch Positions 9	10	Position Pulse Multiplier
L	L	1
R	L	2
L	R	4

"R" = right (open)
"L" = left (closed)

SECTION 3.0

OPERATION

3.1 COMMUNICATION FORMAT

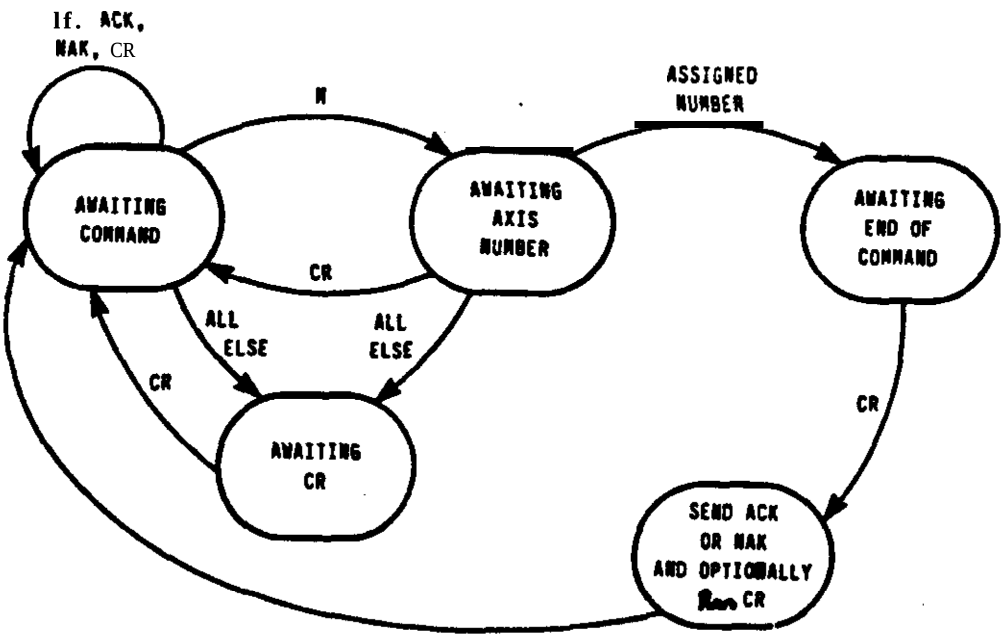
The IMC controllers communicate in one of two formats; non-echo and echo. When the controller is sat to the non-echo mode, the format is as follows:

MaXXXnn<CR>(LF)

where: **M** = motor axis is used to select IMC controllers from other devices on the saris1 line.
- **a** = axis address is a number from 0 to 8.
- **XXX** = command is one of the available commands.
- **nn** = parameter associated with the command in signed BCD format.
- **<CR>** = carriage return is the terminating character.
- **(LF)** = line feed is an optional character.

Upon receiving the carriage return, the selected axis sends back the ASCII control code 06h(ACK) or 15h(NAK) based on whether or not it accepted the command. The device communicating with the axis must wait for the command acknowledgement before sending another command string. If the command reqiures a number to be sent back, the controller will send an "R" for response followed by the signed BCD number terminated with a carriage return. The axis will then ignore (LF), (ACK), (NAK), and <CR> characters. It will respond to all other characters. If the character is other than an "M", the axis will wait for a <CR> before responding to characters again as depicted in Figure 3-1.

Figure 3-l Non-echo Mode State Diagram



When the controller is set to the echo mode, the format is as follows:

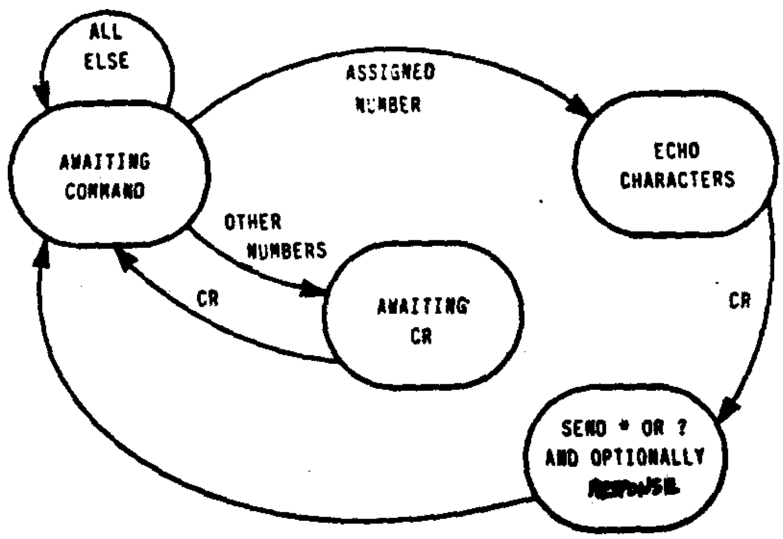
aXXXnn<CR>(LF)

where: a = axis address is a number from 0 to 8.
XXX = command is one of the available commands.
nn = parameter associated with the command in signed BCD format.
<CR> = carriage return is the terminating character.
(LF) = line feed is an optional character.

As each character is sent, the selected axis echoes the character, and after the <CR>, both a <CR> and (LF) are echoed even if an (LF) is not sent. The next character sent back from the axis is a "*" if the command is accepted or a "?" if it is not. The device communicating with the axis must wait for the command acknowledgement before sending another command string. If the command requires a response to be returned, the response is sent directly after the acknowledgement and is terminated with a <CR> (LF).

After receiving a <CR>, an axis will ignore all characters except numbers. If the number is not its assigned address or the common address, the axis will wait for a <CR> before responding to characters again as depicted in Figure 3-2.

Figure 3-2 Echo Hode State Diagram



In both modes, if a mistake is made when sending the command, type a “I” c the command line. If this character appears anywhere in the command lin except as the first character, the controller will reject the command.

3.2 OPERATING STATUS

3.2.1 Status Number.

To read the status number, send the Read Status Command **(mneumonic RS)**. **The** controller will return a number in the range 0 - 65535. This number **is** the result of adding **together** all the powers of two associated with each "true" status flag.- Table 3-1 lists the possible true conditions **for** each bit in the status number.

Table 3-1 Status Register Flags

NUMBER	BIT	**LABEL**	DESCRIPTION
1	B0	MOTOR STOPPED	motor is not moving
2	**B1**	ACCEL/DECEL	accelerating or decelerating
4	B2	DIRECTION	direction is positive (reset when direction is negative)
8	B3	BUFFER FULL	command input buffer is full
16	B4	HOME POSITION	motor is at user defined home
32	**B5**	HOME INPUT	home input detected
64	B6	SET POINT A	set point A is active
128	B7	SET POINT B	set point B is active
256	**B8**	+ 0. T.	positive overtravel input detected
512	**B9**	- 0. T.	negative overtravel input detected
1024	**B10**	SOFT. 0. T.	command position equals or exceeds user defined overtravel limits ACTION: motion stopped
2048	**B11**	**DEADBAND**	user set **deadband** exceeded
4096	B12	**SYSTEM** FAULT	position register overflow, excessive current **draw,** enable line inactive, or other system fault ACTION: bridge and pulses disabled
8192	B13	POSITION **LOST**	motor was not stopped when power failed ACTION: position set to zero
16384	B14	MEMORY **LOST**	memory checked incorrectly on power up ACTION: Cold Boot Command executed
32768	**B15**	PROFILE	predefined complex profile in process
	Note:		If flag = 1, then condition is true

The flags, Position Lost and Memory Lost, are set only when **power** is first applied or in response to the Cold Boot Command. The **Warm** Boot Command resets them. The Cold Boot Command will always cause both of the flags to be set. However, when power is applied, these flags will be set only if the conditions mentioned in Table 3-l are true.

3.2.2 Status LEDs.

Five LEDs on the front panel facilitate system diagnosis. The "xmit" LE lights when the unit transmits over the serial line. This aids in deter mining which unit is rasponsing when more than one is attached to the sa serial port. The LEDs labeled "fwd" and "rev" indicate motor direction , help determine the correct connection of the motor and encoder leads.

The "fault" LED lights whenever the controller detects a problem. The problem can be a system fault, a logic malfunction detected by the watch timer, over-current, the enable line becoming inactive, or the reapplica tion of power. Whenever this LED is active, the drive outputs are disabl to protect the motor and personnel. If the fault is not major, either th Warm Boot Command or shifting the enable line from the inactive to the active state will reset it. If the "ov-cur" LED is also on, then an over current condition was detected and should be investigated before re-enabling the unit. Short circuit protection is discussed in Section 3.9.

3.3 READING POSITION

To determine the motor position, give the Read Position Command (mneumoni RP). The controller will return the position as a signed BCD number. Th number will reflect the position of the motor within 1.5 milliseconds of the request.

In addition to the actual position of the motor, the controller maintains reference position to which, ideally, the motor's actual position is equa This position will always be equal to the motor's position if encoder feedback is not used. To request this value, give the Read Command Comma (mneumonic RC). The command position should be used when making profile calculations since the possible deviation of the motor position could thr the calculations off over time.

3.4 STAND ALONE MODE

The IMC Series Controllers are capable of stand-alone operation. In stan alone operation, the controller executes a previously defined profile as selected by the profile inputs. To allow stand-alone operation the Profi Enable Command (mneumonic PE) is sent along with a data value of one to s the Profile Enable Toggle. This will make the controller respond to the profile input lines. To disable this mode of operation, give the Profile Enable Command with a data value of zero to reset the toggle.

The profile input lines may be either non-encoded or encoded. When the inputs are non-encoded, (see Section 2.3.4), the unit executes the define profile corresponding to the profile line which becomes active. For example, if profile line one is toggled from inactive to active, then the controller will execute the profile defined as profile one. A profile wil not repeat due to a profile input being left in the active state. The profile input must be toggled from inactive to active to execute a profile

When the inputs are encoded, the defined profile corresponding to the binary representation on profile lines one through five is executed when profile line six becomes active. For example, if profile lines one, two, and four are active and profile lines three and five are inactive and profile line six is toggled from inactive to active, then the controller will execute the profile defined as profile 11.

3.5 ENABLE LINE

The enable input to the IMC Controllers controls the power output stage of the unit. When the line is inactive, the output drivers are disabled and the "fault" LED and output Line are active. When this line is taken from the inactive to the active state the controller executes a Warm Boot Command. The output stage will remain enabled until the enable line is made inactive or another fault condition occurs.

3.6 CONTROL CONSTANTS

Twelve constants regulate operation of the controller. These are set up only once during the initialization procedure, but can be changed later if necessary. Table 3-2 lists the control constants which are discussed in detail in the following sections. Appendix C, Table C-l, lists default values.

Table 3-2 Control Constants

Control Constant	Range	Load Command
- S. W. Overtravel	+/- 1,073,741,824	NO
+ S. W. Overtravel	+/- 1,073,741,824	PO
Encoder Ratio	256-65,535	ER
Wrap Position Register	0 or 1	WR
Deadband	0-32,767	DB
Deadband Fault	0 or 1	DF
Retry Number	0-255	RN
Position Correction	0 or 1	PC
Correction Time	0-255	CT
Backlash	0-127	BL
Stop-Start Rate	5-6,400	SS
Power-Save	0 or 1	PS

3.6.1 Software Overtravel Limits.

Software overtravel limits provide variable limits on motor motion using the position register. When encountered, the controller stops the motor, sets the Software Overtravel Flag, and prohibits further motion in that direction.

3.6.2 Encoder Ratio.

The encoder ratio number informs the controller of the ratio between the pulses which it generates and the pulses which it receives as position feedback. The number which should be entered can be determined by the following formula:

$$\frac{\text{Effective Motor Steps per Revolution}}{\text{Effective Position Pulses per Revolution}} \times 4096$$

For example, a 200 step motor which is microstepped at 16 microsteps per step with a 1000 line per revolution encoder and a position pulse multiplier of 2 will give an encoder ratio of 6554. Note that the ratio will always be 4096 when encoder feedback is not used.

When a ratio less than 4096 is used, a deadband value equal to or greater than the ratio between the encoder ratio used and 4096 will be necessary. The reason for this can be made clear by example. If a 200 step motor is used in half step mode with a 512 line encoder and a position pulse multiplier of one, the encoder ratio will be 3200. Now, suppose the motor and encoder line up precisely at position zero. The first step of the motor will make the encoder read one. The next step will make the encoder read two. The third step will make the encoder read three. However, the fourth step will make the encoder read five. It will be impossible for the controller to step the motor to a position which will make the encoder read four. So, a deadband value of at least one will be required.

3.6.3 Wrap Position Register.

The WR command sets or resets a flag which causes the controller to wrap the position register back to a lower value if its value becomes too large. This stops the position register from overflowing in applications where the motor is being slewed indefinitely in one direction such as on conveyors and pumps. The absolute position will be lost when the register is wrapped; so the flag should be set to zero in positioning applications. The command WR1 sets the flag to one which turns on wrapping and the command WRO resets the flag to zero which disables wrapping.

3.6.4 Disturbance Constants.

Six constants correct for system disturbances when using encoder feadbeck: Retry Number, Deadband, Deadband Fault, Position Correction, Correction Time, and Backlash. Deadband &fines the number of position pulses of arror which the controller will allow before it sets the Daadband Flag in the Status Register. The Deadband Fault Constant, if set to one, will cause the controller to fault when the Deadband Flag is set. If zero, the controller will not fault when the Deadband Flag is set. Retry Number indicates the number of times the controller should retry a movement command if it detects, through encoder feedback, that the motor is not following the pulse train. If the movement is still incomplete after attempting the specified number of retries, the controller will fault. The total number of retries since a Warm or Cold Boot is kept in a register which car be read by using the Read Total Retries Command (mneumonic RT).

The Position Correction Constant, if set to one, will cause the controller to automatically correct the position at the stop-start rate when the Deadband Flag is set. If zero, the controller will not output pulses to correct the position. The Deadband Fault Constant has precedence; so, it must be set to zero in order for correction to work. Correction Time specifies the length of time the controller will wait between corrections. The units are milliseconds. For example, the command CT10 will make the controller wait 10 milliseconds before making another correction if the position is still outside of the deadband. This constant allows the controller to compensate for the tendency of stepping motors to "ring" after making a step and also for wind-up in the mechanics. The default value is eight milliseconds which should work for most cases. But, if the motor "hunts" after a move, the constant should be increased. The Backlash Constant is the number of pulses in addition to the position error which the controller will output if the direction of the pulses changes during the position correction. This compensates for mechanical backlash in the system. It should be set to the measured value of backlash. If there is no measured backlash, then be sure to set it to zero because too much backlash correction only makes the "hunting" of the motor worse.

3.6.5 Stop-Start Rate.

The Stop-Start Rate determines the initial and final pulse train frequency. It may be set to any value in the range of 5 to 6400 pulses per second. This constant allows the controller to bypass resonant frequencies typical in stepping motors.

3.6.6 Power-Save.

When the Power-Save Constant is set to one, the controller will reduce the current to the motor windings to 30% of its set value when the motor is at rest. This reduces the energy used by the motor and, therefore, the heating of the motor as well. To use this constant, holding torque requirements should be about 30% or less of the maximum torque of the motor. When Power-Save is enabled, the controller automatically returns the current to its set value before moving the motor and brings the current back down to 30% when the motor stops. If Power-Save is not desired, the constant should be loaded with zero.

3.7 INITIALIZING PROCEDURE

Initialization consists of loading all necessary control constants as discussed in Section 3.6, clearing faults, and defining the starting position using a Set Position Command. The constants will only need to be loaded when the unit is programmed, following a Cold Boot Command, or when the Memory Lost Flag is set. The start position will need to be redefined if the Position Lost Flag is set or if there is reason to suspect the motor was moved when the power was off. The power failure fault will need to be taken care of each time the unit is turned on. During normal operation, the Warm Boot Command is usually the first command executed to put the unit into the operational state. This can be done by sending the WB command or by toggling the enable line from the inactive to the active state. Next, the starting position should be defined if necessary. This is done by using one of the Set Position Commands or a predefined profile using one of the commands as discussed in Section 4.3.3.

3.8 DIAGNOSTICS

The controller has a command called Diagnostics (mneumonic DG) which enables or disables the diagnostic feature of the controller. When diagnostics are enabled by sending DG1, the controller will output the header:

P1--P6 TTBIAB C POSITION

And, each time the controller detects a change on an input line it is moniterlng or changes a set point output line, it will send a string which identifies the state of each of the input and output lines, the position of the line in the string which changed, and the position of the motor. For example, the string:

000000 100001 7 15398

means profile lines 1-6 are inactive, trigger one is active, trigger two. home, index, and set point A are inactive, set point B is active, the seventh line (trigger one) changed state, and the motor is at position 15398. The controller will detect a change of state on an input line only if it is executing a command which uses that line. So, for the above example line to have been sent, the controller must have been executing one of the movement commands which use the trigger one line. The "X" command will cause the string to be sent immediately if the diagnostic mode is enabled. The diagnostic mode is disabled when the command DG0 is sent or when power to the controller is cycled.

3.9 MEMORY TRANSFER

The Memory Save (mneumonic MS) and Memory Load (mneumonic ML) Commands allow the entire memory contents of one IMC controller to be transfered to another IMC controller or to a disk-based computer for storage to be transfered later. Whedco will supply, for a nominal charge, a program to store the memory contents. The program runs under MS-DOS 2.0 or above. All transfers should be done with the enable line inactive, To do a direct transfer, each IMC unit which will be receiving the memory dump needs to be sent the ML command and the IMC unit which will be sending the memory dump needs to be sent the MS command. Then the transmit lines of each receiving IMC unit should be disconnected and the receive lines connected to the transmit line of the sending IMC unit. At this point, the sending IMC unit should be sent the ASCII control code 02h(STX). The transfer will now proceed. At the end of the transfer, each receiving unit will compare the sent checksum with its own checksum. If tha checksum is correct, the unit should have identical programming to the sending unit. If the checksum is incorrect, the unit will execute a Cold Boot Command.

3.10 SHORT CIRCUIT PROTECTION

All Whedco **IMC** Series Intelligent Stepping Motor Controllers are protected against short circuiting. "In the event that any motor output terminal is shorted to another or is shorted to ground, the controller will fault, set the Fault Flag in the Status Register, light both the over-current and the fault **LEDs** and disable the drive. To clear the fault, issue a Warm Boot Command. If the fault persists, disconnect power to the unit and recheck the motor wiring in accordance with Section 2.2.4. **Once** the short has been corrected, reapply power to the unit. If the **fault** still occurs, contact Whsdco Customer Service for assistance.

3.11 WARNINGS

When using an **IMC** controller, keep these items in mind:

1) The 5 volt supply provides 250 **mA** maximum and is internally limited.

2) The 12 volt supply provides 250 **mA** maximum and is internally limited.

3) If the motion profile does not last long enough to obtain maximum velocity, the ratio between the acceleration and deceleration rates cannot be more than **50:1** and the ratio of the deceleration to the acceleration rate cannot be greater than **5:1.**

4) Voltages of up to 400V dc can be present on some of the internal components. Do not attempt to troubleshoot or repair this unit with power applied.

SECTION 4.0

PROGRAMMING

4.1 INTRODUCTION

Programming the IMC Series Intelligent Stepping Rotor Controllers is a two step process. First, the pertinent parameters must be loaded. Then, one of the movement commands or a wait command and a movement command is sent. This sequence repeats until the profile programming is completed. The sequence is the same for programming a profile while it is happening or for programming a predefined profile. In the sections which follow, the parameters, movement and wait commands will be introducted. Then a section detailing the profile programming process will tie them together.

4.2 PARAMETERS

The IMC controllers provide complete programmability of motion profile parameters. The controllers save all entered parameters in battery-backed RAM. Therefore, unchanged profile parameters need not be reloaded between movement commands or after power failure. But, if a Cold Boot Command is sent or the Memory Lost Flag is set on power up, all the parameters will be lost. In the case of the Cold Boot Command, this is useful as it will put the controller back to the initial known state. However, the Memory Lost Flag being set indicates a problem with the unit (see Section 5.1).

To load any parameter, send the appropriate load parameter command with the signed BCD representation of the value. When the number is positive, the plus sign is optional. Leading zeros are also not required but will be accepted. To find the value of a parameter, send the command which loads the parameter with a ? in place of the value. This query suffix works for any of the load commands even the ones not discussed in this section. Table 4-1 shows the available parameters with their associated ranges and the commands used to load them.

Table 4-1 Motion Profile Parameters

Parameter	Range	Load Command
Repeat Number	1-32,000;0-infinite	LR
Speed	5-500.000 pul/sec	SP
Begin Speed	5-500,000 pul/sec	BS
Acceleration(l)	LOO-6.553.500 pul/sec^2	AC
Deceleration	100-6,553,500 pul/sec^2	DC
Delay Time	1-65,000 (.01-650.00sec)	DT
Incremental Move	+/- 1,073,741,824	IM
Absolute Move	+/- 1,073,741,824	AM
Incremental Start	+/- 1,073,741,824	IP
Absolute Start	+/- 1,073,741,824	AP

(1) When the Acceleration Rate is loaded, the Deceleration Rate is set to the same number. This saves one step when the rates are the same. To load different Accel/Decel Rates, load the Acceleration Rate first, then the Deceleration Rate.

4.3 MOVEMENT COMMANDS

The IMC controllers have three different types of movement commands, Run, Slew, and Set Position plus two commands for stopping motion while a motion is in process, Stop and Halt. The movement commands have modifiers associated with them which allow the commands to be executed contingent upon the condition specified by the modifier. These modifiers are: (1) Delay Time, (2) Trigger Input, (3) Home Input, (4) Index Input, (5) Begin Speed, (6) Absolute Start, and (7) Incremental Start. These modifiers provide the foundation for the execution of complex moves during which changes in speed, acceleration, deceleration, and/or dwell times may occur within a single profile.

4.3.1 Run Commands

The run commands cause the motor to go to a specified position. If the motor is stopped when the command is given, then the controller will execute a standard trapezoidal or triangular motion profile to move the motor from its starting position to the desired position. If the motor is moving when the command is given, the controller will do one of two things If the motor is going the wrong way, the controller will first stop it and then execute a trapezodial or triangular motion. If the motor is already going toward the &sired position, the controller will just finish the motion at the desired position.

The run commands always use the Acceleration, Speed, and Deceleration Parameters. The commands which move to an absolute position use the Absolute Move Parameter and the commands which move to an incremental position use the Incremental Move Parameter. The immediate commands are a special case of the normal absolute or incremental run commands in that they use the number sent along with the command instead of one of the move parameters. This number is sent with the command just like the numbers are sent with the load parameter commands. They can be positive or negative and are not loaded into the parameter register. These commands reduce the number of commands needed to make repeated moves to different positions.

The run commands can make use of the Delay Time Modifier and the Trigger Input Modifier. However, these modifiers will work only if the motor is stopped when the commend is given. If the motor is moving, the modifiers are ignored and a normal run command is executed instead. The Delay Time Modifier causes the move to be made after the time loaded into the Delay Time Parameter has elapsed. The Trigger Input Modifier causes the move to be made after either the T1 Input or the T2 Input is detected.

Table 4-2 lists the run commands available and Table 4-3 gives an example listing of a program which makes a trapezodial move.

Table 4-2 Run Commands

Mneumonic Fwd	Rev	Action
RAN		run to absolute position
RAI		run to absolute position immediately given
RAD		run absolute after delay time
RAT		run absolute after T1 input
RAU		run absolute after T2 input
RFN	RRN	run incremental
RFI	RRI	run incremental distance immediately given
RFD	RRD	run incremental after delay
RFT	RRT	run incremental after T1 input
RFU	RRU	run incremental after T2 input

Table 4-3 Run Example Program

```
*4SP2000<CR>          ; Load speed of 2,000 pul/sec
*4AC30000<CR>         ; Load acceleration of 30,000 pul/sec^2
*4AM10000<CR>         ; Load absolute move to 10,000
*4RAN<CR>             ; Give run absolute normal command
*
```

Note : Responses from the controller are underlined and the controller is assumed to be addressed at "4".

4.3.2 Slew Commands.

The slew commands cause the motor to run indefinitely at a given speed. They achieve the specified speed by accelerating or decelerating from the present speed. The commands always make use of the Speed Parameter and depending on whether the target speed is faster or slower than the motor's present speed, they make use of the Acceleration or Deceleration Parameter.

The slew commands can use almost all of the modifiers. The action of each modifier on the slew command will be described in the following sections.

Begin Speed: This modifier causes the slew command to execute after the motor has reached a speed greater or less than the speed specified in the Begin Speed Parameter. This modifier is most useful for generating profiles with variable acceleration or deceleration slopes.

Delay Time: This modifier can either cause the slew command to execute after the time specified in the Delay Time Parameter has elapsed 'or slew until the time has elapsed at which point a Stop Command is executed.

Home Input: This modifier causes the slew command to slew until the Home Input is detected or until the Home Input is no longer detected. In either case, a Stop Command is executed after the condition is met.

Trigger Input: This modifier has four different modes. The slew command can slew until the T2 Input is detected. It can execute after the T1 or the T2 Input is detected. Or it can execute after the T1 Input and slew until the T2 Input.

Absolute Start: This modifier causes the slew command to execute after the motor has achieved the position specified in the Absolute Start Parameter.

Incremental Start: This modifier causes the slew command to execute after the motor has moved the distance specified in the Incremental Start Parameter. This distance is relative to the position of the motor when the command is given.

Table 4-4 lists the slew commands available and Table 4-5 gives an example listing of a program which causes the motor to slew.

Table 4-4 Slew Commands

Mneumonic Fwd	Rev	Action
SFN	SRN	slew motor
SFG	SRG	slew beginning at speeds > begin speed
SFL	SRL	slew beginning at speeds < begin speed
SFD	SRD	slew until end of delay time
SFE	SRE	slew after end of delay time
SFH	SRH	slew until home input
SF0	SRO	slew until not home input
SFT	SRT	slew until T2 input
SFU	SRU	slew after T1 input
SFV	SRV	slew after T2 input
SFW	SRW	slew after T1 input until T2 input
SFA	SRA	slew beginning at absolute start position
SFI	SRI	slew beginning at incremental start pos.

Table 4-5 Slew Example Program

```
*4SP5000<CR>          ; Load speed of 5,000 pul/sec
*4AC50000<CR>         ; Load acceleration of 50,000 pul/sec^2
*4SFN<CR>             ; Give slew forward normal command
*
```

Note: Responses from the controller are underlined and the controller is-assumed to be addressed at "4".

4.3.3 Set Position Commands.

The set position commands set the position register to zero or to the position specified in the Absolute Start Parameter. The commands can execute immediately or can execute after a combination of the Home and/or Index Inputs are detected. When a command needs to find an input before executing, it will cause the motor to move at a slow predetermined speed until the input is found. The commands do not use the Speed or Acceleration Parameters. The commands will not execute if the motor is not stopped when it is given. Table 4-6 lists the set position commands.

Table 4-6 Set Position Commands

Mneumonic Fwd	Mneumonic Rev	Action
PIZ		immediately set at zero
PIA		immediately set at absolute start pos.
PFH	PRH	find home, then set at zero
PFI	PRI	find home, then set at absolute start
PFJ	PRJ	find index, then set at zero
PFK	PRK	find index, then set at absolute
PFB	PRB	find home & index, set at zero
PFC	PRC	find home & index, set at absolute start

Since the speed is very slow when a set position command is looking for an input, the amount of time for the Home Input to be found can be quite long if the motor is far away from the home position. A way to circumvent this problem is to move the motor to the Home Input with a slew command and then give a set position command. An example program which does this is given in Table 4-7. The example is for a home which is in the reverse direction of travel. For a home which is in the forward direction, interchange forward and reverse commands.

Table 4-7 Set Home Position Example

```
*4DE1<CR>                 ; Define profile one
*ED
4SP2000<CR>               ; Load speed of 2,000 pul/sec
*ED
4AC50000<CR>              ; Load acceleration of 50,000 pul/sec^2
*ED
4SRH<CR>                  ; Slew to Home Input
*ED
4SFO<CR>                  ; Back off of Home Input
*ED
4MW<CR>                   ; Wait for motor to stop
*ED
4PRB<CR>                  ; Find Home and Index. then set to zero
*ED
4ED<CR>                   ; End of profile definition
*4EX1<CR>                 ; Execute profile one
*
```

Note: Responses from the controller are underlined and the controller is assumed to be addressed at "4".

4.3.4 Stop and Halt Commands.

The Stop and Halt Commands terminate a motion in progress. These commands will always execute no matter what state the controller is in. The Stop Command (mneumonic ST) ramps down to zero speed at the rate specified in the Deceleration Parsmeter. The distance traveled during stopping will be dependent on the present speed and the deceleration rate. The Halt Command (mneumonic HT) immediately terminates the motion. Execution of this command can cause strain on the system (i.e., motor, controller, and drive components), so it should be used with caution. In addition, the motor may overrun the pulse train signal when halted at high speeds after which the motor position will not be known unless encoder feedback is used.

4.4 WAIT COMMANDS

The IMC controllers have two commands which cause the command interpreter to wait for a condition to be met before continuing the execution of commands. The first and most important of these is the Move Wait Command (mneumonic MW). This command causes the interpreter to wait until the Motor Stopped Flag is set before executing the next instruction. This command is often needed after run commands because the command interpreter finishes executing the run command long before the motor finishes the move. This means that, for example, two run commands in a row would blend together if a Move Wait Command was not put between them. There are many other cases where the motor must be stopped before the next command can be executed.

The second command is the Accel Wait Command (mneumonic AW). This command causes the interpreter to wait until the Accel/Decel Flag is reset before executing the next instruction. This command is most often used between the last slew command and the final run command in a complex profile (see Programming Example 3).

4.5 PROFILE PROGRAMMING

Programming a profile is simply a matter of sending the proper load parameter, movement, and wait commands. A profile can be executed while the commands are being sent in an interactive environment where the state of the motor as indicated by the Status Register and position determine the shape of the profile or a profile can be predsfined and stored for execution later. The same sequence of commands will work either interactively or in a predefined profile.

4.5.1 Interactive Profiles.

The controller has an input command buffer which can store up to 15 commands which are pending execution. This allows the device generating the commands to get ahead of the command interpreter. When the buffer gets full, the controller sets the Buffet Full Flag in the Status Register and will not accept anymore commands which need to be buffeted until the flag is reset. Not all the commands are buffered. Some of them execute immediately upon receipt. These commands are listed in Table 4-8. If the buffet needs to be emptied, us8 the Empty Buffer Command (mneumonic EB). Note that Cold Boot, Warm Boot, Stop, and Halt all empty the buffer. Programming Examples 1, 3, and 4 are examples of interactive profiles.

Table 4-g Commands Which Execute Immediately

Mneumonic	Name	Action
CB	Cold Boot	executes software reset
WB	Warm Boot	clears faults
ST	stop	ramps the motor to a stop
HT	Halt	stops the motor **immediately**
RS	Read Status	reports status
RP	Read Position	reports position
RC	Read Command	reports command position
RE	Read Repeats	reports number of repeats remaining
RT	Read Retries	reports number of retries
FC	Read FC	reports fault code
DE	Define Profile	stores profile
EB	**Empty** Buffer	empties buffer
MS	**Memory** Save	uploads memory contents
ML	**Memory** Load	downloads memory contents
?	Query	tells parameter value

4.5.2 Predefined Profiles.

The **IMC** controllers can store 63 profiles. Each profile has **100** bytes allocated to it for command storage which means a profile can store an average of 35 commands. Predefined profiles can be described using any of the Movement Commands plus the commands listed in Table 4-9.

Table 4-9 Non-Movement Commands Allowed in Stored Profiles

Mneumonic	**Name**	Action
ST	stop	ramps the motor to a stop
AW	Wait	wait for **Accel/Decel** Flag to be false
MW	Wait	wait for Motor Stopped Flag to be true
LR	Load Repeat	loads repeat times--
SP	load Speed	loads speed
AC	Load Accel	loads Acceleration Rate **(decel** also)
DC	Load **Decel**	loads Deceleration Rate
BS	Load BS	loads slew begin speed
DT	Load Delay	loads delay time
IM	Load **IM**	loads incremental move distance
AM	Load AM	loads absolute move distance
IP	Load IP	loads incremental start position
AF	Load AP	loads absolute start position
PTAxxB	Load **PTAxxB**	loads set point A xx beginning position
PTAxxE	Load **PTAxxE**	loads set point **A** xx ending position
PTAxxF	Forget **PTAxx**	forgets set paint A xx pair
PTBxxB	Load **PTBxxB**	loads set point B xx beginning position
PTBxxE	Load **PTBxxE**	loads set point B xx ending position
PTBxxF	Forget **PTBxx**	forgets set point B xx pair
ED	End Profile	ends profile definition
Ex	Execute Profile	executes profile
RM	Repeat	repeats previous command

To define or edit a profile, send the Define Profile Command (mneumonic DE) with the number of the profile to be worked on. Profiles can only be defined or edited when the motor is stopped. The controller will respond with the first command stored in the profile. Note that all profiles always have an End Profile Command (mneumonic ED). To enter a command above the command displayed, just send it. To delete the command displayed, use the Delete Command (mneumonic DEL). To move to the next command, use the Next Command (mneumonic X). To exit from editing the profile, us8 the Escape Command (mneumonic !). To end the profile definition even if there are more commands stored further down, us8 the End Profile Command.

To execute a stored profile, send the Execute Profile Command (mneumonic EX) with the number of the profile to be executed. The controller will set the Profile Flag in the Status Register as long as the profile is executing. None of the buffered commands will execute until the stored profile finishes executing. To stop a profile prematurely, send a Stop or Halt or Warm Boot Command. Note that the Halt and Warm Boot Commands will be harder on the system than the Stop Command.

A stored profile is allowed to call another stored profile by using the Execute Profile Command. The type of call performed is a call and return. Ibis means that on8 profile could potentially be the master profile which calls all the other profiles. There is no limit on the level of nesting allowed.

A stored profile can be repeated one of two ways. The first way is to repeat the Execute Profile Command just as the other movement commands can be repeated. This is explained in more detail in Section 4.7. The second way is to make the last command in the profile just before the End Profile Command be a Repeat Move Command (mneumonic RM). This will automatically cause the profile to be repeated the number of times specified in the Repeat Number Parameter when the profile is executed. The Repeat Move Command can also be put further up in the profile' defintion but only the commands from the beginning to the Repeat Move Command will be repeated.

4.6 SET POINTS

Two set point output lines are available on the IMC Series Controllers. Each has 50 beginning position-ending position pairs associated with it. To program a set point, make us8 of the PTyxxz Commands where "y" represents either the A or the B line, "xx" identifies the pair as a number "01" to "50", and "z" specifies the position as either a beginning or an ending point by either B or E respectively. The set point line will become active starting at the smaller of the two numbers and become inactive at the larger. Thus, for example, given the set point pair PTA01B-40/PTA01E250, the line will turn on at -40 and turn off beginning at 250. But, given the set point pair PTA01B250/PTA01E-40, the line will become active at 249 and become inactive immediately following -40. Note that the lines are direction sensitive so that the set point pair PTA01B0/PTA01E1000 will cause the output to be active only when the motor's direction is forward. To make it insensitive to direction, define the pair twice while reversing the numbers (i.e. PTA01B40/PTA01E50 and PTA02B50/PTA02E40).

The position pairs must be defined without gaps in the sequence. The sequence of position pairs 01, 03, and 04 cannot be defined. Rather define the sequence 01, 02, and 03. To disable a previously defined pair, just redefine it or send the command PTyxxF where the "F" in this command causes the controller to forget the set point pair "yxx". Keep in mind, when using this command, that there can be no gaps in the numerical sequence.

4.7 REPEAT MOVE COMMAND

By using the Repeat Move Command (mneumonic RM), the IMC controller will repeat a given movement command for the number of times specified in the Repeat Number Parameter. The command is used by first sending the command to be repeated. Then, send the Repeat Move Command. The command will actually be done one more time than the repeat number since, when the command is given, it will be done once and thenrepeated "n" times.

When repeating, none of the buffered commands will execute. If during the repeat process you wish to know the number of repeats remaining, use the Read Repeats Command (mneumonic RE). This command returns the number of repeats remaining. When this number reaches zero, the unit is finished repeating. However, if the controller is repeating an infinite number of times, then the unit will always return a repeat number of zero. If you wish to terminate the repeats prematurely, the Stop Command is the most appropriate. Halt and Warm Boot will also work but will be harder on the system,

4.8 PROGRAMMING EXAMPLES

These examples assume the controller has been initialized as described in Section 3.7 and is addressed at "4." Responses from the IMC controller are underlined.

4.8.1 Echo-Mode Programming Examples.

These examples illustrate programming the IMC controller using only an ASCII terminal.

Example 1:

This profile uses the external trigger inputs. The controller waits for an input on one of the trigger lines and then ramps up to a terminal velocity until an input occurs on the other trigger line at which point it ramps to a stop. A much larger deceleration rate is used to make the stop happen quickly after the second input. The profile is pictured in Figure 4-1 and the commands to run it are listed in Table 4-10.

Figure 4-1 First Programing Example: Motion Profile

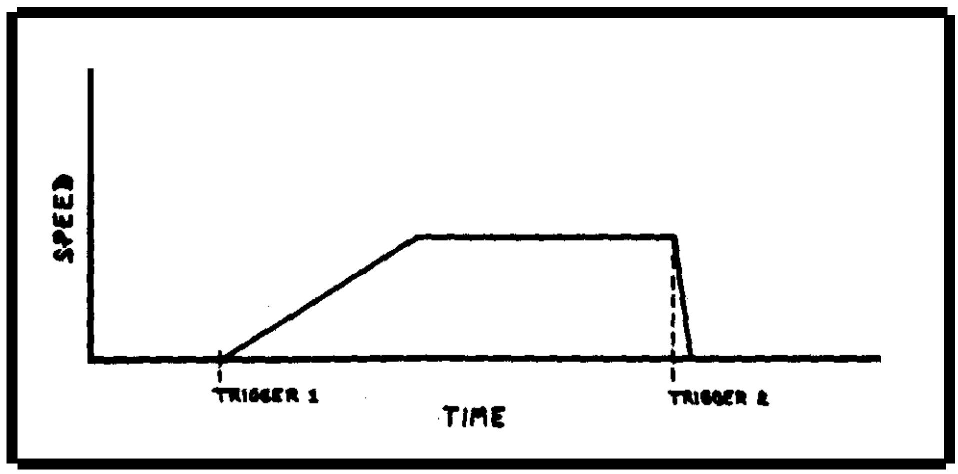


Table 4-10 First Programing Example: Listing

```
*4SP1000<CR>          ; Enter speed of 1.000 pul/sec
*4AC5000<CR>          ; Enter acceleration of 5,000 pul/sec^2
*4DC50000<CR>         ; Enter deceleration of 50,000 pul/sec^2
*4SFW<CR>             ; Give appropriate slew command
*
```

Example 2:

This profile shows how to predefine a profile. The profile requires the controller to ramp up to a high initial speed and run until trigger T1. Then it drops to a lower speed and runs to a specified position. After reaching the position, it rapidly changes direction and runs back to the starting position. This is repeated 4 times. Figure 4-2 illustrates the profile and Table 4-11 lists the command sequence.

Figure 4-2 Second Programming Example: Motion Profile

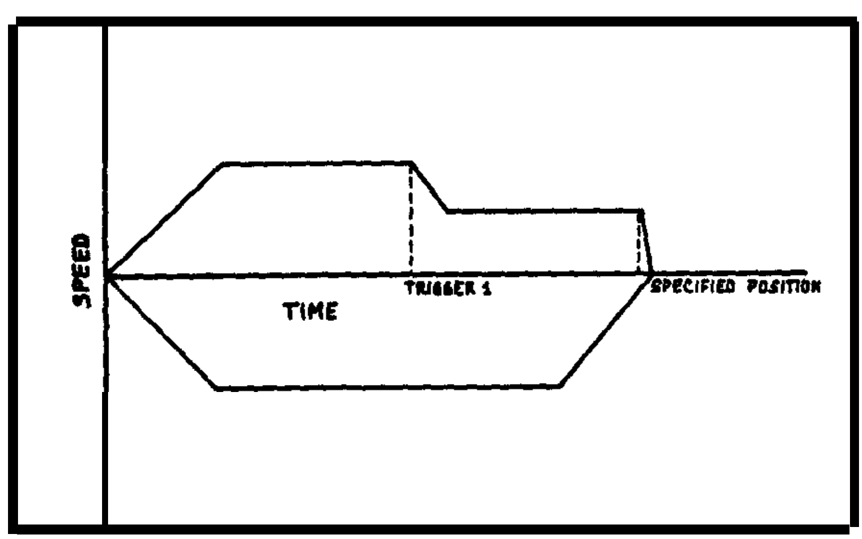


Table 4-11 Second Programming Example: Listing

```
*4DE1<CR>          ; Tell the unit to define profile 1
*ED
4SP1000<CR>        ; Enter initial speed
*ED
4AC5000<CR>        ; Enter acceleration rate
*ED
4SFN<CR>           ; Give Immediate Slew Command
*ED
4SP300<CR>         ; Enter second speed
*ED
4SFU<CR>           ; Give Slew After T1 Input Command
*ED
4AP11000<CR>       ; Enter absolute start position
*ED
4DC50000<CR>       ; Enter deceleration rate
*ED
4SP2000<CR>        ; Enter return speed
*ED
4SRA<CR>           ; Give Slew Starring At Absolute Position Command
*ED
4AW<CR>            ; Give Accel/Decel Wait Command
*ED
4AM0<CR>           ; Enter absolute move position
*ED
4AC5000<CR>        ; Enter old acceleration rate
*ED
4RAN<CR>           ; Give Run Absolute Command
*ED
4MW<CR>            ; Give Move Wait Command
*ED
4RM<CR>            ; Give Repeat Move Command
*ED
4ED<CR>            ; Profile definition is finished
*4LR4<CR>          ; Enter repeat number
*4EX1<CR>          ; Execute profile
*
```

4.8.2 Non-echo Mode Programming Examples.

These examples illustrate programming the **IMC** controller if a host computer is used to communicate with it.

Example 3:

This profile illustrates the techniques used in complex profiling. The controller ramps up to an initial speed at a slow ramp rate. Then at a specified position, the controller ramps up to a higher speed at a much faster ramp rate. Finally, it ramps to a stop at the given end position. Figure 4-3 shows the profile and Table 4-12 lists the command sequence.

Figure 4-3 Third Programing Example: Motion Profile

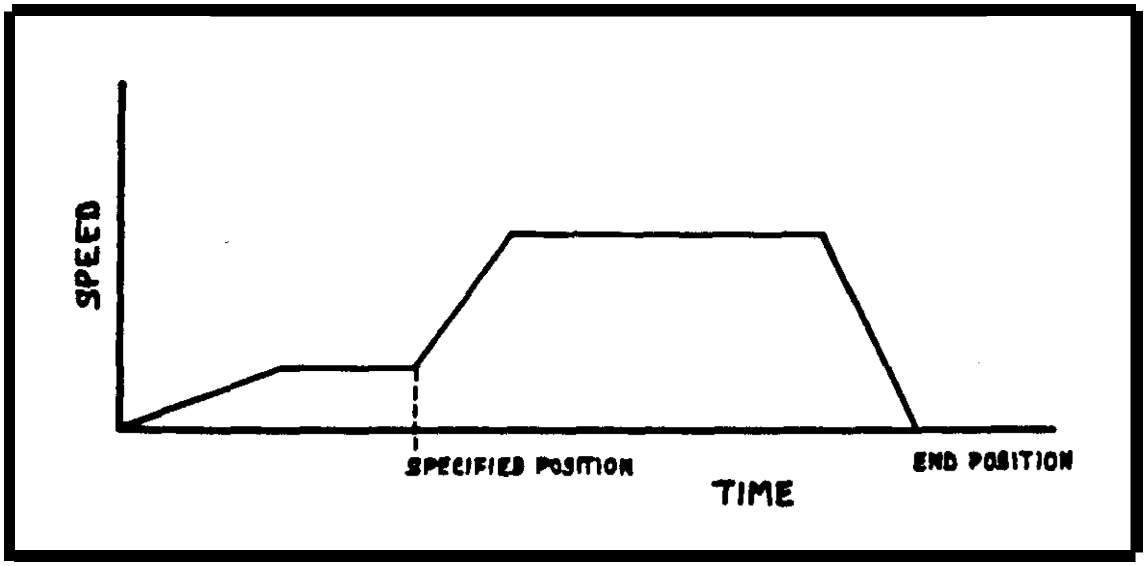


Table 4-12 Third Programming Example: Listing

```
M4SP200<CR>                 ; Enter initial speed
<ACK>M4AC5000<CR>           ; Enter acceleration rate
<ACK>M4AP400<CR>            ; Enter absolute start position
<ACK>M4AM6000<CR>           ; Enter absolute move position
<ACK>M4SFN<CR>              ; Give Immediate Slew Command
<ACK>M4SP1000<CR>           ; Enter next speed
<ACK>M4AC50000<CR>          ; Enter faster acceleration rate
<ACK>M4SFA<CR>              ; Give Slew Starting At Absolute
                               Position Command
<ACK>M4AW<CR>               ; Give Accel Wait Command
<ACK>M4RAN<CR>              ; Give Run Absolute Command
<ACK>
```

Example 4:

This example details a profile with variable accel and decal slopes where the **accel/decel** slopes softly start and stop. This is accomplished by using the Begin At Speed Slew Command. The numbers used in this profile are for a 1000 line encoder. Figure 4-4 illustrates the profile and Table 4-13 lists the command sequence.

Figure 4-4 Fourth Programming Example: Motion Profile

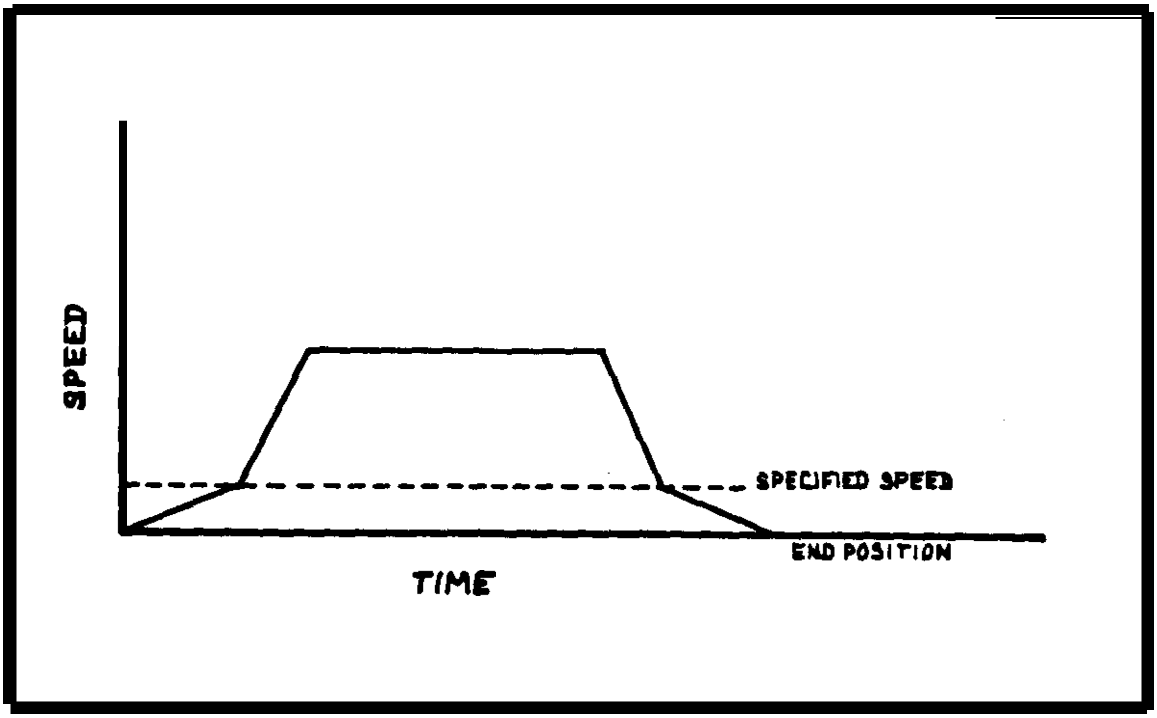


Table 4-13 Fourth Programming Example: Listing

M4SP2000<CR>	; Enter speed
<ACK>M4AC100<CR>	; Enter initial acceleration rate
<ACK>M4AP6950<CR>	; Enter absolute start position
<ACK>M4AM10000<CR>	; Enter absolute move position
<ACK>M4SFN<CR>	; Give Immediate Slew Command
<ACK>M4AC1000<CR>	; Enter faster acceleration rate
<ACK>M4BS500<CR>	; Enter begin speed
<ACK>M4SFG<CR>	; Give Slew Beginning At **>** Speeds Command
<ACK>M4SP200<CR>	; Enter stop-start rate as speed
<ACK>M4SFA<CR>	; Give Slew Starting At Absolute Position Command
<ACK>M4DC100<CR>	; Enter final deceleration rate
<ACK>M4SFL<CR>	; Give Slew Beginning At **<** Speeds Command
<ACK>M4RAN<CR>	; Give Run Absolute Command
<ACK>	

SECTION 5.0

MAINTENANCE

WARRING: Voltages of up to **400V** can be present on some of the components inside the controller. DO NOT attempt to repair this unit with power applied.

5.1 TROUBLESHOOTING

This section is intended to help the user diagnose the majority of problems which might occur with a" **IMC** Controller. If the problem is still unresolved after following these procedures, contact Whedco Customer Service for assistance (see Section 5.3).

Problem 1: Controller does nothing.

Symptoms: When power is applied to the unit, the "fwd" and "rev" **LEDs** do not flash and the "fault" LED does not light. There is no **+12** volts on pin 15 of connector TB-2.

Solution: Check the fuse and power input wiring (see Section 2.2.1). If the fuse is good and the unit **is** getting its proper supply voltage, then contact Customer Service.

Problem 2: Controller will not communicate.

Symptoms: Problem 1 does not exist. The **"xmit" LED** does not light when the unit is addressed. The **IMC** Controller does not return a response after a command is sent.

Solution: Check the serial wiring and serial format of your terminal to see that it conforms with the format which the Controller expects (see Section 2.2.2). Check the settings on D.I.P. switch SW2 for correct address, communication format, baud rate, and serial format (see Sections 2.3.3 **&** 2.3.4). Also, **review** the communication format given in Section 3.1 and note that a CR may be required to initialize the controller.

Problem 3: Controller loses memory.

Symptoms: On power-up, the Memory Lost Flag is set in the Status Register.

Solution: The controller should not have this problem under normal operation unless a Cold Boot Command is given. If a Cold Boot Command was not sent and the unit still loses memory, send it in for repairs.

'Problem 4: Discrete inputs do not function.

Symptoms: Controller does not respond to inputs.

Solution: Refer to Appendix D, Module Configuration, to confirm the type of unit, sourcing or sinking. For sourcing units, the active state of an input line, except for the overtravel lines, is a high voltage state. Make certain the inputs are switched from below 1.6 volts to above 3.4 volts. For sinking units, the active state of an input line, except for the overtravel lines, is a low voltage state. Make certain the inputs are switched from above 3.4 volts to below 1.6 volts. The diagnostic command can be used to see if the controller is detecting the inputs. Use the **"X"** command to print out the current state while toggling the input lines (see Section 3.8).

Problem 5: "fault" and/or **"ov-cur" LEDs** are lit.

Solution: The controller has a command which returns the reason for the fault. It is called the Fault Code command and its **mneumonic** is FC. The response will be in English if the echo format is used or a number if the non-echo format is used. See the appropriate section below for the response obtained.

Code 0 (Power failure) -
This is the response if the most recent fault was loss of power.

Code 1 (Force DAC) -
This is the response if the Force DAC Command was given. This command is used only for calibration.

Code 2 (Over-current) -
This fault indication is usually caused by a short circuit. Check Check the wiring to the stepping motor for shorts (see Section 2.2.4). Also **try** this test:

1) Disconnect the motor entirely;
2) power-up the unit and send a Warm Boot Command;
3) verify that the fault condition no longer exists:
4) turn off power to the unit.

If this test works, the problem lies in the motor or wiring. Otherwise, contact Customer Service.

Code 3 (Encoder lost) -
This fault occurs when the controller expects feedback pulses but does not get any. Check the encoder.

Code 4 **(Motor** stalled) -
This fault occurs when the motor does not follow the pulse train. It can be caused by loss of encoder feedback, too high an acceleration rate or speed, or improperly wired encoder (see Problem 6).

Code **5** (Deadband exceeded) -
The **Deadband** Fault Constant is set to one and the **Deadband** Flag was set.

Code 6 (Lost Enable) -
The enable input went inactive.

Code 7 (Position register overflow) -
The motor was commanded to go to a position past the bounds.

Code 8 (Unit functional) -
There is no fault condition.

Problem 6: Controller faults when the motor is commanded to run.

Symptoms: In this case, encoder feedback is being used. The controller accepts the Warm Boot **Command** to reset the fault condition. However, the controller faults again shortly after the motor starts to move. The FC command returns Motor stalled.

Solution: This indicates that the encoder feedback is backwards. Switch the A and B channel connections of the encoder (see Section 2.2.3).

Problem 7: Motor has insufficient torque.

Symptoms: The motor cannot keep the load from moving when at rest and/or the motor cannot accelerate the Load.

Solution: **Check** the setting of the current switch SW1 (see Section **2.3.2).** The current should be set (at most) for the maximum current rating of the motor. If the current is set to the maximum, check the motor sizing calculations. Also try disabling Power Save if it is being used.

Problem 8: Motor loses steps.

symptoms: Although motor has sufficient holding torque, the motor seems to have a jerky motion in low speed operation.

Solution: Increase the value of the stop-start rate parameter, or add mechanical damping if low speed operation is desired. Selecting a smaller step **size** may also solve the problem (half-stepping is the only alternate setting with **non-micro**-stepping units). Remember the pulse output frequency must be increased proportionally to maintain the same speed.

Problem 9: **Motor** turns in wrong direction.

symptoms: **"fwd"** and **"rev" LEDs** correspond to direction of command given. but motor turns in the opposite direction to that desired.

Solution: Reverse the connection of **one** of the motor windings.

5.2 FIELD REPLACEMENT

All IMC Series Intelligent Stepping Motor Controllers are mechanically interchangeable. In addition, the terminal blocks detach from the unit with the field wiring intact. For these two reasons, field replacement is very simple. However, when replacing a unit, care should be taken to make sure all field adjustments discussed in Sect. 2.3 are the same as the unit being replaced. It is especially important that the current setting not exceed the maximum rating for the motor in order to avoid motor damage.

5.3 SERVICE

If a Whedco product malfunctions, it may be returned using the following procedure:

a) Obtain a Return Merchandise Authorization (RMA)·number by calling the Customer Service Department at 313/665-5473. When packing the unit for shipment, enclose a purchase order for the repair with a description of the problem and the RMA number.

b) Ship to (freight prepaid):

Service Department
WHEDCO INCORPORATED
6107 Jackson Road
Ann Arbor, Michigan 48103

Note that products returned to Whedco without conforming to the procedure outlined above will not be accepted for repair.

All repairs are shipped F.O.B. Ann Arbor, Michigan. Products meeting the criteria for warranty service will be repaired or replaced free of charge. Products not covered by the warranty will be repaired for a time and materials charge against the customer's purchase order, not to exceed the price of a new unit. Repairs are warranted for thirty (30) days from date of shipment from Whedco. The repair warranty does not apply to new or different problems and/or failures which may occur.

Unless otherwise requested, repaired products will be shipped prepaid via United Parcel Service (UPS) ground service. Other methods of shipment such as UPS Blue Label, Federal Express, and Emery Air Freight can be arranged but are at the expense of the owner. On-site repair is available at a daily rate plus all travel expenses portal-to-portal from Whedco Incorporated.

5.4 WARRANTY

Whedco Incorporated warrants its products for one (1) year from the date of original shipment. During the warranty period, Whedco will repair or, at its option, replace without charge, any product returned to Whedco for repair. This warranty does not apply to any product which has been subject to misuse, accident, improper installation, and/or improper application, nor does it apply to any product which has been repaired or altered by anyone other than a Whedco authorized service representative. Whedco shall remain the sole arbitrator in all warranty claims.

There are no warranties which extend beyond those herein specifically given. **In** no event shall Whedco be Liable to the purchaser **for** Loss of use, profit, or consequential damages, or damages of any kind, including, but not limited to accidental loss or damage to other equipment, whether or not said equipment was properly used.

This warranty is in lieu of **any other** warranty, expressed, implied, or statutory, including, without limitation, any implied warranty of **merchant-ibility** or fitness for a particular purpose. No amendment of this warranty may be effected except in writing by a duly authorized representative **of Whedco** Incorporated.

APPENDIX A

SPECIFICATIONS

SPECIFICATIONS

A.1 PERFORMANCE

Table A-l lists performance specifications for the motion profile parameters:

Table A-l Performance Specifications

PARAMETER	RANGE
ACCELERATION	
Minimum	100 pulses/sec^2
Maximum	6,553,500 pulses/sec^2
Resolution	100 pulses/sec^2
DECELERATION	
Minimum	100 pulses/sec^2
Maximum	6,553,500 pulses/sec^2
Resolution	100 pulses/sec^2
SPEED	
Minimum	5 pulses/sec.
Maximum	500,000 pulses/sac.
Resolution	1 pulse/sec.
Accuracy	+/- 0.01%
STOP-START FREQUENCY	
Minimum	5 pulses/sec.
Maximum	6,400 pulses/sec.
Resolution	1 pulse/sec.
INCREMENTAL POSITION MOVE	
Minimum	1 position pulse.
Maximum	+/- 1,073,741,824 position pulses
Resolution	1 position pulse
ABSOLUTE POSITION MOVE	
Minimum	1 position pulse
Maximum	+/- 1,073,741,824 position pulse
Resolution	1 position pulse,
ABSOLUTE POSITION REGISTER	
Minimum	0 position pulses
Maximum	+/- 1,073,741,824 position pulses
Resolution	1 position pulse
POSITION DEADBAND	
Minimum	0 position pulses
Maximum	32,767 position pulses
Resolution	1 position pulse
DWELL TIME	
Minimum	0.01 seconds
Maximum	650.00 seconds
Resolution	0.01 seconds

A.2 ELECTRICAL

SERIAL I/O

RS-232C, RS-422, and US-485 compatible

ENCODER COMPATIBILITY

Two Channel Incremental, Sine or Square Wave Output
Outputs Single-ended or Differential
Output Voltage +/- 5, +/- 12, +/- 15 vdc
Maximum Encoder Frequency 500 kilohertz (position pulse frequency may be up to 1 megahertz)

USER INPUTS AND OUTPUTS

Outputs: 24 VDC (max.), 200 ma current (max.)
Inputs: 24 VDC (max.), 2k resistive load

Table A-2 P1 Connector Pin Outs

	Pin	Mnemonic	Signal Flow	Description
	1	External V+	in	Ext. supply for outputs
Profiles	2	One	in	Profile line one
	3	Two	in	Profile line two
	4	Three	in	Profile line three
	5	Four	in	Profile line four
	6	Five	in	Profile line five
	7	six	in	Profile line six
	8	Ground	return	Sgnl. gnd. & DC return
	9	Trigger 1	in	Trigger 1 input
	10	Trigger 2	in	Trigger 2 input
Inputs	11	Home	in	Home signal input
	12	+overtravel	in	Positive overtrvl. input
	13	-overtravel	in	Negative overtrvl. input
	14	Enable	in	Enable power amplifier
	15	Ground	return	Sgnl. gnd. & DC return
Outputs	16	Set Point A	out	Set point A output
	17	Set Point B	out	Set point B output
	18	Busy	out	Unit Busy Flag
	19	Fault	out	System Fault Flag
	20	Ground	return	Sgnl. gnd. & DC return
	21	Ground	return	Sgnl. gnd. & DC return
	22	Ground	return	Sgnl. gnd. & DC return

Table A-3 P2 Connector Pin Outs

	Pin	Mnemonic	Signal Flow	Description
Serial I/O	1	Tx A	out	Transmit A output line
	2	Tx B	out	Transmit B output line
	3	Rx A'	in	Receive A' input line
	4	Rx B'	in	Receive B' input line
	5	Ground	return	Sgnl. gnd. & DC return
	6	SYNC	in/out	Sync line of IMC units
Encoder Connections	7	Chnl. A+	in	Encoder Chnl. A+ input
	8	Chnl. A-	in	Encoder Chnl. A- input
	9	Chnl. B+	in	Encoder Chnl. B+ input
	10	Chnl. B-	in	Encoder Cbnl. B- input
	11	Index+	in	Encoder index + input
	12	Index-	in	Encoder index - input
	13	Encoder Tie	out	Encoder - input tie
	14	+5 VDC	out	+5 VDC encoder supply
	15	+12 VDC	out	+12 VDC encoder supply
	16	Ground	return	Sgnl. gnd. 6 DC return
Motor	17	Coil 1 +	out	Motor coil 1 connection
	18	Coil 1 -	out	Motor coil 1 connection
	19	Coil 2 +	out	Motor coil 2 connection
	20	Coil 2 -	out	Motor coil 2 connection
	21	Motor GND	return	Motor ground connection
Power	22	AC GND	return	Power input ground
	23	AC -	in	Power input -
	24	AC +	in	Power input +

TIMING

Timing specifications do not Include the serial transmission time.

Command Execution: 2.0 milliseconds average

Absolute Position Register Read: 0.10 milliseconds min.,
1.50 milliseconds max.

Absolute Position Register Update: 76 microseconds

Differential Execution Delay: 0-l.5 milliseconds
(difference in command execution delay between 2 units)

POWER SUPPLY **REQUIREMENTS**

Input Voltage: 90-130 VAC. SO-440 **Hz** for models **so** rated;
or **180-260** VAC, SO-440 Hz for models **so** rated
Input Current (max.): 5.0 amps; or 2.5 amps

MOTOR DRIVE OUTPUTS
Motor Current (max.): 0.20 - 3.0 Amps for units rated at 3.0 Amps
(adjustable in 0.20 Amp increments)
0.33 - 5.0 Amps for units rated et 5.0 Amps
(adjustable in 0.33 Amp **increments)**
Motor Excitation Voltage: **+65** VDC **+/-** 5% (factory set)
Motor Power (max.): 180 Watts for 3.0 Amp units
300 Watts for 5.0 Amp units
Step Size (based on 200 step/revolution motor):

Step Increment	Models Available	Nominal Step Angle
Full step	all	1.800 degrees
Half step	all	0.900 degrees
1/4 step	microstep	0.450 degrees
1/8 Step	microstep	0.225 degrees
1/16 Step	microstep	0.113 degrees
1/32 Step	microstep	0.056 degrees

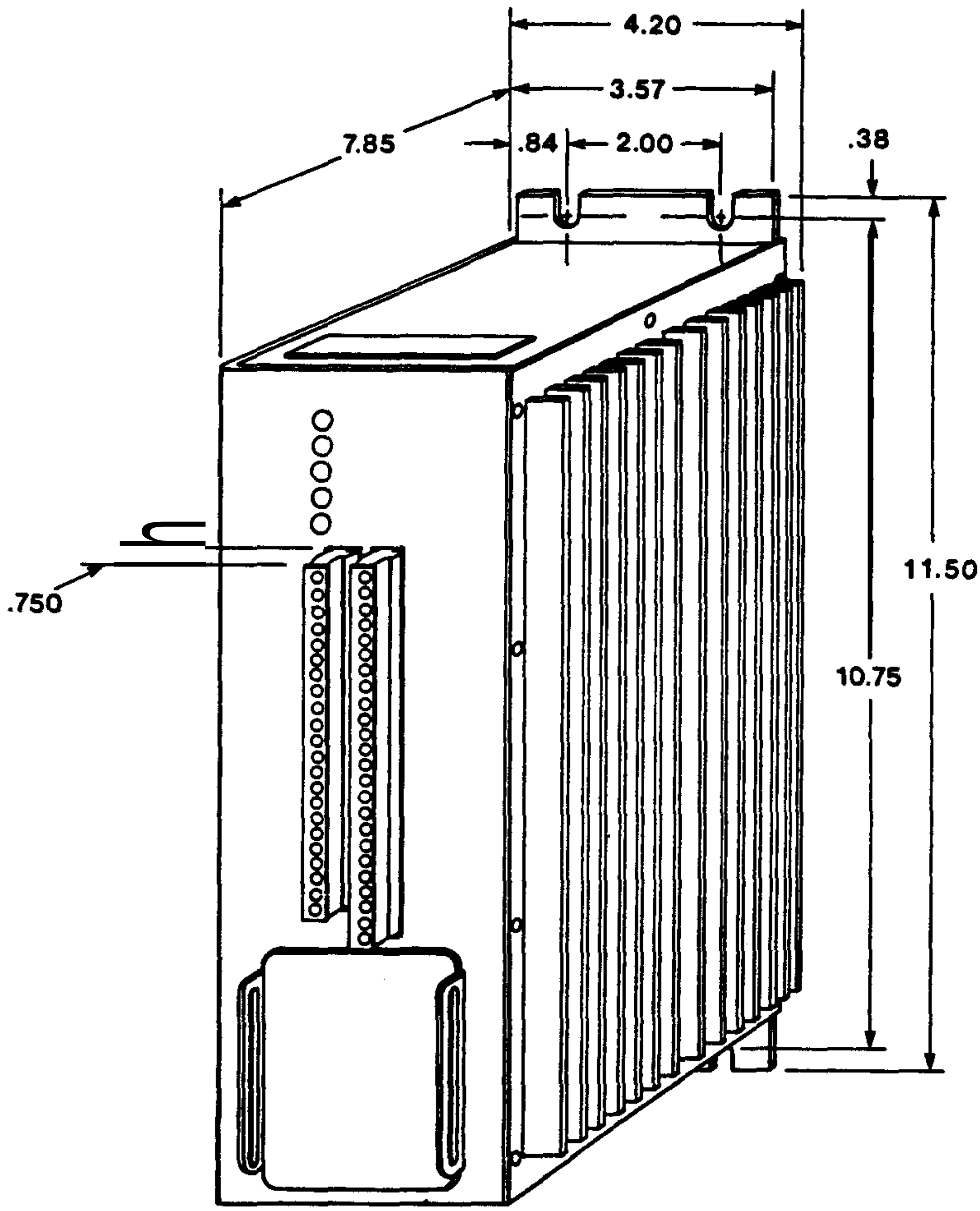
A.3 **ENVIRONMENTAL**

OPERATING **TEMPERATURE:** 0 to 55 degrees C. free air ambient
STORAGE **TEMPERATURE:** -40 to 80 degrees C.

A.4 MECHANICAL

WEIGHT: 8 lbs.
DIMENSIONS: see Figure A-l

Figure A-1 Unit Dimensions



APPENDIX B

SUMMARY OF COMMANDS

Table B-l Summary of Commands

Note: A "b" appended to a command indicates a Boolean value (0 or 1) which acts as a toggle for the function. (0 = off, 1 = on)

System Commands: (Not allowed in defined profiles)

Mnemonic	Name	Action
CB	Cold Boot	Executes software reset
WB	Warm Boot	Clears faults
RS	Read Status	Reports status
RP	Read Position	Reports position
RC	Read Command	Reports command position
RE	Read Repeats	Reports number of repeats remaining
RT	Read Retries	Reports number of retries
FC	Read FC	Reports fault code
EB	Empty Buffer	Empties command buffer
MS	Memory Save	Uploads memory contents from IMC
ML	Memory Load	Downloads memory contents to IMC
PEb	Profile Enable	Enables/disables parallel profile lines
?	Query	Tells parameter value
DGb	Diagnostics	Enables/disables diagnostic mode

Axis Initialization Commands: (Not allowed in defined profiles)

Mnemonic	Name	Action
NO	Negative O.T	Loads negative software O.T. limit
PO	Positive O.T.	Loads positive software O.T. limit
DB	Deadband	Loads deadband
WRb	Wrap Register	Enables/disables position register wrapping
SS	Stop-Start	Loads stop-start rate
RN	Retry Number	Loads retry number
ER	Encoder Ratio	Loads encoder ratio
CT	Correction Time	Loads correction time
BL	Backlash	Loads backlash
DFb	Deadband Fault	Enables/disables fault on deadband
PCb	Pos. Correction	Enables/disables automatic pos. correction
PSb	Power Save	Enables/disables power save feature

Editing Commands: (Not allowed in defined profiles)

Mnemonic	Name	Action
X	Next	Advances line pointer
DEL	Delete	Deletes current line
I	Abort	Aborts command line or ends edit

Table B-l Summary of Commands
(cont.)

Movement and Profile Definition Commands:

Mnemonic	Name	Action
DE	Define Profile	Begins profile definition
ED	End Profile	Ends profile definition
Ex	Execute Profile	Executes profile
ST	stop	Ramps motor to a stop
HT	Halt	Stops motor immediately
AW	Wait	Waits for Accel/Decel Flag to be false
MW	Wait	Waits for Motor Stopped Flag to be true
LR	Load Repeat	Loads repeat number
SP	Load Speed	Loads speed
AC	Load Accel	Loads acceleration rate (decel also)
DC	Load Decel	Loads deceleration rate
BS	Load BS	Loads slew begin speed
DT	Load Delay	Loads delay time
IM	Load IM	Loads incremental move distance
AM	Load AM	Loads absolute move distance
IP	Load IP	Loads incremental start position
AP	Load AP	Loads absolute start position
PTAxxB	Load PTAxxB	Loads set point A xx beginning position
PTAxxE	Load PTAxxE	Loads set point A xx ending position
PTAxxF	Forget.PTAxx	Forgets set point A xx pair
PTBxxB	Load PTBxxB	Loads set point B xx beginning position
PTBxxE	load PTBxxE	Loads set point B xx ending position
PTBxxF	Forget PTBxx	Forgets set point B xx pair
RM	Repeat	Repeats previous command

Slew Commands:

Mnemonic Fwd	Rev	Command Type	Action
SFN	SRN	Slew	Slew normal
SFG	SRG	Slew	Slew beginning at speeds > begin speed
SFL	SRL	Slew	Slew beginning at speeds < begin speed
SFD	SRD	Slew	Slew until end of delay time
SFE	SRE	Slew	Slew after end of delay time
SFH	SRH	Slew	Slew until home input
SFO	SRO	Slew	Slew until not home input
SFT	SRT	Slew	Slew until T2 input
SFU	SRU	Slew	Slew after Tl input
S W	SRV	Slew	Slew after T2 input
SFW	SRW	Slew	Slew after Tl input until T2 input
SFA	SRA	Slew	Slew beginning at absolute start position
SFI	SRI	Slew	Slew beginning at incremental start pos.

Table B-l Summary of Commands
(cont.)

Sat Position Commands:

Mnemonic Fwd	Rev	Command type	Action
PIZ		set	Immediately set at zero
PIA		set	Immediately set et absolute start pos.
PFH	PRH	set	Find home, then set at zero
PFI	PRI	Set	Find home, then set at absolute start
PFJ	PRJ	Set	Find index, then set at zero
PFK	PRK	set	Find index, then set at absolute start
PFB	PRB	Set	Find home & index, set at zero
PFC	PRC	set	Find home & index, set et absolute start

Run Commands:

Mnemonic Fwd	Rev	Command type	Action
RAN		Run	Run to absolute position
RAI		Run	Run to absolute position immediately given
RAD		Run	Run absolute after delay time
RAT		Run	Run absolute after Tl input
RAU		Run	Run absolute after T2 input
RFN	RRN	Run	Run incremental
RFI	RRI	Run	Run incremental distance immediately given
RFD	RRD	Run	Run incremental after delay
RFT	RRT	Run	Run incremental after Tl input
R N	RRU	Run	Run incremental after T2 input

APPENDIX C

DEFAULT VALUES

Table C-l Default Values

PARAMETER	VALUE
Repeat Number	1
Speed	0 pulses/sec.
Acceleration	0 pulses/sec^2
Deceleration	0 pulses/sec^2
Begin Speed	0 pulses/sec.
Delay Time	0 seconds
Incremental Move	0 pulses
Absolute Move	0 pulses
Incremental Start	0 pulses
Absolute Start	0 pulses
- S. W. Overtravel	-1,073,741,824 pulses
+ S. W. Overtravel	+1,073,741,823 pulses
Stop-start Rate	200 pulses/sec.
Deadband	0 pulses
Retry Number	0
Encoder Ratio	4096
Correction Time	8
Backlash	0
Deadband Fault	0
Position Correction	0
Power Save	0
Profile Enable	0
Wrap Position	0
Diagnostics	0

APPENDIX D

HARDWARE CONFIGURATION

HARDWARE CONFIGURATION

The IMC Stepping Motor Controllers offer flexible hardware configuration in a combination of permanent and-field-configurable settings.

D.l Permanent Settings

The catalog numbering system is used to specify permanent settings as described below:

Table D-l Catalog Number Designation

CATALOG NUMBER **FORMAT** I M C - 1 X X X - X - X

product series prefix:

IMC - Intelligent Motor Controller

type of motor:

1 - stepper

motor step size capability:

0 - full/half

1 - full/half/micro

maximum output current to motor:

3 - 3.0 Amp

5 - 5.0 Amp

parallel output current **type**:

0 - sinking

1 - sourcing

input voltage range:

1 - 90-130 **vac**

2 - 180-260 **vac**

version:

blank - first (not recommended for **new** installations)

A - second

EXAMPLE CATALOG NUMBER — IMC-1150-t-A

product series prefix:
IMC = Intelligent Motor Controller

type of motor:
1 = stepper

motor step size capability:
1 = full/half/micro

maximum output current to motor:
5 = 5.0 Amp

parallel output current type:
0 = sinking

input voltage range:
1 = 90-130 vac

version:
A = second

D.2 Field-Configurable Hardware Settings

D.I.P. switch adjustment of field-configurable hardware settings is discussed in detail in Section 2.3 of this manual. The table below summarizes possible field adjustments and the factory settings for them:

Table D-2 Field-Configurable Hardware Settings

Setting Type	Switch Number	Switch Positions	Factory Setting
Motor Step Size	1	1-3	full
Encoder Feedback	1	4	none
Motor current	1	5-8	max.
Address Selection	2	1-3	4
Master for Common Addressing	2	4	off
Communication Format	2	5	echo
Baud Bate	2	6	1200
Serial Standard	2	7	RS-232
Profile Line Format	2	8	non-encoded
Position Pulse Multiplier	2	9-10	1X

Figure D-l Front Panel Control Assignments

(This Figure Intentionally Blank)

APPENDIX E

SCHEMATICS

Schematics **are** available to bona fide customers who feel they are required for equipment use and maintenance. To receive copies of schematics and parts lists, the customer must submit the following non-disclosure agreement. Upon receipt of these items, we will forward schematics.

Users obtaining schematics for maintenance purposes should be advised that any user adjustment or repair to the unit will void the warranty.

DATE: ______________________

SUBJECT: Schematic & Parts List

CATALOG NO.: ______________

COMPANY: ______________________

DESCRIPTION: ______________

ADDRESS: ______________________
____
____

SERIAL NO.: ______________

ATTN: ______________________

PROPRIETARY RIGHTS & NON-DISCLOSURE AGREEMENT

1. Whedco Incorporated considers all data, know-how, and other information (collectively, the "INFORMATION") submitted to the undersigned to be confidential and proprietary to Whedco Incorporated. Reproduction of the INFORMATION by the undersigned, its directors, officers, agents, or employees (collectively, the UNDERSIGNED) or any other third party or its disclosure to any other third party is prohibited.

2. The UNDERSIGNED recognizes the proprietary rights of Whedco Incorporated in and to the INFORMATION and the confidential nature of the INFORMATION and agrees to take every precaution to safeguard and treat the INFORMATION as confidential and to take appropiate action by instruction, agreement or notice to its directors, officers, agents and employees of the confidential and proprietary nature of the INFORMATION submitted by Whedco Incorporated.

3. The UNDERSIGNED further agrees that it will not disclose, publish, or reveal any of the INFORMATION which it receives or has received from Whedco Incorporated, other than for the purpose for which such INFORMATION has been disclosed, except with the specific written authorization of an officer of Whedco Incorporated.

4. The UNDERSIGNED further agrees that it will not disclose, publish, or reveal any of the INFORMATION received from Whedco Incorporated to any other party whatsoever, except with the specific prior written authorization of an officer of Whedco Incorporated.

5. The UNDERSIGNED further agrees to indemnify Whedco Incorporated against any losses sustained by Whedco Incorporated, including counsel fees, by reason of breach of any portion of this Agreement by any director, officer, employee or agent of the UNDERSIGNED.

Witness:

____

Acknowledged: ______________________

Authorized Officer: ______________

Title: ______________________

Date: ______________________