Compumotor

LX Indexer/Drive User Guide

Compumotor Division
Parker Hannifin Corporation
p/n 88-008896-01 A

Parker

User Guide Change Summary

The following is a summary of the primary changes to this user guide since the last version was released. This user guide, version 88-008896-01A, supersedes version 88-008896-01Y.

When a user guide is updated, the new or changed text is differentiated with a change bar in the right margin (this paragraph is an example). If an entire chapter is changed, the change bar is located to the right of the chapter title.

NOTE: All encoder-related information has been deleted. The LX does not support closed loop functions.

Chapter 1.
Introduction

Chapter 1 was changed to include the added features provided by the new Accelerometer (-AC) option for the L20 forcer.

Chapter 2.
Getting Started

Changes to Chapter 2 are summarized as follows:

- Accelerometer (-AC) option set up and connections
- Ship kit changes

Chapter 3.
Installation

Changes to Chapter 3 are summarized as follows:

- All optional DIP switch settings moved to Chapter 5
- New drive and platen mounting information
- New forcer cable mounting information
- New assembly instructions for -RFKC coalescing filter kit
- Updated I/O connection information

Chapter 4.
Application Design

Changes to Chapter 4 are summarized as follows:

- Effects of Accelerometer option on resonance and settling
- New platen thermal expansion information
- Updated sequence programming information
- Updated tuning procedures

Chapter 5.
Software Reference

Chapter 5 was changed to remove the **LCCW**, **LCW**, **LE**, **MC**, **RIFS**, and **SP** commands, plus all encoder and jog-related software commands. It was also changed to add the **IS**, **OS**, **SN**, **SS**, and **XRP** commands. The software-related information in this user guide is based on LX software version Y5. For a list of all commands, refer to the Appendices section.

Chapter 6.
Hardware Reference

Changes to Chapter 6 are summarized as follows:

- Added Accelerometer specifications/settings
- Added default and optional DIP switch settings
- Changed motor current DIP switch settings
- Updated force/speed curves for each forcer
- Updated system specifications

Chapter 7.
Maintenance & Troubleshooting

Changes to Chapter 7 are summarized as follows:

- Updated list of suggested spare parts
- Updated platen maintenance information
- Updated troubleshooting methods
- Added procedures for returning the system for repairs

Table of Contents

List of Figures

List of Tables

How To Use This Manual

The manual is designed to help you install, develop, and maintain your system. Each chapter begins with a list of specific objectives that should be met after you have read the chapter. This section is intended to help you find and use the information in this manual.

Assumptions

This user guide assumes that you have the skills or fundamental understanding of the following information.

- IBM (or IBM-Compatible) computerexperience
- Basic electronics concepts (voltage, switches, current, etc.)
- Basic motion control concepts (torque, velocity, distance, force, etc.)
- Basic serial communication concepts (specifically RS-232C

With this basic level of understanding, you will be able to effectively use this manual to install, develop, and maintain your system.

Contents of This Manual

This user guide contains the following information:

Chapter 1: Introduction

This chapter provides a description of the product and a brief account of its specific features.

Chapter 2: Getting Started

This chapter contains a detailed list of the items you should have received with your LX Series shipment. It will help you to become familiar with the system and ensure that each component functions properly. You will learn how to configure the system properly in this chapter.

Chapter 3: Installation

This chapter provides instructions for you to properly mount the system and make all electrical and non-electrical connections. Upon completion of this chapter, your system should be completely installed and ready to perform basic operations.

Chapter 4: Application Design

This chapter will help you customize the system to meet your application needs. Important application considerations are discussed. Sample applications are provided.

Chapter 5: Software Reference

This chapter explains Compumotor's X-Series programming language in detail. It describes command syntax and system parameters that affect command usage. An alphabetical listing of all commands, with a syntax and command description for each command is included.

Chapter 6: Hardware Reference

This chapter contains information on system specifications (dimensions and performance). This chapter may be used as a quick-reference tool for proper switch settings and I/O connections.

Chapter 7: Maintenance & Troubleshooting

This chapter describes Compumotor's recommended system maintenance procedures. It also provides methods for isolating and resolving hardware and software problems. A list of diagnostic codes is included in this chapter.

Installation Process Overview

To ensure trouble-free operation, you should pay special attention to the environment in which the L Series equipment will operate, the layout and mounting, and the wiring and grounding practices used. These recommendations are intended to help you easily and safely integrate L Series equipment into your manufacturing facility. Industrial environments often contain conditions that may adversely affect solid state equipment. Electrical noise, atmospheric contamination, or installation may also affect the operation of the LX System.

Before you attempt to install this product, you should complete the following steps:

STEP 1 Review this entire manual. Become familiar with the manual's contents so that you can quickly find the information you need.

STEP 2 Develop a basic understanding of all system components, their functions, and interrelationships.

STEP 3 Complete the basic system configuration and wiring instructions provided in Chapter 2, Getting Started. *NOTE: This is a preliminary configuration, not a permanent installation, usually performed in a bench-top environment.*

STEP 4 Perform as many basic moves and functions as you can with the preliminary configuration. You can perform this task only if you have reviewed the entire manual. You should try to simulate the task(s) that you expect to perform when you permanently install your application. *However, do not attach a load at this time.* This will give you a realistic preview of what to expect from the complete configuration.

STEP 5 After you have tested all of the system's functions and used or become familiar with all of the system's features, carefully read Chapter 3, Installation.

STEP 6 After you have read Chapter 3 and clearly understand what must be done to properly install the system, you should begin the installation process. **Proceed in a linear manner**; do not deviate from the sequence or installation methods provided.

STEP 7 Before you begin to tune and customize your system, check all of the system functions and features to ensure that you have completed the installation process correctly.

The successful completion of these steps,will prevent subsequent performance problems and allow you to isolate and resolve any potential system difficulties before they affect your operation.

Developing Your Application

Before you attempt to develop and implement your application, you should address the following issues:

- Recognize and clarify the requirements of your application. Clearly define what you expect the system to do.
- Assess your resources and limitations. This will help you find the most efficient and effective means of developing and implementing your application.
- Follow the guidelines and instructions outlined in this user guide. **Do not skip any steps or procedures**. Proper installation and implementation can only be ensured if all procedures are completed in the proper sequence

Conventions

To help you understand and use this user guide effectively, this user guide's conventions are explained in this section.

Commands

The command examples in this user guide are presented vertically to help you read and understand them. When you actually type these commands at your computer keyboard, they will be displayed horizontally on your computer. All commands that you are instructed to enter are displayed in all capital letters, just as they appear on your computer CRT. A one-line explanation of the command is provided next to each example. The command is displayed in boldface. Be sure to add a delimiter (space or carriage return) after each command in a sequence. Refer to the example below.

Command	Description
> **XE4**	Erases sequence #4
> **XD4**	Define sequence #4 as follows
A2	Sets acceleration to 2 g's
V5	Sets velocity to 5 ips
D1ØØØØ	Sets distance to 10,000 steps
G	Executes the move (Go)
> **XT**	End definition of sequence #4

Responses are bold-faced and set in all capital letters as they are on the terminal. An example is provided below.

Command	Response
> **1XU4**	***A2_V5_D1ØØØØ_G**

The system generally ignores command syntax that is not within the valid range for a specific command (valid ranges are provided in Chapter 5, Software Reference). Compumotor does not guarantee system performance when the system executes commands that contain invalid syntax (outside the valid range).

Warnings & Cautions

Warning and caution notes alert you to possible dangers that may occur if you do not follow instructions correctly. Situations that may cause bodily injury are presented as warnings. Situations that may cause system damage are presented as cautions. These notes will appear in bold face, within a box. The word warning or caution will be centered and in all caps. Refer to the examples shown below.

WARNING

Do not touch the system immediately after it has been in use for an extended period of time. The unit will be hot.

CAUTION

Setting the motor current too high damages the motor.

Voltage Levels

In this manual, you will deal with inputs and outputs that you can turn on or off. We will define the terminologies needed for these inputs and outputs.

Inputs

ON (Ø, *low*)	Current flows
OFF (1, *high*)	No current flows

Command	Description
> TRØ11	Waits for Trigger 1 to turn on and Triggers 2 and 3 to turn off

Outputs

ON (1, *high*)	Current flows
OFF (Ø, *low*)	No current flows

Command	Description
> OØ1	Turns off Output 1 and turns on Output 2

Change Bars

When a new user guide version is produced, the new or changed text is differentiated with a change bar in the right margin (this paragraph is an example). If the entire chapter is changed, the change bar is located next to the chapter title.

Related Publications

The following publications may be helpful resources:

- *Parker Compumotor Programmable Motion Control Catalog*
- Seyer, Martin. RS-232C Made Easy: Connecting Computers, Printers, Terminals and Modems. Englewood Cliffs, New Jersey: Prentice Hall, Inc., 1984
- Operations manual for the IBM or IBM-compatible computer that you will use with the LX System
- Schram, Peter (editor). *The National Electric Code Handbook (Third Edition).* Quincy, MA: National Fire Protection Association

Chapter 1. INTRODUCTION

Chapter Objective

The information in this chapter will enable you to understand the product's basic functions and features.

Product Description

The LX Indexer/Drive combines the functions of an indexer and microstepping drive into one compact system. The system is designed to control linear step motors.

You can create and store up to forty motion programs in the LX's EEPROM memory. You can execute 39 of these programs with simple switch closures from a BCD switch or programmable logic controller (PLC). Up to 40 sequences can be accessed via a computer or a terminal with an RS-232C interface. More than 100 commands are applicable to the LX's powerful command language.

The LX built-in drive is a bi-polar, microstepping drive designed to drive two-phase, permanent magnet, hybrid linear motors. The LX has been specifically designed to take full advantage of the performance of linear step motors.

The LX has an integral power supply, meaning it operates directly from 120VAC. It will produce up to 3A of current per motor phase.

LX systems are fully-packaged, single-axis, linear motion systems. No additional mechanical hardware is necessary for most LX system applications. The standard system consists of a forcer, platen, indexer/drive, and interconnecting cables. Four forcer sizes are available with static forces of 3, 6, 9, and 20 pounds.

The LX Indexer/Drive is designed for panel mounting and may be installed in a minimum depth or minimum width configuration by moving its mounting brackets to the side or back of the unit.

Accelerometer (-AC) Option

The -AC option includes an *accelerometer* which is mounted in the L20 forcer to electronically dampen the resonance, or *ringing*, which is present in all stepper systems. The -AC option is available for new and existing LX systems that use an L20 forcer.

Product Features

The LX Indexer/Drive provides the following features:

- A complete system with drive electronics, indexer integral supply, heatsink, and cables
- Four forcer sizes are available with forces of 3, 6, 9, and 20 pounds
- A highly flexible 12-foot forcer cable provided for L20 forcers, and a coiled 10-foot forcer cable provided for L3, L5 and L9 forcers
- High-voltage operation for optimal high-speed performance
- Bipolar drive provides 0-3 Amperes/Phase
- 95-135VAC, 50/60 Hz input power
- Protected against short circuit, brownout, and over temperature (but not short circuit to ground)
- Microprocessor controlled microstepping provides smooth operation and prevents mid-frequency instability problems
- 10,000 steps/inch is the standard resolution for the L5A, L9A, and L20 forcers (7,500 is standard for L3C forcers)
- Built-in RS-232C indexer with an X Series command language superset
- Up to eight units daisy-chained on a single RS-232C port
- 1,800 characters of non-volatile memory can store up to forty motion programs
- Nine optically isolated inputs for end-of travel limits, home, triggers, and sequence execution
- Two optically isolated programmable outputs
- Automatic power-up motion program is offered
- Motion sequence selection via RS-232C or external switch closures
- Sequence upload and download functions
- Incremental encoder interface is standard (need to purchase the encoder separately)
- Automatic acceleration roll-off provides maximum force utilization and shorter move times
- Flexible acceleration profiling is available for customized acceleration profiles
- Transmit messages from the RS-232C port to control other RS-232C controllers or for message display
- **MT** command provides move time information on the currently defined move

-AC Option Features

The Accelerometer option offers the following enhancements to the LX Indexer/drive using the L20 forcer:

- Higher Throughput
- Faster Settling
- Smoother Operation

Higher Throughput

The -AC option allows more of the L20's force to be used for acceleration; therefore, less force is needed for a safety margin. Only 25-35% of the linear motor force can be used for acceleration if an accelerometer is not used. This number is 60-70% for linear motors with the accelerometer option. For most applications, you can double the acceleration.

Faster Settling

The duration of settling at the end of a move is reduced by more than 90%. If the settling time for an unloaded L20 forcer without an accelerometer is 190ms, it will be 13ms for an L20 forcer with accelerometer feedback. A similar reduction in settling time exists for L20 forcers with payloads.

Smoother Operation

The -AC option provides smoother operation at velocities around the motor's resonant frequency. When the pole frequency of its 2^{nd} or 4^{th} harmonic (1 ips = 25Hz) falls on or near resonance, the forcer will run very roughly. The accelerometer feedback reduces this roughness by two-thirds.

Linear Motor Theory of Operation

The linear hybrid stepping motor operates on the same electromagnetic principles as the rotary hybrid motor. The stator or ***platen*** is a steel bar (with teeth) that extends over the distance to be traversed. The platen is entirely passive. All magnets and electromagnets are incorporated into the armature or ***forcer***. The forcer moves (bi-directionally) along the platen, assuming discrete locations in response to the state of the currents in its field windings. Figure 1-1 shows a basic structure of the forcer and platen.

Figure 1-2 is a simplified schematic representation of the motor geometry and shows the forcer in four distinct stages of excitation. The forcer consists of two electromagnets (A and B) and a strong, rare-earth, permanent magnet (hence the name *hybrid*). The two pole faces of each electromagnet have teeth (to concentrate the magnetic flux).

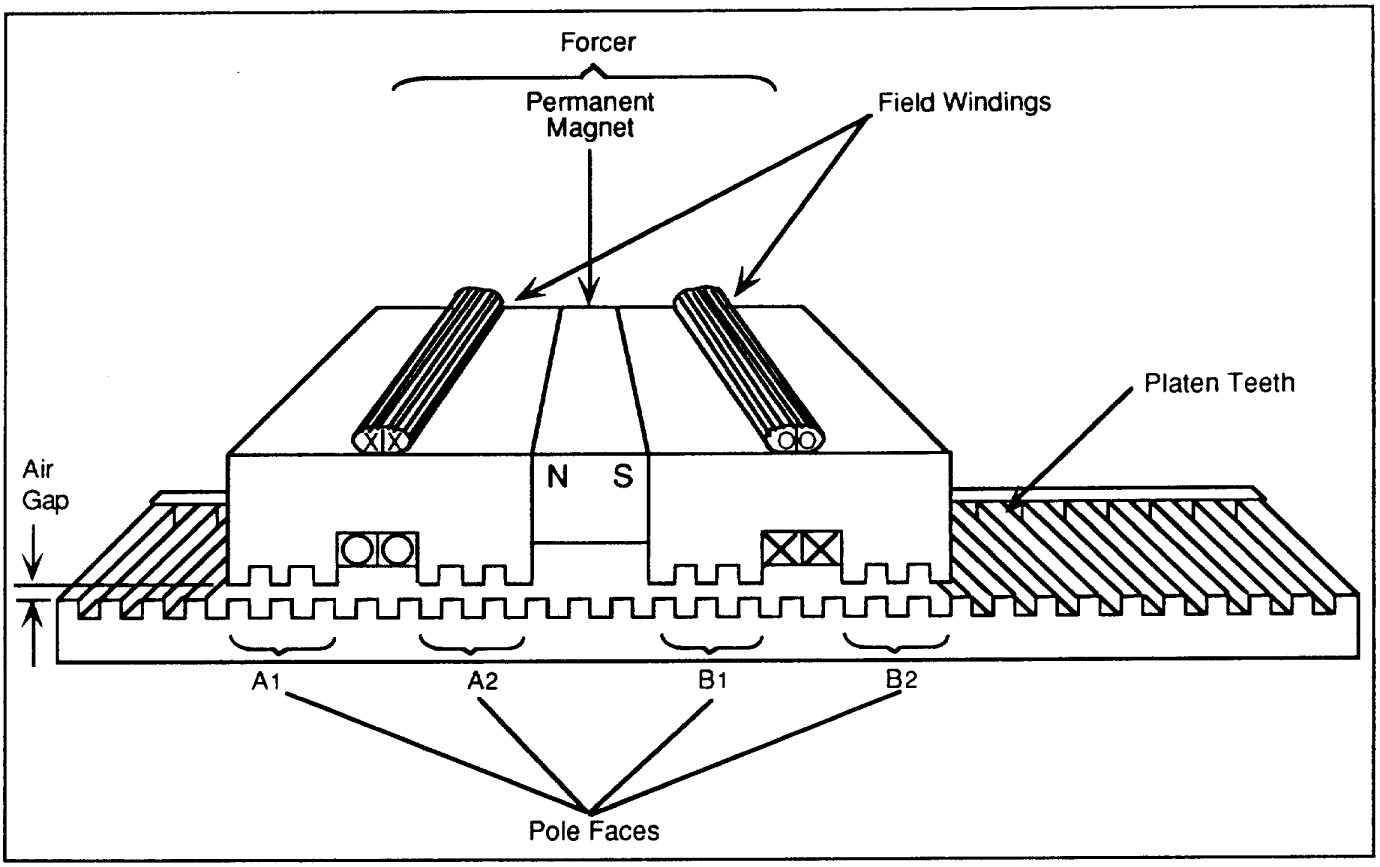


Figure 1-1. Construction of a Linear Motor

The four sets of teeth are spaced in mechanical quadrature, so that only one set at a time can be aligned with the platen teeth. When current is established in a field winding, the resulting magnetic field reinforces permanent magnet flux at one pole face and cancels it at the other. By reversing the current, the reinforcement and cancellation are exchanged. When current is removed, the PM flux divides equally between the pole faces. By selectively applying current to phase A and B, it is possible to concentrate flux at any of the forcer's four pole faces. The face receiving the highest flux concentration will attempt to align its teeth with the platen. Figure 1-2 depicts the four cardinal states or full steps of the forcer. The four steps result in motion of one tooth interval to the right. If you reverse the sequence, the forcer will move to the left.

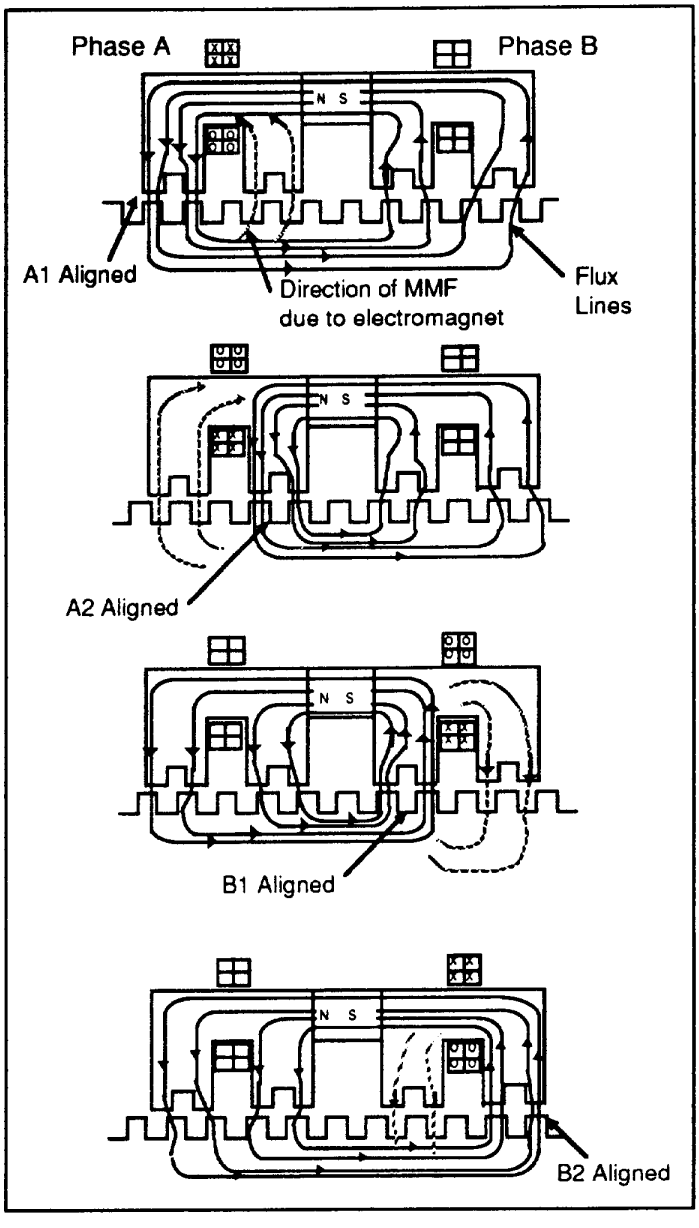


Figure 1-2. Four Steps of a Linear Step Motor

Linear motors are microstepped by proportioning current to the two phases of the forcer (similar to rotary stepping motors). The LX steps the linear motor at 125, 100, 90, or 50 microsteps per full step.

If the sequence in Figure 1-2 is repeated, the forcer continues to move. When the sequence is stopped, the forcer stops with the appropriate tooth set aligned. At rest, the forcer develops a restoring or holding force that opposes any attempt to displace it from equilibrium. As the resting forcer is displaced further from equilibrium, the restoring force increases until the displacement reaches one-fourth of a tooth interval (see Figure 1-3). Beyond this point, the restoring force drops. If the forcer is pushed over the crest of its holding force, it slips or jumps rather sharply and comes to rest at an integer number of tooth intervals away from its original location. If this occurs while the forcer is traveling along the platen, it results in a stall condition.

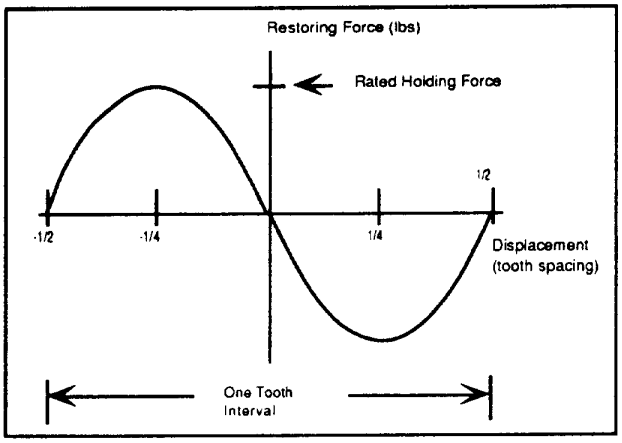


Figure 1-3. Force vs. Displacement Curve

The magnetic flux that passes between the forcer and the platen creates a very strong normal force of attraction between the two pieces. This force is as much as ten times the peak holding force of the forcer. The attractive force must be offset with some bearing arrangement to maintain precise clearance between the pole faces and the platen teeth. Compumotor linear motors use either mechanical roller bearings or air bearings to maintain clearance.

Chapter 2. GETTING STARTED

Chapter Objectives

The information in this chapter will enable you to do the following:

- Verify that each component of your system has been delivered safely.
- Ensure that the unit functions properly by bench testing
- Configure the system properly

What You Should Have

Upon receipt, you should inspect your LX Indexer/Drive system for obvious damage to its shipping container. Report any damage to the shipping company as soon as possible. Parker Compumotor cannot be held responsible for damage incurred in shipment. Carefully unpack and inspect your L Series shipment. The ship kit items listed in Table 2-1 should be present and in good condition. *The Fan Kit and the Air Regulator/Filter Kit are optional.*

Description	Part Number
Possible Indexer/Drive Units:	
LX Indexer/Drive	LX-DRIVE
LX Indexer/Drive with -AC	LX-DRIVE-AC
Possible Forcers:	
L3C	L3C
L5A	L5A
L9A	L9A
L20	L20
L20 with -AC Option	L20-AC
Possible Platens:	
L3C	PO-L3C-P15
L5A	PO-L5A-Pxx
L9A	PO-L9A-Pxx
L20	PO-L20-Pxxx
Forcer-to-Drive Cable*	71-007553-12
AC Power Cable Assembly	71-006593-01
Male 25-Pin D-Connector	43-001989-01
D-Connector Shell	43-001990-01
5-Pin Phoenix Connector**	43-005561-01
LX Indexer/Drive User Guide	88-008896-01
Optional Parts:	
Air Regulator/Filter Kit	-RFKC
Fan Kit	-LFK

* L20 forcer-to-drive cable is integral to the forcer
** Included only in *LX-Only* shipments

Table 2-1. Ship Kit List

NOTE: When a drive/forcer system is ordered, the forcer is packaged with the LX Indexer/Drive. The platen is shipped separately.

Verify Factory Settings

The LX has DIP switches for configuring drive and indexer functions. These DIP switches are set at the factory to default settings. Motor current and current profile are set to accommodate the motor that you ordered with your LX system.

Before you perform system connections you should **inspect the DIP switches to ensure that they are set to the factory default configuration**. Figure 2-1 illustrates the location of the DIP switches on the LX.

Once you have permanently installed your LX system refer to Chapter 6, Hardware Reference, for alternative DIP switch settings and adjustment procedures.

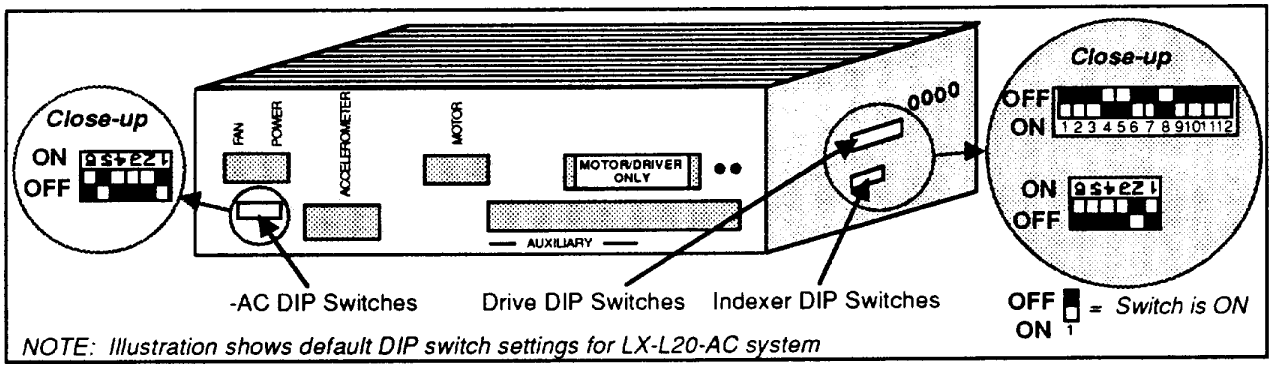


Figure 2-1. Location of LX DIP Switches

Drive Settings

The standard LX sets its drive functions with the 12-position Drive DIP switch, while the -AC option LX uses the Drive DIP switch and the 6-position -AC DIP switch.

Motor Current

Refer to Table 2-2 for the factory default motor current settings. Motor current is set with the Drive DIP switch. *NOTE: These current values are approximate settings in the LX. The actual current will not match these settings exactly.*

CAUTION

In order to measure these currents with an oscilloscope, the LX must be powered by an *isolation* transformer.

Forcer Size	Phase Current Amps	Maximum Force (Lbs)	Drive DIP Switch Settings				
			1	2	3	4	5
L-L3C	0.961	3.0	OFF	ON	OFF	ON	OFF
L-L5A	1.450	6.0	OFF	ON	ON	ON	ON
L-L9A	1.450	9.0	OFF	ON	ON	ON	ON
L-L20	2.721	20.0	ON	ON	ON	OFF	OFF
LX-ONLY*	0.000	0.0	OFF	OFF	OFF	OFF	OFF

* Default setting if no motor is shipped with the LX order

Table 2-2. Factory Default Motor Current Settings

Motor Resolution

For proper operation of the -AC system, the resolution DIP settings on the -AC DIP switch must be the same as those on the Drive DIP switch. Table 2-3 provides the default motor resolution settings.

Drive DIP Switch Settings			
8	9	10	Resolution
OFF	ON	ON	10,000 steps/in pure*

*6,000 steps/in for L3A forcers

-AC DIP Switch Settings *(-AC System Only)*			
4	5	6	Resolution
ON	OFF	ON	10,000 steps/in

Table 2-3. Factory Default Drive and -AC Motor Resolution Settings

Automatic Run

The Automatic Run feature allows you to perform a quick functional check of the system. Automatic Run is enabled or disabled using **Drive DIP switch #6**. The default setting is ON (disables the automatic run function). With the Automatic Run function enabled, powering up the system causes the forcer to move at 50 poles/sec (200 full steps/sec) until the LX is powered down. This corresponds to the following speeds:

- L3C: 3.33 ips
- L5A, L9A, & L20: 2 ips

The forcer will continue to move until it reaches the end of the platen, or until the LX is powered down.

CAUTION

Do not allow the forcer to travel over the end of the platen. This may damage the platen and the forcer.

Wait for the power-up instructions before using this function.

Automatic Standby

The Automatic Standby feature allows the forcer to cool when it is not moving. When enabled, this feature drops the current to the forcer by 50% when the forcer receives no step pulses for one second. Full power is restored when a step pulse is received. Automatic Standby is enabled or disabled using **Drive DIP switch #7**. The default setting is ON (disables the automatic standby function).

Motor Waveforms

Drive DIP switches 11 and 12 select the current waveform to achieve the smoothest system operation. Leave switches 11 and 12 set to ON (factory setting) for optimum forcer performance (see Table 2-4).

Waveform Shape	11	12
Pure Sine	ON	ON

Table 2-4. Default Motor Waveform

Servo Gain (-AC Systems Only)

The servo gain is set using -AC DIP switches 1 - 3. The default setting is for a payload of 0 - 1 lb (see Table 2-5).

-AC DIP Switch			Description
1	2	3	
OFF	ON	ON	payload: 0 to 1.0 lb

Table 2-5. Default Gain Settings

Indexer Settings

The LX indexer functions are set with the 6-position Indexer DIP switch (see Figure 2-1).

Profile Selection

You can use the Indexer DIP Switch to select different profile settings. The LX has predefined acceleration profiles stored in ROM that are designed to maximize force utilization and minimize the move time for the forcer. This acceleration profile is customized for different forcers. Table 2-6 provides the factory default DIP Switch settings for each forcer.

Indexer DIP Switch Settings			Forcer
1	2	3	
OFF	OFF	ON	L3C
OFF	ON	ON	L5A
ON	OFF	OFF	L9A
ON	OFF	ON	L20
OFF	OFF	OFF	LX-Only*

* Default setting if no motor is shipped with the LX order

Table 2-6. Default Profile Settings

Address

Using the Indexer DIP Switch (shown in Figure 2-1), make sure the address is set to 1. The proper DIP Switch settings for Address 1 are shown in Table 2-7 below.

Indexer DIP Switch Settings			Address
4	5	6	
ON	ON	ON	1

Table 2-7. Default Address Setting

You will have a chance to verify that the Indexer DIP Switch is properly set for Address 1 after you connect the LX to your computer.

Basic System Configuration

This section provides procedures and information to perform the basic configuration of your LX System. Please note that this is a temporary *bench-top* configuration. Instructions for complete permanent configuration are discussed in Chapter 3, Installation.

System Connections

Figure 2-2 shows the location of the connectors on the LX. Refer to Chapter 6, Hardware Reference, for detailed pinouts of each LX connector.

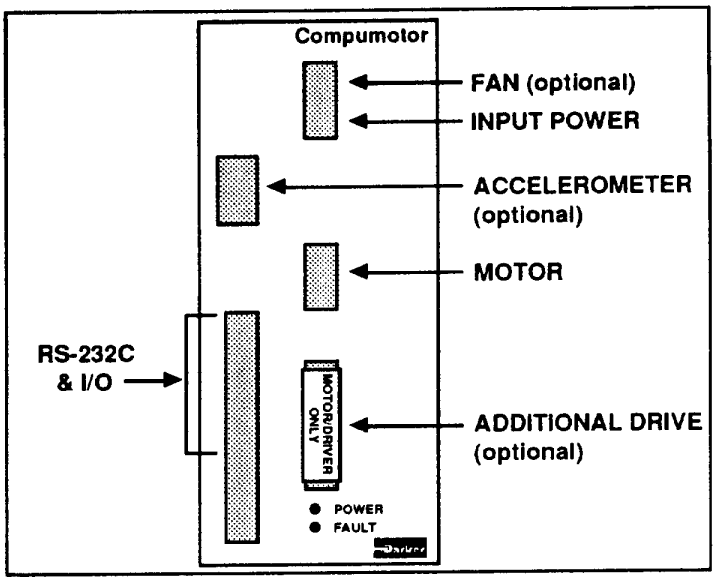


Figure 2-2. Location of LX Connectors

The LX is protected against over-temperature and short-circuits between forcer phases (but not phase-to-ground). Compumotor does not recommend that you test these features or operate your system in such a way as to induce short-circuit or over-temperature situations.

WARNING

Ensure that AC power is disconnected before attempting to perform any wiring. NEVER disconnect the forcer with power applied to the drive.

Air Connections

The air gap between the L3C and L5A forcers and platens is maintained with roller bearings. The L20 and L9A forcers use an air bearing to maintain a precise air gap between the forcer and the platen. The L20 requires 40 to 45 pounds per square inch (PSI) pressure at a flow rate of about 20 cubic feet of air per hour (SCFH). For an unloaded L20 forcer, the recommended air pressure is 40 PSI. The L9A requires between 50 and 70 PSI and a flow rate of about 30 SCFH.

Available *shop* air should be 70 PSI or higher in order to be regulated and filtered to the required pressure. For long life operation, the air must be non-lubricated, clean, and dry. An optional Air Regulator/Filter Kit (-RFKC) can be purchased from Compumotor. In applications where air cannot be used, nitrogen is an excellent substitute.

NOTE: Too much air pressure makes the air bearing unstable. An unstable air bearing produces a loud tone in the forcer.

Figure 2-3 illustrates the air hose connections. Use the following procedures for mounting the forcer on the platen and connecting the air.

STEP 1 Connect the L20 or L9A forcer to the air regulator and apply air pressure (40 PSI for L20; 60 PSI for L9A).

STEP 2 Make sure the air bearing surfaces are clean.

STEP 3 Carefully place the forcer on the platen.

STEP 4 Manually move the forcer back and forth over the platen to verify that the forcer moves freely over the platen surface. Refer to Chapter 3, Installation, for a more detailed description.

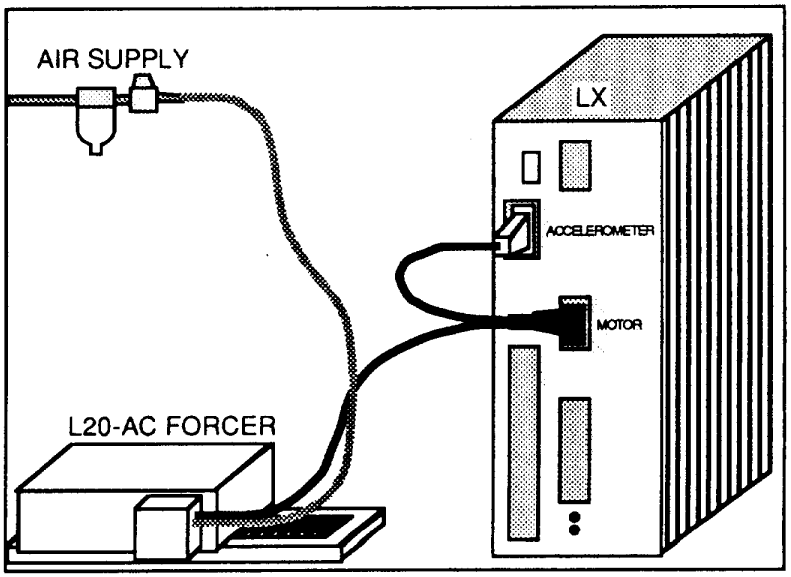


Figure 2-3. Forcer, Air, and Optional Accelerometer Connections

Forcer Connections

CAUTION

Do not energize the forcer when it is not on the platen; this will cause the forcer to overheat. Never disconnect the forcer when power is being applied to the drive; this will damage the contacts of the motor connector.

Do not operate the LX with a forcer that is not supplied by Compumotor.

If you have an L3C, L5A, or L9A forcer, connect the supplied coil cable between the forcer and the 5-Pin phoenix motor connector on the drive and verify the following color codes:

- Phase A+ = Red
- Phase A- = Black
- GND = Shield
- Phase B+ = White
- Phase B- = Green

Tighten the 9-Pin D-Connector screws to the forcer so that the cable will not come loose during operation.

The L20 forcer cable is prewired directly to the forcer. Simply plug the end with the 5-pin connector into the motor connector on the LX . If you have an L20 forcer with the -AC option, you must plug the additional 9-Pin D-Connector to the Accelerometer connector (see Figure 2-3) and tighten the screws.

Detailed cable mounting and extension instructions are provided in Chapter 3, Installation.

Establish Communication

To communicate with the LX Indexer/Drive, your computer or terminal must have an RS-232C serial port. If your computer does not have a serial port, you can purchase one from your local computer dealer.

You cannot change the communication parameters on the LX. Therefore, you must set your computer/terminal to the following communication settings:

Baud Rate: 9,600
Data Bits: 8
Parity: None
Stop Bits: 1
Echo: Off (Full Duplex)

Refer to your computer's operations manual for instructions on how to set the computer's communications parameters so that they are compatible with the LX's settings. If you have an IBM or IBM-compatible computer, you can use different types of software (i.e., PC-TALK, XTALK, etc.) to set up the communication parameters.

RS-232C CONNECTIONS

The LX uses a simple three-wire implementation of RS-232C serial communication. RECEIVE DATA (Rx), TRANSMIT DATA (Tx), and GROUND (GND) signals are transmitted via pins 1, 2, and 3 on the LX auxiliary connector (as shown in Figure 2-4).

WARNING

The (GND) ground connections on the auxiliary connector are signal ground or common as opposed to earth ground on the motor and power connectors. The GND on the auxiliary connector should be isolated from the earth ground. Do not connect the GNDs on the auxiliary connector to the GNDs on the motor and power connectors. This type of miswiring can cause system damage.

The LX does not support handshaking of any form; therefore, you should disable the handshaking function of the computer or terminal sending characters to the LX. Normally, handshaking can be disabled via software; however, if that is not possible you can disable the handshaking function by connecting RTS to CTS (usually pins 4 and 5) and DSR to DTR (usually pins 6 and 20) on the computer's or terminal's 25-pin RS-232C port. Refer to your computer or terminal user guide for the exact instructions to disable handshaking.

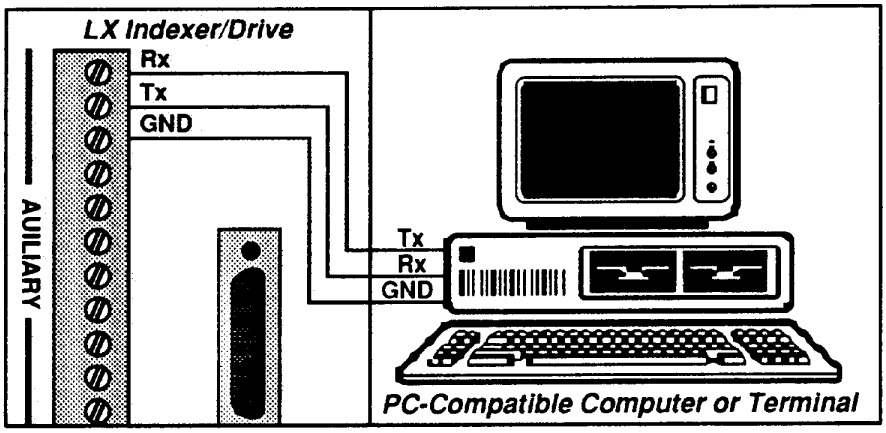


Figure 2-4. Wiring RS-232C Interface to a Computer or a Terminal.

Transformer Connections (Optional)

This section addresses connecting an optional transformer to the LX system. An isolation transformer can enhance phase-to-earth ground short circuit protection, personal safety, and electrical noise immunity. If you are not using a transformer with this system, simply proceed to the Power Connections section. ***The LX Indexer/Drive requires a transformer rated at 250VA.***

CAUTION

Do not apply power to the LX Indexer/Drive at this time. AC power is limited to 132VAC. Higher voltages damage the LX. The low-voltage limit is 95VAC.

Refer to the transformer user guide to determine which output leads correspond to LINE, NEUTRAL, and GROUND. As illustrated in Figure 2-5, connect the transformer leads to the AC power connector on the drive.

WARNING

Do not connect the transformer to the LX while power is applied to the transformer. Do not touch the wiring studs on the transformer after it is plugged into an AC outlet. This can cause serious personal injury.

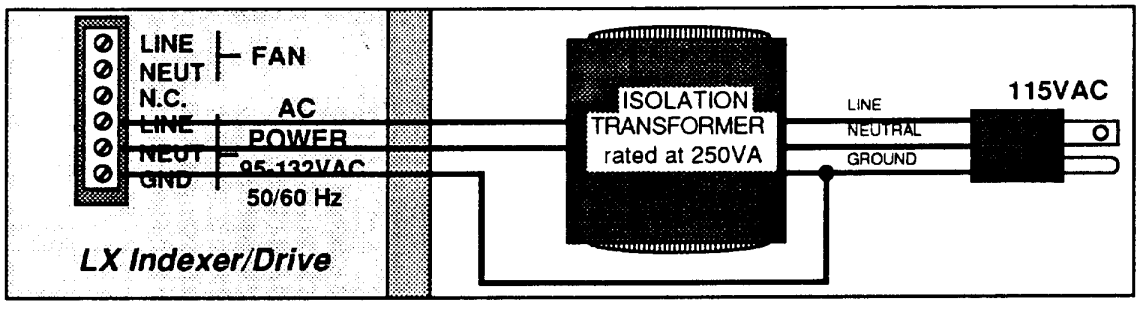


Figure 2-5. Transformer Connections (Optional)

Power Connections

The LX system is shipped with the power cable pre-wired to a 6-pin phoenix connector. Connect the power cable assembly to the LX power connector (see Figure 2-2) and verify that the color codes are correct.

- LINE = Black
- NEUT = White
- GND (Earth) = Green

European color codes differ in that LINE = Brown, NEUT = Blue, and GND = Green or Yellow.

REMEMBER: AC power is limited to 132VAC. Higher voltages damage the LX. The low-voltage limit is 95VAC.

Fan Connections (Optional)

If your LX system will be operating in temperatures exceeding 45°C, you should use the optional fan kit for additional cooling. Connect the fan cable to the LX fan/power connector (see Figure 2-2). Make sure that line and neutral are connected to the appropriate screw terminals.

Applying Power

Before you power-up the system, you should verify that all cables and wires are properly connected and that the forcer is free from obstructions. If everything is OK, the drive will be enabled when you apply power. The green **POWER** LED will be on and the red **FAULT** LED will be off.

If there is a short circuit, over temperature, or undervoltage condition, the red **FAULT** LED will be illuminated. If this happens remove power to the system and refer to Chapter 7, Maintenance & Troubleshooting.

NOTE: The L20 system with the -AC option is a little louder than the standard L20 systems. During warm-up, the L-Servo PCB may generate audible noise in the forcer. This is normal.

System Tests

Test RS-232C Communication

After installing your RS-232C interface, set up your computer as a terminal (serial communication capability). You may do this with the LX Applications Software and an IBM or IBM-compatible computer. Do not apply power to the LX at this time.

STEP 1 After booting up your IBM/IBM compatible, insert the LX Application Software in disk drive A.

STEP 2 Type **RUN** and press the Enter key. A welcome message appears on the screen.

STEP 3 Press the Enter key three times. The main menu appears on the screen.

STEP 4 To move the cursor to the **LX Programming** option, press the right arrow key twice.

STEP 5 To move the cursor to the **Terminal Emulator** option, press the down arrow key once (this is the option that converts your computer into a communicating terminal).

STEP 6 Apply power to the LX.

STEP 7

Press the Enter key once to activate the Terminal Emulator option and initiate serial RS-232C communications between your computer and the LX.

You will see the following blinking message in the center of your screen:

```
RS-232C  Checkout
```

If the **LX Terminal Emulator** message appears, you are communicating with the LX and everything you type on the keyboard is being transmitted to the LX.

If you have a dumb terminal instead of a computer, you do not need to use any special software for communication. The terminal automatically sends and receives characters. However, make sure that the terminal is set up at a baud rate of 9600, 8 data bits, 1 stop bit, no Parity, and full duplex. You should refer to the terminal user guide or operations manual for instructions on setting these values.

STEP 8

The LX Indexer/Drive's factory default operating mode is not the Interactive mode. The Interactive mode is desirable when you are becoming familiar with the system. In the Interactive mode, the indexer identifies valid or invalid commands. When the indexer receives a valid command, it will respond with a prompt (>). When it receives an invalid command, it will respond with a question mark (?). To enter the Interactive mode, issue the **SSI1** command. To operate in this mode permanently, enter the Save (**SV**) command after issuing the **SSI1** command.

NOTE: Do not use the interactive mode with the LX Application Software. Doing so may cause a PC system error.

The remainder of this user guide assumes that you are operating in the Interactive mode. If you are not operating in this mode, the prompts that are shown in these examples will not be displayed on your terminal or computer screen.

STEP 9

To verify that your DIP Switch 1 address setting (for Address 1) is correct enter the following command:

> 1R [cr]

The system responds with ***S** [cr] or ***R** [cr].

If you have system communications problems, refer to Chapter 7, Maintenance & Troubleshooting .

Disabling Limit Switch Inputs for Test Moves

If you wish to test your motor without connecting the limit switches, you can disable them by issuing the Limit Disable (**LD3**) command. Compumotor ships the drive from the factory with the limits enabled. This means that the motor will not run unless you have properly connected the limit switches, or you have issued the **LD3** command.

Make Open-Loop Test Moves

After verifying that your RS-232C interface is working properly and that your address setting is correct, type the following commands. *(The following example disables the limit switches. Never attach a load when you disable the limit switches.)*

Commands	Description
> LD3	Disables CW and CCW limits
> A.2	Sets acceleration to 0.2 g's
> V1	Sets velocity to 1 ips
> D1ØØØØ	Sets distance to 10,000 steps
> G	Executes the move (Go)
> 1PR	Requests for position report from Device 1

The linear motor moves 10,000 steps. After the motor stops, the system provides the following response.

*+0000010000[cr]

If the motor does not move, refer to Chapter 7, Maintenance & Troubleshooting.

If you were able to perform the test routine and the open-loop move successfully, try to perform as many moves as possible with the basic configuration. Simulate the moves you intend to perform with your permanent application. After you complete these exercises, read Chapter 3, Installation.

To verify the LX system performance, enter the following commands to make the forcer move back and forth on the platen.

Commands	Description
> A_*	See below for proper acceleration value
> V50	Sets velocity to 50 ips
> D2ØØØØ	Sets distance to 20,000 steps
> L	Begins a loop as defined between this command and the N command
> G	Executes the move (Go)
> H	Changes the direction the forcer will move
> N	Ends the loop definition

* Use the following acceleration values for the different forcers:

- L3C: **1.0 g**
- L5A: **2.7 g's**
- L9A: **2.5 g's**
- L20: **3.0 g's**
- L20-AC: **6.0 g's**

NOTE: With the move parameters provided above, the forcer should not stall or slip. If the forcer stalls or slips, refer to Chapter 7, Maintenance & Troubleshooting.

Accelerometer Functional Test

Use the following procedures to verify the functionality of the Accelerometer (L20-AC):

STEP 1 Make sure there is no payload attached to the forcer.

STEP 2 Set -AC DIP switches #1 through #3 ON. This disables the accelerometer.

STEP 3 Command the indexer to perform a move with the following parameters. *Note that the acceleration value can be higher for single moves than for repeated moves.*

Commands	Description
> **MN**	Sets indexer to normal mode
> **A2**	Sets acceleration to 2 g's
> **V40**	Sets velocity to 40 ips
> **D1ØØØØØ**	Sets distance to 100,000 steps (10 inches at 10,000 steps/in resolution)
> G	Executes the move (Go)

STEP 4 Determine the maximum acceleration by repeating step 3. Each time you repeat step 3, increase the Acceleration (**A**) value by increments of 0.25 g's until the forcer stalls or slips. The maximum acceleration should be around 3 g's.

STEP 5 Change -AC DIP switch #1 to OFF (enables the accelerometer).

STEP 6 Repeat step 4 to determine the maximum acceleration. The maximum acceleration should be around 6 or 7 g's. If the forcer stalls or slips at an acceleration value below 6 g's, the accelerometer is probably not operating properly. Make sure Drive DIP switches #8 through #10 and -AC DIP switches #4 through #5 are set to the same resolution (see Table 2-3). If the resolution settings are correct, the accelerometer is faulty and must be returned for repair.

Chapter 3. INSTALLATION

Chapter Objectives

The information in this chapter will enable you to do the following:

- Properly mount all system components
- Connect all electrical and non-electrical system inputs and outputs properly
- Ensure that the complete system is installed correctly
- Perform basic system operations

NOTE: You must complete all the basic system configuration procedures in Chapter 2, Getting Started, before proceeding with the procedures in this chapter.

Adjusting DIP Switches

If the factory default DIP switch settings (discussed in Chapter 2) are not appropriate for your application, refer to Chapter 6, Hardware Reference, for adjustment procedures and optional settings.

System Mounting

You should give special attention to the environment and location in which you will operate the LX system. Atmospheric contamination and excess heat should be considered before installing and operating the LX System.

Enclosure Considerations

You should install the LX in an enclosure to protect it against atmospheric contaminants such as oil, moisture, and dirt. The National Electrical Manufacturers Association (NEMA) has established standards that define the degree of protection that electrical enclosures provide. The enclosure should conform to NEMA Type 12 standards if the intended environment is industrial and contains airborne contaminants. Proper layout of components is required to ensure sufficient cooling of equipment within the enclosure.

Drive Mounting

Proper mounting and panel layout are essential for trouble-free operation of the LX. You should allow sufficient space for unrestricted air flow over the heatsink. The drive must always be mounted so that the heatsink fins are positioned vertically to allow efficient convection cooling of the drive (refer to Figure 3-1).

If you mount the LX in an enclosure, observe the following guidelines:

- The vertical clearance between the LX and other equipment, or the top or bottom of the enclosure, should be no less than 6 inches.
- The horizontal clearance should be no less than 4 inches.
- Large, heat-producing equipment should not be mounted directly beneath the LX.
- The LX should not be mounted directly below an indexer (the LX produces more heat than an indexer).
- The maximum allowable ambient temperature directly below the LX is 113°F (45°C). Fan cooling may be necessary if adequate air flow is not provided.

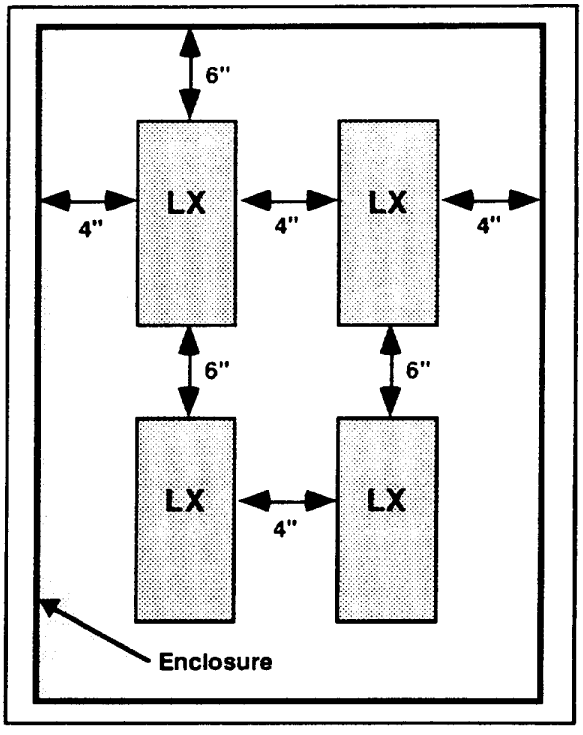


Figure 3-1. Panel Layout Guidelines

As discussed below, the LX can be mounted for minimum width or minimum depth, depending on how the mounting clips are attached to the casing.

Minimum-Width Mounting

Two clips are attached to the top and bottom rear of the drive, away from the power connectors (see Figure 3-2). This gives you the maximum amount of panel space possible. Units are shipped with clips in this configuration.

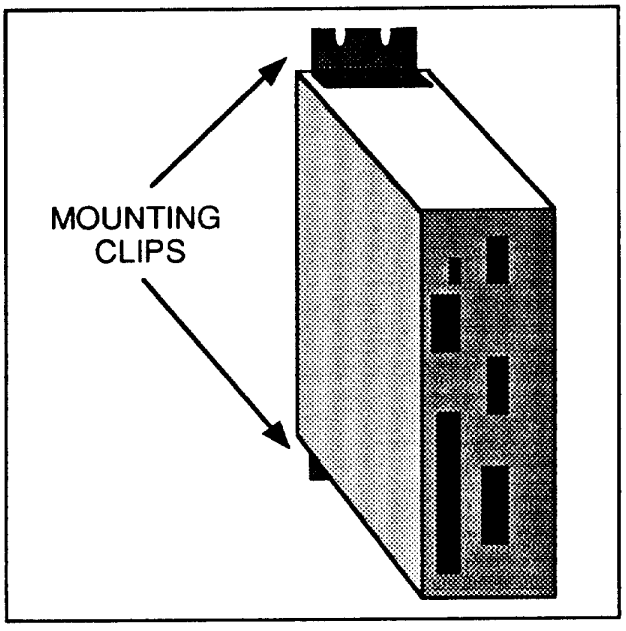


Figure 3-2. Minimum Width Panel Mount

Minimum-Depth Mounting

The two clips are moved and attached to the drive on the side away from the heatsink to provide minimum depth (see Figure 3-3). This allows you to mount the drive in the shallowest possible enclosure.

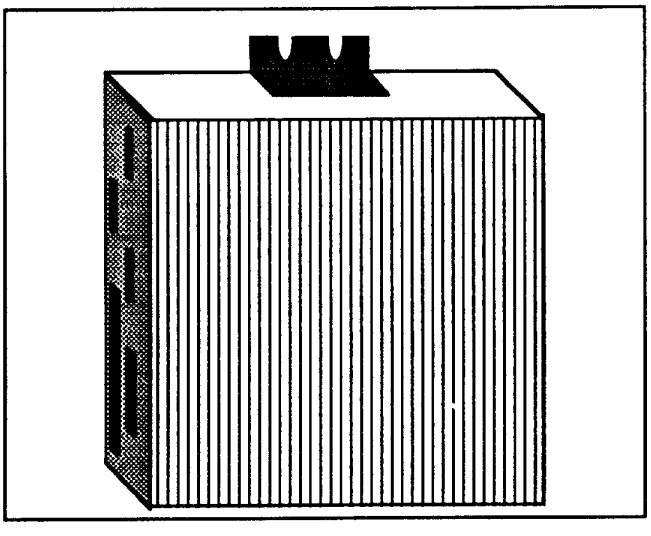


Figure 3-3. Minimum Depth Panel Mount

Platen Mounting

Platen Flatness

The platen is ground flat and parallel to the proper specifications while being held down on a magnetic chuck. When released from the magnetic chuck, mechanical stresses in the platen can cause it to bow or twist slightly. However, by mounting the slightly bowed or twisted platen on a ground flat surface, it will become flat again.

When the platen is not mounted, the L20 and L9A forcers may rub or even bind on uneven areas. The air gap between the L20 or L9A forcer and the platen is approximately 0.0005 inches. The forcer length is 4.6 inches. Therefore, when the platen has a *bump* with a slope greater than 0.0001 to 1 inches (vertical to horizontal), the forcer will not float properly and will rub the platen surface, and/or the air bearing will become unstable.

The air gap between the L3C or L5A forcer and the platen is approximately 0.0015 inches. This larger air gap makes these forcer's less sensitive to bowed or twisted platens.

The recommended platen flatness for Compumotor forcers is provided in Table 3-1.

Platen	To Operate
PO-L3C-PXX	0.001 in/4 in (0.00025 in/in)
PO-L5A-PXX	0.001 in/4 in (0.00025 in/in)
PO-L9A-PXX	0.00025 in/5 in (0.00005 in/in)
PO-L20-PXX	0.00025 in/5 in (0.00005 in/in)

Table 3-1. Flatness Required for Platen Mounting

To prevent platen flatness problems, especially in the prototype phase of a project, Compumotor recommends using platens mounted on platen stiffeners. You can order platen stiffeners through the Compumotor Custom Product Group.

Platen Straightness

Long platens can also bend laterally (side-to-side). Therefore, when mounting a long platen, make sure that the platen is mounted as straight as possible. Refer to Chapter 6, Hardware Reference, for specifications.

Platen Mounting

Platens are not self-supporting. A platen suspended only by the ends will bow from 0.001 inches to more than 0.020 inches, depending on the length of the platen and the weight of the forcer with its payload. Also, mounting the platen by attaching only the ends, even on a flat surface, does not guarantee a flat platen surface. Consequently, ***you must bolt the platen on a flat surface using every available platen mounting hole.***

Mounting holes are drilled and tapped from the bottom of the platen. Refer to Table 3-2 for platen mounting hole specifications. **Do not over-tighten mounting bolts; this can warp the platen and the mounting surface.**

CAUTION

You must exercise caution if you are drilling holes into the platen. Be sure that the holes are not in the travel path of the forcer. Also, avoid warping or contaminating the platen. Refer to Chapter 6 for mechanical specifications.

Platen Type	Hole Specifications
L3C	8-32 UNC X 0.31 Deep
L5A	10-32 UNF 28 X 0.31 Deep
L9A	10-32 UNF 28 X 0.31 Deep
L20	10-32 UNF 28 X 0.31 Deep

Table 3-2. Platen Mounting Holes

For high-precision applications, mount the platen with dowel pins and then fasten with bolts. If you cannot bolt the platen from the bottom, you may glue the platen onto the mounting surface.

For suspended operation, such as an X-Y gantry system, you may mount the platen on a stiffener. Compumotor can provide a custom light-weight platen, if required.

To avoid warping the platen, you should make sure the base on which the platen is attached is constructed of a material with a similar coefficient of thermal expansion. The coefficient is 0.00000633 $in^2/°F$. For example, a platen mounted on an aluminum surface may warp when the forcer is run at high speeds and high duty cycles (see Figure 3-4).

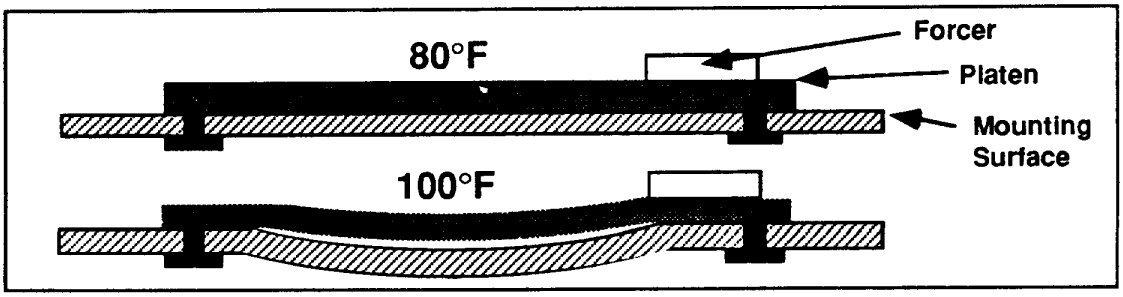


Figure 3-4. Mounting Platen for Heat Variances

Platen Contamination

Since the air gap of the forcer is crucial to its operation, you should take precautions to prevent contamination. You may therefore choose to mount the platen on its side, or upside down, to allow any non-adhesive material to fall off the platen. In other cases, a simple metal bracket can be put around two or three sides of the forcer to prevent contamination from reaching the platen. If a light amount of dust is present (such as flour or dry sawdust) the air bearing of the L20 will clean the unit as air moves in front of the forcer.

Periodic cleaning of the platen surfaces with alcohol or acetone is required, even in relatively clean environments such as normal engineering labs and offices. In dirtier environments, a protective boot may be needed. The boot covers both the platen and the forcer. As the forcer moves, the boot folds (like an accordion). Compumotor recommends the following source for protective boots:

A & A Manufacturing Co.
2300 S. Calhoun Rd.
New Berlin, WI 53151
Phone: (414) 786-1500
Specify: die #85-2524 Hypalon polyester material

Shiny spots or scratches on the platen surface indicate that the forcer is not floating properly. This is due to improper platen mounting and/or clogged orifices in the forcer. Using the forcer for extended periods with clogged orifices will damage the air bearing surface. Refer to Chapter 6, Maintenance & Troubleshooting, for procedures on checking and cleaning clogged orifices.

Use Lapping paper or a lapping stone to remove any burrs or scratches on the forcer and platen surfaces. After deburring, thoroughly clean the platen and forcer surfaces with alcohol to remove any loose particles and to prevent additional scratches.

Platen Corrosion

The L5A platen is chrome-plated on all sides. The L9A platen is also chrome-plated except for the top (*teeth*) surface. The L20 platen is electro-nickel plated on the sides and the bottom. The top and ends of the platen are treated with a chemical to prevent corrosion. However, over time (esp. in humid climates) the surface can corrode. You can use a polishing compound or lapping paper to remove oxidation spots (see below for recommended vendor). After polishing and/or lapping, the platen is more susceptible to corrosion, since the protective coating has been removed. Applying a paste wax to the top platen surface provides added protection in this circumstance.

If corrosion cannot be tolerated in your application, Compumotor can provide a magnetic stainless steel platen. The forcer's performance with this type of platen is reduced by 10% to 20%. Note that the lead time for special platens can be 12 to 16 weeks.

RECOMMENDED LAPPING PAPER

Compumotor recommends using *3M* 261X Imperial Lapping Film (8.5 x 11 in. sheets) A/O (Aluminum Oxide) 3.0 Mil. This Lapping film can also be used to remove oxidation spots on the platen.

Checking and Correcting Flatness

After the platen is mounted, the flatness of the platen surface should be checked. The simplest method is to manually push a non-energized L20 forcer (with air on) back and forth over the platen while checking where it rubs or binds on the platen. Rubbing spots are usually high spots. Another way of checking the platen flatness is to use a dial indicator referenced from a flat surface such as a granite surface plate. The most accurate way is to use a laser measurement system with flatness optics.

To lower the high spots, tighten the nearby bolts. To raise the low spots, loosen the nearby bolts. In worse-case situations, the platen may have to be shimmed or lightly lapped to obtain the required flatness.

Forcer Mounting

To place the forcer on the platen, refer to the Air Connection section in Chapter 2, Getting Started.

CAUTION

Do not energize the forcer when the forcer is not on the platen. Never disconnect the forcer when power is applied to the drive; this will damage the contacts to the motor connector.

Compumotor recommends installing rubber end-stops at each end of the platen to prevent the forcer from traveling beyond the platen's length. L20-AC forcers are shipped with rubber bumpers for added shock protection.

CAUTION

If the L20-AC forcer runs into a fixed metal object or into another forcer at high speeds, the impact can break the accelerometer.

Removing the L9A or L20 Forcer

Because of their large normal force (100 to 200 lbs), the L9A and L20 linear forcers are difficult to remove from the platen. The easiest way is to remove power to the drive and push the forcer (**with the air on**) off the end of the platen. The end of the platen should be deburred to prevent scratching the forcer air bearing surface. Another way to remove the forcer is to use a lever to pry the forcer off the platen. This should be done carefully so that the forcer will not be permanently warped, or the side roller bearings and/or side magnets damaged. Protect the platen surface when using the platen as a prying surface.

WARNING

Always remove the forcer with the air turned on. Do not remove the forcer with your hands. The sudden release of the forcer can cause personal injury.

Forcer Cable Mounting

The 12-foot cable supplied with the L20 forcer is highly flexible and abrasion resistant. A cable trough is an effective way to guide the forcer's cable. The cable is laid in a trough that runs parallel to the platen. As the forcer travels, the cable extends and lays out in the trough. As the forcer returns, the cable retracts and forms a vertical loop (see Figure 3-5).

The L20-AC cable is custom-designed for high-flex and long-life. However, be especially careful when mounting the cable in the machine. An L20 forcer running back and forth at full speed can make the cable whip back and forth in a manner that could stall the forcer.

If the cable bends while moving, make sure that the radius of the bend is ≥2 inches; avoid *kinking* the cable.

With the L20 forcer, air is routed through 1/8" inner diameter by 1/4" outer diameter urethane tubing. Jacketing the cable and tubing helps to avoid restricting forcer movement. Compumotor recommends braided expandable sleeving for such applications. This product is available from the following source (Specify: 1/2" GP, clear or black length):

Natvar Company,
Clayton, North Carolina
Phone: (919) 553-4151

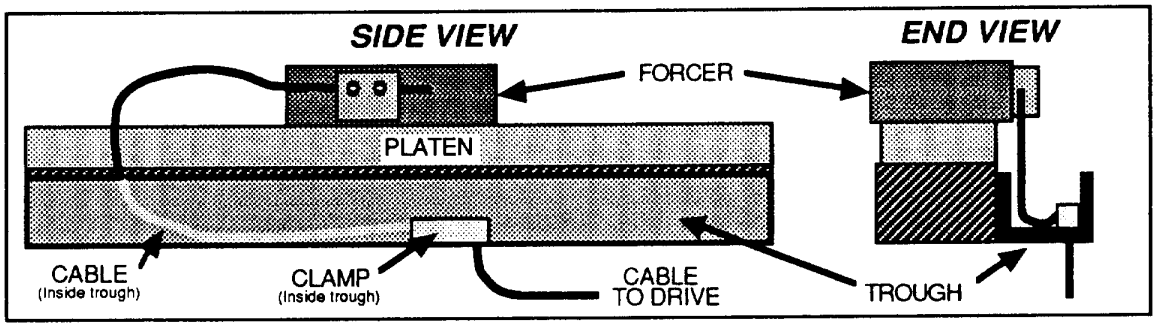


Figure 3-5. Cable Trough

-RFKC Option Assembly Instructions

The contents of the -RFKC Option kit are listed in Table 3-3. The kit provides a reliable air source for the L20 forcer. Figure 3-6 shows how the -RFKC components fit together.

Qty	Name	Function
1	Filter	Filters moisture and dust from air
1	Regulator	Regulates air pressure to the L20
1	Pressure Gauge	Visual pressure reading
1	Mounting Bracket	Mounts filter/regulator and gauge
1	Brass Quick Coupler	Connection to air supply source
2	Reducer	Reducer for hose fitting
1	Nipple	Connection between filter and regulator
1	Plug	Plugs into spare output in regulator
3	1/8" Male Hose Connector	Tubing connection
30 Ft.	Urethane Tubing	Routes air to L20
1	Roll of Teflon Tape	Seals threads
1	Assembly drawing and instructions	

Table 3-3. -RFKC Components

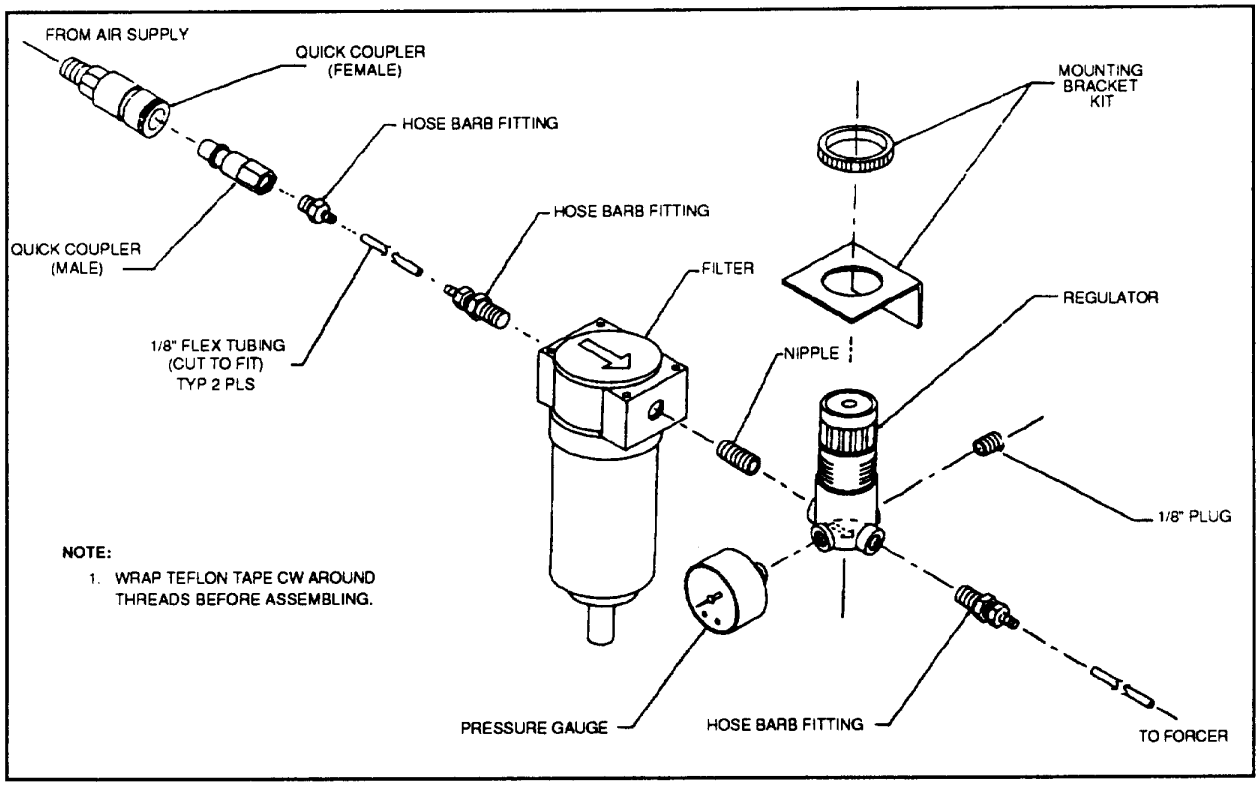


Figure 3-6. -RFKC Option Assembly

System Connections

Refer to Chapter 2, Getting Started, for instructions to connect the following:

- AC Power
- Fan (optional)
- Accelerometer (optional)
- Forcer
- Air

Detailed system pinouts are provided in Chapter 6, Hardware Reference.

CAUTION

Do not energize the forcer when the forcer is not on the platen. Be certain that AC power is disconnected before performing any wiring. Never disconnect the forcer when power is applied to the LX.

Grounding Information

Proper grounding of electrical equipment is essential to ensure the safety of personnel. The effects of electrical noise due to electromagnetic interference (EMI) can also be reduced by grounding. All Compumotor equipment should be properly grounded. A good source of information on grounding requirements is the National Electrical Code published by the National Fire Protection Association of Boston, Massachusetts.

In general, all components and enclosures must be connected to earth ground through a grounding electrode conductor to provide a low-impedance path for ground-fault or noise-induced currents. All earth ground connections must be continuous and permanent. Compumotor recommends a single-point grounding setup. You should prepare components and mounting surfaces prior to installation so that good electrical contact is made between mounting surfaces of equipment and enclosure. Remove the paint from equipment surfaces where the ground contact will be bolted to a panel and use star washers to assure solid, bare metal contact. The case of the forcer should be connected to the GND terminal (pin 3) on the LX motor connector. *This is done for you with Compumotor-supplied forcers.*

For temporary installation, or when the grounding method described above cannot be used, the GROUND terminal on the AC power connector must be connected to earth ground.

Wiring to Prevent EMI Problems

The LX delivers pulse-width-modulated (PWM) current to the forcer by switching 170VDC (120VAC input) at 20 kHz. The drive may radiate or conduct electrical noise along the forcer cable, through the forcer, and into the frame to which the forcer is attached. It can also be conducted out of the LX, into the AC power line. If electrical noise generated by the LX is a problem for other electronic equipment, take the following wiring precautions:

- Ground the forcer. This is already done for you with forcers that Compumotor supplies.
- Avoid extended forcer cable runs.
- Mount the electronic equipment affected by the noise as far as possible from the LX and forcer.
- Filter the power to the LX using a PI-type filter and an isolation transformer.
- Provide a separate power line for the LX. Do not use the same power circuit for the noise-sensitive electronic equipment and the LX.

I/O Connections

Figure 3-7 below illustrates the I/O connections to the LX auxiliary connector. Figure 3-8 is a diagram showing typical internal LX input and output circuits to the auxiliary connector.

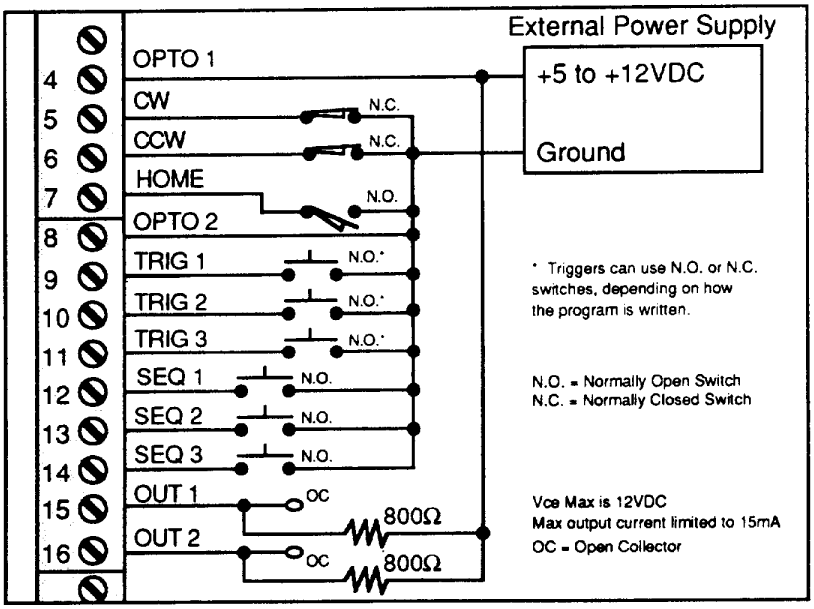


Figure 3-7. I/O Wiring Diagram

Typical I/O Circuits

All inputs and outputs are optically isolated. Regarding input switches, ***open*** means no current flows in the diode; ***closed*** means current flows.

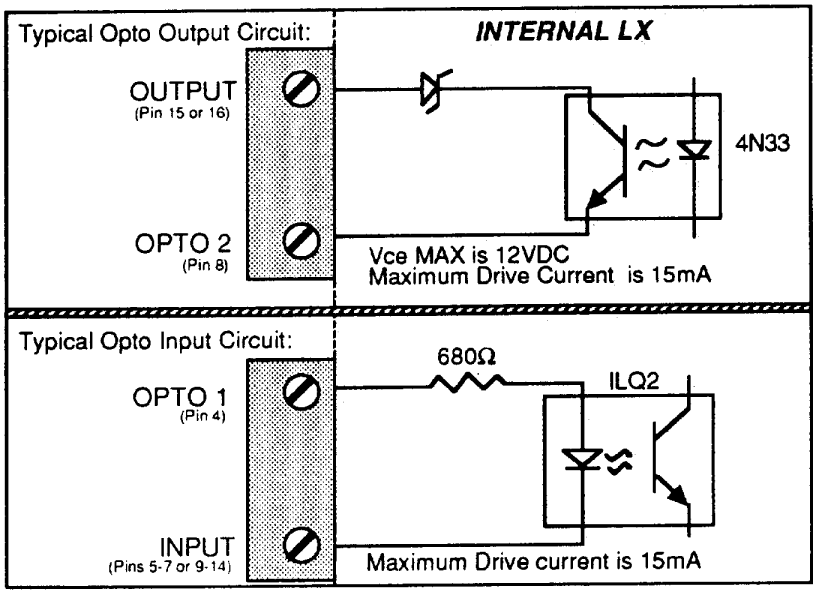


Figure 3-8. Typical Internal I/O Circuits

OPTO 1 & 2

These pins (Auxiliary connector pins 4 and 8) require an external power supply to the opto isolator on the LX to optically isolate the LX's I/O. *You cannot use I/Os without powering up the OPTOs.* Compumotor recommends that you use an external supply. Connect the positive voltage (5-12VDC) to the OPTO 1 input. Connect the return of your supply to OPTO 2 (refer to Figure 3-7).

End-of-Travel Limits

The CW (positive direction) and CCW (negative direction) limits serve as emergency stops to prevent the motor from traveling beyond the operational travel limits. The LX accepts only normally-closed limit switches. To use this feature, you must first connect an external 5-12VDC power supply to the OPTO 1 input (see Figure 3-7). When you close the switch, current flows through the diode of the OPTO isolator. When you open the switch, no current flows. ***In order to establish an external limit switch, the Enable Limits command (LDØ) must be entered.***

When the forcer is traveling in the direction of the active limit, the limit switch input (either CW or CCW) goes active and brings the forcer to an immediate stop. The forcer will not be able to move in the direction of the active limit until the limit input goes inactive. If the forcer is not moving in the direction of the active limit, motion will not be affected.

To test the system, you may use the 5VDC output from the encoder connector to power the OPTO 1 input. For daily system operation, you must use an external 5-12VDC supply for OPTO 1 power inputs. Connect the return of this supply to OPTO 2.

Home Input

You can use the home input (pin 7) to establish a home reference position. A normally-open, load-activated switch to the external supply ground is the most common way for determining the home position. Refer to Figure 3-7 for switch connections.

NOTE: Compumotor cannot guarantee proper homing performance with Home and End-of-Travel limits tied together.

Trigger Inputs

The LX has three trigger inputs. The trigger switches can be either normally-open or normally-closed, depending on how the program is written. Trigger inputs 1 - 3 are connected to pins 9 - 11 respectively (see Figure 3-7). The return (GND) of the external +5 to +12VDC power supply must be connected to the OPTO 2 input (Pin 8). The return line for the Trigger inputs is the OPTO 2 power source ground. Refer to the **TR** and the **TS** command descriptions in Chapter 5 for a functional description of these inputs.

Sequence Inputs

The LX has three Sequence Select inputs (SEQ1, SEQ2, and SEQ3). These inputs require that the return (GND) of the external +5 to +12VDC power supply be connected to the OPTO 2 input (pin 8). To select a sequence, you must ground one or more of the sequence inputs (closed switch). If all sequence switches are open, no sequence is selected. Refer to Figure 3-7 for sequence switch connections.

Outputs

The LX is equipped with 2 optically isolated programmable output bits. They are open-collector outputs and are normally not conducting current. Pins 15 and 16 are Outputs #1 and #2. These outputs may be used to signal a peripheral device that some event in the LX has just been completed, such as completing a move or meeting a required trigger input configuration. These outputs require and external 5-12VDC power supply connected to the OPTO 1input (the return of this supply must be connected to OPTO 2). An 800Ω pull-up resistor must also be connected to the external 5-12VDC power supply. *NOTE: The maximum current is limited to 15mA.* Refer to Figure 3-7 for output connections.

These outputs are controlled with the **O** command.

Auxiliary Drive Connections

When you command the LX to move a preset distance, the LX simultaneously produces step, direction, and remote power shutdown outputs through the 25-pin D connector marked **OPTIONAL MOTOR/DRIVER**. If you use this connector to run an additional external L Drive, its motor will run the same distance synchronously.

If you shut down the LX using the Shutdown (ST1) command, the L Drive connected to the LX via the 25-pin D connector will also shut down. You will not be able to control the step ratio or the velocity ratio; it can only be used for a 1:1 ratio.

The connector is normally used for an X-Y gantry type of application in which the LX must control another L Drive synchronously.

CAUTION

This connector must be used only to operate an external L Drive. Connecting anything else causes extensive damage to the LX and to the equipment you are trying to connect.

Extending Forcer Cables

The L20-AC cable assembly is part of the L20 forcer and should **not** be removed by the user. While the standard L20 cable assembly can be replaced in the field, the L20-AC cable assembly must be replaced at the factory. If you need shorter or longer cable assemblies, order them from Compumotor's Custom Product Group at (800) 358-9068. ***Do not attempt to shorten or lengthen the -AC cable assembly yourself.***

To extend the forcer cable for systems not using the -AC option, use the specifications shown in Table 3-4. When extending the forcer cable, be sure to shield your cable. Compumotor recommends that you use a mating connector when you extend the cable. A non-shielded cable will create electrical noise. Forcer cable color codes are as follows:

- Phase A+ = Red
- Phase A- = Black
- GND = Shield
- Phase B+ = White
- Phase B- = Green

Forcer	Maximum Current Per Phase (Amps)	Less Than 100 Ft (30.5M)	100 to 200 Ft (30.5M to 71M)
L3C	1.0	22 AWG	20 AWG
L5A	1.5	20 AWG	20 AWG
L9A	1.5	20 AWG	20 AWG
L20	2.7	20 AWG	18 AWG

Table 3-4. Current and Wire Size Specifications

Adjusting Air Pressure

Because of its higher force, the L20 air bearing is more sensitive to pressure changes than the L9A air bearing. There are several factors that determine what air pressure is required to make the forcer float properly. Some examples are as follows:

- Air hose length: Longer hoses require more pressure; shorter hoses require less pressure.
- Payload weight: A heavier payload requires more air pressure.
- Energized and non-energized forcers: A non-energized forcer requires less pressure than an energized one.
- Multiple forcers per regulator: More air pressure is required to float additional forcers using the same pressure regulator. Separate pressure regulators may be required, especially if the air hoses are of different lengths.
- Platen flatness: More air pressure is required to float a forcer on an uneven platen.

The best way to adjust the air pressure for the L20 forcer is to increase the air pressure until the forcer air bearing (with payload) becomes unstable (approximately 45 - 50 PSI). An unstable air bearing produces a loud humming sound. Decrease the air pressure by approximately 5 PSI or until the humming sound stops. In the non-energized state, the forcer should float freely over the entire platen.

If the air bearing is unstable only in certain areas of the platen, the problem may be caused by compromised platen flatness (refer to the Checking and Correcting Flatness section above).

Adjusting L5A Forcer Air Gap

Adjusting the air gap allows you to optimize the forcer's performance (speed and force). The L5A forcer allows you to mechanically adjust the air gap between the forcer and the platen by using the following procedure:

STEP 1 Remove the spring from the forcer's spring-loaded side bearing.

STEP 2 Use a piece of 0.0015 (1.5 mil) shim stock for forcer operation below 20 ips, or use 0.002" (2 mil) shim stock for continuous forcer operation more than 20 ips. Place the shim stock on the platen covering the platen teeth, but leave the bearing track open. Approximate dimensions of shim are 5" x 1 1/4".

STEP 3 Place the forcer on the platen over the shim stock. Make sure that the bearings are resting on the platen and not on the shim.

STEP 4 Loosen the four socket head screws on the two side sections of the forcer.

STEP 5 Press down firmly on the side section with the spring loaded bearings and tighten the socket head screws.

STEP 6 Tighten the screws on the other side section. This section has only one bearing in contact with the platen.

Compumotor properly adjusts the air gap before it ships the forcer. *You should have to adjust the air gap only if the forcer's performance deteriorates.*

Adjusting L20 Side Load Magnet Air Gap

The L20 side load magnet assembly is factory-set to provide a 7 to 10-lb side load for both roller bearings. *If the L20 forcer is run into a fixed object at a high speed, the side magnet assembly adjustment may be compromised. This can cause too little preload, or make the assembly touch (and scratch) the platen.* If this happens, the air gap must be reset according to the procedure below:

STEP 1 With the air on, place the forcer on the platen.

STEP 2 Turn the platen, with the forcer in place, upside down.

STEP 3 Loosen the three screws on the side magnet assembly. Make sure the side magnet assembly moves freely. *NOTE: In earlier forcer models, the side magnet assembly was tacked down with* ***TakPak.***

STEP 4 Push the forcer sideways so that the roller bearings make good contact with the side of the platen.

STEP 5 Turn off the air to the forcer.

STEP 6 Place a piece of 2 - 3 mil (0.002 - 0.003") shim-rock between the platen and the side magnet assembly.

STEP 7 Push the side magnet assembly against the platen and tighten the screws *while maintaining side pressure.*

STEP 8 Turn the air on and remove the shim-rock.

STEP 9 With a force gauge, check to see if the side load force is between 7 and 10 lbs. If the force is less than 7 lbs, repeat steps 3 through 8.

Verifying Proper Installation

Operation of the LX Indexer/Drive requires a minimum of three sets of electrical connections. These include an RS-232C serial communication device (which may be removed after you program the system), AC power to the drive, and the forcer. Other optional connections include CW and CCW end-of-travel limit inputs, a Home position limit input, Trigger and Sequence inputs, and Programmable Outputs.

To familiarize yourself with the operation of the system, you should complete these start-up procedures before you begin to install the unit. First, attach the forcer and RS-232C cables to the LX before applying AC power. The LX does not require a transformer to operate from 110VAC; however, Compumotor recommends that you use an isolation transformer. Be careful to keep the forcer away from any cables or other loose objects that could get tangled when the forcer moves.

Verify that all cables and wires are properly connected, and that the forcer is free from obstructions before you apply power to the system. If everything is OK, the drive will be enabled, the **POWER** LED will be on (green), and the **FAULT** LED will be off (not illuminated).

If the **FAULT** LED illuminates, refer to Chapter 7, Maintenance & Troubleshooting.

Verify RS-232C Interface

You should test for proper RS-232C communication before continuing with this section. Refer to the System Tests section in Chapter 2, Getting Started, for RS-232C testing procedures.

Utility Commands

The utility commands below may be useful throughout this section while testing your LX.

Command	Description
`> S`	Decelerates the forcer to a stop using the last defined acceleration value. The system executes this command immediately after you issue it.
`> K`	This is a ***panic*** stop. Halts the forcer immediately. *The immediate deceleration may cause a loss of position.* This command takes effect immediately after you issue it.
`> LD3`	Disables the limit switch functions. It allows forcer motion with no limits connected.
`> ^H`	Instructs the LX to backspace the cursor and delete the last character you entered (hold down the **ctrl** key and press the **H** key)
`> PS`	Pauses executing. Commands that follow the Pause command are not executed until the indexer receives a Continue (C) command to clear the pause and resume execution.
`> Z`	Resets the system (like cycling power)

Test Moves

This section contains examples of open loop moves that you can perform with the LX Drive. Open loop moves *do not* use an encoder to provide position information.

Testing Normal Mode Moves

To test the preset moves, enter the following string of commands:

Command	Description
> LD3	Disables limit switches (*use **only** if limits are not installed*)
> MN	Sets indexer to the normal mode
> A1	Sets acceleration to 1 g
> V1Ø	Sets velocity to 10 ips
> D1ØØØØ	Sets distance to 10,000 steps
> G	Executes the move (Go)

Assuming your resolution is 10,000 steps/in, the forcer moves 1 inch if you are using an L5A, L9A, or L20 forcer. The forcer moves 1.66 inches if you are using a L3C forcer. To move the forcer in the opposite direction, enter the following string of commands:

Command	Description
> D-1ØØØØ	Sets distance to 10,000 steps in the opposite direction
> G	Executes the move (Go)

The forcer moves the same distance in the opposite direction (using the previously-defined acceleration and velocity values).

Testing CW and CCW Limit Switches

Before you verify that the limit switches are working properly, check the following:

- Ensure that the CW (+ direction) and CCW (- direction) limit switches are wired properly (normally closed switches that open when the load moves to the limit position).
- Make sure that the load is not attached to the forcer.
- Make sure that the forcer can move the full length of the platen without running into other components.
- Make sure that you can manually open and close the limit switches.
- Make sure the forcer is positioned between both end-of-travel limits.

Use the following procedures to test the limit input switches:

STEP 1 Make sure both CW and CCW switches are closed (not tripped).

STEP 2 Type `1IS` . If all other inputs are opened, the response should be `*110000000`; this means that both CCW and CW limits (represented by the 1st and 2nd digits) are closed.

STEP 3 Open the CW limit.

STEP 4 Type `1IS`. Assuming no other inputs are closed, the response to this command should be `*100000000`.

STEP 5 Open the CCW limit switch.

STEP 6 Type `1IS`. The response should be `*000000000`.

To test the CW limit with the LX system, enter the following string of commands:

Command	Description
> LDØ	Enables CW and CCW limits
> A1	Sets acceleration to 1 g
> V1	Sets velocity to 1 ips
> D1ØØØØØ	Sets distance to 100,000 steps
> G	Executes the move (Go)

The `LDØ` command enables the limits so the forcer does not move unless the limit input is on. While the forcer is moving, open the CW limit switch. The forcer should come to an immediate halt.

To test the CCW limit, enter the following string of commands:

Command	Description
> A1	Sets acceleration to 1 g
> V1	Sets velocity to 1 ips
> D-1ØØØØØ	Sets distance to 100,000 steps in the opposite (negative) direction
> G	Executes the move (Go)

While the forcer is moving, open the CCW limit switch. The forcer should come to an immediate halt. If the forcer continues to move, open the CW limit switch. If the forcer stops when you open the CW limit input, switch the CW and CCW limit wires.

If neither of these limit switches stop the forcer, recheck your switch wiring and refer to Chapter 7, Maintenance & Troubleshooting.

Homing The Motor

You can initiate the Go Home function by issuing the Go Home (GH) command. When you issue the Go Home command, you must include the direction and velocity that the forcer should use to search for home.

When you command the LX to go home, the forcer begins to move in the direction and at the velocity you specified. It performs this move at the last defined acceleration rate, and looks for the home limit input to go active. If the forcer encounters an end-of-travel limit while it searches for home, it will reverse direction and look for the home limit input to go active in the opposite direction. If the forcer encounters the other limit before it detects the home signal, the Go Home move will be aborted and the forcer will stop.

To test the functionality of the home input switch, manually open the switch and type `1IS` . Assuming your end-of-travel limits are closed and all other inputs are open, the response will be `*000011000`. The fourth digit in the response represents the home input status. Now close the home switch and type `1IS`; the response should be `000111000`. This verifies that the switch is functioning properly.

Use the following procedure to test the LX's homing function:

STEP 1 Enter the following string of characters:

Command	Description
> `1GHF.2`	Sets final go home velocity to 0.2 ips
> `1GH+1`	Instructs the forcer to go home at 1 ips

The forcer moves in the positive direction at constant velocity (1 ips).

STEP 2 Momentarily ground (close) the home limit input.

The forcer decelerates to a stop, then moves in the opposite direction.

STEP 3 Momentarily close the home switch again to stop the forcer.

When the home limit input (pin 7) goes active after a **GH** command, the system recognizes the location where the input became active as *home*. The drive decelerates the forcer at the last rate specified. The LX will move the forcer back at a rate specified by the Go Home Final (**GHF**) command until the limit is again active. To reset the absolute counter to zero at the end of a go home move, issue the Position Zero (**PZ**) command.

Verify Inputs & Outputs Connections

This section presents information about the LX's inputs and outputs. The following inputs and outputs will be addressed:

- Trigger Inputs
- Programmable Outputs

Verify Trigger Inputs

The LX has three trigger inputs and three sequence inputs: TRIG 1, 2, and 3 (pins 9 - 11), and SEQ 1, 2, and 3 (pins 12 - 14). See Chapter 5, Software Reference, for a functional description of these inputs. You can also use the sequence select inputs for trigger functions. By using the Trigger (TR) command, you can use all six trigger and sequence inputs as trigger inputs.

Use the the following steps to verify that you have wired the trigger inputs properly.

STEP 1 Type `1TS`, if all the trigger and sequence inputs are open, you should get back the following response:

```
*000000[cr]
```

STEP 2 Close the TRIG 1 input switch

STEP 3 Type `1TS`. You should receive the following response:

```
*100000[cr]
```

STEP 4 This verifies that the TRIG 1 input is turned closed.

Repeat steps 1 through 4 for each input.

STEP 5 To verify that the Trigger function is operating properly, type the following:

Command	Description
> `LD3`	Disables all limits
> `A2`	Sets acceleration to 2 g's
> `V.5`	Sets velocity to 0.5 ips
> `D1ØØØØ`	Sets distance to 10,000 steps
> `TR1XXXXX`	Wait for Trigger Input 1 to turn on
> `G`	Executes the move (Go)

The forcer moves when you close the trigger 1 switch.

Verify Programmable Outputs

The LX is equipped with two programmable output bits. You may used them to signal a peripheral device that the LX has just completed some event. These outputs are controlled with the `O` or `IO` commands. These outputs are open collector outputs.

Perform the following steps to verify that you have wired the outputs properly:

STEP 1 Type `OØØ`.

The response from this command should be a high voltage equal to your OPTO 1 input voltage, as measured between OPTO 2 and Output #1 or #2.

STEP 2 You may change the voltage for outputs 1 and 2 to low voltage by entering the following command: `O11`.

Attaching the Load to the Forcer

The L3C, L5A, L9A, and L20 forcers all have threaded mounting holes located on the top surface (see Table 3-5). The payload should be rigidly attached to the forcer using the proper size hardware.

Forcer Type	Hole Specifications	# of Holes
L3C	8-32 UNC-2B Thru	4
L5A	6-32 UNC-2B X 0.25 Deep	4
L9A	6-32 UNC-2B X 0.21 Deep	4
L20	8-32 UNC-2B X 0.5 Deep	6

Table 3-5. Forcer Hole Specifications

There is an attractive force between the forcer and the platen that is almost ten times the forcer's rated force in the direction of travel. Without air pressure, you can achieve a magnetic force that is 20 times greater than the L20 or L9A forcer's maximum rated force. These forces are attained when the power is turned off. This magnetic attraction allows the forcer to operate in any spatial orientation and allows moment-producing forces to be placed on the forcer.

In some applications, such as when the load is not centered above the forcer's center of gravity, you must consider the forcer's *yaw, pitch, and roll* characteristics. If these torques applied to the forcer while accelerating and decelerating are strong enough, they will cause the forcer to lose index (stall) or even disengage from the platen.

The current revision of the Linear Motor Application software does not take into account any applied torques. Also, the payload is assumed to be centered on top of the forcer. Calculated accelerations will be less for off-centered payloads.

In X-Y gantry systems, the center of gravity of the Y-beam axis should be as close as possible to the platen surface of the two X-axis forcers (see Figure 3-9). The maximum yaw, pitch, and roll stiffness are specified in ounce-inches of external torque applied to the forcer. These values will help you determine the maximum applied torque at which the forcer still operates. These specifications are given in Chapter 6, Hardware Reference. Figure 3-10 illustrates the orientation of yaw, pitch, and roll.

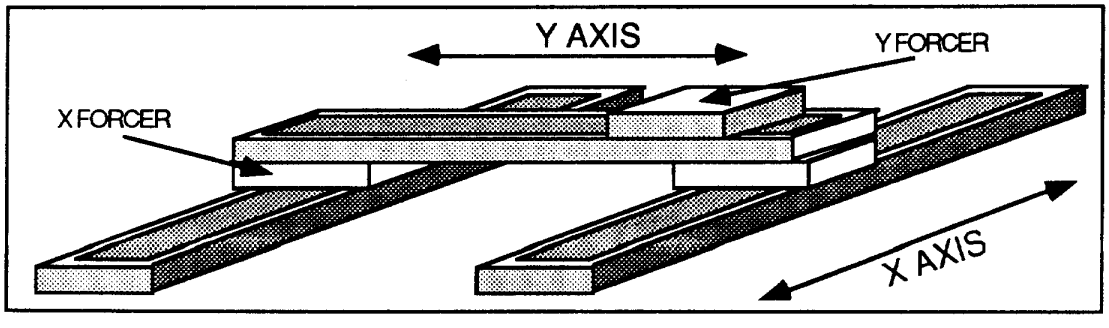


Figure 3-9. X-Y Gantry System

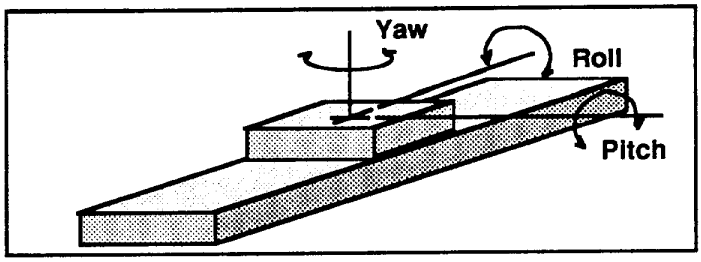


Figure 3-10. Orientation of Yaw, Pitch, and Roll

The following is a list of possible solutions to the problem of having the forcer (plus payload) exceed any of the torque specifications:

- Reduce the acceleration.
- Reduce the payload weight.
- Redesign the payload to bring the payload center of gravity closer to the platen surface (see Figure 3-11).
- Mount the payload on some sort of linear slide (e.g., Thompson rail) and use the forcer merely to push the payload back and forth on the linear slide. *When coupling the load to the forcer in this situation, you must allow some flexibility to compensate for that fact that the platen and the linear slide are not exactly parallel.* For more information, contact a Compumotor Applications Engineer at (800) 358-9070.

You must take special care when mounting a payload to a forcer, particularly the L9A and L20 forcer. **The area of the payload in contact with the forcer should be flat**. A *non-flat* payload, when torqued down, can warp the forcer and make it unable to float. If the payload covers the forcer opening, you can remove the forcer cover before attaching the payload.

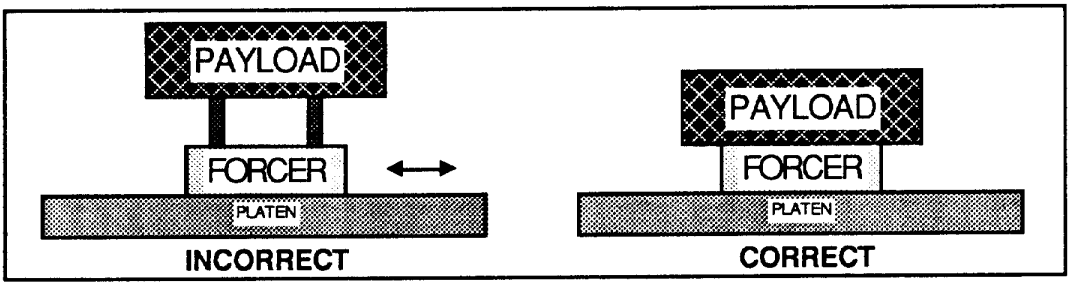


Figure 3-11. Proper Payload Mounting

In some applications, the existing roller bearing and side magnet assembly may not be strong enough. You can replace the existing arrangement with a custom side-roller bearing assembly. Compumotor's Custom Products Group can provide several alternative side-roller bearing arrangements.

The maximum recommended load on the side roller bearing assembly is 15 lbs. Exceeding the side load will cause premature platen wear. Because the platen is made of a low carbon unhardened steel (C1018), a harder roller bearing surface is needed for heavier side loads (e.g., a flat piece of hardened steel glued or bolted on side of platen).

Chapter 4. APPLICATION DESIGN

Chapter Objectives

The information in this chapter will enable you to do the following:

- Recognize and understand important considerations that must be addressed before you implement your application
- Understand the capabilities of the system
- Customize the system to meet your requirements

The sequence examples provided in this chapter assume that the LX device address is set to #1. *Refer to Chapter 6, Hardware Reference, for instructions on setting the device address.*

Application Considerations

Successful application of a linear motor system requires consideration of the following important points:

- Motion Profiles
- Resonance
- Ringing and Overshoot
- Move Times (calculated vs. actual)
- Positional Accuracy and Repeatability

Motion Profiles

A motion profile represents the velocity of the forcer during a period of time in which the forcer changes position. The type of motion profile needed will depend upon the motion control requirement. The basic types of motion profiles are described below.

Triangular and Trapezoidal Profiles

For constant acceleration indexing systems, velocity, acceleration, and distance parameters are defined before a preset move can be executed. The value of these parameters determines the type of motion profile as either triangular or trapezoidal. A triangular profile results when the velocity and acceleration are set such that the defined velocity is not attained before half of the specified distance has been traveled. This results from either a relatively low acceleration or a relatively high velocity or both. For example, if the acceleration is set to 1 g, velocity is set to 2 ips and distance is set to 10,000 steps (1 inch @ 10,000 steps/inch resolution), a triangular motion profile will result (by the time the forcer reaches the defined velocity of 2 ips, it will also have traveled half of the defined distance due to the acceleration setting of 1 g). The motion profile for this move is illustrated in Figure 4-1.

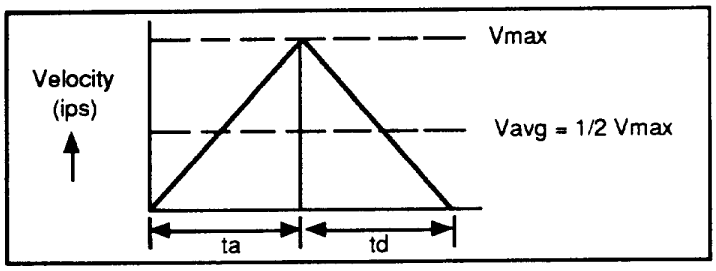


Figure 4-1. Triangular Profile

A trapezoidal move profile results when the defined velocity is attained before the forcer has moved half of the specified distance. This is due to a velocity that is relatively low, an acceleration that is relatively high, a move distance that is long, or a combination of all three. The resulting motion profile will resemble the profile shown in Figure 4-2.

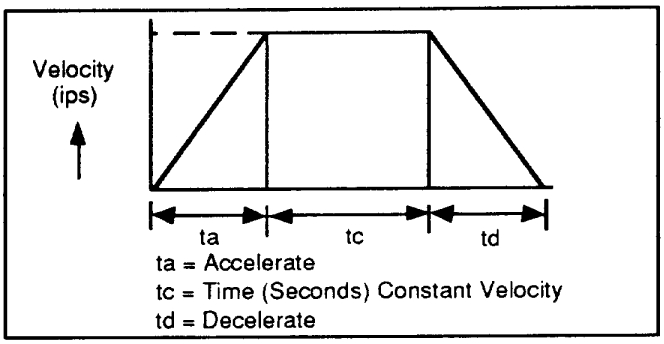


Figure 4-2. Trapezoidal Profile

Parabolic Profile

The parabolic velocity profile is used most often to reduce the acceleration time and move time of motors that are not already characterized by the LX (see Figure 4-3).

The LX software (demo diskette) can help you define a parabolic velocity profile.

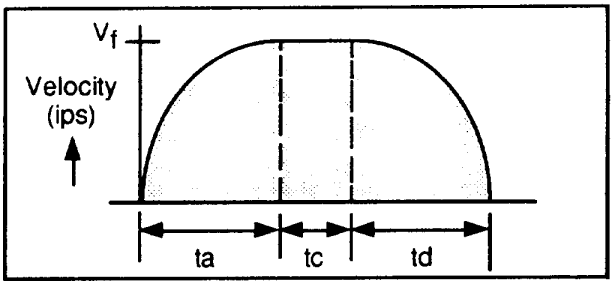


Figure 4-3. Parabolic Profile

S-Curve Profile

S-Curve velocity profiles are used often for gentle acceleration of liquid or fragile loads. The LX software (demo diskette) can help you set up the S-curve velocity profile (see Figure 4-4).

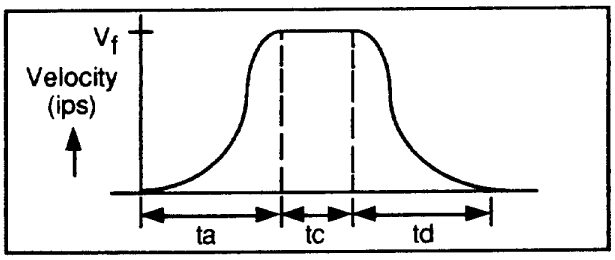


Figure 4-4. S-Curve Profile

Mechanical Resonance

Resonance is present in all stepper motors. If operated in a full-step or half-step mode, the forcer's performance is greatly reduced at speeds below about two ips. The forcer may even stall. The LX's microstepping capability allows smoother operation at low speeds.

Resonance in the L20 forcer may be remedied by adding the Accelerometer (-AC) Option.

-AC Option Reduces Resonance

The function of the Accelerometer option is to add *electronic friction* to the forcer, providing a convenient way to dampen motor resonance. By feeding back to the commanded position a displacement proportional to velocity, the -AC Option introduces the equivalent of *viscous friction* into the forcer's dynamics. While some systems obtain velocity information with a velocity transducer (tachometer), the -AC Option system obtains the velocity information by integrating the difference between the commanded acceleration and the measured acceleration from an accelerometer mounted in the L20 forcer. The velocity error is introduced into the commanded position via the servo circuitry. The magnitude of the velocity error feedback required for the fastest settling time (damping ratio of ≈ 0.707) is payload-dependent. The gain switches (-AC DIP switches 1 - 3) can be used to select the proper magnitude (see Chapter 6, Hardware Reference). Figure 4-5 illustrates how the -AC option provides smoother operation at velocities around the forcer's resonant frequency.

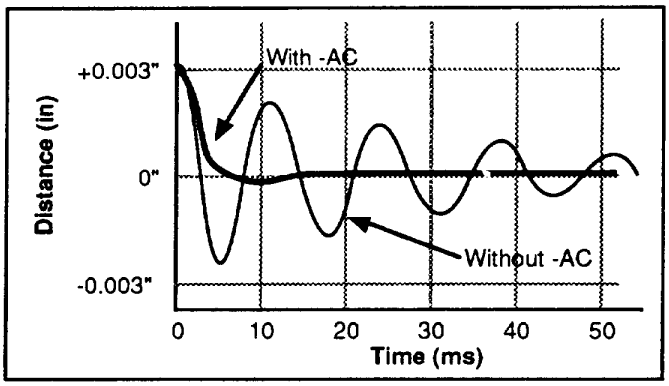


Figure 4-5. -AC Option Provides Smoother Operation

Constant Velocity Near the Resonant Frequency

The first step in determining if resonance will be a problem is to determine the forcer's resonant frequency. This value is predicted accurately with the following equation:

$$f = 2.26 \sqrt{Fmax / W * T}$$

Where: f = Natural resonant frequency in Hz
F_{max} = Static force of the forcer (lbs)
W = Weight of the forcer plus its load (lbs)
T = Tooth pitch of the platen (ft)
T(L5A) = T(L20) T(L9A) = 0.00333 ft
T(L3C) = 0.00555 ft

The equation can be further simplified to:

f = 40 * sqrt (Fmax / W) for the L5A, L9A and L20
f = 30 * sqrt (Fmax / W) for the L3C

Velocity ripple will be most noticeable at the resonant frequencies of the system. Once the resonant frequency is determined, the velocities that have first, second, and fourth harmonics of force ripple at the resonant frequency can be predicted. These three velocities are influenced by the tooth pitch of the platen and can be calculated as follows:

For the L5A, L9A and L20, velocities are f/25, f/50, and f/100. For the L3C, velocities are f/15, f/30, and f/60.

As a general rule, you should consider the accelerometer damping option for applications that require smooth velocity at speeds above 0.5 ips.

Ringing or Overshoot

The forcer's springiness and its mass form an underdamped resonant system that rings in response to acceleration transients (such as at the end of a move). Ringing at the end of a move prolongs settling time. The actual settling time of a system will depend on the forcer stiffness, the mass of the load, and any frictional forces that may be present.

-AC Option Reduces Settling Time

With the accelerometer option, the duration of settling at the end of a move is reduced by more than 90%. Figure 4-5 illustrates how the -AC option reduces settling time.

Positional Accuracy vs. Repeatability

In linear positioning systems, some applications require high absolute accuracy. Others require repeatability. You should clearly define and distinguish these two concepts when you address the issue of system performance.

If the positioning system is taken to a fixed place and the coordinates of that point are recorded, the only concern is how well the system repeats when you command it to go back to the same point. For many systems, what is meant by accuracy is really repeatability. Repeatability measures how accurately you can repeat moves to the same position.

Accuracy, on the other hand, is the error in finding a random position. For example, suppose the job is to measure the size of an object. The size of an object is determined by moving the positioning system to a point on the object and using the move distance required to get there as the measurement value. In this situation, basic system accuracy is important. The system accuracy must be better than the tolerance on the measurement that is desired.

Consult the technical data section of the *Compumotor Programmable Motion Control Catalog* for more information on accuracy and repeatability.

Open-Loop Accuracy

Open-loop absolute accuracy of a linear step motor is typically less than a precision grade lead screw system, but is better than most tangential drive systems.

The **worst-case accuracy** of the open-loop system is the sum of these errors. Accuracy = **A+ B+ C+ D + E**.

A **Cyclic Error**: The error that occurs due to motor magnetics. This error recurs once every pole pitch as measured on the body of the forcer.

B **Unidirectional Repeatability**: The error measured by repeated moves to the same point from different distances in the same direction.

C **Hysteresis**: The backlash of the forcer when it changes direction due to magnetic non-linearity and mechanical friction.

D **Thermal Expansion Error**: The error caused by a change in temperature, which expands or contracts the platen.

E **Random Platen Error**: The non-linear errors remaining in the platen after the thermal expansion error is disregarded.

Typical open loop accuracy for an L20 system is ±0.0035 in (90 microns) plus the effects of platen contraction and expansion due to ambient temperature changes and heating caused by forcer movement.

PLATEN THERMAL EXPANSION

The platen is manufactured at an ambient temperature between 65°F and 75°F (18°C - 24°C). The expansion rate is 0.000032 inches per 5°F (≈3°C) change. For example, a 36-inch platen, manufactured at 65°F (18°C), is 0.0012 inches longer at 70°F (21°C), an error of 0.003%.

High-speed moves (average velocity greater than 20 ips) at high duty cycles over distances of less than 2 feet can raise the platen temperature by 60°F (15.5°C). This kind of heating can make an 18-inch platen 0.007 inches longer (error of 0.039%).

Platen heating is caused by eddy current losses. To prevent over-heating, stepper motors normally have both the rotor and stator laminated. In linear motors, the forcer's magnetics is built around a laminated structure. For practical and cost reasons, the platen is made of a solid magnetic steel. As the forcer moves faster and the voltage across the forcer coils increases, more eddy currents are generated in the platen and heat the platen.

Use one or more of the following guidelines to minimize platen heating:

- Mount the platen on a good heat-conducting surface.
- Use a fan to increase the air flow over the forcer and the platen.
- Reduce the drive voltage. The LX bus voltage is 170VDC. With two forcers connected in series, the voltage over the forcer coils drops by 50% and the heating is reduced (as long as the forcers are not next to each other). Note that when two forcers are connected in series, the maximum possible velocity is reduced by 50%.
- Use the LX's Standby function to reduce the current while not moving.
- Reduce the average velocity. To calculate the average velocity, add all the move distances for a certain sequence of moves and divide the sum by the total time it takes to execute the sequence (including time delays). Use the following equation:

$$V_{average}\ [ips] = \frac{D_{total}\ [in]}{T_{total}\ [sec]}$$

Figure 4-6 illustrates a graph of platen temperature rise versus average velocity worse-case (platen length = 15 in).

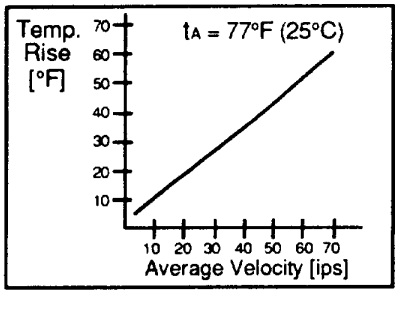


Figure 4-6. Platen Temperature Rise vs. Average Velocity

Modes of Operation

This section describes and compares the following LX operational modes:

- Interactive vs. Non-interactive
- Incremental vs. Absolute Positioning

Interactive & Non-interactive Modes

The LX indexer can be either interactive or non-interactive. In the Interactive mode, the LX returns a prompt (>) when the indexer is ready for the next command. You can use the Enable Interactive Mode (SSI1) command to make the LX drive interactive.

When you enable the Interactive mode, the indexer responds with a prompt (>) when it receives a valid command and a question mark (?) when it does not receive a valid command. If you enter a valid command, but enter an invalid range (e.g., A20), the LX will respond with a pound sign (#). These interactive responses are preceded with a carriage return and a line feed. While defining loops or sequences, no responses are received until the loop or sequence is executed or until the XT command is executed.

Normal (Preset) Mode

A preset move is a move distance that you specify in motor or encoder steps. You can select preset moves by putting the LX into normal mode using the Mode Normal (MN) command. Preset moves allow you to position the motor in relation to the motor's previous stopped position (incremental moves) or in relation to a defined zero reference position (absolute moves). You can select incremental moves by using the Mode Position Incremental (MPI) command. You can select absolute moves using the Mode Position Absolute (MPA) command.

Incremental Preset Mode Moves

When you are in the Incremental mode (MPI), a preset move moves the forcer the specified distance from its starting position. For example, to move the forcer 1.5 inches, a preset move with a distance of +15,000 steps (assuming a 10,000 steps/inch resolution) would be specified. Every time the indexer executes this move, the motor moves 1.5 inches from its resting position. Specify the direction by using the optional sign (e.g., D+2ØØØØ or D-1ØØØØ), or define it separately with the Set Direction (H) command (e.g., H+ or H-). Whenever you do not specify the direction, the unit defaults to the positive (CW) direction.

SAMPLE INCREMENTAL MODE MOVES

The moves shown below are incremental moves. The distance specified is relative to the motor's current position. This is the default (power-up) positioning mode.

Command	Description
> MPI	Sets unit to Incremental Position Mode
> A2	Sets acceleration to 2 g
> V15	Sets velocity to 15 ips
> D5ØØØ	Sets distance to 5,000 steps
> G	Executes the move (Go)
> G	Repeats the move (Go)
> H	Reverses direction of next move
> G	Executes the move (Go)

The motor moves 5,000 steps in the positive (CW) direction and stops. It then moves another 5,000 steps in the same direction and stops. The forcer then changes direction and moves 5,000 steps.

Command	Description
> D-25ØØØ	Changes the distance to 25,000 steps in the negative (CCW) direction.
> G	Executes the move (Go)

Assuming a resolution of 10,000 steps/inch, the forcer moves 2.5 inches using existing A and V values.

Command	Description
> H	Toggles the forcer direction in the next move, but maintains existing acceleration, velocity, and distance parameters.
> G	Executes the same move profile as the previous move, but in the opposite direction

The forcer moves 25,000 steps in the positive (CW) direction.

If you wish to load all the commands before executing them, you may use the Pause (PS) and Continue (C) commands.

Command	Description
> PS	Pauses execution until the indexer receives a Continue (C) command
> G	Executes the 25,000-step move (Go)
> T3	Waits 3 seconds after the move
> G	Moves 25,000 steps
> C	Starts G T3 G commands

Absolute Preset Mode Moves

A preset move in the absolute mode (MPA) moves the forcer the distance that you specify (in forcer steps) from the absolute zero position. You can set the absolute position to zero with the Position Zero (PZ) command or by cycling the power to the drive. The absolute zero position is initially the power-up position.

The direction of an absolute preset move depends upon the forcer position at the beginning of the move and the position you command it to move to. For example, if the forcer is at absolute position +12,800, and you instruct the forcer to move to position +5,000, the forcer will move in the negative direction a distance of 7,800 steps to reach the absolute position of +5,000.

The LX powers up in Incremental mode. When you issue the Mode Position Absolute (MPA) command, it sets the mode to absolute. When you issue the Mode Position incremental (MPI) command the unit switches to Incremental mode. The LX Indexer/Drive retains the absolute position, even while the unit is in the Incremental mode. You can use the Position Report (PR) command to read the absolute position.

The absolute position mode is automatically saved to EEPROM when you issue the Save (SV) command.

SAMPLE ABSOLUTE MODE MOVES

The moves shown below are absolute mode (MPA) moves. The distance specified is relative to the LX's absolute zero position.

Command	Description
> MN	Sets the LX to normal mode
> MPA	Sets the LX to the Absolute Position mode
> PZ	Sets the current absolute position to zero
> A1	Sets acceleration to 1 g
> V3	Sets velocity to 3 ips
> D5ØØØ	Sets move to absolute position +5,000
> G	Executes move (forcer moves to absolute position 5,000)

Command	Description
> D1ØØØØ	Sets the forcer to absolute position +10,000. (Since the forcer was already at position +5,000, it will move 5,000 additional steps in the + direction.)
> G	Executes the move (Go)
> DØ	Sets the forcer to absolute position 0. (Since the forcer is at absolute position +10,000, the forcer will move 10,000 steps in the - direction.)
> G	Executes the move (Go)

Program Control

This section discusses the program control features of the LX .

Triggers

You can use the Wait for Trigger (TR) command to specify a configuration of trigger conditions to be matched before executing a sequence of buffered commands. You can use TRIG1 - TRIG3 inputs and SEQ1 - SEQ3 inputs as trigger inputs. You may also use trigger inputs as Stop and Jog inputs, or as sequence-select inputs. Refer to the Input Mode (IM) command.

The three possible conditions you can specify are as follows:

- 1 = Wait for trigger input to be high (*open*, no current flows)
- Ø = Wait for trigger input to be low (*closed*, current flows)
- X = Ignore the trigger input

Command	Description
> IM1	Sets unit to Input Mode 1
> MN	Sets unit to Preset mode
> A2	Sets acceleration to 2 g's
> V5	Sets velocity to 5 ips
> D25ØØØ	Sets distance to 25,000 steps
> TRØ1XXXX	Waits for Trigger Input 1 to go high and Trigger Input 2 to go low
> G	Executes 25,000-step move (Go)

The TS command is useful for checking the status of the trigger inputs when it appears as though execution is being halted by the TR command and all conditions for matching the trigger input configuration defined by the TR command appear to be met.

POBs (Programmable Output Bits)

You can turn the programmable outputs on and off using the Output (O) and Immediate Output (IO) commands. One (1) turns on a given output and zero (Ø) turns the output off. The outputs conduct when they are on and do not conduct when they are off (see the O, IO, and OM command descriptions in Chapter 5, Software Reference). OM settings greater than 1 (OM2 through OM5) override the O and IO settings.

Using the Output Mode (OM) command, you can configure the output bits as a programmable output, forcer moving/not moving output, or forcer-at-constant-velocity output (refer to Table 4-1).

	OM1	OM2	OM3	OM4	OM5
OUT 1	Programmable	Prog.	Moving/not moving	Moving/not moving	Moving/not moving
OUT 2	Prog.	*Reserved*	Prog.	*Reserved*	Constant Velocity

Table 4-1. Output Mode Options

Moving : Output goes on (Current flows)
Not Moving: Output goes off (No current)
At Constant Velocity: Output goes on (Current flows)
Not at constant velocity: Output goes Off (No current)

EXAMPLE

Command	Description
> OM3	Out2 becomes a programmable output and Out1 becomes a motor moving/not moving output.

Delays

You can use the Time Delay (T) command to delay the execution of buffered commands in the indexer for a preset time.

In the normal mode (MN), the forcer finishes the move before the indexer executes the time delay.

Command	Description
> PS	Waits for the indexer to receive a Continue (C) command before executing next command
> G	Moves forcer 25,000 steps
> T5	Waits 5 seconds after the move ends
> H	Changes forcer direction
> G	Moves forcer 25,000 steps in the opposite direction
> C	Initiates command execution to resume

Branching

You can perform conditional branching with the Skip if Equal (SKE) and Skip if not Equal (SKN) commands. The syntax for these commands is similar to the Trigger (TR) command. When the LX encounters either of these commands, it evaluates the input pattern on TRIG1 - TRIG3 inputs and SEQ1 - SEQ3 inputs. The unit either skips the following command or executes the next command. Refer to Chapter 5, Software Reference, for a detailed description of the SKE and SKN commands.

Command	Description
> PS	Pauses command execution until the indexer receives a Continue (C) command
> D2ØØØØ	Sets move distance to 20,000 steps
> SKE1XXXXX	If Trigger Input 1 is on, skip the command following the SKE command
> G	Executes 20,000 step move (Go)
> D-2ØØØØ	Sets distance to -20,000 steps
> G	Executes the move (Go)
> C	Initiates command execution to resume

If Trigger 1 is on (1), the forcer only moves -20,000 steps. If Trigger 1 is off (Ø), the forcer moves 20,000 steps out and back.

Loops

You may use the Loop (L) command to repeat certain programs. You can nest Loop commands up to 24 levels deep.

Command	Description
> PS	Pauses command execution until the indexer receives a Continue (C) command
> MPI	Sets mode to incremental
> A2	Sets acceleration to 2 g's
> V5	Sets velocity to 5 ips
> L5	Loops 5 times
> D2ØØØ	Sets distance to 2,000 steps
> G	Executes the move (Go)
> T2	Delays 2 seconds after the move
> N	Ends Loop
> C	Initiates command execution to resume

The forcer moves a total of 10,000 steps

The example below shows how you can nest a small loop inside a major loop. In this example, the forcer makes 2 moves and returns a carriage return. The unit repeats these procedures and will continue to repeated until you instruct the unit to stop.

Command	Description
> PS	Pauses command execution until the indexer receives a Continue (C) command
> L	Loops indefinitely — Loop
> 1CR	Sends a carriage return
> L2	Loops Twice — Nested Loop
> G	Execute 2,000-step move — Nested Loop
> T.5	Waits 0.5 seconds — Nested Loop
> N	Ends loop — Nested Loop
> N	Ends loop — Loop
> C	Initiates command execution to resume

NOTE: The Y command only terminates the loop in progress. To stop execution of nested loops, you must issue the U, S, or K command, or use the SKN or SKE commands to exit loop(s).

Move Completion Signal

When you complete a move, you may use the LX's programming capability to signal the end of the current move. In a normal mode move, you may use one of the following commands:

- LF Line feed (see example #1)
- CR Carriage return (see example #2)
- O Output command (see example #3)
- " Quote command (see example #4)

EXAMPLE# 1

Command	Description
> PS	Pauses execution until indexer receives a Continue (C) command
> MN	Sets move to normal mode
> A2	Sets acceleration to 2 g's
> V.1	Sets velocity to 0.1 ips
> D125ØØ	Sets distance to 12,500 steps
> G	Executes the move (Go)
> 1LF	Sends a line feed over the RS-232C interface
> C	Cancels the Pause and executes the move

The forcer moves 12,500 steps. When you complete the move, the unit issues a line feed from the LX to the host over the RS-232C interface.

EXAMPLE #2

Command	Description
> PS	Pauses execution until indexer receives a Continue (C) command
> MN	Sets move to normal mode
> A2	Sets acceleration to 2 g's
> V.1	Sets velocity to 0.1 ips
> D125ØØ	Sets distance to 12,500 steps
> G	Executes the move (Go)
> 1CR	Sends a carriage return
> C	Cancels the Pause and executes the move

The forcer moves 12,500 steps. When the LX completes the move, the unit issues a carriage return from the LX to the host over the RS-232C interface.

EXAMPLE #3

Command	Description
> PS	Pauses execution until indexer receives a Continue (C) command
> MN	Sets move to normal mode
> OM1	Sets Output mode 1
> A2	Sets acceleration to 2 g's
> V.1	Sets velocity to 0.1 ips
> D125ØØ	Sets distance to 12,500 steps
> G	Executes the move (Go)
> O1X	Turns on Output 1
> C	Cancels the Pause and executes the move

The forcer moves 12,500 steps. When the LX completes the move, Output 1 is turned on.

NOTE: The OM3 command provides the fastest response to signal the completion of a move. This will turn off output #1 as soon as the last pulse has been output. No command execution time is required.

EXAMPLE #4

Command	Description
> PS	Pauses execution until indexer receives a Continue (C) command
> MN	Sets move to normal mode
> A2	Sets acceleration to 2 g's
> V.1	Sets velocity to 0.1 ips
> D125ØØ	Sets distance to 12,500 steps
> G	Executes the move (Go)
> "DONE	Sets the Quote DONE message
> C	Cancels the Pause and executes the move

The forcer moves 12,500 steps. When you complete the move, the unit issues the DONE message from the LX to the host over the RS-232C interface.

Sequences

A sequence is a series of commands. These commands are executed in their programmed order whenever the sequence is run. Immediate commands cannot be stored in a sequence, just as they cannot be stored in the command buffer. Only buffered commands may be used in a sequence.

Sequence Buffer vs. Command Buffer

Individual commands and command groups in any combination are entered into the command buffer until the total number of characters currently stored (including delimiters) fills the buffer. This buffer is 2,000 characters in size. The indexer uses a first-in-first-out serial buffer. As commands are read from the buffer, additional commands can be entered to replace them.

The sequence buffer is separate from the command buffer. The sequence buffer resides in non-volatile memory (EEPROM), while the command buffer resides in RAM. A copy of all information stored in EEPROM is stored in RAM. Parameters and sequences can be changed or deleted without affecting what is stored in the EEPROM until a Save (SV) command is executed.

40 sequences can be defined in the LX. The sequence length is not fixed, as long as the number of characters in all sequences totals less than 1,800 characters. Sequence 40, when defined, is executed on power-up and after the Reset (Z) command or XR40 command is executed. Any sequence (1 through 40) can be executed with the XR command.

In the IM1 mode (set with the IM1 command), only 7 sequences (1 - 7) can be selected externally via the SEQ 1, 2, and 3 inputs in the SSJ1 mode. In IM2 mode (set with the IM2 command), 39 sequences (1 - 39) can be selected via TRIG 1, 2, 3 and SEQ 1, 2, and 3 inputs in the SSJ1 mode. These inputs are BCD-weighted (see the IM command description in Chapter 5).

Sequences function like subroutines, as opposed to GOTOs, and can be nested up to 24 levels deep. Sequences and loops can be nested independently and with the SKN and SKE commands.

Sequence Programming

Use the following commands to define, erase, and run sequences.

Command	Description
> XBS	Reports the number of bytes available for sequence programming
> XD	Starts sequence definition
> XE	Deletes sequence from EEPROM
> XQ	Sets/resets interrupted Run mode
> XRP	Runs a sequence with a pause
> XT	Ends sequence definition
> XU	Uploads sequence
> XR	Runs a sequence
> XSR	Verifies execution of last sequence
> SSJ1	Runs a sequence defined by BCD sequence inputs

The commands that you enter to define a sequence are presented vertically in the examples below. This was done to help you read and understand the commands. When you are actually typing these commands into your terminal, they will be displayed horizontally.

It is a good practice to erase the sequence with the **XE** command before defining the sequence with the **XD** command. To begin the definition of a sequence, enter the **XD** command immediately followed by sequence identifier number (1 to 40) and a delimiter (pressing the space bar or the carriage return key). The **XT** command ends the sequence definition. All commands entered after the **XD** command and before the **XT** command are executed when the sequence is run. Examples are provided below.

Once you define a sequence, it cannot be redefined until you delete it. You can delete a sequence by entering the **XE** command immediately followed by a sequence identifier (1 to 40) and a delimiter. You may then redefine that sequence. You can use the **XBS** command to determine how many bytes are available in the sequence buffer. You can issue the Sequence Status Definition (**XSD**) command (preceded by a device address) to verify if the last sequence definition was successful. The possible responses seen on your terminal are ***0**, ***1**, or ***2**. A ***0** means the sequence was successfully defined. A ***1** means the sequence already exists with the number you have specified. A ***2** means there was not enough space in the sequence buffer for that sequence.

To check the status of a sequence, issue the Sequence Status (**XSS**) command. This command must be preceded by a device address and followed immediately by the number (1 to 40) of the sequence and a delimiter. The possible responses are ***0** (Empty), ***1** (Bad checksum), or ***2** (O.K.).

If you wish to check the contents of a sequence, enter the **XU** command. For example, issuing the **1XU1** command causes the LX to send the contents of sequence number 1 to the computer terminal's screen. The 1 preceding the **XU** command is the device address which must be present since the **XU** command is a *device-specific* command. We are assuming in these examples that the LX is set up at device address 1.

After you define the sequences over the RS-232C interface, you can execute the sequences by using one of the following modes of operation:

- *Stand-alone Operation*
- *Host Computer Operation*
- *Programmable Logic Controller (PLC) Operation*
- *Thumbwheel Operation*

SEQUENCE PROGRAMMING EXAMPLE

Command	Description
> **XE1**	Erases Sequence 1
> **XD1**	Begins definition of Sequence 1
A2	Sets acceleration to 2 g's
V1Ø	Sets velocity to 10 ips
D5ØØØ	Sets distance to 5,000 steps
G	Executes the move (Go)
H	Reverses direction
G	Executes the move (Go)
XT	Ends definition of sequence
> **SV**	Saves the sequence

Saving Programs

*Any command changes made to parameters using the commands demonstrated in the previous section are **NOT** permanent **UNTIL THEY ARE SAVED.** To make a change permanent, you must issue the Save (**SV**) command. The **SV** command will save all changes that have been made into EEPROM. Thus, if you make changes with these commands and the machine is reset, the changes are lost unless you issued the **SV** command before the machine resets.*

Standalone Operation

This section explains and provides examples of how to store programs and run them with remote switches, and run them automatically when you power up the system. First, you will need to enter the programs into the LX. You will need a terminal or a computer with RS-232C communication capabilities for programming the LX.

Power-Up Sequence Execution

You can program the LX to execute a sequence of commands on power-up (sequences can be used as subroutines).

Sequence 40 always runs on power-up. To run another sequence on power up, place an XR command (followed by the number of the sequence you want executed) at the end of sequence 40 as demonstrated below. If sequence 40 is empty, nothing happens on power-up.

Command	Description
> XE40	Erases Sequence 40
> XD40	Begins definition of sequence 40
A2	Sets acceleration to 2 g's
V5	Sets velocity to 5 ips
D125ØØ	Sets distance to 12,500 steps
G	Executes the move (Go)
XR4	Runs sequence #4
XT	Ends sequence definition
SV	Saves the sequence
> Z	Resets the indexer and runs Sequence 40

A power-up sequence is typically used for storing set-up or initialization parameters that are always necessary in your application.

Input Modes

The IM command specifies how you may define the three sequences and three trigger inputs. In Mode 1 (IM1), the three sequence inputs are used to select sequences 1 through 7 (see Table 4-2). In Mode 2 (IM2), all six inputs are used to select sequences 1 through 39 (see Table 4-3). Mode 3 (IM3) dedicates the triggers for use as Stop, Jog+, and Jog-.

The IM command can be executed within the sequence to change the IM mode.

In input modes one through three, the SSJ1 command allows sequences to be selected by grounding the appropriate inputs. If the XQ command is issued, the input lines must be released (turned off) before another sequence may be run.

The SN command is used to determine how long the input must be stable before executing the sequence.

Selecting Sequences

To call different sequences, you must ground the proper inputs. Table 4-2 shows the LX's input patterns to select one of 7 different sequences, and Table 4-3 shows the LX's input patterns to select one of 39 different sequences (IM2 Mode). These inputs are BCD-weighted (see the IM command description in Chapter 5). You must make sure that OPTO 1 is connected to a 5 - 12VDC power source. OPTO 2 must be connected to the return of this supply. *NOTE: You must issue the SSJ1 command to enable the remote sequencing function.*

Sequence #	SEQ 1	SEQ 2	SEQ 3
0*	OFF	OFF	OFF
1	ON	OFF	OFF
2	OFF	ON	OFF
3	ON	ON	OFF
4	OFF	OFF	ON
5	ON	OFF	ON
6	OFF	ON	ON
7	ON	ON	ON

* Non-valid sequence (no sequence is selected)

Table 4-2. Sequence Selection Table (IM1 Mode)

Sequence #	TRIG 1	TRIG 2	TRIG 3	SEQ 1	SEQ 2	SEQ 3
0*	OFF	OFF	OFF	OFF	OFF	OFF
1	OFF	OFF	OFF	OFF	OFF	ON
2	OFF	OFF	OFF	OFF	ON	OFF
3	OFF	OFF	OFF	OFF	ON	ON
4	OFF	OFF	OFF	ON	OFF	OFF
5	OFF	OFF	OFF	ON	OFF	ON
6	OFF	OFF	OFF	ON	ON	OFF
7	OFF	OFF	OFF	ON	ON	ON
8	OFF	OFF	ON	OFF	OFF	OFF
9	OFF	OFF	ON	OFF	OFF	ON
10	OFF	ON	OFF	OFF	OFF	OFF
11	OFF	ON	OFF	OFF	OFF	ON
12	OFF	ON	OFF	OFF	ON	OFF
13	OFF	ON	OFF	OFF	ON	ON
14	OFF	ON	OFF	ON	OFF	OFF
15	OFF	ON	OFF	ON	OFF	ON
16	OFF	ON	OFF	ON	ON	OFF
17	OFF	ON	OFF	ON	ON	ON
18	OFF	ON	ON	OFF	OFF	OFF
19	OFF	ON	ON	OFF	OFF	ON
20	ON	OFF	OFF	OFF	OFF	OFF
21	ON	OFF	OFF	OFF	OFF	ON
22	ON	OFF	OFF	OFF	ON	OFF
23	ON	OFF	OFF	OFF	ON	ON
24	ON	OFF	OFF	ON	OFF	OFF
25	ON	OFF	OFF	ON	OFF	ON
26	ON	OFF	OFF	ON	ON	OFF
27	ON	OFF	OFF	ON	ON	ON
28	ON	OFF	ON	OFF	OFF	OFF
29	ON	OFF	ON	OFF	OFF	ON
30	ON	ON	OFF	OFF	OFF	OFF
31	ON	ON	OFF	OFF	OFF	ON
32	ON	ON	OFF	OFF	ON	OFF
33	ON	ON	OFF	OFF	ON	ON
34	ON	ON	OFF	ON	OFF	OFF
35	ON	ON	OFF	ON	OFF	ON
36	ON	ON	OFF	ON	ON	OFF
37	ON	ON	OFF	ON	ON	ON
38	ON	ON	ON	OFF	OFF	OFF
39	ON	ON	ON	OFF	OFF	ON

OFF = open switch high (not pulled to ground)
ON = closed switch low (pulled to ground)

* Non-valid sequence

Table 4-3. Sequence Selection Table (IM2 Mode)

Using Remote Sequence Inputs

The following are step-by-step procedures demonstrating how to remotely execute sequences. Using a terminal or a computer, key in the following commands:

STEP 1 Define a power-up sequence

Command	Description
> XE40	Erase sequence 40
> XD40	Define sequence 40
IM3	Configure LX to Input mode 3
SN1ØØ	Set scan time to 100 milliseconds
SSJ1	Set LX to execute programs from remote interface
XQ1	Enable Sequence Hold
XT	End Sequence Definition

The IM2 command selects the input mode for externally selecting up to 39 pre-defined sequences. ***Sequence 40 cannot be executed through remote inputs.***

STEP 2 Define all sequences that your application may require.

Command	Description
> XE1	Erase Sequence 1
> XD1	Define Sequence 1
A1	Set acceleration to 1 g
V2	Set velocity to 2 ips
D1ØØØØ	Set Distance to 10,000 steps
G	Executes the move (Go)
XT	End sequence definition

Command	Description
> XE2	Erase Sequence 2
> XD2	Define Sequence 2
A1	Set acceleration to 1 g
V2	Set velocity to 2 ips
D-1ØØØØ	Set Distance to 10,000 - steps
G	Executes the move (Go)
XT	End sequence definition

STEP 3 Save the commands by issuing the SV command.

STEP 4 Verify that your programs were stored properly by uploading each entered sequence (XU command preceded by the device address and followed by the number of the sequence).

Command	Response
1XU1	*A1_V2_D10000_G
1XU2	*A1_V2_D-10000_G

If you receive responses that differ from what you programmed, re-program those sequences.

STEP 5 Make sure all the sequence select and trigger inputs are off (not grounded).

STEP 6 Reset the LX by cycling power or entering the **Z** command.

STEP 7 To run sequence #1, momentarily close the switch from SEQ 3 to GND, then open the switch. The motor should move 10,000 steps in the positive direction.

STEP 8 To run sequence #2, momentarily close the switch from SEQ 2 to GND, then open the switch. The motor should 10,000 steps in the opposite (negative) direction.

Using the Remote Stop Input

When you use Input Mode 3 (IM3), the LX reconfigures the TRIG1 input as a stop input. You should not use the TR command. It conflicts with the stop.

Trigger Input 1 causes the forcer to stop, if it is moving, as if you had issued an S command (i.e., the forcer makes a controlled stop).

Host Computer Operation

Immediate Sequence Execution

You can execute a sequence by entering the XR command immediately followed by a sequence identifier number (1 to 40) and a delimiter. The sequence will be executed immediately after the delimiter.

You can issue the Sequence Status Run (XSR) command to verify if the last sequence you issued was executed successfully. The possible responses are as follows:

- *Ø Last attempt to run a sequence was successful
- *1 Last attempt to run a sequence was unsuccessful
- *2 Invalid sequence number was requested

LX Application Software

An IBM-compatible application software diskette is provided with each LX. You can use this software for the tasks listed below:

1. Getting Started
   - System Components
   - Hooking up the system
   - Mounting the system
   - Setting DIP Switches
   - Hooking up RS-232C Communications

2. Sizing and Selection of Linear Motors
   - Single-Axis Application
   - XY Gantry Application

3. LX Programming
   - Setting up IBM/Compatible as a dumb terminal for communicating over RS-232C interface.
   - Creating and Editing LX Programs
   - Uploading & Downloading programs

This software is a user-friendly, menu-driven program that guides you through all the options. To start the program, all you need to do is to type **RUN** from your IBM DOS Operating System. Use the arrow keys to move to the appropriate pull down menus.

Single Axis Control

This manual generally assumes that you are using a single LX. Most programming examples are designed with this in mind and assume your LX is at Address #1. The principles developed for a single axis system apply as well to multi-axis systems.

SINGLE-AXIS INTERFACE PROGRAM EXAMPLE

If you already have BASICA or GW BASIC programming languages on your computer, you may use the following sample program designed to open a serial communication port and send and receive LX commands. The program performs the following steps:

- Executes the first move upon user input
- Waits for a line feed from the LX drive, which indicates the end of the move.
- Executes the second move upon user input.
- Waits for a line feed from the LX Drive, which indicates the end of the move. It then begins the process again.

This application can be looked on as moving a part out, machining the part, then bringing the part back.

```
1 '            LX.BAS PROGRAM
2 '
3 '
4 '  ********************************************************************************************************
5 '  * This program controls the RS232 Communication line to execute 2                                      *
6 '  * different moves using the LX                                                                         *
7 '  *                                                                                                      *
8 '  ********************************************************************************************************
15 OPEN "COM1:9600,N,8,1,RS,CS,DS,CD" AS #1              ' Open Communication port
20 V$ = "": Q$ = "": ECHO$ = "": LF$ = "":               ' Initialize variables
90 CLS
100 LOCATE 12,15
105 PRINT " PRESS ANY KEY TO START THE PROGRAM "
107 V$ = INKEY$: IF LEN(V$) = 0 THEN 100                 ' Wait for input from user
120 Z$ = "Z"                                             ' Reset the LX indexer
122 PRINT #1,Z$;
124 Q$ = INPUT$(2,1)
900 '
901 '  ********************************************************************************************************
902 '  * Line 1000-1060 sends a move down to the first LX.  Computer                                          *
903 '  * waits for the Line Feed from the LX indicating that the motor                                        *
904 '  * has finished its move.  Computer will not command second LX to move                                 *
905 '  *                                                                                                      *
906 '  ********************************************************************************************************
1000 MOVE1$ = "1A1 1V2 1D25,000 1G 1LF "          ' Define move for Axis 1
1005 CLS
1007 LOCATE 12,15: PRINT " DOING MOVE 1 "
1010 PRINT #1,MOVE1$                             ' Move axis 1.
1015 ECHO$ = INPUT$(23,1)                        ' Read echoes from LX
1020 LF$ = INPUT$(1,1)                           ' Wait for line feed from LX
1040 IF LF$ <> CHR$(10) GOTO 1020                ' indicating end of move.
1045 CLS
1047 LOCATE 12,15
1050 PRINT "MOVE 1 DONE"                         ' Let user know axis 1 is done
1060 LOCATE 15,15: PRINT " PRESS ANY KEY TO GO ON TO SECOND MOVE "
1070 V$ = INKEY$: IF LEN(V$) = 0 THEN 1060
1900 '  ********************************************************************************************************
1901 '  *                                                                                                      *
1902 '  * After axis one is done, we request that you press any key to go on                                   *
1903 '  * to the second move.  In real application, we would expect you to                                     *
1904 '  * go ahead with the process and work on the part before going on to                                    *
1905 '  * next move.  (i.e.  Activate a punch)                                                                 *
1906 '  *                                                                                                      *
1907 '  * Now that first move is finished, we go on to move #2.                                               *
1908 '  * LX also prints a line feed after finishing the second move.                                         *
1909 '  * As soon as computer receives the line feed from LX, program will                                    *
1910 '  * go back to the first move.                                                                           *
1911 '  *                                                                                                      *
1912 '  ********************************************************************************************************
2000 MOVE2$ = "A10 V5 D-5000 G H G 1LF"
2005 CLS
2007 LOCATE 12,15: PRINT " DOING MOVE 2 "
2010 PRINT #1,MOVE2$
2015 ECHO$ = INPUT$(25,1)
2020 LF$ = INPUT$(1,1)
2040 IF LF$ <> CHR$(10) GOTO 2020
2045 CLS
2047 LOCATE 12,15
2050 PRINT "MOVE 2 DONE "
2060 FOR I = 1 TO 1000: NEXT I
2070 GOTO 20                          ' Go back to beginning of program.
```

Multi-Axis Control (Daisy Chaining)

You may daisy chain LX units as described in Chapter 6, Hardware Reference. Individual drive addresses are set with the LX's DIP switches. When daisy chained, the units may be addressed individually or simultaneously. You should establish a unique device address for each LX with internal DIP switches. Figure 4-7 illustrates how LX's must be wired in a daisy chain configuration.

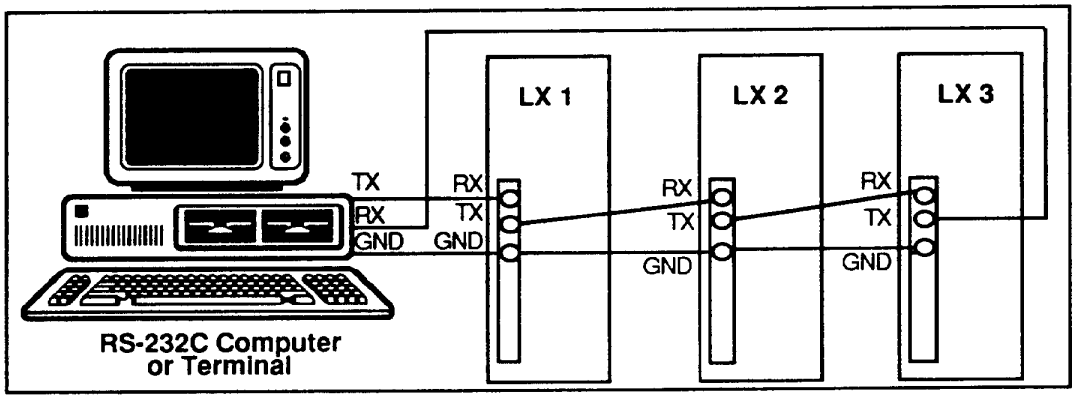


Figure 4-7. Daisy Chained LX Indexer/Drives

Device-specific commands require that a device address precede them. The LX will not execute a device-specific command if there is no device address preceding the command, or if the device address setting in the LX does not match the address preceding the command. *Universal* commands do not require device identifiers preceding them. A universal command with no device address will be executed regardless of the address setting of the LX. If a device address does precede a universal command, it will only be executed by an LX set to that particular address.

The **E** (Enable RS-232 communication) and **F** (Disable RS-232 communication) commands are useful in a daisy chain for locking out particular indexers from responding to universal commands with no preceding device address.

*NOTE: The **F** command will keep the LX from executing any commands (except the **E** command) sent to it over the RS-232C interface, but will not prevent the command from being echoed.*

For Example, to lock-out the LX unit set to device address 1, so that universal commands are only executed by the LX's set to address 2 and 3, the following step is performed:

- Send the **1F** command over the RS-232C communication line, locking out the LX at device address 1.

All universal commands (with no preceding device address) will now be executed only by the LX's at Device addresses 2 and 3. Entering a **2F** command in addition to **1F** command would allow only the LX at device address 3 to execute universal commands with no preceding device address. This eliminates the need to precede every command intended for a specific LX (in this case, the LX at device address 3) with a device address. Sending an **E** command over the RS-232C line will re-enable all drives previously disabled with the **F** command. Preceding the **E** command with a device address will re-enable only the LX set to that particular device address.

If you enable the Interactive mode (**SSJ1**), only the LX that is set to Address 1 will respond with the prompt (>). This prevents all the LX's from sending out > in a daisy chain. However, to prevent RS-232C communication problems, do not use the interactive mode when the LX units are daisy-chained.

If you daisy-chain, be sure to repeat the Verifying Proper Installation section in Chapter 3 for each axis.

SAMPLE APPLICATION AND COMMANDS

Example: Three indexers are on an RS-232C daisy chain. Send the following commands:

Command	Description
> **MN**	Sets unit to Preset mode
> **A2**	Sets acceleration to 2 g's for all three indexers
> **V5**	Sets velocity to 5 ips for all three indexers
> **1D25ØØØ**	Sets Axis 1 distance to 25,000 steps
> **2D5ØØØØ**	Sets Axis 2 distance to 50,000 steps
> **3D1ØØØØØ**	Sets Axis 3 distance to 100,000 steps
> **G**	Moves all axes.

Unit 1 moves 25,000 steps, unit 2 moves 50,000 steps, and unit 3 moves 100,000 steps. All three units use the same acceleration and velocity rates. Unit 2 starts about 2 ms later than unit 1. Unit 3 starts about 4 ms later than unit 1.

MULTI-AXIS INTERFACE PROGRAM EXAMPLE

The following program is very similar to LX.BAS, except this program controls 2 LXs on a daisy chain. This program assumes the device address of the two LXs to be 1 and 2 respectively. The program does the following:

- Executes the first move upon user input
- Waits for a line feed from the LX, which indicates the end of the move
- Executes the second move upon user input
- Waits for a line feed from the LX, which indicates the end of the move. It then begins the process again.

```
1 '            LX2.BAS PROGRAM
2 '
3 '  ***************************************************************************************
4 '  * This program controls the RS232 Communication line to execute 2                     *
5 '  * different moves using 2 LX units..                                                  *
6 '  *                                                                                     *
7 '  ***************************************************************************************
8 '
15 OPEN "COM1:9600,N,8,1,RS,CS,DS,CD" AS #1          ' Open Communication port
20 V$ = "": Q$ = "": ECHO$ = "": LF$ = "":           ' Initialize variables
90 CLS
100 LOCATE 12,15
105 PRINT " PRESS ANY KEY TO START THE PROGRAM "
107 V$ = INKEY$: IF LEN(V$) = 0 THEN 100             ' Wait for input from user
120 Z$ = "Z "                                        ' Reset the LX indexer
122 PRINT #1,Z$;
124 Q$ = INPUT$(2,1)
900 '
901 '  ***************************************************************************************
902 '  * Line 1000-1060 sends a move down to the first LX.                                   *
903 '  * Computer waits for the Line Feed from the LX indicating that the motor              *
904 '  * has finished its move.                                                              *
905 '  *                                                                                     *
906 '  ***************************************************************************************
1000 MOVE1$ = "1A1 1V2 1D25,000 1G 1LF "             ' Define move for Axis 1
1005 CLS
1007 LOCATE 12,15: PRINT " MOVING AXIS 1 "
1010 PRINT #1,MOVE1$                                 ' Move axis 1.
1015 ECHO$ = INPUT$(22,1)                            ' Read echoes from LX
1020 LF$ = INPUT$(1,1)                               ' Wait for line feed from LX
1040 IF LF$ <> CHR$(10) GOTO 1020                    ' indicating end of move.
1045 CLS
1047 LOCATE 12,15
1050 PRINT "LXIS 1 FINISHED ITS MOVE "               ' Let user know axis 1 done
1060 LOCATE 15,15: PRINT " PRESS ANY KEY TO MOVE SECOND AXIS "
1070 V$ = INKEY$: IF LEN(V$) = 0 THEN 1060
1900 '  ***************************************************************************************
1901 '  *                                                                                     *
1902 '  * After axis one is done, we request that you hit any key to go on                   *
1903 '  * to second move. In real application, we would expect you to                        *
1904 '  * go ahead with the process and work on the part before going on to                  *
1905 '  * next move. (i.e. Activate a punch)                                                  *
1906 '  *                                                                                     *
1907 '  * Now that first axis finished its move, we go on to move axis 2.                    *
1908 '  * Second LX also prints a line feed after finishing the move.                        *
1909 '  * As soon as computer receives the line feed from LX, program will                   *
1910 '  * go back to the first move.                                                          *
1911 '  *                                                                                     *
1912 '  ***************************************************************************************
2000 MOVE2$ = "2A1 2V2 2D5,000 2G 2LF "
2005 CLS
2007 LOCATE 12,15: PRINT " AXIS 2 MOVING "
2010 PRINT #1,MOVE2$
2015 ECHO$ = INPUT$(22,1)
2020 LF$ = INPUT$(1,1)
2040 IF LF$ <> CHR$(10) GOTO 2020
2045 CLS
2047 LOCATE 12,15
2050 PRINT "AXIS 2 FINISHED ITS MOVE "
2060 FOR I = 1 TO 1000: NEXT I
2070 GOTO 20                          ' Go back to beginning of program.
```

PLC Operation

You can use a PLC to execute 39 different sequences that are stored in the LX Drive. Six outputs from the PLC can be used to execute sequences and two inputs to the PLC can be used to monitor the LX indexer/drive.

PLC Connections

Assuming your PLC accepts open-collector outputs, connect the inputs and outputs as shown in Figure 4-8. If not, contact a Compumotor Applications Engineer at (800) 358-9070.

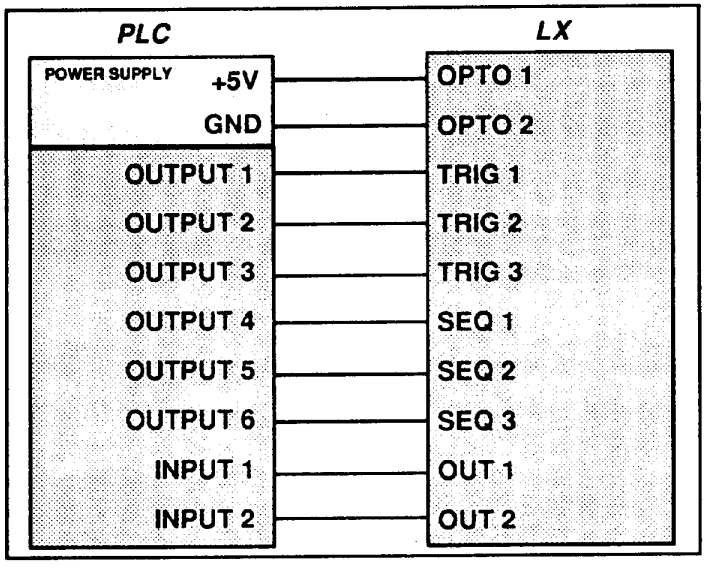


Figure 4-8. PLC Connections

Scanning for Sequence Execution

Changing the BCD values of sequence input lines results in a new sequence being run that corresponds to the new value. The sum of the values issued determines which sequence the indexer will run. For example, turning on SEQ2 and SEQ3 executes Sequence 3.

The Scan (SN) command determines how long the sequence select input must be maintained before the indexer executes the program. This is a debounce time.

Sample Applications and Commands

This section provides step-by-step procedures to run sequences from your PLC. First, you need to enter the programs into the LX drive. You will need a terminal or a computer with RS-232C communication capability. You need to define the sequences before you can execute them with your PLC's BCD outputs.

Using a terminal or a computer, key in the commands below.

STEP 1 Define a power-up sequence as follows:

Command	Definition
> XE40	Erase sequence 40
> XD40	Define sequence 40
IM2	Change to input mode 2
SSJ1	Execute sequences vis PLC input
SN20	Set scan time to 20 msec
XQ	Set LX to interrupted run mode
A1	Set acceleration to 1 g
V20	Set velocity to 20 in/sec
XT	End sequence definition

Every time you power up the LX, the indexer executes these commands and enables the LX to read up to 39 sequences from the sequence select inputs.

STEP 2 Define any sequences that your application may require.

Command	Description
> XE1	Erases Sequence 1
> XD1	Defines Sequence 1
A1	Sets acceleration to 1 g
V5	Sets velocity to 5 ips
D2ØØØ	Sets distance to 2,000 steps
G	Executes the move (Go)
XT	Ends Sequence 1 definition

Command	Description
> XE2	Erases Sequence 2
> XD2	Defines Sequence 2
A1	Sets acceleration to 1 g
V5	Sets velocity to 5 ips
D4ØØØ	Sets distance to 4,000 steps
G	Executes the move (Go)
XT	Ends Sequence 2 definition

Command	Description
> XE3	Erases Sequence 3
> XD3	Defines Sequence 3
A1	Sets acceleration to 1 g
V5	Sets velocity to 5 ips
D8ØØØ	Sets distance to 8,000 steps
G	Executes the move (Go)
XT	Ends Sequence 1 definition

Command	Description
> XE28	Erases Sequence 28
> XD28	Defines Sequence 28
A1	Sets acceleration to 1 g
V5	Sets velocity to 5 ips
D-14ØØØ	Sets distance to -14,000 steps
G	Executes the move (Go)
XT	Ends Sequence 28 definition

STEP 3 Save your sequences by issuing the SV command.

STEP 4

Verify that your programs were stored properly by uploading each entered sequence (**XU** command preceded by the device address and followed by the number of the sequence).

Command	Response
> 1XU1	*A1_V5_D2000_G
> 1XU2	*A1_V5_D4000_G
> 1XU3	*A1_V5_D8000_G
> 1XU28	*A1_V5_D-14000_G

If you receive responses that differ from what you programmed, re-enter those sequences. Remember to Save (**SV**) your changes before resetting the LX.

STEP 5

Your PLC program must turn on the appropriate sequence select input for at least the Scan Time (**SN**) to run these sequences. Since sequence hold (**XQ1**) was enabled in the power-up sequence, your program must turn off all the sequence select inputs before another sequence can be selected.

Thumbwheel Operation

The following example shows how the LX is typically used with remote thumbwheels. In the example below, only 3 sequences are entered. As many as 39 sequences may be entered (in **IM2**) and executed using remote thumbwheels. Refer to Figure 4-9 for thumbwheel connections.

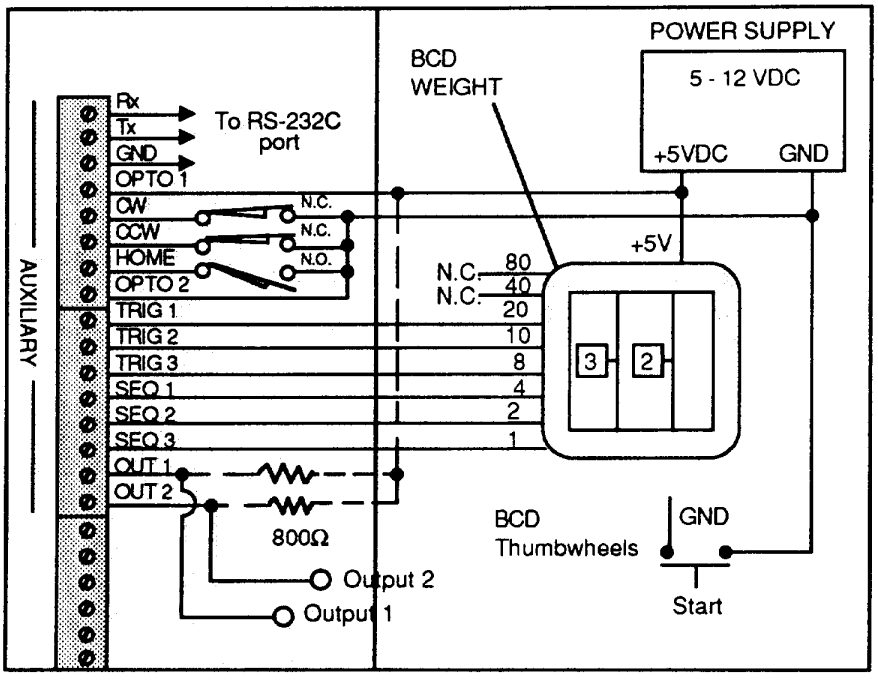


Figure 4-9. Thumbwheel Connections

STEP 1 Define a power-up sequence.

Command	Description
> XE40	Erase sequence 40
> XD40	Define sequence 40
> IM2	Change to Input mode 2
SSJ1	Set LX to execute programs from remote interface
SN1ØØ	Set scan time to 100 msec
XQ1	Enable sequence hold
XT	Ends sequence definition

STEP 2 Define any any sequences that your application may need.

Command	Description
> XE1	Erases Sequence 1
> XD1	Starts Sequence 1 definition
MN	Sets mode to normal
A2	Sets acceleration to 2 g's
V5	Sets velocity to 5 ips
D25ØØØ	Sets distance to 25,000 steps
G	Executes the move (Go)
XT	Ends sequence definition

Command	Description
> XE5	Erases Sequence 5
> XD5	Starts Sequence 5 definition
MN	Sets mode to normal
A2	Sets acceleration to 2 g's
V5	Sets velocity to 5 ips
D1ØØØØ	Sets distance to 10,000 steps
G	Executes the move (Go)
XT	Ends sequence definition

Command	Description
> XE39	Erases Sequence 39
> XD39	Starts Sequence 39 definition
MN	Sets mode to normal
A2	Sets acceleration to 2 g's
V5	Sets velocity to 5 ips
D-35ØØØ	Sets distance to -35,000 steps
G	Executes the move (Go)
XT	Ends sequence definition

STEP 3 Make your changes permanent by saving them to EEPROM.

Command	Description
> SV	Save to EEPROM
> Z	Reset the LX Drive

STEP 4 Set your thumbwheel to 1 and press the start button to move the forcer 2,500 steps in the positive direction.

STEP 5 Set your thumbwheel to 2 and press the start button to move the forcer 10,000 steps in the positive direction.

STEP 6 Set your thumbwheel to 39 and press the start button to move the forcer 35,000 steps in the negative direction.

Note: If you select (with the thumbwheel) an unprogrammed sequence, nothing will happen.

Special Modes of Operation

The LX has a Step and Direction Output on its 25 pin D connector.

Synchronous Motor Control

When you command the LX to move (while the forcer is moving) identical step and direction signals will be output through the 25 pin D connector on the LX. You may connect another Compumotor L Drive and move both forcers at the exact same speed and acceleration.

The step and direction output from an LX Indexer/Drive is designed for applications where two forcers must make identical moves. For example, in an X-Y gantry system the two *X* axes move in unison (refer to Figure 4-10). The step and direction output signals from the LX are used to control an L Drive and forcer in unison with the LX-controlled forcer. To maintain system accuracy, the LX allows individual alignment of the forcers after establishing a home or zero reference position.

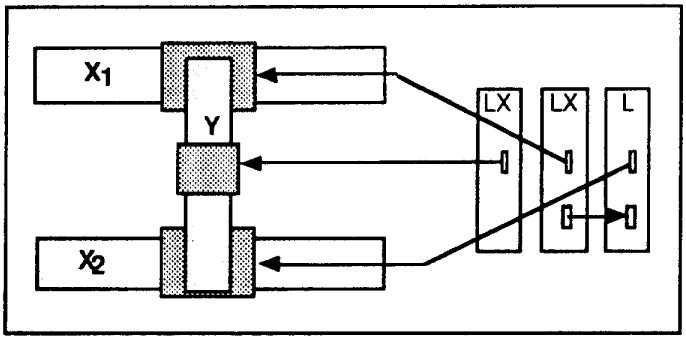


Figure 4-10. X-Y Gantry

You must run the two forcers in this example at the same resolution. If the forcers run at different resolutions, they will perform different moves.

Tuning Linear Motors

The majority of applications will not require you to tune the LX. Tuning is performed at the factory.

If tuning is required, adjust the small potentiometers (pots) on the bottom of the drive. One situation which may require tuning is if a large load (more than 10 lbs) or very sensitive load, such as a camera, is mounted to the forcer.

To detect vibration, an accelerometer can be mounted to the forcer and the output monitored with an oscilloscope, or you can lightly place your hand on the forcer as it moves back and forth. The pot adjustments on the bottom of the LX are listed below. Pot #1 is closest to the DIP switches (see Figure 4-11).

Tuning the drive will reduce the fundamental and second harmonic component of the force ripple of the stepper motor. The fourth harmonic component of the force ripple is not affected by tuning, but can be reduced by selecting the waveform with -4% 3rd added. The fourth harmonic can also be reduced with the special LX firmware, p/n 92-010449-01SP (for more information, contact the Compumotor Applications Department at (800) 358-9070).

Tuning Pots

#1 Current Trim: Adjusts the current setting approximately ±10%.

#2 Phase B Offset: Adjusts the DC offset of the Phase Current for Phase B.

#3 Phase A Offset: Adjusts the DC offset of the Phase Current for Phase A.

#4 Phase Balance: Adjusts the phase current of Phase B to approximately +10% of Phase A.

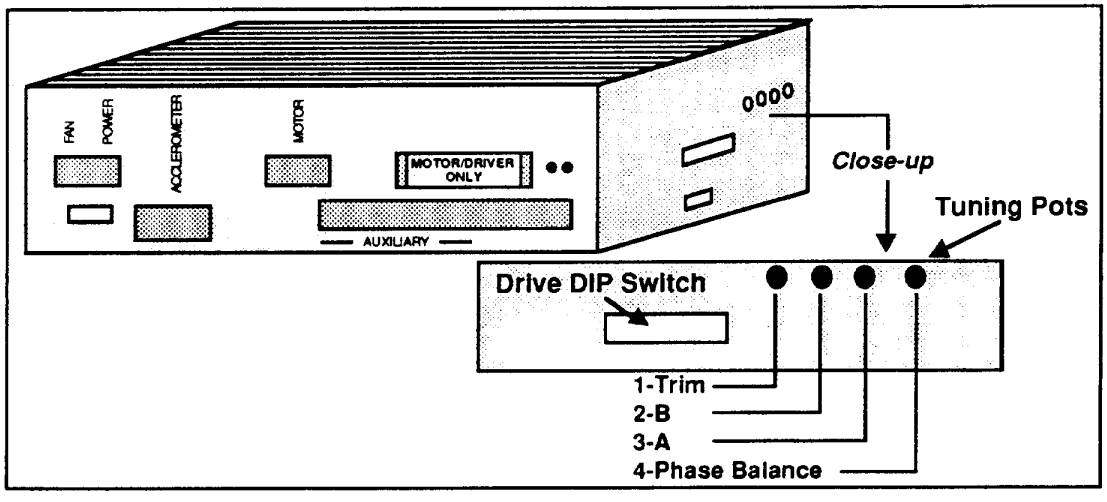


Figure 4-11. Location of Tuning Potentiometers (Pots)

Tuning Procedure

Use the following procedure to tune your system. For best results, the drive and forcer should be turned on, connected to the load, and warmed for 30 minutes prior to tuning. You must also disable the accelerometer if your system is equipped with one.

STEP 1 Connect an indexer and set the controls so that the forcer is running back and forth about 6 inches at *maximum roughness*. Maximum roughness can be observed by lightly touching the forcer to feel for excessive vibration. Table 4-4 gives the speed at this fundamental resonant frequency for the appropriate forcer. Adjust Offset A and Offset B for best smoothness.

STEP 2 Cut the forcer's speed in half (the second harmonic of force ripple is now at the resonant frequency). Adjust the Balance for best smoothness.

STEP 3 Repeat the steps above until no further improvement is noted. The pot adjustments on the bottom of the LX are listed below.

STEP 4 Current Trim will increase and decrease the current that is set to the forcer. Decreasing the current will make the system run more smoothly. However, decreasing the current will also decrease the force. Use this adjustment tool if pots 2 - 4 do not provide measurable performance benefits.

Forcer	Resonant Frequency (Hz)	Fundamental/2nd/4th Resonant
L3C	f = 30*sqrt (3/W)	V = f/15,f/30,f/60
L5A	f = 40*sqrt (6/W	V = f/25,f/50,f/100
L9A	f = 40*sqrt (9/W)	V = f/25,f/50,f/100
L20	f = 40*sqrt (20W)	V = f/25,f/50,f/100

Table 4-4. Fundamental Resonant Frequency

Chapter 5. SOFTWARE REFERENCE

Chapter Objectives

Use this chapter as a reference for the function description of each software command for the LX Indexer/Drive.

Command Descriptions

(1) A (2) Motion	(3) Acceleration			(4) VALID Software Version Y5
SYNTAX (5) <a>An	UNITS (6) n = g's	(7) RANGE 0.001 to 10.000	DEFAULT (8) 0.200	(9) ATTRIBUTES Buffered Savable in Sequence Automatically Saved
EXECUTION TIME <2ms (10)		SEE ALSO D, V, G (11)		
RESPONSE TO <a>A IS *n.nnng (12)				

Figure 5-1. Command Example

1. ***Command Mnemonic*** — This box contains the command's mnemonic value and type. The command types are described below.
2. ***Command Type***

   Set-Up — Set-Up commands define set-up conditions for the application. These commands establish the output data format from the encoder, as well as other functions.

   Motion — Motion commands affect motor motion, such as acceleration, velocity, distance, go home, stop, direction, mode, etc.

   Programming — Programming commands affect programming and program flow for trigger, output, all sequence commands, quote, time delays, pause and continue, enable and disable, loop and end-loop, line feed, carriage return, and backspace.

   Status — Status commands respond (report back) with information. These commands instruct the system to send data out from the serial port for host computer use.

3. ***Command Name*** — This field contains the actual (and complete) name of the command.
4. ***Valid*** — This field contains the current revision level of the command at the time this user guide was released.

5. Syntax

The proper syntax for the command is shown here. The specific parameters associated with the command are also shown. If any of these parameters are shown in brackets, such as <a>, they are optional. Definitions of the parameters are described below.

a — An ***a*** indicates that a device address must accompany the command. Only the device specified by this parameter will receive and execute the command. Valid addresses are 1-8.

n — An ***n*** represents an integer. An integer may be used to specify a variety of values (acceleration, velocity, etc.).

s — An ***s*** indicates that a sign character, either positive or negative (+ or -), is required.

x — An ***x*** represents any character or string of characters.

6. Units

This field describes what unit of measurement the parameter in the command syntax represents.

7. Range

This is the range of valid values that you can specify for n (or any other parameter specified).

8. Default

The default setting for the command is shown in this box. A command will perform its function with the default setting if you do not provide a value.

9. Attributes

This box indicates if the command is ***immediate*** or ***buffered.***

The system executes immediate commands as soon as it receives them. Buffered commands are executed in the order that they are received with other buffered commands. All buffered commands can be stored in a sequence and saved in permanent memory.

The lower portion of the box explains how you can save the command.

- Savable in Sequence
- Never Saved
- Automatically Saved

Savable in Sequence commands are those commands which, when defined in a sequence, are saved in that sequence with the SV command. A command that is **Never Saved** is executed without being saved into the system's permanent memory (EEPROM). **Automatically Saved** commands do not have to be in a sequence, but are saved into EEPROM after issuing the SV command.

10. Execution Time

The execution time is the span of time that passes from the moment you issue a command to the moment the indexer has executed it.

11. See Also

Commands that are related or similar to the command described are listed here.

12. Response

A sample status command is given (next to **RESPONSE TO**) and the system response is shown (next to **IS**). When the command has no response, this field is not shown. Note that after each system response the indexer transmits a carriage return [cr].

Command Listing

A Motion	Acceleration			VALID Software Version Y5
SYNTAX <a>An	**UNITS** n = g	**RANGE** 0.00 - 10.00	**DEFAULT** 0.20	**ATTRIBUTES** Buffered Savable in Sequence Automatically Saved
EXECUTION TIME ≤2.5ms		**SEE ALSO** D, V, G		
RESPONSE TO <a>A **IS** *n.nnng				

Description

The Acceleration command specifies the starting acceleration rate to be used upon executing the next Go (G) command. The acceleration remains set until you change it. You do not need to reissue this command for subsequent Go (G) commands. Accelerations outside the valid range causes the acceleration to remain at the previous valid acceleration setting.

The LX imposes limits on the maximum command acceleration for each of the Compumotor forcers. The LX determines the maximum values by looking at the configuration of the dip switches which select motor type and resolution. *Low acceleration values may limit maximum velocity.*

The following table gives the maximum **A** command that the LX will accept for the forcer and resolution selected.

Acceleration Limits

(1 g = 386 ips^2)

Forcer	Acceleration Limit
L3C	2.00 g
L5A	5.00 g
L9A	5.00 g
L20	10.00 g

Example

Command	Description
> **MN**	Sets the moves to mode normal (preset moves)
> **A2**	Sets Acceleration to 2 g
> **V1Ø**	Sets Velocity to 10 ips
> **D1ØØØØ**	Sets Distance to 10,000 steps
> **G**	Executes the move (Go)

B Status	Buffer Status Report			VALID Software Version Y5
SYNTAX aB	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME <2ms		**SEE ALSO** BS		
RESPONSE TO aB IS *B or *R				

Description

The buffer status command will report the status of the command buffer. If the command buffer is empty or less than 90% full, the controller will respond with a ***R**[cr].

The command buffer is 1,800 bytes long. A ***B**[cr] response will be issued if less than 10% of the command buffer is free.

***R** = More than 10% of the buffer is free
***B** = Less than 10% of the buffer is free

This command is commonly used when a long series of commands will be loaded remotely. If the buffer size is exceeded, the extra commands will not be received by the Controller.

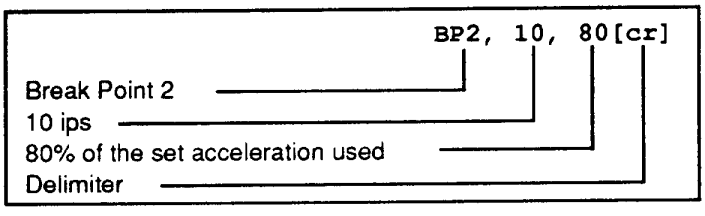
Example

Command	Response
> 1B	***R** (more than 10% of the Buffer is free)

BP Set-up	Break Point			VALID Software Version Y5
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>**BP**n^1, n^2, n^3	n^1 = break pt # n^2 = ips n^3 = % of **A**	n^1 = 1 - 16 n^2 = 0 - 250 n^3 = 5 - 100	N/A	Buffered Savable in Sequence Automatically Saved
EXECUTION TIME <2ms		**SEE ALSO** BPA, BPC, BPL, MF		
RESPONSE TO a BP IS *BPx_MFnn_BPn,nn,nn (see examples below)				

Description

Three different **BP** tables can be defined. This command sets up break points for the currently active Move Form (**MF**). The purpose of this command is to provide a custom acceleration profile. The format of the break point command is shown below:



You may select up to 16 break points, but may define fewer break points. Undefined break points will be ignored. To add, delete, or modify break points you must be in **BPC** mode.

The first Break Point (**BP1**) command you enter must have a velocity setting of 0 ips, and the Last Break Point (**BP16**) command you enter must have a velocity setting of the absolute maximum velocity for that motor. See the **V** command for the maximum velocity settings for each motor.

BP1 and BP16 cannot be deleted - only modified.

When you first power on the indexer, factory selected break points (which is optimum for the motor you have purchased) are loaded into the LX. For most applications, these break points will be sufficient.

To define a custom move profile, you must first erase the existing break points by entering **BP2, BP3, BP4 . . . BP15**. Then type in the new break point values. As you define the break points, you must increase the velocity as you increase the break points (e.g. , BP1, 0, 90, BP2, 10, 90 . . .).

The profile that the **BP** commands generate is initialized during acceleration.

Example

Command	Description/Response
> A3	Sets acceleration to 3 g
> MF2	Selects move form 2
> BP3	Deletes break point 3
> BP4	Deletes break point 4
> BP5	Deletes break point 5
> BP6	Deletes break point 6
> BP3,20,85	Sets break point 3 to 20 ips at 85% of the set acceleration
> BP4,30,73	Sets break point 4 to 30 ips at 73% of the set acceleration
> BP5,40,63	Sets break point 5 to 40 ips at 63% of the set acceleration
> BP6,50,54	Sets break point 6 to 50 ips at 54% of the set acceleration
> BP7,60,46	Sets break point 7 to 60 ips at 46% of the set acceleration
> BP8,70,39	Sets break point 8 to 70 ips at 39% of the set acceleration
> BP9,80,33	Sets break point 9 to 80 ips at 33% of the set acceleration
> BP10,90,28	Sets break point 10 to 90 ips at 28% of the set acceleration
> BP16,100,25	Sets break point 16 to 100 ips at 25% of the set acceleration
> 1BP	*BPC_MF2 Display break points:

```
*BP1,Ø,1ØØ_____BP5,40,63_____BP9,80,33
*BP2,10,98_____BP6,50,54_____BP10,90,28
*BP3,20,85_____BP6,50,54_____BP16,100,25
*BP4,30,73_____BP8,70,39
```

> 1SV	*Ø Save break points and move form and interactive mode in EEPROM
> Z	Resets indexer
> 1BP	*BPC_MF2 Display break points:

```
*BP1,Ø,1ØØ_____BP5,40,63_____BP9,80,33
*BP2,10,98_____BP6,50,54_____BP10,90,28
*BP3,20,85_____BP6,50,54_____BP16,100,25
*BP4,30,73_____BP8,70,39
```

BPA Set-up	Break Point Automatic			VALID Software Version Y5
SYNTAX <a>**BPA**	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Buffered Savable in Sequence Automatically Saved
EXECUTION TIME <2ms		**SEE ALSO** BP, BPC, BPL, MF		

Description

The indexer has pre-defined acceleration profiles stored in ROM (Read Only Memory), which are designed to maximize force utilization and minimize the move time for the selected linear motor. To use the motor's pre-programmed break points for automatic acceleration roll off, the **BPA** command is entered.

Since the indexer already contains the optimal break points, it is not necessary to enter break points (**BP**). The indexer powers up in Automatic Acceleration Roll Off mode (**BPA**). To customize the acceleration profile, use the Break Point Custom (**BPC**) command. For linear acceleration, use the Break Point Linear (**BPL**) command.

Example

Command	Description
> **BPA**	Sets all 16 break points to the default values for the motor being used.

For optimum performance, the linear motor used should match the corresponding profile selected with Indexer DIP switches 1, 2, and 3 (see Chapter 6, Hardware Reference).

BPC Set-up	Break Point Custom			VALID Software Version Y5
SYNTAX <a>**BPC**	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Buffered Savable in Sequence Automatically Saved
EXECUTION TIME <2ms		**SEE ALSO** BP, BPA, BPL, MF		

Description

This command sets the current Move Form (**MF**) to custom acceleration profiling mode. Once you issue the Break Point Custom(**BPC**) command, you can define different break points using the Break Point (**BP**) command, resulting in non-linear accelerations/decelerations.

You must issue a **BPC** command to use the custom profile. **BPA** and **BPL** commands are available for automatic acceleration roll off and linear acceleration profiling respectively. In **BPA** and **BPL** mode, you do not need to define break points (**BP**).

The indexer has pre-defined acceleration profiles stored in ROM (Read Only Memory), which are designed to maximize force utilization and minimize the move time for the motor. To use the motor's pre-programmed break points for automatic acceleration roll off, the **BPA** command is entered. Since the indexer already contains the optimal break points, it is not necessary to enter break points (**BP**). The indexer powers up in Automatic Acceleration Roll Off mode (**BPA**). To customize the acceleration profile, use the Break Point Custom (**BPC**) command. For linear acceleration, use the Break Point Linear (**BPL**) command.

Example

Command	Description
> **BPC**	Select custom acceleration profiling
> **BP1,Ø,1ØØ**	(Enter break point 1 as 0 g's at 100% of acceleration)
> **BP2,1Ø,96**	(Enter break point 2 as 10 g's at 96% of acceleration)
> **BP5,25,86**	(Enter break point 5 as 25 g's at 86% of acceleration)

NOTE: The acceleration ramp is equal to the deceleration ramp.

BPL Set-up	**Break Point Linear**			**VALID** Software Version Y5
SYNTAX a**BPL**	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Buffered Savable in Sequence Automatically Saved
EXECUTION TIME <2ms		**SEE ALSO** BP, BPA, BPC, MF		

Description

This command sets a standard linear acceleration profile to the currently active move form (**MF**). All moves made after the Break Point Linear (**BPL**) command will have a constant acceleration. This command results in a trapezoidal or triangular move profile. For non-linear acceleration, you must use the Break Point Custom (**BPC**) or Break Point Automatic (**BPA**) command.

In this mode, the LX operates the same as any other Compumotor indexer.

Example

Command	Description
> **MF1**	Sets to move form 1
> **BPL**	Sets move form 1 to have linear acceleration
> **A2**	Sets acceleration to 2 g
> **V5**	Sets velocity to 5 ips
> **D1ØØØØ**	Sets distance to 10,000 steps
> **G**	Executes the move (Go)

BS Status	Buffer Size Status			VALID Software Version Y5
SYNTAX aBS	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME <2ms		**SEE ALSO** B		
RESPONSE TO aBS **IS** *nnnn				

Description

This command reports the number of bytes remaining in the command buffer. When entering long string commands, check the buffer status to be sure that there is enough room; otherwise, commands may be lost. Each character (including delimiters) uses one byte. The range for the response is 0000 - 2,000 bytes.

Example

Command	Response
> 1BS	*0100 (Space for 100 characters is remaining in the command buffer.)

C Programming	Continue			VALID Software Version Y5
SYNTAX <a>C	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME <2ms		**SEE ALSO** PS, U		

Description

The Continue (C) command ends a pause state. It enables your indexer to continue executing buffered commands. After you initiate a pause with the Pause (PS) command or the Pause and Wait for Continue (U) command, you can clear it with a Continue (C) command. This command is useful when you want to transmit a string of commands to the command buffer before you actually need to execute them.

Example

Command	Description
> PS	Pauses execution until the indexer receives a C command
> MC	Sets move to Continuous mode
> A2	Sets acceleration to 2 g
> V5	Sets velocity to 5 ips
> G	Executes the move (Go)
> C	Starts executing commands in buffer

CR Programming	Carriage Return			VALID Software Version Y5
SYNTAX aCR	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME <2ms		**SEE ALSO** LF, " (Quote)		

Description

When the indexer reaches this command in the buffer, it responds by issuing a carriage return (ASCII 13) over its interface back to the host computer. If you place the CR command after a Go (G) command, it indicates when a move is complete. If you place the CR command after a Trigger (TR) command, it indicates when the trigger condition is met.

Example

Command	Description
> MPA	Sets mode for absolute position
> A2	Sets acceleration to 2 g
> V5	Sets Velocity to 5 ips
> D5ØØØ	Sets distance to 5,000 steps
> G	Executes the move (Go)
> CR	Sends a carriage return

The motor moves 5,000 steps. When the motor stops, the indexer sends a carriage return over its interface.

D Motion	Distance			VALID Software Version Y5
SYNTAX <a>Dn	**UNITS** n = steps	**RANGE** 0 - ±1,999,999,999	**DEFAULT** # of steps per inch (resolution)	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME <2ms		**SEE ALSO** A, G, V, MN		
RESPONSE TO aD **IS** *±n				

Description

The Distance (D) command defines either the number of steps the motor will move or the absolute position it will seek after a Go (G) command is entered.

In incremental mode (MPI), the value set with the Distance (D) command will be the distance (in steps) the motor will travel on all subsequent Go (G) commands.

In absolute mode (MPA), the distance moved by the motor will be the difference between the current motor position and the position (referenced to the zero position) set with the D command. **A distance must be defined with the D command before a preset move can be executed.**

Example

Command	Description
> MN	Preset mode to normal (preset)
> A2	Set Acceleration to 2 g
> V1Ø	Set velocity to 10 ips
> D5ØØØØ	Set Distance to 50,000 steps
> G	Executes the move (Go)

DR Status	Default Report			VALID Software Version Y5
SYNTAX aDR	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME <2ms		**SEE ALSO** SV, SA		
RESPONSE TO aDR **IS** (see below)				

Description

The Default Report (DR) command is used while performing set-up procedures. This status request command responds with complete data on the indexer's configuration. If you change a function, this change will be displayed when you execute the DR command. The following table shows the different parameters displayed.

Line Number	Display
1	Motor Type Resolution Limit Switch Settings Output Mode Input Mode
2	Absolut Position Report (PR) OS Settings Trigger Status Shutdown Status
3	SS Settings Encoder Resolution Dead Band Settings Backlash Settings Correction Gain Ready Status
4	Limit Status Indexer Status Go Home Status Indexer DIP Settings Mode Normal/Continuous Incremental/Absolute Move Form
5	Acceleration Velocity Distance

Example

Command	Response
> 1DR	(see below)

```
*MOTOR_L20__RES10000_LDØ_OM1_IM1
*PR+0000000000_OS00000000_TS000000_ST0
*SS0000000000_ER2000_DB256_DW0_CG0_R*R
*RA*@_RB*@_RG*@_SW1111ØØ_MN_MPI_MFI_
*A0.20_V1.00_D6000
```

E Programming	Enable Communications Interface			VALID Software Version Y5
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>**E**	N/A	N/A	N/A	Immediate Never Saved
EXECUTION TIME <2ms		**SEE ALSO** F		

Description

The Enable Communications Interface (**E**) command allows the indexer to accept commands over the communications interface. You can re-enable the communications interface with this command if you had previously disabled the interface with the Disable Communications Interface (**F**) command. This command is useful when units are daisy-chained and you want to upload a sequence without other indexers executing the commands issued.

Example

Command	Description
> **F**	Disables all units (axes) on the communications interface
> **1E**	Enables Device 1
> **4E**	Enables Device 4
> **G**	Executes the move (Go — only axes 1 and 4 will move)

F Programming	Disable Communications Interface			VALID Software Version Y5
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>**F**	N/A	N/A	N/A	Immediate Never Saved
EXECUTION TIME <2ms		**SEE ALSO** E		

Description

The Disable Communications Interface (**F**) command is useful when you are programming multiple units on a single interface. Axes that are not intended to process global commands receive device specific **F** commands. This allows you to program other units without specifying a device identifier on every command. If you do not disable other units in a daisy chain, uploading programs may cause other units on the daisy chain to perform uploaded commands.

Example

Command	Description
> **1F**	Disables the communications interface on the unit with device address 1
> **3F**	Disables the communications interface on the unit with device address 3
> **G**	All of the indexers except 1 and 3 will execute a move (Go)

G Motion	Go			VALID Software Version Y5
SYNTAX <a>G	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME (see description)		**SEE ALSO** A, D, I, MN, RA, RB, S, V		

Description

The execution time of this command varies as follows:

- ≈2 ms if acceleration table is defined with I command
- ≈30 ms if acceleration table is not defined

The Go (G) command instructs the motor to make a move using motion parameters that you have previously entered. You do not have to re-enter Acceleration (A), Velocity (V), Distance (D), or the current mode (MN) commands with each G command. In the Incremental Preset mode (MPI), a G command will initiate the steps you specified with the D command.

A Go (G) command in the Absolute Preset mode (MPA) will not cause motion unless you enter a change in D command first.

No motor motion will occur until you enter the G command.

If motion does not occur with the G command, enter the status commands RA and RB and check the limit switches.

Example

Command	Description
> MN	Sets mode to normal (preset)
> A2	Sets acceleration to 2 g
> V1Ø	Sets velocity to 10 ips
> D25ØØØ	Sets distance to 25,000 steps
> G	Executes the move (Go)
> A1	Sets acceleration to 1 g
> G	Executes the move (Go), moves 25,000 steps

Assuming the indexer is in the incremental preset mode (MPI), the motor turns 25,000 steps, then repeats the 25,000-step move using the new Acceleration (A) value of 1 g (total distance moved = 50,000 steps).

GH Motion	Go Home			VALID Software Version Y5
SYNTAX <a>**GH**sn	**UNITS** n = ips	**RANGE** 0.01 - 20.00	**DEFAULT** None	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME (see **G** command)		**SEE ALSO** RG, OS commands, V		
RESPONSE TO a**GH** **IS** *±n				

Description

The Go Home (**GH**) command instructs the controller to search for a home switch in the positive or negative direction at the commanded velocity.

This command causes the controller to seek the home position. If in the motor step mode, the controller looks only at the Home Limit input. It will define Home as the position where the Home Limit signal changed states nearest the edge selected with the **OSH** command (usually via a load activated switch).

The controller will reverse direction if an end-of-travel limit is activated while searching for Home; however, if a second end-of-travel limit is encountered in the new direction, the Go Home procedure will stop and the operation will be aborted. The response to **1RG** is ***B**.

The **RG** command response indicates if the operation was successful. ***A** = successful Go Home and ***B** = not successful Go Home.

After the **GH** command is issued, the motor will run in the direction and velocity specified. The motor will keep running after the home switch is activated until it is deactivated. It will then decelerate and reverse direction. Depending on the **OSH** setting, the motor will either creep to the home switch edge or move past the limit switch. It will then reverse direction and creep towards the home switch edge at the **GHF** (final homing) velocity.

Example

Command	Description
> **GH-2Ø**	The motor moves in the negative direction at 20 ips and looks for the Home Limit input to go active.

GHF Motion	Go Home Velocity			VALID Software Version Y5
SYNTAX <a>**GHF**n	**UNITS** n = ips	**RANGE** 0.01 - 5.00	**DEFAULT** 0.1	**ATTRIBUTES** Buffered Savable in Sequence Automatically Saved
EXECUTION TIME <2ms		**SEE ALSO** GH, OS commands, V		
RESPONSE TO a**GHF** **IS** *n				

Description The Go Home Final Velocity (**GHF**) command sets the velocity for the final approach in the go home sequence. When selecting your final go home velocity, keep in mind that in most applications the higher the speed the lower the repeatability of homing accuracy.

Example

Command	Description
> **GHF.1**	The velocity of the final approach of the next Go Home move will be 0.1 ips
> **GH2**	Execute Go Home at 2 ips in + direction

^H Programming	Backspace			VALID Software Version Y5
SYNTAX ^**H**	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME <2ms		**SEE ALSO** None		

Description This command allows you to delete the last character that you entered (unless it was a delimiter). The ^**H** command will not prevent execution of an immediate command. A new character may be entered at that position to replace the existing character. (^**H** indicates that the Ctrl key is held down when the **H** key is pressed.) This command prompts the indexer to backup one character in the command buffer, regardless of what appears on the terminal. On some terminals, the Ctrl and the left arrow <-- keys produce the same character. *NOTE: Pressing the delete key does not delete the previous character.*

Example None

H Motion	Set Direction			VALID Software Version Y5
SYNTAX <a>**H**<s>	**UNITS** N/A	**RANGE** s = + or -	**DEFAULT** +	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME <2ms		**SEE ALSO** D		

Description

The Set Direction (**H**) command changes or defines the direction of the next move that the system will execute. This command does not effect moves already in progress.

H+ = Sets move to + direction
H− = Sets move to - direction
H = Changes direction from the previous setting

A Distance (**D**) command entered after the Set Direction (**H**) command overrides the direction set by the **H** command.

Example

Command	Description
> **MN**	Sets Normal mode
> **A5**	Sets acceleration to 5 g
> **V5**	Sets velocity to 5 ips
> **D25ØØØ**	Sets distance to 25,000 steps
> **G**	Executes the move (Go) in - direction
> **H**	Reverses direction
> **G**	Executes the move (Go) in - direction

I Programming	**Precalculate Acceleration Table**			**VALID** Software Version Y5
SYNTAX <a>I	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME ≤30ms		**SEE ALSO** MF		

Description

To allow for acceleration roll-off, the indexer must create a 128-value acceleration table in memory. The acceleration table needs to be recalculated every time the acceleration (**A**) value changes. This can take up to 30 ms, which delays the start of a move after issuing the **G** command.

The Velocity (**V**) and **A** commands are usually executed only during set up. The acceleration table will not be calculated until the **G** command is executed. Therefore, to reduce the delay between the **G** command execution and actual movement, the **I** command can be executed during set up to precalculate the table. As long as the **A** value is the same, all successive **G** commands will be executed in less than 2 ms.

For applications in which the **A** value must be changed and throughput is important, you can use the **MF** command to quickly change between 3 different precalculated tables (see example below).

Example

Command	Description
> MF1	Define move form 1
> A2.5	Sets acceleration to 2.5 g
> V5Ø	Sets velocity to 50 ips
> BPA	Select automatic break point mode
> I	Precalculates move profile 1
> MF2	Define move form 2
> AØ.75	Sets acceleration to 0.75 g
> V25	Sets velocity to 25 ips
> BPC	Select custom acceleration profiling
> I	Precalculates move profile 2
> MF3	Define move form 3
> AØ.5	Sets acceleration to 0.5 g
> V1Ø	Sets velocity to 10 ips
> BPL	Sets move form 1 to have linear acceleration
> I	Precalculates move profile 3
> SV	Save
•	
> D1ØØØØ	Sets distance to 10,000 steps
> V4Ø	Sets velocity to 40 ips
> MF1	Loads move profile 1
> G	Execute the move (Go) using stored parameters
> MF2	Loads move profile 2
> G	Execute the move (Go) using stored parameters
> MF3	Loads move profile 3
> G	Execute the move (Go) using stored parameters

IM Set-Up	Input Mode			VALID Software Version Y5
SYNTAX <a>IMn	**UNITS** N/A	**RANGE** n = 1, 2, or 3	**DEFAULT** 1	**ATTRIBUTES** Buffered Savable in Sequence Automatically Saved
EXECUTION TIME 2 ms		**SEE ALSO** FR, SN, XQ, TR, SSJ1, XD		
RESPONSE TO aIM **IS** *1, 2, or 3				

Description

This command configures the auxiliary inputs for different functions. By reconfiguring the inputs, the same input may be used for trigger inputs, sequence select inputs, or a remote stop input.

This command specifies how you may define the three sequences and three trigger inputs. In Mode 1, the inputs are used to select sequences 1 through 7. (Note the BCD weighting of the sequence lines listed below.) In Mode 2, the inputs are all used to select sequences 1 through 39, using BCD weighting for the input lines. Mode 3 dedicates the trigger #1 for use as as a stop input. The size of the sequence you define may be variable, as long as the total length of the sequences does not exceed 1800 bytes.

You could still use TRIG1 to TRIG3 and SEQ1 to SEQ3 as Trigger inputs whenever the **TR** command is executed, the inputs will be looked at as trigger inputs.

Sequence 40 may be defined by the **XD** command but can not be selected via the sequence select inputs. Sequence 40 is always run on power up. It is typically used to store (and run) setup (initialization) parameters on power up. It may also be executed via the **XR40** command over the RS-232C interface.

The **IM** command can be executed within the sequence to change the **IM** mode.

In input modes one through three, the **SSJ1** command allows sequences to be selected by grounding the appropriate sequence select inputs to obtain a desired BCD sequence value. If the **XQ** command is issued, the input lines must be released (turned off) before another sequence may be run.

Changing the BCD values of sequence input lines results in a new sequence being run that corresponds to the new value after the current sequence is completed. The sum of the BCD values of the activated lines determines which sequence is run. For example, grounding SEQ1 and SEQ2 results in sequence 6 being run (see Sequence Select Table in Chapter 4).

The **SN** command is used to determine how long the input must be stable before executing the sequence.

The following table shows the different input modes available.

Input	Function	BCD Value
IM1 Mode		
TRIG 1	Trigger 1	0
TRIG 2	Trigger 2	0
TRIG 3	Trigger 3	0
SEQ 1	Sequence Select	4
SEQ 2	Sequence Select	2
SEQ 3	Sequence Select	1
IM2 Mode		
TRIG 1	Sequence Select	20
TRIG 2	Sequence Select	10
TRIG 3	Sequence Select	8
SEQ 1	Sequence Select	4
SEQ 2	Sequence Select	2
SEQ 3	Sequence Select	1
IM3 Mode		
TRIG 1	Remote Stop	0
TRIG 2	**Not Functional**	0
TRIG 3	**Not Functional**	0
SEQ 1	Sequence Select	4
SEQ 2	Sequence Select	2
SEQ 3	Sequence Select	1

Example

Command	Description
> **XE4Ø**	Erases sequence 40
> **XD4Ø**	Defines sequence 40
> **IM1**	Changes to input mode 1
> **XR1Ø**	Runs sequence 10
> **XT**	Ends sequence definition

Indexer will run sequence 10 upon power-up.

IO Status	Immediate Output			VALID Software Version Y5
SYNTAX <a>IOnn	**UNITS** N/A	**RANGE** n = 0, 1, or x	**DEFAULT** N/A	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME <2ms		**SEE ALSO** O, OM		
RESPONSE TO aIO **IS** *nn				

Description

This command immediately sets the output bits as specified.

1 = Turn output bit ON
Ø = Turn output bit OFF
X = Leave unchanged

This command can only be used with a host with an RS-232C interface on-line with the indexer. Typically, this command is used to turn a peripheral device on or off any time during the machine process.

Note that output bits can be redefined by the OM command. If an output is reconfigured as a fault, moving/not moving, or stall output, you will not be able to use the IO command to control the output.

Example

Command	Description
> MN	Sets to preset mode
> A5	Sets acceleration to 5 g
> V1Ø	Sets velocity to 10 ips
> D25ØØØ	Sets distance to 25,000 steps
> IO1X	Sets Output 1 ON and leaves Output 2 unaffected
> G	Executes the move (Go)
> IOØ1	Sets Output 1 OFF and sets Output 2 ON

If the IOØ1 command is issued while the motor is moving, Output 1 turns off, and Output 2 turns on. This is different from the Output (O) command. The O command waits until the move is finished before changing the state of the output.

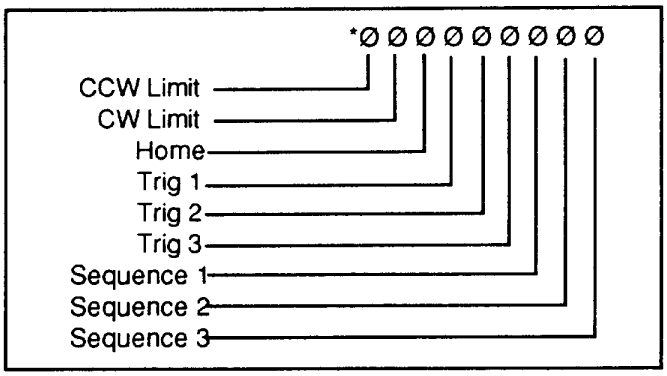
IS Status	Input Status			VALID Software Version E2
SYNTAX <a>IS	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME <2ms		**SEE ALSO** None		
RESPONSE TO aIS **IS** *nnnnnnnnn				

Description

This command reports the status of all hardware inputs, including trigger, sequence select lines, and limits.

This is not a software status. It will report the actual hardware status of the inputs. This command is useful for troubleshooting an application and verifying that Limit switches, Trigger and Sequence inputs, and Home switches work.

The response format is illustrated below. 1 = open (high) and Ø = closed (low).



Example

Command	Response
> 2IS	*010000000 (The input status of device #2 is reported: The CCW limit is open [high] and all other inputs are closed [low].)

K Motion	Kill			VALID Software Version Y5
SYNTAX <a>K	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME <2ms		**SEE ALSO** S		

Description

The Kill (K) command is an emergency stop command and should only be used as such. This command causes indexing to cease immediately. There is *NO* deceleration of the motor.

The Kill command may cause the motor to slip. Consequently, the load could be driven past limit switches and cause damage to the machinery and/or operator.

In addition to stopping the motor, the K command will terminate a loop, end a time delay, abort down-loading a sequence (XD), and clear the command buffer.

Example

Command	Description
> A1	Sets acceleration to 1 g
> V2	Sets velocity to 2 ips
> D1ØØØØØ	Sets distance to 100,000 steps
> G	Executes the move (Go)
.	
.	
> K	Stops the motor instantly

L Programming	Loop			VALID Software Version Y5
SYNTAX <a>L<n>	**UNITS** n = # of Loops	**RANGE** n = 0 - 65,535	**DEFAULT** None	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME <2ms		**SEE ALSO** Y, N, U, C		

Description

When you combine the Loop (L) command with the End-of-Loop (N) command, all of the commands between L and N will be repeated the number of times indicated by *n*. If you enter the L command without a value specified for *n*, or with a Ø, subsequent commands will be repeated continuously.

The End-of-Loop command prompts the indexer to proceed with further commands after the designated number of loops have been executed. The Stop Loop (Y) command causes the inner loop execution to stop. The Immediate Pause (U) command allows you to temporarily halt loop execution. You can use the Continue (C) command to resume loop execution.

Nested loop are allowed up to 24 levels deep.

Example

Command	Description
> A2	Sets acceleration to 2 g
> V1Ø	Sets velocity to 10 ips
> D1ØØØØ	Sets distance to 10,000 steps
> L5	Loops 5 times
> G	Executes the move (Go)
> N	Specifies the above 10,000-step move to be repeated five times

LA Set-Up	**Limit Acceleration**			**VALID** Software Version Y5
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>LAn	n = g	0.00 - (see below)	0.00	Buffered Savable in Sequence Automatically Saved
EXECUTION TIME		**SEE ALSO** A, LD		
RESPONSE TO aLA **IS** *n				

Description

The Limit Acceleration (LA) command allows you to define the deceleration rate that should be used when an end-of-travel limit is encountered. This command is useful if you do not want an abrupt stop upon encountering a limit. However, you should be careful to specify a deceleration rate that will stop the load before it can do any damage. Normally, limit switches should be placed so that the motor has room to safely decelerate the load to a stop.

Setting the LA command to LAØ selects no deceleration (immediate stop).

Acceleration Limits

(1 g = 386 ips^2)

Forcer	Acceleration Limit
L3C	2.00 g
L5A	5.00 g
L9A	5.00 g
L20	10.00 g

Example

Command	Description
> LA1.0	The motor decelerates at 1.0 g when it encounters an end-of-travel limit.

LD Set-Up	Limit Disable			VALID Software Version Y5
SYNTAX <a>LDn	**UNITS** N/A	**RANGE** n = 0 - 3	**DEFAULT** 0	**ATTRIBUTES** Buffered Savable in Sequence Automatically Saved
EXECUTION TIME <2ms		**SEE ALSO** RA		
RESPONSE TO aLD **IS** *n				

Description

The Limit Disable (LD) command allows you to enable/disable the end-of-travel limit switch protection. The LDØ condition does not allow the motor to turn without properly installing the limit inputs. If you want motion without wiring the limits, you must issue the LD3 command.

n = 0: Enable - and + limits
n = 1: Disable + limit
n = 2: Disable - limit
n = 3: Disable - and + limit

Example

Command	Description
> LDØ	Enables + and - limits. The motor will move only if the limit inputs are bypassed or connected to normally closed limit switches.
> LD3	Allows you to make any move, regardless of the limit input state.

LF Execute	Line Feed			VALID Software Version Y5
SYNTAX a**LF**	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME <2ms		**SEE ALSO** " (Quote), CR		
RESPONSE TO a**LF** **IS** [LF]				

Description

Issuing the **LF** command transmits a line feed over the RS-232C link. This command is useful when you send messages during buffered moves or sequences, for display screen control, and to signal move completion when interfacing with a host computer.

You can use the **LF**, **"**, and **CR** commands to display multiple lines of text via the RS-232C interface.

Example

Command	Description
> **1MPA**	Sets mode position to absolute
> **1PS**	Pause move until C command is executed
> **1A1**	Sets acceleration to 1 g
> **1V1**	Sets velocity to 1 ips
> **1D15ØØØ**	Sets distance to 15,000 steps
> 1G	Executes the move (Go)
> **1"DONE**	Transmit message
> **1CR**	Transmits a carriage return
> **1LF**	Transmits a line feed
> 1C	Cancel pause and execute the move

The motor will move to an absolute position of 15,000 steps. Upon completion of this move, the LX will transmit "**DONE**" message and a carriage return with a line feed.

MF Programming	Move Form			VALID Software Version Y5
SYNTAX <a>**MF**n	**UNITS** N/A	**RANGE** n = 1, 2, or 3	**DEFAULT** None	**ATTRIBUTES** Buffered Savable in Sequence Automatically Saved
EXECUTION TIME <2ms		**SEE ALSO** XR, BP, BPC, I, A		
RESPONSE TO a**MF** **IS** *n				

Description

This command lets you define different acceleration profiles. It eliminates the long time delay associated with calculating a new profile. By defining a new move profile and precalculating (**I**), you may save (**SV**) this profile into EEPROM. Recalling this profile is much faster than changing Acceleration (**A**) values between Go (**G**) commands.

Every move form saves its own values for **A**, **V**, **BPA** (or **BPL** or **BPC**) with custom break points. **MF1** contains default values for **A**, **V** and **BPA**. **MF2** and **MF3** are set to **BPA**, but **A** and **V** are set to zero.

For example, (**MF1 A2 I MF2 A.5 I MF1 G MF2 G**) does the same function as (**A2 G A.5 G**), but calling the acceleration profile using the **MF** command is much faster than with the Acceleration (**A**) command. Typically, recalculating a new move profile takes approximately 30 ms. This long delay in calculation time makes the Move Form (**MF**) command a very useful command.

You can define up to three different move forms: **MF1**, **MF2**, or **MF3**. The move form must be present in RAM, which means that if a move form is stored in EEPROM as part of a sequence, the sequence must be executed before the move form can be used.

Example

Command	Description
> D5ØØØØ	Sets distance to 50,000 steps
> MF1	Define move form 1
> A2	Sets acceleration to 2 g
> V40	Sets velocity to 40 ips
> I	Precalculates move profile 1
> MF2	Define move form 2
> A.5	Sets acceleration to .5 g
> V20	Sets velocity to 20 ips
> I	Precalculates move profile 2
> MF1	Loads move profile 1
> G	Executes the move (Go)
> MF2	Loads move profile 2
> H	Change direction
> G	Execute the move (Go)

There will be no calculation delay when executing these moves even though they may use different accelerations, velocities and break points. This is useful where the load on the motor changes, such as when the motor carries a heavy payload in one direction and returns empty. The acceleration profiles of the two moves would be quite different if you were trying to increase throughput.

To define and save a new move form:

STEP 1 Type **MF** 1, 2, or 3 to start definition of a move form

STEP 2 Type **A**, **V**, **BPA**, **BPL**, **BPC** , and **BP** commands to be used in the acceleration profile.

STEP 3 Type I command to preload the acceleration profile

STEP 4 Type Save (SV) command to save the profile in the EEPROM

MN Motion	Mode Normal			VALID Software Version Y5
SYNTAX <a>**MN**	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Buffered Savable in Sequence Automatically Saved
EXECUTION TIME <2ms		**SEE ALSO** MC, D, A, V		
RESPONSE TO a**MN** **IS** *N or *C (N = normal, C = continuous)				

Description

The Mode Normal (**MN**) command sets the positioning mode to preset. In Mode Normal, the motor will move the distance specified with the Distance (**D**) command. To define the complete move profile, you must define Acceleration (**A**), Velocity (**V**), and the Distance (**D**).

To use the **MPA** or **MPI** commands, the indexer must be in mode normal (**MN**).

Example

Command	Description
> **MN**	Set positioning mode to preset
> **A2**	Set acceleration to 2 g
> **V5**	Set velocity to 5 ips
> **D15ØØØ**	Set distance to 15,000 steps
> **G**	Executes the move (Go)

MPA Set-Up	Mode Position Absolute			VALID Software Version Y5
SYNTAX <a>**MPA**	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Buffered Savable in Sequence Automatically Saved
EXECUTION TIME <2ms		**SEE ALSO** MN, MPI, D, PZ		
RESPONSE TO a**MP** **IS** *A or *I				

Description

This command sets the positioning mode to absolute. In this mode all move distances are referenced to absolute zero.
*NOTE: In absolute preset mode (**MPA**), giving two consecutive go (**G**) commands will cause the motor to move once, since the motor will have achieved its desired absolute position at the end of the first move.*

Mode Position Absolute (**MPA**) is most useful in applications that require moves to specific locations, while keeping track of the beginning position.

You can set the absolute counter to zero by cycling power or issuing a Position Zero (PZ) command.

The indexer must be in Mode Normal (MN) to use this command.

Example

Command	Description
> MN	Set mode normal (preset)
> MPA	Set position mode absolute
> A2	Set acceleration to 2 g
> V1Ø	Set velocity to 10 ips
> D25ØØØ	Set distance to +25,000 steps
> G	Motor will move to absolute position +25,000
> D125ØØ	Set absolute position to +12,500 steps
> G	Motor will move - to absolute position +12,500

MPI Set-Up	**Mode Position Incremental**			**VALID** Software Version Y5
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>MPI	N/A	N/A	N/A	Buffered Savable in Sequence Automatically Saved
EXECUTION TIME <2ms		**SEE ALSO** MN, MPA, D		
RESPONSE TO aMPI **IS** *A or *I				

Description

This command sets the positioning mode to incremental. In incremental mode all move distances specified with the Distance (D) command will be referenced to the current position. Mode Position Incremental (MPI) is most useful in applications that require repetitive movements, such as feed to length applications.

The indexer must be in Mode Normal (MN) to use this command.

Example

Command	Description
> MN	Set positioning mode normal (preset)
> MPI	Set positioning mode incremental
> A2	Sets acceleration to 2 g
> V1Ø	Sets velocity to 10 ips
> D1ØØØØ	Sets distance of move to 10,000 steps
> G	Move 10,000 steps (positive direction)
> G	Move another 10,000 steps (positive direction)

The motor moves 10,000 steps (positive direction) after each G command (total move = 20,000 steps).

MT Status	Move Time			VALID Software Version Y5
SYNTAX <a>**MT**	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME varies		**SEE ALSO** MF, DR		
RESPONSE TO a**MT** **IS** (see example below)				

Description

This command reports specific information on the currently defined move. The indexer will calculate the following parameters and report them:

- **AT** Acceleration Time (ms)
- **AD** Acceleration Distance (steps)
- **MV** The total calculated time to complete the move (ms)
- **PV** The peak velocity to be reached during the move (ips)

This command is useful for figuring out the proper acceleration and velocity when there is a time constraint for completing your move.

If you define a long move, the indexer will require a long time to calculate each value. Sometimes it could take a few minutes to do the calculations.

Example

Command	Response
> 1MT	*AT13ms,AD61steps,MT1.1014ms, PV1.00ips

For the defined move, the acceleration time is set to 13 ms, acceleration distance is set to 61 steps, move time is 1.1014 ms, and peak velocity is 1.00 ips.

N Programming	End of Loop			VALID Software Version Y5
SYNTAX <a>**N**	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME <2ms		**SEE ALSO** L, PS, C, Y		

Description

This command marks the end of loop. You can use this command in conjunction with the Loop (**L**) command. All buffered commands that you enter between the **L** and **N** commands are executed as many times as the number that you enter following the **L** command. You may nest loops 24 levels deep.

Example

Command	Description
> PS	Pauses the execution of buffered commands until the indexer receives a Continue (C) command
> MN	Sets move to mode normal
> A2	Sets acceleration to 2 g
> V5	Sets velocity to 5 ips
> D25ØØØ	Sets move distance to 25,000 steps
> L5	Loops five times
> G	Executes the move (Go)
> N	Ends the loop
> C	Clears pause and executes all the buffered commands

O Programming	Output			VALID Software Version Y5
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>Onn	N/A	n = 0, 1 or X	None	Buffered Savable in Sequence
EXECUTION TIME <2ms		**SEE ALSO** OM		

Description

The Output (O) command turns programmable output bits #1 and #2 on and off. This is used for signaling remote controllers, turning on LEDs, or sounding whistles. The output can indicate that the motor is in position, about to begin its move, at constant velocity, etc. The command is in the form O*nn*, where *nn* represents output #1 and output #2 respectively. Outputs may be redefined with the **OM** command. The valid values for *n* are as follows:

1 = Turns on output
0 = Turns off output
X = Leaves output unchanged

Example

Command	Description
> A2	Set acceleration to 2 g
> V5	Sets velocity to 5 ips
> D25ØØØ	Set move distance to 25,000 steps
> O1X	Set programmable output 1 to on
> G	Executes the move (Go)
> OXØ	After the move ends, turn off output 2

OFF Programming	Off			VALID Software Version Y5
SYNTAX <a>OFF	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME <2ms		**SEE ALSO** ON, ST		

Description

This command allows you to shut the amplifier off and remove current from the motor. When you issue an **OFF** command, the fault LED light will be on to indicate that the drive is shut down. You must issue an **ON** command to re-energize the motor and clear the fault. The contents of the command buffer will be cleared when you execute this command.

The **OFF** command removes motor force, allowing you to move the motor manually. This command is similar to the **ST1** command. However, this command is immediate, and **ST1** is buffered.

Example

Command	Description
> OFF	Powers down the motor (no force)

OM Setup	Output Mode			VALID Software Version Y5
SYNTAX <a>OMn	**UNITS** N/A	**RANGE** n = 1 - 5	**DEFAULT** 1	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME <2ms		**SEE ALSO** IM, IO, O		
RESPONSE TO aOM **IS** *n				

Description

This command allows you to set different output modes or to reconfigure the functionality of each output. The mode that you decide to use depends on your application.

	OM1	OM2	OM3	OM4	OM5
Output 1	Programmable	Progr.	Motor Moving	Motor Moving	Motor Moving
Output 2	Programmable	*Reserved*	Progr.	*Reserved*	Constant Velocity

Constant Velocity means that in Output Mode 5, output 2 goes on when the motor reaches constant velocity.

Programmable means the output may be turned on and off with the Output (O) and Immediate Output (IO) commands.

Motor Moving means the output is turned on when the motor is in motion and off when it is not in motion.

Example

Command	Description
> OM1	Both outputs may be programmed to turn on and off using the O and IO commands.
> O1X	Turns on Output 1 and leaves Output 2 at its previous setting.

ON Programming	On			VALID Software Version Y5
SYNTAX <a>ON	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME <2ms		**SEE ALSO** OFF, ST		

Description

This command turns the amplifier back on from the off state. If you issue the OFF command to shut the drive down, issuing the ON command re-enables the current to the motor, restoring motor torque force.

This command is similar to the STØ command. The ON command, however, is immediate and STØ is buffered.

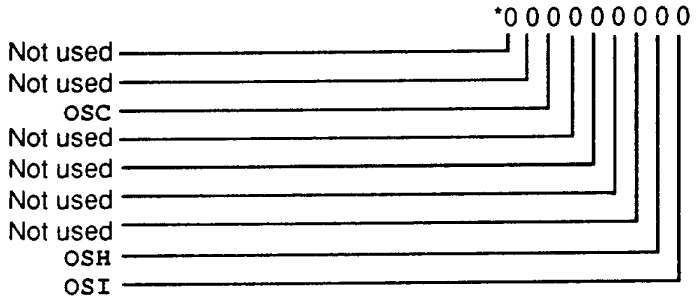
Example

Command	Description
> ON	Turns the motor on

OR Status	Report Homing Function Set-Ups			VALID Software Version Y5
SYNTAX aOR	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME <2ms		**SEE ALSO** OS commands		
RESPONSE TO aOR **IS** *nnnnnnnnn				

Description

This command results in a report of which software switches have been set by the OS commands. The reply is eight digits. This command reports OSA through OSI set-up status in binary format. The digit *1* represent ON (enabled), the digit *0* represents OFF (disabled). The default response is 000000000.



Example

None

OSC Set-Up	Define Active State of Home Switch			VALID Software Version Y5
SYNTAX <a>OSCn	**UNITS** N/A	**RANGE** n = 0, 1	**DEFAULT** 0	**ATTRIBUTES** Buffered Savable in Sequence Automatically Saved
EXECUTION TIME <2ms		**SEE ALSO** GH, OSH		
RESPONSE TO aOSC **IS** *n				

Description

OSCØ = Active state of home input is closed.
OSC1 = Active state of home input is open.

This command inverts the active state of the home input. It enables you to use either a normally closed or a normally open switch for homing.

OSCØ requires that a normally open (high) switch be connected to the home limit input. OSC1 requires that a normally closed (low) switch be connected to the home limit input.

Example

Command	Description
> OSC1	Sets the active state of the home input to open

OSH Set-Up	Reference Edge of Home Switch			VALID Software Version Y5
SYNTAX <a>OSHn	**UNITS** N/A	**RANGE** n = 0, 1	**DEFAULT** 1	**ATTRIBUTES** Buffered Savable in Sequence Automatically Saved
EXECUTION TIME <2ms		**SEE ALSO** GH		
RESPONSE TO aOSH **IS** *n				

Description

OSHØ Selects the positive of the Home signal as the *edge* on which the final approach will stop

OSH1 Selects the negative edge of the home signal as the *edge* on which the final approach will stop

The positive edge of the Home switch is defined as the first switch transition seen by the indexer when traveling off of the positive direction limit in the negative direction. If n = 1, the negative edge of the Home switch will be referenced as the Home position. The negative edge of the Home switch is defined as the first switch transition seen by the indexer when traveling off of the negative direction limit in the positive direction.

Example

See OSB example.

OSI Set-Up	Linear Gantry System Homing			VALID Software Version Y5
SYNTAX <a>OSIn	**UNITS** N/A	**RANGE** n = 0, 1	**DEFAULT** 1	**ATTRIBUTES** Buffered Savable in Sequence Automatically Saved
EXECUTION TIME <2ms		**SEE ALSO** GH		
RESPONSE TO aOSI **IS** *n				

Description

OSI1 = Enables linear gantry system homing function
OSIØ = Disables linear gantry system homing function

This command is useful for setting up the X-Y gantry system. The two X motors carrying the Y beam must be aligned; use the Gantry System Homing (**OSI**) and Go Home (**GH**) commands to accomplish this.

When the **OSI1** command is enabled and a **GH** command is executed, the controller will search for the extreme limit switch in the opposite direction given by the **GH** command.

When the limit switch is tripped, the X-axis forcers of the X-Y gantry system are then run at very a low velocity into physical end-stops at both ends of the each platen. These end-stops are perfectly aligned with each other in order to align the Y-axis beam. The forcers will bounce off the end-stops but never more than one tooth pitch; therefore, both X-axis forcers are eventually aligned with each other. After alignment, the controller continues with the normal **GH** sequence to find the home limit switch.

To allow the forcer enough time to decelerate to the low alignment velocity, you should mount the end-of-travel limit switches a short distance away from the physical end-stops.

Example

Command	Description
> **OSI1**	Sets motor to align the beam on gantry system
> **GH2**	Executes homing, and lines up the motors

PR Status	Absolute Position Report in Motor Steps			VALID Software Version Y5
SYNTAX a**PR**	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME <2ms		**SEE ALSO** MPI, MPA, MN, PZ, D		
RESPONSE TO a**PR** **IS** *±nnnnnnnnnn				

Description

This command reports motor position with respect to power-up position or last place a **PZ** command was issued. The absolute position counter can track (report on) up to 2,147,483,647 motor steps. If the counter is over-run in the relative position mode (by running the motor continuously for long periods of time, 24 hours at 20 RPS), the absolute position will be invalid.

The response to the **PR** command represents the motor (forcer) position after the move is complete.

You may reset the position counter to zero with the Position Zero (**PZ**) command.

This command can only respond when the motor is not being commanded to move. Should you need positional information when the motor is moving, use the **W** commands.

Example

Command	Description
> **PZ**	Resets the absolute counter to zero
> **LD3**	Disable both + & - limits
> **MN**	Set indexer to normal mode
> **A2**	Set Acceleration to 2 g
> **V5**	Set velocity to 5 ips
> **D25ØØØ**	Set move distance to 25,000 steps
> **G**	Executes the move (Go)
> **1PR**	Request absolute position report. (Response should be ***+0000025000**)

PS Programming	Pause			VALID Software Version Y5
SYNTAX <a>PS	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME <2ms		**SEE ALSO** C, U, SSB		

Description

This command pauses execution of a command string or sequence following the Pause (PS) command until the indexer receives a Continue (C) command. This command is useful if you need to enter a complete string of commands before you can execute your other commands.

This command is useful for interactive tests and in synchronizing multiple indexes that have long command strings.

Example

Command	Description
> PS	Pauses execution of following commands until the indexer receives the Continue (C) command
> A2	Sets acceleration to 2 g
> V5	Sets velocity to 5 ips
> D25ØØØ	Sets move distance to 25,000 steps
> G	Executes the move (Go)
> T2	Delays the move for 2 sec
> G	Executes the move (Go)
> C	Continues Execution

When the indexer receives the C command, the motor moves 25,000 steps twice with a 2-second delay between them.

PZ Set-Up	Set Absolute Counter to Zero			VALID Software Version Y5
SYNTAX <a>PZ	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME <2ms		**SEE ALSO** MN, MPI, MPA, PR, D		

Description

This command sets the absolute position counter to zero. When you power-up the LX, the position counter is automatically set to zero. After moving the motor, the **PZ** command is used to reset the absolute position counter.

If the motor has not moved since the last **PZ** command, the response to the **PR** command will be ***+0000000000**.

In the absolute mode (**MPA**), all the moves are made with respect to the absolute counter.

Example

Command	Description
> **PZ**	Sets the absolute counter to zero
> **MPA**	Make all preset moves with respect to absolute zero position
> **A2**	Set Acceleration to 2 g
> **V5**	Set Velocity to 5 ips
> **D25ØØØ**	Set move distance to 25,000 steps
> **G**	Executes the move (Go)
> **1PR**	Report Absolute Position (Response = ***+0000025000**)
> **PZ**	Resets the absolute counter to zero
> **1PR**	Report absolute position (Response = ***+0000000000**)

" Programming	Quote			VALID Software Version Y5
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
a"x	x = ASCII characters	up to any 11 characters	None	Buffered Savable in Sequence
EXECUTION TIME		**SEE ALSO** CR, LF		
RESPONSE TO a"x **IS** x				

Description

Any characters entered after the quotation marks (") (up to 11 characters) will be transmitted, exactly as they were entered over the RS-232C link. A space entered by the space bar indicates the end of the command. You must use a character such as an underscore (_) to represent a space in the message. A space is always sent after the last character in the string. This command is used during buffered moves or sequences, or to command other Compumotor devices to move.

Example #1

Command	Description
> **PS**	Sets pause of execution of current command string
> **MPA**	Sets mode to position absolute
> **A2**	Sets acceleration to 2 g
> **V5**	Sets velocity to 5 ips
> **D125Ø**	Sets distance to 1,250 steps
> **G**	Executes the move (Go)
> **1"MOVE_DONE**	Reports back **MOVE_DONE**
> **C**	Continues the move

The motor will travel to an absolute position of 1,250 steps. Upon completion of this move, the LX will send the message **MOVE_DONE** out the RS-232C port.

Example #2

For two LXs in a daisy-chain:

Command	Description
> **A1**	Sets acceleration to 1 g
> **V6**	Sets velocity to 6 ips
> **D25ØØØ**	Sets distance to 25,000 steps
> **G**	Executes the move (Go)
> **1"2XR1**	Reports back **2XR1**

Once the LX completes its move, it instructs a second LX to run sequence 1.

R Status	Request Indexer Status			VALID Software Version Y5
SYNTAX aR	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME <2ms		**SEE ALSO** RA, RB		
RESPONSE TO aR IS *x				

Description

The Request Indexer Status (R) command can be used to indicate the general status of the indexer. Possible responses are as follows:

Response	Definition
*R	Ready
*S	Ready, Attention Needed
*B	Busy
*C	Busy, Attention Needed

The following conditions will cause a response indicating that the indexer is busy:

- Performing a preset move
- Accelerating/decelerating during a continuous move
- Time delay in progress (T command)
- Paused (PS command)
- Waiting on a Trigger (TR command)
- Going Home
- In remote sequencing mode (SSJ1 command)
- Running a sequence
- Executing a loop

The following conditions will cause a response indicating than an error exists:

- Motor current DIP switch setting is incorrect
- Limit has been encountered
- Sequence execution was unsuccessful

When the response indicates that attention is required, more details on the error condition are available by using the RA or RB command.

It is not recommended that this command be used in tight polling loops which could result in microprocessor over load. Time delays can alleviate this problem.

This command is not intended to be used to determine if a move is complete. Rather, it should be used after the move is complete to determine if there might be other errors or faults.

Use a buffered status request command or a programmable output to indicate move completion.

Example

Command	Response
> 1R	*R (Indexer ready, no error conditions exist)

RA Status	Limit Switch Status Report			VALID Software Version Y5
SYNTAX a**RA**	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME <2ms		**SEE ALSO** R, RA, RB		
RESPONSE TO a**RA** **IS** *x				

Description

The Limit Switch Status Report (**RA**) command responds with the status of the end of travel limits during the last move as well as the present condition. This is done by responding with one of 16 characters representing the conditions listed below.

	Last Move Terminated by		Current Limit Status	
Response Character	+ Limit	- Limit	+ Limit	- Limit
*@	NO	NO	OFF	OFF
*A	YES	NO	OFF	OFF
*B	NO	YES	OFF	OFF
*C	YES	YES	OFF	OFF
*D	NO	NO	ON	OFF
*E	YES	NO	ON	OFF
*F	NO	YES	ON	OFF
*G	YES	YES	ON	OFF
*H	NO	NO	OFF	ON
*I	YES	NO	OFF	ON
*J	NO	YES	OFF	ON
*K	YES	YES	OFF	ON
*L	NO	NO	ON	ON
*M	YES	NO	ON	ON
*N	NO	YES	ON	ON
*O	YES	YES	ON	ON

The **RA** command is useful when the motor will not move in either or both directions. The report back will indicate whether or not the last move was terminated by one or both end-of-travel limits.

NOTE: This command is not intended to be used to determine if a move is complete. Rather, it should be used after the move is complete to determine if there might be other errors or faults.

Example

Command	Response
1RA	*@ (By issuing a **1RA** command to the indexer with address of 1, the indexer responded with *@ indicating that the last move was not terminated by a limit and that no limits are currently active.)

RB Status	Loop, Pause, Shutdown, Trigger Status Report			VALID Software Version Y5
SYNTAX aRB	UNITS N/A	RANGE N/A	DEFAULT N/A	ATTRIBUTES Immediate Never Saved
EXECUTION TIME <2ms		SEE ALSO L, PS, R, RA, ST, TR		
RESPONSE TO aRB IS *x				

Description

This command receives a response from *@ to *O, as defined below. The four conditions for which status is indicated are as follows:

Loop Active: A loop is in progress.

Pause Active: Buffered commands are not being executed due to a Pause (PS) command (waiting for a C command).

Shutdown Active: The motor is shutdown by the ST1 command.

Trigger Active: At least one trigger is active.

Response Character	Loop Active	Pause Active	Shutdown Active	Trigger Active
*@	NO	NO	NO	NO
*A	YES	NO	NO	NO
*B	NO	YES	NO	NO
*C	YES	YES	NO	NO
*D	NO	NO	YES	NO
*E	YES	NO	YES	NO
*F	NO	YES	YES	NO
*G	YES	YES	YES	NO
*H	NO	NO	NO	YES
*I	YES	NO	NO	YES
*J	NO	YES	NO	YES
*K	YES	YES	NO	YES
*L	NO	NO	YES	YES
*M	YES	NO	YES	YES
*N	NO	YES	YES	YES
*O	YES	YES	YES	YES

NOTE: This command is not intended to be used to determine if a move is complete. Rather, it should be used after the move is complete to determine if there might be other errors or faults.

Example

Command	Response
> 1RB	*A (After issuing 1RB, the response came back as *A. This means that the indexer is currently within a loop.)

RG Status	Go Home Status			VALID Software Version Y5
SYNTAX aRG	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME <2ms		**SEE ALSO** GH, R, RA, RB		
RESPONSE TO aRG **IS** *x				

Description

The Go Home Status (RG) command responds with either *@, *A, or *B, indicating the status of the last go home attempt.

Response Character	Go Home Status
*@	Not commanded
*A	Successful
*B	Not Successful

This command is useful when you are performing a GH command and need to know when the homing is completed.

Example

Command	Description
> 1RG	*A (Indicates that the last go home attempt was successful)

RS Status	Status of Sequence Execution			VALID Software Version Y5
SYNTAX aRS	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME <2ms		**SEE ALSO** R, RA, RB, XR, XP		
RESPONSE TO aRS **IS** *x				

Description

The **RS** command indicates the status of the latest sequence execution. Possible responses are as follows:

Response	Sequence Started	Sequence Ended	Bad Loop
*@	NO	NO	NO
*A	YES	NO	NO
*B	NO	YES	NO
*D	N/A	N/A	YES

***D** is reported when there is an unbalanced number of loops and end loops inside a sequence. Starting a loop in one sequence and terminating it in another sequence is not allowed. Nested loops require complete closure before execution will begin.

Should the response from the **RS** command be ***C**, Sequence Started and Sequence Ended may indicate that a sequence was interrupted due to a Stop (**S**), or Kill (**K**) command, or due to an error condition like activation of an end-of-travel limit. This condition should be further explored by the use of the **RA** or the **RB** command.

When the response only indicates that a sequence has ended (***B**), then a sequence has been successfully completed.

Example

Command	Response
> 1RS	*A (Sequence in progress)

RV Status	Revision Level			VALID Software Version Y5
SYNTAX aRV	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME <2ms		**SEE ALSO** None		
RESPONSE TO aRV IS *nn-nnnn-nn<xn>				

Description

The Revision (RV) command responds with the software part number and its revision level. The response is in the form shown below:

***nn-nnnn-nn<xn>**
(nn-nnnn-nn = part number; <xn> = revision level)

The part number identifies which product the software is written for, as well as any special features that the software may include. The revision level identifies when the software was written. You may want to record this information in your own records for future use. This type of information is useful when you consult Parker Compumotor's Applications Department.

Example

Command	Response
> 1RV	*92-ØØ7822-Ø1Y5

The product is identified by **92-007822-01**. The revision level is identified by **Y5**.

S Motion	Stop			VALID Software Version Y5
SYNTAX <a>S	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME <2ms		**SEE ALSO** K, SSH, A		

Description

This command decelerates the motor to a stop using the last defined Acceleration (A) command. This command normally clears any remaining commands in the command buffer, unless prevented from doing so by the Clear/Save The Command Buffer On Stop (SSH1) command. When the SSH1 command is present, the S command stops only the current move. The indexer executes the next command in the buffer.

Example

Command	Description
> **A1**	Sets acceleration to 1 g
> **V1Ø**	Sets velocity to 10 ips
> **D1ØØØØØ**	Sets distance to 100,000 steps
> **G**	Executes the move (Go)
> **S**	Stops motor (motor goes to 0 ips at a deceleration rate of 1 g

NOTE: The S command is not buffered since it is an immediate command. As soon as the indexer receives the S command, it stops motion.

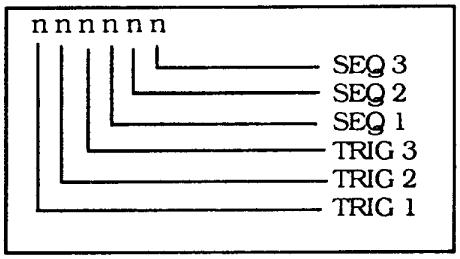
SKE Programming	**Skip If Equal**			**VALID** Software Version Y5
SYNTAX <a>**SKE**n	**UNITS** N/A	**RANGE** n = 0, 1, or X	**DEFAULT** None	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME <2ms		**SEE ALSO** SKN, TR		

Description

If the state (value nnnnnn) of the programmable inputs matches the pattern specified in the Skip If Equal (**SKE**) command, the command following the **SKE** command is skipped (not executed). If the input pattern does not match, the command following **SKE** will be executed.

Input states:
- 1 means the input is on
- Ø means the input is off
- X means either state Ø or 1 is acceptable (*Don't Care*)

The order of inputs is shown in the figure.



Example

Command	Description
> **MN**	Sets mode to normal
> **A1**	Sets acceleration to 1 g
> **V5**	Sets velocity to 5 ips
> **D1ØØØØ**	Sets distance to 10,000 steps
> **SKE1ØXXXX**	If trigger input 1 is on, and input 2 is off, skip the next Go (G) command
> **G**	Executes the move (Go)
> **T1Ø**	Pauses the motor for 10 seconds
> **D-1ØØØØ**	Sets move distance to -10,000 steps
> **G**	Executes the move (Go)

SKN Programming	Skip If Not Equal			VALID Software Version Y5
SYNTAX <a>**SKN**n	**UNITS** N/A	**RANGE** n = 0, 1, or X	**DEFAULT** None	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME <2ms		**SEE ALSO** SKE, TR		

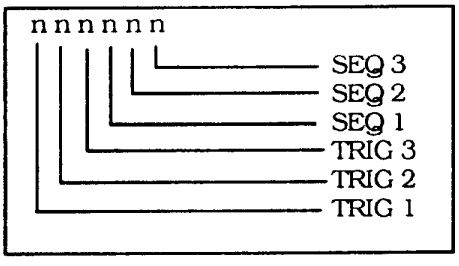
Description

If the state (value nnnnnn) of the programmable inputs matches the pattern specified in the Skip Not Equal(**SKN**) command, the command following the **SKN** command is executed. If the input pattern does not match, the command following the **SKN** command will be skipped (not executed).

Input states:

- **1** means the input is on
- **Ø** means the input is off
- **X** means either state Ø or 1 is acceptable (*Don't care*)

If the state (value nnnnnn) of the programmable inputs does not match the pattern specified in the Skip Not Equal (**SKN**) command, the command following the **SKN** command is skipped. If input pattern matches the trigger state, the **SKN** command is ignored.



Example

Command	Description
> **MN**	Sets mode normal
> **A2**	Sets acceleration to 2 g
> **V5**	Sets velocity to 5 ips
> **D1ØØØØ**	Sets distance to 10,000 steps
> **L**	Loops indefinitely or until it receives a Stop (**S**) command
> **G**	Executes the move (Go)
> **SKNXXXX0X**	Skips the next command (**GH-1**) if the fifth trigger bit is off
> **GH-1**	Instructs the motor to go home in CCW direction at a velocity of 1 ips
> **N**	Ends loop

The motor repeats the same move until it receives a signal on Sequence input two; at which time the indexer goes home and repeats the 10,000-step move.

SN Set-Up	Scan			VALID Software Version Y5
SYNTAX <a>SNn	**UNITS** n = msec	**RANGE** 1 - 200	**DEFAULT** 50	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME <2ms		**SEE ALSO** None		
RESPONSE TO a**SN** **IS** *n				

Description

The Scan (SN) command allows you to define the *debounce time* (in milliseconds) for external sequence selection inputs. The debounce time is the amount of time that the sequence inputs must remain constant for a proper reading from a remote controller, such as a programmable logic controller (PLC). If you are using a PLC you should change the debounce time to match the *on time* of the PLC outputs.

If you precede the command with a device address and do not specify the scan time, the system will report the present scan time setting for the specified unit. The report is a decimal character.

The **SN** command allows you to select the best possible trade-off between noise immunity and speed for a given application. If you make your scan time too short, the LX may respond to an electrical glitch.

Example

Command	Description
> **SN15Ø**	Sets scan time of sequence select inputs to 150 milliseconds

SS Set-Up	Software Switch Function Status			VALID Software Version Y5
SYNTAX aSS	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME <2ms		**SEE ALSO** SR, SS commands		
RESPONSE TO aSS **IS** *nnnnnnnnnn				

Description This command reports the status of the **SS** commands. From left to right, the ten-character response corresponds to **SSA** through **SSJ**.

SSA Set-Up	RS-232C Echo Control			VALID Software Version Y5
SYNTAX <a>SSAn	**UNITS** N/A	**RANGE** n = 0, 1	**DEFAULT** 0	**ATTRIBUTES** Buffered Savable in Sequence Automatically Saved
EXECUTION TIME <2ms		**SEE ALSO** None		
RESPONSE TO aSSA **IS** *n				

Description This command turns the RS-232C echo (transmission of characters received from the remote device by the indexer) on and off.

SSAØ = Echo on
SSA1 = Echo off

In the Echo On (**SSAØ**) mode, characters that are received by the indexer are echoed automatically. In the Echo Off (**SSA1**) mode, characters are not echoed from the indexer. This command is useful if your computer cannot handle echoes. In a daisy chain, you must have the echo turned on (**SSAØ**) to allow indexers further down the chain to receive commands.

Status commands do not echo the command sent, but transmit the requested status report.

Example

Command	Description
> **SSA1**	Turns echo off (Characters sent to the indexer are not echoed back to the host.)

SSB Set-Up	Clear Pause on Trigger 1			VALID Software Version Y5
SYNTAX <a>**SSB**n	**UNITS** N/A	**RANGE** n = 0, 1	**DEFAULT** 0	**ATTRIBUTES** Buffered Savable in Sequence Automatically Saved
EXECUTION TIME <2ms		**SEE ALSO** TR		
RESPONSE TO a**SSB** **IS** *n				

Description

In some cases it is desirable to be able to clear a pause command remotely without sending a Continue (C) command. This mode of operation is possible using the Trigger 1 input and the **SSB** command.

If the **SSB1** command is issued, turning on TRIG1 input will clear a pause command and the indexer will commence execution of commands stored in the buffer. **SSBØ** leaves TRIG1 as a standard input.

Example

Command	Description
> **SSB1**	Uses TRIG 1 to clear a pause
> **PS**	Pauses until continue command is issued
> **MN**	Sets move to continuous
> **A2**	Sets acceleration to 2 g
> **V5**	Sets velocity to 5 ips
> **D25ØØØ**	Set distance to 25,000 steps
> **G**	Executes the move (Go)

The indexer will wait until TRIG1 is closed before executing the move.

SSD Set-Up	Stop On Trigger 1			VALID Software Version Y5
SYNTAX <a>SSDn	**UNITS** N/A	**RANGE** n = 0, 1	**DEFAULT** 0	**ATTRIBUTES** Buffered Savable in Sequence Automatically Saved
EXECUTION TIME <2ms		**SEE ALSO** S, IM		
RESPONSE TO aSSD **IS** *n				

Description

Trigger 1 is dedicated as a stop input with the SSD1 command. When the indexer is in an input mode where the trigger is active, you may elect to use trigger 1 as a dedicated stop input. The result of activating input 1 will be the same as if you had sent an S command via the RS-232C link. A sequence is aborted and any move in progress is stopped using the last specified acceleration. In input modes where Trigger Input 1 is used as a sequence select input, using this command will still cause the input to be used as a STOP command; so care must be taken to avoid using trigger one as a sequence select and stop input. SSDØ leaves TRIG1 as a standard input.

Example

Command	Description
> SSD1	TRIG1 is a dedicated stop input

SSF Set-Up	Sequence Strobe Trigger 1			VALID Software Version Y5
SYNTAX <a>SSFn	**UNITS** N/A	**RANGE** n = 0, 1	**DEFAULT** 0	**ATTRIBUTES** Buffered Savable in Sequence Automatically Saved
EXECUTION TIME <2ms		**SEE ALSO** IM, SS, SSJ, SN		
RESPONSE TO aSSF **IS** *n				

Description

This command dedicates trigger 1 as a sequence select strobe. Sequence select data is not read until trigger one is turned on. The unit must be in IM1 or IM2 to use this function. The SSF1 command defines trigger one as a strobe input when continuous sequence scan mode has been selected (SSJ1).

BCD sequence data should be present before the strobe (trigger 1) is turned on and should remain valid at least as long as the time defined by (SN) command. The SN command defines how long the indexer waits after the input lines are stable before executing the sequence designated by the sequence select lines. When the indexer is in input mode two (IM2) you can only select 19 different sequences because TRIG 1 cannot be used as a BCD value.

Example

Command	Description
> SSF1	TRIG 1 input is used as sequence strobe input

SSG Set-Up	Clear/Save the Command Buffer on Limit			VALID Software Version Y5
SYNTAX <a>SSGn	**UNITS** N/A	**RANGE** n = 0, 1	**DEFAULT** 0	**ATTRIBUTES** Buffered Savable in Sequence Automatically Saved
EXECUTION TIME <2ms		**SEE ALSO** LD		
RESPONSE TO aSSG **IS** *n				

Description

In most cases, it is desirable that upon activating an end of travel limit input, all motion should cease until the problem causing the over-travel is rectified. This will be assured if all commands pending execution in the command buffer are cleared when hitting a limit. This is the case if **SSGØ** is specified. If SSG1 is specified and a limit is activated, the current move is aborted, but the remaining commands in the buffer continue to be executed.

Example

Command	Description
> SSG1	Save buffer on limit
> A2	Set acceleration to 2 g
> V5	Set velocity to 5 ips
> D25ØØØ	Set distance to 25,000 steps
> G	Executes the move (Go)
> Ø11	Turn on outputs 1 and 2

If a limit switch is encountered while executing the move, outputs 1 and 2 will still go on.

SSH Set-Up	Clear/Save the Command Buffer on Stop			VALID Software Version Y5
SYNTAX <a>SSHn	**UNITS** N/A	**RANGE** n = 0, 1	**DEFAULT** 0	**ATTRIBUTES** Buffered Savable in Sequence Automatically Saved
EXECUTION TIME <2ms		**SEE ALSO** LD, SS		
RESPONSE TO aSSH **IS** *n				

Description

SSHØ = Clears command buffer on stop
SSH1 = Saves command buffer on stop

In Normal Operation (**SSHØ**), the Stop (**S**) command or a dedicated stop input will cause any commands in the command buffer to be cleared. If you select the Save Buffer On Stop (**SSH1**) command a Stop (**S**) command will only stop execution of a move in progress. It will not stop execution of any commands that remain in the buffer.

Example

Command	Description
> SSHØ	Clears command buffer on stop
> A2	Sets acceleration to 2 g
> V5	Sets velocity to 5 ips
> D25ØØØ	Sets distance to 25,000 units
> L5Ø	Loop 50 times
> G	Executes the move (Go)
> T.5	Pauses the motor 500 msec
> N	Ends Loop
> S	Stops motion

When you issue the S command, the indexer will clear the buffer and stop the move.

SSI Set-Up	Interactive Mode			VALID Software Version Y5
SYNTAX <a>SSIn	**UNITS** N/A	**RANGE** n = 0, 1	**DEFAULT** 1	**ATTRIBUTES** Buffered Savable in Sequence Automatically Saved
EXECUTION TIME <2ms		**SEE ALSO** SSA		
RESPONSE TO aSSI **IS** *n				

Description

When the Interactive Mode (SSI1) is enabled and the device address is set to one, the indexer responds with a ">" when it understands a command and a "?" when it does not understand a command. If your command was valid but out of range, you will received a "#." Both responses are followed by a carriage return and a line feed. The Interactive Mode is useful during initial setup.

If you try to define a loop (L), a sequence(LD), or a pause(PS), you will not get back a ">" until you finish defining the string of commands.

The SSIØ command disables the interactive mode. In this mode you will not receive ">" ,"#".or "?" from the indexer. The Interactive Mode is typically disabled when used with a host computer.

Select SSIØ to run an LX from a PC with the LX application software.

Example

Command	Description
> SSI1	Sets up the controller in interactive mode
> A2	Sets acceleration to 2 g
> V5	Sets the velocity to 5 ips
> XYZ	XYZ is an invalid command, so the next prompt will be a?
? D10000	Sets distance to 10,000 steps
> G	Executes the move (Go) Prompt waits for your next command

SSJ Set-Up	Continuous Sequence Scan Mode			VALID Software Version Y5
SYNTAX <a>SSJn	**UNITS** N/A	**RANGE** n = 0, 1	**DEFAULT** 0	**ATTRIBUTES** Buffered Savable in Sequence Automatically Saved
EXECUTION TIME <2ms		**SEE ALSO** IM, SV, XD, XR, XS, XT		
RESPONSE TO aSSJ **IS** *n				

Description

With **SSJ1** enabled, the indexer continuously scans the sequence select inputs and executes the sequence represented by the BCD number present on the inputs. To power up in continuous scan mode, you must put **SSJ1** command in sequence 40 and save the sequence to non volatile memory using the save (SV) command. A Stop (S) or Kill (K) command halts continuous sequence scanning. When in Continuous scan mode, you can only execute immediate commands over the RS-232C interface.

With **SSJØ** enabled, the indexer does not scan the inputs for BCD numbers for sequence execution. In this mode, you can only execute sequences using the RS-232C interface.

Example

Command	Description
> SSJ1	The indexer scans the BCD input for sequence execution

ST Programming	Shutdown			VALID Software Version Y5
SYNTAX <a>STn	**UNITS** N/A	**RANGE** n = 0, 1	**DEFAULT** 0	**ATTRIBUTES** Buffered Never Saved
EXECUTION TIME <2ms		**SEE ALSO** None		
RESPONSE TO aST **IS** *n				

Description

The Shutdown (ST1) command rapidly decreases the motor current to zero. The system ignores move commands that you issue after the ST1 command. Force on the motor is not maintained after you issue the ST1 command. The fault LED will illuminate to indicate that the motor is shut down.

The STØ command rapidly increases the motor current to normal. Once you restore the current, you can execute moves.

This command is useful for reducing motor heating and allows you to manually position the load. The position counter is reset to zero when you issue the STØ command.

Example

Command	Description
> ST1	Shuts off current to the motor

SV Programming	Save			VALID Software Version Y5
SYNTAX <a>SV	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME <2ms		**SEE ALSO** None		
RESPONSE TO aSV **IS** *n				

Description

The Save (SV) command saves all sequences and set up parameters currently defined in non-volatile memory. Only the commands in a sequence and the commands that are Independently Saveable will be saved into the EEPROM.

Sequences or parameters not changed remain the same as when previously saved. Sequence or parameter changes not saved will be lost if power is removed from the device.

If device address is specified with **SV**, the LX responds with the following information:

***Ø** = Save is finished and successful
***1** = Save was not successful (BAD EEPROM)

You can save up to 1800 characters when defining sequences.

Example

Command	Description
> 1SV	***Ø** (All valid parameters are saved correctly)

T Programming	**Time Delay**			**VALID** Software Version Y5
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>Tn	n = seconds	0.01 - 65,536.00	None	Buffered Savable in Sequence
EXECUTION TIME <2ms		**SEE ALSO** None		

Description

The Time (**T**) command causes the indexer to wait the number of seconds that you specify (*n*) before it executes the next command in the buffer. This command is useful whenever you need to delay the motor's actions.

Position maintenance is not active during a Time delay.

Example

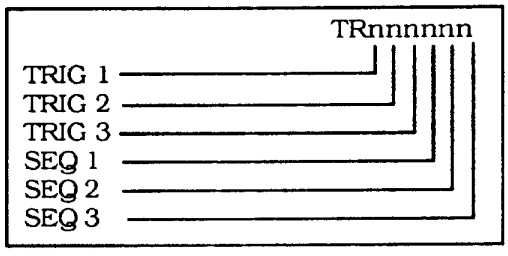
Command	Description
> MN	Set to mode normal
> A2	Set acceleration to 2 g
> V6	Set velocity to 6 ips
> D125ØØ	Set distance to 12,500 steps
> G	Execute the move (Go)
> T3	Delays 3 seconds between moves
> G	Execute the move (Go)

TR Programming	Wait for Trigger			VALID Software Version Y5
SYNTAX <a>TRn	**UNITS** N/A	**RANGE** n = 0, 1, X	**DEFAULT** None	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME <2ms		**SEE ALSO** TS, SSI		
RESPONSE TO aTR **IS** *nnnnnn				

Description

Triggers are used to synchronize indexer operations with external events. They can be used to implement a handshaking function with other devices. Three characters that are used for n variable is listed below:

n = 1 Wait for the input to be grounded
n = 0 Wait for the input to be opened
n = X Ignore the input



All six triggers are available in all input modes. If you are using some of these inputs for other functions, use X when you are using this command. For example, if you are in IM3 mode, TRIG 1 is used as a dedicated STOP input. So you should always have X for trigger 1 input.

When TR command is used in a buffer, the indexer will get to this command and wait until the input pattern is matched before going on to the next command.

Example

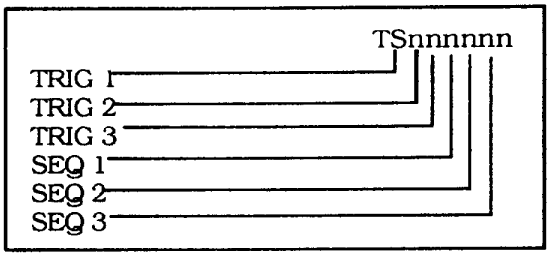
Command	Description
> TR1ØXXXX	Wait for input 1 to be closed and input 2 to be opened before going on to the next command. Inputs 3, 4, 5, & 6 are ignored.
> A2	Set acceleration to 2 g
> V5	Sets velocity to 5 ips
> D25ØØØ	Sets distance to 25,000 steps
> G	Executes the move (Go)

TS Status	Trigger Input Status			VALID Software Version Y5
SYNTAX a**TS**	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME <2ms		**SEE ALSO** TR, IS		
RESPONSE TO a**TS** **IS** *nnnnnn				

Description

The Trigger Status command retrieves the status of the trigger inputs.

n= 1 (Input is grounded)
n = Ø (Input is opened)



Since the **TS** command is immediate, the host controller can determine the status of the trigger inputs at any time, even during execution of other commands. You can use this command to make sure that your trigger pattern is met, when you have issued the trigger (**TR**) command.

This command is also useful to check the functionality (installation) of your hardware inputs.

Example

Command	Response
> 1**TS**	*101000 (Trigger bits 1 and 3 are grounded and Trigger 2, and sequence inputs 1, 2, and 3 are open.)

U Programming	**Pause and Wait for Continue**			**VALID** Software Version Y5
SYNTAX <a>**U**	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME <2ms		**SEE ALSO** PS, C		

Description

This command causes the indexer to complete the move in progress, then wait until it receives a Continue (C) to resume processing. Since the buffer is saved, the indexer continues to execute the program (at the point where it was interrupted). The indexer continues processing when it receives the C command. This command is typically used to stop a machine while it is unattended.

Example

Command	Description
> **MN**	Sets move to Normal mode
> **A2**	Set acceleration to 2 g
> **V5**	Sets velocity to 5 ips
> **L**	Loops indefinitely
> **D25ØØØ**	Sets distance to 25,000 steps
> **G**	Executes the move (G)
> **T1Ø**	Waits 10 seconds after the move
> **N**	Ends loop
> **U**	Halts execution until the indexer receives the Continue command.

This command string pauses at the point where the U command is entered. A Continue (C) command causes execution to resume at the point where it was paused. In this example, the loop stops at the end of a move, and resumes when the indexer receives the C command. There may be a 10-second delay before motion resumes after the C command is executed, depending on when the Pause and Wait for Continue (U) command is completed.

V Motion	Velocity			VALID Software Version Y5
SYNTAX <a>Vn	**UNITS** n = ips	**RANGE** 0.00 - 100.00	**DEFAULT** 1.00	**ATTRIBUTES** Buffered Savable in Sequence Automatically Saved
EXECUTION TIME <2ms		**SEE ALSO** A, D, G		
RESPONSE TO aV **IS** *n				

Description

The Velocity (V) command defines the maximum speed at which the motor will run when given the Go (G) command. The actual speed of the motor or output frequency of the indexer will vary, depending on the motor drive resolution. The following formula is used to determine the output frequency of the indexer:

Frequency = (n) * (Motor Resolution) in steps/in.

The top speed of the motor drive is limited by the motor type. Entering a velocity higher than the top speed of a motor drive system will cause the motor to stall and may cause the drive to fault.

Velocity Limits

The following table shows the top speed that can be programmed by the LX using different linear motors. The DIP Switch 1 is used to select different motors that are used by the LX drive.

Forcer	Velocity (ips) @ given Resolution (steps/inch)			
L3C	60 ips @ 3,000	60 ips @ 5,400	60 ips @ 6,000	60 ips @ 7,500
L5A	60 ips @ 5,000	60 ips @ 9,000	60 ips @ 10,000	60 ips @ 12,500
L9A	50 ips @ 5,000	50 ips @ 9,000	50 ips @ 10,000	50 ips @ 12,500
L5A	100 ips @ 5,000	85 ips @ 9,000	75 ips @ 10,000	60 ips @ 12,500

Example

Command	Description
> A2	Set acceleration to 2 g
> V5	Sets velocity to 5 ips
> D1ØØØØ	Sets distance to 10,000 steps
> G	Executes the move (Go)

The maximum velocity may also be limited when the resulting move profile is triangular.

W1 Status	Signed Binary Position Report			VALID Software Version Y5
SYNTAX a**W1**	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME		**SEE ALSO** W3, PR		
RESPONSE TO a**W1** **IS** nnnn				

Description

Report back gives immediate binary representation of position relative to start of the current move. The format of the response is a four character response (**nnnn**) that is interpreted as a 32-bit binary number. The number must then be interpreted by the computer to give a numerical position in steps. The format is in 2's complement notation. Moves in the negative direction (-) will report back negative numbers (bit 31 is set to 1).

If you are using a terminal to communicate with the indexer, the response may not be a printable character. The response must be decoded using a computer.

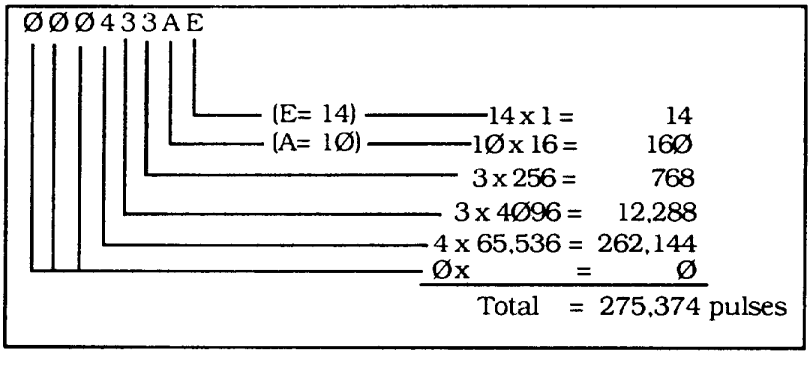
This command is useful if you want to receive a position report while the motor is moving.

W3 Status	Hexadecimal Position Report			VALID Software Version Y5
SYNTAX aW3	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME <2ms		**SEE ALSO** W1, PR		
RESPONSE TO aW3 **IS** *nnnnnnnn				

Description

This command will respond with an immediate hexadecimal character position report back in 2's complement format. The position response indicates the motor position relative to the current move. The format of the response is an eight digit ASCII hexadecimal number. The **PZ** command does not affect this value.

Assume the response was ***ØØØ433AE**. The decimal value would be:



If the first digit of the response is an *F* (e.g., *Fnnnnnnnn), then the response represents a *2s complement* negative number.

Use the following steps to interpret a negative number (starting with *F*)

The binary approach:

1. Convert the hexadecimal response to binary form.
2. Complement the binary number
3. Add 1 to the binary result
4. Convert the binary result to decimal value with a minus sign placed ahead of the decimal value.

The computer approach:

Subtract the hexadecimal number from 16^8 (4,294,967,296).

The easy way:

1. Leave off all the leading *F*s, and convert to decimal
2. Convert and subtract the next largest power of 16.

Example: *FFFF9E58

1. Leave off the *F*s:	9E58 hex	=	40,536
2. Subtract from 164	10000 hex	=	65,536
	Results	=	-25,000

Example

Command	Response
`> 1W3`	`*FFFFA19C` In the current move, you are at 24,163 steps from initiation of move.

XBS Status	Sequence Buffer Size			VALID Software Version Y5
SYNTAX a**XBS**	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME <2ms		**SEE ALSO** XD, XE, XT		
RESPONSE TO a**XBS** **IS** *n				

Description

This command is used to find out how much space is available for programming sequences. You may program up to 40 sequences, and you may vary the size of the sequence. The Indexer has 1,800 characters available for storing sequences. If you define a sequence 1,800 characters long, you cannot define anymore sequences. This command is useful for finding out how many more characters are available for sequence programming The response includes the space characters.

Example

Command	Description
`> 1XBS`	*15ØØ (You can enter up to 1,500 characters, including the delimiters, for defining sequences.)

XC Status	Sequence Checksum			VALID Software Version Y5
SYNTAX a**XC**	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME <2ms		**SEE ALSO** XD, XE		
RESPONSE TO a**XC** **IS** *nnnn				

Description

This command computes the EEPROM checksum. After the indexer has been programmed, the response can be used for system error checking. The response is in the form nnn, where the range for nnn is 0000 - 0255. The number reported does not indicate the number of bytes programmed. This response is designed to be used for comparison. As long as the indexer is not reprogrammed, the checksum response should always be the same. A checksum error = **1XC = *9999**.

Example

Command	Response
> **1XC**	***149**

XD Programming	Sequence Definition			VALID Software Version Y5
SYNTAX <a>**XD**n	**UNITS** n = sequences	**RANGE** 1 - 40	**DEFAULT** None	**ATTRIBUTES** Buffered Never Saved
EXECUTION TIME <2ms		**SEE ALSO** XE, XR, XT		

Description

This command begins sequence definition for a specific sequence. All the commands between the **XD** command and the Sequence Termination (**XT**) command will be defined as a sequence. The sequences will not be saved until you issue the **SV** command. If a sequence you are trying to define already exists, you must erase that sequence (**XE** command) before defining it. Sequences can be any length, as long as the total number of characters in all sequences totals less than 1800. Sequences can be nested up to 24 levels deep. Nested loops (**L.....N**) and nested **XR** commands can also be used in sequences.

Immediate commands cannot be entered into a sequence.

Example

Command	Description
> XE1	Erase sequence #1
> XD1	Define sequence #1
MN	Set to mode normal
A2	Sets acceleration to 2 g
V5	Sets acceleration to 5 ips
D25ØØØ	Sets distance to 25,000 steps
G	Executes the move (Go)
XT	End defining sequence #1
> XR1	Execute sequence #1
> SV	Saves sequence #1

The commands in sequence 1 are defined, executed, and saved.

XE Programming	Sequence Erase			VALID Software Version Y5
SYNTAX <a>XEn	**UNITS** n = sequences	**RANGE** 1 - 40	**DEFAULT** None	**ATTRIBUTES** Buffered Never Saved
EXECUTION TIME <2ms		**SEE ALSO** XD, XT, XR, SV		
RESPONSE TO aXE **IS** *n				

Description

This command allows you to delete a sequence. The sequence that you specify (n) will be deleted when you issue the command.

As a good practice, you should delete a sequence before defining it.

If you put a device address in front of this command, you will get back a response indicating:

*0 = Existing sequence deleted
*1 = No program existed in the sequence you are deleting

Sequences you erase will not be erased from the EEPROM until you issue a save (SV) command, so if you get back a response of *0, you could cycle power and still keep the old program.

Example

See example for XD command.

XQ Set-Up	Sequence Interrupted Run Mode			VALID Software Version Y5
SYNTAX <a>**XQ**n	**UNITS** N/A	**RANGE** 0, 1	**DEFAULT** 0	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME <2ms		**SEE ALSO** XD, XE, XT, XZ		
RESPONSE TO a**XQ** **IS** *n				

Description

XQ1 = Set interrupted run mode (on)
XQØ = Clear interrupted run mode (off)

This command can be used only when **SSJ1** is enabled. If **XQ1** is executed, the indexer will ignore sequence select inputs, until all sequence select lines have been brought to a high state. After all lines have simultaneously been brought to a high state, the indexer will then read the sequence select lines and execute the sequence whose number appears there. This paused mode will continue until an **XQØ** command is executed. You may use S or K command to stop sequence execution.

The interrupted run mode is cleared at the start of execution of a sequence.

Example

Command	Description
> **XE40**	Erase sequence #40
> **XD40**	Define sequence #40
XQ1	Sets interrupted mode
SSJ1	Enables continuous scan mode
XT	End Sequence #40
> **XR40**	Runs sequence #40

When the indexer power up sequence #40 is executed, the interrupted run mode will be set. Sequence select input lines all need to go high (switch opened) before selecting any subsequent sequences. This will continue until an **XQØ** command is encountered or the continuous scan mode is terminated.

XR Programming	Run A Sequence			VALID Software Version Y5
SYNTAX <a>**XR**n	**UNITS** n = sequences	**RANGE** 1 - 40	**DEFAULT** N/A	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME <2ms		**SEE ALSO** XE, XD, XT, XRP, SN		
RESPONSE TO a**XR** **IS** *n				

Description

This command executes the commands in a predefined sequence (identified by *n*).

An **XR** command can be used within one sequence to start execution of another sequence. An **XR** command acts like GOSUB. If using continuous mode, the velocity must be 0 ips when calling another sequence.

Example

Command	Description
> **XE1**	Erase sequence #1
> **XD1**	Define sequence #1
A2	Sets acceleration to 2 g
V5	Sets acceleration to 5 ips
D25ØØØ	Sets distance to 25,000 steps
G	Executes the move (Go)
XT	End defining sequence #1
> **XR1**	Execute sequence #1

Sequence 1 is defined and executed using **XD1** and **XR1** commands respectively

XRP Programming	Sequence Run With Pause			VALID Software Version Y5
SYNTAX <a>**XRP**n	**UNITS** n = sequences	**RANGE** 1 - 40	**DEFAULT** None	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME <2ms		**SEE ALSO** XR, XD, XT, XE, C, SN		
RESPONSE TO a**XRP** **IS** *n				

Description

This command is identical to the Sequence Run (**XR**) command, except that it automatically generates a pause condition. You must clear this condition with the Continue (**C**) command before the indexer executes the command buffer. The pause condition is asserted only if the sequence is valid. This allows you to execute a sequence without the delay of buffering that sequence.

Example

Command	Description
> **XE5**	Erases Sequence #5
> **XD5**	Defines Sequence #5
A	Sets acceleration to 2 g
V5	Sets velocity to 5 ips
D2ØØØØ	Sets distance to 20,000 steps
G	Executes the move (Go)
XT	Ends defining Sequence #5
> **XRP5**	Runs Sequence #5 with a pause
> **C**	Indexer executes Sequence #5

Upon issuing **XRP5**, Sequence #5 is entered into the command buffer, but is not executed. You must issue a Continue (**C**) command to execute Sequence #5.

XSD Status	Sequence Status			VALID Software Version Y5
SYNTAX a**XSD**	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Buffered Never Saved
EXECUTION TIME <2ms		**SEE ALSO** XD, XE, XT, XU		
RESPONSE TO a**XSD** **IS** *n				

Description

This command reports back the status of the previous sequence definition (**XD**...**XT**). The response is in the form *n. The VALID values and descriptions of possible responses are shown below:

- ***0 =** Sequence definition is O.K.
- ***1 =** A sequence already exists with the number you have specified.
- ***2 =** Out of memory. The sequence buffer is full.

The **XSD** command is useful for verifying that the last sequence definition attempt was successful. To retain the sequence, you must save it with the Save (**SV**) command.

Example

Command	Response
> **1XSD**	*1 (This response indicates that you need to erase the existing sequence to define that specific sequence.)

XSR Status	Sequence Run Status			VALID Software Version Y5
SYNTAX a**XSR**	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Buffered Never Saved
EXECUTION TIME <2ms		**SEE ALSO** XR		
RESPONSE TO a**XSR** **IS** *n				

Description

This command allows you to check whether or not the last sequence you issued was executed successfully without hitting limits, Stop (**S**), or Kill (**K**). The valid values and descriptions for *n* are shown below:

- ***0 =** Last attempt to run the sequence was successful
- ***1 =** Last attempt to run the sequence was unsuccessful
- ***2 =** NON-VALID sequence was requested

Example

Command	Response
> **XR2**	Runs Sequence 2
> **1XSR**	*0 (Sequence ran OK)

XSS Status	Sequence Status			VALID Software Version Y5
SYNTAX a**XSS**n	**UNITS** n = sequences	**RANGE** 1 - 40	**DEFAULT** None	**ATTRIBUTES** Buffered Never Saved
EXECUTION TIME <2ms		**SEE ALSO** XD, XE, XT		
RESPONSE TO a**XSS**n **IS** *n				

Description

This command reports whether the sequence specified by *n* is empty, has a bad checksum, or is OK. The possible responses are as follows:

***0** = Empty
***1** = Bad Checksum
***2** = O.K.

This command is useful to see if the particular sequence exists and if that portion of memory has been corrupted.

Example

Command	Response
> **1XSS1**	*0 (Nothing programmed in sequence #1)

XT Programming	Sequence Termination			VALID Software Version Y5
SYNTAX <a>**XT**	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Buffered Never Saved
EXECUTION TIME		**SEE ALSO** XD, XE, XR, SV		

Description

The **XT** command is a sequence terminator. This command flags the end of the sequence currently being defined. Sequence definition is not complete until this command is issued.

After completing the sequence definition, you must save the sequence into EEPROM with the **SV** command.

Example

Command	Description
> **XD1**	Define sequence #1
MN	Sets move to continuous
A2	Sets acceleration to 2 g
V5	Sets velocity to 5 ips
D25ØØØ	Sets distance to 25,000 steps
G	Executes the move (Go)
XT	End sequence definition
> SV	Saves the sequence

XU Status	Upload Sequence			VALID Software Version Y5
SYNTAX a**XU**n	**UNITS** n = sequences	**RANGE** 1 - 40	**DEFAULT** N/A	**ATTRIBUTES** Buffered Never Saved
EXECUTION TIME		**SEE ALSO** XD, XE, XT, F		
RESPONSE TO a**XU**n **IS** *(contents of sequence **n**)				

Description

This command sends the contents of sequence n to the host computer via RS-232C interface. All commands in sequence n are displayed on the CRT. All command delimiters in the sequence will be sent out as spaces. Any device identifiers that were included in the original sequence will also be eliminated (they are not stored in the sequence).

NOTE: When using a daisy-chain, this command must be used cautiously as the contents of the sequence will go to all controllers in the loop between the indexer that is uploading and the host. The **F** *command may be useful in this context to turn off communication on units you are not uploading from.*

Example

Command	Description
> **F**	Turn off communication to all units
> **1E**	Turn on communication to unit #1
> **1XU7**	Upload sequence #7 from unit #1:
	***A2_V5_D1Ø_L_G_H_N**
> **E**	Enable other units

Y Programming	Stop Loop			VALID Software Version Y5
SYNTAX <a>**Y**	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME <2ms		**SEE ALSO** L, N		

Description

The Stop Loop (**Y**) command takes you out of a loop when the loop completes its current pass. This command does not halt processing of the commands in the loop until after the indexer processes the last command of the current loop. At that time, the indexer executes the command that follows the End Loop (**N**) command. *NOTE: The* **Y** *command cannot be used to stop execution of nested loops.* **Y** *only terminates the current active loop. To stop nested loop execution, use the* **U**, **S**, *or* **K** *commands.*

Example 1

Command	Description
> L	Loops indefinitely
> A2	Sets acceleration to 2 g
> V5	Sets velocity to 5 ips
> D25ØØØ	Sets distance to 25,000 steps
> T2	Waits 2 seconds
> G	Executes the move (Go)
> N	Ends loop
> Y	Stops loop

The loop requires the motor to move 25,000 + steps and then wait for 2 seconds. The loop terminates at the end of the loop cycle it is executing when it receives the **Y** command.

Example 2

Command	Description
> L	Loops indefinitely
> A2	Sets acceleration to 2 g
> V50	Sets velocity to 50 ips
> D1ØØØØ	Sets distance to 10,000 steps
> L	Loops indefinitely
> G	Executes the move (Go)
> H	Changes direction of motion
> T.1	Waits 0.1 seconds
> N	Ends loop
> A1	Sets acceleration to 1 g
> V25	Sets velocity to 25 ips
> D25ØØ	Sets distance to 2,500 steps
> L	Loops indefinitely
> G	Executes the move (Go)
> H	Changes direction of motion
> T.5	Waits 0.5 seconds
> N	Ends loop
> N	Ends loop

Entering a Y command exits the first infinite loop and starts the second infinite loop. Entering a second Y command exits the second loop and starts over at the first infinite loop.

Z Programming	Reset			VALID Software Version Y5
SYNTAX <a>Z	**UNITS** N/A	**RANGE** N/A	**DEFAULT** N/A	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME 1 second		**SEE ALSO** S, K		

Description

The Reset (Z) command is equivalent to cycling AC power to the indexer. This command returns all internal settings to their power-up values unless they were changed by the **SV** command. It clears the command buffer. Like the Kill (**K**) command, the Z command immediately stops output pulses to the motor.

Any commands entered while resetting are ignored.

This command sets all position counters to zero.

Chapter 6. HARDWARE REFERENCE

Chapter Objectives

This chapter is designed to function as a quick-reference tool for the following information:

- System specifications (dimensions & performance)
- Proper I/O connections
- DIP switch settings

System Specifications

LX Specifications

Parameter	Value
Amplifier	
Type	20 kHz fixed frequency, variable duty cycle, pulse-width modulated. Current controller, bipolar type. MOSFET construction.
Number of Phases	Two
Protection (auto-shutdown if detected):	
Short-circuit	Phase-to-phase (NOT phase-to-ground)
Brownout	If AC supply drops below 95 VAC
Over-temperature	If internal air temperature exceeds 162°F (75°C)
Current Rating	0 - 3 amps/phase (DIP switch-selectable)
Input Power	
Voltage	95-132 VAC, single phase
Frequency	47 - 66 Hz
Current	3A maximum continuous (RMS)
Output Power to Forcer	
Voltage	170 VDC (20 kHz PWM)
Current	3A maximum
Command Interface	
Type	RS-232C, 3-wire (Tx, Rx, GND). Required min. voltage swing on Rx line is ±3V.
Communications parameters	Fixed at 9600 baud, 8 data bits, 1 stop bit, no parity
Configuration	Up to 8 LX indexer/drives may controlled from a single RS-232C port when connected in a daisy-chain configuration.
Performance	
Position range	0 - 1,999,999,999 steps
Coordinate system	Absolute or incremental
Velocity range	0.01 - 75.0 inches/second @ 10,000 steps/inch resolution
Acceleration range	0.01 - 10.0 g's
Signal Output	Open-collector step, direction, and shutdown signals
Sequence Storage	
Memory type	EEPROM (1,800 characters)
Number of motion programs	40
Program length	Variable sequence lengths
Accelerometer	
Input Signal	Analog accelerometer output voltage
Power	±12 VDC at 15mA provided to power the accelerometer
Accelerometer	Strain gauge output
Inductance	
Minimum Forcer Inductance	5 mH/phase measured end-to-end
Environmental	
Driver Operating Temperature	32° to 113°F (0° to +45°C). Optional fan kit available for extra cooling.
Storage Temperature	-40° to 185°F (-40° to +85°C)
Humidity	0 to 95% Non-condensing

Table 6-1. LX Indexer/Drive Specifications

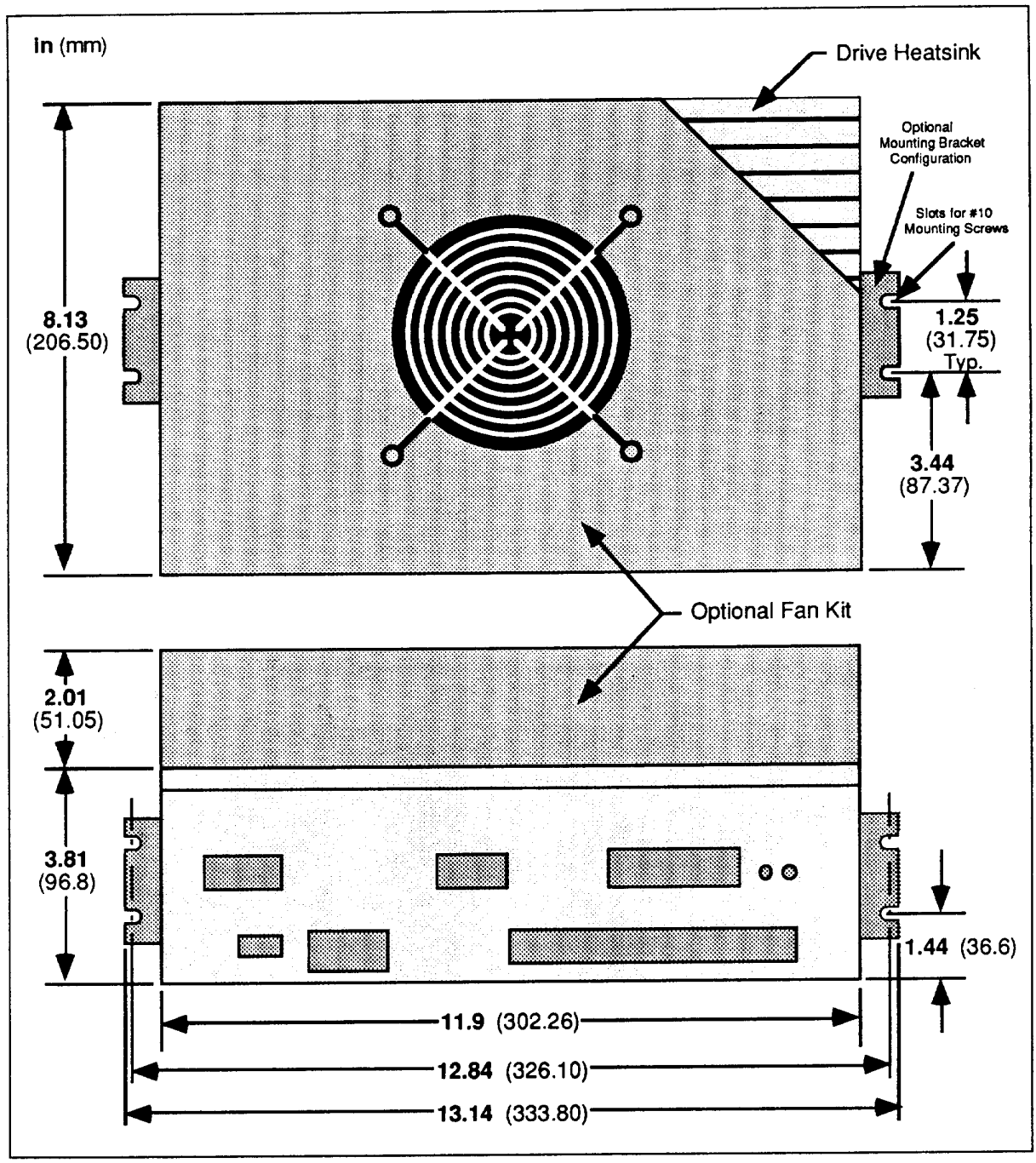


Figure 6-1. LX Dimensional Drawing (includes fan kit

LX-L3C System Specifications

L3C System	
Static Force	3.0 lbs (1.36 kgs)
Normal force between forcer and platen	28.0 lbs (12.7 kgs)
Accuracy (worst case)	±0.004 in (100 microns)*
Repeatability (unidirectional)	±0.0003 in (±7.5 microns)
Hysteresis	0.0008 in (20 microns)
Cyclic error (TIR/0.0167")	±0.002 in (50 microns)
Platen Errors: Cumulative Non-cumulative random Thermal expansion	 0.0003 in/in (0.3 µm/mm) ±0.0006 in (15 microns) 0.00000633 in/in/°F (11.4 microns/meter/°C)
Straightness of travel	0.005 inches TIR** (127 microns)
Pitch Torque	Max: 180 oz-in (13.0 kg-cm)
Roll Torque	Max: 310 oz-in (22.0 kg-cm)
L3C Forcer	
Type	Two-phase, PM hybrid
Current per phase	1.0A
Maximum forcer case temp	190°F (88°C)
Operating Temperature	32°F - 104°F (0°C - 40°C)
Bearing type	Ball bearing, ABEC 1 (Top and Side)
Air gap	0.003 in (75 microns)
Maximum forcer load	Top 10.0 lb (4.54 kg); Side 3 lbs (1 kg)
Forcer weight	1 lb (0.45 kg)
L3C Platen	
Flatness (assumes flat mounting)	0.003 in TIR (75 microns)
Standard length	15 in (381 mm)
Weight	2.42 oz/in (27.0 gm/cm)
Base material	Low carbon steel
Surface flatness required for forcer operation	0.0001 in/4 in (25 microns/100 mm)
Tooth Pitch	0.0667 in (1.693 mm)

* Platen heating error not included
** TIR = Total Indicated Reading (difference between maximum and minimum reading)

Table 6-2. LX-L3C System Specifications

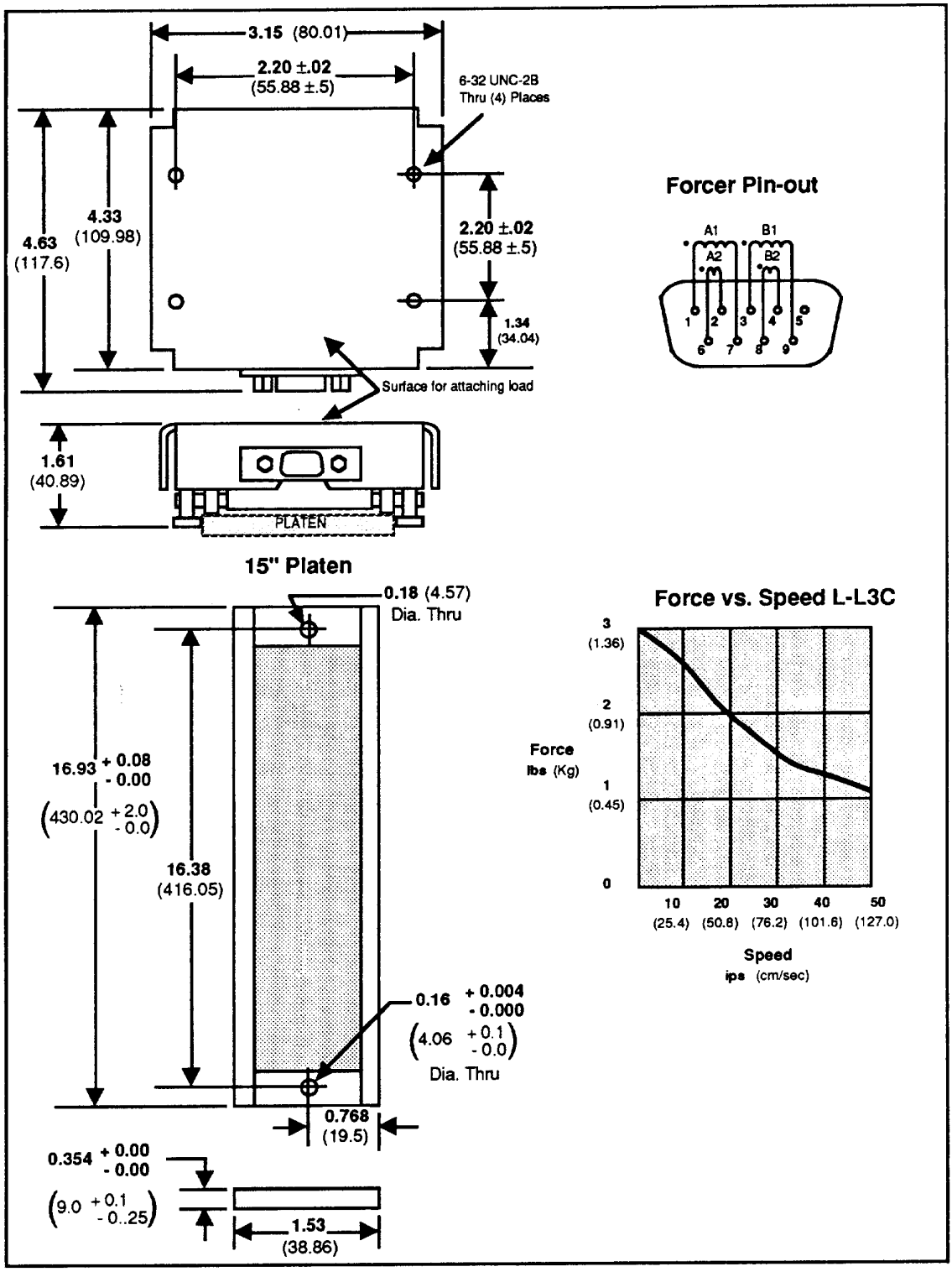


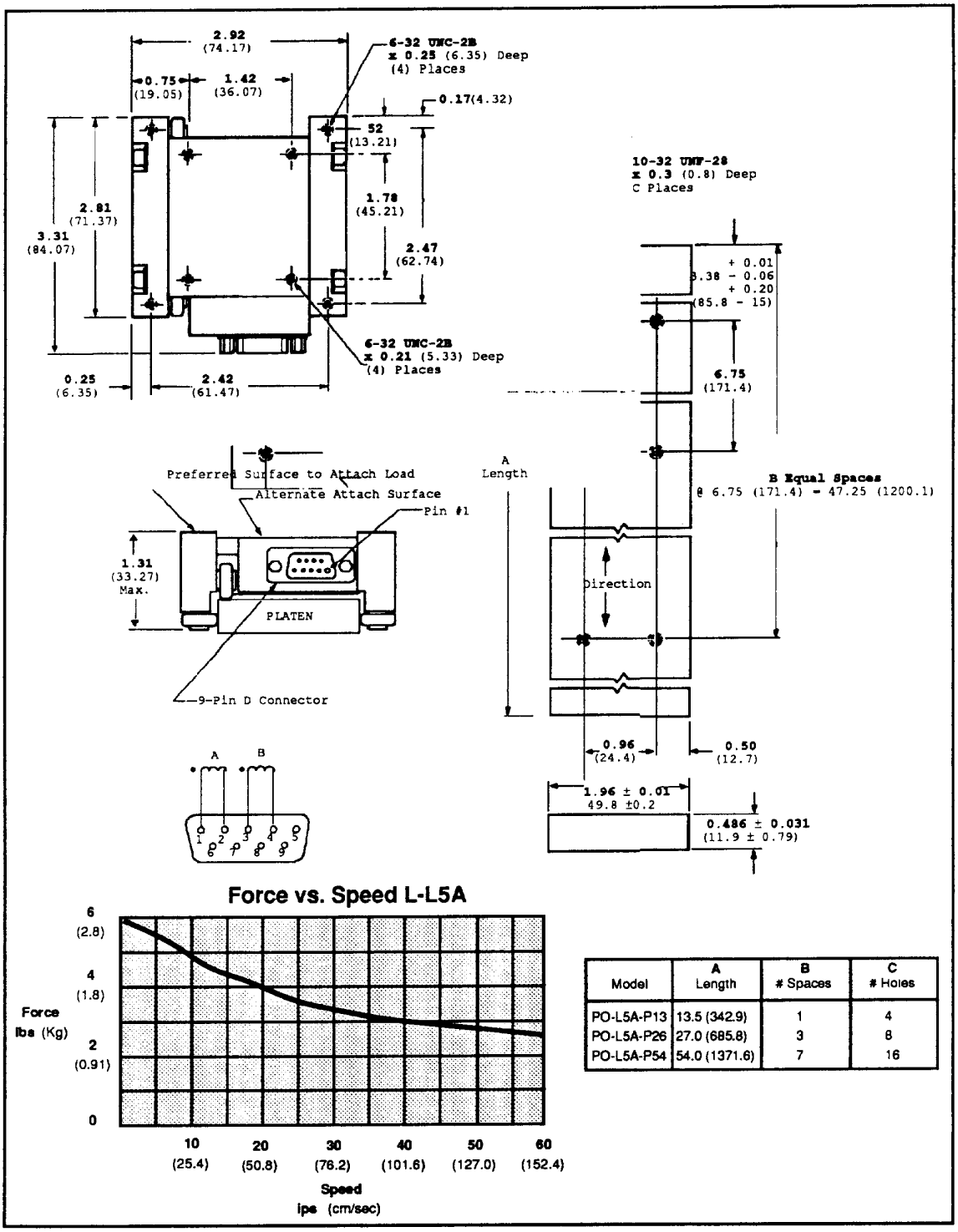
Figure 6-2. LX-L3C Dimensional Drawing and Performance Curve

LX-L5A System Specifications

L5A System	
Static Force	6.0 lbs (2.27 kgs)
Normal force between forcer and platen	46.0 lbs (20.9 kgs)
Accuracy (worst case)	±0.0035 in (90 microns) *
Repeatability (uni-directional)	±0.0003 in (±7.5 microns)
Hysteresis	0.0008 in (20 microns)
Cyclic error	±0.002 in (±50.0 microns)
Platen Errors: Cumulative Non-cumulative random Thermal expansion	 0.0001 in/in (0.1 micron/mm) ±0.0006 in (15 microns) 0.00000633 in/in/°F (11.4 microns/meter/°C)
Straightness of travel	0.002 in TIR (50 microns)
Pitch Torque	Max: 90 oz-in (7.0 kg-cm)
Roll Torque	Max: 190 oz-in (14.0 kg-cm)
L5A Forcer	
Type	Two-phase, PM hybrid
Current per phase	1.5A
Maximum forcer case temp	190°F (88°C)
Operating Temperature	32°F - 104°F (0°C - 40°C)
Bearing type	Ball bearing, ABEC 3 (Top and Side)
Air gap (typical)	0.0015 in (37.5 microns)
Maximum forcer load	Top 10.0 lbs (4.54 kgs); Side 10.0 lbs (4.45 kgs)
Forcer weight	0.8 lbs (0.36 kgs)
L5A Platen	
Flatness (assumes flat mounting)	0.002 in TIR (50 microns)
Standard lengths	13, 27, or 54 inches (330, 686, 1372 mm)
Weight	4.15 oz/in (45.0 gm/cm)
Platen material	1018 steel
Platen plating	Chromium, class 2B
Surface flatness required for forcer operation	0.001 in/4 in (25 microns/100 mm)
Tooth pitch	0.040 in (1.016 mm)

* Platen heating error not included

Table 6-3. LX-L5A System Specifications



Model	A Length	B # Spaces	C # Holes
PO-L5A-P13	13.5 (342.9)	1	4
PO-L5A-P26	27.0 (685.8)	3	8
PO-L5A-P54	54.0 (1371.6)	7	16

Figure 6-3. LX-L5A Dimensional Drawing and Performance Curve

LX-L9A System Specifications

L9A System	
Static Force	10.0 lbs (4.54 kgs)
Static normal force between forcer and platen	80.0 lbs (36.4 kgs)
Accuracy (worst case)	±0.0025 in*
Repeatability (uni-directional)	±0.0001 in (±2.5 microns)
Hysteresis	0.0005 in (12.5 microns)
Cyclic error (TIR/0.01")	±0.0015 in (35.5 microns)
Platen Errors: Cumulative Non-cumulative random Thermal expansion	 ±0.0001 in/in (0.1 µm/mm) ±0.0004 in (10 microns) 0.00000633 in/in/°F (11.4 microns/meter/°C)
Straightness of travel	0.002 in TIR (50 microns)
Pitch Torque	Max: 120 oz-in (9.0 kg-cm)
Roll Torque	Max: 120 oz-in (9.0 kg-cm)
L9A Forcer	
Type	Two-phase, PM hybrid
Current per phase	1.5A
Maximum forcer case temp	167°F (75°C)
Operating Temperature	32°F - 104°F (0°C - 40°C)
Bearing type	Ball bearing, ABEC 3 (top and side)
Air gap (typical)	0.0005 in (12.5 microns)
Maximum forcer load	Top: 50 lbs (22.7 kgs); Side: 10 lbs (4.54 kgs)
Forcer weight	1.4 lbs (0.64 kgs)
L9A Platen	
Flatness (assumes flat mounting)	0.002 in TIR (50 microns)
Standard lengths	13, 27, or 54 inches (330, 686, 1372 mm)
Weight	4.15 oz/in (45.0 gm/cm)
Platen material	1018 steel
Bottom and sides surface plating (top is untreated)	Chromium, class 2B
Surface flatness required for forcer operation	0.00025 in/5 in (0.635 microns/125 mm)
Tooth pitch	0.040 in (1.016 mm)

* Platen heating error not included

Table 6-4. LX-L9A System Specifications

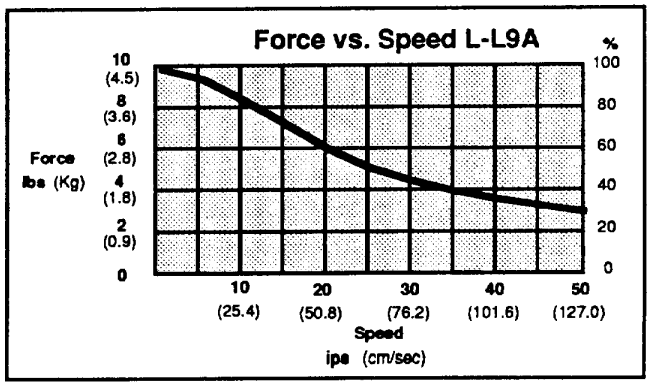


Figure 6-4. LX-L9A Performance Curve

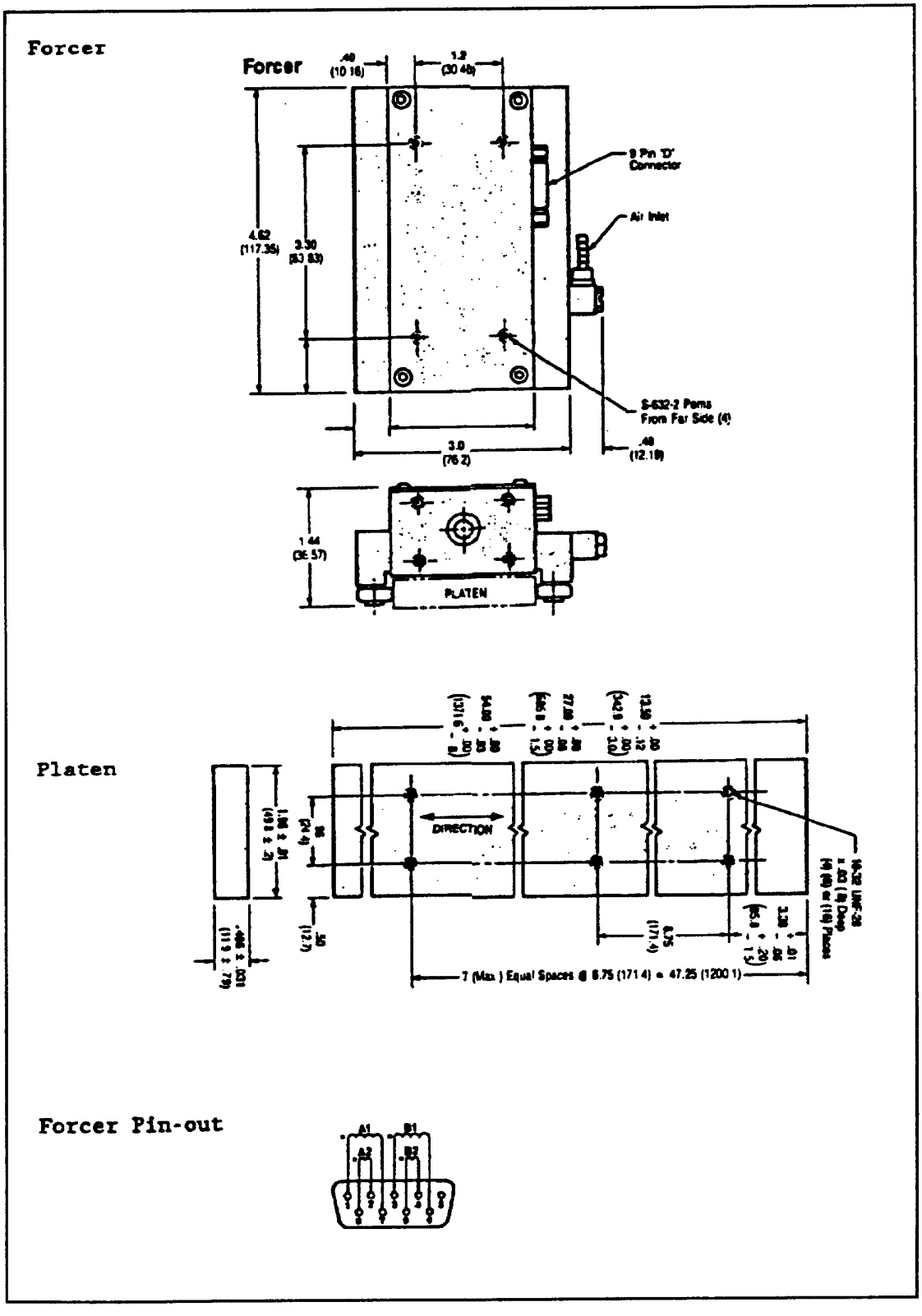


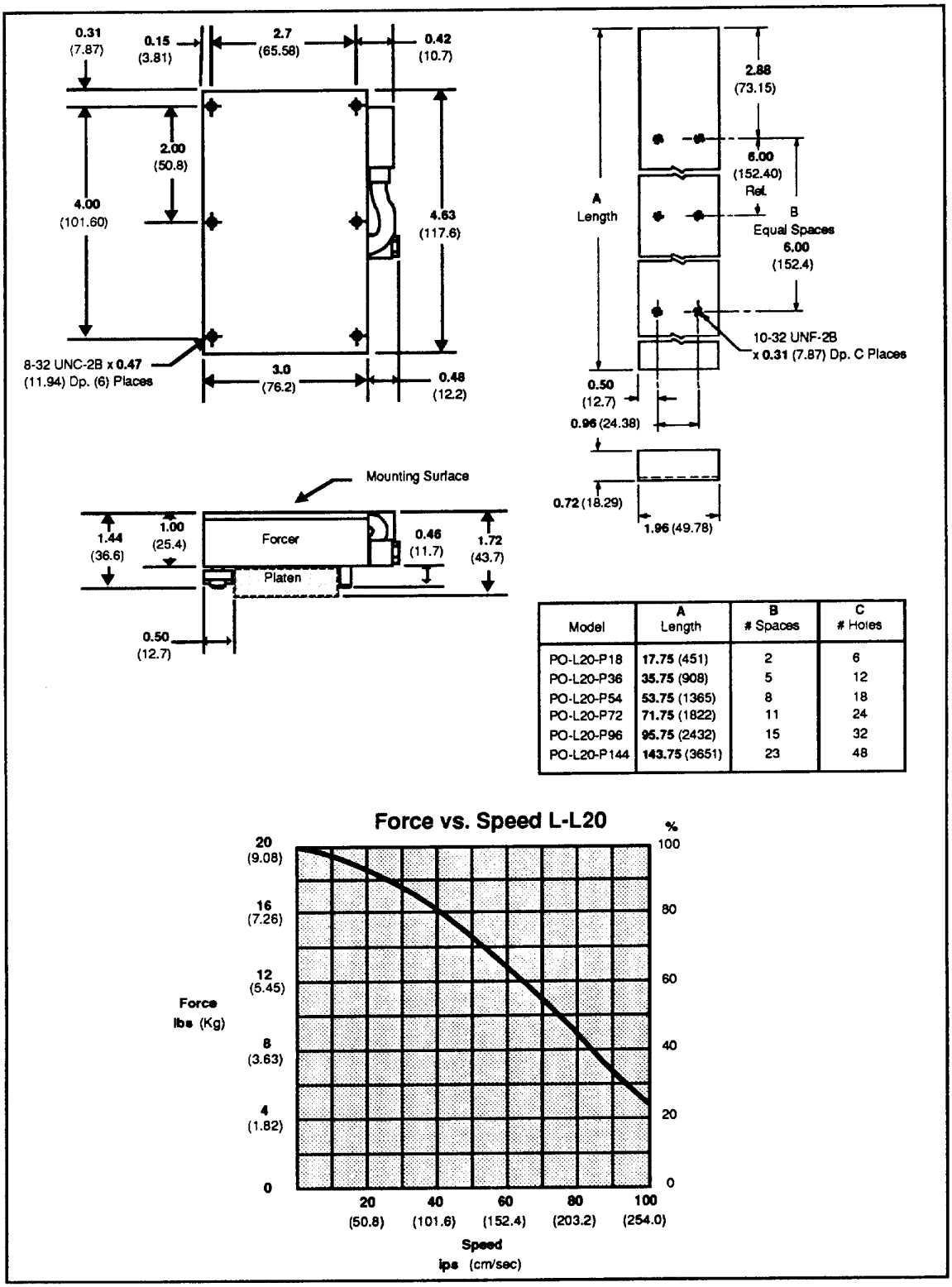
Figure 6-5. LX-L9A Dimensional Drawing

LX-L20 System Specifications

L20 System	
Static Force	20.0 lbs (9.087 kgs)
Static normal force between forcer and platen	180.0 lbs (81.8 kgs)
Accuracy (worst case)*	±0.0035 in (90 microns)
Repeatability (uni-directional)	±0.0001 in (±2.5 microns)
Hysteresis	0.0005 in (12.5 microns)
Cyclic error (TIR/0.01")	±0.0015 in (±37.5 microns)
Platen Errors: Cumulative Non-cumulative random Thermal expansion	 ±0.0001 in/in (0.1 µm/mm) ±0.0015 in (±37.5 microns) ± 0.00000633 in/in/°F (11.4 microns/meter/°C)
Straightness of travel	0.0025 inches TIR (62.5 microns)
Yaw Torque**	Max: 90 oz-in (7.0 kg-cm)
Pitch Torque**	Max: 120 oz-in (9.0 kg-cm)
Roll Torque**	Max: 120 oz-in (9.0 kg-cm)
L20 Forcer	
Type	Two-phase, PM hybrid
Current per phase	2.7A
Operating temperature, forcer (Ta)	32° - 104°F (0° - 40°C)
Maximum forcer case temp	167°F (75°C)
Operating Temperature	32°F - 104°F (0°C - 40°C)
Bearing type	Top: Air bearing, forced air @ 40 PSI, 20 SCFH; Side: Ball bearing (2), ABEC 3, 8 lb magnetic preload
Air gap (typical)	0.0005 in (12.5 microns)
Maximum forcer load***	Top: 50 lbs (22.7 kgs); Side: 5 lbs (2.27 kgs)
Forcer weight	2 lbs (0.8 kgs)
L20 Platen	
Flatness (assumes flat mounting)	0.002 in TIR (50 microns)
Standard lengths	18, 36, 54, 72, 96, or 144 inches (457, 914, 1372, 1828, 2438, 3658 mm)
Weight	6.23 oz/in (67.5 gm/cm)
Platen material	1018 steel
Bottom and side surface plating	Nickel plated (top is treated to prevent corrosion)
Surface flatness required for forcer operation	0.00025 in/5 in (0.635 microns/125 mm)
Tooth pitch	0.040 in (1.016 mm)

* Platen heating error is not included.
** Refer to Chapter 3, Installation, for a description of *Yaw, Pitch, and Roll.*
*** *Top* load refers to the vertical force on top of the forcer in the normal horizontal (flat) mounting configuration. *Side* load refers to the side force (perpendicular to the platen length, pressing on the forcer side bearings) on a forcer when the forcer/platen is mounted vertically.

Table 6-5. LX-L20 System Specifications



Model	A Length	B # Spaces	C # Holes
PO-L20-P18	17.75 (451)	2	6
PO-L20-P36	35.75 (908)	5	12
PO-L20-P54	53.75 (1365)	8	18
PO-L20-P72	71.75 (1822)	11	24
PO-L20-P96	95.75 (2432)	15	32
PO-L20-P144	143.75 (3651)	23	48

Figure 6-6. L20 Dimensional Drawings and Performance Curve

Connector Summary and Pinouts

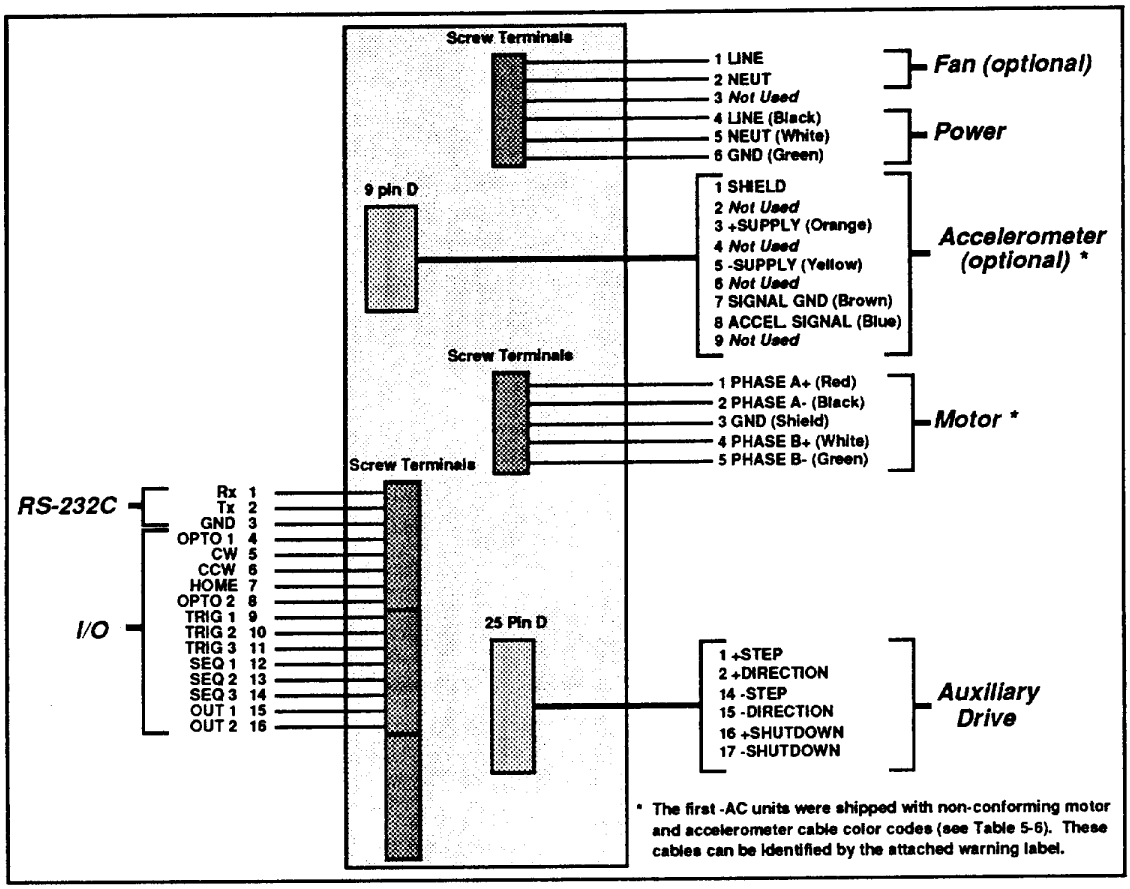


Figure 6-7. LX Connectors and Pinouts

Accelerometer		
Pin	Description	Color
1	Shield	Shield
2	*not used*	———
3	+Supply	Red
4	*not used*	———
5	-Supply	White
6	*not used*	———
7	Signal GND	Black
8	Accel. Signal	Green
9	*not used*	———

Motor		
Pin	Description	Color
1	Phase A+	Brown
2	Phase A-	Blue
3	GND	Shield
4	Phase B+	Orange
5	Phase B-	Yellow

Table 6-6. Non-Conforming Color Codes for L20-AC Forcer and Accelerometer Cables (Initial Shipments Only)

Wire Gauges

Accelerometer		
Pin	Description	Gauge
1	Shield	-------
2	*not used*	-------
3	+Supply	30AWG
4	*not used*	-------
5	-Supply	30AWG
6	*not used*	-------
7	Signal GND	30AWG
8	Accel. Signal	30AWG
9	*not used*	-------

Motor		
Pin	Description	Gauge
1	Phase A+	22AWG
2	Phase A-	22AWG
3	GND	-------
4	Phase B+	22AWG
5	Phase B-	22AWG

Table 6-7. Wire Gauges

I/O Descriptions

Refer to Chapter 3 for I/O wiring diagrams.

Optional Fan Output

Output power for the optional fan kit is 110VAC (500mA).

Input Power Connections

Input power pinouts and color codes are illustrated in Figure 6-7. Note that AC power is limited to 95VAC to 132VAC. **Higher voltages damage the LX.**

Motor Connections

Motor (forcer) pinouts and color codes are illustrated in Figure 6-7. Output power is 170VDC switching at 20 kHz.

RS-232C

RS-232C Transmit (Tx), Receive (Rx) and Ground (GND) Connections. The LX accepts standard EIA RS-232C signals from +15V to -15V. RS-232C communication parameters are set at 9,600 baud, 8 data bits, 1 stop bit, no parity, and no echo (full duplex). *NOTE: The LX does not support hardware or software handshaking.*

OPTO 1

This input provides power for the optical isolation of CW, CCW, Home limit inputs, Trigger inputs, and Sequence Select inputs. You need to supply 5 to 12VDC (450mA) to power these I/Os.

OPTO 2

OPTO 2 must be connected to the logic ground of the user-supplied 5 to 12VDC power supply powering the I/Os.

TRIG1, TRIG2, and TRIG3

These triggers are optically isolated inputs. They are normally open (no current flowing). To activate the Trigger input, you need to close the Trigger input to GND (OPTO 2).

You can also use these inputs as sequence select inputs by reconfiguring the inputs with the Input Mode (IM) command. In IM3 mode, Trigger #1 functional as a remote stop input.

SEQ1, SEQ2, and SEQ3

These are optically isolated inputs and are normally open (no current flows). To activate sequence input, you need to close the sequence input to the user-supplied 5 to 12VDC power supply GND to OPTO2.

You can use these inputs to execute sequences with a PLC, a manual switch, or an electronic relay.

You can also use these inputs as trigger inputs with a Trigger (TR) command.

OUT1 and OUT2

These are optically isolated outputs. They are normally off (0VDC and no current flows) You can use these outputs as Programmable Outputs. You can use the Output Mode (OM) command to reconfigure the outputs.

The OUT1 and OUT2 outputs are the collectors of a transistor; therefore, you need to supply an external 5 to 12VDC supply with a 10kΩpull up resistor.

+5V Output

This output serves as the isolated power supply for the quadrature encoder. It provides up to 250mA.

LX DIP Switch Settings

You can select options and functions for the motor being used and your application. If purchased with a linear motor the LX is configured at the factory for the proper current and standard resolution of the system. To change any of the settings, use the DIP switches. Refer to Figure 2-1 in Chapter 2 for the location of the DIP switches. The ON/OFF switch convention is as follows:

- **ON** = DIP switch moved towards the PCA
- **OFF** = DIP switch moved away from the PCA

The following procedure should be followed when adjusting the LX's DIP switches:

STEP 1 Remove power.

STEP 2 Check all DIP switch settings. Use the tables in this manual to ensure that the switches are set for the proper current, motor resolution, address, etc.

STEP 3 If you must make DIP Switch adjustments, use a narrow instrument, such as a thin, standard screw driver.

Motor Current Settings

	Drive DIP Switch Settings				
Motor Current (amps)	**1**	**2**	**3**	**4**	**5**
0.000*	OFF	OFF	OFF	OFF	OFF
0.093	OFF	OFF	OFF	OFF	ON
0.186	OFF	OFF	OFF	ON	OFF
0.279	OFF	OFF	OFF	ON	ON
0.396	OFF	OFF	ON	OFF	OFF
0.489	OFF	OFF	ON	OFF	ON
0.582	OFF	OFF	ON	ON	OFF
0.675	OFF	OFF	OFF	OFF	OFF
0.775	OFF	ON	OFF	OFF	OFF
0.868	OFF	ON	OFF	OFF	ON
0.961 (L3C)**	**OFF**	**ON**	**OFF**	**ON**	**OFF**
1.054	OFF	ON	OFF	ON	ON
1.171	OFF	ON	ON	OFF	OFF
1.264	OFF	ON	ON	OFF	ON
1.357	OFF	ON	ON	ON	OFF
1.450 (L5A & L9A)**	**OFF**	**ON**	**ON**	**ON**	**ON**
1.550	ON	OFF	OFF	OFF	OFF
1.643	ON	OFF	OFF	OFF	ON
1.736	ON	OFF	OFF	ON	OFF
1.824	ON	OFF	OFF	ON	ON
1.946	ON	OFF	ON	OFF	OFF
2.039	ON	OFF	ON	OFF	ON
2.132	ON	OFF	ON	ON	OFF
2.225	ON	OFF	ON	ON	ON
2.325	ON	ON	OFF	OFF	OFF
2.418	ON	ON	OFF	OFF	ON
2.511	ON	ON	OFF	ON	OFF
2.604	ON	ON	OFF	ON	ON
2.721 (L20)**	**ON**	**ON**	**ON**	**OFF**	**OFF**
2.814	ON	ON	ON	OFF	ON
2.907	ON	ON	ON	ON	OFF
3.000	ON	ON	ON	ON	ON

** Factory Default for LX-only (shipped without a forcer)*
*** Factory Default for the forcer indicated*

Table 6-8. Motor Current Settings

Automatic Run Settings

CAUTION

Be sure that the forcer is free to move. *Be ready to remove power.*

Automatic Run is enabled and disabled with Drive DIP switch #6. Auto standby, motor resolution, and waveform options are disabled when the Automatic Run mode is used.

SW6 - ON Disables the function*
SW6 - OFF Enables the function

** Factory Default*

Automatic Standby Settings

Automatic Standby is enabled/disabled with Drive DIP switch #7. When enabled, this feature drops the current to the forcer by 50% when the forcer receives no step pulses for one second. Full power is restored when a step pulse is received.

SW7 - ON Disables the function* SW7 - OFF Enables the function

** Factory Default*

Motor Resolution Settings

Table 6-9 shows motor resolution settings for the standard LX system using the Drive DIP switches 8 - 10.

Microsteps Per Full Step	Steps/In L20, L9A & L5A	Steps/In L3C	Drive DIP Switch Settings 8	9	10
125	12,500	7.500	ON	ON	ON
*100**	*10,000**	*6,000**	*OFF*	*ON*	*ON*
90	9,000	5,400	ON	OFF	ON
50	5,000	3,000	OFF	OFF	ON

**Factory Setting*

Table 6-9. Motor Resolution Settings (Drive DIP switches 8 - 10)

-AC Option Resolution Settings

For proper operation of the LX L20-AC system, the resolution DIP switch settings on the -AC DIP switch must be the same as those on the Drive DIP switch. Note that not all Drive resolutions are available on the -AC DIP switch (see Tables 6-10 and 6-11). ***Selecting the wrong resolution setting reduces motor performance.***

The -AC can be made compatible with other resolutions as a special through Compumotor's Special Products Group.

Drive DIP Switch Settings			Standard LX Firmware
8	9	10	Standard resolution
OFF	OFF	OFF	*invalid*
ON	OFF	OFF	*invalid*
OFF	ON	OFF	*invalid*
ON	ON	OFF	*invalid*
OFF	OFF	ON	5,000 steps/in pure*
ON	OFF	ON	9,000 steps/in pure
OFF	ON	ON	10,000 steps/in pure*
ON	ON	ON	12,500 steps/in pure

** Resolutions Compatible with the -AC option*
NOTE: Factory Default Setting is 10,000 steps/in

Table 6-10. Drive DIP Switch Resolution Settings

-AC DIP Switch Settings			Resolution
4	5	6	
OFF	OFF	OFF	RESERVED
ON	OFF	OFF	5,000 steps/in
OFF	ON	OFF	*do not use*
ON	ON	OFF	*do not use*
OFF	OFF	ON	RESERVED
ON	OFF	ON	10,000 steps/in*
OFF	ON	ON	RESERVED
ON	ON	ON	20,000 steps/in

* ***Factory Default Setting***

Table 6-11. -AC DIP Switch Resolution Settings

Motor Waveform Settings

Use Drive DIP switches 11 and 12 to select the current waveform which provides the smoothest operation (see Table 6-12).

Waveform Shape	11	12
Pure Sine	OFF	OFF
+4% 3rd Harmonic	ON	OFF
-4% 3rd Harmonic	OFF	ON
Pure Sine*	ON	ON

* ***Factory Default Setting***

Table 6-12. Motor Waveform Settings

Gain Settings (-AC Option Only)

To achieve a damping ratio of approximately 0.707, you must set -AC DIP switches 1 - 3 according to Table 6-13. Too much gain makes the forcer behavior sluggish, taking longer for the forcer to reach the end-point of the move. Too little gain will make the forcer ring. This reduces the performance of the system by requiring a larger safety margin. With no gain (all DIP switches on), the forcer will ring the same as a system without accelerometer feedback.

-AC DIP Switch Settings			Payload
1	2	3	
OFF	OFF	OFF	77 to 105 lb**
ON	OFF	OFF	53 to 77 lb**
OFF	ON	OFF	33 to 53 lb**
ON	ON	OFF	17.6 to 33 lb
OFF	OFF	ON	6.6 to 17.6 lb
ON	OFF	ON	1.0 to 6.6 lb
OFF	ON	ON	0 to 1.0 lb*
ON	ON	ON	disable accelerometer feedback (gain 0)

* ***Factory Default Setting***

Table 6-13. Gain Settings for Different Payloads

** *NOTE: The maximum recommended payload for the L20 forcer is 50 lbs. The L20 forcer can move higher payloads if it is supported by some kind of linear guide along the platen.*

Profile Settings

Use Indexer DIP switches 1, 2, and 3 to select the proper acceleration profile for the forcer you are using (see Table 6-14). Each profile is stored in ROM.

Indexer DIP Switch Settings			
1	2	3	Forcer
OFF	OFF	ON	L3C
OFF	ON	ON	L5A
ON	OFF	OFF	L9A
ON	OFF	ON	L20
OFF	OFF	OFF	LX-ONLY*

** Factory Setting for LX shipments without motors*

Table 6-14. Profile Settings

Address Settings

Use Indexer DIP switches 4, 5, and 6 to select different device addresses for daisy-chaining (see Table 6-15).

Indexer DIP Switch Settings			
4	5	6	Address
ON	ON	ON	1*
ON	ON	OFF	2
ON	OFF	ON	3
ON	OFF	OFF	4
OFF	ON	ON	5
OFF	ON	OFF	6
OFF	OFF	ON	7
OFF	OFF	OFF	8

** Factory Default*

Table 6-15. Device Address Settings

Chapter 7. MAINTENANCE & TROUBLESHOOTING

Chapter Objectives

The information in this chapter will enable you to do the following:

- Maintain the system's components to ensure smooth, efficient operation
- Isolate and resolve system hardware problems
- Use this chapter as a quick-reference tool for a description of system error codes

Maintenance

The following system components require periodic maintenance:

- The Forcer
- The LX Indexer/Drive
- The Platen

Spare Parts Table

Table 7-1 provides a list of recommended spare parts to use with the L Drive system.

Description	Compumotor Part #
Forcer Cables: L3C, L5A, L9A L20 L20-AC*	 71-007553-12 71-008024-12 71-010635-12
AC Power Cable	71-006593-01
6-Position Power Connector	43-006606-01
5-Position Motor Connector	43-005561-01
Mounting Brackets	53-006007-01
Air Regulator/Filter Kit (optional)	-RFKC
Fan Kit (optional)	-LFK
Male 25-Pin D-Connector	43-001989-01
D-Connector Shell	43-001990-01

** This cable must be factory-installed.*

Table 7-1. Spare Parts List

Forcer Maintenance

You should inspect all mechanical parts of the forcer regularly to ensure that no bolts have become loose during normal operation. This will prevent some minor problems from developing into more serious problems.

The ball bearings used in the L3C, L5A, L9A, and L20 forcers are not sealed against severe environments, but are permanently lubricated and require no maintenance.

You should inspect the forcer cable periodically for signs of wear. This inspection interval is duty-cycle, environment, and travel-length dependent. You should not apply excessive tensile force to the cable. Also, you should not bend the cable beyond a two-inch radius of curvature during normal operation. Tighten all cable connectors.

LX Maintenance

Check that the LX heatsink is free of dust and has a free flow of air over its entire surface. The LX should be allowed to function without shutting itself down at an internal temperature of 165°F (75°C).

Enclosures must be connected to earth ground through a grounding electrode conductor to provide a low-impedance path for ground-fault or noise-induced currents. All earth ground connections must be continuous and permanent.

Platen Maintenance

If you shorten the length of your platen, treat the end of the platen with non-corrosive Rust-Oleum™. If your platen is corroded, use lapping paper to remove the corrosion, then clean it with rubbing alcohol. Compumotor recommends the following Lapping paper: *3M* 261X Imperial Lapping Film (8.5 x 11 in sheets) A/O (Aluminum Oxide) 3.0 Mil.

Periodically check the platen mounting bolts for tightness. Periodically clean the platen surfaces with alcohol or acetone, even in relatively clean environments such as normal engineering labs and offices. In dirtier environments, a protective boot may be needed. The boot covers both the platen and the forcer. As the forcer moves, the boot folds (like an accordion).

Shiny spots or scratches on the platen surface indicate that the forcer is not floating properly on the air bearing. This is due to improper platen mounting and/or clogged air orifices in the L20 or L9A forcer. Use the following procedure to check and clean clogged orifices:

STEP 1 Remove the non-energized forcer carefully from the platen.

STEP 2 With the air on, check with your finger tips for a fine stream of air from the orifices. The air flow for each orifice should be similar.

STEP 3 When a clogged orifice is found, turn off the air and remove the orifice with a small screw driver.

STEP 4 Dislodge the debris from the orifice with a small piece of wire (≈0.01 inches in diameter) or with a burst of air.

STEP 5 Turn the air on to remove other debris in the air channels of the forcer.

STEP 6 Reinstall the orifice and repeat the process to check for other clogged orifices.

Troubleshooting

This section discusses methods to identify, isolate, and resolve problems that may occur with your L Drive system.

Problem Isolation

When your system does not function properly (or as you expect it to operate), the first thing that you must do is identify and isolate the problem. When you accomplish this, you can effectively begin to eradicate and resolve the problem.

The first step is to isolate each system component and ensure that each component functions properly when it is run independently. You may have to dismantle your system and put it back together piece by piece to detect the problem. If you have additional units available, you may want to use them to replace existing components in your system to help identify the source of the problem.

Try to determine if the problem is mechanical, electrical, or software-related. *Can you repeat or recreate the problem?* Do not attempt to make quick rationalizations about problems. Random events may appear to be related, but they are not necessarily contributing factors to your problem. You must carefully investigate and decipher the events that occur before the subsequent system problem.

You may be experiencing more than one problem. You must solve one problem at a time. Log (document) all testing and problem isolation procedures. You may need to review and consult these notes later. This will also prevent you from duplicating your testing efforts.

Once you have isolated the problem, take the necessary steps to resolve it. Refer to the problem solutions contained in this chapter. If your system's problem persists, contact Parker Compumotor's Applications Department at (800) 358-9070.

WARNING

Be sure to remove power before disconnecting LX system components or changing wiring.

Forcer Fails to Move

Test the forcer to see if it has holding force. If there is no holding force, here are some probable causes:

- There is no AC power.
- The forcer cable is disconnected.
- There are bad connections or bad cables (disconnect power connector, then use an ohm meter to monitor continuity).
- The drive may be disabled through software (`ST1` command)

If the unit has holding force but does not move, here are some probable causes:

- There is no air to the air bearing.
- The limit switches have been tripped or are faulty.
- The load is jammed. You should hear the drive attempting to move the forcer. Remove AC power from the driver and verify that the forcer and the payload can be moved manually away from the point of the jam.
- Indexer parameters are incorrectly set up. If certain parameters are out of range or are missing, nothing will happen when you issue the Go (G) command.

Use R, RA, RB, and RS status commands to determine what is preventing the move. Also check A, V, and D commands to make sure that all the parameters are set properly. The following are additional troubleshooting techniques:

- Check the forcer for damage. Also check the forcer cable to see if it is damaged or shorted. These conditions may cause the drive to fault.

- Ohm the motor and cables to make sure that short-circuits do not exist between phases or to earth GND. On your most sensitive scale, the resistance across each motor phase should be consistently low (but not zero) and similar to each other. On your highest scale, the resistance between motor phases and between each phase and earth ground should be infinite.

Forcer Stalls

A forcer stall during acceleration may be caused by one or more of the following factors:

- The force requirements may be excessive.
- The acceleration ramp may be too steep.
- The payload torque specifications may be exceeded.
- The platen surface may be uneven or damaged.
- The forcer air orifices may be clogged.

If the forcer stalls during the constant velocity portion of a move, the platen may be damaged or is uneven. It is also possible that dirt on either the forcer or platen surface is obstructing the air gap. You should inspect the system for flatness and damage, then clean the system thoroughly.

A stall may also occur if the motor current DIP switch setting is incorrect. The forcer may not be receiving enough current to operate. Refer to Chapter 6 for the proper current selection.

Fault LED

There is a red FAULT LED located on the LX front panel. The LED may be activated (illuminated) if one of the following conditions exist:

- The drive is overheating. You may consider cooling the cabinet to the temperature specified in the Hardware Reference section. The fan kit will solve the problem.
- A short circuit exists in the forcer current output. Use the Ohm meter to make sure that there is not a short circuit between phase A, phase B, or to earth ground. **NOTE: Make sure power is removed before you test the forcer.**
- You commanded the drive to shutdown. Use the **STØ** command to turn the drive on again.
- A brownout condition exists. Check the AC input voltage to verify that the drive is receiving more than 95VAC.

If the fault condition is caused by short circuit or brownout, the LED remains on until you cycle power to the LX. If the fault is caused by overheating, the LED will turn off when the drive cools to an acceptable operating temperature.

Forcer Fails to Run at High Speeds

If the forcer fails to run at high speeds, it is possible that the forcer may not produce enough force to move a given load at the specified velocities. Check the force/speed curve in Chapter 6 to make sure you are trying to operate the forcer in the proper range

Forcer is Jerky or Weak

Check that there are no mechanical problems at the load causing a highly variable loading conditions. Disconnect the forcer from the load and run it without a load connected. To determine if the forcer is developing its full holding force, use a fish scale to pull in the direction of travel until the stationary forcer slips. If these problems persist, service repair is necessary.

Forcer Overheats

If the forcer exceeds its maximum forcer case temperature rating, failure will eventually result. Check your DIP Switch setting to ensure that the current setting is correct for the forcer you are using (refer to Chapter 6). If the current setting is correct, and you want to run the forcer at a lower temperature, turn DIP Switch 7 to the OFF position. This will reduce your motor current to 1/2 of the set value that is used when your forcer is moving. This should make the forcer run cooler. Use a fan to increase the airflow around the forcer and the platen.

Forcer Disengages from Platen

The forcer will disengage from the platen if the maximum yaw, pitch, or roll specifications are exceeded. One or more of the following may cause such a situation:

- Overhung or cantilevered loads
- A load that is not sufficiently fixed to the forcer
- A moment-producing force being applied to the forcer

Refer to Chapter 6 for the maximum pitch and roll specifications (maximum yaw specifications are applicable only to the L20 forcer).

Platen Develops Signs of Wear

Small amounts of pitting is a normal occurrence for mechanical bearing surfaces and is not an indication of platen failure. The platen's functional life span is limited and its wear is governed by standard mechanical bearing considerations. Load, speed, duty cycle, temperature, cleaning, and abrasion all affect the platen's life span.

I/O Switch failure

If you are having problems using the Trigger (TR), Home (GH) commands, and the Trigger, Home, CW, CCW and Sequence Select inputs, you must first check your wiring for proper installation. Use an Ohm meter to verify proper connection of the switches and inputs. If the hardware connection seems correct, you can manually change the input switches and use the IS command to verify if the LX recognizes the input change. The IS command provides a hardware status of the LX inputs. If the status does not change, check the hardware settings and wiring.

Remote Sequencing (BCD Inputs) failure

If you are having problems trying to run sequences from BCD interfaces, the first thing you must verify is the hardware interface. Use the Ohm meter to verify proper wiring. Then use the IS command to read the status of the inputs. Change the input setting and check the Input Status (IS) again to make sure that the LX recognized the change in the sequence select input. Make sure that your BCD input is calling the proper sequences. Refer to Chapter 4 for the Sequence Select Table. If you have a problem running a sequence from the remote input, try running the sequence using the XR command before attempting to run it using BCD input.

Reducing Electrical Noise

For information on identifying and suppressing electrical noise, refer to the Technical Data section of the *Compumotor Programmable Motion Control Catalog.*

Returning The System

If you must return your LX Indexer/Drive system to effect repairs or upgrades, use the following steps:

STEP 1 Get the serial number and the model number of the defective unit, and a purchase order number to cover repair costs in the event the unit is determined by Parker Compumotor to be out of warranty.

STEP 2 Before you ship the drive to Parker Compumotor, have someone from your organization with a technical understanding of the LX Indexer/Drive system and its application include answers to the following questions:

- What is the extent of the failure/reason for return?
- How long did it operate?
- How many units are still working?
- How many units failed?
- What was happening when the unit failed (i.e., installing the unit, cycling power, starting other equipment, etc)?
- How was the product configured (in detail)?
- What, if any, cables were modified and how?
- With what equipment is the unit interfaced?
- What was the application?
- What was the system sizing (speed, acceleration, duty cycle, inertia torque, friction, etc.)?
- What was the system environment (temperature, enclosure, spacing, unit orientation, contaminants, etc.)?
- What upgrades, if any, are required (hardware, software, user guide)?

STEP 3 Call Parker Compumotor for a Return Material Authorization (RMA) number. Returned products cannot be accepted without an RMA number. The phone number for Parker Compumotor Applications Department is (800) 358-9070.

STEP 4 Ship the unit to:

Parker Compumotor Corporation
5500 Business Park Drive
Rohnert Park, CA 94928
Attn: RMA # xxxxxxx

APPENDICES

Command Listing

A (Acceleration)
B (Buffer Status)
BP (Break Point)
BPA (Break Point Automatic)
BPC (Break Point Custom)
BPL (Break Point Linear)
BS (Buffer Size Status)
C (Continue)
CR (Carriage Return)
D (Distance)
DR (Default Report)
E (Enable Communications)
F (Disable Communications)
G (Go)
GH (Go Home)
GHF (Go Home Velocity)
^H (Backspace)
H (Set Direction)
I (Precalculate Acceleration Table)
IM (Input Mode)
IO (Immediate Output)
IS (Input Status)
K (Kill)
L (Loop)
LA (Limit Acceleration)
LD (Limit Disable)
LF (Line Feed)
MF (Move Form)
MN (Mode Normal)
MPA (Mode Position Absolute)
MPI (Mode Position Incremental)
MT (Move Time)
N (End of Loop)
O (Output)
OFF (Off)
OM (Output Mode)
ON (On)
OR (Report Homing Function Setups)
OSC (Define Active State of Home)
OSH (Reference Edge of Home)
OSI (Linear Gantry System Homing)
PR (Absolute Position Report)
PS (Pause)
PZ (Set Absolute Counter to Zero)
" (Quote)
R (Request Indexer Status)
RA (Limit Switch Status Report)
RB (Loop, Pause, Shutdown, Trigger Status)
RG (Go Home Status)
RS (Status of Sequence Execution)
RV (Revision Level)
S (Stop)
SKE (Skip If Equal)
SKN (Skip If Not Equal)
SN (Scan)
SS (Software Switch Function Status)
SSA (RS-232C Echo Control)
SSB (Clear Pause on Trigger #1)
SSD (Stop on Trigger #1)
SSF (Sequence Strobe Trigger #1)
SSG (Clear/Save the Command Buffer On Limit)
SSH (Clear/Save the Command Buffer on Stop)
SSI (Interactive Mode)
SSJ (Continuous Sequence Scan Mode)
ST (Shutdown)
SV (Save)
T (Time Delay)
TR (Wait for Trigger)
TS (Trigger Input Status)
U (Pause and Wait for Continue)
V (Velocity)
W1 (Signed Binary Position Report)
W3 (Hexadecimal Position Report)
XBS (Sequence Buffer Size)
XC (Sequence Checksum)
XD (Sequence Definition)
XE (Sequence Erase)
XQ (Sequence Interrupt Run Mode)
XR (Run a Sequence)
XRP (Sequence Run with Pause)
XSD (Sequence Status)
XSR (Sequence Status Run)
XSS (Sequence Status)
XT (Sequence Termination)
XU (Upload Sequence)
Y (Stop Loop)
Z (Reset)

ASCII Table

DEC	HEX	GRAPHIC	DEC	HEX	GRAPHIC	DEC	HEX	GRAPHIC
ØØØ	ØØ	NUL	Ø57	39	9	114	72	r
ØØ1	Ø1	SOH	Ø58	3A	:	115	73	s
ØØ2	Ø2	STX	Ø59	3B	;	116	74	t
ØØ3	Ø3	ETX	Ø6Ø	3C	<	117	75	u
ØØ4	Ø4	EOT	Ø61	3D	=	118	76	v
ØØ5	Ø5	ENQ	Ø62	3E	>	119	77	w
ØØ6	Ø6	ACK	Ø63	3F	?	12Ø	78	x
ØØ7	Ø7	BEL	Ø64	4Ø	@	121	79	y
ØØ8	Ø8	BS	Ø65	41	A	122	7A	z
ØØ9	Ø9	HT	Ø66	42	B	123	7B	{
Ø1Ø	ØA	LF	Ø67	43	C	124	7C	|
Ø11	ØB	VT	Ø68	44	D	125	7D	}
Ø12	ØC	FF	Ø69	45	E	126	7E	~
Ø13	ØD	CR	Ø7Ø	46	F	127	7F	DEL
Ø14	ØE	SO	Ø71	47	G			
Ø15	ØF	S1	Ø72	48	H			
Ø16	1Ø	DLE	Ø73	49	I			
Ø17	11	DC1	Ø74	4A	J			
Ø18	12	DC2	Ø75	4B	K			
Ø19	13	DC3	Ø76	4C	L			
Ø2Ø	14	DC4	Ø77	4D	M			
Ø21	15	NAK	Ø78	4E	N			
Ø22	16	SYN	Ø79	4F	O			
Ø23	17	ETB	Ø8Ø	5Ø	P			
Ø24	18	CAN	Ø81	51	Q			
Ø25	19	EM	Ø82	52	R			
Ø26	1A	SUB	Ø83	53	S			
Ø27	1B	ESC	Ø84	54	T			
Ø28	1C	FS	Ø85	55	U			
Ø29	1D	GS	Ø86	56	V			
Ø3Ø	1E	RS	Ø87	57	W			
Ø31	1F	US	Ø88	58	X			
Ø32	2Ø	SPACE	Ø89	59	Y			
Ø33	21	!	Ø9Ø	5A	Z			
Ø34	22	"	Ø91	5B	[			
Ø35	23	#	Ø92	5C	\			
Ø36	24	$	Ø93	5D	]			
Ø37	25	%	Ø94	5E	^			
Ø38	26	&	Ø95	5F	_			
Ø39	27	'	Ø96	6Ø	`			
Ø4Ø	28	(	Ø97	61	a			
Ø41	29	)	Ø98	62	b			
Ø42	2A	*	Ø99	63	c			
Ø43	2B	+	1ØØ	64	d			
Ø44	2C	,	1Ø1	65	e			
Ø45	2D	-	1Ø2	66	f			
Ø46	2E	.	1Ø3	67	g			
Ø47	2F	/	1Ø4	68	h			
Ø48	3Ø	Ø	1Ø5	69	i			
Ø49	31	1	1Ø6	6A	j			
Ø5Ø	32	2	1Ø7	6B	k			
Ø51	33	3	1Ø8	6C	l			
Ø52	34	4	1Ø9	6D	m			
Ø53	35	5	11Ø	6E	n			
Ø54	36	6	111	6F	o			
Ø55	37	7	112	7Ø	p			
Ø56	38	8	113	71	q			

Warranty

The items described in this document are hereby offered for sale at prices to be established by Parker Hannifin Corporation, its subsidiaries, and its authorized distributors. This offer and its acceptance by any customer ('Buyer') shall be governed by all of the following Terms and Conditions. Buyer's order for any item described in its document, when communicated to Parker Hannifin Corporation, its subsidiary, or an authorized distributor ('Seller') verbally or in writing, shall constitute acceptance of this offer.

1. Terms and Conditions of Sale
All descriptions, quotations, proposals, offers, acknowledgements, acceptances, and sales of Seller's products are subject to and shall be governed exclusively by the terms and conditions stated herein. Buyer's acceptance of any offer to sell is limited to these terms and conditions. Any terms or conditions in addition to, or inconsistent with those stated herein, proposed by Buyer in any acceptance of any offer by Seller, are hereby objected to. No such additional, different or inconsistent terms and conditions shall become part of the contract between Buyer and Seller unless expressly accepted in writing by Seller. Seller's acceptance of any offer to purchase by Buyer is expressly conditional upon Buyer's assent to all the terms and conditions stated herein, including any terms in addition to, or inconsistent with those contained in Buyer's offer. Acceptance of Seller's products shall in all events constitute such assent.

2. Payment
Payment shall be made by Buyer net 30 days from the date of delivery of the items purchased hereunder. Amounts not timely paid shall bear interest at the rate of 1-1/2% for each month or a portion thereof that Buyer is late in making payment. Any claims by Buyer for omissions or shortages in a shipment shall be waived unless Seller receives notice thereof within 30 days after Buyer's receipt of the shipment.

3. Delivery
Unless otherwise provided on the face hereof, delivery shall be made F.O.B.Seller's plant. Regardless of the method of delivery, however, risk of loss shall pass to Buyer upon Seller's delivery to a carrier. Any delivery dates shown are approximate only and Seller shall have no liability for any delays in delivery.

4. Warranty
Seller warrants that the items sold hereunder shall be free from defects in material or workmanship for a period of 365 days from the date of shipment to Buyer. THIS WARRANTY COMPRISES THE SOLE AND ENTIRE WARRANTY PERTAINING TO ITEMS PROVIDED HEREUNDER. SELLER MAKES NO OTHER WARRANTY, GUARANTEE, OR REPRESENTATION OF ANY KIND WHATSOEVER. ALL OTHER WARRANTIES, INCLUDING BUT NOT LIMITED TO, MERCHANTABILITY AND FITNESS FOR PURPOSE, WHETHER EXPRESS, IMPLIED, OR ARISING BY OPERATION OF LAW, TRADE USAGE, OR COURSE OF DEALING ARE HEREBY DISCLAIMED. NOTWITHSTANDING THE FOREGOING, THERE ARE NO WARRANTIES WHATSOEVER ON ITEMS BUILT OR ACQUIRED WHOLLY OR PARTIALLY, TO BUYER'S DESIGNS OR SPECIFICATIONS.

5. Limitation of Remedy
SELLER'S LIABILITY ARISING FROM OR IN ANY WAY CONNECTED WITH THE ITEMS SOLD OR THIS CONTRACT SHALL BE LIMITED EXCLUSIVELY TO REPAIR OR REPLACEMENT OF THE ITEMS SOLD OR REFUND OF THE PURCHASE PRICE PAID BY BUYER, AT SELLER'S SOLE OPTION. IN NO EVENT SHALL SELLER BE LIABLE FOR ANY INCIDENTAL, CONSEQUENTIAL, OR SPECIAL DAMAGES OF ANY KIND OR NATURE WHATSOEVER, INCLUDING BUT NOT LIMITED TO LOST PROFITS ARISING FROM OR IN ANY WAY CONNECTED WITH THIS AGREEMENT OR ITEMS SOLD HEREUNDER, WHETHER ALLEGED TO ARISE FROM BREACH OF CONTRACT, EXPRESS OR IMPLIED WARRANTY, OR IN TORT, INCLUDING WITHOUT LIMITATION, NEGLIGENCE, FAILURE TO WARN, OR STRICT LIABILITY.

6. Changes, Reschedules and Cancellations
Buyer may request to modify the designs or specifications from the items sold hereunder as well as the quantities and delivery dates thereof, or may request to cancel all or part of this order, however, no such requested modification or cancellation shall become part of the contract between Buyer and Seller unless accepted by Seller in a written amendment to this Agreement. Acceptance of any such requested modification or cancellation shall be at Seller's discretion, and shall be upon such items and conditions as Seller may require.

7. Special Tooling
A tooling charge may be imposed for any special tooling, including without limitation, dies, fixtures, molds and patterns, acquired to manufacture items sold pursuant to this contract. Such special tooling shall be and remain Seller's property notwithstanding payment of any charges therefore by Buyer. In no event will Buyer acquire any interest in apparatus belonging to Seller which is utilized in the manufacture of the items sold hereunder, even if such apparatus has been specially converted or adapted for such manufacture and notwithstanding any charges paid by Buyer therefor. Unless otherwise agreed, Seller shall have the right to alter, discard or otherwise dispose of any special

tooling or other property in its sole discretion at any time.

8. Buyers Property
Any designs, tools, patterns, materials, drawings, confidential information, or equipment furnished by Buyer or any other items which become Buyer's property, may be considered obsolete and may be destroyed by Seller after two (2) consecutive years have elapsed without Buyer placing an order for the items which are manufactured using such property. Seller shall not be responsible for any loss or damage to such property while it is in Seller's possession or control.

9. Taxes
Unless otherwise indicated on the face hereof, all prices and changes are exclusive of excise, sales, use, property, occupational or like taxes which may be imposed by any taxing authority upon the manufacture, sale, or delivery of the terms sold hereunder. If any such taxes must be paid by Seller or if Seller is liable for the collection of such tax, the amount thereof shall be in addition to the amount for the items sold. Buyer agrees to pay all such taxes or to reimburse Seller therefore upon receipt of its invoice. If Buyer claims exemption from any sales, use, or other tax imposed by any taxing authority, Buyer shall save Seller harmless from and against any such tax, together with any interest or penalties thereon which may be assessed if the items are held to be taxable.

10. Indemnity for Infringement of Intellectual Property Rights
Seller shall have no liability for infringement of any patents, trademarks, copyrights, trade dress, trade secrets, or similar rights except as provided in the Part 10. Seller will defend and indemnify Buyer against allegations of infringement of US patents, US trademarks, copyrights, trade dress, and trade secrets (hereinafter 'Intellectual Property Rights'). Seller will defend at its expense and will pay the cost of any settlement or damages awarded in any action brought against Buyer based on an allegation that an item sold pursuant to this contract infringes the Intellectual Property Rights of a third party. Seller's obligation to defend and indemnify Buyer is contingent on Buyer notifying Seller within ten (10) days after Buyer becomes aware of such allegations of infringement, and Seller having sole control over the defense of any allegations or actions including all negotiations for settlement or compromise. If an item sold hereunder is subject to a claim that it infringes the Intellectual Property Rights of a third party, Seller may, at his sole expense and option, procure for Buyer the right to continue using said item, replace or modify said item so as to make it non-infringing, or offer to accept return of said item and return the purchase price less a reasonable allowance for depreciation. Notwithstanding the foregoing, Seller shall have no liability for claims of infringement based on information provided by Buyer, or directed to items delivered hereunder for which the designs are specified in whole or part by Buyer, or infringements resulting from the modification, combination, or use in a system of any item sold hereunder. The foregoing provisions of this Part 10 shall constitute Seller's sole and exclusive liability and Buyer's sole and exclusive remedy for infringement of Intellectual Property Rights.

If a claim is based on information provided by Buyer or if the design for an item delivered hereunder is specified in whole or in part by Buyer, Buyer shall defend and indemnify Seller for all costs, expenses, or judgments resulting from any claim that such item infringes any patent, trademark, copyright, trade dress, trade secret, or any similar right.

11. Force Majeure
Seller does not assume the risk of and shall not be liable for delay or failure to perform any of Seller's obligations by reason of circumstances beyond the reasonable control of Seller (hereinafter 'Events of Force Majeure'). Events of Force Majeure shall include without limitation, accidents, acts of God, strikes or labor disputes, acts, laws, rules or regulations of any government or government agency, fires, floods, delays or failures in delivery of carriers or suppliers, shortages of materials, and any other cause beyond Seller's control.

12. Entire Agreement /Governing Law
The terms and conditions set forth herein, together with any amendments, modifications, and any different terms or conditions expressly accepted by Seller in writing, shall constitute the entire Agreement concerning the items sold, and there are no oral or other representations or agreements which pertain thereto. This Agreement shall be governed in all respects by the law of the State of Ohio. No actions arising out of the sale of the items sold hereunder of this Agreement may be brought by either party more than two (2) years after the cause of action accrues.

Glossary

Absolute Positioning
Refers to a motion control system employing position feedback devices (absolute encoders) to maintain a given mechanical location.

Absolute Programming
A positioning coordinate reference wherein all positions are specified relative to some reference, or *home* position. This is different from incremental programming, where distances are specified relative to the current position.

Acceleration
The change in velocity as a function of time. Acceleration usually refers to increasing velocity and deceleration describes decreasing velocity.

Accelerometer
A small sensor that measures forcer acceleration and provides an electrical signal proportional to acceleration.

Accuracy
A measure of the difference between expected position and actual position of a forcer or mechanical system. Forcer accuracy is usually specified as a distance representing the maximum deviation from expected position.

Address
Multiple devices, each with a separate address or unit number, can be controlled on the same bus. The address allows the host to communicate individually to each device.

Ambient Temperature
The temperature of the cooling medium, usually air, surrounding the forcer or another device.

ASCII
American Standard Code for Information Interchange. This code assigns a number to each numeral and letter of the alphabet. In this manner, information can be transmitted between machines as a series of binary numbers.

Bandwidth
The frequency range in which the size of the system gain expressed in dB is greater than -3 dB.

Baud Rate
The number of bits transmitted per second. Typical rates include 300, 600,1200, 2400, 4800, 9600, 19,200. This means at 9600 baud, one character can be sent nearly every millisecond.

BCD
Binary Coded Decimal is an encoding technique used to describe the numbers 0 - 9 with four digital (on or off) signal lines. Popular in machine tool equipment, BCD interfaces are now giving way to interfaces requiring fewer wires—such as RS-232C.

Bipolar
The Drive current is bi-directional through each motor phase. There are two motor phases: *Phase A* (A+/A-) & *Phase B* (B+/B-).

Bit
Abbreviation of Binary Digit, the smallest unit of memory equals 1 or 0.

Block Diagram
A simplified schematic representing components and signal flow through a system.

Bode Plot
A graph of system gain and phase versus input frequency which graphically illustrates the steady state characteristics of the system.

Break Frequency
Frequency(ies) at which the gain changes slope on a Bode plot. (Break frequencies correspond to the poles and zeroes of the system.)

Brownout
Low-line voltage at which the device no longer functions properly.

Byte
A group of 8 bits treated as a whole, with 256 possible combinations of ones and zeros, each combination representing a unique piece of information.

Closed Loop
A broadly applied term relating to any system where the output is measured and compared to the input. The output is then adjusted to reach the desired condition. In motion control, the term is used to describe a system wherein a velocity or position (or both) transducer is used to generate correction signals by comparison to desired parameters.

Critical Damping
A system is critically damped when the response to a step change in desired velocity or position is achieved in the minimum possible time with little or no overshoot.

Crossover Frequency
The frequency at which the gain intercepts the 0 dB point on a Bode Plot. (Used in reference to the open-loop gain plot.)

Cyclic Error
The difference between the commanded and actual position over a distance of one tooth pitch. These position errors are repeatable. This error occurs due to motor magnetics. This error recurs once every pole pitch as measured on the body of the forcer.

Daisy-Chain
A term used to describe the linking of several RS-232C devices in sequence such that a single data stream flows through one device and on to the next. Daisy-chained devices usually are distinguished by device addresses, which serve to indicate the desired destination for data in the stream.

Damping
An indication of the rate of decay of a signal to its steady state value. Related to settling time.

Damping Ratio
Ratio of actual damping to critical damping. Less than one is an underdamped system and greater than one is an overdamped system.

Data Bits
Since the ASCII character set consists of 128 characters, computers may transmit only seven bits of data. However, most computers support an eight bit extended ASCII character set.

DCE
Data Communications Equipment transmits on pin three and receives on pin two.

Dead Band
A range of input signals for which there is no system response.

Decibel
A logarithmic measurement of gain. If G is a system gain (ratio of output to input), then 20 log G equals gain in decibels (dB).

Delimiter
A character (space or carriage return) used to separate fields in a command.

Detent Force
The minimal force present in an unenergized forcer. The detent force of a Compumotor or step motor is typically about one percent of its static energized force.

Drive
This is the electronics portion of the system that controls power to the forcer. This portion controls the forcer to provide micro-stepping.

DTE
Data Transmission Equipment transmits on pin two and receives on pin three.

Duty Cycle
For a repetitive cycle, the ratio of on time to total cycle time.
Duty Cycle = *On Time* ÷ *On Time + Off Time*

Efficiency
The ratio of power output to power input.

Encoder
A device which translates mechanical motion into electronic signals used for monitoring position or velocity.

Following
The ability to make one axis perform motion based on the motion of a second, or *master*, axis.

Friction
A resistance to motion caused by surfaces rubbing together. Friction can be constant with varying speed (Coulomb friction) or proportional to speed (viscous friction).

Full Duplex
The terminal will display only received or echoed characters.

Gain
The ratio of system output signal to system input signal.

Half Duplex
In half duplex mode, a terminal will display every character transmitted. It may also display the received character.

Hand Shaking Signals
RTS: Request To Send
CTS: Clear To Send
DSR: Data Set Ready
DTR: Data Terminal Ready
IDB: Input Data Buffer
ODB: Output Data Buffer

Holding Force
Sometimes called static force, it specifies the maximum external torque that can be applied to a stopped, energized forcer without causing it to move.

Home
A reference position in a motion control system, usually derived from a mechanical datum. Often designated as the *zero* position.

Hysteresis
The difference in response of a system to an increasing or a decreasing input signal.

IEEE-488
A digital data communications standard popular in instrumentation electronics. This parallel interface is also known as GPIB, or General Purpose Interface Bus.

Incremental Motion
A motion control term that is used to describe a device that produces one step of motion for each step command (usually a pulse) received.

Incremental Programming
A coordinated system where position or distances are specified relative to the current position.

Indexer
This portion of the system provides communication with the external I/O. It allows you to program sequences and direct motion control.

Inertia
A measure of an object's resistance to a change in velocity. The larger an object's inertia, the larger the torque that is required to accelerate or decelerate it. Inertia is a function of an object's mass and its shape.

Limits
Properly designed motion control systems have sensors called limits that alert the control electronics that the physical end of travel is being approached and that motion should stop.

Logic Ground
An electrical potential to which all control signals in a particular system are referenced.

Microstepping
An electronic control technique that proportions the current in a step motor's windings to provide additional intermediate positions between poles. Produces smooth movement over a wide speed range and high positional resolution.

Null Modem
A simple device or set of connectors which switches the receive and transmit lines of a three wire RS-232C connector.

Open Collector
A signal output that is performed with a transistor. An open collector output acts like a switch closure with one end of the switch at ground potential and the other end of the switch accessible.

Open Loop
A motion control system where no external sensors are used to provide position or velocity correction signals.

Opto-Isolated
A method of sending a signal from one unit to another without the usual requirement of common ground potentials. The signal is transmitted optically with a light source (usually a Light Emitting Diode) and a light sensor (usually a photosensitive transistor). These optical components provide electrical isolation.

Parallel
A data communication format wherein many signal lines communicate several pieces of data simultaneously.

Parity
An RS-232C error detection scheme that detects an odd number of transmission errors.

PCA
Printed circuit (board) assembly.

PLC
Programmable logic controller. An industrial control device that turns on and off outputs based upon responses to inputs.

Pole
A frequency at which a system's transfer function goes to infinity.

Primary
With respect to following, this refers to the *master* axis motion being followed by another axis.

Pulse Rate
The frequency of the step pulses applied to a motor driver. The pulse rate multiplied by the resolution of the motor/drive combination (in steps per revolution) yields the speed in inches per second.

Quadrature
A type of incremental encoder output in which the two square wave outputs are offset by 90°.

Ramping
The acceleration and deceleration of a motor. May also refer to the change in frequency of the applied step pulse train.

Rated Force
The force-producing capacity of a forcer at a given speed. This is the maximum force the forcer can deliver to a load and is usually specified with a force/speed curve.

Registration
The ability to execute a preset move with reference to an external event while the forcer is executing another move.

Relative Accuracy
Also referred to as *Step-to-Step Accuracy*, this specification tells how microsteps can change in size. In a perfect system, microsteps would all be exactly the same size, but drive characteristics and the absolute accuracy of the motor cause the steps to expand and contract by an amount up to the relative accuracy figure. The error is not cumulative.

Repeatability
The degree to which the positioning accuracy for a given move performed repetitively can be duplicated.

Resolution
The smallest positioning increment that can be achieved.

Ringing
Oscillation of a system following a sudden change in state.

RS-232C
A data communications standard that encodes a string of information on a single line in a time sequential format. The standard specifies the proper voltage and timing requirements so that different manufacturers' devices are compatible.

Secondary
With respect to following, this refers to the *slave* axis motion being controlled by the primary axis.

Sequence
A series of motion control commands. These commands are created, stored, and executed from the indexer's non-volatile memory such as EEPROM or battery-backed RAM..

Short-Circuit
A defect in a winding which causes part of the normal electrical circuit to be bypassed. This frequently results in reducing the resistance or impedance to such an extent (near zero) as to cause overheating of the circuit, and subsequent burnout.

Slew
In motion control, the portion of a move made at a constant non-zero velocity.

Speed
Used to describe the linear velocity of a forcer or other object in motion.

Start Bits
RS-232C character transmissions begin with a bit which signals the receiver that data is now being transmitted.

Static Force
The maximum force available at zero speed.

Stiffness
The ability to resist movement induced by an applied force. Is often specified as a force displacement curve, indicating the amount a forcer will move upon application of a known external force when stopped.

Stop Bits
When using RS-232C, one or two bits are added to every character to signal the end of a character.

Text/Echo (Off/On)
This setup allows received characters to be re-transmitted to the original sending device. Echoing characters can verify or *close the loop* on a transmission.

Transfer Function
A mathematical means of expressing a system's output to input relationship.

TTL
Transistor-Transistor Logic. Describes a common digital logic device family that is used in most modern digital electronics. TTL signals have two distinct states that are described with voltages: a logical **Ø** or *low* is represented by a voltage of less than 0.8V and a logical **1** or *high* is represented by a voltage from 2.5 to 5V.

XON/XOFF
Two ASCII characters supported in some serial communication programs. If supported, the receiving device transmits an XOFF character to the host when its character buffer is full. The XOFF character directs the host to stop transmitting characters to the device. Once the buffer empties the device will transmit an XON character to signal the host to resume transmission.

Zero
A frequency at which the transfer function of a system goes to zero.

Index