The Only Factory Authorized Repair Center for Superior Electric Stepper Drive & Motor Repair

Motor Systems Inc.
460 Milford Parkway
Milford, OH . 45150
www.motorsystems.com
513-576-1725
INSTRUCTIONS
for
SLO-SYN® MICRO SERIES
MOTION CONTROLS
INDEXER MODELS
230-PI and 430-PI

WARNER ELECTRIC
Superior Electric
© 1989, The Superior Electric Company
EXPRESS START-UP PROCEDURE

I. POWER SUPPLY

CAUTION: Never connect or disconnect anything from the unit with the power on.

1. Be sure the ac power line is within specified limits and is connected correctly (see Section 3.5.1).
2. If the unit is an "X" model, be sure the proper tap is selected on the transformer, depending on the ac mains voltage.
3. Be sure the unit is properly grounded ("G" terminal on the ac power strip).

II. DRIVE

CAUTION: Drive is not short circuit protected. Do not short motor outputs or the drive will be damaged.

1. Make sure the motor to be used is compatible with the drive. Refer to Section 3.3 for a list of compatible motors.
2. Use the motor connection diagrams shown in Figure 2.2 of the manual. When using a 6-lead motor, be sure to insulate and isolate the unused wires. Be sure to insulate all motor leads to prevent inadvertent shorts to ground or to each other.
3. For Reduced Current operation, see Section 3.8. Install a resistor of the appropriate value between the REDUCE CURRENT pin and the LOGIC COMMON pin. Refer to the speed/torque data and the resistor versus current table included in the manual.
4. To connect to the logic controls, refer to Sections 3.6 and 5 of the manual for the connections.

CAUTION: The motor may operate erratically at speeds below 350 steps per second due to motor resonance. Avoid this speed range if a problem exists.

III. INDEXER

This Instruction Manual MUST BE READ IN ITS ENTIRETY to correctly operate the Indexer. This Express Start-Up Section only highlights the important steps needed to ensure correct operation.

A. SERIAL (RS232) OPERATION

1. SPECIFICATIONS - Be certain that the following signal specifications are met:
   - RS232 signal characteristics:
     - Output Voltage Swing: ±5 Vdc minimum, ±10 Vdc maximum
     - Input Voltage Range: -30 Vdc minimum, +30 Vdc maximum

2. CONNECTIONS - The serial port (9-pin "D" connector) is designated as follows:

<table>
<thead>
<tr>
<th>Pin</th>
<th>Assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Vo (Signal Common)</td>
</tr>
<tr>
<td>2</td>
<td>RS232 Chain Out</td>
</tr>
<tr>
<td>3</td>
<td>RS232 Receive Data</td>
</tr>
<tr>
<td>4</td>
<td>Vo (Signal Common)</td>
</tr>
<tr>
<td>5</td>
<td>Vo (Signal Common)</td>
</tr>
<tr>
<td>6</td>
<td>RS232 Echo</td>
</tr>
<tr>
<td>7</td>
<td>RS232 Chain In</td>
</tr>
<tr>
<td>8</td>
<td>+5 Vdc (SSP-500 only)</td>
</tr>
<tr>
<td>9</td>
<td>+5 Vdc (SSP-500 only)</td>
</tr>
</tbody>
</table>

Connect the host computer or terminal as shown in Figure 2.4 of the manual for single indexer interfacing or as in Figure 2.5 for daisy-chain operation. Use caution when connecting the indexer to the host device as +5 Vdc is present on the connector. Only use the +5 Vdc for operation with the SSP-500 and the SSP-525 hand-held pendants; this connection is already made in the cable.

3. COMMUNICATIONS - Configure the RS232 communications parameters to correspond to the protocol of the host device. The indexer utilizes the Xon/Xoff handshaking technique, which should be followed to ensure proper serial communications. The indexer must be addressed with the proper attention character (e.g., "<") and the device identification number (e.g., indexer #1) that is contained in parameter L21 to initiate communications.

4. PROGRAMMING - Always program L70, the step resolution parameter, first, followed by the remainder of the parameters. Carefully read Section 4 to implement the powerful and varied instruction set. Factory defaults have been set for your unit to aid in first-time operation. These are listed in Section 4.3 of the manual.

5. EXECUTION - The indexer executes its program based on the present settings of the various modes. If program or manual operations are not correct, verify the mode and parameter settings. The Trace mode is a valuable aid in observing program operation.

If the motor operates erratically, the motion parameters may need adjusting.
B. PARALLEL OPERATION

1. SPECIFICATIONS - Be certain that your signals which communicate with the indexer meet the following specifications. (More details are found in Section 3.6 of this manual).

**Input Characteristics:**
- high level (inactive) voltage: +8.5 Vdc minimum, +15.0 Vdc maximum
- high level current: 1 milliampere maximum
- low level (active) voltage: +0.0 Vdc minimum, +6.5 Vdc maximum
- low level current: 3.5 milliamperes maximum

**Output Characteristics:**
- high level (inactive) voltage: +24.0 Vdc maximum, open collector
- high level leakage current: 250 microamperes maximum
- leakage current:
  - low level output: +0.4 Vdc at 16 milliamperes sink current
  - +0.7 Vdc at 40 milliamperes sink current

2. CONNECTIONS - Parallel signals are obtained via the 25-pin "D" type connector. The pin assignments are as follows:

<table>
<thead>
<tr>
<th>Pin Assignment</th>
<th>Pin Assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Vo (Signal Common)</td>
<td>14 Vo (Signal Common)</td>
</tr>
<tr>
<td>2 D7 Input-</td>
<td>15 D6 Input-</td>
</tr>
<tr>
<td>3 D5 Input-</td>
<td>16 D4 Input-</td>
</tr>
<tr>
<td>4 D3 Input-</td>
<td>17 D2 Input-</td>
</tr>
<tr>
<td>5 D1 Input-</td>
<td>18 D0 Input-</td>
</tr>
<tr>
<td>6 Motion Busy-</td>
<td>19 Not Used</td>
</tr>
<tr>
<td>7 Strobe 7-</td>
<td>20 Strobe 6-</td>
</tr>
<tr>
<td>8 Strobe 5-</td>
<td>21 Strobe 4-</td>
</tr>
<tr>
<td>9 Strobe 3-</td>
<td>22 Strobe 2-</td>
</tr>
<tr>
<td>10 Strobe 1-</td>
<td>23 Strobe 0-</td>
</tr>
<tr>
<td>11 Output 2-</td>
<td>24 Output 1-</td>
</tr>
<tr>
<td>12 All Windings Off Output-</td>
<td>25 Pulse Output-</td>
</tr>
<tr>
<td>13 Direction Output-</td>
<td></td>
</tr>
</tbody>
</table>

**NOTE:** Minus sign (-) indicates signal is active when low, inactive when high.

3. COMMUNICATIONS - Proper parallel operation must observe the timing requirements as shown in Figure 5.3. The LD7 parameter determines the timing of the strobes during the load sequence.

4. PROGRAMMING - L70, the step resolution parameter, must be programmed first, followed by the rest of the parameters. Carefully read Sections 4 and 5 of the manual to implement the powerful and varied instruction set. Factory default values have been set for your unit to aid first-time operation. These are listed in Section 4.3.

5. EXECUTION - The indexer executes based on the present settings of the various modes. If program or manual operations are not correct, verify the mode and parameter settings.

If the motor operates erratically, adjust the motion parameters.

Fuses and Connectors for 230 and 430 Series packaged drives.
- Fuse: 3 ampere, 250 volt slow-blow
  - Bussman part number MDA-3
  - Littelfuse part number 326003
- Motor Connector (mates with female connector J2 on drive)
  - Male Connector Body: AMP part number 206434-1
  - Pins (5 required): AMP part number 66506-8
  - Cable Clamp: AMP part number 206062-1

### INSTALLATION GUIDELINES FOR REDUCED NOISE INTERFERENCE

#### I General Comments
SLO-SYN Micro Series drives use modern solid-state electronics such as microprocessors to provide the features needed for advanced motion control applications. In some cases, these applications produce electromagnetic interference (EMI, or electrical "noise") that may cause inappropriate operation of the microprocessor logic used in the Micro Series product, or in any other computer-type equipment in the user's system.

This guide is aimed toward helping users avoid such problems at the start by applying "good engineering practices" when designing their systems. Following these guidelines will usually prevent EMI noise from interfering with drive operation.

#### II Noise Sources
What causes electrical noise? In general, any equipment that causes arcs or sparks or that switches voltage or current at high frequencies can cause interference. In addition, ac utility lines are often "polluted" with electrical noise from sources outside a user's control (such as equipment in the factory next door).

The following are some of the more common causes of electrical interference:
- power from the utility ac line
- relays, contactors and solenoids
- light dimmers
- arc welders
- motors and motor starters
- induction heaters
- radio controls or transmitters
- switch-mode power supplies
- computer-based equipment
- high frequency lighting equipment
- dc servo and stepper motors and drives
III Mounting Location

When selecting a mounting location, it is preferable to keep the drive away from obvious noise sources, such as those listed above. If possible, locate the drive in its own metal enclosure to shield it and its wiring from noise sources. If this cannot be done, keep the drive at least three feet from any noise sources.

IV Wiring Practices - “Dos” and “Don’ts”

Do the following when installing or wiring your drive or indexer:

- **Do** keep the drive and its wiring as far away from noise sources as possible.
- **Do** provide a good, solid ground connection to the ac system earth ground conductor. Bond the drive case to the system enclosure.
- **Do** use a single-point grounding scheme for all related components of a system (this looks like a “hub and spokes” arrangement).
- **Do** keep the ground connection short and direct.
- **Do** use a line filter on the ac input (Corcom type 10B1, 10S1 or 10K1 or equivalent) for noisy ac lines. Particularly bad ac lines may need to be conditioned with a ferroresonant type isolation transformer to provide "clean" power to the drive or indexer.
- **Do** keep signal and drive wiring well separated. If the wires must cross, they should do so at right angles to minimize coupling. Power wiring includes ac wiring, motor wiring, etc. and signal wiring includes inputs and outputs (I/O), serial communications (RS232 lines), etc.
- **Do** use separate conduits or ducts for signal and I/O wiring. Keep all power wiring out of these signal line conduits.
- **Do** use shielded, twisted-pair cables for indexer I/O lines.
- **Do** ground shields only at one end, the indexer/drive end.
- **Do** use twisted-pair, shielded cable for the motor wiring.
- **Do** use solid-state relays instead of electromechanical contact types wherever possible to minimize noise generation.
- **Do** suppress all relays to prevent noise generation. Typical suppressors are capacitors or MOV's. See manufacturers literature for complete information.
- **Do** use shielded, twisted-pair cable for connections to the RS232 serial port.

Do not do the following when installing your drive or indexer:

- **Do not** install sensitive computer-based equipment (such as an indexer/drive) near a source of electromagnetic noise.
- **Do not** bundle power and signal lines together.
- **Do not** bundle motor cables and signal lines together.
- **Do not** fail to use shielded, twisted-pair cables for signals.
- **Do not** fail to properly connect the system grounds.
- **Do not** use “daisy-chained” grounds.
- **Do not** fail to ground signal cable shields at only one end.
- **Do not** assume that power from the ac line is adequately "clean".

V Troubleshooting Guide

1. Check the quality of the ac line voltage using an oscilloscope and a line monitor, such as Superior Electric's VMS series. If line voltage problems exist, use appropriate line conditioning, such as line filters or isolation transformers.

2. Be certain all of the previous Dos and Don’ts are followed for location, grounding, wiring and relay suppression.

3. Double check the grounding connections to be sure they are good electrical connections and are as short and direct as possible.

4. Try operating the drive with all suspected noise sources switched off. If the drive functions properly, switch the noise sources on again, one at a time, and try to isolate which ones are causing the interference problems. When a noise source is located, try rerouting wiring, suppressing relays or other measures to eliminate the problem.

5. Don’t get discouraged. Electrical interference problems are common with today’s computer-based controls, and such problems are often difficult to diagnose and cure. Sometimes, finding and curing such problems involves a bit of “black magic”. Keep trying various combinations of the “good wiring practices” outlined in this document. Some of them are bound to work.
TABLE OF CONTENTS

PRECAUTIONS
WARNINGS, CAUTIONS, LIMITS OF USE, etc.

SECTION 1: INTRODUCTION
1.1 Features overview
1.2 Inspection parts list
1.3 Using this manual
1.3.1 Organization
1.3.2 Logic, voltage and programming conventions

SECTION 2: MOUNTING, CONNECTIONS AND PIN ASSIGNMENTS
2.1 Mounting
2.2 Motor connections
2.3 External connectors
2.3.1 J1: Parallel I/O connector
2.3.2 J2: Motor connector
2.3.3 J3: Power input
2.3.4 J4: Serial I/O
2.4 Serial communications connection diagrams
2.4.1 Single indexer system
2.4.2 Multiple indexer system

SECTION 3: SPECIFICATIONS
3.1 Drive description
3.2 Drive performance
3.3 Motor compatibility
3.4 Drive mechanical specifications
3.5 Electrical specifications
3.5.1 Power supply requirements
3.5.2 Output to motor
3.6.1 Parallel I/O
3.6.1.1 Parallel output characteristics
3.6.1.2 Parallel input characteristics
3.6.2 Serial I/O connector
3.7 Environmental requirements
3.8 Reduced current

SECTION 4: PROGRAMMING GUIDE
4.1 Overview and set-up
4.2 Immediate commands
Clear
Feedhold
Device Attention Character
Cycle Stop
Backspace and Delete
Delete Line
4.3 L codes: indexer parameters
L06, Program Execution Format
L07, Strobe Delay Time
L08, Mechanical Home Direction
L09, Jog Speed
L11, Acceleration/Deceleration
L12, Low Speed
L14, Home Speed
L17, Offset Direction and Distance From Mechanical Home
L18, Clockwise Software Travel Limit
L19, Counterclockwise Software Travel Limit
L21, Assign Device Identification Number
L22, Baud Rate
L23, Character Length
L25, Parity
L26, Acknowledge
L41, Auto Start Line Number
L44, Program Line Delay
L45, Limit Switch Enable
4.4 H Codes: Commands for Modes of Operation
H1, Cycle Start
H2, Step Mode
H3, Jog Mode
H4, High Speed Mode
H5, Low Speed Mode
H6, Turn In CW Direction
H7, Turn In CCW Direction
H8, Return To Electrical Home
H9, Set Electrical Home
H10, Return To Mechanical Home
H11, Clear Present Program Line
H12, Clear Program
H13, Transmit Contents Of Present Program Line
H14, Transmit Entire Program
H15, Transmit The Current Program Line Number
H16, Transmit Parameters
H17, Transmit Absolute Position
H18, Transmit Motion Status
H19, Transmit Mode Status
H20, Transmit I/O Status
H23, Transmit Software Revision Date
H24, Enable Trace Mode
H25, Disable Trace Mode
4.5 Programming Codes
N Codes, Line Number
X Codes, Move Direction and Distance
F Codes, Feedrate
G Codes, Programmable Commands
G04, Dwell Time
G11, Call a Subroutine
G20, Conditional Branch
G22, Wait for Input
G30, Return From Subroutine Program End
G31, Program Stop
G36, Strobe X Code Data
G37, Strobe N Code Data
G47, Set Output
G64, Enable Reduced Current
G65, Cancel Reduced Current
G66, Enable Boost Current
G67, Cancel Boost Current
G68, Enable All Windings Off
G69, Cancel All Windings Off
G76, Return to Electrical Home
G77, Set Electrical Home
G78, Return to Mechanical Home
G90, Absolute Mode
G91, Incremental Mode
4.6 Sample Program
4.7 Code Assignment Table

SECTION 5: OPERATING INSTRUCTIONS
5.1 Overview
5.2 Operating from a Switch Panel
5.3 Operating from a Remote Terminal
5.4 Operation from a Host Computer

SECTION 6: SPEED/TORQUE CURVES
6.1 Motor Performance

SECTION 7: TROUBLESHOOTING

SECTION 8: COMPONENT LAYOUT

WARRANTY AND LIMITATION OF LIABILITY
1.1 FEATURES OVERVIEW

The 230-PI and 430-PI are differentiated as follows:

<table>
<thead>
<tr>
<th>MOTOR CURRENT PER PHASE</th>
<th>VA PER PHASE</th>
</tr>
</thead>
<tbody>
<tr>
<td>230-PI</td>
<td>2 Amps. peak</td>
</tr>
<tr>
<td>430-PI</td>
<td>3.5 Amps. peak</td>
</tr>
</tbody>
</table>

The 230-PI and 430-PI motor drive/indexer package is a low cost, energy efficient motor drive module that is coupled to a programmable microcontroller indexer. An integral power supply provides the necessary dc voltages required to operate the indexer and drive. This indexer/driver module is capable of driving a wide range of Superior Electric S10-Syn stepper motors, and allows for a wide range of functions. The driver/indexer has several features, including:

- Programming is done in a simple, easy-to-use format.
- Up to 400 lines of program instructions can be stored.
- Program storage is in nonvolatile memory.
- Easy programming allows motion and input/output control.
- Up to 99 indexers may be daisy-chained via RS232 communications.
- Program entry and execution may be done in a variety of ways:
  - Switch panel, including the Superior Electric SSP-100 Slo-Syn Indexer Programmer.
  - Remote terminal, including the Superior Electric SSP-500 or SSP-525 Slo-Syn Indexer Programmer.
  - Host computer.

1.2 INSPECTION PARTS LIST

The drive module, heat sink and power supply come fully assembled as a single unit and are marked with the part number, either 230-PI or 430-PI.

1.3 USING THIS MANUAL

This manual is an installation and operating guide to the 230-PI and 430-PI motor drive and Micro Series Indexer modules. All the information provided is necessary for using these modules successfully.

We strongly recommend that this manual be read thoroughly and completely before attempting to install and operate the equipment.

1.3.1 Organization

All entries in this manual refer to both the 230-PI and the 430-PI modules, unless otherwise specified.

This manual is organized for the convenience of the operator. Section 2, "Mounting, Connections and Pin Assignments", provides diagrams and reminders that are necessary even for the experienced user and installer.
Complete specifications, listed in Section 3, will provide easily referenced information concerning all aspects of installation, power and interface requirements, as well as performance specifications.

Section 4 is a "Programming Guide" that explains all the parameters and commands used by the indexer, and gives examples of how these commands are used.

Section 5, entitled "Operating Instructions", provides information on how to operate the indexer from a switch panel, remote terminal or host computer. Detailed information on switch and strobe settings will be provided there.

The remaining sections contain additional drawings and information useful for setting and operating the indexer modules.

**1.3.2 LOGIC, VOLTAGE AND PROGRAMMING CONVENTIONS**

- All logic is LOW TRUE. This means that a logic convention is active when low and inactive when high. The low true condition is designated by a minus sign (-). In the case of step/jog-, jog- is active when low.

- If a logic control function pin is left open, the function will be clamped in a high or inactive condition.

- When a sign is to be used in conjunction with a move distance or an offset direction, + will cause clockwise motion as viewed from the motor's LABEL END.

- Certain commands are designated as MODE commands. Examples are: ABSOLUTE MODE, INCREMENTAL MODE, STEP MODE, JOG MODE, etc. Care should be taken to assure that the correct MODE is operation for each command. Once a mode is set, it remains active until a canceling or alternate mode is chosen.

- Motion performance and the ranges listed for motion parameters are dependent on the translator resolution chosen with the L70 parameter. L70 must be programmed prior to any motion parameter entry. If the L70 parameter is modified, the motion parameters must be reentered.

- The red "power on" LED indicator detects the presence of the +5 Vdc voltage output from the internal power supply.

- The unit's ac input is fused. A blown ac input fuse will prevent the power supply from energizing any of its outputs.

**SECTION 2: MOUNTING, CONNECTIONS AND PIN ASSIGNMENTS**

**2.1 MOUNTING**

The 230-PI and 430-PI modules are mounted by affixing the package to a flat surface in one of four possible configurations, using the brackets provided. Figure 2.1 shows the mounting hole locations and diameters.

The heatsink should always be mounted with the fins oriented vertically, or proper cooling will not occur. Air flow through the unit should not be obstructed. Maximum drive heatsink temperature should not exceed +70°C (+158°F).
Recommended Wiring Practices

WHEN MOUNTING INDEXER, BE SURE TO PROVIDE SOLID, CLEAN CONNECTION BETWEEN INDEXER CASE AND ENCLOSURE

- KEEP THESE WIRES SHORT AS POSSIBLE
- GROUND SHIELD ONLY AT INDEXER
- KEEP SIGNAL CABLES WELL AWAY FROM MOTOR WIRING AND AC POWER WIRING

- SHIELDED TWISTED-PAIR CABLES
- RS232 SHIELDED CABLE (FOR BEST RESULTS, GROUND SHIELD ONLY BY INDEXERS)
- AC INPUT

- AC LINE FILTER G
- SINGLE POINT GROUND TO CHASSIS

- MOTOR
- MOTOR CABLE
- 25-PIN "D" TO TERMINAL STRIP CONVERTER *

- TERMINAL #26
- SHIELDED GROUND CONNECTION
- LIMIT SWITCH

- DO NOT GROUND SHIELD AT SWITCH END.

* SUCH AS MAGNUM CONNECTOR #715125-NL TYPE 15(FEMALE) COOPER INDUSTRIES, BUSSMAN CORP.

Don't forget diodes. (See Indexer Manual)
2.2 MOTOR CONNECTIONS

All motor connections are made via the 8-pin circular AMP connector. Figure 2.2 shows the possible motor wiring configurations.

---

Figure 2.2, Motor Connections

*NOT connected to drive or ground. These leads must be insulated and isolated or damage to drive may occur.
It is suggested that a Superior Electric motor cable be used. They are available as follows:

<table>
<thead>
<tr>
<th>CABLE CONFIGURATION</th>
<th>CABLE PART NUMBER</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRIVE END MOTOR END</td>
<td>10 FEET LONG</td>
</tr>
<tr>
<td>Twist-Lock Connector</td>
<td>B215801-001</td>
</tr>
<tr>
<td>Plug*</td>
<td>B216066-001</td>
</tr>
<tr>
<td>Twist-Lock Connector</td>
<td>B215801-002</td>
</tr>
<tr>
<td>Motor Connector</td>
<td>25 FEET LONG</td>
</tr>
<tr>
<td>Plug*</td>
<td>B216066-002</td>
</tr>
<tr>
<td>Twist-Lock Connector</td>
<td>B215801-003</td>
</tr>
<tr>
<td>Plug*</td>
<td>B216066-003</td>
</tr>
</tbody>
</table>

* Mates with receptacle on M061, M062 and M063 motors that have receptacles (M061-CS08, etc.).

2.3 EXTERNAL CONNECTORS

2.3.1 J1: Parallel I/O Connector (25-pin “D” type), see Section 3.6.1 for pin assignments.

2.3.2 J2: Motor Connector, see Section 3.5.2.1 for pin assignments

The 230-PI and 430-PI can be used with 6-lead and 8-lead Slo-Syn motors.

Figure 2.2 shows the correct connections for each possible motor configuration.

2.3.3 J3: Power Input, see Section 3.5.1.2 for pin assignments.

2.3.4 J4: Serial I/O (9-pin “D” type)

2.4 SERIAL COMMUNICATION CONNECTION DIAGRAMS

2.4.1 Single Indexer System

In a single indexer system, it is necessary to activate the index by initially transmitting the device attention character and corresponding device address. A new indexer has a device address of 1. See Figure 2.4 for wiring instructions.

---

Figure 2.3, Connector Locations

Figure 2.4, Single Indexer Connection
2.4.2 Multiple Indexer System

In a multiple indexer system, up to 99 indexers may be daisy-chained together. Daisy-chaining is a method by which multiple indexers can communicate with a single host using only one serial port. In order to daisy-chain multiple indexers, each indexer must first be programmed with a unique device address. The device address can be any number from 1 through 99, as long as each indexer has a unique address. The device addresses need not be consecutive; the indexers can be placed in the chain in any position regardless of their device addresses. The device address of zero is used to communicate with all indexers simultaneously. See Figure 2.5 for wiring instructions.

Figure 2.5, Daisy-Chain Connections
SECTION 3: SPECIFICATIONS

3.1 DRIVE DESCRIPTION
Bipolar, speed adjustable, 2-phase chopper drive with translator and indexer.
Power semiconductor type: H bridge power IC
Translator: internal IC
Control signals are optically isolated from the motor drive module (except Reduce Current).

3.2 DRIVE PERFORMANCE
Step resolution: Half-step or full-step
Step Rate:
 0 to 10,000 full-steps/sec.
 0 to 10,000 half-steps/sec.
Speed/Torque: See Section 6 for typical Speed/Torque curves.

3.3 MOTOR COMPATIBILITY
Motor Families and Size:

<table>
<thead>
<tr>
<th>MOTOR FAMILIES</th>
<th>MOTOR FOR USE WITH 230-PI</th>
<th>MOTOR FOR USE WITH 430-PI</th>
</tr>
</thead>
<tbody>
<tr>
<td>WITH CONNECTOR</td>
<td>M061-CS08 M062-CS09 M063-CS06</td>
<td>M061-CS08 M062-CS09 M063-CS06</td>
</tr>
<tr>
<td>M061-CE08 M062-CE09 M063-CE09</td>
<td>M061-CE08 M062-CE09 M063-CE09</td>
<td></td>
</tr>
</tbody>
</table>

| WITH LEADS | M061-LS08 M062-LS09 M063-LS06 | M061-LS08 M062-LS09 M063-LS06 |
| M061-LE08 M062-LE09 M063-LE09 | M061-LE08 M062-LE09 M063-LE09 |

| WITH CONNECTOR | M061-CS08 M062-CS09 M063-CS06 | M061-CS08 M062-CS09 M063-CS06 |
| M061-CE08 M062-CE09 M063-CE09 | M061-CE08 M062-CE09 M063-CE09 |

| WITH LEADS | M061-LS08 M062-LS09 M063-LS06 | M061-LS08 M062-LS09 M063-LS06 |
| M061-LE08 M062-LE09 M063-LE09 | M061-LE08 M062-LE09 M063-LE09 |

3.4 DRIVE MECHANICAL SPECIFICATIONS
Size: (Inches): 6.5 L x 3.94 W x 7.09 H
  (mm): 165.1 L x 100.01 W x 180.1 H
Weight: 230-PI: 8.5 lbs (3.86 kg)
  430-PI: 9.0 lbs (4.07 kg)

3.5 ELECTRICAL SPECIFICATIONS

3.5.1 INPUT

3.5.1.1 Input Power Requirements

3.5.1.2 AC Input Connections

<table>
<thead>
<tr>
<th>Pin</th>
<th>Assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Hot (black)</td>
</tr>
<tr>
<td>2</td>
<td>Neutral (white)</td>
</tr>
<tr>
<td>3</td>
<td>Ground (green)</td>
</tr>
</tbody>
</table>

3.5.2 OUTPUT TO MOTOR

3.5.2.1 Motor Connections

<table>
<thead>
<tr>
<th>Pin</th>
<th>Assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>M4</td>
</tr>
<tr>
<td>2</td>
<td>M1</td>
</tr>
<tr>
<td>3</td>
<td>no connection</td>
</tr>
<tr>
<td>4</td>
<td>Ground</td>
</tr>
<tr>
<td>5</td>
<td>no connection</td>
</tr>
<tr>
<td>6</td>
<td>M5</td>
</tr>
<tr>
<td>7</td>
<td>no connection</td>
</tr>
<tr>
<td>8</td>
<td>M3</td>
</tr>
</tbody>
</table>

NOTE: Motor phase A is M1 and M3 and motor phase B is M4 and M5.

Method: Mates to male Amp connector part number 206434-1 (Amp pin part number 66506-8 and Amp cable clamp part number 206062-1). Superior Electric cables are recommended (see Section 2.2). Cable with shielded, twisted pairs (one pair for each motor phase) is highly recommended. Six twists per foot is a good guideline.
3.6 MICRO SERIES INDEXER I/O (INTERFACE)

3.6.1 PARALLEL I/O

J1: 25-Pin “D” Type Connector, Female

<table>
<thead>
<tr>
<th>Pin</th>
<th>Assignment</th>
<th>Pin</th>
<th>Assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Vo (Signal Common)</td>
<td>14</td>
<td>Vo (Signal Common)</td>
</tr>
<tr>
<td>2</td>
<td>D7 INPUT-</td>
<td>15</td>
<td>D6 INPUT-</td>
</tr>
<tr>
<td>3</td>
<td>D5 INPUT-</td>
<td>16</td>
<td>D4 INPUT-</td>
</tr>
<tr>
<td>4</td>
<td>D3 INPUT-</td>
<td>17</td>
<td>D2 INPUT-</td>
</tr>
<tr>
<td>5</td>
<td>D1 INPUT-</td>
<td>18</td>
<td>D0 INPUT-</td>
</tr>
<tr>
<td>6</td>
<td>MOTION BUSY-</td>
<td>19</td>
<td>not used</td>
</tr>
<tr>
<td>7</td>
<td>STROBE 7-</td>
<td>20</td>
<td>STROBE 6-</td>
</tr>
<tr>
<td>8</td>
<td>STROBE 5-</td>
<td>21</td>
<td>STROBE 4-</td>
</tr>
<tr>
<td>9</td>
<td>STROBE 3-</td>
<td>22</td>
<td>STROBE 2-</td>
</tr>
<tr>
<td>10</td>
<td>STROBE 1-</td>
<td>23</td>
<td>STROBE 0-</td>
</tr>
<tr>
<td>11</td>
<td>OUTPUT 2-*</td>
<td>24</td>
<td>OUTPUT 1-*</td>
</tr>
<tr>
<td>12</td>
<td>ALL WINDINGS OFF OUT-</td>
<td>25</td>
<td>PULSE OUTPUT</td>
</tr>
<tr>
<td>13</td>
<td>CW/CCW- OUT*</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* These outputs are open collectors. If monitored, they must be pulled up to an external voltage source (maximum 24 Vdc) through a series resistor to limit the sink current to a maximum of 40 milliamperes.

The strobe outputs are also open collectors; if they are connected directly to the indexer’s inputs, pull-ups are not required.

For maximum noise immunity, it is recommended that a twisted, shielded cable no longer than 20 feet be used.

3.6.1.1 Parallel Output Characteristics

The following pertain to Output 1- (pin 24), Output 2- (pin 11), AWO Out- (pin 12), CW/CCW-Out (pin 13), Pulse Output- (pin 25), Motion Busy- (pin 6) and Strobe 0- through Strobe 7- Outputs.

High level output: +24 Vdc max., open collector
High level leakage current: 250 microamperes max.
Low level output: +0.4 Vdc @ 16 mA sink current
Low level leakage current: +0.7 Vdc @ 40 mA sink current

3.6.1.2 Parallel Input Characteristics

The following pertain to D0- through D7- inputs:

High level (inactive) voltage: +8.5 Vdc min.; +15 Vdc max.
High level current: 1 milliampere maximum leakage
Low level (active) voltage: 0.0 Vdc min.; +6.5 Vdc max.
Low level current: 3.5 milliamperes max.

3.6.2 SERIAL I/O CONNECTOR

J4: 9-Pin “D” Type Connector, Female

<table>
<thead>
<tr>
<th>Pin</th>
<th>Assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Vo (Signal Common)</td>
</tr>
<tr>
<td>2</td>
<td>RS232 Chain Out</td>
</tr>
<tr>
<td>3</td>
<td>RS232 Receive Data</td>
</tr>
<tr>
<td>4</td>
<td>Vo (Signal Common)</td>
</tr>
<tr>
<td>5</td>
<td>Vo (Signal Common)</td>
</tr>
<tr>
<td>6</td>
<td>RS232 Echo</td>
</tr>
<tr>
<td>7</td>
<td>RS232 Chain In</td>
</tr>
<tr>
<td>8</td>
<td>+5 Vdc (SSP-500 only)</td>
</tr>
<tr>
<td>9</td>
<td>+5 Vdc (SSP-500 only)</td>
</tr>
</tbody>
</table>

Wire size: 24 AWG minimum. Cable with shielded, twisted pairs is highly recommended.

Run length: 50 feet (15.2m) max.

Cables available from Superior Electric to connect the indexer to typical RS232 devices:

(25-pin “D” connector on one end, 9-pin “D” connector on other)

<table>
<thead>
<tr>
<th>Length</th>
<th>Part No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 feet</td>
<td>B216059-001</td>
</tr>
<tr>
<td>10 feet</td>
<td>B216059-002</td>
</tr>
</tbody>
</table>

(25-pin female “D” connector on one end, 9-pin “D” connector on other)

<table>
<thead>
<tr>
<th>Length</th>
<th>Part No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 feet</td>
<td>B216059-003</td>
</tr>
<tr>
<td>10 feet</td>
<td>B216059-004</td>
</tr>
</tbody>
</table>

3.7 ENVIRONMENTAL REQUIREMENTS

Indexer operating Temp.: +32° F to +122° F (0° C to +50° C)
Free-air ambient

Drive heatsink operating temp.: +32° F to +158° F (0° C to +70° C) measured at center of heatsink

Storage Temp.: -40° F to +167° F (-40° C to +75° C)

Humidity: 95% max., noncondensing

Altitude: 10,000 feet (3048m) max.

Cooling: Will operate at up to +122° F (50° C) as long as maximum heatsink temperature of +158° F (+70° C) is maintained; forced-air (fan) cooling may be required.

NOTE: Forced-air cooling is required to operate the 430-PI indexer at ambient temperatures greater than 95° F (35° C).

![Figure 3.2: Typical Wiring for Open Collector Output](image-url)
3.8 REDUCED CURRENT

WARNING: The following procedure requires disassembly of the enclosure and skilled workmanship must be accomplished. Failure to strictly follow these procedures or entering the enclosure for any other purpose will void the warranty. ALWAYS DISCONNECT POWER AND ALL CABLES BEFORE PROCEEDING. REASSEMBLE THE ENCLOSURE BEFORE POWER IS REAPPLIED.

3.8.1 Reduced Current for 230-PI and 430-PI

It is possible to configure the 230-PI and 430-PI to supply less than rated current to the motor. To do this, a jumper or a resistor is connected between LOGIC COMMON (pin 3) and REDUCE CURRENT- (pin 4) on the indexer circuit card location J5. These pins protrude through holes in the indexer circuit board.

To gain access to the pins, first remove the six screws that hold the nameplate side of the enclosure in position. Then remove the two screws that fasten the heatsink to the other side of the enclosure. Pivot the heatsink together with the attached drive module and indexer card away from the assembly to gain access to pins 3 and 4 of J5.

Care should be taken to keep the resistor or jumper leads less than 2 inches (51 mm) long and to prevent them from contacting each other or any other part of the assembly. This signal is not optically isolated.

The proper resistor values or jumper leads and their associated current values are shown in 3.8.1.1 for the 230-PI and in 3.8.1.2 for the 430-PI.

3.8.1.1 Reduced Current for 230-PI

<table>
<thead>
<tr>
<th>CURRENT (amps.)</th>
<th>RESISTOR (ohms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00</td>
<td>0 (jumper)</td>
</tr>
<tr>
<td>1.25</td>
<td>2.49k, 1/4 watt, 1%</td>
</tr>
<tr>
<td>1.50</td>
<td>7.50k, 1/4 watt, 1%</td>
</tr>
<tr>
<td>1.75</td>
<td>23.7k, 1/4 watt, 1%</td>
</tr>
<tr>
<td>2.0</td>
<td>open</td>
</tr>
</tbody>
</table>

3.8.1.2 Reduced Current for 430-PI

<table>
<thead>
<tr>
<th>CURRENT (amps.)</th>
<th>RESISTOR (ohms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.5</td>
<td>0 (jumper)</td>
</tr>
<tr>
<td>2.0</td>
<td>1.78k, 1/4 watt, 1%</td>
</tr>
<tr>
<td>2.5</td>
<td>5.62k, 1/4 watt, 1%</td>
</tr>
<tr>
<td>3.0</td>
<td>16.2k, 1/4 watt, 1%</td>
</tr>
<tr>
<td>3.5</td>
<td>open</td>
</tr>
</tbody>
</table>
SECTION 4: PROGRAMMING GUIDE

This section contains the information necessary to set up, enter, and edit programs and also to execute programs with the Micro Series Indexer.

By carefully reading this section through in its entirety, the user will fully understand the wide range of applications possible with the Micro Series Indexer.

4.1 OVERVIEW AND SET-UP

EIA (Electronic Industries Association) Standard RS274-D is the programming guide for numerically controlled machines. Superior Electric has utilized this standard to form the basis for the Micro Series Indexer’s command structure. It was not technically desirable to conform to the standard in complete detail, but it proved beneficial in the program structure to perform complex and varied operations with a simple format.

Using straightforward programming formats, the Micro Series Indexer enables the user to program and execute in either parallel (switch panel) or serial (remote terminal or host computer) communication modes.

In general, all parameters and commands can be broadly grouped into four categories, which correspond to these code groupings:

1. L Codes
2. H Codes
3. N, G, X and F Codes
4. Immediate Codes

1. “L Codes” (discussed in Section 4.3) are used to set parameters for each indexer. These commands are not considered part of an indexer program, that is, they are made prior to any motion programming and do not have program line numbers.

It is important to remember that the L codes are used exclusively to set the initial parameters of a particular indexer and should not be thought of as part of the program option for the indexer.

2. “H Codes” (Section 4.4) are used to set indexer modes, control manual and program execution and to transmit parameters and indexer status via the serial communications port.

H codes are not part of the programming commands for the indexer. There are no program line numbers associated with the “H” codes and they are not considered to be part of the programming function.

3. “N, G, X and F Codes” (Section 4.5) are the programming commands for the indexer. Up to 400 lines of program instructions can be stored as a unique motion control program.

Each program is in a fixed format, and is composed of a line number, a “G” code, an “X” code and an “F” field.

A line of program has this format:

A space is used to separate the codes.
Not all codes need be programmed for each program line.
The G, X and F codes may be programmed in any order.

NOTE: The brackets, [], are used in this manual for clarity and are not to be used when entering data or variables.

4. Immediate Codes (Section 4.2) are executed immediately upon receipt and are not stored as part of the program. All commands are highlighted in bold face for easy reference.

4.1.2 General Programming Comments

**IMPORTANT PROGRAMMING NOTES**

1. The indexer contains a 40 character serial buffer to accept all data and programming entries. If a COMMAND TERMINATOR (CARRIAGE RETURN and/or LINE FEED) is not received by the 40th character, the buffer contents are dumped and that 40 character string is lost.

Upon receipt of a CR and/or LF, an XOFF (ASCII Code 19) character is transmitted to the host; no further data transmission from the host should occur. However, any characters transmitted subsequent to XOFF will be stored in the buffer until the buffer capacity is reached. If the capacity is exceeded, the buffer contents are dumped.

The receipt of a COMMAND TERMINATOR character will cause the commands in the buffer to be executed sequentially. That is, the first command that was entered will be the first command executed. The COMMAND DELIMITER for a series of commands is a space. Once all the commands in the buffer have been executed, the indexer will send an XON (ASCII Code 17) character and will be ready to receive further data.

2. Whether the indexer is being operated from a switch panel, remote terminal or host computer, the first task that faces an operator, after all circuit connections have been made, is that of setting “L codes”, the parameters of each indexer.

3. In the following descriptions, it is important to note the factory default values for each parameter as entry steps can be eliminated. Upon receipt of a new indexer, the default values will have been entered for the parameters.

4. Motor speeds and acceleration will depend on the TRANSLATOR RESOLUTION setting (L70 nnn). Set this parameter first and then work in ascending numerical order starting with the L06 parameter.

NOTE: If the L70 value is changed, the values for L09, L11, L12, L14 and the “F” field values must be reentered.

5. Entry of invalid data for a parameter or program field will result in the previous data being left intact.

6. If the number of characters entered exceeds the number of required characters, the data is truncated to the maximum field length for the entry.
"On The Fly" commands can be sent to the indexer when it is "BUSY" (When motion or program execution is active or when previously transmitted commands are being processed). "On The Fly" commands allow the indexer to send status or position information to the host. The sequence for sending "On The Fly" commands is shown below. The host must send an "On The Fly" command, followed immediately by a carriage return (CR) and/or Line Feed (LF). If the command is not sent in this manner, it will not be treated as an "On The Fly" command. Instead, it will be placed in the buffer and executed when the indexer is no longer "BUSY".

The following commands can be sent the indexer while it is "BUSY":

- `HI5crlf` Transmit Present Program Line Number
- `H17crlf` Transmit Present Absolute Position
- `H18crlf` Transmit Mode Status
- `H19crlf` Transmit Motion Status
- `H20crlf` Transmit I/O Status

Any other commands sent to the indexer will be executed when the indexer is no longer "BUSY".

**P.C. PROGRAM FLOWCHART FOR INDEXER COMMUNICATIONS**

```
BEGIN
HOST DEVICE SENDS ATTENTION STRING
"<nn"
WHERE nn = INDEXER I.D.
WAIT FOR INDEXER RESPONSE CHARACTER
TEST RESPONSE CHARACTER
RESPONSE ":" INDEXER IS BUSY
TRY AGAIN
RESPONSE ":=" INDEXER IS READY
CONTINUE
SEND COMMAND STRING
TERMINATE WITH CR OR LF
EXIT
```

Figure 4.1, Program Flow Chart For Indexer Communications
4.2 IMMEDIATE COMMANDS

* CLEAR (ASCII Code 42)

This command IMMEDIATELY halts all motor motion and program execution and will cause a loss of home position. The program line pointer is reset to the line number specified with the L41 parameter.

The RS232 input buffer is also cleared and the indexer transmits an XOFF and then an XON when a CLEAR command is used.

$ FEEDHOLD (ASCII CODE 36)

This command will immediately bring current motor motion to a controlled stop with deceleration determined by the L11 parameter (ACCELERATION/DECELERATION).

Motion can be continued without loss of position by using a H01 CYCLE START command if the feedhold was applied during program execution. If a feedhold was applied during a H08 (return to electrical home) command, a subsequent H08 will continue the motion. A feedhold during a H10 (return to mechanical home) command requires a H10 to continue the motion. The line number cannot be altered during a feedhold as the remaining move distance needs to be completed. A CLEAR (*) can be issued if it is not desired to complete the motion.

<nn DEVICE ATTENTION CHARACTER

If <[nn] = 00, all indexers will be addressed; otherwise, the indexer with the value set by the L21 parameter that matches the <[nn] value will become the active indexer. The indexer will respond with "XON" when the <[nn] value matches the value set with the L21 parameter and the indexer is ready for commands. If the indexer is not ready, the indexer responds with "-". If <00 is used, there is no indexer response.

# CYCLE STOP (ASCII Code 35)

This command halts program execution after the current program line is completed.

This command is used to stop programs executing with LO6 [2] or LO6 [3] EXECUTION FORMAT settings.

^H (CONTROL H) BACKSPACE AND DELETE (ASCII Code 8)

The CONTROL H (^H) command will cause a backspace and delete one character on the current program line. The ^H command is executed immediately upon entry and is not stored as part of the program.

^X (CONTROL X) DELETE LINE (ASCII Code 24)

The CONTROL X (^X) command will delete the program line that has been entered in the line buffer. This command must be used prior to a COMMAND TERMINATOR.

The ^X command is executed immediately upon entry and is not stored as part of the program.

4.3 L Codes: INDEXER PARAMETERS

L codes are used to set parameters for each indexer. These parameters affect the entire operation of the indexer and cannot be changed during program execution; therefore, they should be programmed first. The L codes can be categorized into three areas of interest: motion, execution and serial communication parameters.

Motion parameters: L08, L09, L11, L12, L14, L17, L18, L19, L70

Execution parameters: LO6, LO7, L41, L44, L45, L48, L49

Communication parameters: L21, L22, L23, L25, L26

The L codes have been preset at the factory to default values; these are listed as “Factory Default” settings.

INDEXER PARAMETER FACTORY DEFAULT VALUES

<table>
<thead>
<tr>
<th>Full Step Translator</th>
<th>1/10 Step Translator</th>
<th>1/125 Step Translator</th>
</tr>
</thead>
<tbody>
<tr>
<td>L06 = 1</td>
<td>L06 = 1</td>
<td>L06 = 1</td>
</tr>
<tr>
<td>L07 = 0100</td>
<td>L07 = 0100</td>
<td>L07 = 0100</td>
</tr>
<tr>
<td>L08 = +</td>
<td>L08 = +</td>
<td>L08 = +</td>
</tr>
<tr>
<td>L09 = 0001000</td>
<td>L09 = 0010000</td>
<td>L09 = 0125000</td>
</tr>
<tr>
<td>L11 = 0002500</td>
<td>L11 = 0025000</td>
<td>L11 = 0312500</td>
</tr>
<tr>
<td>L12 = 0000250</td>
<td>L12 = 0002500</td>
<td>L12 = 00312500</td>
</tr>
<tr>
<td>L14 = 001000</td>
<td>L14 = 00100</td>
<td>L14 = 012500</td>
</tr>
<tr>
<td>L17 = 00000000</td>
<td>L17 = 00000000</td>
<td>L17 = 00000000</td>
</tr>
<tr>
<td>L18 = 00000000</td>
<td>L18 = 00000000</td>
<td>L18 = 00000000</td>
</tr>
<tr>
<td>L19 = 00000000</td>
<td>L19 = 00000000</td>
<td>L19 = 00000000</td>
</tr>
<tr>
<td>L21 = 01</td>
<td>L21 = 01</td>
<td>L21 = 01</td>
</tr>
<tr>
<td>L22 = 9600</td>
<td>L22 = 9600</td>
<td>L22 = 9600</td>
</tr>
<tr>
<td>L23 = 8</td>
<td>L23 = 8</td>
<td>L23 = 8</td>
</tr>
<tr>
<td>L25 = 1</td>
<td>L25 = 1</td>
<td>L25 = 1</td>
</tr>
<tr>
<td>L26 = 0</td>
<td>L26 = 0</td>
<td>L26 = 0</td>
</tr>
<tr>
<td>L41 = 000</td>
<td>L41 = 000</td>
<td>L41 = 000</td>
</tr>
<tr>
<td>L44 = 0050</td>
<td>L44 = 0050</td>
<td>L44 = 0050</td>
</tr>
<tr>
<td>L45 = 0</td>
<td>L45 = 0</td>
<td>L45 = 0</td>
</tr>
<tr>
<td>L48 = 000</td>
<td>L48 = 000</td>
<td>L48 = 000</td>
</tr>
<tr>
<td>L49 = 00</td>
<td>L49 = 00</td>
<td>L49 = 00</td>
</tr>
<tr>
<td>L70 = 001</td>
<td>L70 = 010</td>
<td>L70 = 125</td>
</tr>
</tbody>
</table>
LO6 \[n\] PROGRAM EXECUTION FORMAT

This command determines the fashion in which a program will be executed after a CYCLE START (H01) command is issued. The options are:

\[n\] = 1 Single-line program execution format

When a CYCLE START command is given, the current program line is executed. The line pointer is then incremented to the next program line and the cycle stops. The program line pointer may be moved to another line by using the N [nnn] command. Another CYCLE START (H01) command will repeat the process.

\[n\] = 2 Automatic program execution format

In this mode, a CYCLE START command will cause the program to execute from the present program line to line number 400 or to a line command that contains a G30 that is not a part of a G11 (subroutine call) command. The program will then halt execution and the program line pointer will be set to the line number specified with the L41 parameter. A second CYCLE START command will cause the program to start execution from this line number.

\[n\] = 3 Continuous program execution format

In this mode, a CYCLE START command will cause the program to execute from the present program line to line number 400 or to a line command that contains a G30 that is not part of a G11 (subroutine call) command. The program line pointer is then set to the line number specified with the L41 parameter and program execution continues until a # CYCLE STOP command or a "CLEAR command is issued.

Example: L06 2 CRLF sets automatic program execution format. [CRLF = carriage return, line feed].

Factory default: L06 1

NOTES: When line 00 (MDI line) is executed, the execution format is ignored as line 00 executes one time for each CYCLE START and the line pointer remains at line 00.

This command is transparent when using the SSP-500. The mode is changed when using specific displays. Refer to the SSP-500 manual for details.

LO7 [nn] STROBE DELAY TIME

This sets the length of time, in milliseconds, that the output strobe line will be ACTIVE (low) before the connected data line is read. The delay range is 0 milliseconds to 9999 milliseconds. See Section 5 for an explanation of the STROBE and DATA functions.

Factory default L07: [nn] = 100 milliseconds

Strobe delay accuracy is ±2.5% of the selected value.

Example: L07 1000 CRLF sets a 1000 millisecond strobe delay.

LO8 [s] MECHANICAL HOME DIRECTION

Sets the direction that the motor will turn when a H10 or G78 RETURN TO MECHANICAL HOME command is issued.

A "+" will cause the motor to turn clockwise.

A "-" will cause the motor to turn counterclockwise.

Factory default L08: s = +.

Example: L08 + CRLF sets a clockwise mechanical home direction.

LO9 [nnnnnn] JOG SPEED

Sets the desired speed, in pulses/second, to be used when the motor is run in the JOG mode and the HIGH SPEED mode. Allowable values for this parameter are:

- full step (L70 = 1) 0 to 115,000 pulses/sec.
- half step (L70 = 2) 0 to 115,000 pulses/sec.
- 1/5 step (L70 = 5) 0 to 115,000 pulses/sec.
- 1/10 step (L70 = 10) 0 to 115,000 pulses/sec.
- 1/125 step (L70 = 125) 0 to 1,875,000 pulses/sec.

Factory default:

- full step (L70 = 1) 1,000 pulses/sec.
- half step (L70 = 2) 2,000 pulses/sec.
- 1/5 step (L70 = 5) 5,000 pulses/sec.
- 1/10 step (L70 = 10) 10,000 pulses/sec.
- 1/125 step (L70 = 125) 125,000 pulses/sec.

Example: L09 1200 CRLF sets a jog speed of 1200 pulses/sec.

PLEASE NOTE: The range of values that will be accepted by the L09, L11, L12 and L14 parameters is set by the L70 TRANS- LATOR RESOLUTION value.

If L70 is set or changed after setting the L09, L11, L12 or L14 parameters, the range may not be valid and incorrect motion may occur.

ALWAYS SET THE L70 PARAMETER FIRST.
**L11 [nnnnnnn] ACCELERATION/DECELERATION**

Sets the value, in pulses/sec/sec, for acceleration and deceleration. The same rate applies to both.

Whenever the indexer initiates motion, it is always at the speed set with the L12 LOW SPEED parameter. This is the instantaneous starting speed of a motor; it will then ramp up to the JOG or HOME SPEED value or the FEED RATE using the acceleration rate set with the L11 ACCELERATION/DECELERATION parameter.

When the motor is running at JOG SPEED and a feedhold is asked for, the motor will decelerate from JOG SPEED using the deceleration rate set with the L11 parameter and will then stop.

When a move distance and feed rate have been programmed, the motor will decelerate to a stop at the end of the move using the deceleration rate set with the L11 parameter.

The ranges for ACCELERATION/DECELERATION are: [nnnnnnnn] = 25 to 9,999,999 pulses/sec/sec.

Factory defaults:

- **full step** \( (L70) = 1 \) 2,500 pulses/sec/sec
- **half step** \( (L70) = 2 \) 5,000 pulses/sec/sec
- **1/5 step** \( (L70) = 5 \) 12,500 pulses/sec/sec
- **1/10 step** \( (L70) = 10 \) 25,000 pulses/sec/sec
- **1/125 step** \( (L70) = 125 \) 312,500 pulses/sec/sec

Example: L11 5000 CRLF sets an acceleration/deceleration rate of 5000 pulses/second/second.

---

**L12 [nnnnnnn] LOW SPEED**

This command sets the speed at which the motor will run if the H05 LOW SPEED MODE command is used, or the speed at which the motor will start before accelerating to the HIGH SPEED value.

The ranges for LOW SPEED are:

- **full step** \( (L70) = 1 \) 0 to 115,000 pulses/sec.
- **half step** \( (L70) = 2 \) 0 to 115,000 pulses/sec.
- **1/5 step** \( (L70) = 5 \) 0 to 115,000 pulses/sec.
- **1/10 step** \( (L70) = 10 \) 0 to 115,000 pulses/sec.
- **1/125 step** \( (L70) = 125 \) 0 to 1,875,000 pulses/sec.

Factory defaults:

- **full step** \( (L70) = 1 \) 250 pulses/sec.
- **half step** \( (L70) = 2 \) 500 pulses/sec.
- **1/5 step** \( (L70) = 5 \) 1,250 pulses/sec.
- **1/10 step** \( (L70) = 10 \) 2,500 pulses/sec.
- **1/125 step** \( (L70) = 125 \) 31,250 pulses/sec.

Example: L12 300 CRLF sets a low speed of 300 pulses/sec.

---

**L14 [nnnnnnn] HOME SPEED**

This command sets the speed, in pulses/second, with which the motor will return home when a H10 or G78 RETURN TO MECHANICAL HOME, or a H08 or G76 RETURN TO ELECTRICAL HOME command is executed.

The ranges for the HOME SPEED are:

- **full step** \( (L70) = 1 \) 0 to 115,000 pulses/sec.
- **half step** \( (L70) = 2 \) 0 to 115,000 pulses/sec.
- **1/5 step** \( (L70) = 5 \) 0 to 115,000 pulses/sec.
- **1/10 step** \( (L70) = 10 \) 0 to 115,000 pulses/sec.
- **1/125 step** \( (L70) = 125 \) 0 to 1,875,000 pulses/sec.

Factory default:

- **full step** \( (L70) = 1 \) 1,000 pulses/sec.
- **half step** \( (L70) = 2 \) 2,000 pulses/sec.
- **1/5 step** \( (L70) = 5 \) 5,000 pulses/sec.
- **1/10 step** \( (L70) = 10 \) 10,000 pulses/sec.
- **1/125 step** \( (L70) = 125 \) 125,000 pulses/sec.


---

**L17 [nnnnnnnnn] OFFSET DIRECTION AND DISTANCE FROM MECHANICAL HOME**

This command sets the direction and distance, in pulses, that the motor will automatically move after a TO MECHANICAL HOME command has been issued and the motor has returned to the HOME LIMIT switch.

See Section 5.2.3 on hardware inputs for a description of the HOME LIMIT switch function.

\[ [s] = + \text{ or } - \]

\[ [nnnnnnnn] = 0 \text{ to } 99,999,999 \text{ pulses}. \]

Factory default L17: \[ [s] = + [nnnnnnnn] = 0 \text{ pulses}. \]

Example: L17 -1000 CRLF sets an offset of 1000 pulses in the negative direction.

---

**L18 [nnnnnnnnn] CLOCKWISE SOFTWARE TRAVEL LIMIT**

\[ s = + \text{ to enable the feature} \]

\[ s = - \text{ to disable the feature} \]

\[ [nnnnnnnn] = 0 \text{ to } 99,999,999 \text{ pulses} \]

When enabled, motion in the clockwise direction that causes the absolute position to exceed the L18 value will result in a feedhold being activated. Only motion in the opposite direction is permitted.

Factory default L18: \[ [s] =-[nnnnnnnn] = 0 \text{ pulses} \]

Example: L18 +5000 CRLF will enact a feedhold should the absolute position exceed +5000 and no further clockwise motion is permitted.
L19 \{snnnnnnnn\} COUNTERCLOCKWISE SOFTWARE TRAVEL LIMIT

s = - to enable the feature
s = + to disable the feature
\[nnnnnnnn\] = 0 to 99,999,999 pulses

When enabled, motion in the counterclockwise direction that causes the absolute position to exceed the L19 value will result in a feedhold being activated. Only motion in the opposite direction is permitted.

Factory default L19: \[s\] = + [nnnnnnnn] = 0 pulses

Example: L19 -7500 CRLF will enact a feedhold should the absolute position exceed -7500 and no further counterclockwise motion is permitted.

L21 \{nn\} ASSIGN DEVICE IDENTIFICATION NUMBER

This will assign a unique device number to each indexer in situations where multiple indexers (up to 99 are permitted) have been daisy-chained.

This number will be used with the \<\{nn\}\> DEVICE ATTENTION command to activate a particular indexer.

The range for L21 is 01 to 99
Factory default L21: \[nn\] = 01

Example: L21 05 CRLF sets an identification number of 5.

L22 \{nnnn\} BAUD RATE

This command sets the serial port data transfer rate in bits/second (baud).

The acceptable values for \{nnnn\} are:
\[nnnn\] = 300, 1200, 2400 and 9600
Factory default L22: \{nnnn\} = 9600

NOTE: Most of the recently manufactured terminals and desktop computers are set at 9600 baud. If the computer or terminal used is not 9600 baud, it will not be able to communicate with a new indexer. A nondefault rate will have to be set either with a compatible device or with a switch panel as described in Section 5.

L23 \{nn\} CHARACTER LENGTH

This sets the number of data bits in a character sent via serial communications.

The acceptable values for \{nn\} are: \{nn\} = 7 or 8.
Factory default L23: \{nn\} = 8

L25 \{n\} PARITY

This setting determines whether parity applies to serial communication commands and, if parity is selected, whether odd or even parity is applied. The acceptable values for \{nn\} are:
\[nn\] = 1 Parity disabled
\[nn\] = 2 Parity enabled, odd parity
\[nn\] = 3 Parity enabled, even parity
Factory default L25: \[nn\] = 1, Parity disabled.

NOTE: When parity is disabled, two stop bits will be sent, regardless of the character length. When odd or even parity is set, one stop bit will be sent if L23 CHARACTER LENGTH is set to \[nn\] = 8. Two stop bits will be sent if L23 is set to \[nn\] = 7.

L26 \{n\} ACKNOWLEDGE

This parameter selects the transmission protocol which the Indexer will use when responding to input commands. If the command L26 \{n\} is issued while the Indexer is busy, the command will be processed when the Indexer is no longer busy.

Range: 0 to 7
Factory Default L26: \[n\] = 0

\n
<table>
<thead>
<tr>
<th>Transmission Mode Selected</th>
<th>Xon/Xoff Protocol Enabled</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 Normal Transmission Mode (No &quot;EOT&quot; or &quot;;&quot; characters)</td>
<td>Xon/Xoff Protocol Disabled</td>
</tr>
<tr>
<td>1 &quot;EOT&quot; follows each complete data transmission to the host</td>
<td>5 &quot;EOT&quot; follows each complete data transmission to the host</td>
</tr>
<tr>
<td>2 &quot;;&quot; is transmitted when Indexer is ready for more commands</td>
<td>6 &quot;;&quot; is transmitted when Indexer is ready for more commands</td>
</tr>
<tr>
<td>3 &quot;EOT&quot; follows each complete data transmission to the host and &quot;;&quot; is transmitted when the Indexer is ready for more information</td>
<td>7 &quot;EOT&quot; follows each complete data transmission to the host and &quot;;&quot; is transmitted when the Indexer is ready for more information</td>
</tr>
</tbody>
</table>

Xon character is ASCII code 17
Xoff character is ASCII code 19
EOT character is ASCII code 04

Xon/Xoff Protocol:
The Xoff character is transmitted to the host when a CR or LF is received. The Indexer will process the information it has received and will transmit an Xon character when it is ready to accept more information from the host. The Indexer should be polled to determine when it is ready to accept more information. If L26 \{n\} is selected, the Indexer will transmit an ";" if it is ready to accept more information. If an Xoff character has been transmitted to the host and the command received by the Indexer calls for motion or program execution, the Indexer will send an Xon to the host. This allows the host to send any of the immediate commands such as " " (Clear), "$" (Feed Hold) or ";" (Cycle Stop). The host should not send "normal" commands until the Indexer is ready to accept more information. The Indexer will be ready to accept more information when motion is stopped, program is stopped and all previous commands have been executed.
L41 [nnn] AUTO START LINE NUMBER

This parameter determines the line number to which the program line pointer will automatically be set upon powerup, during invalid program execution, upon encountering a G30 command that is not part of a G11 subroutine call and after a clear command (*) has been executed.

The acceptable values for [nnn] = 0 to 400.
Factory default L41: [nnn] = 0

Example: L41 200 CRLF sets the autostart line number to 200.

L44 [nnnn] PROGRAM LINE DELAY

During program execution, a delay of L44 parameter length in milliseconds will occur after each line completes executing. This delay should be set to allow sufficient motor setting time. The delay is also invoked after G76 or G78 execution.

The acceptable values for [nnnn] = 0 to 9999 milliseconds.
Factory default L44: [nnnn] = 50 milliseconds.

Example: L44 250 CRLF sets a delay of 250 milliseconds after execution of each line.

L45 [n] LIMIT SWITCH ENABLE

The user can utilize the inputs labeled CW LIMIT- and CCW LIMIT- either as limit switches (L45 0) or as additional programmable inputs (L45 1).

[n] = 0 to enable limit switch operation
[n] = 1 to disable limit switch operation for utilization as additional programmable inputs.
Factory default L45: [n] = 0

Example: L45 1 CRLF disables limit switch operation as the two limit switch inputs are used as additional programmable inputs.

Note: When [n] = 1, the CW LIMIT- input becomes input 3 and the CCW LIMIT- input becomes input 4.

L48 [nnn] PROGRAM LINE TRANSFER COUNT

Used in conjunction with the H12 and H14 commands, L48 contains the number of lines the command is to act upon. If L48 contains 0, lines 1 through 400 will be cleared (H12) or transferred (H14). Otherwise, the command is effective for the number of lines indicated by the L48 value starting from the present line number.

Factory default L48: [nnn] = 000

Example: L48 10 CRLF sets a 10 line execution block for H12 and H14 commands.

NOTE: L48 MUST BE SET TO 0 TO OPERATE WITH AN SSP-500.

L49 [nn] PARAMETER TRANSFER DESIGNATION

Used in conjunction with the H16 command, L49 contains the designated parameter to transfer. If L49 contains 0, all parameters are transferred. Otherwise, only the parameter designated by L49 is transferred. If L49 contains an invalid parameter, H16 transmits only a carriage return/line feed. This parameter is NOT stored in nonvolatile memory.

Factory default L49: [nn] = 00

Example: L49 06 CRLF causes H16 to transfer the contents of the L06 parameter.

NOTE: L49 MUST BE SET WITH 0 TO OPERATE WITH AN SSP-500.

L70 [nnn] TRANSLATOR RESOLUTION

This sets the step resolution of the motor drive translator.

The acceptable values for [nnn] are:
- full step 1
- half step 2
- 1/5 step 5
- 1/10 step 10
- 1/125 step 125

* These settings are only valid for the 440, 3180 and 6180 Series drives.

Factory default:
- full step 1
- 1/5 step 5
- 1/10 step 10
- 1/125 step 125

Example: L70 2 CRLF sets a half step translator resolution.

NOTE: For all 230 and 430 Series indexers, the L70 parameter must be set to 1 or 2 for the drive to work properly.

NOTE: Since the ranges of L09, L11, L12 and L14 depend on the setting of L70, it is important to set the L70 parameter first so that range confusion will be avoided.

4.4 H CODES: COMMANDS FOR MODES OF OPERATION

H codes perform four different functions in indexer operations:

1. They control manual and program execution. H codes, however, cannot be used as programmable instructions.
2. They are used to set MODES OF OPERATION.
3. They are used in PROGRAM EDITING to clear program data.
4. They issue transmission instructions, that is, they allow stored parameters and status data to be TRANSFERRED via the serial communications port.

The H codes are categorized into the following four areas of interest:

Mode Commands: H2, H3, H4, H5, H24, H25
Execution Commands: H1, H6, H7, H8, H9, H10
Transfer Commands: H13, H14, H15, H16, H17, H18, H19, H20, H23
Edit Commands: H11, H12

For ease of reference, the H codes are presented here in numerical order.
**H01 CYCLE START**

This is a "GO" command. It will start program execution from the present line number based on the execution format set with L06.

This command will also restart motion during program execution after a $ FEEDHOLD command.

**H02 STEP MODE**

This command sets the indexer in the single-step mode for manual motion operation.

**NOTE:** The H02 and H03 commands do not cause motion, they only set the mode in which motion will occur when it is called for.

Motion is started by issuing a H06 CLOCKWISE or a H07 COUNTERCLOCKWISE command after setting the H02 STEP or H03 JOG mode. When the H02 STEP mode command is followed by a H06 or H07 command, a single step will be made by the motor.

Upon power up of the system, the indexer is set in the STEP mode.

**H03 JOG MODE**

This command sets the indexer in the continuous motion mode for manual motion operation.

When a H06 or H07 command is issued after a H03 command, the motor will turn continuously until a $ FEEDHOLD command is issued.

The speed at which the motor will turn is dependent on whether the HIGH SPEED or the LOW SPEED mode has been selected.

If the HIGH SPEED mode is enabled, the motor will run at the value set with the L09 parameter. If the LOW SPEED mode is enabled, the motor will run at the value set with the L12 parameter.

**H04 HIGH SPEED MODE**

This command allows the motor to run at the speed, in pulses per second, that was set with the F[x] FEEDRATE command during program execution, or at the value set with the L09 parameter when the JOG mode is selected.

When a H06 or a H07 command is issued after a H04 command, the motor will accelerate according to the value set with the L11 parameter to the jog speed set with the L09 value. JOGGING is terminated with a $ FEEDHOLD command and the motor decelerates to a stop according to the value set with the L11 parameter.

When the indexer is powered up, it is set in the HIGH SPEED MODE.

**H05 LOW SPEED MODE**

This command allows the motor to run at the speed, in pulses per second, that was set with the L12 LOW SPEED parameter. No acceleration or deceleration is allowed.

A H05 command followed by a H03 and a H06 or H07 command will cause the motor to JOG at the speed set with the L12 parameter. If the LOW SPEED mode is enabled when a H01 CYCLE START command is issued, all motion will be at the value set with the L12 parameter. Any F field values will be ignored.

**H06 TURN IN CW DIRECTION**

This command will cause the motor to move in the CLOCKWISE direction.

The type of motion will depend on whether the indexer is in the STEP or JOG mode, the HIGH SPEED or LOW SPEED mode and the parameters set with the L09, L11 and L12.

**H07 TURN IN CCW DIRECTION**

This command will cause the motor to turn in the COUNTERCLOCKWISE direction. The type of motion will depend on whether the indexer is in the STEP or JOG mode, the HIGH SPEED or LOW SPEED mode and the parameters set with the L09, L11 and L12.

**H08 RETURN TO ELECTRICAL HOME**

Electrical home is established when the indexer is powered up or when a H09 SET HOME command is issued. This command will cause the motor to return to the absolute position of 0 (electrical home).

In executing this motion, the motor will move in the opposite direction of the absolute position sign until the absolute position counter reaches 0.

**H09 SET ELECTRICAL HOME**

This command will set the absolute position counter to zero. This, in effect, sets the current motor position as the electrical home position.

**H10 RETURN TO MECHANICAL HOME**

This command will cause the indexer to return the motor to the MECHANICAL HOME LIMIT SWITCH.

When this command is being executed, the motor will turn in the direction set with the L08 parameter and will offset the motor from the home limit switch the direction and distance set with the L17 parameter, at the speed set with the L14 parameter. When motion is completed, the motor position becomes electrical home as the absolute position counter is set to zero.
**H11 CLEAR PRESENT PROGRAM LINE**

This command will delete the contents of the program line designated by the line pointer.

**USE WITH CAUTION.**

Once deleted, the information contained in that program line is lost and is impossible to retrieve.

Example: **N020 H11 CRLF** will delete the contents of line 20.

**H12 CLEAR PROGRAM**

This command, with **L48 = 0**, will clear the entire program that is stored in the indexer EEROM.

*Lines 1 through 400 will be IRRETRIEVABLY ERASED.*

NOTE: After a CLEAR PROGRAM command, with **L48 = 0**, the line pointer will be returned to the value set by **L41**.

If all 400 lines are programmed, this command may take as long as 15 seconds to complete execution.

An **H12** is not allowed while in a FEEDHOLD condition.

Example: **L48 5 N010 H12 CRLF** results in the deletion of the contents of lines 10, 11, 12, 13 and 14.

An additional **H12** command deletes the contents of lines 15, 16, 17, 18 and 19.

**H13 TRANSMIT CONTENTS OF PRESENT PROGRAM LINE**

This command causes the indexer to transmit the contents of the program line at the program line pointer through the RS232 serial interface port.

**H13** transfers are always in a fixed format as follows:

- **Nnnn Gnn X+nnnnnnnn FnnnnnnnCRLF**
  - **N** is in column 1
  - **G** is in column 6 (if programmed, otherwise a space)
  - **X** is in column 10 (if programmed, otherwise a space)
  - **F** is in column 21 (if programmed, otherwise a space)

If a program line does not contain all the fields, spaces fill in the unprogrammed fields.

Example: If program line number 100 contains "G90 X+1234 F5678.", then

**N100 H13 CRLF** would result in the following transmission:

**N100 G90 X+0001234 F0005678CRLF**

**H14 TRANSMIT PROGRAM**

This command, with **L48 = 0**, causes the indexer to transmit program lines 1 through 400 through the RS232 serial port.

Each program line is transmitted in the same format as H13.

- **N001 Gnn X+nnnnnnnn FnnnnnnnnCRLF**
- **N002 Gnn X+nnnnnnnn FnnnnnnnnCRLF**
- **N399 Gnn X+nnnnnnnn FnnnnnnnnCRLF**
- **N400 Gnn X+nnnnnnnn FnnnnnnnnCRLF**

With **L48 = 0**, after the **H14** command is executed, the line number is reset to the value set by **L41**.

After **H14** execution during a feed hold, the line number is reset to the line number that was active during the feed hold.

Example: **L48 2 N020 H14 CRLF** results in the transfer of the contents of lines 20 and 21.

- **N020 G99 X+0001000 F0002000CRLF**
- **N021 G91 X+0002000 F0002500CRLF**

An additional **H14** CRLF transfers the contents of line 22 and 23.

- **N022 G64 X+0001234 CRLF**
- **N023 G30 CRLF**

**H15 TRANSMIT THE CURRENT PROGRAM LINE NUMBER**

This command will transmit the current program line number. This H code can be executed while the indexer is in motion, executing a program or in parallel mode.

Example: If the line number is equal to 100 **H15 CRLF** results in **N100CRLF** being transmitted by the Indexer.

**H16 TRANSIT PARAMETERS**

This command, with **L49 = 0**, will cause the contents of the L parameters to be transmitted through the RS232 serial port in the order shown below:

**NOTE:** The Brackets [ ] are for clarity and are not transmitted.

- **L06 [n]CRLF**
- **L07 [nnnn]CRLF**
- **L08 [s]CRLF**
- **L09 [nnnnnn]CRLF**
- **L11 [nnnnnn]CRLF**
- **L12 [nnnnnnn]CRLF**
- **L14 [nnnnnnnn]CRLF**
- **L17 [nnnnnnnn]CRLF**
- **L18 [nnnnnnnn]CRLF**
- **L19 [nnnnnnnn]CRLF**
- **L21 [nn]CRLF**
- **L22 [nnnn]CRLF**
- **L23 [n]CRLF**
- **L25 [n]CRLF**
- **L26 [n]CRLF**
- **L41 [nn]CRLF**
- **L44 [nnn]CRLF**
- **L45 [n]CRLF**
- **L48 [nn]CRLF**
- **L49 [nn]CRLF**
- **L70 [nn]CRLF**

With **L49 = 0**, after the **H14** command is executed, the line number is reset to the value set by **L41**.

After **H14** execution during a feed hold, the line number is reset to the line number that was active during the feed hold.

Example: **L48 2 N020 H14 CRLF** results in the transfer of the contents of lines 20 and 21.

- **N020 G99 X+0001000 F0002000CRLF**
- **N021 G91 X+0002000 F0002500CRLF**

An additional **H14** CRLF transfers the contents of line 22 and 23.

- **N022 G64 X+0001234 CRLF**
- **N023 G30 CRLF**

**L49 99 H16 CRLF** results in no parameter transfer as **L99** is nonexistent: **CRLF** is transmitted.
**H17 TRANSMIT ABSOLUTE POSITION**

This command causes the indexer to transmit the current contents of the absolute position counter through the RS232 serial interface port. This H Code can be executed while the indexer is in motion, executing a program or in the parallel mode. The data transmission format is:

```
 snnnnnnnnnn CRLF
```

**EXAMPLE:** If H17 CRLF results in +0000001000CRLF being transmitted, the absolute position is +1000 pulses.

**CAUTION:** If motion is called for that exceeds the maximum absolute position (9,999,999,999 pulses), the absolute position counter will roll over (reset to 0 and continue counting). In this case, the absolute position counter would contain erroneous information.

**H18 TRANSMIT MOTION STATUS**

This command will transmit via the RS-232 serial port the present status of the + LIMIT, - LIMIT, HOME LIMIT, CLEAR, FEED HOLD, and STOP EXECUTION data inputs. This H Code can be executed while indexer is in motion, executing a program or in parallel mode.

The data is transmitted in the following format:

```
00000000CRLF

.n...... STOP EXEC  0=inactive  1=active
..n.... CCW DIR     0=inactive  1=active
...n.... CW DIR     0=inactive  1=active
....n.... FEEDHOLD  0=inactive  1=active
.....n... CLEAR     0=inactive  1=active
.......n HOME LIMIT 0=inactive  1=active
..........n -LIMIT   0=inactive  1=active
...........n +LIMIT 0=inactive  1=active
```

**EXAMPLE:** If H18 CRLF results in 00001001CRLF being transmitted, CW limit and clear are active, the others are inactive.
This command will transmit through the RS232 serial port the status of the indexer modes. This H Code can be executed while the indexer is in motion, executing a program or in parallel mode.

The format for the data transfer is as follows:

```
00000000CRLF
```

<table>
<thead>
<tr>
<th>n</th>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>PROGRAM EXECUTION INACTIVE</td>
<td>0</td>
</tr>
<tr>
<td>.</td>
<td>MOTION INACTIVE</td>
<td>0</td>
</tr>
<tr>
<td>..</td>
<td>INCREMENTAL MODE SELECTED</td>
<td>0</td>
</tr>
<tr>
<td>...</td>
<td>ALL WINDINGS OFF CANCELLED</td>
<td>0</td>
</tr>
<tr>
<td>....</td>
<td>BOOST CURRENT CANCELLED</td>
<td>0</td>
</tr>
<tr>
<td>......</td>
<td>REDUCE CURRENT CANCELLED</td>
<td>0</td>
</tr>
<tr>
<td>........</td>
<td>LOW SPEED MODE SELECTED</td>
<td>0</td>
</tr>
<tr>
<td>........</td>
<td>STEP MODE SELECTED</td>
<td>0</td>
</tr>
</tbody>
</table>

1 = PROGRAM EXECUTION ACTIVE
1 = MOTION ACTIVE
1 = ABSOLUTE MODE SELECTED
1 = ALL WINDINGS OFF ENABLED
1 = BOOST CURRENT ENABLED
1 = REDUCE CURRENT ENABLED
1 = HIGH SPEED MODE SELECTED
1 = JOG MODE SELECTED

EXAMPLE: If H19 CRLF results in 00100011CRLF being transmitted, jog mode, high speed mode and absolute mode are enabled.
**H20 TRANSMIT I/O STATUS**

This command causes the indexer to transmit the status of the two inputs and the two outputs. This H code can be executed while the indexer is in motion, executing a program or in parallel mode.

The format for the data transfer is as follows:

00000000CRLF
0000n... INPUT 2 0=inactive 1=active
0000 .n. INPUT 1 0=inactive 1=active
0000 .n. OUTPUT 2 0=inactive 1=active
0000...n. OUTPUT 1 0=inactive 1=active

**EXAMPLE:** If H20 CRLF results in 00001001CRLF being transmitted, then output 1 and input 2 are active, and output 2 and input 1 are inactive.

**H23 TRANSMIT SOFTWARE REVISION DATE**

This command causes the indexer to transmit the software revision date via the RS232 serial interface port. The date is transmitted in the following format:

IND mm/yy/xCRLF

where IND=indexer, mm=month, yy=year of the software revision and x=software level

**EXAMPLE:** H23 CRLF results in IND 04/87/C being transmitted, which indicates that indexer software level C was released in April of 1987.

**H24 ENABLE TRACE MODE**

This command causes the indexer to transmit the contents of each line being executed during program execution.

The data is transmitted in the same format as an H13 transfer.

**NOTE:** TRACE MODE MUST BE OFF TO OPERATE WITH AN SSP-500 INDEXER PROGRAMMER

**H25 DISABLE TRACE MODE**

This command causes the indexer to disable the trace mode during program execution. Upon power up, the indexer is set in the trace disabled mode.

**NOTE:** The following H codes are able to be executed while the indexer is in motion, executing a program, or in parallel mode: H15, H17, H18, H19, and H20. These codes enable the user to monitor the Indexer "on the fly". Each command must be immediately followed with a carriage return and/or line feed during "on the fly" operations.

4.5 N, G, X, F Codes: PROGRAMMING CODES

**N[nnn] LINE NUMBER**

This command will set the program line pointer to line specified by [nnn].

[nnn] can have a value from 0 to 400.

Powerup default N: [nnn] = The line number in the L41 parameter.

If a line number greater than 400 is entered, the program line pointer will reset to the line number specified with the L41 parameter.

**NOTE:** If program line 00 (N00) is used at any time, the MANUAL DATA INTERFACE mode (MDI) will be enabled.

Program line 00 is stored in a volatile memory only and its contents will be lost when power is removed from the indexer.

N00 is used to store a single program line for manual operation and will not change the contents of the stored program. Upon power up, the MDI line’s F code is loaded with the contents of the L09 parameter.

**X [nnnnnnnn] MOVE DIRECTION AND DISTANCE**

If in incremental mode, the motor will move in either the + (clockwise) or - (counterclockwise) direction for the distance specified, in pulses. If in absolute mode, the motor will move to the absolute position specified in the X field. The X field range is from 0 pulse to 99,999,999 pulses.

The X field is also used with the G04, G11, G20, G22 and G47 commands to specify additional data. See the descriptions of The G04, G11, G20, G22 and G47 commands for additional information.

**NOTE:** No motion can be programmed on a line that contains a G04, G11, G20, G22, G30, G31, G37 or G47 command.

**F [nnnnnn] FEEDRATE**

This command is used to specify the motor speed, in pulses/second, at which a move will occur.

The range of F [nnnnnn] is:

0 to 115,000 pulses/second if the L70 parameter is 1, 2, or 10.

0 to 1,875,000 pulses/sec if the L70 parameter is 125

The F field is also used with G11 and G20 commands to specify additional data.

**NOTE:** A FEEDRATE SHOULD BE SPECIFIED IN THE FIRST LINE OF THE PROGRAM.

If no FEEDRATE is specified as part of the program line, the last FEEDRATE that was programmed will be used.

4.5.5 G Codes PROGRAMMABLE COMMANDS

G codes are preparatory commands that are stored and executed within a program. Several G codes are modal commands which are those functions that retain their state until canceled or superseded by a subsequent command. Attempting to execute a G code without required additional data causes program execution to halt and the line number is loaded with the L41 parameter.
**G04 Dwell Time**

This command allows a delay to be entered in the program.

The command G04 must be followed by an X field value that is the desired delay in milliseconds. The X value range is 0 to 9999 msec and any sign is ignored. The correct format for entering a delay is:

```
N001 G04 X1000 CRLF
```

G04 calls for a delay and X1000 sets the delay at 1000 milliseconds.

**NOTE:** The F field value is ignored during the execution of a G04 command.

If any value is entered that is greater than 9999, it will automatically be truncated to the four least significant digits. Therefore, entering N120 G04 X612000 CRLF will cause a delay of 2000 milliseconds.

Dwell time accuracy is ±2.5% of the selected value.

**G11 Call a Subroutine**

This command calls a subroutine.

A G11 command must be followed by a X value that gives the starting line number of the subroutine and an F value that is the number of times the subroutine is to be repeated.

The range for the X value is 1 to 400 (with the sign being ignored); the range for the F value is 0 to 9999. The correct format for a subroutine call is:

```
N001 G11 x15F1 CRLF
```

This will set the program line pointer to line 15 and execute the subroutine twice.

See the G30 command for terminating a subroutine.

**NOTE:** Using an F value of 0 will cause the subroutine to be executed once; an F value of 1 will cause the subroutine to be executed twice, etc. The number of executions is the F field value +1.

The F field cannot be left blank as this may cause the program to hang up.

**NOTE:** Subroutines can not be nested. Trying to execute a G11 command while a G11 subroutine call is in progress will cause program execution to cease and the program line pointer will automatically reset to the L41 value.

The X field value is truncated to the four least significant digits. The only values that are valid are numbers from 1 to 400. A 0 or a number greater than 400 will cause program execution to cease and the program line pointer will reset to the L41 value.

**G20 Condition L Branch (L45=0)**

This command consists of a G20, an X value and an F value.

The G20 command will cause the program line pointer to jump to the program line number given by the F field if the state of the two inputs correspond to the values of the X field.

The acceptable values for the X field are:

```
x [n2] [n1]  
```

- n1 = 0 = input 1 inactive
- n1 = 1 = input 1 active
- n2 = 0 = input 2 inactive
- n2 = 1 = input 2 active

Any value for n1 or n2 from 2 to 9 will signify a "don't care" condition and that input will be ignored.

If the states of input 1 and input 2 match the values of n1 and n2, the program will jump to the line number indicated by the F field value (1 TO 400 ONLY).

If the states of input 1 and input 2 do not match the programmed values of n1 and n2, the program line pointer is incremented to the next program line.

**G20 Options:**

- **G20 X00:** Go to line number (F field) if input 2 off and input 1 off
- **G20 X01:** Go to line number (F field) if input 2 off and input 1 on
- **G20 X10:** Go to line number (F field) if input 2 on and input 1 off
- **G20 X11:** Go to line number (F field) if input 2 on and input 1 on
- **G20 X0d:** Go to line number (F field) if input 2 off, ignore input 1
- **G20 X1d:** Go to line number (F field) if input 2 on, ignore input 1
- **G20 X00:** Go to line number (F field) if input 1 off, ignore input 2
- **G20 X1d:** Go to line number (F field) ignore input 2, ignore input 1

```
d = 2 through 9: don't care condition
```

**Example:** N100 G20 X 22 F120 CRLF

Will always cause a jump to line 120 when executed because both n1 and n2 signify a "don't care" condition.

```
N100 G20 X01 F120 CRLF
```

When executed, will cause a jump to line 120 if input 1 is active and input 2 is inactive. Otherwise, the program will increment to line 101.

The F field value is truncated to the least four significant digits, and if the F field value is 0 or greater than 400, program execution will cease and the program line pointer will be set to the L41 parameter.
**G20 CONDITIONAL BRANCH (L45 = 1)**

This command consists of a **G20**, an X value and an F value.

The **G20** command will cause the program line pointer to jump to the program line number given by the F field if the state of the four inputs correspond to the values of the X field.

The acceptable values for the X field are:

- X[n4][n3][n2][n1]
  - n1 = 0 = input 1 inactive
  - n1 = 1 = input 1 active
  - n2 = 0 = input 2 inactive
  - n2 = 1 = input 2 active
  - n3 = 0 = CW limit input (input 3) inactive
  - n3 = 1 = CW limit input (input 3) active
  - n4 = 0 = CCW limit input (input 4) inactive
  - n4 = 1 = CCW limit input (input 4) active

Any value for n1, n2, n3, or n4 from 2 to 9 will signify a "don't care" condition and that input will be ignored.

If the states of the four inputs match the values of n1, n2, n3, and n4, the program will jump to the line number indicated by the F field value (1 TO 400 ONLY).

If the states of the four inputs do not match the programmed values of n1, n2, n3, and n4, the program line pointer is incremented to the next program line.

**G20 OPTIONS:**

- **G20 X0000:** Go to line number (F field) if input 4 off, input 3 off, input 2 off, input 1 off
- **G20 X0001:** Go to line number (F field) if input 4 off, input 3 off, input 2 off, input 1 on
- **G20 X0010:** Go to line number (F field) if input 4 off, input 3 off, input 2 on, input 1 off
- **G20 X0011:** Go to line number (F field) if input 4 off, input 3 off, input 2 on, input 1 on
- **G20 X0100:** Go to line number (F field) if input 4 off, input 3 on, input 2 off, input 1 off
- **G20 X0101:** Go to line number (F field) if input 4 off, input 3 on, input 2 off, input 1 on
- **G20 X0110:** Go to line number (F field) if input 4 off, input 3 on, input 2 on, input 1 off
- **G20 X0111:** Go to line number (F field) if input 4 off, input 3 on, input 2 on, input 1 on
- **G20 X1000:** Go to line number (F field) if input 4 on, input 3 off, input 2 off, input 1 off
- **G20 X1001:** Go to line number (F field) if input 4 on, input 3 off, input 2 off, input 1 on
- **G20 X1010:** Go to line number (F field) if input 4 on, input 3 off, input 2 on, input 1 off
- **G20 X1011:** Go to line number (F field) if input 4 on, input 3 off, input 2 on, input 1 on
- **G20 X1100:** Go to line number (F field) if input 4 on, input 3 on, input 2 off, input 1 off
- **G20 X1101:** Go to line number (F field) if input 4 on, input 3 on, input 2 off, input 1 on
- **G20 X1110:** Go to line number (F field) if input 4 on, input 3 on, input 2 on, input 1 off
- **G20 X1111:** Go to line number (F field) if input 4 on, input 3 on, input 2 on, input 1 on

Any 0 or 1 can be replaced by a "don't care" character (2 through 9). A "don't care" condition causes that particular input to be ignored.

**EXAMPLE:**

- **N100 G20 X2222 F120**
  - Will always cause a jump to line 120 because all inputs signify a "don't care" condition.
- **N100 G20 X1010 F120 CRLF**
  - When executed, will cause a jump to line 120 if 4 is active, input 3 is inactive, input 2 is active, and input 1 is inactive. Otherwise, the program will increment to line 101.

The F field value is truncated to the least four significant digits, and if the F field value is 0 or greater than 400, program execution ceases and the line number is set with the L41 parameter.

**G22 WAIT FOR INPUT (L45 = 0)**

This command consists of a **G22** and X field.

The **G22** command will cause the program to wait until the state of the two inputs correspond to the values of the X field.

The acceptable values for the X field are:

- X[n2][n1]
  - n1 = 0 = input 1 inactive
  - n1 = 1 = input 1 active
  - n2 = 0 = input 2 inactive
  - n2 = 1 = input 2 active

Any value for n1 or n2 from 2 to 9 will signify a "don't care" condition and that input will be ignored.

If the states of input 1 and input 2 match the values of n1 and n2, the program continues execution.

If the states of input 1 and input 2 do not match the programmed values of n1 and n2, the program waits until a match occurs.

- **N100 G22 X10 CRLF**
  - When executed will cause the program to wait until input 2 is active and input 1 is inactive. Then, the program will increment to line 101. The F field is ignored during a **G22** command.

**G22 OPTIONS:**

- **G22 X00:** Wait until input 2 off, input 1 off
- **G22 X01:** Wait until input 2 off, input 1 on
- **G22 X10:** Wait until input 2 on, input 1 off
- **G22 X11:** Wait until input 2 on, input 1 on
- **G22 X0d:** Wait until input 2 off, ignore input 1
- **G22 X1d:** Wait until input 2 on, ignore input 1
- **G22Xd0:** Wait until input 1 off, ignore input 2
- **G22Xd1:** Wait until input 1 on, ignore input 2
- **G22 Xdd:** Don't wait

d = 2 through 9: don't care condition
**G22  WAIT FOR INPUT (L45 = 1)**

This command consists of a **G22** and X field.

The **G22** command will cause the program to wait until the state of the four inputs correspond to the values of the X field.

The acceptable values for the X field are:

- **X [n4] [n3] [n2] [n1]**
- **n1 = 0** input 1 inactive
- **n1 = 1** input 1 active
- **n2 = 0** input 2 inactive
- **n2 = 1** input 2 active
- **n3 = 0** CW limit input (input 3) inactive
- **n3 = 1** CW limit input (input 3) active
- **n4 = 0** CCW limit input (input 4) inactive
- **n4 = 1** CCW limit input (input 4) active

Any value for n1, n2, n3, or n4 from 2 to 9 will signify a “don’t care” condition and that input will be ignored.

If the state of the four inputs **match** the values of n1, n2, n3 and n4, the program continues execution.

If the states of the four inputs **do not match** the programmed values of n1, n2, n3 and n4, the program waits until a match occurs.

**G22 OPTIONS**

- **G22 X0000**: Wait until input 4 off, input 3 off, input 2 off, input 1 on
- **G22 X0001**: Wait until input 4 off, input 3 off, input 2 on, input 1 off
- **G22 X0010**: Wait until input 4 off, input 3 on, input 2 on, input 1 off
- **G22 X0011**: Wait until input 4 off, input 3 on, input 2 on, input 1 on
- **G22 X0100**: Wait until input 4 on, input 3 on, input 2 off, input 1 off
- **G22 X0101**: Wait until input 4 on, input 3 on, input 2 off, input 1 on
- **G22 X0110**: Wait until input 4 on, input 3 on, input 2 on, input 1 off
- **G22 X0111**: Wait until input 4 on, input 3 on, input 2 on, input 1 on
- **G22 X1000**: Wait until input 4 on, input 3 off, input 2 on, input 1 off
- **G22 X1001**: Wait until input 4 on, input 3 off, input 2 on, input 1 on
- **G22 X1010**: Wait until input 4 on, input 3 on, input 2 on, input 1 off
- **G22 X1011**: Wait until input 4 on, input 3 on, input 2 on, input 1 on
- **G22 X1100**: Wait until input 4 on, input 3 on, input 2 off, input 1 off
- **G22 X1101**: Wait until input 4 on, input 3 on, input 2 off, input 1 on
- **G22 X1110**: Wait until input 4 on, input 3 on, input 2 on, input 1 off
- **G22 X1111**: Wait until input 4 on, input 3 on, input 2 on, input 1 on

Any 0 or 1 can be replaced by a “don’t care” character (2 through 9). A “don’t care” condition causes that particular input to be ignored.

**EXAMPLE:**

```
N100 G22 X1010 CRLF
```

When executed will cause the program to wait until input 4 is active, input 3 is inactive, input 2 is active, and input 1 is inactive. Then the program will increment to line 101.

**G30  RETURN FROM SUBROUTINE/PROGRAM END**

This command has two uses.

1. **G30**, used in conjunction with a **G11**, indicates end of a subroutine. A **G30** command that is issued after a **G11** command will cause the program line pointer to be set to the next program line after the line containing the **G11** command.

2. **G30**, used by itself, that is not following a **G11** command, indicates the **end of the program**.

The program line pointer will reset to the program line indicated by the L41 parameter.

In the automatic execution format (L06 n = 2), this will stop program execution. In the continuous execution format (L06 n = 3), program execution will continue from the line number indicated by the L41 parameter.

If no **G30** command is used, program line number 400 is automatically set as the end of the program.

**G31  PROGRAM STOP**

During program execution, the **G31** causes the program to cease executing and the line pointer increments to the next line. A CYCLE START (H1) will continue execution from that point.

**G36  STROBE X CODE DATA**

During program execution, the **G36** command causes the indexer to load the parallel data inputs. The Field Data entry becomes the X field for the program line. The Select and Code Data entries are ignored. The program line’s F Code, if programmed, will be utilized as the active feed rate. Any previously programmed X field on a line with a **G36** command is ignored.

**G37  STROBE N CODE DATA**

During program execution, the **G37** command causes the indexer to load the parallel data inputs. The Code Data entry becomes the active program line as the program branches to that line. The Select and Field Data entries are ignored. Any previously programmed X of F fields on a line with a **G37** command are ignored. Attempting to load a line number of 0 or greater than 400 will cause program execution to cease, and the program line pointer will reset to the L41 value.
This command will set the states of the two programmable outputs according to the value of the last 2 digits of the X field in the following fashion:

\[ G47 \ X^{n2} \ [n1] \]
- \( n1 = 0 \): output 1 off (inactive)
- \( n1 = 1 \): output 1 on (active)
- \( n2 = 0 \): output 2 off (inactive)
- \( n2 = 1 \): output 2 on (active)

If a value of 2 through 9 is used for \( n1 \) or \( n2 \), it will not change the state of the output.

**G47 OPTIONS**
- \( G47 \ X00 = \) Output 2 off, output 1 off
- \( G47 \ X01 = \) Output 2 off, output 1 on
- \( G47 \ X10 = \) Output 2 on, output 1 off
- \( G47 \ X11 = \) Output 2 on, output 1 on
- \( G47 \ X0d = \) Output 2 off, output 1 no change
- \( G47 \ X1d = \) Output 2 on, output 1 no change
- \( G47 \ Xd0 = \) Output 2 no change, output 1 off
- \( G47 \ Xd1 = \) Output 2 no change, output 1 on
- \( G47 \ Xdd = \) Output 2 no change, output 1 no change

\( d = 2 \) through 9, don't care condition

The F field is ignored on a line containing a \( G47 \) command.

* The following four commands apply to 3180 and 6180 series drives only. They are not functional with 230 and 430 series units.

**G64 ENABLE REDUCED CURRENT**
This command will reduce the motor current at standstill to a value set on the drive. If a command for motion is issued after a \( G64 \) command, the motor will run at normal current, and return to a reduced current when motion ceases.

**G65 CANCEL REDUCED CURRENT**
This command cancels the \( G64 \) command. When the indexer is powered up, it is in the CANCEL REDUCED CURRENT mode.

**G66 ENABLE BOOST CURRENT**
This command will increase the motor current during acceleration and deceleration to a value set on the drive. The boost will turn off 5 seconds after application to prevent excessive motor heating during long acceleration/deceleration ramps.

**G67 CANCEL BOOST CURRENT**
This command cancels the \( G66 \) command. When the indexer is powered up, it is in the CANCEL BOOST CURRENT mode.

**G68 ENABLE ALL WINDINGS OFF**
This command sets the average motor current to zero when the motor is at a standstill. Voltage is still present at the motor terminals.

If a command for motion is issued after a \( G68 \) command, the motor will turn at normal current and return to a zero condition when motion ceases.

**G69 CANCEL ALL WINDINGS OFF**
This command cancels a \( G68 \) command. When the indexer is powered up, it is in the CANCEL ALL WINDINGS OFF mode.

**G76 RETURN TO ELECTRICAL HOME**
This command causes the motor to turn until an absolute position of 0 is reached.

Electrical home is established when the indexer is powered up or when a \( H09 \) SET HOME command is used.

The motor will move in the **opposite direction of the absolute position sign** and will index the value contained in the absolute position counter.

**G77 SET ELECTRICAL HOME**
This command sets the absolute position counter to zero. The current motor position is established as electrical home.

**G78 RETURN TO MECHANICAL HOME**
This command causes the motor to turn in the direction set with the \( L08 \) parameter, at the speed set with \( L14 \) until the home limit switch is activated. The motor then offsets from the switch the direction and distance set with the \( L17 \) parameter at the speed set with \( L14 \). At motion completion, the motor position becomes electrical home as the absolute position counter is set to zero.

**G90 ABSOLUTE MODE**
This command sets the indexer to operate in the absolute mode. All moves made by the motor are counted either plus or minus from the zero position (**ELECTRICAL HOME**) set with the \( H09 \) or \( G77 \) command.

**G91 INCREMENTAL MODE**
This command sets the indexer to operate in the incremental mode.

All moves of the motor are counted either plus or minus from the present motor position.

When the indexer is powered up, it is automatically set in the **INCREMENTAL MODE**.

**NOTE:** Even when the indexer is operating in the incremental mode, the **ABSOLUTE POSITION** counter is operational and a **RETURN TO ELECTRICAL HOME** command may be used.
4.6 SAMPLE PROGRAM DESCRIPTION

The following is a sample program which demonstrates some of the capabilities of the Micro Series Indexer. With L06 set to 3, L41 at 1 and L45 at 0, lines 1 through 3 will execute continuously until an input condition matches the branch condition and jumps to the branch line number.

Selecting program segment 1 through input 1 results in the following lines being executed:
20, 21, 22, 23, 24, 25, 26, 27, 28, 29, and back to 1.

Selecting program segment 2 through input 2 results in the following lines being executed:
9, 10, 11, 12, 13, 14, 15, 16, 30, 31, 32, 33, 34, 30, 31, 32, 33, 34, 17, 18, 19, and back to 1.

N001 G20 X + 0000010 F000010 Line 1 is a CONDITIONAL BRANCH. If input 2 is active and input 1 is inactive, the line pointer jumps to line10.

N002 G20 X + 000001 F000020 Line 2 is a CONDITIONAL BRANCH. If input 2 is inactive and input 1 is active, the line pointer jumps to line 20.

N003 G30 Line 3 is a PROGRAM END. The L41 Value (1) is loaded in the line pointer and since L06 is set in the continuous mode (3), the program continues to execute.

N004 Line 4 contains no programming.

N005 Line 5 contains no programming.

N006 Line 6 contains no programming.

N007 Line 7 contains no programming.

N008 Line 8 contains no programming.

N009 G77 Line 9 sets ELECTRICAL HOME with G77

N010 G90 X + 00001000 F0002400 Line 10 sets ABSOLUTE MODE with G90 and the motor moves to position +1000 at a FEEDRATE of 2400 pulses/sec.

N11 G04 X +00002000 Line 11 is a DWELL time of 2 seconds.

N012 X - 00002000 Line 12 moves to position -2000 by moving -3000 pulses at a FEEDRATE of 2400 pulses/sec. (Since no feedrate was programmed, it moved at the last programmed feedrate.)

N013 G66 X + 00010000 F0001200 Line 13 ENABLES BOOST CURRENT with G66 and the motor moves to position +10000 by moving +12000 pulses at a FEEDRATE of 1200 pulses/sec.

N014 G04 X + 00003000 Line 14 is a DWELL of 3 seconds.


N016 G11 X + 00000030 F0000002 Line 16 is a SUBROUTINE CALL to line 30. The subroutine will be executed 3 times.

N017 G76 Line 17 RETURNS HOME (position 0) by moving the necessary pulses at the HOME SPEED (L14).

N018 G47 X + 00000001 Line 18 turns off output 2 and turns output 1 on.

N19 G20 X + 00000022 F0000001 Line 19 is an UNCONDITIONAL BRANCH to Line 1.

N020 G91 X -00002500 F0002400 Line 20 sets INCREMENTAL MODE with G91 and the motor moves -2500 pulses at a FEEDRATE of 2400 pulses/sec.
N021 G64 X + 00000500

Line 13 ENABLES REDUCE CURRENT with G64 and the motor moves +500 pulses at a FEEDRATE of 2400 pulses/sec. (Since no feedrate was programmed, it moved at the last programmed feedrate.)

N022 G04 X + 00004000

Line 22 is a DWELL of 4 seconds.

N23 G65 X + 00005500

Line 23 CANCELS REDUCE CURRENT with G65 and moves the motor +5500 pulses at a FEEDRATE of 2400 pulses/sec. (Since no feedrate was programmed, it moved at the last programmed feedrate.)

N24 G68 X - 00009120 F0002000

Line 24 ENABLES ALL WINDINGS OFF with G68 and the motor moves -9120 pulses at a FEEDRATE of 2000 pulses/sec.

N025 G04 X +00009000

Line 25 is a DWELL of 9 seconds.

N26 G69

Line 26 CANCELS ALL WINDINGS OFF with G69.

N027 G76

Line 27 RETURNS HOME (position 0) by moving the necessary pulses at the HOME SPEED (L14).

N028 G47 X +00000010

Line 28 turns off output 1 and turns output 2 on.

N029 G20 X + 00000022 F0000001

Line 29 is an UNCONDITIONAL BRANCH to Line 1, effectively ending the second program segment.

N030 X - 00000100 F0000800

Line 30 moves the motor to position -100 at a FEEDRATE of 800 pulses/sec.

N031 G04 X + 00005000

Line 31 is a DWELL of 5 seconds.

N032 X + 00000500 F0001500

Line 32 moves the motor to position +500 at a FEEDRATE of 1500 pulses/sec.

N033 G04 X + 00003500

Line 33 is a DWELL of 3.5 seconds.

N034 G30

Line 34 is a SUBROUTINE END.

4.7 CODE ASSIGNMENT TABLES

PARAMETERS

<table>
<thead>
<tr>
<th>code</th>
<th>field data</th>
<th>function</th>
</tr>
</thead>
<tbody>
<tr>
<td>L06</td>
<td>n</td>
<td>Execute Format (1, 2, 3)</td>
</tr>
<tr>
<td>L07</td>
<td>nnnn</td>
<td>Strobe delay Time (milliseconds)</td>
</tr>
<tr>
<td>L08</td>
<td>s</td>
<td>Mechanical Home Direction (+, -)</td>
</tr>
<tr>
<td>L09</td>
<td>nnnnnnn</td>
<td>Jog Speed (pulse/sec)</td>
</tr>
<tr>
<td>L11</td>
<td>nnnnnnn</td>
<td>Acceleration/Deceleration (pulses/sec/sec)</td>
</tr>
<tr>
<td>L12</td>
<td>nnnnnnn</td>
<td>Low Speed (pulses/sec)</td>
</tr>
<tr>
<td>L14</td>
<td>nnnnnnn</td>
<td>Home Speed (pulses/sec)</td>
</tr>
<tr>
<td>L17</td>
<td>snnnnnnnn</td>
<td>Offset Distance and Direction from Home (pulses)</td>
</tr>
<tr>
<td>L18</td>
<td>snnnnnnnn</td>
<td>CW software Travel Limit (pulses)</td>
</tr>
<tr>
<td>L19</td>
<td>snnnnnnnn</td>
<td>CCW software Travel Limit (pulses)</td>
</tr>
<tr>
<td>L21</td>
<td>nn</td>
<td>RS232 Device ID (01 thru 99)</td>
</tr>
<tr>
<td>L22</td>
<td>nnn</td>
<td>RS232 Baud Rate (300, 1200, 2400, 9600)</td>
</tr>
<tr>
<td>L23</td>
<td>n</td>
<td>RS232 Word Length (7, 8)</td>
</tr>
<tr>
<td>L25</td>
<td>n</td>
<td>RS232 Parity (1, 2, 3)</td>
</tr>
<tr>
<td>L26</td>
<td>n</td>
<td>Indexer Ready Acknowledge (0 - 7)</td>
</tr>
<tr>
<td>L41</td>
<td>nnn</td>
<td>Line Number for Auto Start (00 through 400)</td>
</tr>
<tr>
<td>L44</td>
<td>nnn</td>
<td>Program Line Delay (milliseconds)</td>
</tr>
<tr>
<td>L45</td>
<td>n</td>
<td>Limit Switch Enable (0, 1)</td>
</tr>
<tr>
<td>L48</td>
<td>nnn</td>
<td>Program Line Count</td>
</tr>
<tr>
<td>L49</td>
<td>nn</td>
<td>Parameter Transfer Designation</td>
</tr>
<tr>
<td>L70</td>
<td>nnn</td>
<td>Translator Resolution (1, 2, 5, 10, 125)</td>
</tr>
</tbody>
</table>
### COMMANDS

<table>
<thead>
<tr>
<th>code</th>
<th>type</th>
<th>function</th>
</tr>
</thead>
<tbody>
<tr>
<td>H01</td>
<td>motion</td>
<td>Cycle Start</td>
</tr>
<tr>
<td>H06</td>
<td>motion</td>
<td>CW Direction for STEP or JOG mode</td>
</tr>
<tr>
<td>H07</td>
<td>motion</td>
<td>CCW Direction for STEP or JOG mode</td>
</tr>
<tr>
<td>H08</td>
<td>motion</td>
<td>Return to Electrical Home</td>
</tr>
<tr>
<td>H09</td>
<td>motion</td>
<td>Set Home</td>
</tr>
<tr>
<td>H10</td>
<td>motion</td>
<td>Return to Mechanical Home</td>
</tr>
<tr>
<td>*</td>
<td>motion</td>
<td>Clear (uncontrolled stop)</td>
</tr>
<tr>
<td>$</td>
<td>motion</td>
<td>Feed hold (controlled stop)</td>
</tr>
<tr>
<td>#</td>
<td>motion</td>
<td>Stop Program Execution Cycle</td>
</tr>
<tr>
<td>H02</td>
<td>mode</td>
<td>set Step Mode</td>
</tr>
<tr>
<td>H03</td>
<td>mode</td>
<td>set Jog Mode</td>
</tr>
<tr>
<td>H04</td>
<td>mode</td>
<td>set High Speed Mode operation</td>
</tr>
<tr>
<td>H05</td>
<td>mode</td>
<td>set Low Speed Mode of operation</td>
</tr>
<tr>
<td>H11</td>
<td>edit</td>
<td>Clear present Program Line</td>
</tr>
<tr>
<td>H12</td>
<td>edit</td>
<td>Clear Program</td>
</tr>
<tr>
<td>&lt;nn</td>
<td>edit</td>
<td>Device Attention Character</td>
</tr>
<tr>
<td>Cntl H</td>
<td>edit</td>
<td>Back Space</td>
</tr>
<tr>
<td>Cntl X</td>
<td>edit</td>
<td>Cancel Line</td>
</tr>
<tr>
<td>H13</td>
<td>transfer</td>
<td>transfer Present Program Line</td>
</tr>
<tr>
<td>H14</td>
<td>transfer</td>
<td>transfer Program</td>
</tr>
<tr>
<td>H15</td>
<td>transfer</td>
<td>transfer Present Line Number (see note)</td>
</tr>
<tr>
<td>H16</td>
<td>transfer</td>
<td>transfer Parameters</td>
</tr>
<tr>
<td>H17</td>
<td>transfer</td>
<td>transfer Absolute Position (see note)</td>
</tr>
<tr>
<td>H18</td>
<td>transfer</td>
<td>transfer Motion Status (see note)</td>
</tr>
<tr>
<td>H19</td>
<td>transfer</td>
<td>transfer Mode Status (see note)</td>
</tr>
<tr>
<td>H20</td>
<td>transfer</td>
<td>transfer I/O Status (see note)</td>
</tr>
<tr>
<td>H23</td>
<td>transfer</td>
<td>transfer Software Revision Date</td>
</tr>
<tr>
<td>H24</td>
<td>mode</td>
<td>enable Trace Mode</td>
</tr>
<tr>
<td>H25</td>
<td>mode</td>
<td>disable Trace Mode</td>
</tr>
</tbody>
</table>

**NOTE:** H15, H17, H18, H19, and H20 can be executed during motion or program execution, and also while in parallel mode, to obtain "on the fly" information.

### PROGRAM DATA

<table>
<thead>
<tr>
<th>code</th>
<th>field data</th>
<th>function</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>nnn</td>
<td>Line Number (000 thru 400)</td>
</tr>
<tr>
<td>X</td>
<td>snnnnnnnnn</td>
<td>Move Distance or Data field for G04, G11, G20, G22, G47 codes</td>
</tr>
<tr>
<td>F</td>
<td>nnnnnnn</td>
<td>Feed Rate or Data field for G11, G20 codes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>code</th>
<th>function</th>
<th>X field data</th>
<th>F field data</th>
</tr>
</thead>
<tbody>
<tr>
<td>G04</td>
<td>Program Dwell Time</td>
<td>nnnnn Dwell in milliseconds</td>
<td></td>
</tr>
<tr>
<td>G11</td>
<td>Subroutine call</td>
<td>nnn Line Number of Subroutine</td>
<td></td>
</tr>
<tr>
<td>G20</td>
<td>Conditional Branch</td>
<td>(nn)nn Input condition for Branch</td>
<td></td>
</tr>
<tr>
<td>G22</td>
<td>Wait for Input</td>
<td>(nn)nn Input Wait Condition</td>
<td></td>
</tr>
<tr>
<td>G30</td>
<td>Return from Subroutine/Program End</td>
<td></td>
<td></td>
</tr>
<tr>
<td>G31</td>
<td>Program Stop</td>
<td></td>
<td></td>
</tr>
<tr>
<td>G36</td>
<td>Strobe X Code Data</td>
<td></td>
<td></td>
</tr>
<tr>
<td>G37</td>
<td>Strobe N Code Data</td>
<td></td>
<td></td>
</tr>
<tr>
<td>G47</td>
<td>Set/Reset Output Condition</td>
<td></td>
<td></td>
</tr>
<tr>
<td>G64</td>
<td>Enable Reduce Current Mode</td>
<td></td>
<td></td>
</tr>
<tr>
<td>G65</td>
<td>Cancel Reduce Current Mode</td>
<td></td>
<td></td>
</tr>
<tr>
<td>G66</td>
<td>Enable Boost Current Mode</td>
<td></td>
<td></td>
</tr>
<tr>
<td>G67</td>
<td>Cancel Boost Current Mode</td>
<td></td>
<td></td>
</tr>
<tr>
<td>G68</td>
<td>Enable All Windings Off Mode</td>
<td></td>
<td></td>
</tr>
<tr>
<td>G69</td>
<td>Cancel All Windings Off Mode</td>
<td></td>
<td></td>
</tr>
<tr>
<td>G76</td>
<td>Return To Electrical Home</td>
<td></td>
<td></td>
</tr>
<tr>
<td>G77</td>
<td>Set Electrical Home</td>
<td></td>
<td></td>
</tr>
<tr>
<td>G78</td>
<td>Return to Mechanical Home</td>
<td></td>
<td></td>
</tr>
<tr>
<td>G90</td>
<td>Set Absolute Motion Mode</td>
<td></td>
<td></td>
</tr>
<tr>
<td>G91</td>
<td>Set Incremental Motion Mode</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
SECTION 5: OPERATING INSTRUCTIONS

5.1 OVERVIEW

The Micro-Indexer allows for program entry and execution from a variety of sources. The options are:

- Operation from a user-made switch panel.
- Operation from Superior Electric switch panel SSP-100. These options are discussed in Section 5.2.
- Operation from a Superior Electric SSP-500 hand-held intelligent terminal.
- Operation from a user supplied remote terminal. These options are covered in Section 5.3.
- Operation from a host computer. This option is covered in Section 5.4.

5.1.1 Instructions for Entering Data

5.1.1.1 General Instructions

1. Lines may be programmed in any order but are executed in numerical sequence.

2. Variables may be programmed or reprogrammed in any order.

3. Program area should be cleared prior to new program entry to avoid execution of previously programmed lines.

4. If feed rate is not programmed for a given line, the last programmed feed rate is used.

5. Selection of invalid data will result in the previous data being left intact.

5.2 OPERATION FROM A SWITCH PANEL

5.2.1 Overview

Because of the nature of the parallel interface, all indexer functions can be controlled by the connection of one of 8 STROBE pins to one of 8 DATA pins. A matrix diagram of the STROBE/DATA functions is given in Figure 5.1.

Referring to this figure, it can be seen that if the STROBE 0 line is connected to the DATA 2 line, a Home Limit condition will be indicated.

All STROBE/DATA functions are controlled by either STROBE 0 or STROBE 1. These two STROBE lines are polled by the indexer every several milliseconds, so that any contact closure between STROBE 0 or STROBE 1 and any of the DATA lines will be detected by the indexer within a maximum of 12 milliseconds. Figure 5.3 shows a strobe signal timing diagram.

The other STROBES listed in Figure 5.1 are used to enter programs and parameters. After switches are set, the LOAD input must be activated for data entry to be executed (LOAD is a connection between STROBE 1 and DATA 2).

<table>
<thead>
<tr>
<th>INPUT</th>
<th>STROBE 0 MOTION</th>
<th>STROBE 1 MOTION</th>
<th>STROBE 2 MOTION</th>
<th>STROBE 3 MOTION</th>
<th>STROBE 4 MOTION</th>
<th>STROBE 5 MOTION</th>
<th>STROBE 6 MOTION</th>
<th>STROBE 7 MOTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>D0</td>
<td>CW LIMIT-</td>
<td>CW DIR-</td>
<td>CODE 1-</td>
<td>CODE100-</td>
<td>DATA 1M-</td>
<td>DATA 10K-</td>
<td>DATA 100-</td>
<td>DATA 1-</td>
</tr>
<tr>
<td>D1</td>
<td>CCW LIMIT-</td>
<td>CCW DIR-</td>
<td>CODE 2-</td>
<td>CODE 200-</td>
<td>DATA 2M-</td>
<td>DATA 20K-</td>
<td>DATA 200-</td>
<td>DATA 2-</td>
</tr>
<tr>
<td>D2</td>
<td>HOME LIMIT-</td>
<td>LOAD-</td>
<td>CODE 4-</td>
<td>CODE 400-</td>
<td>DATA 4M-</td>
<td>DATA 40K-</td>
<td>DATA 400-</td>
<td>DATA 4-</td>
</tr>
<tr>
<td>D3</td>
<td>CLEAR CYCLE-</td>
<td>AWO-</td>
<td>CODE 8-</td>
<td>CODE 800-</td>
<td>DATA 8M-</td>
<td>DATA 80K-</td>
<td>DATA 800-</td>
<td>DATA 8-</td>
</tr>
<tr>
<td>D4</td>
<td>FEED HOLD-</td>
<td>LOW/HIGH-</td>
<td>CODE 10-</td>
<td>SIGN-</td>
<td>DATA 10M-</td>
<td>DATA 100K-</td>
<td>DATA 1K-</td>
<td>DATA 10-</td>
</tr>
<tr>
<td>D5</td>
<td>IN 1-</td>
<td>STEP/JOG-</td>
<td>CODE 20-</td>
<td>SEL 1-</td>
<td>DATA 20M-</td>
<td>DATA 200K-</td>
<td>DATA 2K-</td>
<td>DATA 20-</td>
</tr>
<tr>
<td>D6</td>
<td>IN 2-</td>
<td>CYCLE START-</td>
<td>CODE 40-</td>
<td>SEL 2-</td>
<td>DATA 40M-</td>
<td>DATA 400K-</td>
<td>DATA 4K-</td>
<td>DATA 40-</td>
</tr>
<tr>
<td>D7</td>
<td>STOP CYCLE-</td>
<td>SER/PAR-</td>
<td>CODE 80-</td>
<td>SEL 4-</td>
<td>DATA 80M-</td>
<td>DATA 800K-</td>
<td>DATA 8K-</td>
<td>DATA 80-</td>
</tr>
</tbody>
</table>

NOTE: All signals are low true logic except LOW, STEP and SER.
PARALLEL INPUT EVALUATION
Due to necessary timing restrictions, not all inputs are monitored at all times. Below is a description of which inputs are active and when.
Inputs must be active for a minimum of 12 milliseconds to be considered valid.

<table>
<thead>
<tr>
<th>INPUT</th>
<th>SERIAL MODE</th>
<th>PARALLEL MODE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PROGRAM EXECUTION</td>
<td>MANUAL MOTION</td>
</tr>
<tr>
<td>CW LIMIT</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>CCW LIMIT</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>HOME LIMIT</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>CLEAR</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>FEEDHOLD</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>CW DIRECTION</td>
<td></td>
<td>•</td>
</tr>
<tr>
<td>CCW DIRECTION</td>
<td></td>
<td>•</td>
</tr>
<tr>
<td>CYCLE STOP</td>
<td></td>
<td>•</td>
</tr>
<tr>
<td>INPUT 1</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>INPUT 2</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>LOAD</td>
<td></td>
<td>•</td>
</tr>
<tr>
<td>AWO</td>
<td></td>
<td>•</td>
</tr>
<tr>
<td>LOW/HIGH</td>
<td></td>
<td>•</td>
</tr>
<tr>
<td>STEP/JOG</td>
<td></td>
<td>•</td>
</tr>
<tr>
<td>CYCLE START</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>SERIAL/PARALLEL</td>
<td></td>
<td>•</td>
</tr>
<tr>
<td>CODE 1-CODE 800</td>
<td></td>
<td>•</td>
</tr>
<tr>
<td>DATA 1-DATA 80M</td>
<td></td>
<td>•</td>
</tr>
<tr>
<td>SIGN</td>
<td></td>
<td>•</td>
</tr>
<tr>
<td>SEL 1, SEL 2, SEL 4</td>
<td></td>
<td>•</td>
</tr>
</tbody>
</table>

5.2.2 Data Entry Switch

The schematic diagram shown in Figure 5.2 gives the wiring details for the construction of a Parallel Data Entry Interface.

**NOTE:** Implementation of steering diodes is essential to insure proper operation.

The SSP-100, a fully functional parallel interface data entry switch panel, can be ordered from Superior Electric.

The following table gives a summary of the data entry format.

<table>
<thead>
<tr>
<th>FUNCTIONAL SELECT</th>
<th>CODE DATA THUMBWHEEL</th>
<th>FIELD DATA THUMBWHEEL</th>
</tr>
</thead>
<tbody>
<tr>
<td>N=Line No.</td>
<td>Line Number</td>
<td>Not Used</td>
</tr>
<tr>
<td>G Code</td>
<td>G Code Selection</td>
<td>Not Used</td>
</tr>
<tr>
<td>X Code</td>
<td>Not Used</td>
<td>X Field Data</td>
</tr>
<tr>
<td>F Code</td>
<td>Not Used</td>
<td>F Field Data</td>
</tr>
<tr>
<td>L Code</td>
<td>L Code Selection</td>
<td>L Field Data</td>
</tr>
<tr>
<td>H Code</td>
<td>H Code Selection</td>
<td>Not Used</td>
</tr>
</tbody>
</table>
Figure 5.2, Switch Panel Connection
5.2.3 Description Of Hardware Inputs

CODE DATA THUMBWHEEL  Input for entry of line number, L, H or G code

FIELD DATA THUMBWHEEL  Input for entry of data associated with the X field, F field or L field data

CW LIMIT-  Input to cause motion to come to an uncontrolled immediate stop if moving in the CW direction. Motion is only allowed in the CCW direction until the CW Limit- is inactive. Line pointer is reset to L41 value, serial buffer is cleared and XOFF and then XON are sent.

CCW LIMIT-  Input to cause motion to come to an uncontrolled immediate stop if moving in the CCW direction. Motion is only allowed in the CW direction until the CCW Limit- is inactive. Line pointer is reset to L41 value, serial buffer is cleared and XOFF and then XON are sent.

CLEAR-  Input to cause motion to come to an uncontrolled immediate stop. Position is lost, no motion is allowed while CLEAR- is active. Resets line pointer to L41 value, clears the serial buffer and XOFF and then XON are sent.

FEEDHOLD-  Input to cause motion to come to a controlled stop (pause) with no loss of position.

HOME LIMIT-  Input to indicate mechanical home position.

CW DIR-  Input to cause motor motion in the CW direction. If STEP was selected, motor will single-step each time this input is activated. If JOG- is selected, motor will turn continuously while input is active.

CCW DIR-  Input to cause motor motion in the CCW direction. If STEP was selected, motor will single-step each time this input is activated. If JOG- is selected, motor will turn continuously while input is active.

CYCLE STOP-  Program execution is stopped after the present line is executed.

LOAD-  Input to enter data selected on CODE DATA thumbwheel and corresponding data on FIELD DATA thumbwheel.

AWO-  Input to turn windings off when the motor is at a standstill.

LOW/HIGH-  Input to determine speed range limit. If HIGH- is selected, motor will be allowed to ramp to selected speed. If LOW is selected, motor will not be allowed to ramp and will run at selected low speed.

STEP/JOG-  Input to determine if motor will single-step or jog (continuous motion) when manual motion is called for.

CYCLE START-  Input to start execution of program or to continue motion if a FEED HOLD has been executed.

SERIAL/PARALLEL-  Input to determine if operation is via SERIAL (RS232) or PARALLEL inputs. If SERIAL mode is selected, the following parallel inputs are ignored. CW DIR-, CCW DIR-, LOW/HIGH-, AWO-, STEP/JOG-, LOAD-. Changing this input automatically invokes a CLEAR command.

FUNCTION SELECT inputs SEL 4-, SEL 2- and SEL 1- are used in combination to select the variable to be programmed on a selected line.

<table>
<thead>
<tr>
<th>SEL 4-</th>
<th>SEL 2-</th>
<th>SEL 1-</th>
<th>Function Selected</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>N Line Number</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>G Code</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>X Code</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>F Code</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>L Code</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>H Code</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>Not Used</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>Not Used</td>
</tr>
</tbody>
</table>

NOTE: 0=high input

1=low input

Minus sign (-) indicates signal is active when low.
5.2.4 PARALLEL DATA LOADING

1) Parallel mode must be selected
2) Program execution and manual modes must be stopped
3) The proper data must be selected via the function select switches, code data thumbwheels and field data thumbwheels.
4) The load input must be activated.

EXPLANATION OF HOW INDEXER LOADS PARALLEL DATA

1) Strobe 0 and Strobe 1 are disabled.
2) Strobe 2 is activated for the delay time indicated by the L07 parameter.
3) At the end of the delay period the data is read and the strobe deactivated.
4) Strobe 3 is activated for the delay time indicated by the L07 parameter.
5) At the end of the delay period the data is read and the strobe deactivated.
6) Strobe 4 is activated for the delay time indicated by the L07 parameter.
7) At the end of the delay period the data is read and the strobe deactivated.
8) Strobe 5 is activated for the delay time indicated by the L07 parameter.
9) At the end of the delay period the data is read and the strobe deactivated.
10) Strobe 6 is activated for the delay time indicated by the L07 parameter.
11) At the end of the delay period the data is read and the strobe deactivated.
12) Strobe 7 is activated for the delay time indicated by the L07 parameter.
13) At the end of the delay period the data is read and the strobe deactivated.
14) An additional time delay indicated by the L07 parameter is executed to allow sufficient time to remove the data prior to Strobe 0 and Strobe 1 actuation.
15) The command is then executed.

5.2.4.1 Data Entry By Letter Code

Minus sign (-) indicates signal is active when low.

Data Entry N:
1. Select N code via SEL 4-, SEL 2- and SEL 1- inputs.
2. Select desired line number on code data thumbwheel.
4. Code data thumbwheel value is then strobbed in.
5. This becomes the active program line until new line number is selected.
6. Field data thumbwheel value is ignored.

Data Entry G:
1. Select G code via SEL 4-, SEL 2- and SEL 1- inputs.
2. Select desired G code on code data thumbwheel.
4. Code data thumbwheel value is then strobbed in.
5. G code is placed in active program line.
6. Field data thumbwheel value is ignored.

Data Entry X:
1. Select X code via SEL 4-, SEL 2- and SEL 1- inputs.
2. Select desired X value on field data thumbwheel.
4. Field data thumbwheel value is then strobbed in.
5. Value is placed in active program line X field.
6. Code data thumbwheel value is ignored.

Data Entry F:
1. Select F code via SEL 4-, SEL 2- and SEL 1- inputs.
2. Select desired value on the field data thumbwheel.
4. Field data thumbwheel value is then strobbed in.
5. Value is placed in active program line F field.
6. Code data thumbwheel is ignored.
Data Entry L:
1. Select L code via SEL 4-, SEL 2- and SEL 1- inputs.
2. Select desired L code data on code data thumbwheel.
3. Select associated value on field data thumbwheel.
5. Code data thumbwheel value is then strobed in.
6. Field data thumbwheel value is then strobed in.
7. L code is entered.

Data Entry H:
1. Select H code via SEL 4-, SEL 2- and SEL 1- inputs.
2. Select desired H code on code data thumbwheel.
4. Code data thumbwheel value is then strobed in.
5. H code is executed.
6. Field data thumbwheel is ignored.

NOTE: The following H codes cannot be executed while in parallel mode: H2, H3, H4, H5, H6 and H7.

5.2.4.2 Execution
Manual motion is possible whenever a program is not being executed. The STEPJOG- input and the LOW/HIGH- input will determine the response to the CW DIR- and CCW DIR- inputs becoming active. The absolute position counter is maintained.

Program Execution is accomplished by activating the CYCLE START- input and releasing. The program then executes based on the L parameters.

5.2.4.3 Outputs Accessed Through The External Parallel I/O Port
These outputs are open collectors. They must be pulled up to an external voltage source (maximum +24Vdc) through a series resistor to limit the sink current to a maximum of 40 milliamperes.

<table>
<thead>
<tr>
<th>Output</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OUTPUT 1-</td>
<td>Programmable outputs controlled through use of G47 code.</td>
</tr>
<tr>
<td>OUTPUT 2-</td>
<td></td>
</tr>
<tr>
<td>ALL WINDINGS OFF OUT-</td>
<td>Output indicating the status of the ALL WINDINGS OFF- mode.</td>
</tr>
<tr>
<td>CW/CCW OUT-</td>
<td>Output indicating the direction of motor motion.</td>
</tr>
<tr>
<td>MOTION BUSY OUT-</td>
<td>Output indicating when any type of motion is occurring.</td>
</tr>
<tr>
<td>PULSE OUT-</td>
<td>Output of the pulses being applied to the drive.</td>
</tr>
</tbody>
</table>

5.3 OPERATION FROM A REMOTE TERMINAL

5.3.1 Overview
This section contains information pertaining to operating the indexer from a remote terminal via the RS232 serial port.

5.3.2 Connections and Interface
The 9-pin serial interface connector can be used to connect a remote terminal to the indexer for programming and program execution.

The SSP-500, a hand-held intelligent terminal available from Superior Electric, can be used for debugging, loading and executing indexer programs.

When a remote terminal is used with the indexer, the baud rate must be set correctly so that the indexer will match the terminal. This parameter is set with the L22, L23 and L25 commands. See the Command Description section for the applicable values and defaults.

NOTE: The baud rate, word length and default settings are correct for operation with the Superior Electric hand-held remote terminal SSP-500.

If another type of remote terminal is used which has a baud rate other than 9600 baud, the terminal will not be able to communicate with the terminal. A compatible terminal will have to be used to set the baud rate or, alternatively, a switch panel can be used to set the baud rate. (See Section 5.2.1 for a description of the switch panel operation).

5.4 OPERATION FROM A HOST COMPUTER
The hardware connections for a host computer are the same as those for a remote terminal. The connection is made through the 9-pin serial interface connector.

An IBM compatible program (on a 5.25" floppy disk) that has editing, manual motion and program execution modes of operation to facilitate Micro Series indexer functions is available from Superior Electric. Local and remote modes allow programs to be developed off-line in the IBM PC and later loaded into the indexer.

To interface with an IBM PC, the connections must be configured as follows:

<table>
<thead>
<tr>
<th>IBM (25-pin &quot;D&quot; Connector)</th>
<th>Positioner (9-pin &quot;D&quot; Connector)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pin Number</td>
<td>Pin Number</td>
</tr>
<tr>
<td>2</td>
<td>3 Receive</td>
</tr>
<tr>
<td>3</td>
<td>2 Chain Out</td>
</tr>
<tr>
<td>7</td>
<td>4 Signal Common</td>
</tr>
</tbody>
</table>

Following are three sample IBM PC compatible programs to display the ease of indexer/host computer software interfacing. Program 1 uploads an indexer program to the PC disk. Program 2 downloads a program from the PC disk to the indexer; and program 3 sets up interactive communications between the PC and the indexer. Enter BASIC with the following:

BASICA/C:4096
PROGRAM LISTING 1

10 REM THIS PROGRAM TAKES THE APPLICATION PROGRAM THAT IS IN
20 REM THE INDEXER AND STORES IT ON A DISK.
25 REM PROGRAM DOES NOT SUPPORT DAISY CHAINING
30 REM CONFIGURE SERIAL PORT 1 FOR INDEXER COMMUNICATIONS
40 REM INDEXER SET AT 9600 BAUD, 8 DATA BITS, 1 STOP BIT, AND
50 REM NO PARITY
60 OPEN "COM1:9600, N,8,2,CS,DS,CD" AS#1
70 INPUT "ENTER DEVICE NUMBER",D
80 REM OUTPUT DEVICE CHARACTER AND DEVICE NUMBER
90 IF D>99 OR D<1 THEN GO TO 70
100 REM
110 IF D<10 THEN PRINT #1 USING "<##",D
120 IF D>9 THEN PRINT #1 USING "##",D
130 REM RECEIVE EQUAL SIGN AND XON
140 G$=INPUT$(2,#1)
150 REM SET L26 PARAMETER TO 0
160 PRINT #1, "L26 0"
170 REM RECEIVE XOFF AND XON
180 G$=INPUT$(2,#1)
190 REM INPUT FILENAME TO WHICH THE INDEXER PROGRAM WILL BE STORED
200 INPUT "ENTER FILENAME TO STORE INDEXER PROGRAM",F$
210 REM OPEN THE DISK FILE TO RECEIVE THE INDEXER PROGRAM
220 OPEN "O", #2, F$
230 REM INITIALIZE LINE COUNT
240 C=0
250 REM SEND PROGRAM TRANSFER COMMAND TO INDEXER
260 PRINT #1, "L48 0 H14"
270 REM INPUT XOFF
280 G$=INPUT$(1,#1)
290 REM RECEIVE PROGRAM LINE
300 INPUT #1, L$
310 REM SEND XOFF TO ALLOW TIME TO STORE DATA ON DISK
320 PRINT #1, CHR$(19);
330 REM STORE DATA ON DISK
340 PRINT #2, L$
350 REM DISPLAY PROGRAM LINE ON SCREEN
360 PRINT L$
370 REM INCREMENT LINE COUNT
380 C=C+1
390 REM SEND XON TO ALLOW TRANSMISSION TO CONTINUE
400 PRINT #1, CHR$(17);
410 REM GET NEXT LINE IF 400 LINES HAVE NOT BEEN RECEIVED
420 IF C<400 THEN GO TO 300
430 REM DATA TRANSFER AND STORAGE COMPLETE
440 PRINT "INDEXER PROGRAM STORED IN FILE"; F$
PROGRAM LISTING 2

10 REM THIS PROGRAM LOADS THE INDEXER WITH AN APPLICATION PROGRAM
20 REM THAT HAS BEEN STORED ON A DISK
25 REM PROGRAM DOES NOT SUPPORT DAISY CHAINING
30 REM CONFIGURE SERIAL PORT 1 FOR INDEXER COMMUNICATIONS
40 REM INDEXER SET AT 9600 BAUD, 8 DATA BITS, 2 STOP BITS AND
50 REM NO PARITY
60 OPEN "COM1:9600,N,8,2,CS,DS,CD" AS #1
80 REM OUTPUT XON
90 PRINT #1,CHR$(17);
100 IF D>99 OR D<1 THEN GO TO 70
120 REM
130 IF D<10 THEN PRINT #1, USING "<#;D"
140 IF D>9 THEN PRINT #1, USING "<##"; D
150 REM RECEIVE EQUAL SIGN, AND XON
160 G$=INPUT$(2,#1)
170 REM FILENAME FROM WHICH THE INDEXER WILL BE LOADED
180 INPUT "ENTER FILENAME TO LOAD INDEXER" ;F$
200 REM OPEN THE DISK FILE TO LOAD INDEXER
220 OPEN #2, F$
240 REM INITIALIZE LINE COUNT
260 C=0
270 REM READ PROGRAM LINE FROM DISK
280 INPUT #2, LS
290 REM SEND PROGRAM LINE TO INDEXER
300 PRINT #1, LS
310 REM INPUT XOFF AND XON
320 GS=INPUT$(2,#1)
330 REM FILENAME FROM WHICH THE INDEXER WILL BE LOADED
340 PRINT LS
350 REM INCREMENT LINE COUNT
360 C=C+1
370 REM GET NEXT LINE IF 400 LINES HAVE NOT BEEN TRANSFERRED
380 IF C<400 THEN GO TO 280
390 REM DATA TRANSFER AND STORAGE COMPLETE
400 PRINT "INDEXER LOADED WITH FILE";F$
410 REM CLOSE THE OPEN FILES
420 CLOSE
430 REM PROGRAM END
440 END

PROGRAM LISTING 3

10 REM THIS PROGRAM ALLOWS INTERACTIVE COMMUNICATIONS BETWEEN THE
20 REM HOST AND THE INDEXER
30 REM CONFIGURE SERIAL PORT 1 FOR INDEXER COMMUNICATIONS
40 REM INDEXER SET AT 9600 BAUD, 8 DATA BITS, 2 STOP BITS AND
50 REM NO PARITY
60 OPEN "COM1:9600,N,8,2,CS,DS,CD" AS #1
70 PRINT #1, CHR$(17);
80 REM SET L26 PARAMETER IN ALL INDEXERS
90 PRINT #1,"<00 L26 0"
120 REM ENTER INDEXER COMMAND STRING
130 REM IF END IS ENTERED THEN PROGRAM IS TERMINATED
140 INPUT "ENTER INDEXER COMMAND";L$
150 IF L$= "END" THEN GO TO 350
160 PRINT #1,CHR$(17);
170 REM OUTPUT INDEXER COMMAND STRING
180 PRINT #1,L$
200 FOR I=1 TO 25
210 NEXT I
220 REM READ INPUT BUFFER FOR DATA
230 IF EOF(I)=-1 THEN GO TO 140
240 REM DO NOT PRINT CHARACTER IF XOFF OR XON
250 PRINT #1,
260 B$=INPUT$(I, #1)
270 IF B$=CHR$(17) THEN GO TO 230
280 IF B$=CHR$(19) THEN GO TO 230
290 REM
300 IF B$=CHR$(10) THEN PRINT B$;: GOSUB 360: GO TO 230
310 REM PRINT INPUT CHARACTER ON DISPLAY
320 PRINT B$;
330 REM PROGRAM END
350 END
360 REM THIS SUBROUTINE IS CALLED IF A MULTI-LINE TRANSFER HAS BEEN COMMANDED. IT UTILIZES XON AND XO
370 REM IF EOF(1)= -1 THEN RETURN
380 REM OUTPUT XO
390 REM PROGRAM END
400 PRINT #1,CHR$(19);
410 REM INPUT CHARACTER
420 BS=INPUT$(1, #1)
430 REM DO NOT PRINT IF XO OR XON
440 IF BS=CHR$(17) THEN GO TO 230
450 REM
460 REM
470 REM PRINT INPUT CHARACTER ON DISPLAY
480 PRINT BS;
490 IF EOF(1)=0 THEN GO TO 420
500 REM OUTPUT XON
510 PRINT #1,CHR$(17);
520 RETURN
SECTION 6  SPEED/TORQUE CURVES

TYPICAL SPEED VS. TORQUE CHARACTERISTICS

230 SERIES MOTION CONTROLS

TYPICAL PERFORMANCE CHARACTERISTICS

SERIES CONNECTION
M061-CS08 AND M061-LS08 MOTORS

PARALLEL CONNECTION
M061-CE08 AND M061-LE08 MOTORS

SERIES CONNECTION
M062-CS09 AND M062-LS09 MOTORS
230 SERIES MOTION CONTROLS

(Continued)

TYPICAL PERFORMANCE CHARACTERISTICS

PARALLEL CONNECTION
M062-CE09 AND M062-LE09 MOTORS

SERIES CONNECTION
M063-CS09 AND M063-LS09 MOTORS

PARALLEL CONNECTION
M063-CE09 AND M063-LE09 MOTORS
230 SERIES MOTION CONTROLS
(Continued)

TYPICAL PERFORMANCE CHARACTERISTICS

SERIES CONNECTION
M063-CS06 AND M063-LS06 MOTORS

TYPICAL PERFORMANCE CHARACTERISTICS

SERIES CONNECTION
M091-FC09 AND M091-FD09 MOTORS

TYPICAL PERFORMANCE CHARACTERISTICS

PARALLEL CONNECTION
M091-FD8009 AND M091-FD8109 MOTORS
TYPICAL PERFORMANCE CHARACTERISTICS

SERIES CONNECTION
M092-FC09 AND M092-FD09 MOTORS

PARALLEL CONNECTION
M092-FD8109 AND M092-FD8009 MOTORS

SERIES CONNECTION
M092-FD310 MOTORS
TYPICAL PERFORMANCE CHARACTERISTICS

SPEED (1.8° STEPS PER SECOND)
PARALLEL CONNECTION
M092-FD8814 MOTOR

TYPICAL SPEED VS. TORQUE CHARACTERISTICS
430 SERIES MOTION CONTROLS

TYPICAL PERFORMANCE CHARACTERISTICS

SPEED (1.8° STEPS PER SECOND)
SERIES CONNECTION
M061-CS08 AND M061-LS08 MOTORS

TYPICAL PERFORMANCE CHARACTERISTICS

SPEED (1.8° STEPS PER SECOND)
PARALLEL CONNECTION
M061-CE08 AND M061-LE08 MOTORS
430 SERIES MOTION CONTROLS

(Continued)

TYPICAL PERFORMANCE CHARACTERISTICS

SERIES CONNECTION
M062-CS09 AND M062-LS09 MOTORS

PARALLEL CONNECTION
M062-CS09 AND M062-LS09 MOTORS

SERIES CONNECTION
M063-CS06 AND M063-LS06 MOTORS
430 SERIES MOTION CONTROLS
(Continued)

TYPICAL PERFORMANCE CHARACTERISTICS

SPEED (1.8" STEPS PER SECOND)

PARALLEL CONNECTION
M063-CEO6 AND M063-LEO6 MOTORS

TYPICAL PERFORMANCE CHARACTERISTICS

SPEED (1.8" STEPS PER SECOND)

SERIES CONNECTION
M063-CS09 AND M063-LS09 MOTORS

TYPICAL PERFORMANCE CHARACTERISTICS

SPEED (1.8" STEPS PER SECOND)

PARALLEL CONNECTION
M063-CEO9 AND M063-LEO9 MOTORS
430 SERIES MOTION CONTROLS
(Continued)

TYPICAL PERFORMANCE CHARACTERISTICS

SERIES CONNECTION
M093-FC14 AND M093-FD14 MOTORS

TYPICAL PERFORMANCE CHARACTERISTICS

PARALLEL CONNECTION
M093-FD8014 MOTOR

TYPICAL PERFORMANCE CHARACTERISTICS

SERIES CONNECTION
M111-FD12 MOTOR
TYPICAL PERFORMANCE CHARACTERISTICS

PARALLEL CONNECTION
M111-FO16 MOTOR

SERIES CONNECTION
M111-FD16 MOTOR

PARALLEL CONNECTION
M112-FD8012 AND M112-FJ8012 MOTORS
430 SERIES MOTION CONTROLS
(Continued)

TYPICAL PERFORMANCE CHARACTERISTICS

SERIES CONNECTION
M112-FJ327 MOTOR

SERIES CONNECTION
M112-FJ8030 MOTOR

PARALLEL CONNECTION
M112-FJ8030 MOTOR
6.1 MOTOR PERFORMANCE
All stepper motors exhibit instability at their natural frequency and harmonics of that frequency. Typically, this instability will occur at speeds between 50 and 600 full steps per second and, depending on the dynamic motor load parameters, can cause excessive velocity modulation or improper positioning.

There are also other instabilities which may cause a loss of torque at stepping rates outside the range of natural resonance frequencies. One such instability is broadly defined as mid-range instability. This is identified by the dotted area (...) on the speed/torque curves.

Usually, the dampening of the system and acceleration/deceleration through the resonance areas aid in reducing instability to a level that provides smooth shaft velocity and accurate positioning. If instability does cause unacceptable performance under actual operating conditions, the following techniques can be used to reduce velocity modulation.

1) Avoid constant speed operation at the motor's unstable frequencies. Select a base speed that is above the motors resonant frequencies and adjust acceleration and deceleration to move the motor through unstable regions quickly.

2) The motor winding current can be reduced as discussed in Section 3.7. Lowering the current will reduce torque proportionally. The reduced energy delivered to the motor can decrease velocity modulation.

SECTION 7: TROUBLESHOOTING

WARNING:
Motors connected to this drive can develop high torque and large amounts of mechanical energy.

Keep clear of the motor shaft and all parts mechanically linked to the motor shaft.

Turn off the power to the drive before performing work on parts mechanically coupled to the motor.

If installation and operating instructions have been followed carefully, this unit should perform correctly. If the motor fails to step properly, the following check list will be helpful.

In general:
- Check all installation wiring carefully for wiring errors or poor connections.
- Check to see that the proper voltage levels are being supplied to the unit.
- Be sure that the motor is a compatible model for use with this unit.
- Check to see that baud rates, parity and other communication parameters are properly set.
- When connecting this unit to a host computer, be sure device addresses are correct.

7.1 IF MOTOR DIRECTION (CW, CCW) IS REVERSED, Check:
Connection to the Motor Connector may be incorrect.

7.2 IF THE MOTOR MOTION IS ERRATIC, Check:
Low filter capacitor.
Supply voltage out of tolerance.

Motion parameters (low speed, acceleration/deceleration, jog speed, home speed, and feedrate) may need adjustment.
Operation in dotted area of speed/torque curve.

7.3 IF TORQUE IS LOW, Check:
AWO-(All Windings Off-) active or REDUCED CURRENT-active.
Improper supply voltage.
Operation in dotted area of speed/torque curve.

If a malfunction occurs that cannot be corrected by following the preceding check list, contact The Superior Electric Company.

SECTION 8: COMPONENT LAYOUT

INDEXER BOARD

APPENDIX

MICRO SERIES SINGLE-LINE PRESET INDEXER EMULATION

This Appendix provides two methods of emulating a one-line indexer function with a Superior Electric Micro Series Indexer. Method 1 utilizes the MDI (line 000) mode to function as a one-line indexer, while method 2 requires programming of lines 1 and 2. An assumption was made that a Superior Electric SSP-100 Programmer was interfaced to the Micro Series Indexer, although any parallel data entry device (such as a programmable controller) can utilize these methods.

Only the required parameters and program lines are programmed; the other parameters and program lines retain their factory defaults as shown on page 55.