PACIFIC SCIENTIFIC
HIGH PERFORMANCE MOTORS & DRIVES

110 Fordham Road
Wilmington, MA 01887
(978) 988-9800
Fax (978) 988-9940

Part# MA6430
List Price $30 U.S.
January, 1999
Rev 2

MA6440

6440 Indexer/Drive

User Manual

WARRANTY AND LIMITATION OF LIABILITY

Includes software provided by Pacific Scientific

Pacific Scientific warrants its motors and controllers (“Product(s)”) to the original purchaser (the “Customer”), and in the case of original equipment manufacturers or distributors, to their original consumer (the “Customer”) to be free from defects in material and workmanship and to be made in accordance with Customer’s specifications which have been accepted in writing by Pacific Scientific. In no event, however, shall Pacific Scientific be liable or have any responsibility under such warranty if the Products have been improperly stored, installed, used or maintained, or if customer has permitted any unauthorized modifications, adjustments, and/or repairs to such Products. Pacific Scientific’s obligation hereunder is limited solely to repairing or replacing (at its option), at its factory any Products, or parts thereof, which prove to Pacific Scientific’s satisfaction to be defective as a result of defective materials or workmanship, in accordance with Pacific Scientific’s stated warranty, provided, however, that written notice of claimed defects shall have been given to Pacific Scientific within two (2) years after the date of the product date code that is affixed to the product, and within thirty (30) days from the date any such defect is first discovered. The products or parts claimed to be defective must be returned to Pacific Scientific, transportation prepaid by Customer, with written specifications of the claimed defect. Evidence acceptable to Pacific Scientific must be furnished that the claimed defects were not caused by misuse, abuse, or neglect by anyone other than Pacific Scientific.

Pacific Scientific also warrants that each of the Pacific Scientific Motion Control Software Programs (“Program(s)”) will, when delivered, conform to the specifications therefore set forth in Pacific Scientific’s specifications manual. Customer, however, acknowledges that these Programs are of such complexity and that the Programs are used in such diverse equipment and operating environments that defects unknown to Pacific Scientific may be discovered only after the Programs have been used by Customer. Customer agrees that as Pacific Scientific’s sole liability, and as Customer’s sole remedy, Pacific Scientific will correct documented failures of the Programs to conform to Pacific Scientific’s specifications manual. PACIFIC SCIENTIFIC DOES NOT SEPARATELY WARRANT THE RESULTS OF ANY SUCH CORRECTION OR WARRANT THAT ANY OR ALL FAILURES OR ERRORS WILL BE CORRECTED OR WARRANT THAT THE FUNCTIONS CONTAINED IN PACIFIC SCIENTIFIC’S PROGRAMS WILL MEET CUSTOMER’S REQUIREMENTS OR WILL OPERATE IN THE COMBINATIONS SELECTED BY CUSTOMER. This warranty for Programs is contingent upon proper use of the Programs and shall not apply to defects or failure due to: (i) accident, neglect, or misuse; (ii) failure of Customer’s equipment; (iii) the use of software or hardware not provided by Pacific Scientific; (iv) unusual stress caused by Customer’s equipment; or (v) any party other than Pacific Scientific who modifies, adjusts, repairs, adds to, deletes from or services the Programs. This warranty for Programs is valid for a period of ninety (90) days from the date Pacific Scientific first delivers the Programs to Customer.

THE FOREGOING WARRANTIES ARE IN LIEU OF ALL OTHER WARRANTIES (EXCEPT AS TO TITLE), WHETHER EXPRESSED OR IMPLIED, INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR OF FITNESS FOR ANY PARTICULAR PURPOSE, AND ARE IN LIEU OF ALL OTHER OBLIGATIONS OR LIABILITIES ON THE PART OF PACIFIC SCIENTIFIC. PACIFIC SCIENTIFIC'S MAXIMUM LIABILITY WITH RESPECT TO THESE WARRANTIES, ARISING FROM ANY CAUSE WHATSOEVER, INCLUDING WITHOUT LIMITATION, BREACH OF CONTRACT, NEGLIGENCE, STRICT LIABILITY, TORT, WARRANTY, PATENT OR COPYRIGHT INFRINGEMENT, SHALL NOT EXCEED THE PRICE SPECIFIED OF THE PRODUCTS OR PROGRAMS GIVING RISE TO THE CLAIM, AND IN NO EVENT SHALL PACIFIC SCIENTIFIC BE LIABLE UNDER THESE WARRANTIES OR OTHERWISE, EVEN IF PACIFIC SCIENTIFIC HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES, FOR SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, INCLUDING WITHOUT LIMITATION, DAMAGE OR LOSS RESULTING FROM INABILITY TO USE THE PRODUCTS OR PROGRAMS, INCREASED OPERATING COSTS RESULTING FROM A LOSS OF THE PRODUCTS OR PROGRAMS, LOSS OF ANTICIPATED PROFITS, OR OTHER SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER SIMILAR OR DISSIMILAR, OF ANY NATURE ARISING OR RESULTING FROM THE PURCHASE, INSTALLATION, REMOVAL, REPAIR, OPERATION, USE OR BREAKDOWN OF THE PRODUCTS OR PROGRAMS, OR ANY OTHER CAUSE WHATSOEVER, INCLUDING NEGLIGENCE.

The foregoing shall also apply to Products, Programs, or parts for the same which have been repaired or replaced pursuant to such warranty, and within the period of time, in accordance with Pacific Scientific's date of warranty.

No person, including any agent, distributor, or representative of Pacific Scientific, is authorized to make any representation or warranty on behalf of Pacific Scientific concerning any Products or Programs manufactured by Pacific Scientific, except to refer purchasers to this warranty.

Table of Contents

1 Overview of 6440

In this chapter

This chapter introduces the 6440 Indexer/Drive. Topics covered are:

- 6440 definition
- Other system components
- System diagram
- Warranty information

1.1 6440 Definition

Introduction

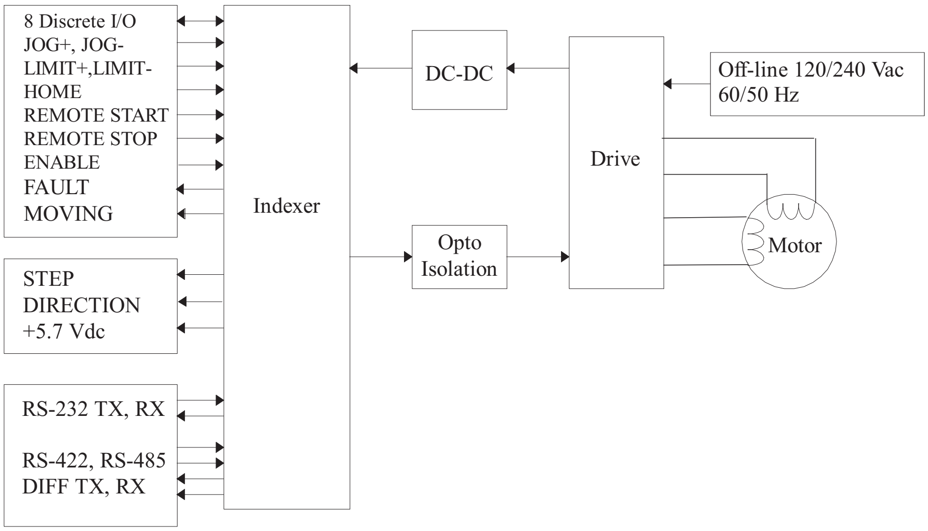
The 6440 Microstepping Indexer/Drive provides economical microstepping control with a simple mnemonic programming interface. Operation is programmed via the serial communications port. A combination of dedicated and user-programmable I/O provides motor control, status indication and sensor feedback.

Simple, single-letter mnemonics are used to specify a wide variety of motion commands. A terminal or PC with terminal software can be used to familiarize oneself with 6440 operation. Various motion parameters can be modified and motion executed immediately from the command line or from on-board non-volatile memory. Many applications can take advantage of the simplicity of developing programs for on-board execution. If more versatility is required, the user can write a custom program running on a host computer issuing immediate motion commands to the 6440.

The output current of the 6440 is dip switch selectable from 0.625A rms (0.88A peak in microstep mode) to 5A rms (7.1 A peak in microstep mode).

The Pacific Scientific 6440 can be powered from 120 or 240 Vac (60/50 Hz). This input is switch selectable for either 120 or 240 Vac. An internal PWM switching power supply provides up to 300 W ± 10% of power to the stepper drive.

Block diagram



The 6440 can communicate with RS-232, RS-422 or RS-485 serial protocols. Discrete I/O lines provide external start/stop and motor enable control, home and limit switch testing, motor jogging, slave drive interface and eight programmable bi-directional discrete I/O lines. The eight programmable bi-directional discrete I/O lines are individually jumper configurable for input or output. Immediate command line instructions can read back and write all eight bits.

Fixed resolution mode

The unit operates in one of two modes, either Fixed Resolution Mode or Variable Resolution Mode. In Fixed Resolution Mode the resolution can be set from 200 steps/rev to 51,200 steps/rev. The minimum and maximum step rate range in value from 0.08 to 19,000 steps/sec and vary depending on the resolution. Of course, the finer step sizes result in high shaft resolution at the expense of shaft speed. The motion profile is trapezoidal with programmable initial and final speeds. The acceleration and deceleration rates are specified by a single command in terms of acceleration and deceleration factors ranging in value from 1 to 255.

Variable resolution mode

Variable Resolution Mode is the recommended alternative where micro-stepping is desired without compromising speed. This mode allows for positioning at a resolution of 20,000 steps/revolution and a top speed of 5700 RPM.

Drive features

Bipolar chopper drive - patented 4-phase PWM (pulse width modulation) chopping electronically controls the motor winding currents at 20 KHz frequency. This combines the best of recirculating and non-recirculating current regulation producing high back EMF rejection with low chopping ripple current. Benefits include: reduced heat dissipation, low electric noise and improved current control during motor braking.

Digital Electronic Damping - patented circuit eliminates torque and/or motor stalling through mid-speed region that is inherent in all open loop stepper applications.

Short circuit protection circuitry - disables the drive if a short circuit occurs on the motor outputs. The drive must be power cycled to clear fault.

Fault protection - Line-to-line and line-to-neutral shorts, Internal power supply under-voltage, Bus overvoltage.

Fixed Resolution microstepping - Binary Steps: Full, 1/2, 1/4, 1/8, 1/16, 1/32, 1/64, 1/128, 1/256.

Variable Resolution microstepping - Decimal step resolution 0.01 steps at effective full step speeds of 20-19,000 steps/second.

Optical isolation - Indexer optically isolated from drive.

UL Recognized - 508C (Type R) - **PENDING** File E-137798. This also complies with CSA Standard for Process Control Equipment, C22.2 No. 142-M1987.

Vibration - **PENDING** - IEC Standard 68-2-6.

Power supply features

66 Vdc Output - three pin pluggable connector (J6) designed to supply 66 Vdc to power an additional drive. The total power available for both the internal and external drives is 66 Vdc @ 4.6 A or 300 W ± 10%.

Indexer features

Simple mnemonic command set - may be executed from internal NVRAM or from a more powerful host.

Serial port communications - supporting RS-232, RS-422 and RS-485 multi-drop.

Multi-axis - Single 6440 can control two independent or synchronous axes with an additional drive and no glue logic.

Input/Output ports - Eight general purpose bi-directional user programmable input/output ports.

Internal Memory - Approximately 1792 bytes of non-volatile memory is available for user programs.

User adjustments

Motor current - sets the motor phase current to 5.0, 4.375, 3.75, 3.125, 2.5, 1.875, 1.25, or 0.625 A rms.

using DIP switch S1

Digital Electronic Damping control - enables this patented feature which eliminates loss of torque and possible motor stalling conditions when operating at mid-range speeds. This instability is a phenomenon of the electronic, magnetic and mechanical characteristics of a stepping motor system. The compensation circuit damps mid-range oscillations by advancing or delaying switching of the output current relative to the incoming pulse train.

Idle current reduction (ICR) - enables or disables idle current reduction which reduces motor winding current by 50% of its rated value during motor dwell periods. ICR begins 0.1 second after the last input step pulse occurs.

Note: *The current will return to 100% at the next step pulse.*

using plug-on jumpers

RS Communication Configuration - allows the user to select RS-232/RS-422 communication or RS-485 communication.

I/O Configuration - Allows the user to configure lines as inputs or outputs.

Synchronous or Alternating Operation - allows the user to select synchronous or alternating operation of slave drives.

using AC Switch SW1

Off line 120/240 Vac - switch selects AC input. **DO NOT** apply 240 Vac with AC Switch in 120 Vac position.

Typical applications

Typical applications for the 6440 include:

- Clutch Brake Replacement
- Labeling Machines
- Packaging/Specialty Machinery
- Semiconductor Wafer Polishing
- Constant Speed Applications

1.2 Other System Components

Overview

The other components that, along with the drive comprise a complete motor control system are:

- Motor

Installation guidelines for these components are described in Chapter 2, “Installing the 6440.”

1.3 Warranty

The Pacific Scientific 6440 Indexer/Drive has a **two year warranty** against defects in material and assembly. Products that have been modified by the customer, physically mishandled or otherwise abused through miswiring, incorrect switch settings and so on, are exempt from the warranty plan.

2 Installing the 6440

In this chapter

This chapter explains how to install the 6440 Indexer/Drive. Topics covered are:

- Unpacking and inspecting the 6440
- Installing and using the 6440 unit safely
- Selecting other system components
- Mounting the 6440 in your installation
- Connecting input/output cables

2.1 Unpacking and Inspecting the 6440

Unpacking procedure

1. Remove the 6440 from the shipping carton. Make sure all packing materials are removed from the unit.
2. Check the items against the packing list. A label located on the unit identifies the unit by model number, serial number and date code.

Inspection procedure

Inspect the unit for any physical damage that may have been sustained during shipment.

If you find damage, either concealed or obvious, promptly contact your buyer to make a claim with the shipper. Do this within 10 days of receipt of the unit.

Storing the unit

After inspection, store the controller in a clean, dry, place. The storage temperature must be between -40 degrees C and 70 degrees C. To prevent damage during storage, replace the unit in the original shipping carton.

2.2 Installing and Using the 6440 Unit Safely

Your responsibility

As the user or person applying this unit, you are responsible for determining the suitability of this product for any application you intend. In no event will Pacific Scientific Company be responsible or liable for indirect or consequential damage resulting from the misuse of this product.

Note: *Read this manual completely to effectively and safely operate the 6440 unit.*

Warning

The circuits in the 6440 drive are a potential source of severe electrical shock. Follow the safety guidelines to avoid shock.

Safety guidelines

To avoid possible personal injury whenever you are working with the 6440 unit:

- Do not operate the drive without the motor case tied to earth ground.

  Note: *This is normally done by connecting the motor's case to J3-5 of the 6440 and connecting J1-3 of the 6440 to earth ground. These pins are marked by a* ⏚ *symbol on the silk-screen and are shown on the diagram on page 2-5.*

- Do not make any connections to the internal circuitry. The input and output signals are the only safe connection points.
- Always remove power before making or removing connections from the unit.
- Be careful of the J3 motor terminals when disconnected from the motor. With the motor disconnected and power applied to the drive, these terminals have high voltage present, even with the motor disconnected.
- Do not use the ENABLE input as a safety shutdown. Always remove power to the drive for a safety shutdown.

2.3 Selecting Other System Components

Selecting a motor

The 6440 is designed for use with Pacific Scientific's line of hybrid stepper motors or most other 2 phase stepper motors. The drive works with either the standard line or the enhanced high performance line of stepper motors. The motor winding current rating must be compatible with the output current of the drive package.

Refer to the Torque/Speed Curves in the Pacific Scientific "Motion Control Solutions Catalog" or contact your local Pacific Scientific distributor for sizing and motor compatibility assistance.

2.4 Mounting the 6440 in Your Installation

Cabinet selection

Select a standard 8-inch (205 mm) or deeper NEMA (National Electrical Manufacturers Association) enclosure appropriate for industrial applications.

Caution

The internal operating temperature should not exceed 50° C. If the cabinet is ventilated by filtered or conditioned air, make sure to prevent the accumulation of dust and dirt on the unit's electronic components. The air should also be free of corrosive or electrically conductive contaminants.

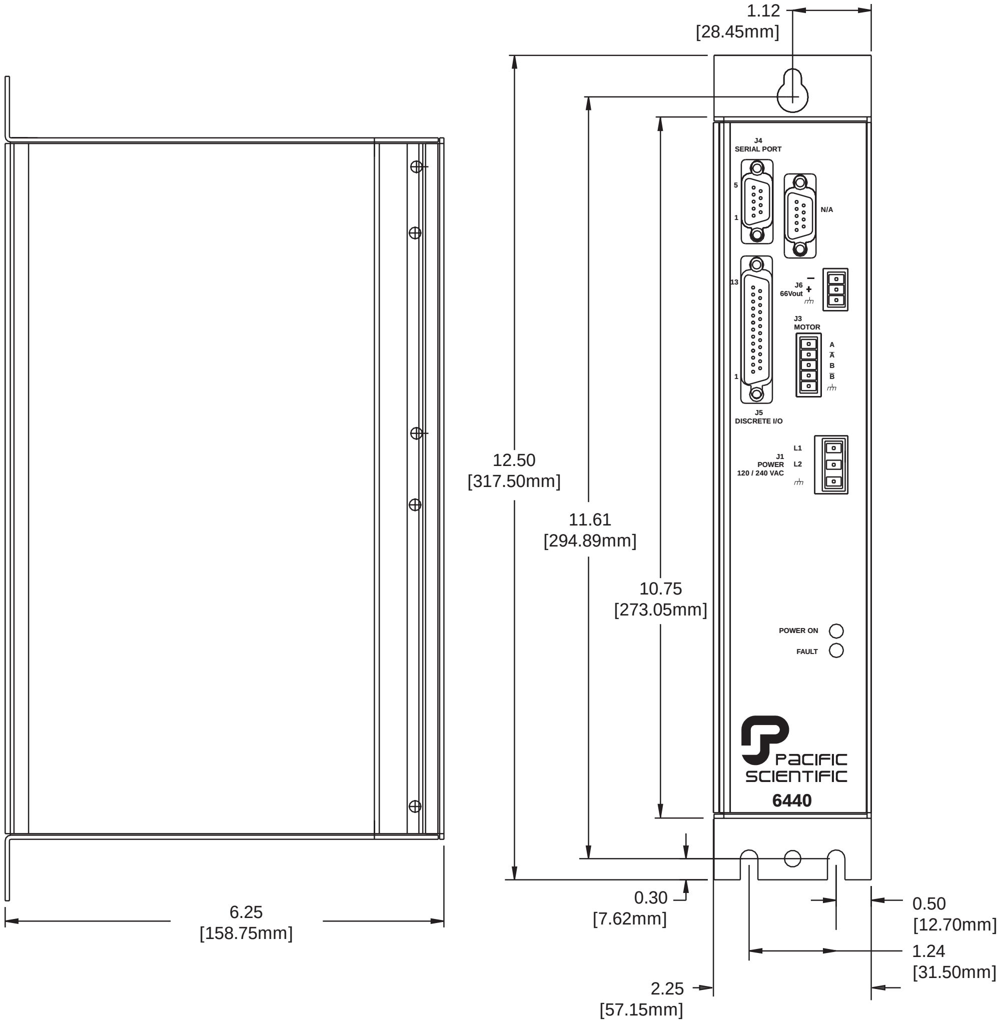
Mounting guidelines

Your installation should meet the following guidelines:

- Vertical orientation for the unit
- Flat, solid surface capable of supporting the approximate 6.0 lb. weight (2.7 kg. mass) of the unit.
- Free of excessive vibration or shock.
- Minimum unobstructed space of 4 inches (10 cm) above and below the unit.
- Maximum ambient temperature of 50°C

Mounting dimensions

When mounting the 6440, please refer to the dimensions below:



2.5 Connecting the Five Input/Output Cables

Introduction

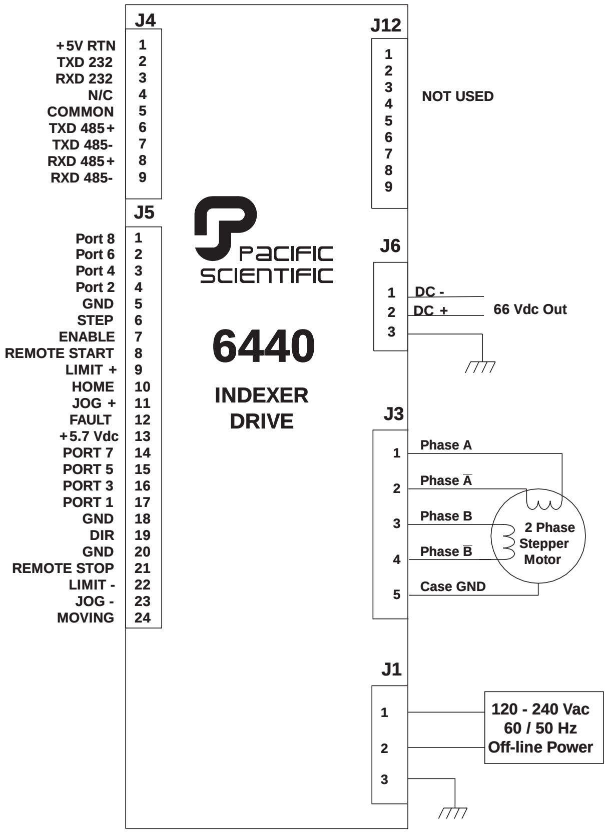
The five input/output cables are:

- J1 - Power
- J3 - Motor
- J4 - Serial port
- J5 - Discrete inputs and outputs
- J6 - 66 Vdc Out

Note: *J12 Connector is* *NOT USED.*

Connection diagram

These inputs and outputs are shown as follows:



Wiring is application specific

Wiring sizes, wiring practices and grounding/shielding techniques described in the following section represent common wiring practices and should prove satisfactory in the majority of applications.

Caution

Non-standard applications, local electrical codes, special operating conditions, and system configuration wiring needs take precedence over the information included here. Therefore, you may need to wire the drive differently then described here.

Noise pickup reduction

Use shielded and twisted cabling for the signal and power cables as described below. This precaution reduces electrical noise.

Shock hazard reduction

Refer to section 2.2 for safety information that must be followed to reduce shock hazard.

2.5.1 J1 120/240 Vac Power Connector

Introduction

The J1 power connector should be used to power the 6440 from 120/240 Vac (60/50 Hz).

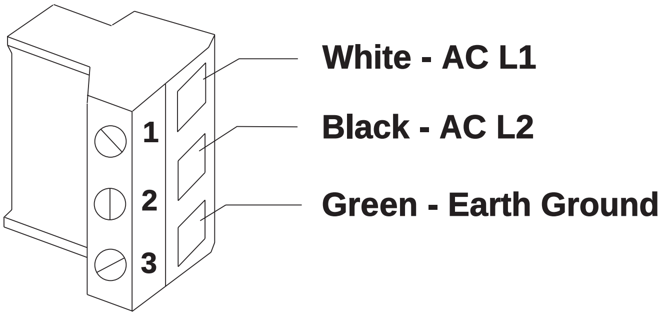
Mating connector

The J1 120/240 Vac power connector mates to a PCD 3-pin screw cable connector. The mating connector, supplied with the unit, is type ELFP03210.

Cable requirements

Use 16- to 14-gauge shielded wire for the cabling.

Cable diagram - J1 power cable



Procedure

1. Strip the wires to 0.27 inch (7 mm).
2. Attach the wire to the connector as indicated in the diagram.

   Note: *Make sure the screws on the PCD connector are tightened down firmly to the wiring.*

Caution

Do not pre-tin (solder) the tops of the cables going into the PCD connector. This can result in a loose connection.

2.5.2 J3 - Motor Connection

Introduction The J3 motor cable connects the drive to the motor windings. Motor cables are available from Pacific Scientific, or you can make your own.

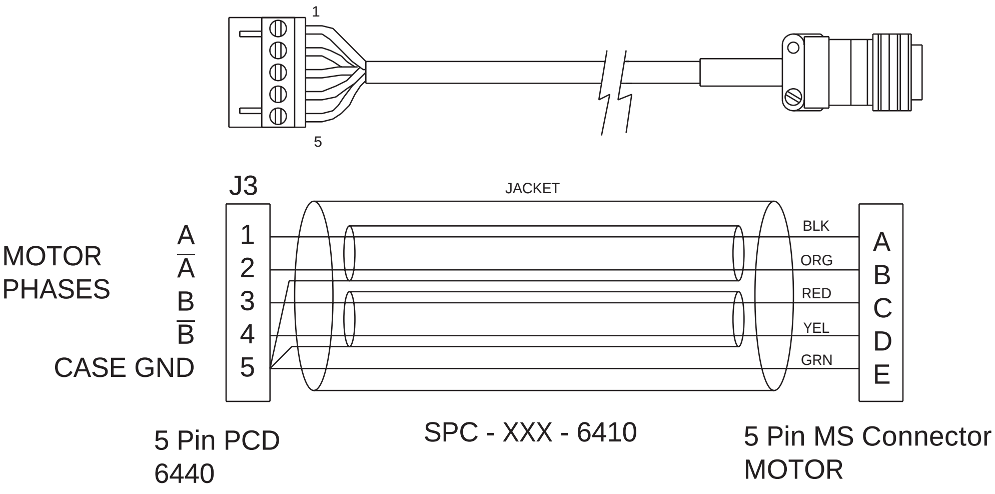
Pacific Scientific cable

Pacific Scientific makes cables that connect directly from J3 to our system motors. To order the cable from Pacific Scientific, use the order number SPC-xxx-6410, where “xxx” is the length, in feet (one-foot increments) up to 50 feet. For example, SPC-050 is a 50 feet long cable.

Pacific Scientific makes cables for both E and H series stepper motors (SPC-xxx-6410) and K and N series stepper motors (SPC-xxx-6410-KN). Please refer to the correct diagram on the following page.

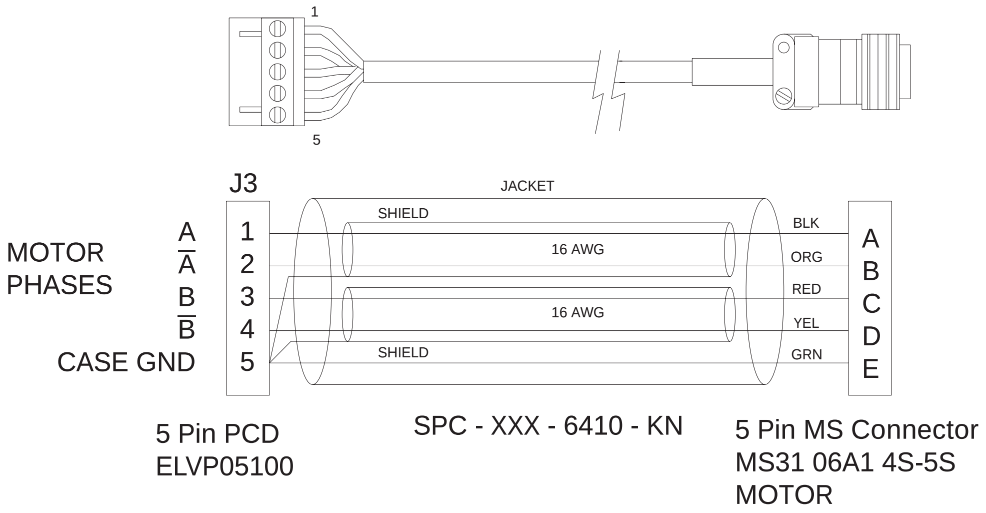
Pacific Scientific cabling diagram

If you are using Pacific Scientific **E** and **H** series motor cable, with the mating connectors already attached, install as follows:



Pacific Scientific cabling diagram

If you are using Pacific Scientific's **K** or **N** series stepper motors, install as follows:



Making your own cable

To make your own motor cable, follow the guidelines given below for wiring to the J3 mating connector. Depending on your motor configuration, refer to the appropriate diagram at the end of this section to determine the motor connections required.

J3 connection table

Output	Pin	Explanation
Motor Phase A	J3-1	Motor Phase A excitation. Twisted Pair.
Motor Phase $\overline{A}$	J3-2	
Motor Phase B	J3-3	Motor Phase B excitation. Twisted Pair.
Motor Phase $\overline{B}$	J3-4	
Drive Case (Earth) Ground	J3-5	Connected to the motor case ground.

Mating connector

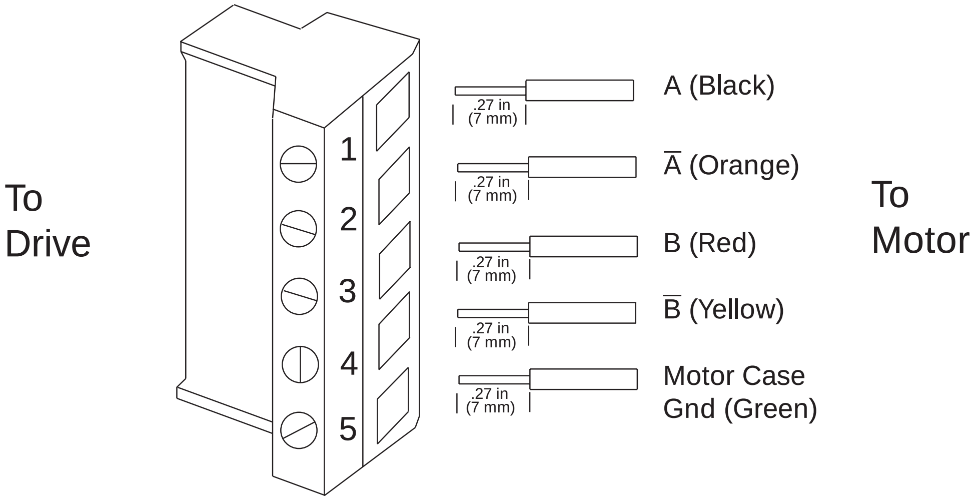
The J3 motor connector on the 6440 mates to a 5-pin PCD screw cable connector. The mating cable connector is type ELVP05100.

Cable requirements

The mating connector terminals will accept #16 to #18 AWG wire. Pacific Scientific recommends using #16 AWG.

For the motor cable, use cable with two twisted pairs twisted at about 3 to 4 turns per inch (1 to 1.5 turns per centimeter) for the motor phase excitations and a fifth wire for the case ground. As an option, the cable may be shielded to reduce radiated noise. A single shield can be used around both phase excitations and the ground wire or each phase excitation (twisted pair) can be individually shielded as in the Pacific Scientific cables. Connect shields to pin 5 of the mating connector.

Cable diagram - J3 motor



Note: *The colors in the diagram follow the Pacific Scientific stepper motor cable color code.*

Procedure

1. Strip the wires to 0.27 inch (7 mm).
2. Attach the wire to the connector as indicated in the diagram.

   Note: *Make sure the screws on the PCD connector are tightened down firmly to the wiring.*

Caution

Do not pre-tin (solder) the tops of the cables going into the PCD connector. This can result in a loose connection.

Procedure (cont'd)

3. Remove power from the 6440.

Warning

Always remove power before making or removing connections to the unit. The motor terminals have high voltage present when the 6440 is On.

4. Plug the mating connector firmly into the 6440.
5. Connect the cable shield to 6440 ground, if applicable.
6. Plug the other mating connector into the motor.
7. Switch On the 6440.

2.5.2.2 8-Lead Motor, Series Connected

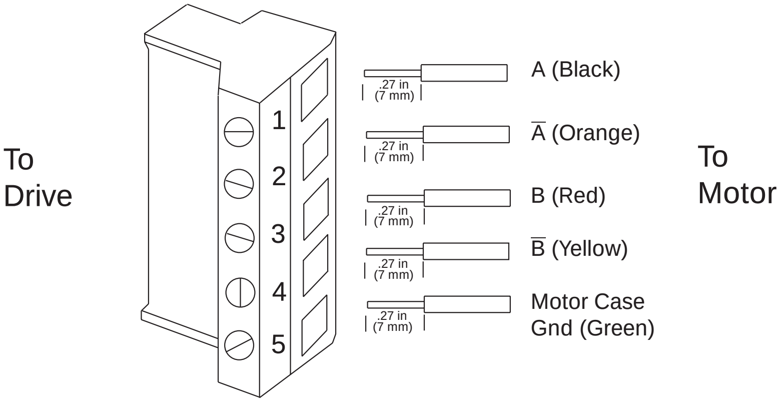
Introduction

For an 8-lead motor to be wired in series, build and install the cable as follows:

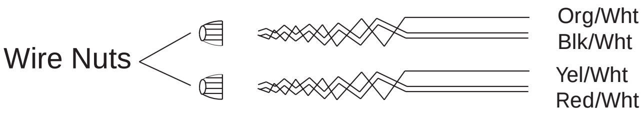
Building 8-lead series motor cable

The connector assembly using standard Pacific Scientific stepper motor color coding is shown below:

Cabling diagram



In addition, make the following connections at the motor:

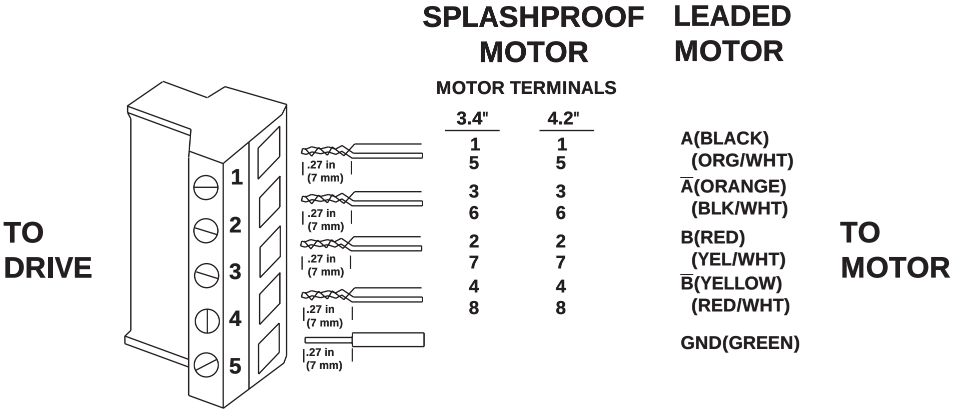


2.5.2.1 8-Lead Motor, Parallel Connected

Introduction For an 8-lead motor to be wired in parallel, build and install the cable as follows:

Building 8-lead The connector assembly using standard Pacific Scientific parallel motor cablestepper motor color coding is shown below:

Cabling diagram



2.5.3 J4 - Serial Port Connection

Introduction The serial port transmits and receives RS-232 communication for the user interface of your unit.

Mating connector The J4 serial port mating connector, supplied with the unit, is an ITT Cannon DE-9S 9-pin plug-in female D connector.

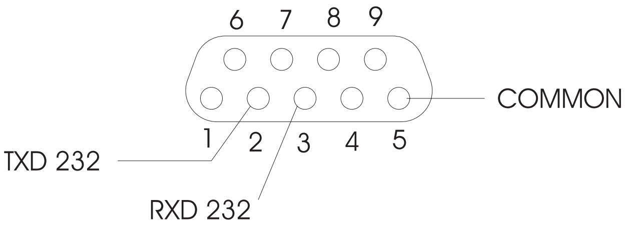
Baud rate

The 6440 supports single-unit RS-232, RS-422 and multi-unit RS-485 communication protocols running at 9600 BPS, 1 start bit, 1 stop bit. RS-232 and RS-422 mode selection is automatic dependent upon whether connections are made to the single-ended RS-232 TX and RX lines or to the differential TX and RX lines. RS-485 mode is jumper selected and allows up to 32 units on a single differential RS-485 communications link. The host computer is the bus master and always initiates packet transfers. External termination may be required in some installations. The unit is factory configured for RS-232 protocol. The table below shows the Serial Port pin designations.

Input/Output	Pin Number	Description
+ 5 Vdc RTN/Shield	J4-1	RS-232 Transmit Data (output)
TXD_232	J4-2	
RXD_232	J4-3	RS-232 Receive Data (input)
COMMON	J4-5	
TXD_485 +	J4-6	Differential Transmit Data (output)
TXD_485 -	J4-7	Differential Transmit Data (input)
RXD_485 +	J4-8	Differential Receive Data (output)
RXD_485 -	J4-9	Differential Receive Data (input)

2.5.3.1 Single-unit RS-232 Connection

Solder connections to TXD_232 (output), RXD_232 (input) and COMMON. After installing the power and serial communications cable you can quickly test the unit by applying power and pressing <Esc> and then the space bar. You should see the Pacific Scientific sign-on message and copyright notice.



2.5.3.2 Single-unit RS-422 Connection

Solder connections to TXD_485± and RXD_485 ± differential lines. After installing the power and serial communications cable you can quickly test the unit by applying power and pressing <Esc> and then the space bar. You should see the Pacific Scientific sign-on message and copyright notice.

Build the cable to connect to your computer or terminal by referring to the documentation for the device.

Note: *Pinouts vary among computer manufacturers. Check your computer's hardware reference manual before wiring.*

2.5.3.3 Multi-unit RS-485 Connection

Solder connections to TXD_485± and RXD_485 ± differential lines. This mode requires each 6440 to be initialized with a single-letter node address to be used with each subsequent command. This node address must be setup in Single-Unit mode before the RS-485 mode operation is attempted. To set this address, perform the following:

Procedure

1. Apply power and press a key (a-z, A-Z) **NOT the Spacebar**, to be used as the address followed by a single space. Execute the P command to program the node address into non-volatile memory.
2. Power-down the unit and insert jumper E11 to configure the unit for RS-485 mode. There is no sign-on procedure while in RS-485 multi-unit mode. This mode is designed to be used with a host computer, but can be tested using a terminal.
3. Type in the node address. The character should be echoed if the unit recognizes the address.
4. Type a simple motion command, **R 500** followed by a <LF> (line feed) terminating character (press <**Ctrl**> <**J**>). The motor should now run.

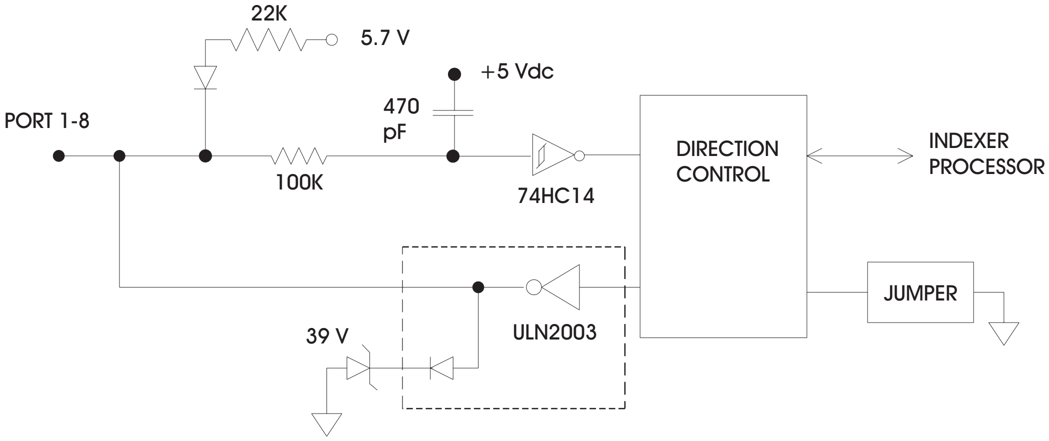
Note: *Units configured for single-unit RS-232 or RS-422 always have the serial drive lines enabled. Therefore, avoid connecting one of these units into a chain of RS-485 configured units.*

2.5.4 J5 - Discrete Inputs/Outputs Connection

Introduction

The 6440 features 8 user-programmable discrete input/output lines. Input/output selection for Port 1 to 8 is configured by jumpers E3 to E10 respectively. To enable a port as an input, remove the respective jumper. To enable as an output, install the jumper. Always remove the external cabling when making I/O configuration changes to ensure that an external low impedance input is not driving an I/O Port configured as an output The block diagram of a single I/O bit is shown below.

Block diagram



Note: *Inputs accept 0-30 VDC max, where $V_{in} \leq .8V$ is a logic low and $V_{in} \geq 3.7V$ is a logic high. All outputs are Open Collector 30 VDC, 70 ma sink, $V_{sat} \leq 1.0V$. All I/O signals are active low.*

Note: *If a bit is desired to be an input, the jumper must be out and an inactive level (low) written to the bit using the Y command. Otherwise, as a consequence of the wired "AND" design of the indexer, a logic "1" written to a bit configured as an input will always be read back as a "1." Programs executing from non-volatile memory can only output to these discrete bits, reading is not allowed.*

J5 Discrete I/O

The J5 connector pin designations are shown below:

Port	Pin Number	Jumper	I/O
1	J5-17	E3	Input 1 OR Output 1
2	J5-4	E4	Input 2 OR Output 2
3	J5-16	E5	Input 3 OR Output 3
4	J5-3	E6	Input 4 OR Output 4
5	J5-15	E7	Input 5 OR Output 5
6	J5-2	E8	Input 6 OR Output 6
7	J5-14	E9	Input 7 OR Output 7
8	J5-1	E10	Input 8 OR Output 8
GND	J5-5	N/A	Ground return for I/O signals.
GND	J5-18		
GND	J5-20		

Note: *The Discrete I/O connector has its own ground points, completely isolated from the unit's bus power.*

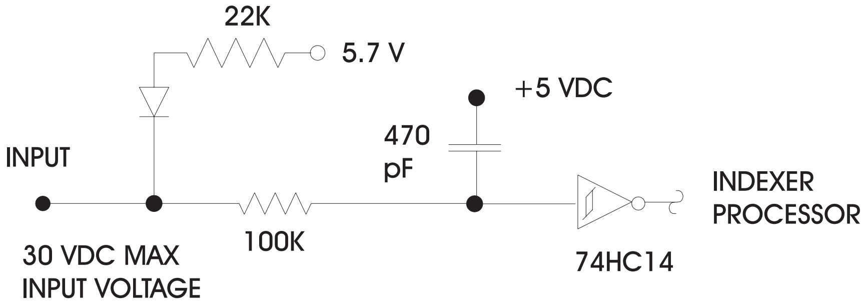
The minimum requirement to operate the unit is to enable motor current by connecting J5-7 (ENABLE) to one of the three J5 ground points (J5-5, J5-18, or J5-20).

2.5.4.1 Dedicated Discrete Input/Output Lines

Introduction

The 6440's operation is controlled by several input lines. These lines are active low. Input lines are RC filtered and passed to a 74HC14 Schmidt Trigger. The block diagram of input and output lines are shown below.

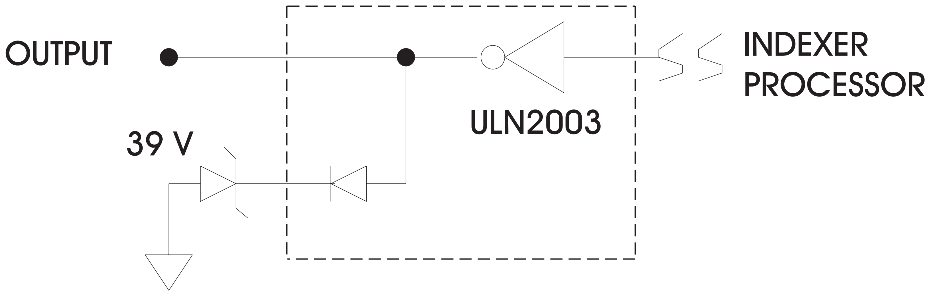
Input



Note: *Inputs accept 0-30 VDC max, where $V_{in} \leq .8V$ is a logic low and $V_{in} \geq 3.7V$ is a logic high.*

The 6440's outputs are open-collector Darlington ULN2003 drives. Inductive kickback protection is provided by the built-in diode and 39 V Zener diode.

Outputs



Note: *All outputs are Open Collector 30 VDC, 70 ma sink, $V_{sat} \leq 1.0V$. All I/O signals are active low.*

2.5.4.2 Dedicated Input/Output Pin Designations - J5

Introduction

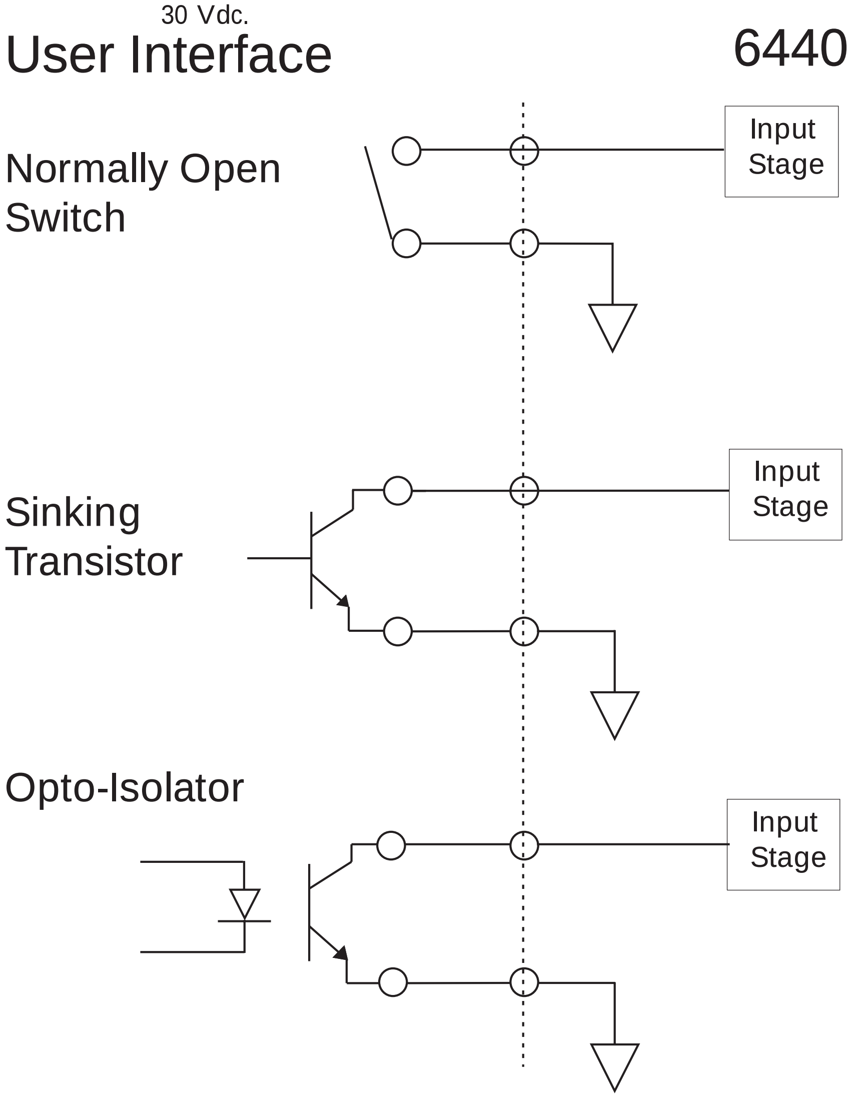
The pin definitions for all 6440 control lines are shown below:

J5 Discrete I/O table

Input/Output	Pin	Description
Remote Start	J5-8	This input activates program at location zero.
Remote Stop	J5-21	This input halts 6440 program and motion.
LIMIT +/LIMIT -	J5-9, J5-22	Activation of Limit + prohibits motion in CW direction. Activation of Limit - prohibits motion in CCW direction.
Home	J5-10	A transition at this input defines the electrical home position.
JOG +/JOG -	J5-11, J5-23	These inputs manually step the motor in the CW and CCW direction.
ENABLE	J5-7	This input is an active-low motor drive enable.
STEP	J5-6	This output is an active-low step pulse to slave drive approximately 400-600 Ns in duration. Only in Fixed Resolution Mode.
DIR	J5-19	This output provides direction to slave drive. CCW is a logic low, CW is a logic high. Only in Fixed Resolution Mode.
+5.7 V	J5-13	This output provides power for slave STEP and DIR when used with a 6410 drive.
FAULT	J5-12	The output is an active-low fault indicator. It must be cleared by powering down.
MOVING	J5-24	This output is an active-low motion indicator.
GND	J5-20	Indexer ground, isolated from drive ground.

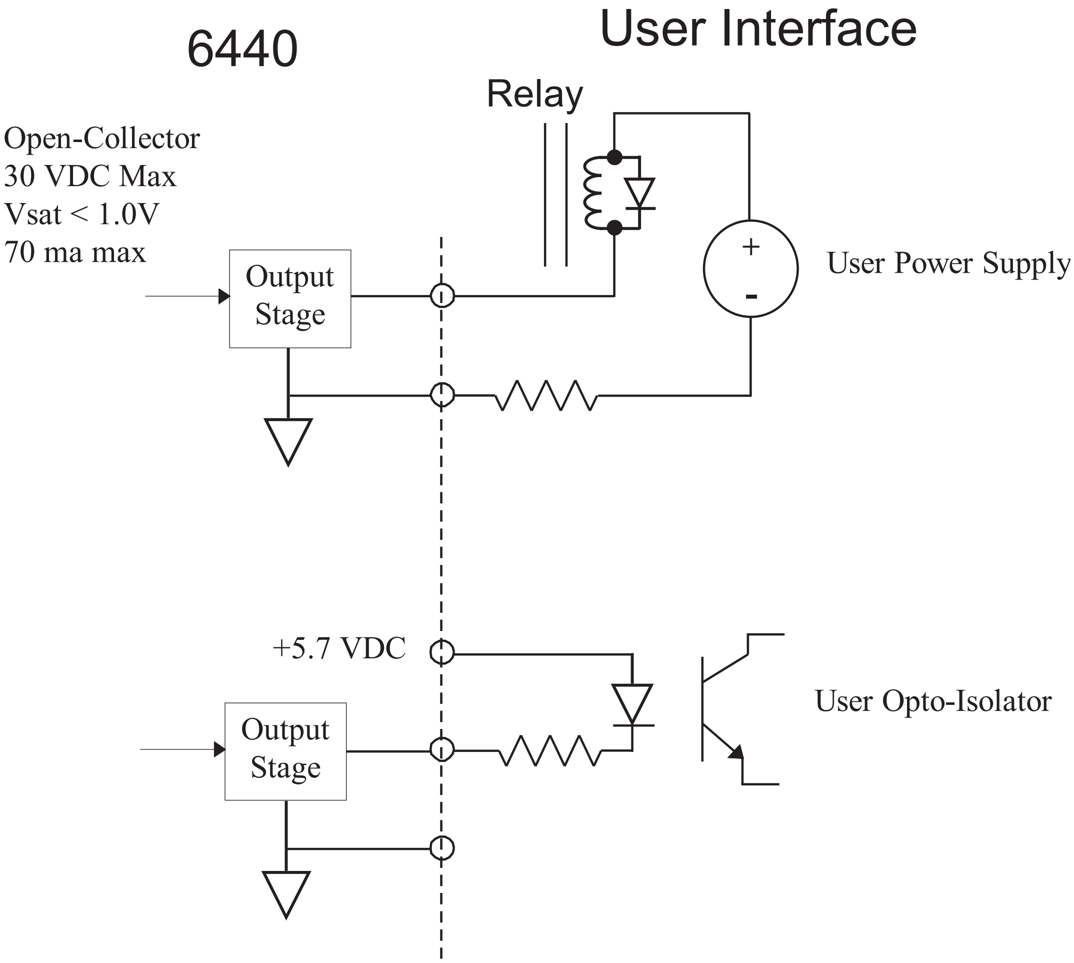
2.5.4.3 Input Interface Suggestions

These are examples of driving regular inputs and I/O pins configured as inputs. The input lines pull down to ground and the internal 22k ohm resistor through a 1N914 diode pulls up to 5.7 Vdc. The input is designed such that the user can actively drive the inputs with a low impedance source from 0 to 30 Vdc.



2.5.4.4 Output Interface Suggestions

The output stage is a Darlington sink to ground. The user generally supplies the +V power. However, the 6440 can provide up to 200 mA from the 5.7 Vdc intended for the slave 6410 interface, for applications such as driving 5 Volt nominal Opto 22 devices.



2.5.5 J6 - External 66 Vdc Output Connection

Introduction

The J6 external 66 Vdc output allows the 6440 to power additional drives.

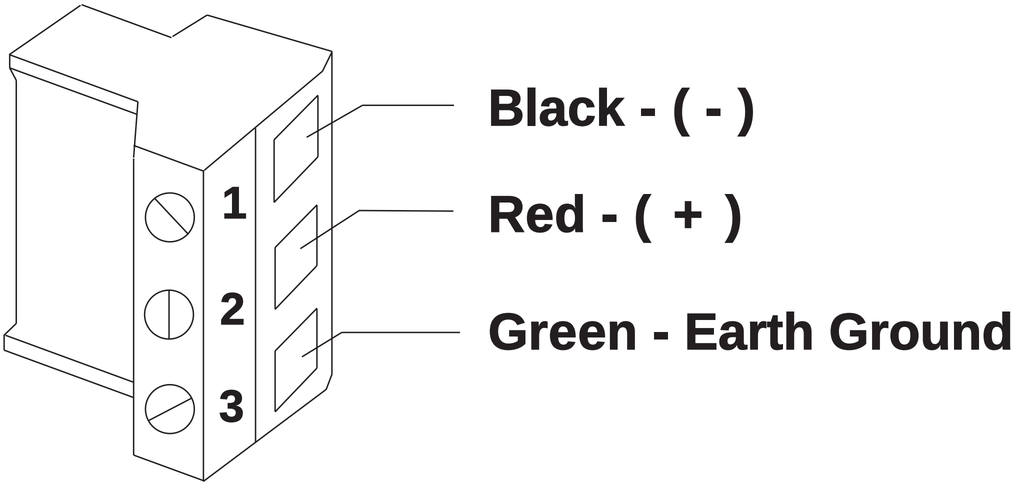
Mating connector

The J6 output connector mates to a PCD 3-pin screw cable connector. The mating connector, supplied with the unit, is type ELVP03100.

Cable requirements

Use 18- to 16-gauge shielded wire for the cabling.

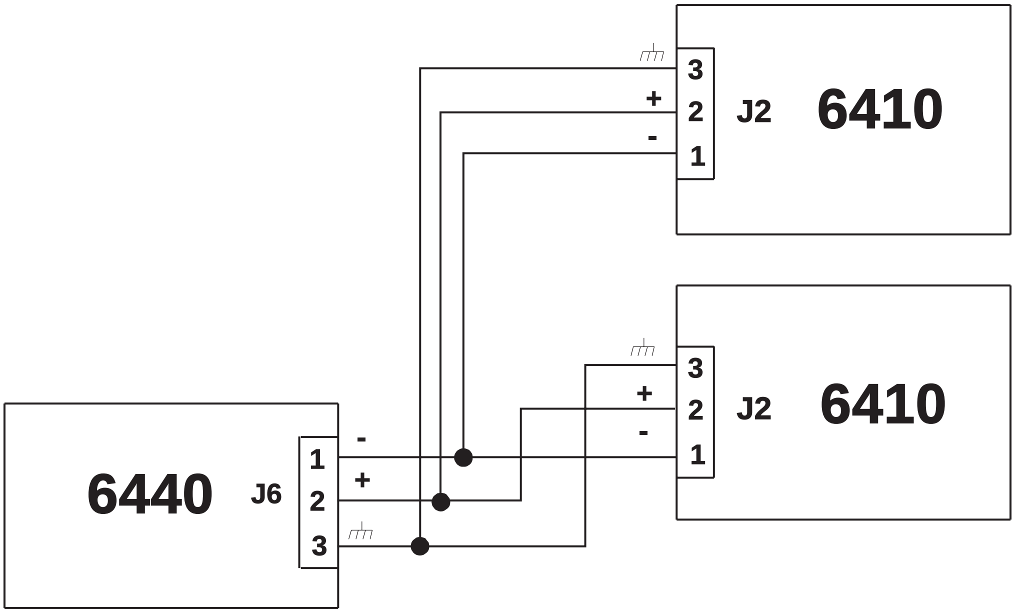
Cable diagram - 66 Vdc Output



Connection diagram

In multi-axis applications, if additional 6410s are added, it is preferable to run each power connection from the J6 DC output to the 6440 as shown below. **DO NOT** daisy-chain the power connections.

Note: *The total power available for both the internal and external drives is 66 Vdc @ 4.6 A or approximately 300 W ±10%. If the two drives are running simultaneously and require more than 4.6 A, the voltage will drop. The power supply has a low voltage protection circuit that will fault the drive if the voltage is < 55 Vdc.*



Note: *If the 6430 is powering additional 6410s with J6, a total of 1000 μf (maximum) 100 Vdc aluminum electrolytic capacitor, rated for 2A ripple current or greater @ 10 KHz and 105°C, must be installed at the 6410 (as close to the 6410 as possible) if the cable length is over 3 feet. DO NOT exceed 1000 μf total on J6 external connector.*

2.6 Using the 6440 to Control a 6410 Drive

Introduction A single 6440 can be used to control two axes in either synchronous or alternating fashion. The 6440 Indexer/Drive must be in Fixed Resolution mode. The slave 6410 drive is usually set for the same step resolution but doesn't necessarily have to be. Jumpers E12 and E13 are used to configure these modes.

Installation

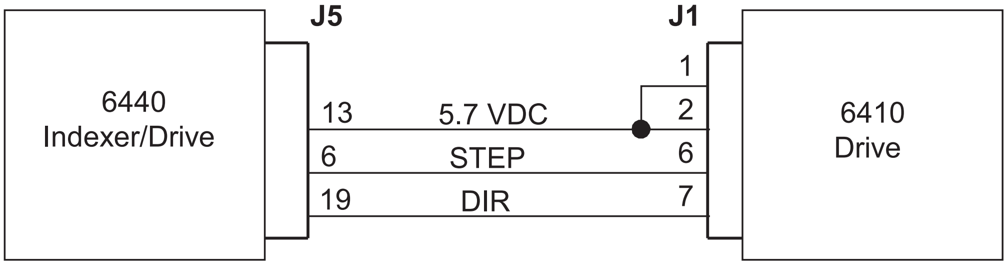
2.6.1 Synchronous Operation - E12 OUT, E13 IN

The T command, with enb = 1 enables step pulses to both master and slave. Issuing T with enb = 0 disables step pulses to both master and slave.

2.6.2 Alternating Operation - E12 IN, E13 OUT

The T command, with enb = 1 enables step pulses to the master and disables the slave. Issuing T with enb = 0 disables the master and enables slave.

Note: *Do not insert both E12 and E13 at the same time.*



Note: *All slave 6410 jumpers and DIP switches must be properly set.*

3 Powering up the 6440

In this chapter

This chapter explains how to power up the 6440 after installation. Topics covered are:

- Setting up functions using switch S1
- Setting up functions using jumpers E1- E13
- AC Switch (SW1) Settings
- Testing the installation

This section is intended to familiarize the 6440 user with the hardware adjustments and settings required to power up and operate the 6440 drive.

Introduction

The 6440 unit is a three board assembly incorporating a drive and an Indexer card set. With the cover removed, the topmost visible board is the Indexer. The Indexer mounts on the drive board and is separated by standoffs.

Drive

The Drive has an eight position DIP switch (S1) controlling drive current, digital electronic damping, and idle current reduction. The DIP switch (S1) is easily accessible without removing the Indexer card.

Indexer

The Indexer has thirteen plug on jumpers (E1 through E13) controlling RS Communications, Synchronous or Alternating Operation, and I/O Configuration.

Power

The power board has an AC switch (SW1) to select 120 or 240 Vac operation.

Warning

Connecting 240 (230) Vac with switch in 120 (115) Vac position will permanently damage the drive.

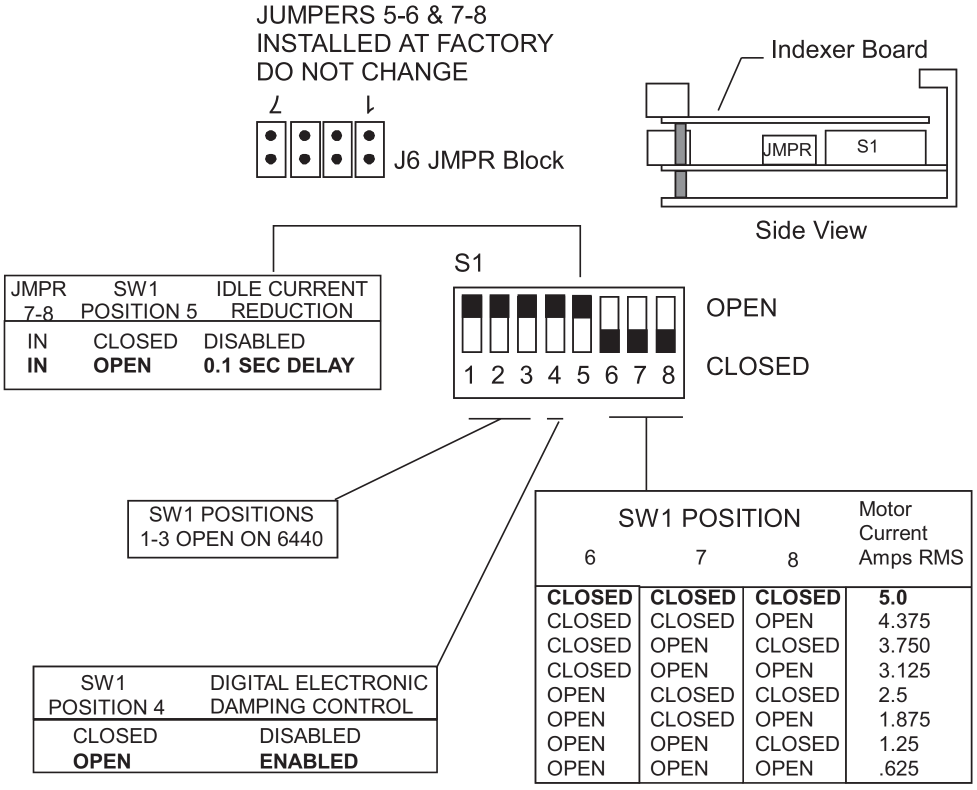
3.1 Setting Switch S1 on Drive Board

Introduction

DIP switch S1 sets the following:

- Motor current level
- Digital electronic damping ON/OFF
- Idle current reduction

Location of S1



JMPR 7-8	SW1 POSITION 5	IDLE CURRENT REDUCTION
IN	CLOSED	DISABLED
IN	**OPEN**	**0.1 SEC DELAY**

SW1 POSITIONS 1-3 OPEN ON 6440

SW1 POSITION 4	DIGITAL ELECTRONIC DAMPING CONTROL
CLOSED	DISABLED
OPEN	**ENABLED**

SW1 POSITION			Motor Current Amps RMS
6	7	8	
CLOSED	**CLOSED**	**CLOSED**	**5.0**
CLOSED	CLOSED	OPEN	4.375
CLOSED	OPEN	CLOSED	3.750
CLOSED	OPEN	OPEN	3.125
OPEN	CLOSED	CLOSED	2.5
OPEN	CLOSED	OPEN	1.875
OPEN	OPEN	CLOSED	1.25
OPEN	OPEN	OPEN	.625

3.1.1 Digital Electronic Damping Control

Definition

Mid-speed instability and the resulting loss of torque occurs in any step motor/drive system due to the motor back EMF modulating the motor winding currents at certain speeds. Mid-speed instability can be explained as a region of potential instability that occurs as a result of the electronic, magnetic, and mechanical characteristics of any stepping motor system. The circuitry used to control this phenomenon does so by advancing or delaying the switching of the output current with respect to the incoming pulse train. This should be taken into account if the user is attempting to employ pulse placement techniques.

Enable the digital electronic damping function by placing DIP switch S1 position 4 in the open position as shown. This is the default position and should be used for most applications if your application is affected by loss of torque at mid-range speeds. If pulse placement techniques are being used, disable the digital electronic damping function by placing DIP switch S1 position 4 in the open position.

Benefit

This feature controls torque loss at mid-range speeds. When enabled, the motor maintains torque at mid-range operation, provided the torque load does not exceed motor torque ratings.

3.1.2 Idle Current Reduction

Definition

The Idle Current Reduction (ICR) function reduces the phase current at times when no motion is commanded. Motor current is reduced when no step commands are received for a given time. This time can be set to 0.1 seconds. Current to both motor windings is reduced by one-half.

The ICR function can be enabled/disabled and the time delay between the last step command and current reduction can be set to 0.1 seconds using DIP switch S1 position 5. ICR is disabled when DIP Switch S1 position 5 is in the closed position and enabled with a delay of 0.1 second (current is reduced by 50% when no step command is received for 0.1 second when the switch is open.

Note: *When ICR is active, both the holding torque generated by the motor and the motor stiffness around the holding position are reduced by approximately 50%.*

Benefits

The ICR function:

- Reduces motor and drive heating during standby operation

3.1.3 Setting Motor Current

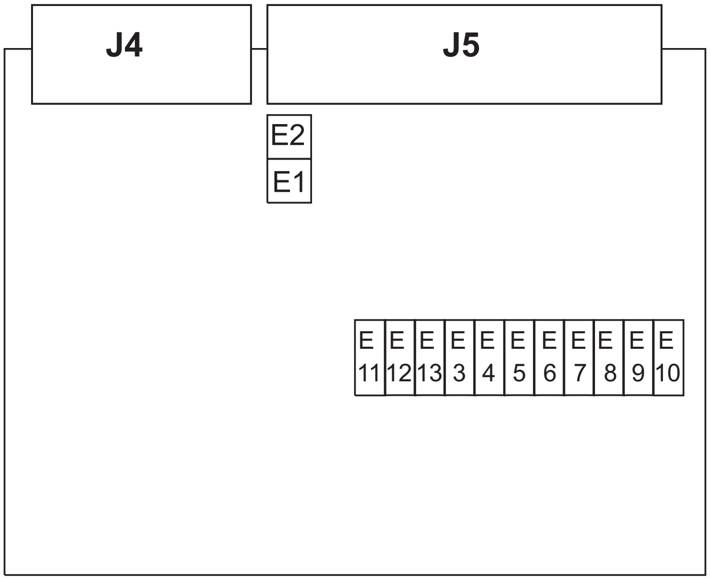
Motor current can be set using DIP Switch S1 positions 6, 7, and 8 as shown on page 3-2. Current should be compatible with motor current ratings.

3.2 Indexer Board Settings

Indexer

The Indexer has thirteen plug on jumpers (E1 through E13) controlling RS Communications, Synchronous or Alternating Operation, and I/O Configuration.

Jumper location



3.2.1 Jumper Settings

E1 and E2

Note: *These jumpers should NOT be modified by user.*

Configuration	E1	E2
	IN	OUT

E11

E11	RS Communication Configuration
IN	RS-485 Communications
OUT	RS-232/RS-422 Communications

Note: *The factory default is Jumper OUT enabling RS-232 and RS-422 communications.*

E12 and E13

E12	E13	Synchronous or Alternating Operation Configuration
IN	OUT	Alternating Operation
OUT	IN	Synchronous Operation

Note: *Factory default setting is E12 OUT, E13 IN.*

E3 to E10

Jumpers E3 to E10 control discrete I/O bit directions.

E3 to E10	I/O Configuration
IN	Output
OUT	Input

Note: *Factory default settings are all jumpers OUT configuring all lines as INPUTS.*

3.3 Setting AC Switch on Power Board

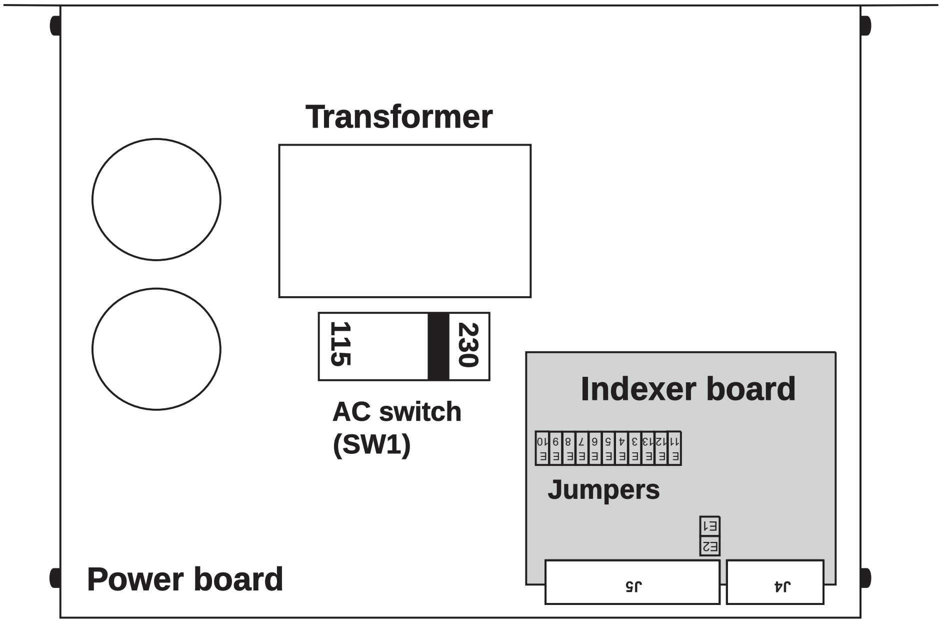
Introduction

The AC Switch (SW1) on the 6440 power board allows the user to select 120 OR 240 Vac.

Warning!

Connecting 240 (230) Vac with switch in 120 (115) Vac position will permanently damage the drive.

Location of AC Switch



Note: *AC Switch is preset at the factory in the 230 Vac position.*

3.4 Testing the Installation

Background

Perform the following test procedure to verify that the 6440 is installed properly and was not damaged internally during shipment.

Configuration

The installation and power-up procedure requires a motor and computer or terminal to test the basic functionality of the 6440.

Procedure

After performing the installation per the guidelines given in Chapter 2, "Installing the 6440," test your installation as follows.

Warning

Perform this initial power-up with the motor shaft disconnected from the load. Improper wiring or undiscovered shipping damage can result in undesired motor motion. Be prepared to remove power if excessive motion occurs.

3.4.1 Connections test

Introduction

Before beginning the connections test, please check the following:

- all wiring and mounting to verify correct installation
- specifications to ensure that voltages being applied do not exceed the voltages specified.

Procedure

1. Connect only J1 120 or 240 Vac, depending on the location of the AC Switch (SW1), to the 6440 then apply power.
2. Verify +5.7 Vdc at the 6440 J5-13 to J5-5 (voltage should be 5.2V to 5.9V). If it is not, refer to "Troubleshooting section."
3. Switch power OFF.
4. Connect the J3 motor connector.
5. Switch power ON again.
6. Connect the J4 RS-232 connector. Cycle power to the controller. If using a PC, use the 6440 Dialogue disk to set up your PC as a dumb terminal as described in Section 6.
7. Type ESC and then a single space character. The controller should respond with a predefined sign-on message:

   5134 PACIFIC SCIENTIFIC ©1994
   v1.10
8. Enable the 6440 by connecting J5-7 (Enable1) to J5-5 (GNDA).
9. Verify that the motor has holding torque by attempting to rotate the motor shaft. The energized shaft is either immovable or very resistant to rotation when the drive is enabled.
10. From the terminal or computer type, **R100<enter>**.

    This will move the motor at a velocity of 100 steps per second clockwise.
11. Type **R-100<enter>**. This will change the direction of the motor rotation.
12. Type **S <enter>**. The motor should stop rotating.

    If the controller passes all of the above, then you may start exercising the unit on your own. If the 6440 does not pass all of the above steps, refer to Section 4.2, "Troubleshooting."

4 Maintaining\Troubleshooting

In this chapter This chapter covers maintenance and troubleshooting of the 6440.

4.1 Maintaining the 6440

Introduction The 6440 package is designed for minimum maintenance. The following cleaning procedure as needed will minimize problems due to dust and dirt build up.

Procedure Remove superficial dust and dirt from the package using clean, dry, low-pressure air.

4.2 Troubleshooting the 6440

Introduction Use the following tables to diagnose and correct most problems. If you are unable to achieve satisfactory operation, contact your local Pacific Scientific distributor or Applications Engineering Department.

4.2.1 Troubleshooting the Power Board

Corrective action table Use the following table to trouble shoot the 6440's power supply.

SYMPTOM	POSSIBLE CAUSE	ACTION
Motor does not turn, LEDs ON (green and/or red)	120/240 Vac switch in 240 position, input from 120 Vac	Turn power off, correct switch position.
	AC Input line low	Increase Input AC to spec.
	Dead short or overload across external 66 Vdc output connector (J6).	Remove short or reduce load.
	Over temperature	Check ambient temperature or internal fan malfunction/blockage.
	Bad load connection	Check load connection. Check J6 Vdc output with a voltmeter and ensure output voltage is 66V ±3%. 1. If output voltage > 70 Vdc and < 78 Vdc add a load and ensure Vdc is ≈ 66Vdc. 2. If output voltage > 78 Vdc, return 6440 to factory for service.
	Drive board fault	See Section 4.2.2

Table (cont'd)

SYMPTOM	POSSIBLE CAUSE	ACTION
Motor does not turn, LEDs OFF	Check AC input	Use proper input.
	240 Vac applied and switch in 120 Vac position.	Return to factory for service.
Motor runs for a while and stops, both LEDs come on	Over temperature.	Reduce load. Check for excessive ambient temperature. Check for internal fan malfunction/blockage.
Motor turns on and off on its own and red LED flashes **OR** Motor stops after running once.	120 Vac applied and switch in 240 Vac position	Correct switch position.
	Over load.	Reduce load.
	AC input line low.	Check input AC line voltage for low line.
	Drive Board Fault.	See Section 4.2.2
	Internal failure.	Return to factory for service.

4.2.2 Troubleshooting the Drive Board

Corrective action table Use the following table to troubleshoot the drive board.

SYMPTOM	CORRECTIVE ACTION
Motor produces no torque.	Disconnect AC Power. Disconnect the motor cable and cycle the J1 power supply Off and On. Check the step output and VCO input monitor point. Also, check motor cable and motor for shorts across the windings or between the windings and the motor case.
	Verify that DIP Switch S1 position 6, 7, and 8 (current select) are set correctly. Re-check that the motor cable is wired correctly and properly plugged into the drive.
Motor rotates in the wrong direction.	Check polarity of the DIRECTION input. Reverse the A and $\overline{A}$ motor phases.
Motor does not reach expected position.	Check that the step size setting of the drive is set correctly. Verify that the motor does not stall. If it does: 1. Use a finer step size to avoid low-speed resonance problems. 2. Enable Digital Electronic Damping (S1 position 4 OFF).

4.2.3 Troubleshooting the Indexer Board

Corrective action table Use the following table to troubleshoot the indexer board.

SYMPTOM	CORRECTIVE ACTION
Motor doesn't spin	Verify that the motor is properly connected with no open wires or shorts. With the 6440 energized, try twisting the motor shaft. If you're able to easily spin the shaft, especially at the higher motor current levels, chances are the motor is disabled. J5-7 (ENABLE) should be wired to J5-5 (GND). Verify that some motor problem has not created a Fault condition. A FAULT has occurred if J5 pin 12 is low or the Read HW status command returns with Bit 7 high. If so, correct the condition that created the fault. If you issued the T command with Indexer jumpers E12 and E13 set for alternating operation, you may have disabled step pulses to the motor. Double check your jumper configuration and usage of the T command.
Motor spins, but tends to stall	Experiment with the M command by issuing M 255 255 which results in the slowest possible acceleration and deceleration. Possibly your motor cannot generate enough torque for your application. Try resizing.
Unit will not respond to commands	If the power supply is not adequate for the load, a momentary drop in power may reset the unit, and require that the sign-on procedure be repeated. It is possible a maximum velocity value, well above the specified range may have been executed. Try cycling power and resetting any erroneous parameters before issuing another motion command. If the error occurs in an AUTOSTART program, try ESCaping as soon as possible before the maximum velocity is attained. Cycle power.

4.2.4 Troubleshooting Communications Interface

Corrective action table Use the following table to troubleshoot the communications.

SYMPTOM	CORRECTIVE ACTION
No RS-232 Communication	Verify that the 6440 E11 jumper is OUT, placing the 6440 into single-unit, RS-232/RS-422 mode. Verify that the transmit of the host terminal is wired to the 6440 receive and vice-versa. Normally DTE (Data Terminal Equipment) devices transmit data on pin 2 and receive data on pin 3. Verify the serial port settings to be 9600, N, 1. Make sure your host is working properly. Disconnect the D-9 connector from the 6440 unit. Wire pins 2 and 3 of this connector together. Try typing characters on the keyboard, each character should be echoed back to the screen. If not, check your cabling and connectors and refer to your host reference guide.

If the 6440 is defective

If you cannot correct the drive problem, or if it is defective, return it to Pacific Scientific for repair or replacement.

Return procedure

1. Call Pacific Scientific at (815) 226-3100 from 8 am to 6 pm Eastern Standard Time to get a Returned Materials Authorization Number (RMA#).

Note: *Do not attempt to return the 6440 or any other equipment without a valid RMA#. Returns received without a valid RMA# will not be accepted and will be returned to the sender.*

2. Pack the drive in its original shipping carton. Pacific Scientific is not responsible or liable for damage resulting from improper packaging or shipment.

3. Ship the drive to:

Pacific Scientific

110 Fordham Road

Wilmington, MA 01887

Attn: Repair Department, RMA# ________

Note: *Do not ship Pacific Scientific motors to the above address. The correct address for motors is:*

Pacific Scientific

4301 Kishwaukee Street

Rockford, IL 61105

Attn: Stepper Repair Department, RMA# _____

Shipment of your drive or motor to Pacific Scientific constitutes authorization to repair the unit. Refer to Pacific Scientific's repair policy for standard repair charges. Your repaired unit will be shipped via UPS Ground delivery. If another means of shipping is desired, please specify this at the time of receiving an RMA#.

5 Programming Overview and Instruction Set Description

5.1 Modes of Operation

Introduction

The 6440 can operate in Immediate or Execution Mode. Immediate Mode is an interactive, command line mode where commands are entered and immediately executed. Immediate Mode is used during program development or when controlling the unit with a host computer. Execution Mode is used when the 6440 is executing a program out of its non-volatile memory, usually when used as a standalone Indexer. During power-up if the AUTOSTART program is found at location 1600 in non-volatile memory, execution will begin at 1600. If the REMOTE START line is activated (LOW) in Immediate Mode, the unit will start executing the program at location 0 in memory.

5.1.1 Immediate Mode

After power-up, providing that neither REMOTE START nor the AUTOSTART are active, the 6440 will be in Immediate Mode. If the 6440 is configured for RS-232/RS-422 serial communications, it will wait for one ESC and one space character and then respond with the Pacific Scientific sign-on message and copyright notice. Commands are sent via the serial link and are terminated with a carriage return <CR>. The 6440 responds with any requested data followed by a <CR><LF>. Only one command can be sent at a time. Motion commands will immediately echo a <CR><LF> and carry out the motion in the background. If a second motion command is sent to the 6440 while another is in progress, the <CR><LF> response will be delayed until the original command is complete.

Programs can be entered into non-volatile memory while in Immediate Mode. Either a terminal emulator program or the Pacific Scientific "6420 Dialogue" can be used to facilitate program development. The E address command allows you to edit a program starting at the location specified by address. The starting address is echoed indicating where the command will go in memory. You can continue to enter successive commands line by line. If an entry error occurs before <CR>, the backspace or delete key can be used to correct the error. If the error is noticed sometime later, leave the edit mode by entering E <CR> and start editing at the line where the error was made by entering E error_address followed by <CR> . If you want to selectively change only the line where the error was made, enter the new command followed by <CR> as you normally would do, but leave the edit mode by pressing the ESC key instead of E <CR>. If you inadvertently enter the E <CR> combination, the next command in the program will be over-written with an invisible program termination code and you will have to go back and selectively replace this line. If you want to continue editing the program from this point on, continue to do so and conclude editing with E <CR>. You can test the program by using the G address command to execute the code. The program will continue to execute until it completes or you press the ESC key.

If the 6440 is configured for multi-unit RS-485 communication, no sign-on message will appear. Commands are sent to individual units based on a pre-assigned node address (assigned in single-unit mode as described in the "Serial Communications Port - J4" section) followed by the command, terminated by a line feed <LF> character. Once a unit recognizes it's single-letter node address, that address character and commands entered up to the <LF> will be echoed back to the user.

5.1.2 Execution Mode

Once the program starts it will continue to execute until it is done, aborted with the S0 command embedded in the program, is aborted by the ESC or S keys or by the REMOTE STOP line.

5.1.3 The AUTOSTART Program

During power-up if a program is found at location 1600 in non-volatile memory, that program (the AUTOSTART program) will be executed. This program can be aborted by pressing the ESC or S keys, both forcing the unit back to the power-up state, waiting for the ESC and space character sign-on sequence.

5.2 Instruction Overview

Introduction

The 6440 offers a variety of instructions for motion and program control in addition to several utility instructions. All instructions can be entered on a command line basis in immediate mode with many, but not all available for use within programs executed from on-board non-volatile memory. The instruction categories include:

Motion Control

Incremental and absolute indexes, run at constant velocity and home to a known position. All the motion commands immediately echo <CR><LF> and carry out their motion in the background. The incremental and absolute indexes execute a trapezoidal velocity profile, accelerating from the initial velocity, accelerating to the final velocity, decelerating back to the initial velocity ending at zero velocity.

Program Control

Instructions for program control include P for program mode, G for executing programs and jumping within a program, J and B for executing interactive loops, U for conditional branches based in the input ports.

Parameter Initialization

These instructions include F to set the initial velocity, V to set the final velocity, M to set the acceleration and deceleration profiles, O to setup for a position trip point.

Initialization and Utility Commands

These commands include A to initialize memory, P to store current parameter values, E to edit a program.

5.3 Memory Map

User programs

Approximately 1792 bytes are available for user programs. The maximum size of a program in terms of number of command lines is dependent on the actual commands used since each command is composed of 1 to 5 bytes. In general, programs can be entered into the entire 1792 byte range of "User Programs" memory, however, certain commands when utilized, require specific areas of memory to be reserved for their use.

Fast RAM

In addition, there is a 64 byte segment, extending between locations 128 to 191 marked as "Fast Ram." This area is fast static RAM inside the microprocessor used when a specific command (the Trip and Output) requires it and when higher response or looping speed is necessary in an application. As the name implies, this area is NOT saved when the unit is powered-down. To save the program(s) in this area to non-volatile memory, you must issue the P command. Upon power-up the contents will be restored automatically.

Memory Map Details

Address	Description
0-255	User Programs. Location 0 is the starting location when the REMOTE START input is activated. Locations 128-192 are FAST RAM locations and are not saved when the unit is powered down until **P** (Store parameters) is issued.
256-511	User programs and G 2048 Indexed jump
512-767	User Programs
768-1023	User Programs
1024-1279	User Programs
1280-1535	User Programs
1536-1791	User Programs. Location 1600 is starting address for AUTOSTART.
1792-1893	RESERVED: Variable resolution fraction lookup table.
1894-2047	RESERVED: Motion parameter storage, may be read by the Read Memory command >.

6 6420 Dialogue

6.1 Description of 6420 Dialogue

Introduction

6420 Dialogue is a menu driven software package that contains several software utilities to aid in the use of Pacific Scientific Digital Motion Control Products. These utilities are designed to run on an IBM compatible PC and interface to the Pacific Scientific 6440 via an RS-232 serial link. The Main Menu summarizes the tools available:

1) Terminal Emulator

- The Terminal Emulator utility allows the PC to be used as a dumb terminal. In this mode, the PC is acting as a terminal and allows the generation and editing of programs directly on the hardware (i.e. on-line).

2) Upload Utility

- The Upload Utility allows files to be read out of the controller's memory into a file on the PC's floppy or hard disk drive.

3) Download Utility

- The Download Utility allows files to be taken from the PC's disk drive and transferred into the position controller's memory.

4) Syntax Checker

- The Syntax Checker allows programs to be checked for errors before transferring them to the controller.

5) Editor

- The Editor allows the creation and editing of programs. Programs can be created and edited without the controller being connected to the PC (i.e. off-line). The resulting program can be stored on the PC's disk drive for downloading to the controller at a convenient time. Also, programs can be read from the hardware using the Upload Utility and be modified using the Editor.

6) Init Serial Port

- The Init Serial Port option allows the user to select a COM Port for serial communication.

6420 Dialogue is contained on a single 3-1/2 inch diskette.

6.2 Getting Started

6420 Dialogue can be executed directly from the 6420 Dialogue disk supplied or it can be installed on the PC's hard disk. If 6420 Dialogue is to be executed directly off the disk supplied, it is recommended that a backup copy be made and stored in a safe place.

6.2.1 Using the 6420 Dialogue Floppy Disk

Follow the steps below to use the 6420 Dialogue floppy disk:

1. Boot up the PC
2. Set the PC to directory A:
3. Insert the 6420 Dialogue floppy disk into drive A
4. Type 6420 **<enter>**

6420 Dialogue will be loaded and the Main Menu screen will appear. 6420 Dialogue is now running and you can select the tool desired.

6.2.2 Installing 6420 Dialogue on a Hard Drive

The 6420 Dialogue disk is supplied with an installation program. This program will create a sub-directory named \6420 on the hard drive and copy the 6420 Dialogue files from the floppy disk into the sub-directory. The installation program assumes that the hard drive is designated C:.

To install 6420 Dialogue on the PC's hard drive, follow the steps below:

1. Boot up the PC
2. Insert 6420 Dialogue disk in drive A
3. Type A: INSTALL <**Enter**>

6420 Dialogue is now installed on the hard disk. To run 6420 Dialogue get to the 6420 directory, and type 6420 < **Enter** >.

6.2.3 Serial Port Connections

6420 Dialogue can be used to generate programs off-line. However at some point programs have to be downloaded or uploaded between the PC and the motion control hardware. The RS-232 serial link is used to communicate between the PC and the hardware.

Many PCs have two serial communication ports, COM1 and COM2. One of these must be wired to the motion control hardware. Refer to the PC's Hardware Reference Manual and the Instruction Manual provided with the motion control hardware for wiring information. Use the Init Serial Port tool on the Main Menu to select the correct COM port and to set the serial link parameters.

Note: *Default Serial Link Parameters for the 6440 are displayed.*

6.3 Keyboard Commands

The choices from a main menu are highlighted by typing the **<Up >** or **<Down>** arrow keys. Selection of the highlighted item is made by entering the choice with the **<Enter >** key. This selection will bring up a prompt, a list and/or a sub-menu. Choices are made from a sub-menu or list using the **<Up>** key to move up, or the **<Down >** key to move down, then entering a choice with the **<Enter >** key.

Typing the **<Esc>** key will abort the current operation, and return to the previous menu, or exit the program. The **<Esc>** key is also used to exit from the Editor. To exit from a Terminal Emulator session, type **<Ctrl><E>**.

The keyboard command definitions are displayed in appropriate places on the screen to assist the user.

6.3.1 Terminal Emulator

The PC can be used as a dumb terminal to communicate with the selected controller. The PC's serial port is initialized when 6420 Dialogue is started. The parameters can be changed with the Init Serial Port utility.

To exit the Terminal Emulator at any time, type **<Ctrl><E>**, which returns to the Main Menu.

6.3.2 Upload Utility

After selecting this utility, the user is prompted for what filename to name the uploaded file. By default, the uploaded file will be created in the current directory. If the user selects a filename that already exists, then it will be overwritten, thus it is good practice to use a different extension or filename for the uploaded file. In particular, the filename.ext of the source file that was downloaded should not have the same filename.ext as the uploaded file, or the source file contents will be replaced with the uploaded file. When choosing extensions for filenames, the use of ".LST" should be avoided. The ".LST" extension is used by the 6420 Dialogue program for the listing files created by the Syntax Checker.

Note: *The Upload Utility will not function properly if voids (empty addresses) exist between sections of code in the memory. Contact the factory for additional information.*

After a file has been successfully uploaded, the message "Upload completed" is displayed. If the **<Path>**\filename.ext is unacceptable, then the message "Couldn't open file" is displayed.

When uploading a file, labels will be generated for any branch to address command. Note that for a program that has been downloaded, the symbolic labels and references are stripped out and replaced with numeric addresses. At uploading, these have been replaced with numeric sequential labels. It is good practice to use a different extension or filename for uploaded files. Label generation for Pacific Scientific Indexer Language is covered in Section 6.7, "Upload Utility."

6.3.3 Download Utility

After selecting this utility, the user is prompted for a **<Path>**\filename.ext which can be typed in or selected from the current directory list. If the download is successful, the message "Download completed" is displayed.

6.3.4 Syntax Checker

By using the Syntax Checker, programs can be checked for syntax validity, prior to downloading or running.

Syntax Checker creates a listing file with the extension ".LST" given to the file that was sent to Syntax Checker or sent to the Download Utility (which automatically calls the Syntax Checker). If the syntax check is successful, the message "No syntax error(s) detected" is displayed and the listing file with the message "NO ERRORS FOUND" is sent to the display. If errors are found, then the message "Syntax error(s) detected" is displayed, and the listing file, containing the program code and syntax error messages, is displayed. The Editor can be used to review the syntax errors listed in the .LST file. The syntax checking for Pacific Scientific Indexer Language is discussed in Section 6.4.

To reposition the syntax error dialog box, press **F2** and then use the arrow keys to move to the desired position. Once the dialog box is positioned, press **F2** or <**Enter**> to see errors. To edit a line, you must return to the editor.

6.3.5 Editor

After selecting this utility, the user is prompted for a**<Path>**\filename.ext which can be typed in or selected from the current directory's file listing. If the filename does not exist, the user is prompted with "Create?" to which the response is **Y** (yes) or **N** (no). If answering "no", the system responds with the message "can't load file" (it doesn't exist), after which, a **<Enter>** returns to the Main Menu level. If answering "yes" to the prompt, the user is put into the Editor.

When exiting the Editor (typing **<Esc>**), the user is prompted with the question, "Save file (y/n) ?" Answering n **<Enter >** , returns to the Main Menu, without saving the file, and any edits are lost. Answering y **<Enter>** will bring up a prompt for the filename. The current file being edited can then be saved under the existing filename by typing **<Enter>** , or, the name can be changed by typing over the current filename, followed with **<Enter>** .

While using the Editor, many operations such as justify, cut, paste, etc. are available. These operations are assigned to various keys:

ESC	Exit the Editor.
INSERT	Toggles insert and typeover mode, (**default: typeover mode**).
F7	Attach file to the end of current file (**default: off**).
F8	Toggles whether hard and soft carriage returns are symbolically displayed in the edit window, (default: not displayed)
F9	Toggles editor display from 128 ASCII characters to 256 IBM characters, (**default: ASCII characters**).
F10	Reformats a paragraph.
UP ARROW	Moves cursor up a line.
DOWN ARROW	Moves cursor down a line.
LEFT ARROW	Moves cursor left a space.
RIGHT ARROW	Moves cursor right a space.
CTRL-A	Move cursor one word to the left.
CTRL-F	Move cursor one word to the right.
CTRL-Y	Delete line cursor is on.
CTRL-Z	Scroll up a line.
CTRL-W	Scroll down a line.
HOME	Move cursor to beginning of line.
END	Move cursor to the end of line.
PGUP	Previous Page.
PGDN	Move to top of screen.
CTRL-END	Move to bottom of screen.
CTRL-PGUP	Move to beginning of file.
CTRL-PGDN	Move to end of file.
TAB	Tab.

ALT-M	Begin marking a block of text, use cursor arrows to mark rest of block.
ALT-C	Copy marked block to scrap.
ALT-X	Cut marked block to scrap.
ALT-P	Paste scrap at cursor position.
ALT-S	Search for pattern (target); starts search, cancels search.

6.3.6 Init Serial Port

After selecting this utility, the serial port parameters can be initialized or changed. If the parameters do not need to be changed, then type **<Esc>.** Typing **<Esc>** at any point in the Init Serial Port menu will exit without saving any changes to the parameters.

The parameters are selected by using the up or down arrow keys. As each parameter is highlighted, its sub-menu displays the list of each parameter's values. The choices can be selected with the **<Up>** and **<Down>** keys. A parameter and its value are entered with a **<Enter>**.

To exit the Init Serial Port menu after making changes to any parameters, type **<Enter>** after the last parameter in the menu. Any changes made to the parameters are saved to a disk file in the current directory after exiting the Init Serial Port menu, and thus are used as the default settings, upon the next invocation of 6420 Dialogue. These default settings are stored in a binary data file named 'PORT.CFG'.

6.4 Indexer Language for the 6440 Series

Introduction

Pacific Scientific Indexer Language is a modified form of the program language of the Pacific Scientific Model 5240 Stepping Motor Indexer/Drive.

The language for the 6440 has been modified so that symbolic labels can be used as the operand for branch to address commands, for example “J”, the JUMP command. Syntax checking is done on the Indexer Language, and labels are translated to their target address by the Syntax Checker before a program is downloaded. When a program is uploaded, labels are generated for the operands of branch to address commands.

6.4.1 Use of Symbolic Labels in Programming

A line of program code has this format:

```
<LABEL>    COMMAND    OPERAND1    OPERAND2
```

where LABEL is optional, and OPERAND2 exists for some commands. One or more blanks must be used between a label and a command, or between operands, but blanks are not necessary between the command and OPERAND1. Tabs may be used in place of blanks. The branch to address commands, “G” (GOTO), “J” (JUMP) and “U” (LOOP), may have a label or an address for OPERAND1. The label reference in OPERAND1 must have a matching label somewhere in the program, or a syntax error is generated. Duplicate label names are not allowed and will cause a syntax error to be generated.

The format for a label is:

```
$ < STRING >
```

where there are no blanks between the dollar sign ($) and the string. The string is limited to eight characters composed of any sequence of digits and letters (including the underscore). Any variation from this format will generate a syntax error. The maximum number of labels allowed for any program is 200.

Note: *Labels are only used in the Editor Mode.*

Example

The labels in this program are “begin”, “loop”, and “end.”

```
$ begin     k
            j $ begin 1
$ loop      +1000
            u $ loop 5
            g $ end
$ end       s
```

6.5 Syntax Checker

The Syntax Checker checks for valid syntax for each line of code. If no syntax errors are found, the Syntax Checker takes out all symbolic labels and label references, and inserts the target address for each label reference, (a label reference is a label used as the operand for a branch to an address command).

If an error is found, the program and error messages are put into a listing file which is displayed. The possible error messages are listed below:

1. “invalid input; label or program command expected”
2. “invalid label; expecting alphanumeric after delimiter”
3. “invalid label or, expecting valid command after label”
4. “invalid operand; digit ‘1’ expected”
5. “invalid operand; expecting digit”
6. “EOL encountered; expecting operand”
7. “invalid input;no operand required”
8. “invalid input; expecting EOL character”
9. “invalid input; expecting valid number or label”
10. “illegal for this command to have a label reference”
11. “invalid number; input number is out of range”

Messages (cont'd)

12. "invalid operand; +,- character not allowed after +,- command"
13. "invalid operand; reference to nonexistent label"
14. "invalid label; duplicate label name exists"

Note: *The end of line character (EOL), which is defined as the linefeed character, is expected at the end of each line of code.*

6.6 Download Utility - Address Translation of Labels

Before a 6440 program is downloaded, it is automatically syntax checked. If there are no syntax errors, the Syntax Checker takes out all labels from the program and translates the label references to address. The program thus contains only commands and numeric operands, and is downloaded to the controller.

6.7 Upload Utility - Label Generation for Addresses

When a 6440 program is uploaded, labels are generated for the operands of branch to address commands. The labels are then inserted in the appropriate places in the program. The labels generated have the form $Lxxx, where the labels are sequentially generated starting with $L000.

Note: *If a 6440 program is written off-line on the PC with labels and then downloaded, when the program is uploaded the labels will take the form described above. The labels used in the program prior to downloading will exist only in the original file on the PC.*

To avoid confusion when developing 6440 programs, it is recommended that the labeling scheme used by the Upload Utility be followed. That is, when writing a 6440 program on the PC, use labels of the form $Lxxx starting with $L000 and progressing sequentially through the program. By following this practice, a program will not change labels when downloaded and then uploaded.

7 6420 Programming Instructions

In this chapter

This chapter contains detailed descriptions of the 6420 commands. The command and the mode of operation where it can be used is listed at the top of the page. Commands are then described using the following format:

Purpose - purpose of the instruction

Syntax - exact format of the command

Programming guidelines - pertinent information about using the command

Example - showing use of the instruction in a program segment

A (Clear and Restore)

A *opcode*

Immediate

Purpose

This command is used to initialize a portion of non-volatile memory. Non-volatile memory is segmented into 8 pages. The specified *opcode* determines precisely the page to be erased, reloaded or initialized.

Syntax

A *opcode*

Arguments

A0	Reloads the last saved parameters from NVRAM
A1 to A7	Erase the corresponding page in NVRAM
A8	Completely initializes NVRAM to default values.
A9	Initializes variable resolution fraction lookup table to default values. List of 100 eight-bit values starting at location 1792 in NVRAM

Programming guidelines

Memory Map Details:

Erase Cmmd	Address	Notes
A1	0-255	User programs. Location 0 for REMOTE START input. Locations 128-192 are Fast RAM locations.
A2	256-511	User programs and G 2048 Indexed jump
A3	512-767	User programs
A4	768-1023	User programs
A5	1024-1279	User programs
A6	1280-1535	User programs
A7	1536-1791	User programs. Location 1600 for AUTOSTART.
XX	1792-1893	RESERVED: Variable resolution fraction look-up-table
XX	1894-2047	RESERVED: Motion parameter storage, can be read by the Read Memory command

C (Read Position Counter)

C *arg*

Immediate

Purpose	Displays current position counter value. In Fixed Resolution Mode the value is an integer. In Variable Resolution Mode, a mixed integer/fractional value in range of ± 8,388,607.99 to steps. Specifying *arg* = 1, enables continuous echoing of position via the serial interface only in single-unit mode.
Syntax	C *arg*
Arguments	arg is 0, 1
Programming guidelines	This command is normally used when the 6440 is controlled by a host computer or with a terminal emulator. Type **Z <Enter>** followed by **C <Enter>**. The value 0 should be displayed. Index incrementally by typing **+200 <Enter>**. Type **C <Enter>** again and 200 should now be displayed.

E (Edit Program)

E *addr*

Immediate

Purpose

Allows entry of user programs for subsequent execution by the G (GO) command or execution of a program at location 0 with activation of the Remote Start input or AUTOSTART at power-up. The *addr* argument specifies the address at which to edit a new or existing program.

Syntax

E *addr*

Programming guidelines

Example

Program line	Explanation
E 100	Edit a program starting at location 100
M 10 10	
F 400	
V 1200	
+ 1000	
E	

Edit Program mode is entered by the E 100. Various commands are entered and terminated by <Enter>. When completed, the last E inserts a terminator byte at the end of the program, over-writing any pre-existing data. The ESC key instead of the last E will prevent over-writing the next command, which is useful for patching in corrections to an existing program when using a terminal emulator.

Program execution can be interrupted by Remote Stop, ESC, or the S key. After interruption of an AUTOSTART program, the sign-on procedure must be carried out.

F (Initial Velocity)

F *vel*

Immediate, Execution

Purpose Sets the initial velocity to *vel* pulses/second. In Fixed Resolution Mode the actual speed in steps per second is determined by the current step size.

Example If vel = 1000 and the step resolution is set for Full Steps (200 pulses/rev), then the shaft spins at 1,000 full steps/sec or 300 RPM. If vel = 1,000 and the step resolution is set for 1/8 Steps (1,600 pulses/rev), then the shaft spins at 125 full steps/sec (37.5 RPM)

In Variable Resolution Mode, the speed is determined by the VelScale factor.

Example If VelScale has been set by issuing \ 0 (the nominal value), and vel = 1000, then the shaft spins at 1,000 full steps/sec (300 RPM). If VelScale has been set by issuing \ 2, and vel = 1000 then the shaft spins at 250 full steps/sec (300 RPM * .25 = 75 RPM).

Syntax F *vel*

Arguments 0 ≤ vel ≤ 19,000

Related Commands V — Set final velocity

Programming guidelines

Program line	Explanation
F 300	set the initial velocity
V 3000	

G (Go)

G *addr* [*trace*]

Immediate, Execution

Purpose

Executes a user program at a specified *addr*. The optional trace argument allows execution tracing. The *addr* specifies the starting address of a program in the range of 0 to 1791. A special case is where *addr* = 2048, which executes an "indexed" jump based on the lower 4 I/O port bits to 16 locations within address locations 256 to 496.

Syntax

G *addr* [*trace*]

Arguments

0 ≤ addr ≤ 1791, or 2048 for indexed jump on inputs

trace = 0 or 1

Programming guidelines

```
Program line                 Explanation

    E0
    0    M 10 10
    3    F 300
    6    G 2048              execute Indexed Jump
```

Jump Table

The following jump table is used for an indexed jump which might be used for a rotary switch used to select 1 of 16 different operations.

Jump Loc	P4	P3	P2	P1
256	1	1	1	1
272	1	1	1	0
288	1	1	0	1
304	1	1	0	0
320	1	0	1	1
336	1	0	1	0
352	1	0	0	1
368	1	0	0	0
384	0	1	1	1
400	0	1	1	0
416	0	1	0	1
432	0	1	0	0
448	0	0	1	1
464	0	0	1	0
480	0	0	0	1
496	0	0	0	0

H (Home)

H *speed dir*

Immediate, Execution

Purpose

The Home instruction initiates a search for the home position. Home is executed with *dir* set for motion that will move the assembly towards the home switch at *speed* steps/second. When the home position is reached and the home input switch is activated, the motor will reverse direction and move at a speed equal to the Initial Velocity parameter until the home switch is deactivated. The activated level should be a logic “0,” and the deactivated level a logic “1.”

Syntax

H *speed dir*

Arguments

20 ≤ *speed* ≤ 19,000 steps/second

dir must be 1 for CCW or 0 for CW

Related Commands

F — set Initial Velocity

Programming guidelines

Example 1

The system is designed so that CW motion advances toward Home. The Home switch is not active and H 250 0 is executed: Motion will be CW at 250 steps/second until switch is activated, then motion flips to CCW at the Initial Vel (50 steps/second) until switch is deactivated and then motion stops.

Program line	Explanation
M 10 10	set accel/decel factors
F 50	set initial velocity
H 250 0	home

Example 2

The system is designed so that CCW motion advances toward Home. The Home switch is not active and H 250 1 is executed: Motion will be CCW at *speed* until switch is activated, then motion flips to CW at the Initial Vel (F) until switch is deactivated and then motion stops.

Program line	Explanation
M 10 10	set accel/decel factors
F 50	set initial velocity
H 250 1	home

I (Resolution Mode)

I *mode*

Immediate

Purpose

This instruction selects Fixed or Variable Resolution Modes. Fixed Resolution Mode allows a fixed step size to be selected by the \ (Step Size) command. Of course, fine step sizes result in proportionately slower stepping rates. Variable Resolution Mode allows high-speed, high-resolution indexes to be executed without compromising speed.

A mode value of 0 selects Fixed Resolution Mode. Stepping proceeds at fixed resolution selected by Step Size command (\).

A mode value of 1 selects Variable Resolution Mode. Step resolution is automatically adjusted during an index dependent upon Initial and Final Velocity parameters. Step resolution is 20,000 pulses/rev.

The Q command displays the resulting Initial and Final Velocity parameters and range of step sizes to be used in subsequent absolute and incremental indexes.

Syntax

I *mode*

Arguments

mode = *0* selects Fixed Resolution Mode

mode = *1* selects Variable Resolution Mode

Programming guidelines

Note: *Normally a preset operating parameter. Should not be incorporated into a non-volatile memory based program. Issue the P command to save changes to the operating mode.*

J, B (Jump Loop, Nested)

J *addr cntr*

B *addr cntr*

Execution

Purpose

These two commands allow implementation of a doubly nested loop structure. The primary "J" command and secondary "B" command can be nested. However, "J" is for jump outer loop while "B" is for jump inner loop. The *addr* specifies the loop's destination address. The *cntr* value plus 1 is the actual number of iterations. Therefore, *cntr* value must be set to the desired iterations minus 1. Either command can be used if only a single loop is required.

Syntax

J *addr cntr*

B *addr cntr*

Arguments

0 ≤ *addr* ≤ 1791

0 ≤ *cntr* ≤ 255 for 1 to 256 iterations

Programming guidelines

The following example illustrates a 2 axis system, where a master 6440 controls one axis and a slave 6410 drives the other axis. The program indexes the master axis a total of 4 times; and for each of these indexes, the slave axis 8 times.

Program line	Explanation
E0	
0 F 100	
3 T 1	Outer loop begins, use J
5 + 1000	
10 W 0	
13 T 0	enable slave axis
	Inner loop begins, use B
15 + 1500	index 1500 steps along slave axis
20 W 0	till complete
23 W 100	wait 1 second
26 B 15 7	Repeat slave axis motion 8 iterations
	End of inner loop
30 J 3 3	for every master index, repeated 4 times
	End of outer loop

K (Read Input Port)

K

Immediate

Purpose

Reads user defined I/O ports. The bits are active low, so a logic "0" applied to the input port reads back as a "1."

Syntax

K

Arguments

None. The bit weightings are:

Port 1 = 1	Port 5 = 16
Port 2 = 2	Port 6 = 32
Port 3 = 4	Port 7 = 64
Port 4 = 8	Port 8 = 128

Related Commands

Y — Write Port

Programming guidelines

To program an I/O port bit as an input, the respective I/O configuration jumper must be removed. Ports 1 through Port 8 are configured by jumpers E3 to E10 respectively. Remember, as a consequence of the internal processor wired "AND" design of the I/O port, a logic "1" inadvertently written to an I/O port configured as an input will override the actual input state and hence will always be read back as a logic "1." Therefore, an inactive level (logic "0") must be written to input bits.

Type **K <Enter>**. The value of P8-P1 should be displayed, with the least significant bit representing P1.

L (List Program)

L *addr*

Immediate

Purpose

Produces a listing of instructions starting from the specified *addr* returning up to 20 command lines. Programs will be listed until a program terminator byte is encountered.

Syntax

L *addr*

Arguments

0 ≤ *addr* ≤ 1791

Programming guidelines

Enter a program starting at location 0 in memory. Type **L 0.** The program should now be listed on the screen.

M (Accel/Decel Factor)

M *accel decel*

Immediate, Execution

Purpose

The *accel* and *decel* values determine acceleration and deceleration profiles respectively. Accel/Decel ramps are determined by lookup table and are NOT specified in terms of dv/dt. The Initial and Final Velocity parameters are used as indices into a lookup table to determine a range of step speeds to be applied to the drive. The time spent at any given step speed within the range is determined by the *accel* and *decel* values, and hence, determine how quickly the drive ramps up or down in speed. The arguments must be in the range of 5 to 255. Values towards the low end result in high acceleration rates and conversely, values towards the upper end result in lower acceleration rates. The current *accel* and *decel* values can be saved in non-volatile memory by issuing the P (store Parameters) command.

Syntax

M *accel decel*

Arguments

1 ≤ *accel* ≤ 255

1 ≤ *decel* ≤ 255

Related Commands

F — Set Initial Velocity

V — Set Final Velocity

Programming guidelines

Program line	Explanation
M 100 50	accel factor 100, decel factor 50

O (Trip Point)

O *position vaddr*

Execution

Purpose

Allows program to vector to the location specified by *vaddr* when the position counter equals the specified position value. The *vaddr* must be in the range of 0 to 255 and *position* in range of ±8,388,607. The *position* must be a whole integer value in both Fixed and Variable Resolution modes.

Trip Point service routines generally contain I/O instructions and new trip point specifications and should not contain index, wait or other time consuming instructions. A *position = 0* disables this function. For the trip to occur, the main program must be still actively running and not in idle mode.

Syntax

O *position vaddr*

Arguments

-8,388,607 ≤ *position* ≤ +8,388,607

0 ≤ *vaddr* ≤ 255

Related Commands

T — Set Trip Point

Programming guidelines

Program line	Explanation
E0	
0 Z	
1 O 5000 128	Trip at position 5000 to location 128
6 + 18000	
11 W 0	
14 G 0	Program still running
E	
E128	
128 Y 1	Activate P1 LOW
130 E	

P (Store Parameters)

P

Immediate

Purpose

Stores parameters to non-volatile memory. Values are restored to working memory upon power-up and by issuing the A0 command.

Syntax

P

Arguments

none

Programming guidelines

Type **P <Enter>**. All parameters and fast memory locations 128-191 will be saved to non-volatile memory in addition to the following parameters:

Initial Velocity (F)
Final Velocity (V)
Ramp slope (M)
Jog Speed (^)
Trip Point (O)
Resolution Mode
Limit Switch Polarity
RS-485 Node Address
User Programs

The default mode after memory initialization is Variable Resolution Mode at Full Speed.

Note: *Whenever a change in any of these parameters or code in Fast RAM locations 128-191 is to be saved during power cycling, the P command must be executed.*

Q (Examine Parameters)

Q

Immediate

Purpose

Displays current parameters and system settings. Returns a variable number of lines depending on mode of operation. Only in Single-User Mode.

Syntax

Q

Programming guidelines

Type **Q <Enter>**. The screen should show the following parameter information:

M accel, decel, I = *VR* or *FR*, na = axis name, O = *trip pos* / *trip addr*

F = *initial vel* (*vel/stepsize*), V = *final vel* (*vel/stepsize*), (rl = *accel length*)

The *initial vel* and *final vel* are the programmed initial and final velocity values set by the F and V commands. The actual velocities also depend on the selected StepSize and Resolution Mode. For instance in Fixed Resolution Mode with Half Step Size (\ 1), F=400, V=1000, the second line would appear as:

```
F = 400 (400/2) V = 1001 (1001/2) rl = 2
```

Notice that the displayed velocities may not be exactly those that where specified. This is a result of truncation and roundoff errors during integer division calculations. The values within the parentheses (400/2) and (1001/2) indicate the actual velocity values are 200 and 500.5 full steps/second because of the half step size selected.

Programming guidelines

In Variable Resolution Mode, Full Step Size (\ 0), F=400 and V=1000, the second line would appear as:

F = 400 (12800/32) V = 1001 (16168/16) rl = 2

The values (12800/32) and (16168/16) indicate the actual velocity values are 400 and 1000 full steps/second and the automatically chosen step sizes are fine as 1/32 step. The \ 1 command scales velocity and the chosen automatic step sizes producing the expressions (12800/64) and (16168/32) resulting in actual velocity values of 200 and 500 full steps/second.

R (Run at Constant Velocity)

R *vel*

Immediate, Execution

Purpose

The Run instruction ramps up or down to the specified *vel*, expressed in pulses/second. In Fixed Resolution Mode the actual speed in steps per second is determined by the current StepSize. So, if vel = 1000 and the step resolution is set for Full Steps (200 pulses /rev), then the shaft spins at 1000 full steps/sec or 300 RPM. If vel = 1000 and the step resolution is set for 1/8 Steps (1600 pulses/rev), then the shaft spins at 125 full steps/sec (37.5 RPM).

In Variable Resolution Mode, the speed is determined by the VelScale factor. So, if VelScale has been set by issuing \ 0 (the nominal value), and vel = 1000 then the shaft spins at 1000 full steps/sec or 300 RPM. If VelScale has been set by issuing \ 2, and vel = 1000 then the shaft spins at 250 full steps/sec or 75 RPM.

Note: *In Variable Resolution Mode, direction is determined by the sign of velocity.*

Syntax

R *vel*

Arguments

$20 \le |vel| \le 19{,}000$

Related Commands

M — set acceleration/deceleration factors

Programming guidelines

Program line	Explanation
M 10 10	set accel/decel factors
F 300	set initial velocity
R 100	accelerate to 100 steps/sec (CW)
W 1000	wait for 10 seconds
R -100	accelerate to 100 steps/sec (CCW)
W 1000	wait for 10 seconds
R 0	ramp down to 0 steps/sec

S (Stop)

S [*arg*]

Immediate, Execution

Purpose

Stops motion with ramping deceleration specified by the M instruction. If the Indexer is currently running a program in execution mode or a motion command, motion will cease with ramping deceleration. If embedded in a program an optional argument can be supplied to abort the current program in addition to stopping all motion. If *arg* = 1 , motion will cease but the program will continue, otherwise with *arg* = 0 the program will terminate, placing the indexer into immediate mode.

Syntax

S [*arg*]

Arguments

None

Programming guidelines

Embedded in a program, will stop motion and abort, returning to idle mode.

Program line	Explanation
S 0	or S to stop motion and abort program

Typing S while a program is running will abort the program and return to idle mode.

T (Master/Slave Control)

T *enb*

Immediate, Execution

Purpose This command is used when one 6440 Indexer/Drive controls motion of two axes, both at Fixed Resolution only. The 6440 is the master unit controlling a slave 6410 Drive. Jumpers E12 and E13 determine if the slave unit is running synchronous with the master, that is, identically executing index commands or in alternating fashion where motion is executed on one axis and then the other independently. Power-up state is T 1.

Syntax T *enb*

Arguments Synchronous Operation: E12 OUT, E13 IN

enb = 1 enables step pulses to both master and slave.

enb = 0 disables step pulses to both master and slave.

Alternating Operation: E12 IN, E13 OUT

enb = 1 enables master, disables slave.

enb = 0 disables master, enables slave.

Programming guidelines

The unit is configured for alternating operation. We alternately index each axis by :

Program line	Explanation
T 1	enable master indexing
+ 1000	index 1000 steps at current resolution
W 0	wait till done
T 0	enable slave indexing
- 2000	index at current resolution
W 0	wait till done

Note: To be meaningful, both units must be in Fixed Resolution mode.

U (Loop on Port)

U *addr cond*

Immediate, Execution

Purpose This command tests a single I/O port bit, P1 to P8 and jumps to the specified address *addr* if *cond* is true.

Syntax U *addr cond*

Arguments 0 ≤ *addr* ≤ 1791

The *cond* codes are:

0 => Jump if Port 1 high	1 => Jump if Port 1 low
2 => Jump if Port 2 high	3 => Jump if Port 2 low
4 => Jump if Port 3 high	5 => Jump if Port 3 low
6 => Jump if Port 4 high	7 => Jump if Port 4 low
8 => Jump if Port 5 high	9 => Jump if Port 5 low
10 => Jump if Port 6 high	11 => Jump if Port 6 low
12 => Jump if Port 7 high	13 => Jump if Port 7 low
14 => Jump if Port 8 high	15 => Jump if Port 8 low

Programming guidelines

Program line	Explanation
E0	
0 M 10 10	
3 F 300	
6 V 1000	
9 U 40 1	jump if Port 1 Low
13 U 100 3	jump if Port 2 Low
17 G 9	
E	
E40	
40 + 1000	CW 1000 steps
45 W 0	till done
48 U 48 1	wait for release
E	
E100	
100 - 1000	
105 W 0	
108 U 108 3	
E	

V (Final Velocity)

V *vel*

Immediate, Execution

Purpose

Sets the final velocity of an @ (absolute position) or (+ -) incremental move to vel pulses/sec. In Fixed Resolution Mode the actual speed is determined by the current StepSize. So, if *vel* = 1000 and the step resolution is set for Full Steps (200 pulses /rev), then the shaft spins at 1000 full steps/sec or 300 RPM. If *vel* = 1000 and the step resolution is set for 1/8 Steps (1600 pulses/rev), then the shaft spins at 125 full steps/sec (37.5 RPM).

In Variable Resolution Mode, the speed is determined by the VelScale factor. So, if VelScale has been set by issuing \ 0 (the nominal value), and *vel* = 1000 then the shaft spins at 1000 full steps/sec or 300 RPM. If VelScale has been set by issuing \ 2, and *vel* = 1000 then the shaft spins at 250 full steps/sec or 75 RPM.

This parameter does not affect the velocity supplied in the R (Run at constant Velocity), or that programmed for Jog or Home operations.

Syntax

V *vel*

Arguments

0 ≤ *vel* ≤ 19,000

Related Commands

F — Set Initial Velocity

Programming guidelines

Program line	Explanation
V 1000	set final velocity for +, - and @ indexes

W (Wait)

W *period*

Immediate, Execution

Purpose

Waits for a specified period of time. The argument *period* specifies the wait in terms of 10 millisecond periods, hence, the actual wait time is the *period* multiplied by 10 milliseconds. **Specifying** *period* **to be 0 is a special case that delays completion of the wait instruction until the end of the current motion command, such as an absolute or incremental index.**

Syntax

W *period*

Arguments

0 ≤ *period* ≤ 65,535

Related Commands

+, - and @ indexes

Programming guidelines

Note: *A non-zero wait period executed after issuing a motion command such as absolute or incremental index will immediately start counting down, in parallel to the motion command. Therefore, if the wait period is desired after the motion is completed, use a W 0 followed by the actual wait command.*

Program line	Explanation
W 100	wait 1 second
+ 100000	
W 0	wait till index complete

X (Read Limits)

X *arg*

Immediate

Purpose

This command returns the status of the limit, home and jog switches and the Drive Fault status. The value of *arg* specifies whether limit switch or input line information is returned.

Syntax

X *arg*

Arguments

Specifying *arg* = 0, returns 1 for Limit + active, 2 for Limit - active and 3 for both active. Bit 7 is active high if the drive has faulted.

Specifying *arg* = 1, returns a binary weighted value corresponding to the level of these signals:

1 => Home Input

32 => Jog - Input

64 => Jog + Input

128 => Drive Fault

Y (Write Output Port)

Y *port*

Immediate, Execution

Purpose

Writes to user defined I/O ports where the *port* specifies a binary-weighted port selection bit. The port bits are active LOW, e.g. Y 1 activates Port 1.

Syntax

Y *port*

Arguments

0 ≤ *port* ≤ 255

Y 0 deactivates all bits producing Logic HIGHs

Y 255 activates all bits producing Logic LOWs

Related Commands

K — Read Input Port

Programming guidelines

To program an I/O bit as an output, the respective jumper must be inserted. Ports1 to Port 8 are configured by jumpers E3 to E10 respectively. The bit weightings are as follows:

Port 1 = 1	Port 5 = 16
Port 2 = 2	Port 6 = 32
Port 3 = 4	Port 7 = 64
Port 4 = 8	Port 8 = 128

Note: *Avoid driving a port configured as an output by a low impedance input.*

Program line	Explanation
Y 0	Port 1 to 8 outputs off (open, pulled up)
Y 2	Port 2 output ON (sinking to ground)

Z (Zero Origin)

Z

Immediate, Execution

Purpose

Zeros position counter. This is normally done before a sequence of absolute moves and/or trip sequences.

Syntax

Z

Arguments

None

Related Commands

@ — Absolute Indexes

Programming guidelines

Program line		Explanation
0	Z	zero origin
1	@ 200	index 1 revolution
6	W 0	wait till done
9	W 100	wait 1 second
12	@ 0	index back to position 0
17	W 0	
20	G 0	loop

^ (Set Jog Speed)

^ *speed*

Immediate, Execution

Purpose Sets jog speed. This value is multiplied by 30 to determine speed in pulses/second. Acceleration is ramped. Deceleration is normally not ramped except in the case of activating both JOG inputs and releasing one. Jog inputs are active in idle mode, that is, when not in middle in entering a command and while not running a program.

Syntax ^ *speed*

Arguments $0 \le speed \le 255$

@ (Absolute Move)

@ position

Immediate, Execution

Purpose

Indexes the motor to an absolute position given by the internal position counter. In Fixed Resolution mode, the *position* specifies ±8,388,607 pulses. So, if *position* = 200 and StepSize is set for Full steps, the shaft spins 1 revolution. If *position* = 1600 and StepSize is set for 1/8 step (1600 pulses/rev) then the shaft also spins 1 revolution.

In Variable Resolution Mode, the *position* specifies ± 8,388,607.99 Full steps with 0.01 step size resolution (20,000 pulses/rev). If *position* = 200.00 then the shaft spins 1 revolution.

Syntax

@ *position*

Arguments

- 8,388,607≤ *position* ≤ + 8,388,607 In Fixed Resolution

- 8,388,607.99 ≤ *position* ≤ + 8,388,607.99 In Variable Resolution

Related Commands

M — set acceleration/deceleration profile

F — set initial velocity

V — set final velocity

I — set Resolution Mode

\ — set StepSize

Z — Zero Position Counter

Programming guidelines

Program line	Explanation
M 10 10	
F 300	
V 1000	
Z	zero the position counter
@ 2000	index to position 2000
W 0	wait till index done
W 150	wait 1.5 seconds
Y 1	activate Port 1 (output LOW)
@ 0	index back to initial position
W 150	wait 1.5 seconds
Y 0	deactivate Port 1 (output HIGH)

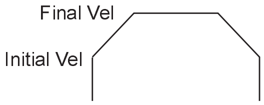
+ (Positive Incremental Move)

+ *steps*

Immediate, Execution

Purpose Moves the motor a total of *steps* in the CW direction with a trapezoidal velocity profile shown below. In Fixed Resolution mode, the *steps* specifies ± 8,388,607 pulses. So, if *steps* = 200 and StepSize is set for Full steps, the shaft spins 1 revolution. If *steps* = 1600 and StepSize is set for 1/8 step (1600 pulses/rev) then the shaft also spins 1 revolution.

In Variable Resolution Mode, the *steps* specifies ± 8,388,607.99 Full steps with 0.01 step size resolution (20,000 pulses/rev). If *steps* = 200.00 then the shaft spins 1 revolution.



Syntax + *steps*

Arguments 0 ≤ *steps* ≤ + 8,388,607 (Fixed Resolution)

0.00 ≤ *steps* ≤ + 8,388,607.99 (Variable Resolution)

Related Commands

M — set accel/decel

F — set initial velocity

V — set final velocity

I — set resolution mode

\ — set StepSize

Programming guidelines

Program line	Explanation
M 10 10	set accel/decel factors
F 300	set initial velocity
V 1000	set final velocity
+ 1000.52	index 1000.52 steps

- (Negative Incremental Move)

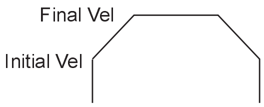
- *steps*

Immediate, Execution

Purpose

Moves the motor a total of *steps* in the CCW direction with a trapezoidal velocity profile shown below. In Fixed Resolution mode, the steps specifies ± 8,388,607 pulses. So, if *steps* = 200 and StepSize is set for Full steps, the shaft spins 1 revolution. If *steps* = 1600 and StepSize is set for 1/8 step (1600 pulses/rev) then the shaft also spins 1 revolution.

In Variable Resolution Mode, the *steps* specifies ± 8,388,607.99 Full steps with 0.01 step size resolution (20,000 pulses/rev). If *steps* = 200.00 then the shaft spins 1 revolution.



Syntax

- *steps*

Arguments

0 ≤ *steps* ≤ + 8,388,607 (Fixed Resolution)

0.00 ≤ *steps* ≤ + 8,388,607.99 (Variable Resolution)

Related Commands

M — set accel/decel

F — set initial velocity

V — set final velocity

I — set resolution mode

\ — set StepSize

Programming guidelines

Program line	Explanation
M 10 10	set accel/decel factors
F 300	set initial velocity
V 1000	set final velocity
- 1000	index 1000 steps at current resolution

\ (StepSize, VelScale)

\ *arg*

Immediate, Execution

Purpose

In Fixed Resolution Mode sets the StepSize. In Variable Resolution Mode determines VelScale to scale actual shaft speed.

Syntax

\ *arg*

The resolution settings also depend on the drive settings. The factory default is binary step sizes.

Arguments

Fixed Binary	VR Speed
0 = Full	Full speed
1 = 1/2	1/2
2 = 1/4	1/4
3 = 1/8	1/8
4 = 1/16	1/16
5 = 1/32	1/32
6 = 1/64	1/64
7 = 1/128	1/128
8 = 1/256	1/256

Programming guidelines

In the following program, assume the unit is set for Fixed Resolution Mode.

Program line	Explanation
E0	
0 \0	Full steps (200 steps/rev)
2 + 1000	spin CW 5 revolutions
7 W 0	
10 W 100	
13 \2	Quarter steps (800 steps/rev)
15 - 1200	spin CCW 1.5 revolutions
20 W 0	
23 W 50	

In Variable Resolution Mode:

0 \0	Full speed
2 + 1000	spin CW 5 revolutions
7 W 0	
10 W 100	
13 \2	1/4 speed
15 - 1200	spin CCW 6 revolutions
20 W 0	
23 W 50	

ESC (Abort)

ESC

Immediate

Purpose Terminates any active operation and returns Indexer to the immediate or idle mode. If motor is in motion, it will be immediately stopped without ramping deceleration. Output ports will not be affected. In RS-485 multi-unit mode motion on all axes is stopped.

Syntax

Arguments None

^ C (Software Reset)

^ C

Immediate

Purpose	Stops all axes, loads default parameter values, zeros origin, executes any user program (AUTOPROGRAM) at location 1600. Indexer will wait for the spacebar sign-on or external REMOTE START or JOG+ or JOG- inputs.
Syntax	
Arguments	None

> (Read Memory)

> addr size

Immediate

Purpose	Displays a specified block of non-volatile memory starting at *addr*, returning a total of *size* bytes. The value of *addr* must be in the range of 0 to 2047 and block *size* 0 - 255. Displayed values are in decimal format.
Syntax	> *addr size*
Arguments	0 ≤ *addr* ≤ 2047 and 0 ≤ *size* ≤ 255

< (Write Memory)

< addr data

Immediate

Purpose Writes a *data* byte to an address specified by *addr* in non-volatile memory. The *addr* must be in range of 0 to 2047 and *data* must be in range of 0 to 255.

Syntax < *addr data*

Arguments 0 ≤ *addr* ≤ 2047 and 0 ≤ *size* ≤ 255

] (Read Moving Status)

]

Immediate

Purpose Returns an integer number representing the current move status. Bit weights:

1 => Indexing

2 => Constant Speed

128 => Drive Fault

Syntax]

Arguments None

‘ (Trip and Output)

‘nextpos port

Execution

Purpose

This instruction is designed to be used within a Trip Point Service Routine (TPSR) and provides the capability of setting or clearing user I/O lines as a function of current position. To use this instruction, set up a Trip Point (O command) in the normal fashion, but with an entry address in fast RAM locations 128 to 191, such that the entire TPSR resides within the fast RAM range. When the Trip and Output instruction is executed, the port parameter is sent out to the bi-directional ports. The position parameter nextpos sets the next trip point position. The Q command always shows the next trip point. Trip and Output instructions must be embedded within the TPSR in contiguous locations. At end of sequence, a new Trip Point can be specified.

Syntax

‘ *nextpos port*

Arguments

None

Programming guidelines

The following example activates ports P1, P2, P3, P4 in sequence as successive positions are traversed. The O command sets the initial trip point to position 1000 and vector address to 128. When this position is encountered, execution vectors to location 128. The Trip and Output instruction at location 128 writes a 1 to the bi-directional ports and sets the next trip point to position 5000 and the vector address 133. Subsequent execution continues in a similar manner.

Note: *Only the E command at the conclusion of editing is necessary to end the trip point service routine.*

```
Program line                Explanation

    E0
    0    Z                  zero position
    1    O 1000 128         trip at pos 1000
    6    + 50000            index
    11    W 0               till done
    14   W 100              wait 1 sec
    17     G 0              loop


    E128
    128   ‘ 5000 1
    133   ‘ 10000 2
    138   ‘ 15000 4
    143   ‘ 20000 8
    148   ‘ 25000 4
    153   ‘ 30000 2
    158   ‘ 35000 1
    E
```

= (Limit Switch Polarity)

= *polarity*

Immediate

Purpose Sets the limit switch polarity. The default limit input upon power up is active LOW.

Syntax = *polarity*

Arguments

1 = Active Low

0 = Active High

: (Selective Termination)

: *axis*

Immediate

Purpose Aborts operations on a particular axis when configured for RS-485 communications mode.

Syntax : *axis*

Arguments Where *axis* is the single-letter axis designation.

Appendix A Specifications

Electrical

Input voltage	120/240 Vac (+10%, -15%) 60/50 Hz	
Rated drive current (motor phase current)	Setting	
	5 A	5A ± 0.25A
	4.375	4.375 ± 0.2A
	3.75	3.75 ± 0.2 A
	3.125	3.125 ± 0.15 A
	2.5	2.5 ± 0.15 A
	1.875	1.875 ± 0.125
	1.25	1.25 ± 0.125
	0.625	0.625 ± 0.1 A
Fuse	5 A Slo-Blo 250 Vac	
Drive circuit	Two-phase bipolar, chopper current regulated	
Chopper frequency	20 KHz, nominal	
Discrete Input Voltage	0-30 Vdc max, $V_{in} \leq 0.8$ V is a logic low and $V_{in} \geq 3.7$ V is a logic high.	
Discrete Output Voltage	Open collector Darlington, 0-30 Vdc max, 70 mA sink, $V_{sat} \leq$ 1.0 Vdc.	

Specifications

Electrical isolation

Indexer option power and interface signals to 6410 completely isolated. All other external inputs and outputs referenced to Indexer ground, unless otherwise specified.

Fixed Resolution Mode

All indexing is specified in terms of an integer number of step pulses at the specified resolution. This mode must be used if a single 6440 indexer controls its own drive and a slave(s) 6410 drive.

Step Size	Minimum Speed RPM (full steps/sec)	Maximum Speed RPM (full steps/sec)	Resolution Pulses/revolution
Binary Steps			
Full	6 (20)	5700 (19,000)	200
1/2	3 (10)	3000 (10,000)	400
1/4	1.5 (5)	1500 (5000)	800
1/8	.75 (2.5)	750 (2500)	1,600
1/16	.37 (1.25)	375 (1250)	3,200
1/32	.19 (.625)	188 (635)	6,400
1/64	.09 (.312)	93.75 (312)	12,800
1/128	.05 (.156)	46.87 (156)	25,600
1/256	.02 (.078)	23.43 (78)	51,200

Variable Resolution Mode

The indexer varies the step size from Full to 1/256 as a function of the specified velocity/position profile. All indexing is specified in terms of a mixed integer/fractional number, with the integer portion specifying the full number of steps and the fractional portion specifying 0.01 to 0.99 decimal step resolution carried out to within 8 binary bits of precision.

Step Size	Minimum Speed, RPM (full steps/sec)	Minimum Speed, RPM (full steps/sec)	Resolution Pulses/revolution
Variable, automatically chosen. User specifies position to within 1/100 of a step.	.02 (.078)	5,700 (19,000)	20,000

Communications RS-232 or RS-422/RS-485 at 9600 bits per second, 1 stop bit and no parity. The RS-485 mode supports both single axis and multi-drop, multi-axis operation.

Multi-axis capability Multiple 6440 units may be connected to a single RS-485 bus for multi-axis operation. Another alternative is to use the Step and Direction outputs to control additional drives such as 6410s. The 6440 is designed to directly drive another 6410 drive with no glue logic. However, if desired, the user can implement external logic controlled by any number of the user bi-directional I/O lines to control more than one drive.

Programming The 6440 is programmed with a simple mnemonic programming language. The unit operates in Immediate and Execution Modes. In Immediate, command line entry mode, the user can interactively specify motor commands, read back status information or enter programs with an ASCII terminal or terminal emulator. The host can also read back the status of both dedicated and user-defined I/O lines connected to external switches and sensors. In Execution Mode, previously entered programs may be executed on the 6440 to control the motor. On-board non-volatile memory offers approximately 1792 bytes for user programs. Program mnemonics vary from 1 to 5 bytes in length. Within the 1792 range there exists a 64 byte region from 128 to 191 designated as “FAST” memory for time critical code sections.

Environmental Requirements

Storage temperature	-40°C to +70°C
Operating temperature	0° to 50° C ambient air
Humidity Range	10 to 90%, non-condensing

Mechanical

Dimensions	6.25" x 2.25" x 12.50"
Weight	6 lb.. nominal

Connectors

Discrete I/O	25 pin male D connector
Serial Port	9 pin male D connector
AC Power	3 pin PCD ELFP03210
66 Vdc Output	3 pin PCD ELVP03100
Motor	5 pin PCD ELVP05100

Appendix B Ordering Information

Background

This appendix lists 6440 part numbers and gives information on ordering.

6440 part number table

Part	Pacific Scientific Order #	Comment
Stepper Drive	6440	
Connector Kit	CK6440	25-pin D connector
		9-pin D connector
		5-pin PCD
		3-pin PCD
		3-pin PCD
6420 Dialogue	904-008101-00	3 1/2 inch
User Manual	MA6440	
Motor Cable	SPC-xxx-6410 SPC-xxx-6410-KN*	xxx represents length in feet; for example, SPC-005 is a cable 5 feet long. For lengths over 50 feet contact Pacific Scientific. The connectors are MS on the motor end and PCD on the drive end to connect to Pacific Scientific motors.

***Note:** *Cables for K and N series stepper motors.*

How to order	Contact Pacific Scientific to order these parts.
Call	815-226-3100 from 8am to 6pm Eastern Standard Time.
Write	Pacific Scientific 4301 Kishwaukee Street Rockford, IL 61105
Fax	(815) 226-3048

Appendix C Quick Reference

In this chapter This section contains an alphabetized list of 6440 programming commands including a brief description and the operating modes where they apply. Detailed descriptions of each command are given on the page indicated in Chapter 7.

Command	Description	Mode	Page #
+ steps	Incremental move +	Immediate/Execution	7-40
- steps	Incremental move-	Immediate/Execution	7-42
:axis	Selective Termination	Immediate	7-54
<addr data	Write Non-volatile Memory	Immediate	7-49
=polarity	Limit Polarity	Immediate	7-53
>addr size	Read Non-volatile Memory	Immediate	7-48
@position	Absolute move	Immediate/Execution	7-38
\step size	Stepsize VelScale	Immediate/Execution	7-44
]	Read Moving Status	Immediate	7-50
^ speed	Set Jog Speed	Immediate/Execution	7-37
^ C	Software Reset	Immediate	7-47
‘nextpos port	Special Trip	Execution	7-51
A opcode	Clear and Restore	Immediate	7-2
B addr cntr	Jump inner loop	Execution	7-13
C arg	Read Position Counter	Immediate	7-4
E addr	Program Mode	Immediate	7-5
ESC	Abort or Terminate	Immediate	7-46
F vel	Initial Velocity	Immediate/Execution	7-6
G addr [trace]	Go	Immediate/Execution	7-8
H speed dir	Home	Immediate/ Execution	7-10

Table cont'd

Command	Description	Mode	Page #
I mode	Resolution Mode	Immediate	7-12
J addr cntr	Jump outer loop	Execution	7-13
K	Read Input Port	Immediate	7-15
L addr	List program	Immediate	7-16
M accel decel	Ramp Slope	Immediate/Execution	7-17
O position vaddr	Trip Point	Execution	7-19
P	Store Parameters	Immediate	7-21
Q	Examine Parameters	Immediate	7-22
R vel	Run at constant velocity	Immediate/Execution	7-24
S [arg]	Stop	Immediate/Execution	7-26
T enb	Master/Slave Control	Immediate/Execution	7-27
U addr cnd	Loop on Port	Immediate/Execution	7-29
V vel	Final Velocity	Immediate/Execution	7-31
W period	Wait	Immediate/Execution	7-32
X arg	Read Limits	Immediate	7-33
Y port	Write Output Port	Immediate/Execution	7-34
Z	Zero Origin	Immediate/Execution	7-36

Appendix D Ramp Algorithm & Lookup Table

Introduction

The 6440 Indexer/Drive uses a lookup table to determine the number of steps in the velocity ramp that occur from the initial (start/stop speed) velocity to the final velocity. The lookup table contents are included at the end of this appendix.

The velocity profile is a quantized linear ramp with discrete velocities stored in a lookup table. The velocity commands are in pulses per second.

The algorithm begins at the exact start/stop velocity entered, then runs at the next highest velocity in the lookup table. The velocity increments until it reaches the value immediately before the final velocity entered. The next velocity will be the final velocity.

The number of pulses output at each velocity is determined by the **M**(slope) command. The **M** value sets the number of pulses at each velocity.

For example:

```
M5
F300
V3000
```

The start/stop velocity and table velocities are:

155	562	820	1036	1227	1407	1581	1742	1905	2058
2190	2340	2472	2620	2736	2864	3004			

Five pulses will be generated at each of these rates as set by the **M** 5 command. The total ramp time from initial to final velocity is given by the sum of the times at each velocity during ramp up.

$$\text{Ramp time} = M * \sum_{V=\text{start/stop velocity}}^{V=\text{Last table entry before final velocity}} {}^{1}\!/_{V}$$

For example, ramp time for the above is:

Ramp time = 5*

$$\left[\frac{1}{300}+\frac{1}{562}+\frac{1}{820}+\frac{1}{1036}+\frac{1}{1227}+\frac{1}{1407}+\frac{1}{1581}+\frac{1}{1742}\right.$$

$$\left.\frac{1}{1905}+\frac{1}{2058}+\frac{1}{2190}+\frac{1}{2340}+\frac{1}{2472}+\frac{1}{2620}+\frac{1}{2736}+\frac{1}{2864}\right]$$

Divide command

The divide command (\n) can be used to modify the ramp profile. The divide command allows you to add more points to the velocity ramp, resulting in smaller velocity increments and smoother ramping.

All commanded velocities are divided by the (\n) command. When using the (\2) command with the previous example, the actual output pulse rate would be divided by 2. The initial velocity (Fnn) will be 150 pulses per second and the final velocity (Vnn) will be 1500 pulses per second. The new, modified ramp time will be :

Ramp time = 5 *2

To run between the same start and final velocities as the original example (300 to 3000) with (\2) command, the program parameters will be:

```
\2
M 5
F 600
V 6000
```

Velocities are found from the lookup table:

155	562	820	1036	1227	1407	1581	1742	1905	2058
2190	2340	2472	2620	2736	2864	3004	3126	3225	3348
3461	3582	3690	3804	3925	4028	4137	4251	4342	4468
4568	4672	4744	4856	4974	5056	5141	5273	5365	5461
5560	5610	5715	5823	5936	5994				

The velocity ramp now consists of 46 points compared to the initial 14 without the (\2) command.

$$\text{Ramp time} = 5 * \left[\frac{1}{300}+\frac{1}{721}+\frac{1}{1054}+\cdots+\frac{1}{5994}\right]$$

Note: *If the final velocity is below 721 sps (the second speed in the lookup table) there will be NO ramping associated with that move. Instead, the move is completed in one step.*

Example

If Initial Velocity were specified at 100 steps/sec, Final Velocity at 1000 steps/sec, acceleration factor 120, deceleration factor 20, referring to the Profile Generator Table shown on the following page, the intermediate velocity points would be 155, 562 and 820 steps/sec. During acceleration 120 steps at 100 steps/sec, followed by 120 steps each at 155 steps/sec, 562 steps/sec, 820 steps/sec, concluding with the Final Velocity at 1000 steps/sec for the duration of the index until deceleration commences. One exception to this would be if the specified duration of the index (in steps) were not long enough to allow the motor to reach the Final Velocity; then the profile would top out at some intermediate point followed by the deceleration interval.

Profile Generator Lookup Table

155	562	820	1036	1227	1407	1581	1742	1905	2058
2190	2340	2472	2620	2736	2864	3004	3126	3225	3348
3461	3582	3690	3804	3925	4028	4137	4251	4342	4468
4568	4672	4744	4856	4974	5056	5141	5273	5365	5461
5560	5610	5715	5823	5936	5994	6113	6174	6301	6366
6433	6501	6606	6678	6788	6864	6942	7021	7271	7358
7447	7538	7585	7680	7777	7826	7927	7979	8084	8137
8246	8302	8416	8474	8533	8593	8714	8777	8840	8904
8969	9035	9170	9239	9309	9380	9452	9525	9600	9675
9752	9752	9830	9909	9990	10072	10155	10240	10240	10326
10413	10502	10593	10593	10685	10778	10778	10874	10971	10971
11070	11170	11170	11273	11377	11377	11592	11702	11702	11815
11815	11930	11930	12047	12166	12166	12288	12288	12412	12412
12538	12538	12538	12668	12668	12800	12800	12934	12934	13072
13072	13072	13212	13212	13356	13356	13356	13503	13503	13653
13653	13653	13806	13806	13806	13963	13963	14124	14124	14124
14288	14288	14288	14456	14456	14456	14456	14628	14628	14628
14804	14804	14804	14985	14985	15170	15170	15170	15360	15360
15360	15360	15554	15554	15554	15554	15753	15753	15753	15753
15958	15958	15958	15958	16168	16168	16168	16168	16168	16384
16384	16384	16384	16605	16605	16605	16605	16605	16832	16832
16832	16832	16832	17066	17066	17066	17066	17066	17307	17307
17307	17307	17307	17307	17554	17554	17554	17554	17554	17554

17808	17808	17808	17808	18070	18070	18070	18070	18070	18070
18340	18340	18340	18340	18340	18340	18340	18618	18618	18618
18618	18618	18618	18618	18618	18904	18904	18904	18904	18904
18904	18904	19200	19200	19200	19200	19200	19200	19200	19200
19200	19504	19504	19504	19504	19504	19504	19504	19504	19819
19819	19819	19819	19819	19819	19819	19819	19819	19819	20144
20144	20144	20144	20144	20144	20144	20144	20480	20480	20480
20480	20480	20480	20480	20480	20480	20480	20480	20480	20827
20827	20827	20827	20827	20827	20827	20827	20827	20827	20827
20827	21186	21186	21186	21186	21186	21186	21186	21186	21186
21186	21186	21186	21186	21186	21557	21557	21557	21557	21557
21557	21557	21557	21557	21557	21557	21557	21557	21557	21942
21942	21942	21942	21942	21942	21942	21942	21942	21942	21942
21942	21942	21942	21942	21942	21942	21942	21942	22341	22341
22341	22341	22341	22341	22341	22341	22341	22341	22341	22341
22341	22341	22341	22341	22341	22341	22341	22341	22341	22341
22341	22755	22755	22755	22755	22755	22755	22755	22755	22755
22755	22755	22755	22755	22755	22755	22755	22755	22755	22755
22755	22755	22755	22755	22755	22755	22755	22755	22755	22755
22755	23184	23184	23184	23184	23184	23184	23184	23184	23184
23184	23184	23184	23184	23184	23184	23184	23184	23184	23184
23184	23184	23184	23184	23184	23184	23184	23184	23184	23184
23184	23184	23184	23184	23184	23184	23184	23184	23184	23184
23184	23184	23184	23184						

D.2 Speed Accuracy

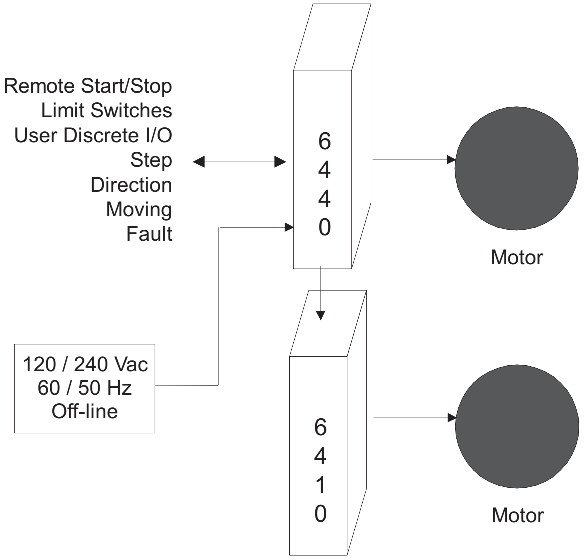
The 6440 Indexer generates step pulses by initializing a counter with a value that generates interrupts at approximately the desired step rate. The counter runs at a frequency of 1.2288 MHz. This value is divided by the desired step rate to yield the integer counter initialization value. Consequently, the rounding process results in reduced accuracy as speeds increase. For example, step rates from 2498 to 2502 steps/second yield a 491 count value and result in an actual speed of 2503 steps/second. Likewise, 2503 to 2507 steps/second yield a 490 count and result in an actual speed of 2508 steps/second. In summary, speed accuracy over the full 20-19,000 step/second range is approximately 0.72 %.

Appendix E Application Examples

Introduction The following examples give a flavor of just a few of the myriad applications for the 6440.

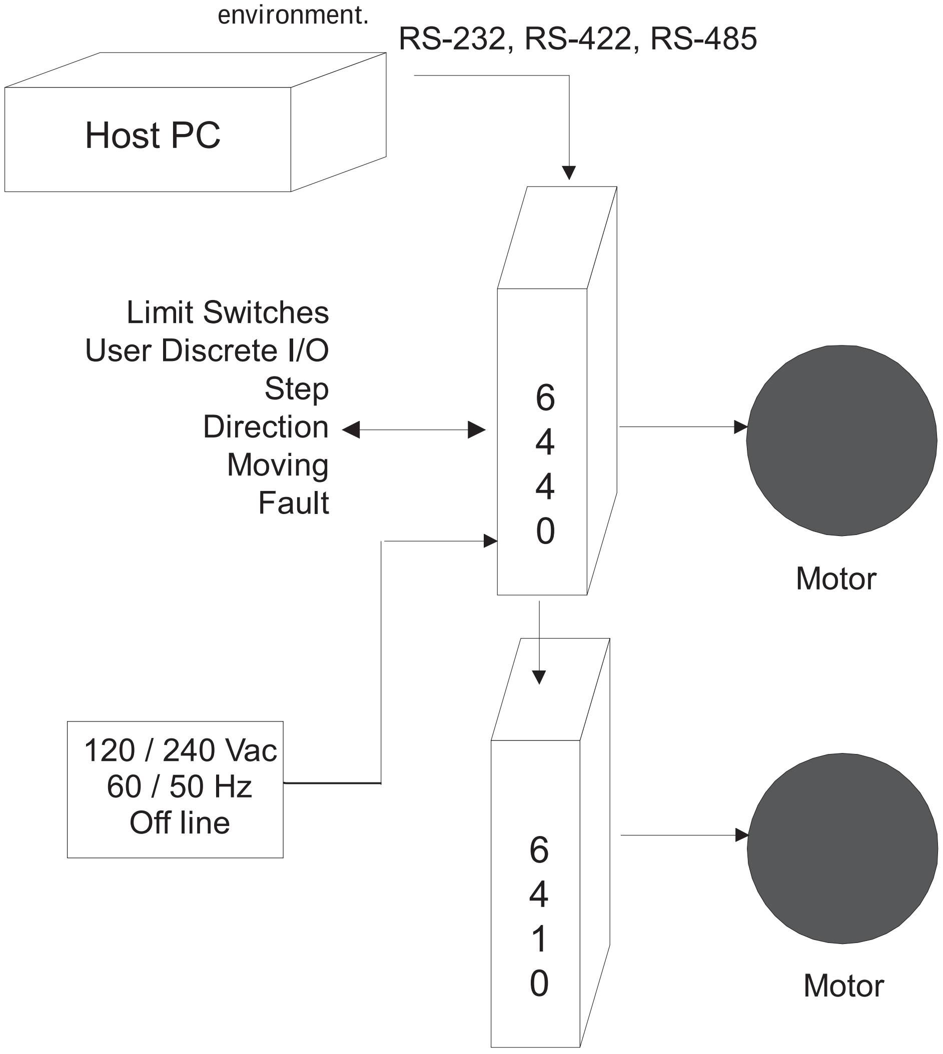
E.1 Standalone Operation

The 6440 has approximately 1792 bytes of internal, non-volatile memory for user programs. Internal programs can set initial and final velocities, acceleration and deceleration rates, execute incremental, absolute and constant velocity moves, program loops, branches based on the state of an input port and also write to the output port(s). In addition, a special case of the G instruction allows up to 16 different motion routines to be executed based on the state of the lower 4 port lines. Programs are entered with the E command and parameters saved with the P command in immediate mode. The external Remote Start and Stop can be used to start and stop program execution. Limit switches attached to the assembly can be used as safeguards for over-motion. A slave 6410 can be controlled by the Direction and Step lines, the Moving line can be polled for motion and the Fault line can be polled for fault conditions.



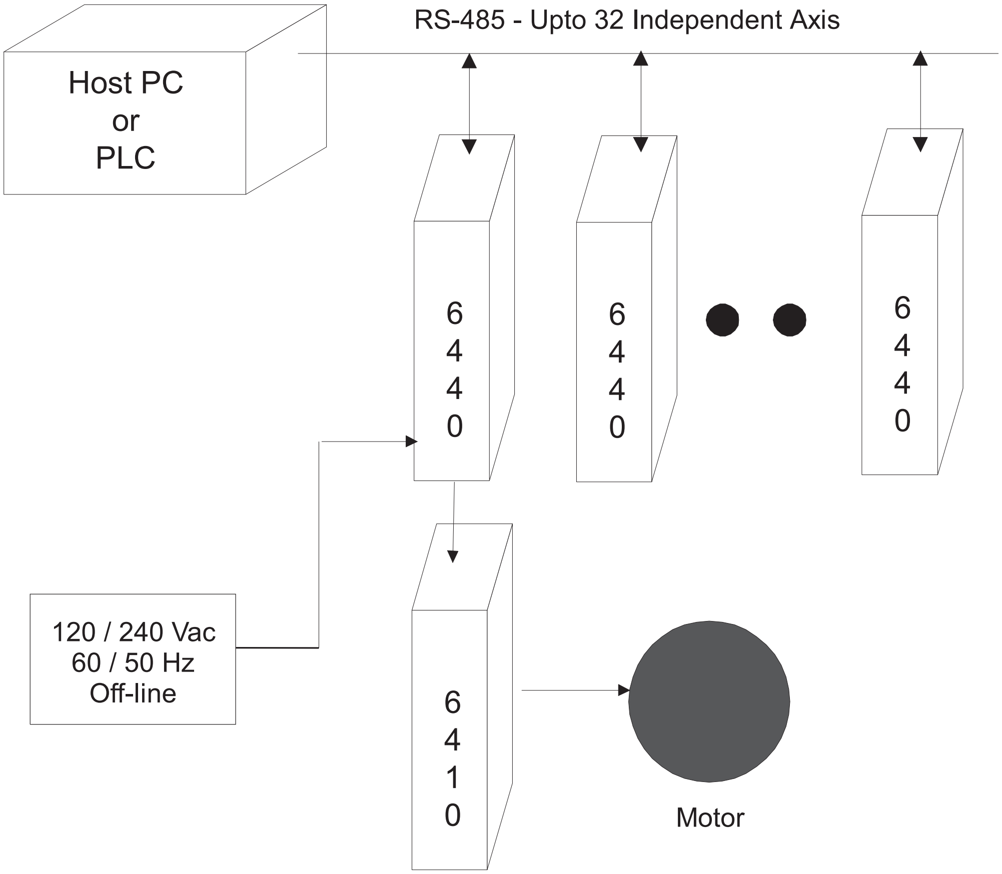
E.2 Host Operation Single Unit

In this configuration a custom host computer program issues immediate mode commands controlling all motion and parameter settings. All operations normally embodied in an on-board non-volatile program are executed on the host with only default motion parameters stored in non-volatile memory. This approach allows the programming freedom and flexibility afforded by the host programming language and development environment.



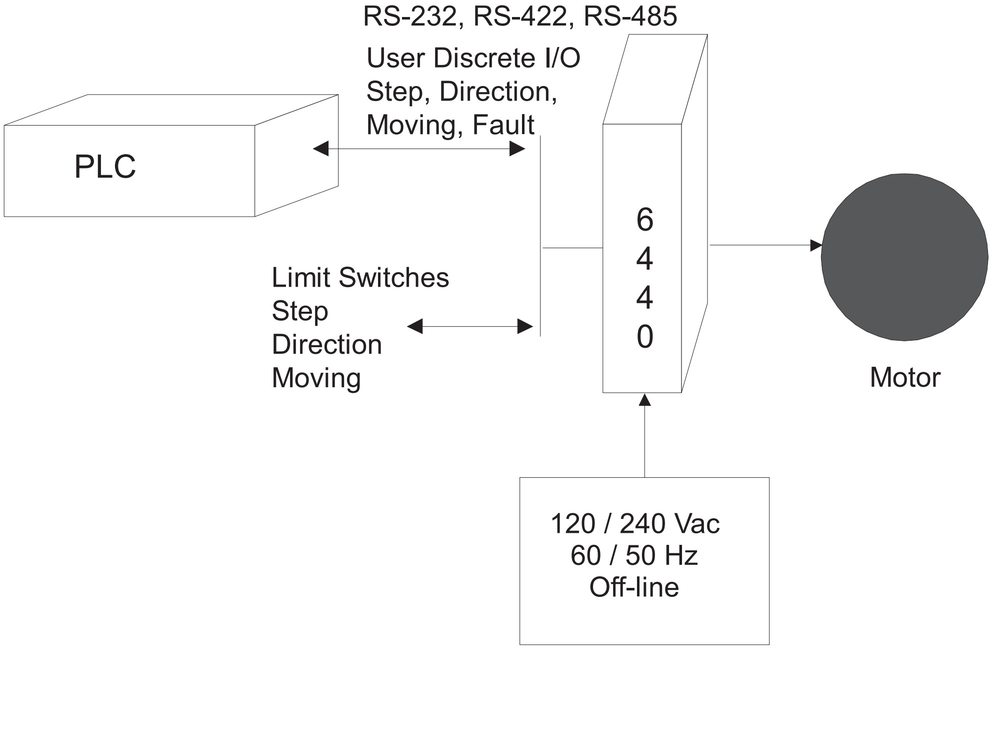
E.3 Host Operation Multiple Units

This configuration allows up to 32 independent axes to be controlled in immediate mode from a host computer or PLC. Normally all the units operate in immediate mode, but can also execute from on-board non-volatile memory. Remember, the RS-485 implementation is designed as a bus for the host computer to send commands to individual 6440s. The host can issue commands to read ports on individual units, but individual 6440s cannot initiate transfers back to the host. Only the host is the bus master. As in other configurations, an individual 6440 can also control a slave 6410.


RS-485 - Upto 32 Independent Axis
Host PC or PLC
6440
6440
6440
120 / 240 Vac 60 / 50 Hz Off-line
6410
Motor

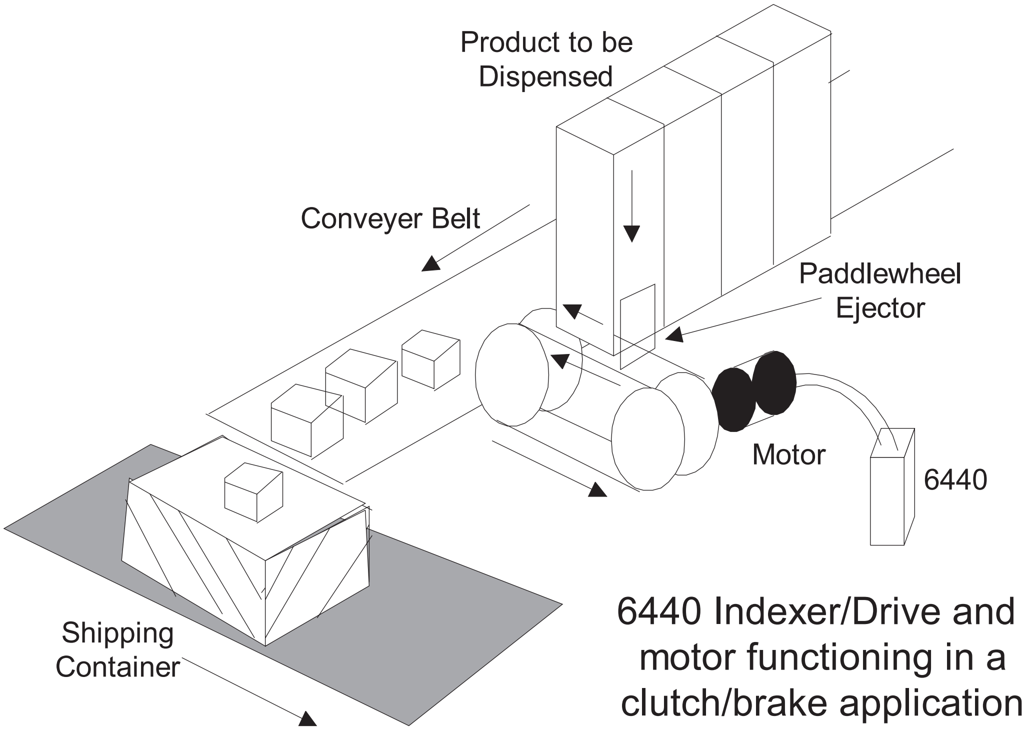
E.4 PLC Based Control

This configuration is similar to the others described above with the PLC acting as the host computer. The PLC can interface to the parallel port lines via a RS-232, RS-422 or RS-485 interface.



E.5 6440 Dispensing Product onto Conveyer Belt

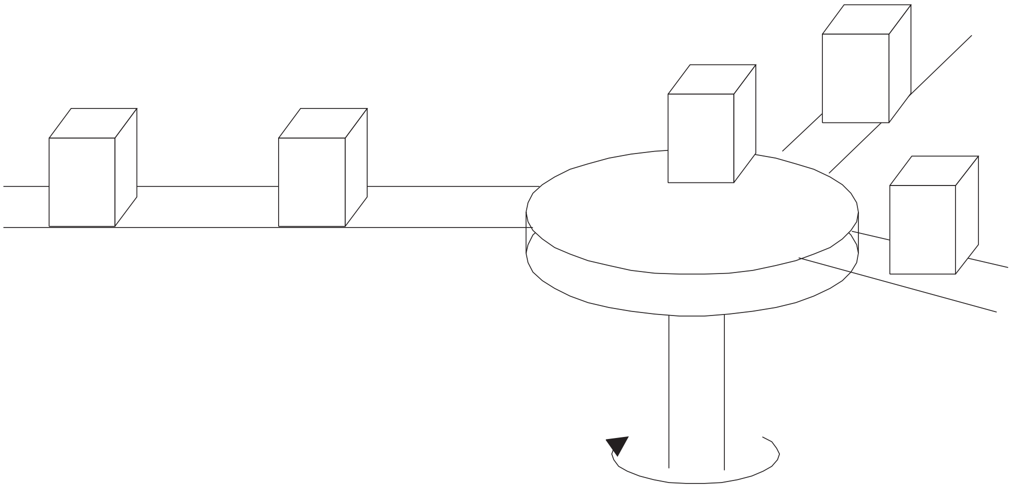
This example shows the 6440 Indexer/Drive and motor dispensing products onto a conveyor belt and into a shipping container. The stepper motor supplies start-stop motion to a paddlewheel ejector to dispense the product. An optic sensor is used to advance the paddlewheel to the starting point. An external Start signal initiates motion to eject the product, the motion continues for one revolution until the paddlewheel is aligned for the next cycle. A host computer or an on-board non-volatile program sets the acceleration, deceleration and velocity parameters and the 6440 executes the motion.



6440 Indexer/Drive and motor functioning in a clutch/brake application

E.6 6440 Controlling Indexer Table

The 6440 Indexer/Drive and motor combination easily drives an Indexing Table. The motor can drive the table directly, with a belt, or through a worm gear. An optic sensor can be used to indicate the starting or home position of the table and motion can be executed with open-loop stepping commands.



Incremental Indexing Table

E.7 Miscellaneous Application Notes

Start-Stop Program

The following is a start-stop motion program example. The program starting at location 128 in “FAST” memory spins the motor until an optic sensor wired to Port 8 goes HIGH. Once this stop point is detected, the motor decelerates until coming to a stop. The program is written such that the wait command executes in parallel with the deceleration. If the deceleration factor is set for maximum deceleration rate (0), motion stops very quickly and dead time will be about 1 second. If deceleration is set for the minimum rate (255), depending on the velocities involved the stopping time may be several seconds possibly over-shooting the stopping point substantially and eliminating the 1 second wait command. The optic design, be it a reflecting or slotted type may require some moving distance to disengage or deactivate its output. The program below runs at a constant slower speed until the sensor is cleared and then loops back to the beginning. This continuous running program can easily be modified to start up with an external input applied to an I/O port.

```
128   R 1200        ;  spin the motor

131   U 140 14      ;  jump to location 140 when stop point is
                       detected by the optic sensor.

135   G 131         ; continue polling Port 8

140   R 0           ; decelerate to 0

143   W 100         ; 1 second wait in parallel to R 0

146   U 146 14      ; loop till optics deactivated

150   G 128         ; back to beginning
```

Input Port Polling Times

Some applications may require estimates of branching times. The program below in “FAST” memory takes between 200 us to 1 ms to respond to an input line. The same program in “SLOW” memory requires approximately 1 ms to 12 ms to respond to an input line change.

```
128    U 140 14

132    G 128

140    W 100

143    G 128
```

Using the Input Port to Select 1 of 16 Different Profiles

An external thumbwheel switch can be used to select 1 of 16 different motion profiles. There are several different approaches to implementing this capability, but the easiest approach uses the G instruction to execute an indexed jump based upon the state of the lower 4 input port lines. This example initiates the indexed jump when the Remote Start input is activated low. Each jump executes a particular motion and then terminates, returning the 6440 to immediate mode waiting for the next Remote Start activation. At the end of motion either a termination byte (put there by the E command) or a branch to another location (G instruction) must occur. Remember that only 16 locations are available for instructions, including the program termination character or G instruction.

Example

```
E 0
        0
        Any initialization code


        G 2048          ; execute indexed jump



E       End of program here
```

This is the first profile to HOME the unit, where P[4-1] = Logic 1, 1, 1, 1.

```
E 256
        256                     H 200 1
E
```

This is the 2nd profile, where P[4-1] = Logic 1, 1, 1, 0.

```
E 272
        272                     M 40 40
                                 +  800
E
```

This is the 3rd profile, where P[4-1] = Logic 1, 1, 0, 1.

```
E 288
        288                     M 100 40
                                 -  200
E
```

Here's another example where an AUTOSTART program waits for Port 8 to activate the program.

```
E 1600
        1600                              *
        Any initialization code
        LoopU Loop 14  ; don't continue until Port 8 inactive
                      ; Port 8 must now be inactive high
        WlowU Wlow 14; don't continue until Port 8 active low
        G 2048              ; execute indexed jump
E       End of program here
```

This is the first profile to HOME the unit, where P[4-1] = Logic 1, 1, 1, 1.

```
E 256
        256                           H 200 1
                                        W 0
                                      G Loop
E
```

This is the 2nd profile, where P[4-1] = Logic 1, 1, 1, 0.

```
E 272
        272                           M 40 40
                                       +  800
                                        W 0
                                      G Loop
E
```

This is the 3rd profile, where P[4-1] = Logic 1, 1, 0, 1.

```
E 288
     288                    M 100 40
                              - 200
                              W 0
                             G Loop
E
```

Note: *Since the jump table is encoded as straight binary rather than grey-coded, the 4 lower bits should be set first followed by a separate activation signal as is done above. Otherwise, if you attempt to set up a continuous loop where the 4 input lines may change state while the G instruction is executing, the program may not jump to the desired profile.*

Appendix F ASCII Codes

ASCII Code Result			ASCII Code Result		ASCII Code Result		ASCII Code Result	
0	^ @	NUL	32		64	@	96	‘
1	^ A	SOH	33	!	65	A	97	a
2	^ B	STX	34	\	66	B	98	b
3	^ C	ETX	35	#	67	C	99	c
4	^ D	EOT	36	$	68	D	100	d
5	^ E	ENQ	37	%	69	E	101	e
6	^ F	ACK	38	&	70	F	102	f
7	^ G	BEL	39	‘	71	G	103	g
8	^ H	BS	40	(	72	H	104	h
9	^ I	HT	41	)	73	I	105	i
10	^ J	LF	42	*	74	J	106	j
11	^ K	VT	43	+	75	K	107	k
12	^ L	FF	44	,	76	L	108	l
13	^ M	CR	45	-	77	M	109	m
14	^ N	SO	46	.	78	N	110	n
15	^ O	SI	47	/	79	O	111	o
16	^ P	DLE	48	0	80	P	112	p
17	^ Q	DC1	49	1	81	Q	113	q
18	^ R	DC2	50	2	82	R	114	r
19	^ S	DC3	51	3	83	S	115	s
20	^ T	DC4	52	4	84	T	116	t
21	^ U	NAK	53	5	85	U	117	u
22	^ V	SYN	54	6	86	V	118	v
23	^ W	ETB	55	7	87	W	119	w
24	^ X	CAN	56	8	88	X	120	x
25	^ Y	EM	57	9	89	Y	121	y
26	^ Z	SUB	58	:	90	Z	122	z
27	^ [	ESC	59	;	91	[	123	{
28	^ \	FS	60	<	92	\	124	|
29	^]	GS	61	=	93	]	125	}
30	^ ^	RS	62	>	94	^	126	~
31	^ _	US	63	?	95	_	127	

INDEX