

Compumotor

**Model RP240
User Guide**

**Compumotor Division
Parker Hannifin Corporation
p/n 88-012156-01 B**



T A B L E O F C O N T E N T S

How To Use This User Guide	i
Assumptions	i
User Guide Contents	i
Installation Process Overview	ii
Installation Procedures	ii
Related Publications	ii
① INTRODUCTION	1
Product Description	1
RP240 Features	1
② GETTING STARTED	3
What You Should Have	3
Basic System Configuration	3
Wiring Connections	4
③ INSTALLATION	5
Environmental Considerations	5
Electrical Noise	5
Airborne Contaminants	5
System Mounting	6
RP240 Door Mount	6
RP240 Dimensions	7
RP240-NEMA4 Door Mount	7
RP240 Connections	8
RP240 to SX Wiring Diagram	9
RP240 to ZX Wiring Diagram	9
RP240 to Model 500 Wiring Diagram	10
RP240 Connector Pin Out	10
Installation Verification	10
6000 Series	10
Using the RP240 With An Extended X Language Product	11
④ EXTENDED X COMMAND SUMMARY AND APPLICATION DESIGN	13
Extended X Language Command Summary	13
Description of Extended X Language Command Format	14
Extended X Language Command Listing	15
Discussion of Extended X Language Commands	20
Prompting an Operator or Displaying Information	20
Extended X Language Command Programming Example	21
Processing Information	23
Enabling STOP and PAUSE Keys	23
Sample Program	24
Editing Sequences	29
Daisy Chaining	30
⑤ 6000 SERIES COMMAND SUMMARY AND APPLICATION DESIGN	33
6000 Series Command Summary	33
Description 6000 Series Command Format	34
RP240 X Language Command Listing	35
Prompting an Operator or Displaying Information	38
6000 Series Command Programming Example	39
Processing Information	41
Enabling Jog Mode	41
Changing the Password	41

⑥ DIRECT RP240 CONTROL	43
RP240 Overview.....	43
Command Summary.....	43
RP240 Response—Extended X Mode.....	45
Key Press Response of RP240—6000 Mode.....	45
⑦ HARDWARE REFERENCE	47
Environmental.....	47
Electrical Specifications.....	47
RP240 Connector.....	47
Power.....	47
RS-232C Communication.....	47
⑧ TROUBLESHOOTING	49
PROM Table.....	49
Troubleshooting.....	49
Reducing Electrical Noise.....	50
Diagnostic LEDs.....	50
Basic Troubleshooting Procedure.....	50
Common Problems & Solutions.....	51
RS-232C Communications.....	51
Editing Sequences—Extended X.....	52
INDEX	53

O V E R V I E W

How To Use This User Guide

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

Assumptions

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

- Basic electronics concepts (voltage, switches, current, etc.)

User Guide Contents

This user guide contains the following information.

Chapter ① Introduction

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

Chapter ② Getting Started

This chapter contains a detailed list of items you should have received with your RP240 or RP240-NEMA4 shipment. It will help you become familiar with the system and ensure that each component functions properly.

Chapter ③ Installation

This chapter provides instructions to properly mount the system and make all electrical and non-electrical connections. Detailed instructions for wiring the RP240 to Compumotor's 6200 Indexer and 6250 Servo Controller, Model 500 Indexer, SX, and ZX Indexer/Drives is provided. Upon completion of this chapter, your system should be completely installed and ready to perform basic operations.

Chapter ④ Extended X Command Summary and Application Design

This chapter helps you integrate the RP240 to the Model 500 Indexer, and SX, and ZX Indexer/Drives. Command enhancements to the X language commands are described, and sample applications are provided.

Chapter ⑤ 6000 Series Command Summary & Application Design

This chapter helps you integrate the RP240 to the 6200 Indexer, and 6250 Servo Controller. RP240 commands within the 6000 Series language are described, and sample applications are provided.

Chapter ⑥ Direct Control of RP240

This chapter describes the internal command set of the RP240.

Chapter ⑦
Hardware
Reference

This chapter contains information on system specifications (electrical, I/O, etc.).

Chapter ⑧
Troubleshooting

This chapter describes Compumotor's recommended system maintenance procedures. It also provides methods for isolating and resolving hardware and software problems.

Installation Process Overview

To ensure safe, trouble-free system integration and operation, you should pay special attention to the environment in which the RP240 equipment will operate. Environmental conditions include the layout, mounting, and wiring, grounding, and shielding practices used.

Installation Procedures

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

- ① Review this entire user guide. Become familiar with the user guide's contents so that you can quickly find the information you need.
- ② Read Chapter ① Introduction to develop a basic understanding of all system components, their functions, and interrelationships.
- ③ Read Chapter ② Getting Started, and verify that you have received all the proper components for your system, and that all the items in your shipment have arrived without damage. Follow the step-by-step bench test procedures to verify the basic functionality of the RP240.
- ④ After reading Chapter ② begin the installation process. Do not deviate from the sequence or installation methods provided.
- ⑤ Perform as many basic functions as you can with the configuration established in Chapter ②. You can perform this task only if you have reviewed the entire user guide.

Related Publications

The following publications may be helpful resources.

- Current Parker Compumotor Motion Control Catalog
- User guide for the Compumotor Indexer or Servo Controller that you will use with the RP240
- Schram, Peter (editor). *The National Electric Code Handbook (Third Edition)*. Quincy, MA: National Fire Protection Association

C H A P T E R ①

Introduction

Chapter Objective

The information in this chapter will enable you to:

- Understand the product's basic functions & features

Product Description

The RP240 is designed to operate as an operator interface for Compumotor's Extended X (the SX Indexer/Drive, ZX Indexer/Drive, and Model 500 Indexer) and 6000 Series products (6200 Indexer and 6250 Servo Controller). The RP240 operates strictly as a dumb terminal. No programming of any kind can be accomplished using the RP240. The RP240 is controlled by commands available in the Extended X and 6000 Series command sets. The commands allow the programmer to create a program (within the Extended X or 6000 Series product) to prompt an operator for numeric information, read function keys, display variables and text, and control the 8 LEDs located on the RP240.

If the RP240 is not used with the SX, ZX, Model 500, 6200, or 6250, ASCII command strings sent directly to the RP240 can be used to control the functions contained within the RP240 (refer to *Chapter ⑥ Direct Control of the RP240*).

RP240 Features

- Operates directly with Compumotor's SX and ZX Indexer/Drives, and the Model 500 Indexer
- Operates directly with Compumotor's stand-alone 6000 Series products (6200 and 6250)
- Can be used with any controller capable of transmitting ASCII strings across RS-232C
- Screw terminal connections for easy wiring
- Optional NEMA specification ratings (standard RP240 is NEMA 12, optional RP240-NEMA4 meets NEMA 4 & NEMA 12 specifications when panel mounted.)
- Adjustable contrast for the LCD display
- Eight independent LEDs
- Numeric and function key data entry
- Uses less than 100 mA of current at 5VDC

C H A P T E R ②

Getting Started

Chapter Objectives

The information in this chapter will enable you to:

- Verify that each component of your system has been delivered safely
- Become familiar with system components and their interrelationships

What You Should Have

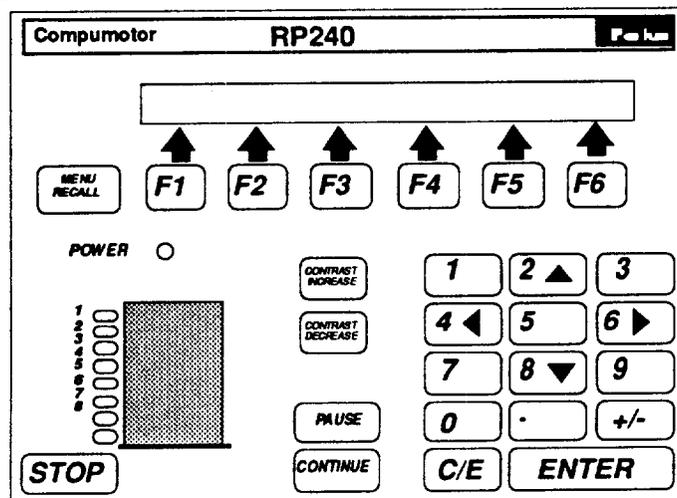
Inspect your RP240 shipment upon receipt for damage to its shipping container. Report any damage to the shipping company as soon as possible. Parker Compumotor cannot be held responsible for damage incurred in shipment. The items listed in the table below should be present and in good condition.

	RP240	RP240-NEMA4
Description	Part #	Part #
RP240 Main Unit	RP-240	RP-240 NEMA4
Gasket	58-009135-01	58-013341-01
RP240 User Guide	88-012156-01	88-012156-01

CAUTION

Do not use sharp objects such as pencils, screw drivers, etc. on the front panel it will cause permanent damage to the membrane and will void the warranty.

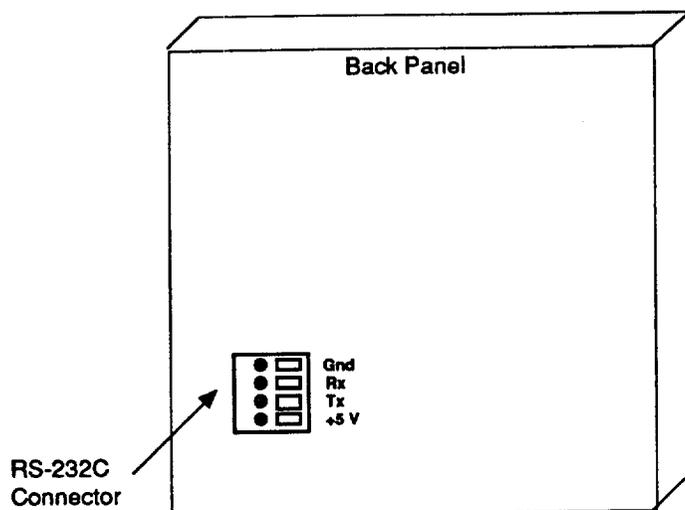
Basic System Configuration



Wiring Connections

The following figure illustrates the RS-232C and power; connector. This connector is located on the back of the RP240.

 Helpful Hint:
RP240 Back Panel



C H A P T E R ③

Installation

Chapter Objectives

The information in this chapter will enable you to:

- Mount the RP240 properly
- Wire the RP240 correctly
- Verify that the complete system is installed properly

Environmental Considerations

Helpful Hint:
For more on NEMA standards, contact the *National Electrical Manufacturers Association*, 2101 L Street Northwest, Washington, D.C. 20037. Ask for NEMA Standards Publication No. 250 Enclosures for Electrical Equipment.

The RP240 is designed to be mounted to a door, on a panel front, desk, or carried by hand. The RP240-NEMA4 is designed to be flush mounted in a door. Both the RP240 and RP240-NEMA4 are designed to work in temperatures up to 122°F (50°C). NEMA 12 specifications are achieved when the RP240 is door or panel mounted using the provided gasket. NEMA 4 specifications are achieved when the RP240-NEMA4 is door or panel mounted using the provided gasket. NEMA Type 12 enclosures provide a degree of protection against dust, falling dirt, and dripping non corrosive liquids. NEMA Type 4 enclosures provide a degree of protection against windblown dust and rain, splashing water, and hose-directed water.

Electrical Noise

Minimize the potential for electrical noise before installing the RP240 rather than attempting to solve such problems after installation. You can prevent electrical noise by observing the following installation precautions:

- Do not put high-voltage wires and low-level signals in the same conduit
- Do not expose wiring (shield all wires)
- Ensure that all components are properly grounded

Airborne Contaminants

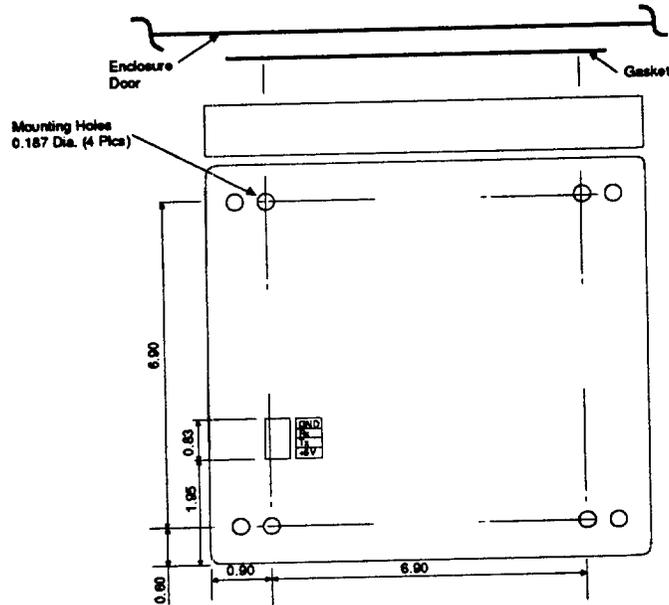
Contaminants that may come in contact with the RP240 should be carefully controlled. Particulate contaminants, especially electrically conductive ones (such as metal shavings), can damage the RP240.

System Mounting

RP240 Door Mount

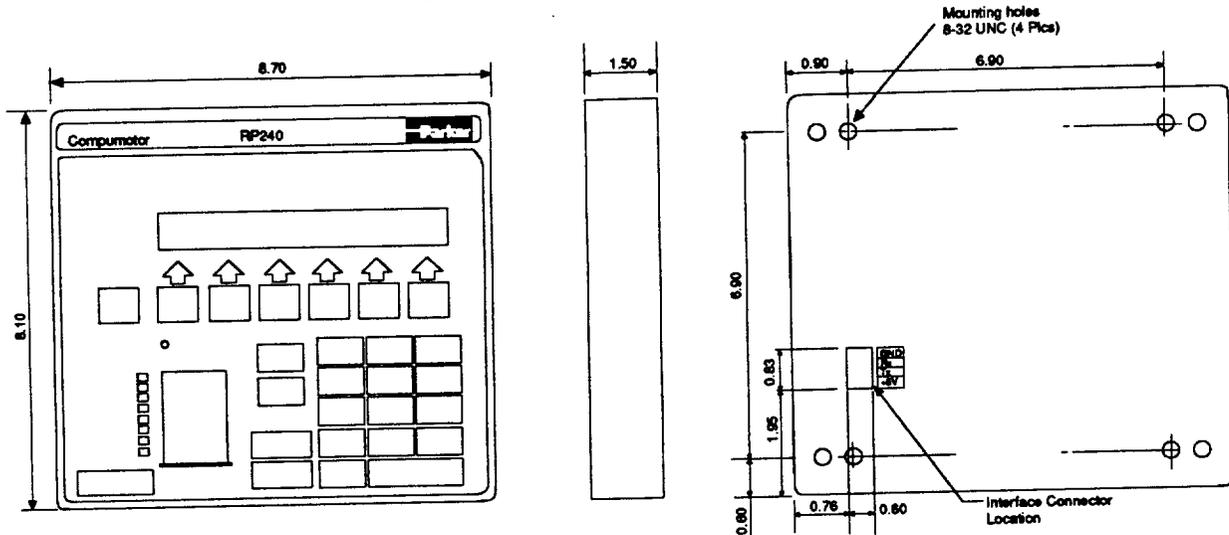
To mount the RP240 operator panel to the door of an enclosure, or on a panel, follow the steps below.

- ① Drill four holes as shown in the following figure
- ② Remove the bayonet mounts (screws and standoffs) and rubber feet from the back of the operator panel (save the screws for mounting).
- ③ Put one screw through the door.
- ④ Put the gasket (provided) on the screw.
- ⑤ Put the operator panel onto the screw and tighten.
- ⑥ Align the panel and gasket and insert the remaining three screws.
- ⑦ Connect the 4 wires between the operator panel and the Extended X or 6000 Series product. See *RP240 Connections* later in this chapter.



RP240 Dimensions

The following figure shows the dimensional drawing for the RP240. Mounting holes and their dimensions are given.

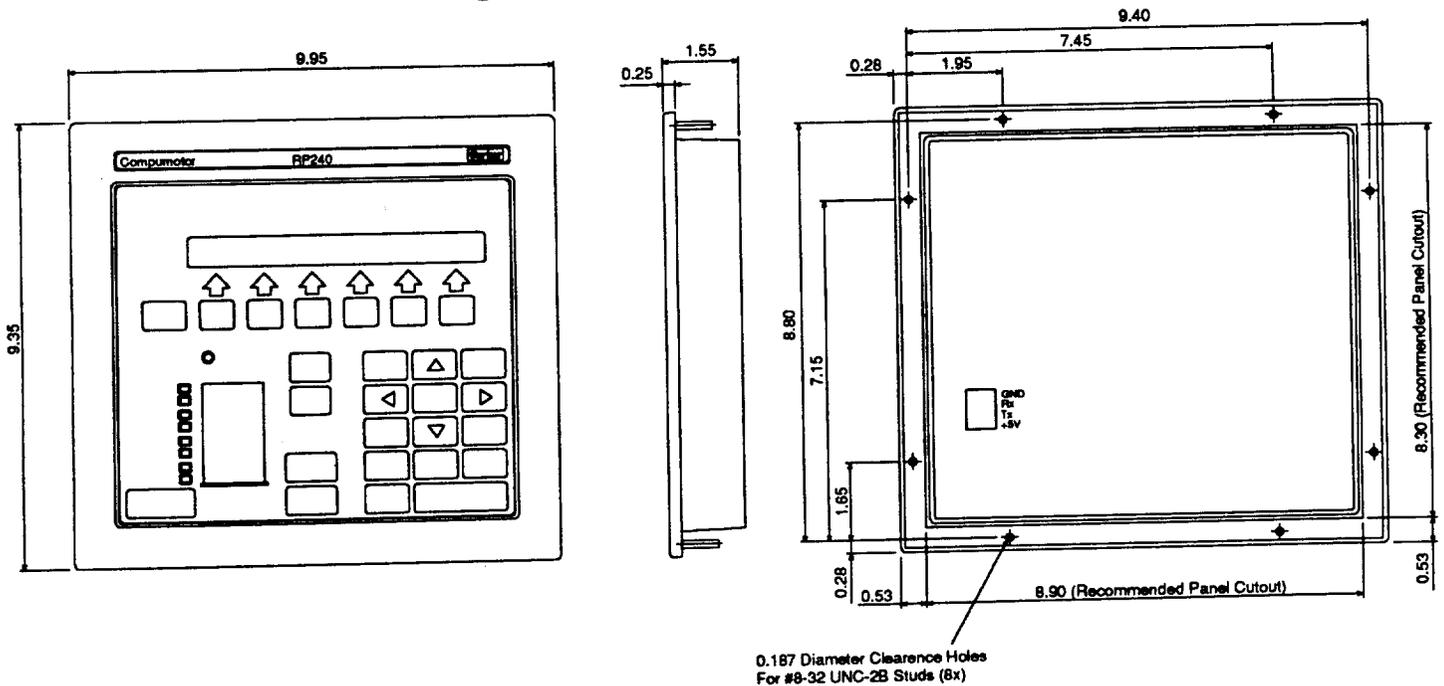


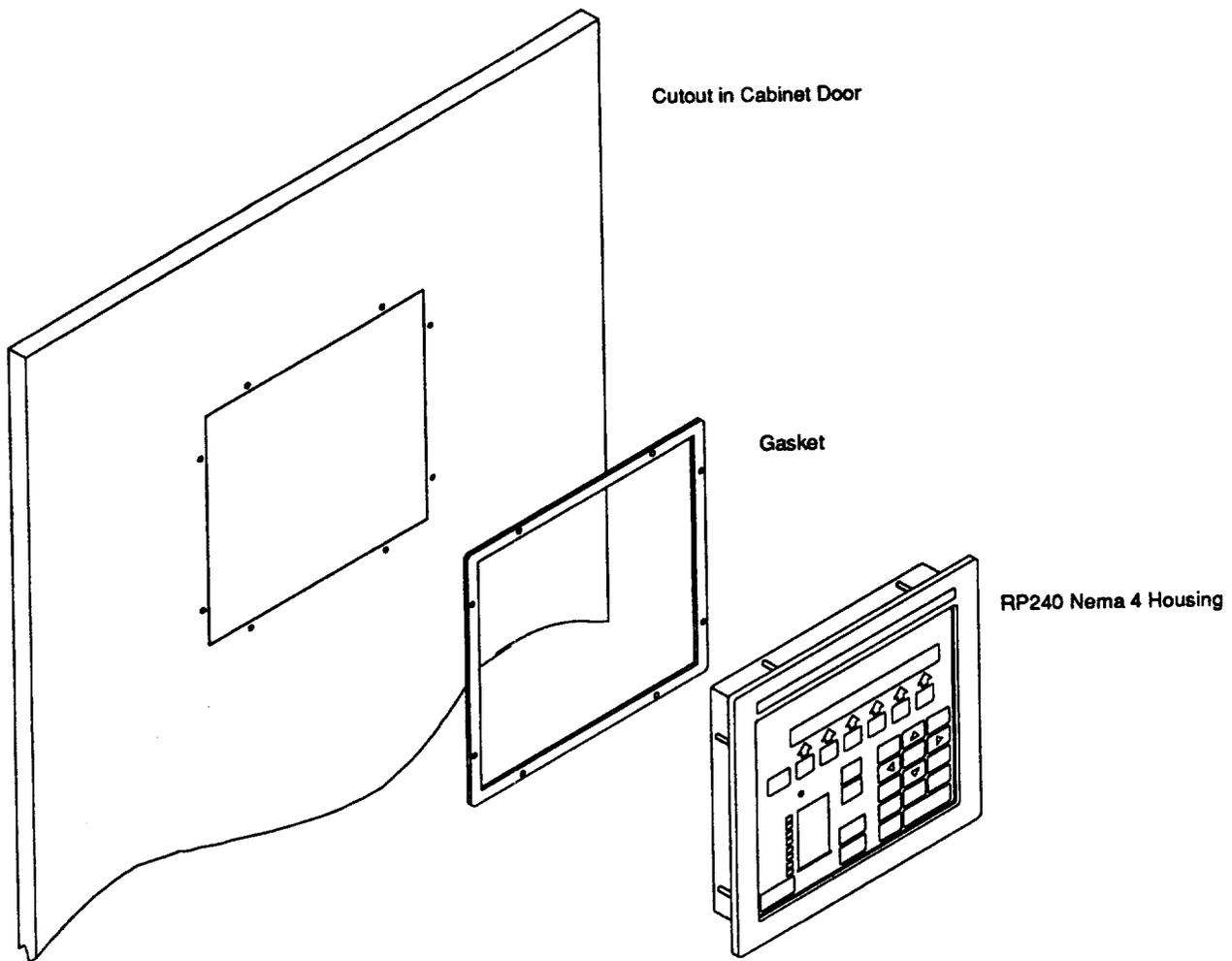
RP240-NEMA4 Door Mount

To mount the RP240-NEMA4 operator panel through the door of an enclosure, or in a panel, follow the steps below.

- ① Cut an 8.3 inch vertical by 8.9 inch horizontal hole in the door of the enclosure or panel as shown on the following page.
- ② Drill eight holes in the locations shown.
- ③ Place the gasket (provided) on the RP240-NEMA4.
- ④ Put the operator panel through the door and tighten the eight nuts provided.
- ⑤ Connect the 4 wires between the operator panel and the Extended X (SX, ZX, or Model 500) or the 6000 Series (6200 or 6250) product.

The following figure shows the dimensional drawing for the RP240-NEMA4. Mounting studs and the required diameter clearance holes are given.





Wiring Guidelines

Helpful Hint:
All earth ground connections must be continuous and permanent. Compumotor recommends a single-point grounding setup.

Proper grounding of electrical equipment is essential to ensure the safety of personnel. You can reduce the effects of electrical noise due to electromagnetic interference (EMI) by proper grounding. All Compumotor equipment should be properly grounded. A good source of information on grounding requirements is the *National Electrical Code* published by the National Fire Protection Association of Boston, MA. In general, all components and enclosures must be connected to earth ground through a grounding electrode conductor to provide a low impedance path for ground fault or noise-induced currents.

RP240 Connections

The RP240 operator panel is connected to the Extended X (SX, ZX, or Model 500) and 6000 Series (6200 or 6250) products by means of a four wire shielded cable (user supplied). For cable lengths up to 50 feet long, Compumotor recommends using 20 AWG wire. Cable lengths longer than 50 feet are not recommended. This in accordance with the Electronic Industries Association (EIA) recommendations for RS-232C at 9600 baud.

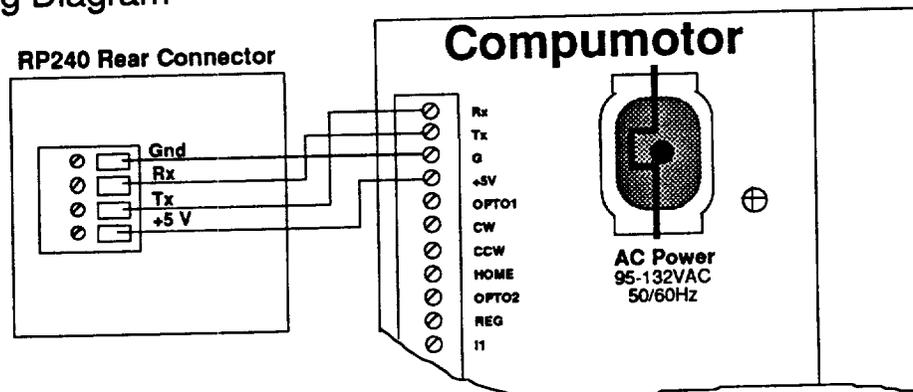
To wire the RP240 to a 6000 Series product (6200 or 6250), connect the +5VDC connection on the RP240 to the +5VDC source on the 6000 Series connector labeled RP240. Connect the RP240 ground connection to the ground on the 6000 Series connector labeled RP240. Connect the RP240 Receive Data (Rx) terminal to the controller's Transmit Data (Tx) terminal, and the RP240 Transmit Data (Tx) terminal to the controller's Receive Data (Rx) terminal. Make all connections on the 6000 Series connector labeled RP240.

To wire the RP240 to an Extended X product, connect the +5VDC connection on the RP240 to a +5VDC source capable of supplying 100 mA of current. You must supply 5VDC $\pm 2\%$ to pin 1 of the RP240. You can use the 5VDC power supplies on the SX, ZX, or Model 500 provided there is enough current remaining. Current levels are shown in the table below.

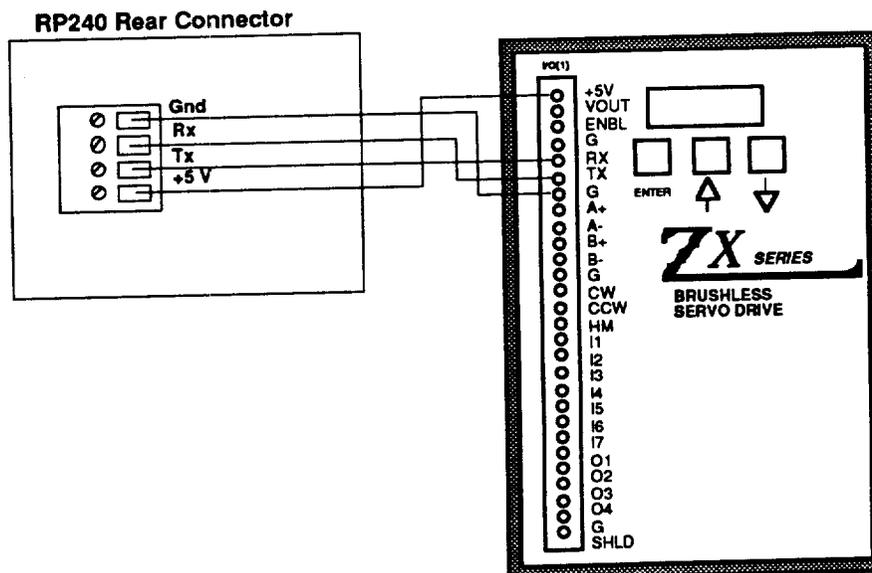
	Available	Current Requirements		
		Encoder	I/O	RP240
SX	250 mA	150 mA	10 mA each	100 mA
ZX	500 mA	150 mA	0 mA	100 mA
500	500 mA	150 mA	0 mA	100 mA

Connect the RP240 ground connection to the ground of the +5VDC power supply. Connect the RP240 Receive Data (Rx) to the controller's Transmit Data (Tx), and the RP240 Transmit Data (Tx) to the controller's Receive Data (Rx). Connect the RP240 only after you have programmed the controller for operation with the RP240.

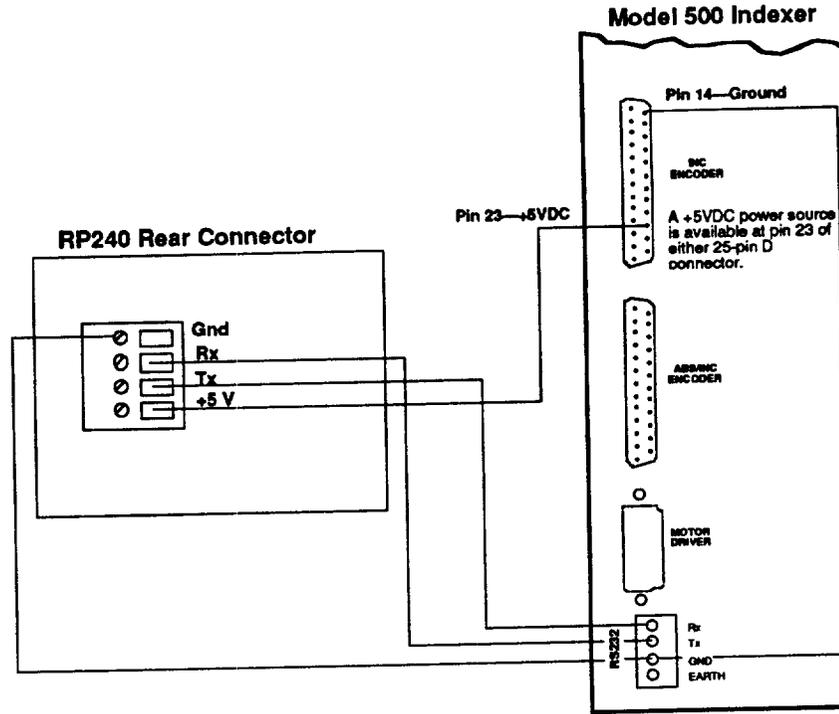
RP240 to SX Wiring Diagram



RP240 to ZX Wiring Diagram



RP240 to Model 500 Wiring Diagram



RP240 Connector Pin Out

The following table provides the pin out for the RP240 connector.

Pin	In/Out	Function	Description
1	In	Ground	Logic Ground
2	In	Receive (Rx)	RS-232C Receive Data input
3	Out	Transmit (Tx)	RS-232C Transmit Data output
4	In	+5VDC	Input Power (100 mA minimum)

Installation Verification

6000 Series

To verify the operation of the RP240 and the 6000 Series product to which it is connected, apply power to the 6000 Series product. The RP240 display should look similar to one of the following screens. If you are using an Extended X product, go to the next section.

				COMPUMOTOR	6200	INDEXER			
RUN	JOG					DISPLAY	ETC		

				COMPUMOTOR	6200	INDEXER			
ACCESS	RUN								

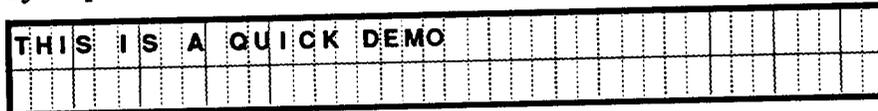
Using the RP240 With An Extended X Language Product

Helpful Hint:
If you are using the RP240 with a 6000 Series product, skip this section.

Before continuing with this section, you should be familiar with the Extended X Language product you will be using. An understanding of programming sequences is important. Download the following program to Sequence #100 in the Extended X Language product.

<u>Command</u>	<u>Description</u>
>XE100	Erase sequence #100
>XD100	Begin definition of sequence #100
DCLR0	Clear all lines of the RP240 display
DPC100	Cursor to line 1, column 0
DTXTTHIS_IS_A_QUICK_DEMO	Display text at current cursor location
VARI=FUN	Wait for function key
DLED11111111	Turn on all LEDs
DCLR0	Clear all lines of RP240 display
DPC100	Cursor to line 1, column 0
DTXTTHE_LEDS_WILL_TURN_OFF_IN_5_SECONDS	Display text
T5	Wait 5 seconds
DLED00000000	Turn off all LEDs
DCLR0	Clear all lines of RP240 display
XT	End sequence definition
>	

Attach the RP240 after you have downloaded the above sequence correctly. Cycle power. The following is displayed.



Press any function key to continue. After pressing any function key, all the LEDs on the RP240 will turn on and the display will read as shown below.



After 5 seconds, all the LEDs on the RP240 will turn off and the display will be cleared.

C H A P T E R ④

Extended X Command Summary and Application Design

Chapter Objectives

The information in this chapter will enable you to:

- Program the SX, ZX, or Model 500 to control the RP240
- Customize the system to meet your requirements

PROGRAMMING NOTE

This chapter provides Extended X programming information that is required to communicate with the RP240. If you are using a 6000 Series product (6200 or 6250), skip ahead to *Chapter ⑤ 6000 Series Command Summary and Application Design*.

Extended X Language Command Summary

Command	Description
DCLR	Clear the display
DCNT	Enable/Disable PAUSE and CONTINUE Keys
DLED	Turn RP240 LEDs On/Off
DPC	Position Cursor
DSTP	Enable/Disable STOP Key
DTXT	Display Text Data on RP240 LCD
DVO	Display Variable Data on RP240 LCD
VARn=FUN	Enable and Read Function Keys
VARn=NUM	Enable and Read Numeric Keypad

Description of Extended X Language Command Format

The following is a definition of the format fields for the Extended X Language products.

① DCLR Clear Display

③ **Type** Programming
④ **Syntax** <DCLRn
⑤ **Units** line number
⑥ **Range** 0, 1, 2
⑦ **Default** None
⑧ **Response** None
⑩ **See Also** DPC, DTXT, DVO

② **Version**
SX Rev C2
ZX Rev D
500 Rev C
③ **Attributes**
[x] Buffered
[] Device specific
[] Saved independently
[] Saved in sequences

- ① **Mnemonic Code and Full Name** This box contains the command's mnemonic code.
- ② **Valid Revision Level** This field contains the revision history of the command. It includes the revision of software when the command was added or modified. If the revision level of software you are using is equal to or greater than the revision level listed in this field, you are using the proper version of the software.
- ③ **Type** This portion of the box contains the command type. All X Language RP240 commands are programming commands. All of these commands affect program flow.
- ④ **Syntax** The proper syntax for the command is shown here. The specific parameters associated with the command are also shown. Definitions of the parameters are described below:
- a This indicates that a device address must accompany the command. Only the device specified by this parameter receives and executes the command. If this parameter, or any other parameter appears within brackets <a>, it is optional.
 - n This represents an integer value.
 - b This represents a binary value. The binary value must be either 0 or 1.
- ⑤ **Units** This field describes what unit of measurement or value the parameter in the command syntax represents.
- ⑥ **Range** This is the range of valid values that you can specify for an argument (or any other parameter specified).
- ⑦ **Default** The default setting for the command is shown in this box. A command will perform its function with the default setting if you do not provide a value.
- ⑧ **Attributes** This field indicates that these commands are buffered. Buffered commands are executed in the order that the system receives them with other buffered commands. Buffered commands can be stored in a sequence.
- ⑨ **Response** There are no system responses associated with these X Language commands.
- ⑩ **See Also** Commands that are related or similar to the command described are listed here.

Extended X Language Command Listing

DCLR Clear Display

Type Programming
Syntax <DCLRn
Units line number
Range 0, 1, 2
Default None
Response None
See Also DPC, DTXT, DVO

Version
 SX Rev C2
 ZX Rev D
 500 Rev C
Attributes
 [x] Buffered
 [] Device specific
 [] Saved independently
 [] Saved in sequences

The Clear Display (DCLR) command clears a specified line of the RP240 display, and repositions the cursor to the beginning of the line.

- n = 0 Clear all lines of the RP240 display
- n = 1 Clear line 1 of the RP240 display
- n = 2 Clear line 2 of the RP240 display

<u>Command</u>	<u>Description</u>
> DCLR1	Clear line one (1) of the RP240 display

DCNT Enable/Disable Pause and Continue

Type Programming
Syntax <a>DCNTb
Units None
Range 0, 1
Default 0
Response None
See Also SSH, SSL

Version
 SX Rev C2
 ZX Rev D
 500 Rev C
Attributes
 [x] Buffered
 [] Device specific
 [] Saved independently
 [] Saved in sequences

This command enables or disables the PAUSE and CONTINUE keys on the RP240.

- DCNT0 Disable PAUSE and CONTINUE keys
- DCNT1 Enable PAUSE and CONTINUE keys

When the PAUSE and CONTINUE keys are enabled (DCNT1), pressing the PAUSE key will cause the RP240 to send a Stop (s) command to the Model 500, SX, or ZX. Pressing the CONTINUE key will cause a Continue (c) command to be sent. To have the Model 500, SX, or ZX pause motion, the Save Command Buffer on Stop (SSH1) and the Resume Execution Enable (SSL1) must be enabled.

Once you have activated the PAUSE and CONTINUE keys, they will remain active at all times, except when numeric or function key information has been requested (VARn=NUM or VARn=FUN).

<u>Command</u>	<u>Description</u>
>XE100	Erase sequence #100
>XD100	Begin definition of sequence #100
SSH1	Enable save command buffer on stop
SSL1	Enable resume execution
DCNT1	Enable PAUSE and CONTINUE keys
DSTP1	Enable STOP key
XT	End definition of sequence #100
>	

DLED Turn RP240 LEDs On/Off

Version
 SX Rev C2
 ZX Rev D
 500 Rev C
Attributes
 Buffered
 Device specific
 Saved independently
 Saved in sequences

Type Programming
Syntax <a>DLEDn
Units None
Range 0, 1, X
Default 0
Response None
See Also O

The DLED command controls the state of the 8 LEDs on the RP240. A one will turn an LED on, a zero will turn an LED off, and an X will leave the LED unchanged from its last state.

The command example below reads from left to right and corresponds to the LEDs from top to bottom.

Command	Description
> DLED1100XX11	Turn LEDs 1,2,7, and 8 on, LEDs 3 and 4 off, and leave LEDs 5 and 6 unchanged

DPC Position Cursor

Version
 SX Rev C2
 ZX Rev D
 500 Rev C
Attributes
 Buffered
 Device specific
 Saved independently
 Saved in sequences

Type Programming
Syntax <a>DPCnxx
Units line number
Range 1, 2
Default None
Response None
See Also DCLR, DTXT, DVO

The Position Cursor (DPC) command places the cursor at line **n**, column **xx**. The lines are numbered from top to bottom, 1 to 2. The columns are numbered from left to right, 00 to 39.

You must use 00, 01, 02,.....,09 instead of 0, 1, 2,....., 9 for the column number (i.e., DPC208, not DPC28).

	Column																																											
Line 1	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39				
Line 2																																												

Once the cursor has been placed, all succeeding text (DTXT) or variable data (DVO) will be displayed beginning at the current cursor location. All numeric data entered using the VARn=NUM command will also be displayed at the current cursor location.

Command	Description
>DPC205	Position the cursor on line 2, column 5
>DTXTCOMPUMOTOR_DEMO_PR	Place message COMPUMOTOR DEMO PROGRAM at current cursor position
OGRAM	

DVO

Display Variable Data on RP240 LCD

Version
 SX Rev C2
 ZX Rev D
 500 Rev C
Attributes
 [x] Buffered
 [] Device specific
 [] Saved independently
 [] Saved in sequences

Type Programming
Syntax <DVOn,n,n,b
Units None
Range See Below
Default None
Response None
See Also DPC, DTXT, DCLR

The DVO command is used to display a variable at the current cursor location. Any of the fifty variables available in the extended X products can be displayed.

1st n = Variable Number (Range = 1 to 50)
 2nd n = Number of whole digits displayed, digits to left of decimal point (Range = 0 to 15)
 3rd n = Number of fractional digits displayed, digits to the right of the decimal point (Range = 0 to 5)
 b = Sign bit, 0 = no sign displayed, 1 = display plus or minus sign

Command	Description
>DPC205	Position the cursor on line 2, column 5
>DVO2,2,1,1	Place variable 2 at current cursor position. If variable 2 contained the number 53.23, then +53.2 is displayed on the RP240 LCD starting at column 5, line 2

VARN=FUN Enable and Read Function Keys

Version
 SX Rev C2
 ZX Rev D
 500 Rev C
Attributes
 [x] Buffered
 [] Device specific
 [] Saved independently
 [] Saved in sequences

Type Programming
Syntax <a>VARN=FUN
Units variable number
Range 1 too50
Default None
Response None
See Also VAR

The VARN=FUN command is used to enable the function keys on the RP240 and retrieve the function key pressed. Once this command is encountered, command processing stops until the RP240 returns a number corresponding to the function key pressed. Function key 1 (F1) returns a 1, function key 2 (F2) returns a 2, etc. MENU RECALL returns a zero. The number that is returned is placed in the variable n.

The RP240 is actually sending an exclamation point (!) before the number returned, allowing the Model 500, SX, or ZX to interpret the value returned.

Note: If you are using a kill sequence (XFK), place the command "*" at the beginning of the kill sequence.

Command	Description
>XE1	Erase sequence 1
>XD1	Begin definition of sequence 1
DPC205	Position the cursor on line 2, column 5
DTXTPRESS_F1	Place message PRESS F1 at current cursor location
VAR1=FUN	Retrieve function key pressed value and place in variable 1
L	Begin endless loop
IF (VAR1=1)	If variable 1 equals 1, do the commands between IF and NIF
XG2	Branch to sequence 2
NIF	End IF statement
N	End endless loop
XT	End definition of sequence 1
>	

VAR=NUM Enable and Read Numeric Keypad

Version
 SX Rev C2
 ZX Rev D
 500 Rev C
 Attributes
 [X] Buffered
 [] Device specific
 [] Saved independently
 [] Saved in sequences

Type Programming
 Syntax <>VAR=NUM
 Units None
 Range 1 to 50
 Default None
 Response None
 See Also VAR

This VAR=NUM command is used to enable the numeric keypad on the RP240 and retrieve the numeric information entered. Once this command is encountered, command processing stops until the RP240 returns a value entered on the numeric keypad. The +/- key changes the sign of the number entered. The +/- key must be pressed before the number is entered, or it will have no effect. The C/E key will clear the number you have entered and allow you to restart. The ENTER key sends the number entered to the 500, SX, or ZX. If no value was entered before the ENTER key was pressed, the ENTER key will have no effect.

The number that is returned by the RP240 is placed in the variable *n* of the 500, SX, or ZX. If either the +/- key or the . key is entered by itself, the variable will be set to zero.

The RP240 is actually sending an exclamation point (!) before the number returned, thus allowing the 500, SX, or ZX to interpret the value returned.

Note: If you are using a kill sequence (XFK), place the command "*" at the beginning of the kill sequence.

Command	Description
>XE1	Erase sequence 1
>XD1	Begin definition of sequence 1
DCLR0	Clear Display
DPC105	Position the cursor on line 1, column 5
DTXTENTER BAG COUNT	Place message ENTER BAG COUNT at current cursor location
DCLR2	Clear Display
DPC121	Position the cursor on line 1, column 21
VAR1=0	Initialize variable 1
WHILE (VAR1<1_OR_VAR1>10)	Do statements between WHILE and NWHILE until 0<VAR1<11
DCLR2	Clear line 2 of RP240 display
DPC121	Position cursor on line 1, column 21
VAR1=NUM	Retrieve numeric value and place in variable 1
IF (VAR1<1)	If variable 1 < 1 do the commands between IF and NIF
DPC205	Position the cursor on line 2, column 5
DTXTBAG_COUNT_TOO_LOW	Place message BAG COUNT TOO LOW at current cursor location
T2	Time delay 2 seconds
NIF	End IF statement
IF (VAR1>10)	If variable 1 > 10 do the commands between IF and NIF
DPC205	Position the cursor on line 2, column 5
DTXTBAG_COUNT_TOO_HIGH	Place message BAG COUNT TOO HIGH at current cursor location
T2	Time delay 2 seconds
NIF	End IF statement
NWHILE	End WHILE statement
MN	Set to Normal mode
LD3	Disable limits (if not connected)
A10	Set acceleration to 10 rps ²
V2	Set velocity to 10 rps
D25000	Set distance to 25000 steps
L (VAR1)	Loop as many times as specified by variable 1
G	Initiate motion
N	End loop
XT	End definition of sequence 1
>	

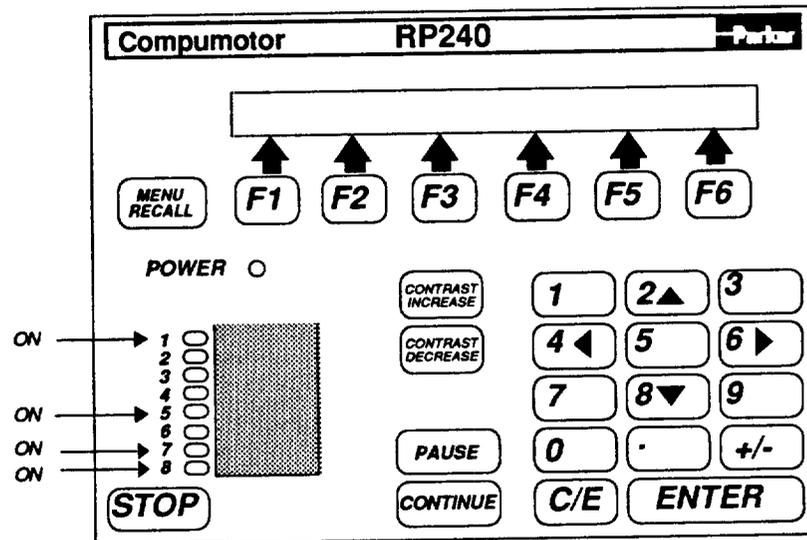
Discussion of Extended X Language Commands

This section discusses the commands described previously in the command reference.

Prompting an Operator or Displaying Information

In many motion control applications, the most important requirement is the operator interface. Presenting information to an operator in a desired format is often difficult at best. The RP240 has two visual indicators to help present information to the operator. The simplest indicator is the 8 LEDs on the panel. These LEDs can be turned on or off with the **DLED** command. The LEDs can be used in conjunction with the outputs to show the state of an output, or they can be used to show status, such as motor moving, specific sequence in progress, etc.

IF a **DLED10001011** command is issued, the LEDs shown below would be illuminated. These eight LEDs can be labeled, using the slide-in card provided, to represent cycle status, output status, etc.



The other indicator is the two line, 40-character LCD display. This display can be controlled with specific Extended X Language commands. The Position Cursor (**DPC**) command allows the user to program the location of the cursor on the LCD display. The Display Text Data (**DTXT**) command on RP240 Display allows the user to place text, beginning at the current cursor location, on the LCD display.

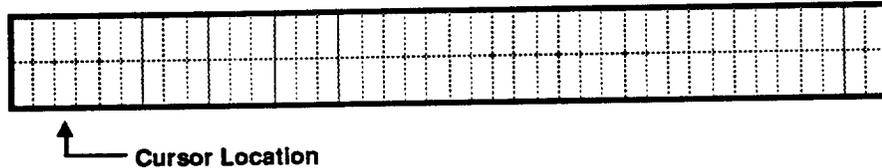
Extended X Language Command Programming Example

A user wants his operator to see the message **ENTER THE CYCLE COUNT**. He wants this message placed on line two, starting after two spaces in from the left. He also wants the user to be able to enter the cycle count three spaces after the message. Below are the steps required to accomplish this.

Step 1

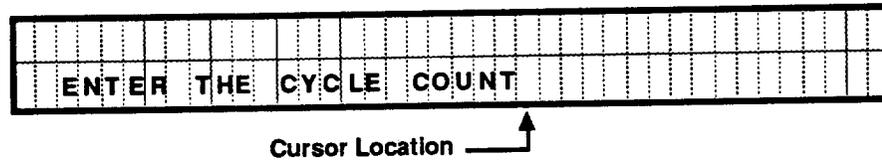
Issue the **DPC202** command

Helpful Hint:
The cursor does not appear on the display. The cursor is displayed when the **VARn=NUM** command is used.



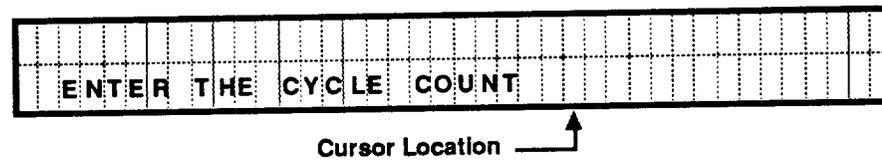
Step 2

Issue the **DTXTENTER_THE_CYCLE_COUNT** command



Step 3

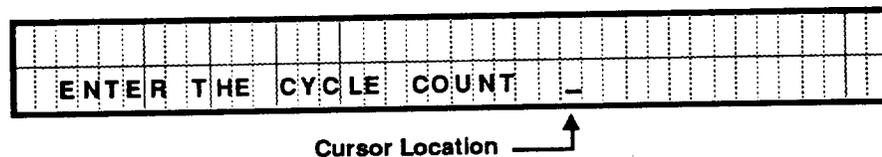
Issue the **DPC225** command



Now that we have provided the operator with the prompting message, how do we obtain the information? The Read and Enable Numeric Keypad (**VARn=NUM**) or Read and Enable Function Keys (**VARn=FUN**) commands provide the answer. The **VARn=NUM** command will enable the numeric keypad and allow the operator to enter information. The numbers, as entered, will be displayed at the current cursor location. Once the **ENTER** key is pressed, the number will be transmitted from the RP240 to the Model 500, SX, or ZX.

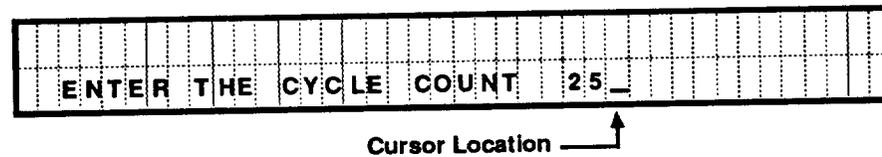
Step 4

Issue the **VAR1=NUM** command



Step 5

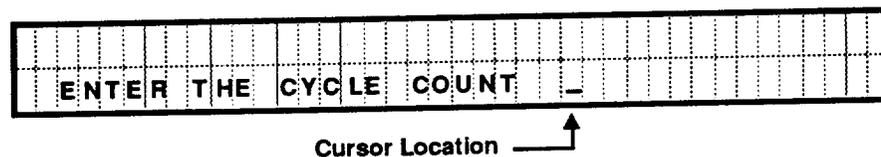
Press a 2, followed by a 5



If the wrong value is entered, press the C/E key and re-enter the value.

Step 6

Press the C/E key



Processing Information

Variables can be used in conditional statements like **IF**, **WHILE**, and **REPEAT**, or as data values in **D**, **V**, **A**, **AD**, **L**, or **T** commands. In step 7 above, the value 15 was entered stored in variable 1 to be used as the cycle count. The following is a command example:

<u>Command</u>	<u>Description</u>
D2400	Set distance to 2400 steps
V1	Set velocity to 1 rps
L (VAR1)	Loop the number of times as specified by variable 1
G	Initiate motion
T1	Wait one second
N	End the loop

In step 14, the operator pressed the **F1** key. This value was stored in variable 2, and was to be used to select PART1. The following is a command example:

<u>Command</u>	<u>Description</u>
IF (VAR2=1) XG11 NIF	Branch to sequence 11 if F1 is pressed
IF (VAR2=2) XG12 NIF	Branch to sequence 12 if F2 is pressed
IF (VAR2=3) XG13 NIF	Branch to sequence 13 if F3 is pressed
IF (VAR2=4) XG14 NIF	Branch to sequence 14 if F4 is pressed
IF (VAR2=5) XG15 NIF	Branch to sequence 15 if F5 is pressed
IF (VAR2=6) XG16 NIF	Branch to sequence 16 if F6 is pressed

For further information on the additional commands used in the examples above, refer to the appropriate user guide, or software reference guide for the Model 500, SX, or ZX.

Enabling STOP and PAUSE Keys

In addition to the function keys and numeric keypad, there are three other keys. The **STOP** key, and the **PAUSE** and **CONTINUE** keys, must be enabled before they can be used.

Typically, if an application uses the **STOP** key, the key will be enabled (**DSTP1**) in the power-up sequence (sequence #100). However, the **STOP** key can be enabled and disabled in any sequence, and at any time. The **STOP** key issues a Kill (**K**) command to the 500, SX, or ZX, immediately halting motion. **There is no controlled deceleration when using the STOP key.**

The **PAUSE** and **CONTINUE** keys can be enabled in any sequence. Before these keys will function as expected, you must enable the **commandSave Command Buffer on Stop (SSH1)** and **Resume Execution Enable (SSL1)** commands. The **DCNT** command enables the **PAUSE** and **CONTINUE** keys, and the commands **SSH1** and **SSL1** cause the **PAUSE** and **CONTINUE** keys to function in different ways.

If **SSH** is disabled (**SSH0**), the **PAUSE** key will function as a controlled stop. The motor will decelerate at the last programmed deceleration rate (**AD**) and the program buffer will be dumped. The program will be exited.

If you issue **SSH1** and **SSL0**, the **PAUSE** key will function as a controlled stop. After motion is halted, the **CONTINUE** key will not resume motion. Instead, it will resume command processing with the command directly following the command that was stopped with the **PAUSE** key.

 **Helpful Hint:**
Both the **SSH** and **SSL** commands should be placed in the power-up sequence (Sequence #100).

If **SSH1** and **SSL1** are issued, the **PAUSE** key will function as a controlled stop. After motion is halted, command processing can be continued by pressing the **CONTINUE** key. If motion was in progress when the **PAUSE** key was pressed, that motion will be resumed. This is the recommended way to use the **PAUSE** and **CONTINUE** keys.

Sample Program

This section provides an example of an RP240 application program. Refer to the specific indexer's Software Reference Guide for a more detailed description of the commands.

Power-Up Sequence

The power-up sequence (Sequence #100) is used to initialize the SX, 500, or ZX to a state compatible with the RP240. The power-up sequence is also used to initialize variables.

XE100	Erase sequence #100
XD100	Begin definition of sequence #100
LD0	Enable limits (only if connected, otherwise LD3)
SSH1	Save command buffer on stop
SSN0	Disable Message mode
DSTP1	Enable STOP key
VAR20=5	Variable 20 will be used for the high jog velocity
VAR21=1	Variable 21 will be used for the low jog velocity
XG1	Branch to sequence #1
XT	End definition of sequence #100

Sequence #1

Sequence #1 provides the main menu for a demonstration program. Control is transferred to another sequence based on a function key input.

XE1	Erase sequence #1
XD1	Begin definition of sequence #1
DCLR0	Clear all lines of the RP240 display
DPC106	Position cursor at row 1, column 6
DTXTCOMPUMOTOR_RP240_DEMO_PROGRAM	
DPC200	Position cursor at row 2, column 0
DTXTACCESS	
DPC235	Position cursor at row 2, column 35
DTXTEXIT	
VAR1=FUN	Wait for function key and place value in variable 1
IF (VAR1=1)	If function key 1 is pressed
XG2	Branch to sequence 2
NIF	End If
IF (VAR1=6)	If function key 6 is pressed
DCLR0	Clear all lines of the RP240 display
SSA0	Enable echo
SSI0	Enable interactive mode
HALT	Halt program execution
NIF	End If
DPC220	Position cursor at row 2, column 20
DTXTWRONG_BUTTON!	
T1	Time delay of 1 second
XG1	Branch to sequence #1
XT	End definition of sequence #1

Sequence #2

Sequence #2 prompts for a code number, and then transfers control to sequence #3. If the code number is incorrect, control is passed back to sequence #1.

XE2	Erase sequence #2
XD2	Begin definition of sequence #2
DCLR0	Clear all lines of the RP240 display
DPC100	Position cursor at row 1, column 0
DTXTENTER_CODE_NUMBER_XXXX	
DPC118	Position cursor at row 1, column 18
VAR1=NUM	Wait for numeric keypad input and place in variable 1
IF (VAR1<1234_OR_VAR1>1234)	If variable 1 is less than 1234 or greater than 1234
DPC220	Position cursor at row 2, column 20
DTXTWRONG_NUMBER	
T1	Time delay of 1 second
XG1	Branch to sequence #1
ELSE	Else part of If statement (Variable 1 equals 1234)
XG3	Branch to sequence #3
NIF	End If
XT	End definition of sequence #2

Sequence #3

Sequence #3 asks for a selection via the function keys. Control is passed to a sequence based upon the function key pressed.

XE3	Erase sequence #3
XD3	Begin definition of sequence #3
DCLRØ	Clear all lines of the RP240 display
DPC1ØØ	Position cursor at row 1, column 1
DTXTJOG ___ I/O ___ TEACH ___ MAKE ___ FEED ___ EXIT	
DPC2Ø1	Position cursor at row 2, column 1
DTXTAXES ___ TEST ___ MODE ___ MOVE ___ TO_LENGTH	
VAR1=FUN	Wait for function key and place value in variable 1
IF (VAR1=1) XR4 NIF	If function key 1 was pressed, gosub to sequence #4
IF (VAR1=2) XR5 NIF	If function key 2 was pressed, gosub to sequence #5
IF (VAR1=3) XR6 NIF	If function key 3 was pressed, gosub to sequence #6
IF (VAR1=4) XR7 NIF	If function key 4 was pressed, gosub to sequence #7
IF (VAR1=5) XR8 NIF	If function key 5 was pressed, gosub to sequence #8
IF (VAR1=6) XG1 NIF	If function key 6 was pressed, branch to sequence #1
XG3	Branch to sequence #3
XT	End definition of sequence #3

Sequence #4

Sequence #4 simulates jogging with CW or CCW jog options, and either low or high velocity. The default jog velocities are stored in variables 20 and 21, which were assigned in power-up sequences (sequence #100).

XE1	Erase sequence #4
XD1	Begin definition of sequence #4
A75 AD75	Set acceleration and deceleration to 75 rev/sec/sec
VAR1=Ø VAR2=Ø	Initialize variables 1 and 2 to zero
MC	Enable continuous mode
MPP	Enable position profiling mode
REPEAT	Repeat all the commands until the UNTIL condition is true
VAR3=Ø	Set variable 3 equal to zero
DCLRØ	Clear all lines of the RP240 display
DPC1Ø2	Position the cursor at row 1, column 2
DTXTCW ___ CW ___ CCW ___ CCW ___ STOP	
DPC2Ø1	Position the cursor at row 2, column 1
DTXTLOW ___ HIGH ___ LOW ___ HIGH ___ MOTION	
VAR1=FUN	Wait for function key and place value in variable 1
WHILE (VAR2<VAR1_OR_VAR2>VAR1_AND_VAR3=0)	
VØ	Set velocity to zero rev/sec
TØ.3	Wait for 0.3 seconds
IF (VAR1=1)	If function key 1 was pressed
H+ V (VAR21) G	Set direction, velocity, and initiate motion
NIF	End If
IF (VAR1=2)	If function key 2 was pressed
H+ V (VAR2Ø) G	Set direction, velocity, and initiate motion
NIF	End If
IF (VAR1=3)	If function key 3 was pressed
H- V (VAR21) G	Set direction, velocity, and initiate motion
NIF	End If
IF (VAR1=4)	If function key 4 was pressed
H+ V (VAR2Ø) G	Set direction, velocity, and initiate motion
NIF	End If
IF (VAR1=5)	If function key 5 was pressed
H+ V (VAR21) G	Set direction, velocity, and initiate motion
NIF	End If
IF (VAR1=1)	If function key 1 was pressed
DCLRØ	Clear all lines of the RP240 display
DPC212	Position the cursor at row 2, column 12
DTXTINVALID_SELECTION	
T1	Wait for 1 second
XG4	Branch to sequence #4
NIF	End If
VAR3=1	Set variable 3 to one
NWHILE	End WHILE loop
VAR2=VAR1	Set variable 2 equal to variable 1
UNTIL (VAR1=6)	Exit REPEAT UNTIL loop when function key 6 is pressed
VØ	Set velocity to zero
NG	Exit position profiling (MPP) mode
MN	Mode normal (exit mode continuous)
VAR1=Ø	Set variable 1 equal to zero
XT	End definition of sequence #4

Sequence #5

Sequence #5 provides input status information and then toggles the eight LEDs of the RP240 on and off.

```

XE5           Erase sequence #5
XD5           Begin definition of sequence #5
DCLRØ        Clear all lines of the RP240 display
DPC1ØØ       Position the cursor at row 1, column 0
DTXTINPUT_STATUS=
DPC2ØØ       Position the cursor at row 2, column 0
DTXTUPDATE
DPC233       Position the cursor at row 2, column 33
DTXTOUTPUTS
REPEAT       Repeat all the commands until the UNTIL condition is true
  VAR1=FUN   Wait for function key and place value in variable 1
  DPC113    Position the cursor at row 1, column 13
  IF(IN1) DTXT1 ELSE DTXTØ NIF
  IF(INX1) DTXT1 ELSE DTXTØ NIF
  IF(INXX1) DTXT1 ELSE DTXTØ NIF
  IF(INXXX1) DTXT1 ELSE DTXTØ NIF
  IF(INXXXX1) DTXT1 ELSE DTXTØ NIF
  IF(INXXXXX1) DTXT1 ELSE DTXTØ NIF
  IF(INXXXXXX1) DTXT1 ELSE DTXTØ NIF
  IF(INXXXXXXX1) DTXT1 ELSE DTXTØ NIF
UNTIL (VAR1=6) Exit REPEAT UNTIL loop when function key 6 is pressed
DCLRØ        Clear all lines of the RP240 display
DPC1ØØ       Position the cursor at row 1, column 0
DTXTTHE_LEDS_BELOW_WILL_CHANGE STATE
DLEDØØØØØØ1 T1 Turn on LED 8 and wait 1 second
DLEDØØØØØØ11 T1 Turn on LEDs 7 and 8 and wait 1 second
DLEDØØØØØØ111 T1 Turn on LEDs 6, 7 and 8 and wait 1 second
DLEDØØØØØØ1111 T1 Turn on LEDs 5, 6, 7 and 8 and wait 1 second
DLEDØØØØ111111 T1 Turn on LEDs 4, 5, 6, 7 and 8 and wait 1 second
DLEDØØØ1111111 T1 Turn on LEDs 3, 4, 5, 6, 7 and 8 and wait 1 second
DLEDØØ11111111 T1 Turn on LEDs 2, 3, 4, 5, 6, 7 and 8 and wait 1 second
DLEDØ111111111 T1 Turn on LEDs 1, 2, 3, 4, 5, 6, 7 and 8 and wait 1 second
VAR1=Ø       Set variable 1 equal to zero.
XT           End definition of sequence #5

```

Sequence #6

Sequence #6 demonstrates teaching moves that can be executed from sequence #7. This sequence uses the jog routine from sequence #4 to move to the desired locations. The distances taught are assigned to variables using other sequences (13, 14, and 15 in this example). The total number of moves taught is limited by the number of variables and the number of available sequences.

```

XE6           Erase sequence #6
XD6           Begin definition of sequence #6
MPA          Enable absolute positioning mode
MN           Enable preset mode
VAR28=13     Set variable 28 equal to 13 (First sequence to teach)
VAR29=1      Set variable 29 equal to 1
VAR3Ø=1      Set variable 30 equal to 1
REPEAT       Repeat all the commands until the UNTIL condition is true
  DCLRØ        Clear all lines of the RP240 display
  DPC1ØØ       Position the cursor at row 1, column 0
  DTXTHOW_MANY_MOVES_TO_TEACH?_x
  DPC2ØØ       Position the cursor at row 2, column 0
  DTXT(MAXIMUM_OF_THREE_MOVES)
  DPC125       Position the cursor at row 1, column 25
  VAR3Ø=NUM    Wait for numeric keypad input and place in variable 30
UNTIL (VAR3Ø>Ø AND VAR3Ø<4) Exit REPEAT UNTIL loop when number entered is 1,2 or 3
L (VAR3Ø)     Loop the number of times specified in variable 30
  PZ           Set the current position to absolute zero
  DCLRØ        Clear all lines of the RP240 display
  DPC1ØØ       Position the cursor at row 1, column 0
  DTXTPLEASE_MAKE_MOVE#_
  DVO29,2,Ø,Ø Display variable 29
  DPC2ØØ       Position the cursor at row 2, column 0
  T1           Wait 1 second
  XR4          Gosub to sequence #4
  XR (VAR28)   Gosub to the sequence specified in variable 28
  VAR28=VAR28+1 Increment variable 28 by 1
  VAR29=VAR29+1 Increment variable 29 by 1
N           End loop
VAR1=Ø       Set variable 1 equal to zero.
XT           End definition of sequence #6

```

Sequence #7

Sequence #7 performs the moves taught in sequence #6. The distance for each move is based upon the distance that was stored in a variable in sequence #6.

XE7	Erase sequence #7
XD7	Begin definition of sequence #7
A100	Set acceleration to 100 rev/sec/sec
AD100	Set deceleration to 100 rev/sec/sec
MPI	Enable incremental position mode
PE	Set the current position to absolute zero
V10	Set velocity to 10 rev/sec
VAR29=20	Set variable 29 equal to twenty
L (VAR30)	Loop the number of times specified by variable 30
XR (VAR29)	Gosub to the sequence specified by variable 29
G	Initiate motion
VAR29=VAR29+1	Increment variable 29
N	End loop
VAR1=0	Set variable 1 equal to zero.
XT	End definition of sequence #7

Sequence #8

Sequence #8 prompts the operator to select the feed length, maximum speed and count of items to be cut.

XE8	Erase sequence #8
XD8	Begin definition of sequence #8
DCLR0	Clear all lines of the RP240 display
DPC101	Position the cursor at row 1, column 1
DTXTSELECT_FEED_LENGTH, _MAX_SPEED, _&_COUNT	
T2	Wait for 2 seconds
REPEAT	Repeat all the commands until the UNTIL condition is true
DCLR0	Clear all lines of the RP240 display
VAR22=0	Set variable 22 equal to zero (used to exit repeat loop)
DPC102	Position the cursor at row 1, column 2
DTXTENTER_FEED_LENGTH_IN_INCHES:_(0-99.9)	
DPC212	Position the cursor at row 2, column 12
DTXTFEED_LENGTH=xx.x	
DPC224	Position the cursor at row 2, column 24
VAR2=NUM	Wait for numeric keypad input and place in variable 2
IF (VAR2>99.9_OR_VAR2<0)	If variable 2 is greater than 99.9 or less than 0
DCLR0	Clear all lines of the RP240 display
DPC215	Position the cursor at row 2, column 15
DTXTOUT_OF_RANGE	
T2	Wait for two seconds
ELSE	Else (variable 2 is greater than 0 and less than 99.9)
DCLR0	Clear all lines of the RP240 display
DPC112	Position the cursor at row 1, column 12
DTXTFEED_LENGTH=	
DPC125	Position the cursor at row 1, column 25
DVO2, 2, 1, 0	Display variable 2
DPC202	Position the cursor at row 2, column 2
DTXTYES	
DPC236	Position the cursor at row 2, column 36
DTXTNO	
VAR22=FUN	Wait for a function key to be pressed
NIF	End If
UNTIL (VAR22=1)	Exit REPEAT UNTIL loop when variable 22 equals one
REPEAT	Repeat all the commands until the UNTIL condition is true
DCLR0	Clear all lines of the RP240 display
VAR22=0	Set variable 22 equal to zero (used to exit repeat loop)
DPC101	Position the cursor at row 1, column 1
DTXTENTER_MAX_SPEED_(RPM):_(0-2400RPM)	
DPC215	Position the cursor at row 2, column 15
DTXTMAX_SPEED=xxxx	
DPC225	Position the cursor at row 2, column 25
VAR3=NUM	Wait for numeric keypad input and place in variable 3
IF (VAR3>2400_OR_VAR3<0)	If variable 3 is greater than 2400 or less than 0
DCLR0	Clear all lines of the RP240 display
DPC215	Position the cursor at row 2, column 15
DTXTOUT_OF_RANGE	
T2	Wait for two seconds
ELSE	Else (variable 3 is greater than 0 and less than 2400)
DCLR0	Clear all lines of the RP240 display
DPC114	Position the cursor at row 1, column 14
DTXTMAX_SPEED=	
DPC125	Position the cursor at row 1, column 25
DVO3, 4, 2, 0	Display variable 3
DPC202	Position the cursor at row 2, column 2

<pre> DTXTYES DPC236 DTXTNO VAR2=FUN NIF UNTIL (VAR22=1) REPEAT DCLRØ VAR22=Ø DPC1Ø3 DTXTENTER_TOTAL_NUMBER_OF_CUTS:_ (1-1ØØ) DPC21Ø DTXTTOTAL_#_OF_CUTS=xxx DPC226 VAR4=NUM IF (VAR4>1ØØ_OR_VAR4<1) DCLRØ DPC215 DTXTOUT_OF_RANGE T2 ELSE DCLRØ DPC113 DTXT#_OF_CUTS= DPC124 DVO4,3,Ø,Ø DPC2Ø2 DTXTYES DPC236 DTXTNO VAR2=FUN NIF UNTIL (VAR22=1) REPEAT DCLRØ DPC2Ø1 DTXTSTART DPC235 DTXTEXIT VAR22=FUN IF (VAR22=1_OR_VAR22=6) ELSE DCLRØ DPC212 DTXTINVALID_SELECTION T2 NIF UNTIL (VAR22=1_OR_VAR22=6) IF (VAR22=6) XG3 NIF DCLRØ VAR5=VAR2*25ØØØ VAR6=VAR3/6Ø VAR7=VAR4 LD3 MN A1ØØ AD1ØØ V (VAR6) D (VAR5) DPC211 DTXT# LEFT_TO_CUT: L (VAR4) DPC225 DVO7,3,Ø,Ø G VAR7=VAR7-1 T.1 N DCLRØ DPC213 DTXTJOB_COMPLETED! T1.5 XG3 XT </pre>	<pre> Position the cursor at row 2, column 36 Wait for a function key to be pressed End If Exit REPEAT UNTIL loop when variable 22 equals one Repeat all the commands until the UNTIL condition is true Clear all lines of the RP240 display Set variable 22 equal to zero (used to exit repeat loop) Position the cursor at row 1, column 1 Position the cursor at row 2, column 10 Position the cursor at row 2, column 26 Wait for numeric keypad input and place in variable 4 If variable 4 is greater than 100 or less than 1 Clear all lines of the RP240 display Position the cursor at row 2, column 15 Wait for two seconds Else (variable 4 is greater than 1 and less than 100) Clear all lines of the RP240 display Position the cursor at row 1, column 13 Position the cursor at row 1, column 24 Display variable 3 Position the cursor at row 2, column 2 Position the cursor at row 2, column 36 Wait for a function key to be pressed End If Exit REPEAT UNTIL loop when variable 22 equals one Repeat all the commands until the UNTIL condition is true Clear all lines of the RP240 display Position the cursor at row 2, column 1 Position the cursor at row 2, column 35 Wait for a function key to be pressed If variable 1 equals 1 or 6 do nothing Else, display an error message Clear all lines of the RP240 display Position the cursor at row 2, column 12 Wait two seconds End If Exit REPEAT UNTIL loop when variable 22 equals 1 or 6 If variable 22 equals 6 Branch to sequence #3 End If Clear all lines of the RP240 display Variable 5 equals distance, assuming 25000 steps/inch Variable 6 equals velocity in RPM Using variable 7 as count down value Enable limits Mode normal (preset moves) Set acceleration to 100 rev/sec/sec Set deceleration to 100 rev/sec/sec Set velocity to the value specified by variable 6 Set distance to the value specified by variable 5 Position the cursor at row 2, column 11 Loop the number of times specified by variable 4 Position the cursor at row 2, column 25 Display variable 7 Initiate motion Decrement variable 7 Wait 0.1 seconds End loop Clear all lines of the RP240 display Position the cursor at row 2, column 13 Set distance to the value specified by variable 5 Branch to sequence #3 End definition of sequence #8 </pre>
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Sequences #13,#14,#15

Sequences #13, #14, and #15 store the current position into variables 6, 7, and 8, respectively.

XE13	Erase sequence #13
XD13	Begin definition of sequence #13
VAR6=POS	Variable 6 equals the current position
XT	End definition of sequence #13
XE14	Erase sequence #14
XD14	Begin definition of sequence #14
VAR7=POS	Variable 7 equals the current position
XT	End definition of sequence #14
XE15	Erase sequence #15
XD15	Begin definition of sequence #15
VAR8=POS	Variable 8 equals the current position
XT	End definition of sequence #15

Sequences #20,#21,#22

Sequences #20, #21, and #22 set distance to the value stored in variables 6, 7, and 8, respectively.

XE20	Erase sequence #20
XD20	Begin definition of sequence #20
D (VAR6)	Set distance to the value stored in variable 6
XT	End definition of sequence #20
XE21	Erase sequence #21
XD21	Begin definition of sequence #21
D (VAR7)	Set distance to the value stored in variable 7
XT	End definition of sequence #21
XE22	Erase sequence #22
XD22	Begin definition of sequence #22
D (VAR8)	Set distance to the value stored in variable 8
XT	End definition of sequence #22

Editing Sequences

If you wish to edit sequences in your SX, ZX, or Model 500, you must re-establish RS-232C communication *before* downloading the new sequences to the controller. Compumotor recommends using the following procedure.

- ① Remove power from your controller and the RP240.
- ② Remove connections between the controller and the RP240.
- ③ Connect Rx, Tx, and Ground between the controller and your RS-232C port.
- ④ Apply power to your controller.
- ⑤ Enter the TERMINAL EMULATOR portion of your communication software (preferably Compumotor's X-Ware).

Initially you will not be able to establish communications if you have turned off command echo (SSA1) in your power-up sequence.

- ⑥ If using X-Ware Device Not Ready will appear on your screen (if you have disabled the command echo). Press the <ENTER> key. Next, press the Y key to enter the Terminal Emulator.
- ⑦ To terminate the power-up sequence, and turn on the command echo as well as the interactive mode, issue the following commands:


```

K <cr>
SSA0 <cr>
SSI0 <cr>

```
- ⑧ Check the controller's status by issuing a **aR** command, where **a** is the device address (**1R <cr>**, for example). The controller should respond with a ***R**. If you do not receive a ***R**, consult the RS-232C Troubleshooting section of your controller user guide.
- ⑨ Edit the sequence as usual (If using X-Ware, hit the <ESC> key, then use arrow keys to select the Editor menu).

Power-Up Sequence

The power-up sequence (Sequence #100) is used to initialize the SX, 500, or ZX to a state compatible with the RP240.

Command	Description
1XE1ØØ	Erase sequence #100
1XD1ØØ	Begin definition of sequence #100
1LD3	Disable limits (Use LDØ if limits are hard wired)
1SSA1	Disable command echo
1SSH1	Save command buffer on STOP
1SSI1	Disable Interactive mode
1"2E	Enable communications on axis 2
1"2LD3	Disable limits on axis 2
1"2SSAØ	Enable command echo on axis 2
1"2SSIØ	Enable interactive mode on axis 2
1"2F	Disable communications on axis 2
1DSTPØ	Disable STOP key
1DCNTØ	Disable PAUSE and CONTINUE keys
1XG1	Branch to sequence #1
1XT	End definition of sequence #100

Sequence #1

Sequence #1 instructs you to enter the axis to jog. The you are allowed to jog axis one or axis two.

```

1XE1
1XD1
1A1ØØ
1V4
1MC
1"2E
1"2A1ØØ
1"2V2
1"2MC
1"2F
1DCLRØ
1DPC1Ø1
1DTXTWHICH_AXIS_DO_YOU_WANT_TO_JOG?
1DPC2ØØ
1DTXTAXIS_1__AXIS_2
1DPC235
1DTXTEXTIT
1L
1VAR1=FUN
1IF (VAR1=1) 1XG2 1NIF
1IF (VAR1=2) 1XG3 1NIF
1IF (VAR1=6) 1DCLRØ 1SSAØ 1SSIØ 1"2E 1HALT 1NIF
1N
1XT

```

Sequence #2

Sequence #2 jogs axis 1.

```

XE2
XD2
1DCLRØ
1DPC1Ø2
1DTXTJOG ___ STOP ___ JOG
1DPC2Ø2
1DTXTCCW _____ CW
1DPC235
1DTXTEXTIT
1MPP
1L
1VAR1=FUN
1IF (VAR1=1) 1H- 1G 1NIF
1IF (VAR1=2) 1STOP 1NIF
1IF (VAR1=3) 1H+ 1G 1NIF
1IF (VAR1=6) 1STOP 1XG1 1NG 1NIF
1N
1NG
XT

```

Sequence #3

Sequence #3 jogs axis 2.

```
XE3
XD3
1DCLR0
1DFC102
1DXTJOG___STOP___JOG
1DFC202
1DXTCCW_____CW
1DFC235
1DXTXIT
1MPP
1L
1VAR1=FUN
1IF (VAR1=1) 1"2E 1"2H- 1"2G 1"2F 1NIF
1IF (VAR1=2) 1"2E 1"2S 1"2F 1NIF
1IF (VAR1=3) 1"2E 1"2H+ 1"2G 1"2F 1NIF
1IF (VAR1=6) 1"2E 1"2S 1"2F 1XG1 1NG 1NIF
1N
1NG
XT
```

C H A P T E R ⑤

6000 Series Command Summary and Application Design

Chapter Objectives

The information in this chapter will enable you to:

- Program the 6200 or 6250 to control the RP240
- Customize the system to meet your requirements

PROGRAMMING NOTE

This chapter provides the 6000 Series programming information that is required to communicate with the RP240. If you are using an Extended X product (SX, ZX, or 500), refer to *Chapter ④ Extended X Command Summary and Application Design*.

6000 Series Command Summary

Command	Description
DCLEAR	Clear the display
DJOG	Enable/Disable RP240 Jog Mode
DLED	Turn RP240 LEDs On/Off
DPASS	Change RP240 Password
DPCUR	Position Cursor
DREAD	Read RP240 Data from Numeric Keypad
DREADF	Read RP240 Function Keys
DVAR	Display Variable on RP240 LCD
DWRITE	Write Text to RP240 LCD

Description 6000 Series Command Format

The following is a definition of the format fields for the 6000 Series commands.

① DCLEAR ② Clear Display

③ Product	Rev
6200	1.0
6250	1.0

- ④ **Type** Display (RP240) Interface
- ⑤ **Syntax** <!>DCLEARin/a
- ⑥ **Units** n/a
- ⑦ **Range** i = 0 (clear all lines), 1 (clear line 1), or 2 (clear line 2)
- ⑧ **Default** n/a
- ⑨ **Response** n/a
- ⑩ **See Also** DLED, DPASS, DPCUR, DVAR, DWRITE

Field Number	Field Description
①	Mnemonic Code: This field contains the command's mnemonic code.
②	Full Name: This field contains the command's full name.
③	Valid Product & Revision: This field lists the 6000 Series products and the revision of each product when this command was incorporated. If the command does not apply to that particular product, the Rev is specified as n/a.
④	Type: This field contains the command's type. Since we are working with the RP240, this field will always display <i>Display (RP240) Interface</i> .
⑤	Syntax: The proper syntax for the command is shown here. The specific parameters associated with the command are also shown.
⑥	Units: This field describes what unit of measurement the parameter (b, d, i, r, or t) in the command syntax represents.
⑦	Range: This is the range of valid values that you can specify for an argument (or any other parameter specified).
⑧	Default: The default setting for the command is shown in this field. A command will perform its function with the default setting if you do not provide a value.
⑨	Response: Some commands allow you to check the status of the command. In the example above, entering the DCLEAR command by itself will result in an error message. The example responses provided are based on the default error level, Error Level 4, established with the ERRLEV command in the <i>6000 Series Software Reference Guide</i> .
⑩	See Also: Commands related or similar to the command described are listed here. Refer to the <i>6000 Series Software Reference Guide</i> for additional information.

RP240 X Language Command Listing

		Product	Rev
DCLEAR	Clear Display	6200	1.0
		6250	1.0
Type	Display (RP240) Interface		
Syntax	<!>DCLEARi		
Units	n/a		
Range	i = 0 (clear all lines), 1 (clear line 1), or 2 (clear line 2)		
Default	n/a		
Response	n/a		
See Also	DLED, DPASS, DPCUR, DVAR, DWRITE		

The Clear Display (DCLEAR) command clears lines (as specified with i) of the RP240 display: After clearing a line, the cursor will be reset to the beginning of that line (or to the beginning of line 1 if all lines are cleared).

		Product	Rev
DJOG	Enable RP240 Jog Mode	6200	1.0
		6250	1.0
Type	Display (RP240) Interface		
Syntax	<!>DJOG		
Units	n/a		
Range	b = 0 (disable) or 1 (enable)		
Default	n/a		
Response	0408 DJOG: *DJOG1		
See Also	JOG, JOGA, JOGAD, JOGVH, JOGVL		

The DJOG command allows you to branch into the RP240 front panel jog mode from within your user-defined program, adjust the position of the axes, and then return to program execution.

The DJOG1 command enables the RP240 jog mode on all axes. Once the RP240 jog mode is enabled, you can use the RP240 arrow keys to jog individual axes. Unlike the JOG command, command processing is suspended after the DJOG1 command is issued. Jogging is performed with the parameters set with the Jog Acceleration (JOGA) and Jog Deceleration (JOGAD) commands.

To disable the RP240 jog mode, press the MENU RECALL key or issue the immediate !DJOG0 command. Upon exiting the RP240 jog mode, the RP240's display is cleared

To have the jog mode continually enabled during program execution, you must use jog inputs and the JOG command.

		Product	Rev
DLED	Turn RP240 Display LEDs On/Off	6200	1.0
		6250	1.0
Type	Display (RP240) Interface		
Syntax	<!>DLED		
Units	n/a		
Range	b = 0 (off) or 1 (on)		
Default	n/a		
Response	DLED: *DLED1101_0001		
See Also	DCLEAR, DPASS, DPCUR, DVAR, DWRITE		

The DLED command controls the state of the 8 programmable LEDs on the RP240. It is legal to substitute a binary variable (VARB) for the DLED command

Example

```
> DLED11XXXX01
> VARB1=b10101010
> DLED(VARB1)
```

Description

Turn on LEDs 1, 2, and 8; turn off LED 7; leave LEDs 3,4,5, and 6 unchanged
Set bits 1, 3, 5 & 7 low, and bits 2, 4, 6, & 8 high
Turn on LEDs 1, 3, 5 & 7; turn off LEDs 2, 4, 6, & 8

DPASS**Change RP240 Password**

<u>Product</u>	<u>Rev</u>
6200	1.0
6250	1.0

Type Display (RP240) Interface
Syntax <!>DPASS<i>
Units i = integer of up to 4 characters
Range 1 - 9999
Default 6200 or 6250 (depending on if you have a 6200 or 6250)
Response DPASS: *DPASS6200
See Also DCLEAR, DLED, DPCUR, DVAR, DWRITE

The DPASS command changes the RP240 password. If the default password is not changed by the user, there will be no password protection.

Example

> DPASS1234

Description

New password = 1234

DPCUR**Position Cursor**

<u>Product</u>	<u>Rev</u>
6200	1.0
6250	1.0

Type Display (RP240) Interface
Syntax <!>DPCURi,i
Units 1st i = line number, 2nd i = column
Range line number = 1 or 2, column = 0 - 39
Default n/a
Response n/a
See Also DCLEAR, DLED, DPASS, DVAR, DWRITE

The DPCUR command changes the location of the cursor on the RP240 display. The RP240 lines are numbered from top to bottom, 1 to 2. The columns are numbered left to right, 0 to 39.

Example

> DPCUR2,15

Description

Position cursor on line 2, column 15

[DREAD] Read RP240 Data

<u>Product</u>	<u>Rev</u>
6200	1.0
6250	1.0

Type Display (RP240) Interface
Syntax See below
Units n/a
Range n/a
Default n/a
Response n/a
See Also DREADF, DVAR, DWRITE, VAR

The Read RP240 Data (DREAD) command allows you to store numeric data entered in from the RP240's keypad into a variable. As the user presses RP240 numeric keys, the data will be displayed on the RP240 starting at the location equal to the current cursor location + 1 (for a sign bit):

VAR1=DREAD Wait for RP240 numeric entry (terminated with the ENTER key), then set VAR1 equal to that value.

Additionally the DREAD command can be used as a variable assignment within another command that is expecting numeric data:

A(DREAD),5.0 Wait for RP240 numeric entry (terminated with the ENTER key), then set axis #1 acceleration to that value and set axis #2 acceleration to 5.0.

The DREAD command cannot be used in an expression such as VAR5=4+DREAD or IF(DREAD=1).

[DREADF] Read RP240 Function Key

Product	Rev
6200	1.0
6250	1.0

Type Display (RP240) Interface
Syntax See below
Units n/a
Range n/a
Default n/a
Response n/a
See Also DREAD, DVAR, DWRITE, VAR

The Read RP240 Function Key (DREADF) command allows you to store numeric data entered in from a RP240 function key into a variable. Function key 1 (F1) = 1, F2 = 2, etc., and MENU RECALL (F0) = 0.

Example

```
> VAR1=DREADF  
  
> IF (VAR1=5)  
GOx1  
NIF
```

Description

Wait for RP240 function key entry, then set VAR1 equal to that value
If function key 5 was hit then ...
Start motion on axis #2
End if statement

DVAR Display Variable on RP240

Product	Rev
6200	1.0
6250	1.0

Type Display (RP240) Interface
Syntax <!>DVARi,<i>,<i>,<i>
Units See below
Range n/a
Default See below
Response n/a
See Also DREAD, DREADF, DWRITE, VAR

The Display Variable on RP240 (DVAR) command is used to display a numeric variable on the RP240's LCD at the current cursor location:

- 1st i = Variable number [Range 1 - 150]
- 2nd i = Number of whole digits displayed (left of decimal point) [Range 0 - 9]
- 3rd i = Number of fractional digits displayed (right of decimal point) [Range 0 - 8]
- 4th i = Sign bit: 0 = no sign displayed, 1 = display + or -

Example

```
> VAR2=542.14  
> DVAR2,6,3,1  
> DVAR2,3,1,0  
> DVAR2,3,,1
```

Description

Assign the value 542.14 to variable #2
Display variable #2 as +000542.140
Display variable #2 as 542.1
Display variable #2 as +542

DWRITE Write Text on RP240

Product	Rev
6200	1.0
6250	1.0

Type Display (RP240) Interface
Syntax <!>DWRITE"message"
Units n/a
Range Message can be ≤ 70 characters (may not use characters ", \ or :)
Default See below
Response n/a
See Also DCLEAR, DLED, DPASS, 'DPCUR, DVAR

The Write Text on RP240 (DWRITE) command displays a message on the RP240's LCD starting at the current cursor location. A message is a character string of up to 70 characters in length. The characters within the string may be any characters except quote ("), backslash (\), and colon (:). Strings that have lower-case letters will be converted to upper case prior to display (see example).

Example

```
> DCLEAR0  
> DPCUR1,12  
> DWRITE"Enter Number of Parts"  
> VAR1=DREAD
```

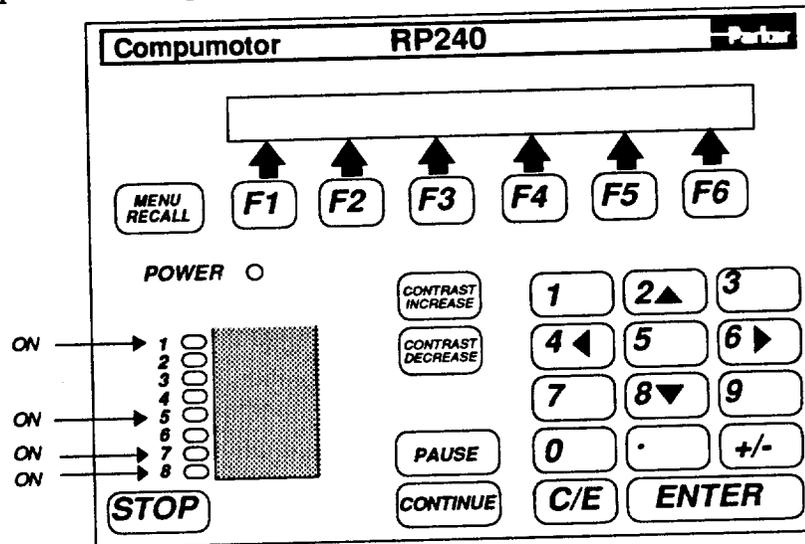
Description

Clear RP240 display
Move cursor to line 1, column 12
RP240 will display: ENTER NUMBER OF PARTS
RP240 waiting for data entry

Prompting an Operator or Displaying Information

In many motion control applications, the most important requirement is the operator interface. Presenting information to an operator in a desired format is often difficult at best. The RP240 has two visual indicators to help present information to the operator. The simplest indicator is the 8 LEDs on the panel. These LEDs can be turned on or off with the **DLED** command. The LEDs can be used in conjunction with the outputs to show the state of an output, or they can be used to show status, such as motor moving, specific sequence in progress, etc.

IF a **DLED10001011** command is issued, the LEDs shown below would be illuminated. These eight LEDs can be labeled, using the slide-in card provided, to represent cycle status, output status, etc.



The other indicator is the two line, 40-character LCD display. The Position Cursor (**DPCUR**) command allows the user to program the location of the cursor on the LCD display. The Write Text Data on RP240 Display (**DWRITE**) command allows the user to place text, beginning at the current cursor location, on the LCD display.

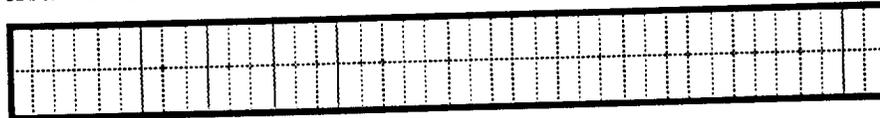
6000 Series Command Programming Example

A user wants his operator to see the message **ENTER THE CYCLE COUNT**. He wants this message placed on line two, starting after two spaces in from the left. He also wants the user to be able to enter the cycle count three spaces after the message. Below are the steps required to accomplish this.

The cursor does not appear on the display. The cursor is displayed when the VARn=DREAD command is used.

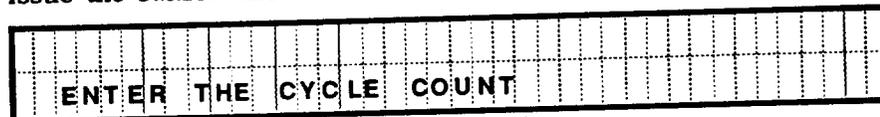
Step 1

Issue the **DPCUR2,2** command



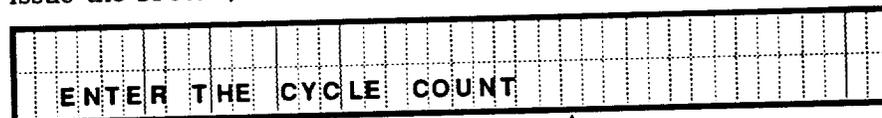
Step 2

Issue the **DWRITE"ENTER THE CYCLE COUNT"** command



Step 3

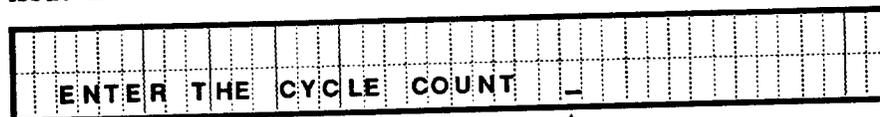
Issue the **DPCUR2,25** command



Now that we have provided the operator with the prompting message, how do we obtain the information? The Read RP240 Data (**VARn=DREAD**) or Read RP240 Function Keys (**VARn=DREADF**) commands provide the answer. The **VARn=DREAD** command will allow the operator to enter information. The numbers, as entered, will be displayed at the current cursor location.

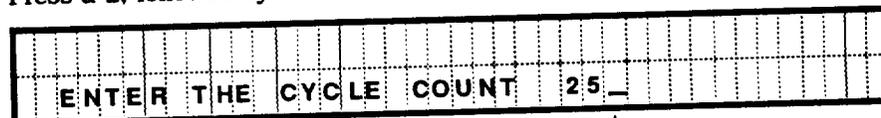
Step 4

Issue the **VAR1=DREAD** command



Step 5

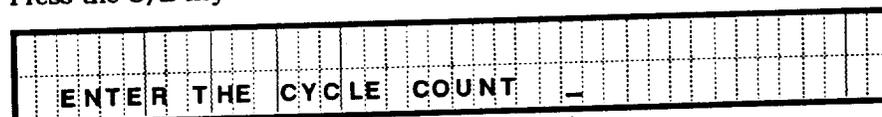
Press a 2, followed by a 5



If the wrong value is entered press the C/E key and re-enter the value.

Step 6

Press the C/E key



Direct RP240 Control

Chapter Objectives

The information in this chapter will enable you to:

- ❑ Control the RP240 without the use of an Extended X or 6000 Series product

PROGRAMMING NOTE

The information provided in this chapter is intended for users of the RP240 that do not have an Extended X or 6000 Series product. If you are using a Extended X product (SX, ZX, or 500), refer to *Chapter ④ Extended X Command Summary and Application Design*. If you are using a 6000 Series product (6200 or 6250) refer to *Chapter ⑤ 6000 Series Command Summary and Application Design*.

RP240 Overview

The RP240 uses a Intel 8051 microprocessor to respond to keypress information and to display data on the LCD. The microprocessor receives commands via RS-232C (9600 baud, 8 data bits, 1 stop bit, no parity) from a "master" controller. The RP240 acts as a "slave" to the "master". There is no program storage and no program execution capability in the RP240. The RP240 can only parse and execute the commands summarized below.

Command Summary

There are two modes of operation for the RP240. The default mode is the Extended X mode. In this mode, the function keys, numeric keypad, stop key, and pause and continue keys must all be enabled before they can be used. The RP240 will not report any keystrokes until the corresponding key or group of keys has been enabled.

The other mode of operation for the RP240 is the 6000 mode. In this mode, the RP240 will report back every key that was pressed and every key that was released.

If the desired mode of operation is Extended X, simply use the commands listed below. If the desired mode of operation is 6000, then the command ***B<cr>** must be issued to the RP240.

The commands below only require sending an asterisk, followed by a letter, and a carriage return. All the commands below are valid with Extended X mode.

Command	Description	Valid in 6000 Mode
*B<cr>	Enable 6000 mode	Yes
*C<cr>	Report # of RP240 display lines	Yes
*D<cr>	STOP key enable	-
*E<cr>	STOP key disable	-
*F<cr>	PAUSE and CONTINUE key enable	-
*G<cr>	PAUSE and CONTINUE key disable	-
*H<cr>	Beeper on	Yes
*I<cr>	Beeper off	Yes
*J<cr>	Cursor on, not blinking	Yes
*K<cr>	Turn LCD display off	Yes
*L<cr>	Turn LCD display on	Yes
*M<cr>	Clear display	Yes
*N<cr>	Test display	Yes
*O<cr>	Cursor on, blinking	Yes
*Q<cr>	Cursor off	Yes
*R<cr>	Clear line 1	Yes
*S<cr>	Clear line 2	Yes
*W<cr>	Report back current revision	Yes
*X<cr>	Enable numeric keypad	-
*Y<cr>	Enable function keys	-
*^<cr>	Move line 1 to line 2	Yes
*a<cr>	Move line 2 to line 1	Yes

The following commands require more detailed command parameters.

Command	Description	Valid in 6000 mode
*A<data><cr>	Update LEDs	Yes
*P<data><cr>	Position cursor	Yes
*V<data><cr>	Display text string	Yes

***A Command**

The Update LEDs command (*A) requires a hexadecimal value to indicate which LEDs will be **off**. The format of the command is *A, followed by the 1 byte (1 character) hexadecimal value, followed by a carriage return (<cr>). From top to bottom, the LEDs correspond to the following hexadecimal values:

LED #	Hexadecimal Value	Decimal Value
1	80	128
2	40	64
3	20	32
4	10	16
5	08	8
6	04	4
7	02	2
8	01	1

To turn off LED 3 and leave LEDs 1,2,4,5,6,7, and 8 on, issue *A<sp><cr>. The space key corresponds to ASCII 32.

***P Command**

The Position Cursor command (*P) requires 3 ASCII characters to indicate which location to move the cursor. The format of the command is *P, followed by 3 characters representing the line and column number, followed by a carriage return. The first of the three characters is the line number, the next two characters are the column number. Line 1, column 15 would correspond to *P115<cr>. Line 2, column 26 would correspond to *P226<cr>. The columns are numbered from 00 to 39, not 01 to 40.

***v Command**

The Display Text String command (*v) requires a text string to be displayed. The format of the command is *V, followed by the text string, followed by a carriage return. (i.e., *VThis is a demo!<cr>) The display text string (*v) command cannot display the underscore () character.

RP240 Response—Extended X Mode

The commands below enable the RP240 for specific functions in Extended X mode (default) only. Once these functions are enabled, the RP240 will respond with the information provided below.

*D<cr> STOP key enable
 *F<cr> PAUSE and CONTINUE key enable
 *X<cr> Enable numeric keypad
 *Y<cr> Enable function keys

When pressed, the STOP key will return a **space**, followed by a **K**, followed by a **line feed and carriage return**.

When pressed, the PAUSE key will return a **space**, followed by a **S**, followed by a **line feed and carriage return**.

When pressed, the CONTINUE key will return a **space**, followed by a **C**, followed by a **line feed and carriage return**.

The numeric keypad will return an **exclamation point (!)** when enabled with the *X<cr> command. As the number keys, decimal key, or plus/minus key are pressed, the corresponding character will be transmitted. When the ENTER key is pressed, a **line feed and carriage return** will be transmitted. If the C/E key is pressed, a single quote('), followed by a **line feed and carriage return**, followed by an **exclamation point (!)** will be returned.

The function keys will return an **exclamation point (!)** when enabled with the *Y<cr> command. The number of the function key (Ø for MENU RECALL, 1 for F1, 2 for F2, etc.) will be returned, along with a **line feed and carriage return**, when the function key is pressed.

Key Press Response of RP240—6000 Mode

Once 6000 mode has been enabled with the *B<cr> command, each key pressed on the RP240 will return a 1 byte hexadecimal value. When the key is released, another 1 byte hexadecimal value will be returned.

Key	Key Press Hexadecimal	Key Release Hexadecimal
STOP	5B	DB
PAUSE	5C	DC
CONTINUE	5D	DD
C/E	2F	AF
ENTER	3D	BD
0	30	B0
1	31	B1
2	32	B2
3	33	B3
4	34	B4
5	35	B5
6	36	B6
7	37	B7
8	38	B8
9	39	B9
+/-	2D	AD
.	2E	AE
F1	21	A1
F2	22	A2
F3	23	A3
F4	24	A4
F5	25	A5
F6	26	A6
MENU RECALL	2C	AC

C H A P T E R 7

Hardware Reference

Chapter Objectives

The information in this chapter will enable you to:

- Use this chapter as a quick-reference tool for most system specifications (environmental, I/O, pin outs, etc.)

Environmental

- Operating Temperature: 36°F to 122°F (5°C to 50°C)
- Storage Temperature: -22°F to 185°F (-30°C to 85°C)
- Humidity: 0% to 95% non-condensing

Electrical Specifications

RP240 Connector

The table below contains pin outs for the RP240's 4-pin Power and RS-232C connector on the back of the RP240. The pin out is listed from top to bottom.

Pin	Input/Output	Description
1	IN	Ground
2	IN	Receive (Rx)
3	OUT	Transmit (Tx)
4	IN	+5VDC

Power

You must supply 5 VDC \pm 2% to pin 1 of the RP240. The power supply must be capable of providing 100 mA of current. You can use the 5 VDC power supplies on board the SX, ZX, or 500 Indexer provided there is enough current remaining. The 6000 Series products have no current restrictions. Current levels are shown in the table below.

	Available	Current Requirements		
		Encoder	I/O	RP240
SX	250 mA	150 mA	10 mA each	100 mA
ZX	500 mA	150 mA	0 mA	100 mA
500	500 mA	150 mA	0 mA	100 mA

CAUTION

The SX's on-board power supply may not provide adequate current for the RP240 if the power supply is also providing power to an encoder and/or I/O.

RS-232C Communication

Refer to *Chapter 3 Installation* for RS-232C wiring instructions. RS-232C communication parameters for all Compumotor Controllers are fixed at:

- 9600 Baud Rate
- 8 Data Bits
- 1 Stop Bit
- No Parity

C H A P T E R ⑧

Troubleshooting

Chapter Objectives

The information in this chapter will enable you to:

- Isolate and resolve system hardware problems
- Isolate and resolve system software problems

PROM Table

In order to use the RP240, the Extended X product must have a PROM with a revision greater than or equal to the ones shown in the table below. If your unit does not have the correct PROM, call the Compumotor Applications Department at 1-800-358-9070 for more information.

Description	PROM Number and Revision
Model 500	92-011512-01 C
Model 500-FOL	92-011513-01 C
SX Indexer/Drive	92-011640-01 C2
SXF Indexer/Drive	92-012039-01 C2
ZX/ZXF Indexer Drive	92-011492-01 D

Troubleshooting

When your system does not function properly (or as you expect it to operate), the first thing that you must do is identify and isolate the problem. When you have accomplished this, you can effectively begin to resolve the problem.

The first step is to isolate each system component and ensure that each component functions properly when it is run independently. You may have to dismantle your system to detect the problem. If you have additional units available, you may want to exchange them with existing components in your system to help identify the source of the problem.

Determine if the problem is mechanical, electrical, or software-related. Can you repeat or re-create the problem? Do not make quick rationalizations about problems. Random events may appear to be related, but they are not necessarily contributing factors to your problem. Carefully investigate and decipher the events that occur before the subsequent system problem.

You may be experiencing more than one problem. You must isolate and solve one problem at a time. Log (document) all testing and problem isolation procedures. You may need to review and consult these notes later. This will also prevent you from duplicating your testing efforts.

If you are having difficulty isolating a problem, be sure to document all occurrences of the problem with specific information (time of occurrence, RP240 status, and anything else that was happening when the problem occurred).

Refer to the problem solutions contained in this chapter. If your system's problem persists, contact Parker Compumotor's Applications Department at (800) 358-9070.

Reducing Electrical Noise

For detailed information on reducing electrical noise in your system, refer to the current Compumotor Catalog.

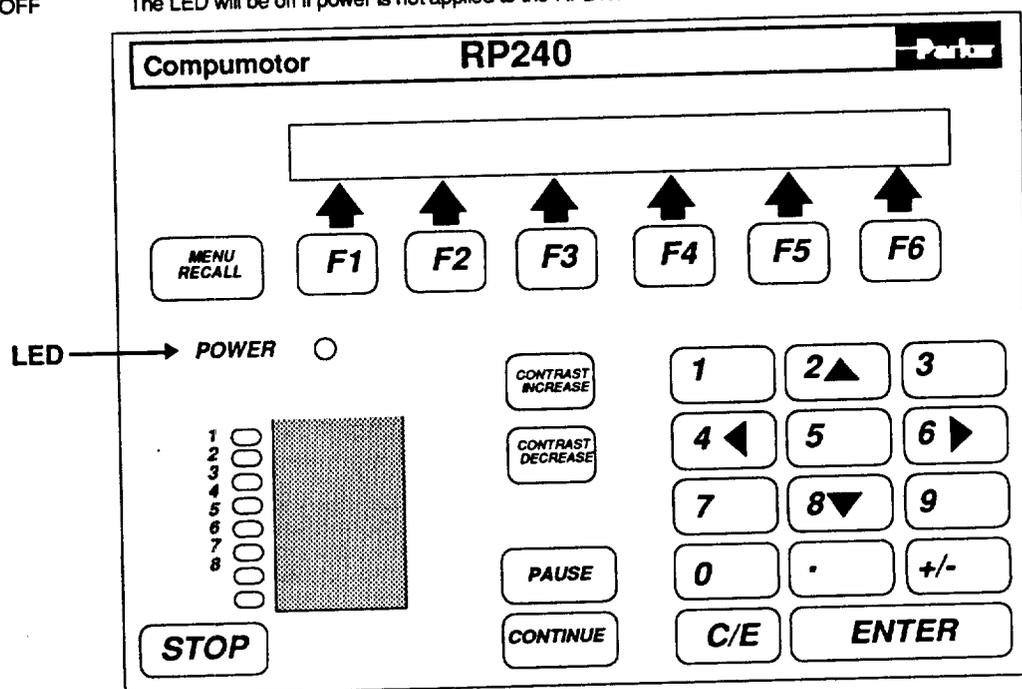
Diagnostic LEDs

RP240 Power LED

The RP240 has one LED indicator that can assist your troubleshooting efforts.

The power LED on the RP240 Main Unit indicates that the internal microprocessor is functioning properly. This two-color LED will be *green*, *red*, or *off*.

GREEN	The LED will be green if power is applied and the RP240 is functioning properly.
RED	The LED will be red if the microprocessor has failed.
OFF	The LED will be off if power is not applied to the RP240.



Basic Troubleshooting Procedure

Answer the following questions when troubleshooting the RP240.

- ① Is the RP240 Status LED on and **green**? If it is off, the RP240 is not receiving the proper DC input voltage. Check the +5VDC power source to make sure it has the proper voltage. The input voltage should be between +5VDC \pm 2%. The RP240 requires at least 100 mA of current to operate reliably. If the voltage is correct and the LED is off, call Compumotor's Applications Engineering Department (800-358-9070).

If Power LED is on and **red**, the internal microprocessor has failed. If possible, document what was happening when it failed (it may be helpful when you attempt to isolate the cause). Cycle power to the RP240. If the LED is still red, there is an internal failure in the unit. If the LED is green, the fault may have been caused by electrical noise.

Common Problems & Solutions

The following table presents some guidelines to help you isolate problems with your motion control system.

- ❑ Look for the symptom that most closely resembles what you are experiencing.
- ❑ Look through the list of possible causes so that you better understand what may be preventing proper operation.
- ❑ Start from the top of the list of remedies and use the suggested procedures to try and isolate the problem.
- ❑ Refer to other sections of the user guide for more information on wiring, software commands, etc.

Extended X

Symptom	Cause	Remedy
STOP key does not work.	STOP key is not enabled.	Enable STOP key with DSTP command.
	Two units daisy chained.	No remedy. STOP key only affects master unit in daisy chain.
When you press the STOP key, your program quits. PAUSE key does not work.	STOP key sends a K command. PAUSE key is not enabled.	This is correct. Enable PAUSE key with DCNT command.
	Two units daisy chained.	No remedy. PAUSE key only affects master unit in daisy chain.
When you press the PAUSE key, your program quits.	Have not enabled Save Command Buffer on Stop (SSH) command.	Enable (SSH1) command.
	Have not enabled Save Command Buffer on Stop (SSH) command.	Enable command (SSH1).
PAUSE key does not pause motion.	Resume Execution (SSL) command has not been enabled.	Enable command (SSL1).
CONTINUE key does not resume program.	Save Command Buffer on Stop (SSH) and Resume Execution (SSL) commands are not enabled.	Enable commands (SSH1, SSL1).
DTEXT command does not display anything on the RP240.	RS-232C connection reversed or not connected.	Verify RS-232C connection.
	A space has been inserted between DTEXT and the text to be displayed.	Remove the space between command and the text.
After running a program with the RP240, cannot communicate with SX, ZX, or 500. Function keys are not being read (No beeping).	SX, ZX, or 500 is still running the program for the RP240. Function keys are not enabled.	Issue a K command, a SSAØ command, and an SSIØ command. Enter a VARn=FUN statement before function key prompt in program.
Numeric keypad is not being read (No beeping).	Numeric keypad is not enabled.	Enter a VARn=NUM statement before keypad entry prompt in program.

6000 Series

Symptom	Cause	Remedy
When you press the STOP key, your program quits.	STOP key sends an IS command.	This is correct. If you would like the stop key to pause motion, refer to the COMEXS command.
PAUSE key does not pause motion.	PAUSE key sends !PS command.	This is correct. If you want to pause motion, use the STOP key in combination with the COMEXS command..
DWRITE command does not display anything on the RP240.	RS-232C connection reversed or not connected.	Verify RS-232C connection
After running a program with the RP240, cannot communicate with 6200 or 6250.	6200 or 6250 is still running the program for the RP240.	Issue an !K command, or press the STOP key.

RS-232C Communications

☞ **Helpful Hint:** If you are having problems communicating with the RP240, use the following procedure to troubleshoot the communications interface.

- ① Make certain the transmit (Tx) of the host is wired to the receive (Rx) of the RP240 and the receive (Rx) of the host (SX, ZX, Model 500, 6200, 6250 or computer) is wired to the transmit (Tx) of the RP240.
- ② Make certain that the ground on the SX, ZX, Model 500, or other device is wired to the ground on the RP240.
- ③ Configure the host to the same baud rate (9600), number of data bits (8), number of stop bits (1), and parity (none) as the RP240.
- ④ Use DC common or signal ground as your reference, *not* earth ground.
- ⑤ Cable lengths should not exceed 50 ft. unless you are using some form of line driver, optical coupler, or shield. As with any control signal, be sure to shield the cable to earth ground at one end only.

Editing Sequences—Extended X

If you wish to edit sequences in your SX, ZX, or Model 500 controller, you must re-establish RS-232C communication *before* downloading the new sequences to the controller. Compumotor recommends using the following procedure.

- ① Remove power from your controller and the RP240.
- ② Remove connections between the controller and the RP240.
- ③ Connect Rx, Tx, and Ground between the controller and your RS-232C port.
- ④ Apply power to your controller.
- ⑤ Enter the TERMINAL EMULATOR portion of your communication software (Compumotor's X-Ware is recommended).

Initially, you will not be able to establish communication if you have turned the echo (SSA1) command off in your power-up sequence.

- ⑥ If you are using X-Ware, **Device Not Ready** will appear on your screen (if you have disabled the command echo). Press the <ENTER> key. Next, press the Y key to enter Terminal Emulator.
- ⑦ To terminate the power-up sequence, and turn on the command echo as well as the interactive mode, issue the following commands:

```
K <cr>  
SSA0 <cr>  
SSI0 <cr>
```

- ⑧ Check the controller's status by issuing an (*R) command, where the device address is 1R <cr>. The controller should respond with a *R. If you do not receive a *R, consult the RS-232C Troubleshooting section of your controller user guide.
- ⑨ Edit the sequence as usual. If you are using X-Ware, press the <ESC> key, then use arrow keys to select the Editor menu.

I N D E X

6000 MODE 45

CABLE LENGTHS 8
COMMANDS

6000 SERIES

DCLEAR 35
DJOG 35
DLED 35
DPASS 36
DPCUR 36
DREAD 36
DREADF 37
DVAR 37
DWRITE 37

EXTENDED X

DCLR 15
DCNT1 15
DLED 16
DPC 16
DSTP 17
DTXT 17
DVO 18
VAR=NUM 19
VARN=FUN 18

CONNECTOR

RP240 10

CONTAMINANTS 5

CURRENT LEVELS 9

DAISY CHAIN 30

POWER-UP SEQUENCE 31

RS-232C 30

DAISY CHAINING 30

ELECTRICAL NOISE 5

EXTENDED X LANGUAGE 14

EXTENDED X MODE

DISPLAY TEXT STRING COMMAND 44

POSITION CURSOR COMMAND 44

UPDATE LEDS COMMAND 44

INSTALLATION

VERIFICATION 10

MICROPROCESSOR 43

MODES OF OPERATION 43

6000 MODE 43

EXTENDED X MODE 43

COMMANDS 44

MOUNTING

DIMENSIONS 7

MOUNTING HOLES 7

OPERATOR PANEL 6

RP240-NEMA4 7

NEMA RATINGS

NEMA 12 5

NEMA 4 5

POWER SUPPLY 9, 47

PROGRAMMING

6000 SERIES

FUNCTION KEYS 40

JOG MODE 41

LCD DISPLAY 38

LEDS 38

PASSWORD 41

READ RP240 DATA 39

EXTENDED X

APPLICATION PROGRAM 24

EDIT SEQUENCES 29

FUNCTION KEYS 22

LCD DISPLAY 20

LEDS 20

NUMERIC KEYPAD 21

PAUSE AND CONTINUE KEYS 23

POWER-UP SEQUENCE 24

STOP KEY 23

PROGRAMMING SEQUENCES 11

RP240

COMMAND SET 43

RS-232C 43

BAUD RATE 47

DATA BITS 47

PARITY 47

STOP BIT 47

WIRING

20 AWG WIRE 8

6000 SERIES 9

6200 9

6250 9

BACK PANEL 4

CABLE LENGTHS 8

CONNECTOR 4

EXTENDED X 8

GROUNDING 8

MODEL 500 8

POWER 4

SX 8

ZX 8

