PACIFIC SCIENTIFIC
HIGH PERFORMANCE MOTORS & DRIVES

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

Part# MA950
List Price $30 U.S.
February, 1999
Rev E

M A 9 5 0

OC950 Installation & Hardware Reference Manual

WARRANTY AND LIMITATION OF LIABILITY

Includes software provided by Pacific Scientific

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

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

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

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

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

Table of Contents

1 Overview of the OC950 Programmable Option Card 1-1

1.1 OC950 Definition 1-1
1.2 How to Use this Manual 1-2
1.3 Warranty Information 1-3

2 Getting Started 2-1

2.1 Setting Up the Hardware 2-1
2.2 Installing 950IDE 2-4
2.3 Starting 950IDE 2-4
2.4 Getting Around in 950IDE 2-5
2.5 Configuring your System 2-6
2.6 Creating a New Program 2-7
2.6.1 Using Wizards 2-8
2.7 Compiling a Program 2-11
2.8 Run Program 2-11
2.9 Working with Breakpoints 2-12
2.10 Using the Variables Window 2-13
2.11 Viewing Variables 2-14
2.12 Saving a Program 2-16
2.13 Opening a Program 2-16
2.14 Exiting 950IDE 2-16

3 OC950 Interfaces and Connections 3-1

3.1 Serial Port J51 3-1
3.2 Setting up Serial Address using Switch S1 3-7
3.3 OCIO (Option Card I O) J52 3-14
3.4 Connecting to the Opto22 Mounting Rack 3-19
3.5 PacLAN Cabling & Hardware Connections - J53 3-24

4 Servo Loop Parameters 4-1

4.1 Velocity Loop 4-1
4.2 Position Loop 4-8
4.3 Advanced Velocity Loop Tuning 4-10

5 Maintaining/Troubleshooting 5-1

5.1 Maintaining the SC950 Servocontroller 5-1
5.2 Troubleshooting the SC950 Series Servocontroller 5-1

Appendix A Selecting Motor Control Functionality A-1

Appendix B OC950 Specifications B-1

Index

1 Overview of the OC950 Programmable Option Card

Introduction

This chapter introduces the OC950 Programmable Option Card. Topics covered are:

- OC950 Overview
- How to Use this Manual
- Warranty Information

1.1 OC950 Programmable Option Card Definition

OC950 overview

The OC950 is an option card that is installed in the option card slot of a Pacific Scientific SC900 Series Servo Drive. Combined, the SC900 and OC950 create the SC950. It provides stand-alone, single-axis programmable positioning capability to a high performance, digital servo drive.

The SC950 inherits all the advanced, high-performance functionality as well as the small package size of the SC900 platform. The SC950 also inherits all the hardware resources of that platform, including 6 points of bi-directional discrete I/O, two analog outputs, one analog input, one encoder input port and one encoder output port. The SC900 platform also provides two high speed registration inputs.

Resources provided on all varieties the OC950 include twenty-one points of bi-directional discrete I/O and an RS-232/RS-485 serial communications port.

The OC950 provides for direct connection to Opto22 Industrial I/O mounting racks and industrial I/O modules with the use of the CA950-IO adapter module.

OC950 Hardware Varieties

Two optional memory (NVRAM: nonvolatile random access memory) configurations are available, 32kx8 or 128kx8. NVRAM is battery backed, with a ten year life, and unlimited amount of write cycles. The memory provides storage for the user programs.

A PacLAN™ Network interface is offered as an OC950 option. PacLAN supports 2.5 Mb/sec communications between a cluster of SC950 drives. The port will be compatible with standard ArcNET products using 93 Ω coax cable in a bus configuration.

Standard version

Combinations of the two memory offerings and the PacLAN option results in four standard versions of the OC950:

OC950-501-01	Base 950 Option Card 32kx8 NVRAM
OC950-502-01	Base 950 Option Card 128kx8 NVRAM
OC950-503-01	Base 950 Option Card 32kx8 NVRAM + PacLAN
OC950-504-01	Base 950 Option Card 128kx8 NVRAM + PacLAN

Enhanced Version

In addition, there is an Enhanced version of the OC950 which supports additional features, including MODBUS and camming.

OC950-601-01	Enhanced 950 Option Card 32kx8 NVRAM
OC950-602-01	Enhanced 950 Option Card 128kx8 NVRAM
OC950-603-01	Enhanced 950 Option Card 32kx8 NVRAM + PacLAN
OC950-604-01	Enhanced 950 Option Card 128kx8 NVRAM + PacLAN

1.2 How to use this manual

Chapter 2, “Getting Started”, is a step-by-step guide allowing you to configure an SC950 and run your motor within a few minutes. It is strongly recommended that you go through Chapter 2 first. This will give you a feel for using the SC950 and lay the framework for reading the other chapters. Chapters 3 through 5 should be read thoroughly to gain the most from the OC950.

1.3 Warranty

The Pacific Scientific OC950 has a two year warranty against defects in material and assembly. Products that have been modified by the customer, physically mishandled, or otherwise abused through miswiring, and so on, are exempt from the warranty plan.

Warning

If the continuous current rating of the drive is greater than the continuous current rating of the motor that it is being used with, then it is possible to cause significant damage to the motor. Pacific Scientific may not honor the warranty of the motor if it is run under these conditions.

2 Getting Started

Introduction

This chapter provides a step-by-step introduction to creating a new program. This procedure uses the minimum possible equipment to run an unloaded motor and sets motor speed from a PC's serial port. It is strongly recommended that all first time users go through this procedure to become familiar with the OC950 and the 950IDE PC interface software before installing the servo system in a machine.

2.1 Setting Up the Hardware

What you will need

To go through this product introduction procedure you will need the following items.

- SC900 Base Servo Drive
- OC950 Programmable Option Card
- Appropriate Brushless Motor with nothing attached to the shaft
- PC Running Windows 3.1 or Windows95/NT
- 950IDE Floppy Disk
- Motor Power and Feedback Cables (J2, J3)
- RS-232 Communications Cable (J51)
- DB-25 Connector Mate (J4)
- AC Power Line (J1)

If your OC950 is not already installed in your SC900, then use the following instructions to install it.

CAUTION

NEVER insert or remove an Option Card with the Control AC Power (J1-5,6) active. Damage to the base SC900 or the Option Card could occur.

Procedure

1. Remove Control AC Power from the SC900. The system status LED should be blank.
2. Loosen the two locking screws counterclockwise on the existing face plate or existing Option Card and remove.
3. Position the new Option Card so that the silk screen reads the same as the base SC900.
4. Insert the Option Card by sliding it in all the way until it is flush with the base SC900.
5. Tighten the two locking screws by turning clockwise.

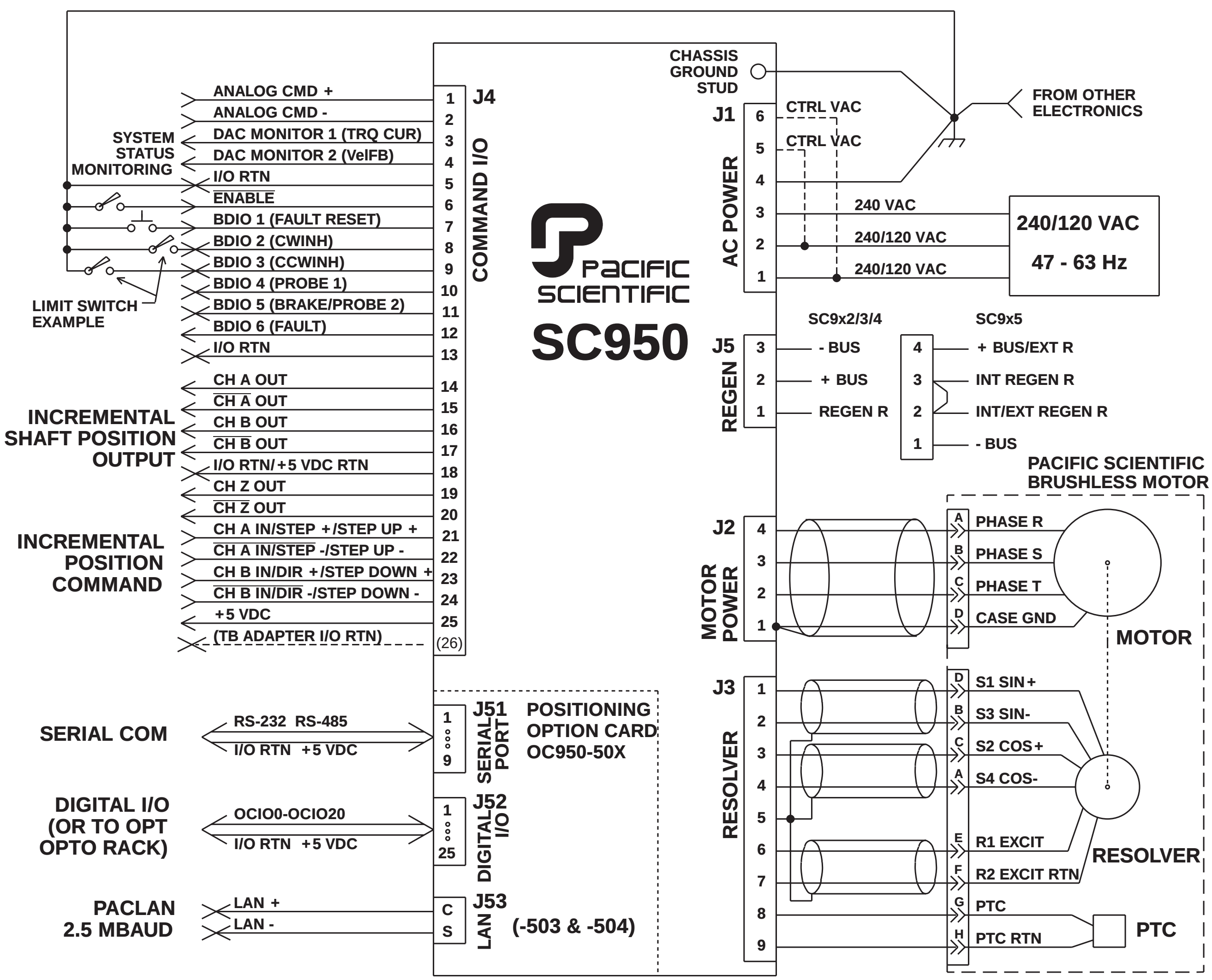
Wiring connections

Connect the motor, feedback, and AC Power cables as shown in the following Connection Diagram but **do not apply the AC Power at this time**. It is recommended that Pacific Scientific motor and feedback cables be used during setup since improper cabling is the number one cause of start up problems.

The RS-232 cable made by Pacific Scientific (order number CS-232-750) can be used to connect the 9 pin serial port socket on the OC950 to the PC. If this cable is unavailable, a simple 3 wire cable can be made using the wiring diagram shown on page 3-11.

The last connection needed is to provide the hardware enable to the SC900 via J4-6 and I/O RTN on J4-5. Preferably connect a toggle switch between J4-6 and J4-5. If a toggle switch is not available a clip lead that can connect or not connect J4-6 to J4-5 will do.

Connection diagram



2.2 Installing 950IDE

Procedure

To install 950IDE, perform the following:

1. Insert the 950IDE diskette in your disk drive (A: or B:). Start Windows and choose **Run** from the **File Menu** of Program Manager. At the Command Line, type **A:\ setup950** (or B:\ setup950) and **press the enter key (↵)** or **click on OK.**
2. Windows will then ask you which drive (default is C) and directory (default is \950Win) you want 950IDE installed in.
3. Windows will then ask you to select a Program Manager Group (default is Pacific Scientific) that you would like the icons placed in.

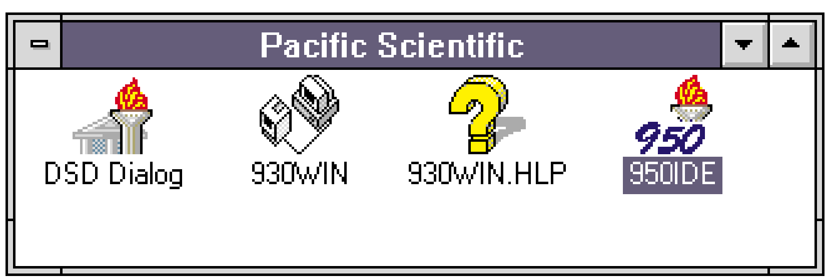
Note: *When finished, the 950IDE disk should be removed from the drive and stored in a safe place.*

2.3 Starting 950IDE

Procedure

To begin using 950IDE, perform the following:

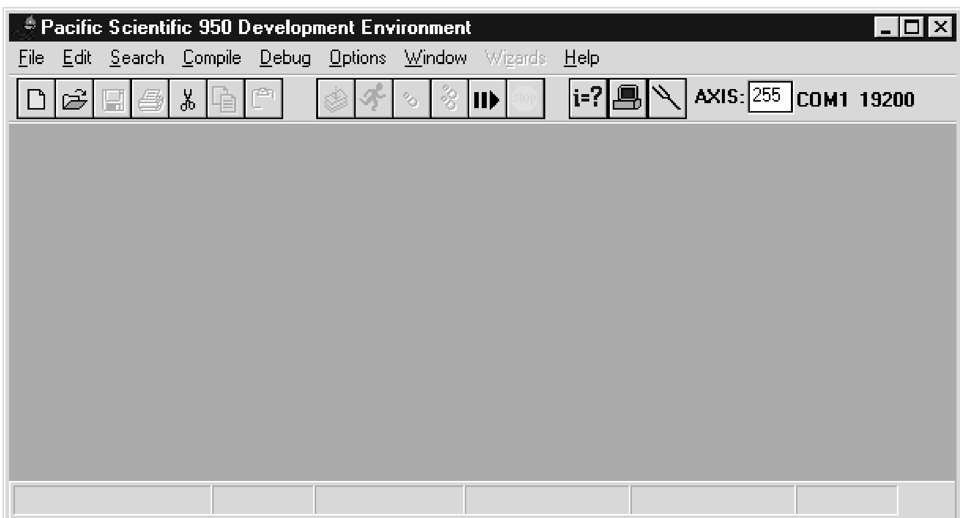
1. If not already open, open the **Pacific Scientific Group** in Program Manager. Double click on the 950 icon.



2.4 Getting Around in 950IDE

950IDE main menu

Once you double-click on the 950 icon, the following window will appear:



Movement keys

950IDE is a standard Windows application and the normal cursor movement keys operate the same way as in all windows applications.

- **<F1>** gives context sensitive on-line help

2.5 Configuring Your System

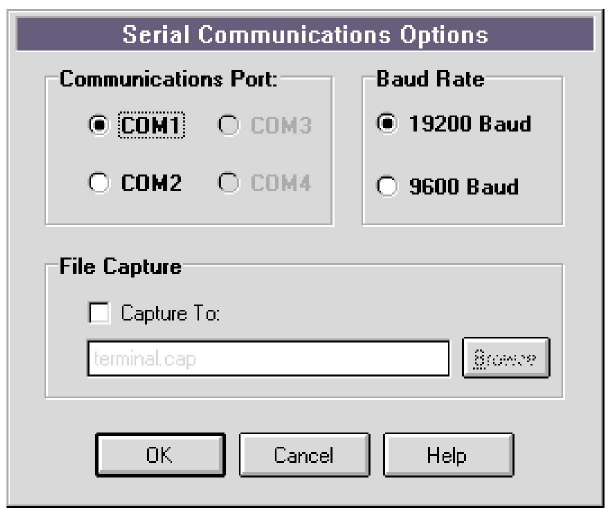
Applying AC Power

Carefully check all wiring connections and ensure that J4-6 is not connected to J4-5. Apply AC power to your controller. The drive status display LED should be alternately flashing U C (for unconfigured) after the power up message.

Serial Port

To specify the PC serial port that is connected to the OC950:

1. Select **Communications** from the **Options** Menu and the following dialogue box will appear:

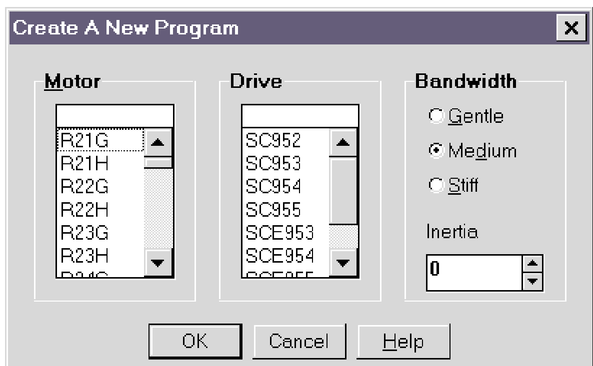


2. Specify the serial port that you want to use.
3. Specify the baud rate that you want to use and click on **OK**.
4. Select **Variables** from the **Compile** Menu. Type **BaudRate** in the Variable/expression box and press ↵.
5. If the current value is incorrect, <**Tab**> to the New Value box and enter the desired baud rate.

2.6 Creating a New Program

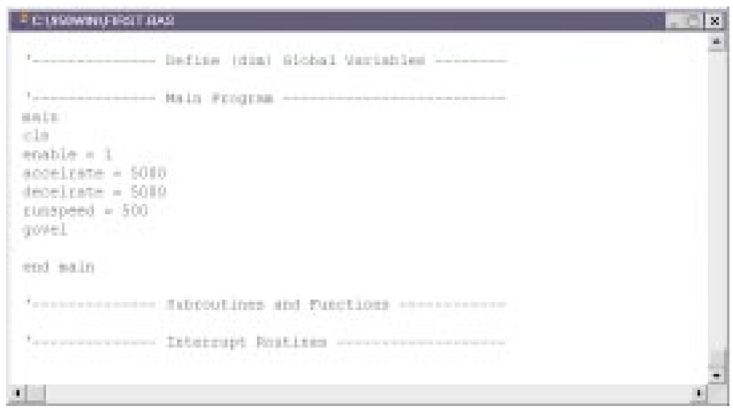
Procedure

Select **New** from the **File** menu and the following dialog box will appear:



- Select the appropriate Motor Part Number from the dropdown menu.
- Select the Drive you're using (Example: SC953)
- Select Medium
- Select an inertia ratio and click on OK.
- Select a filename (with a *.BAS extension) for your new program, click on OK, and the following window will appear:

You may now begin typing in your program. To create a simple jog move, type in the following lines:



Note: *The drive status display should now show a steady 0 for configured and not enabled.*

The current loop has been properly compensated for the selected motor, and the servo parameters have been set to a medium response (approximately 75 Hz velocity loop bandwidth) for the unloaded motor. Additional default settings have also been implemented.

Note: *For additional tuning, click on the on-line tuning button in the toolbar and follow the on-line help.*

2.6.1 Using Wizards

Introduction

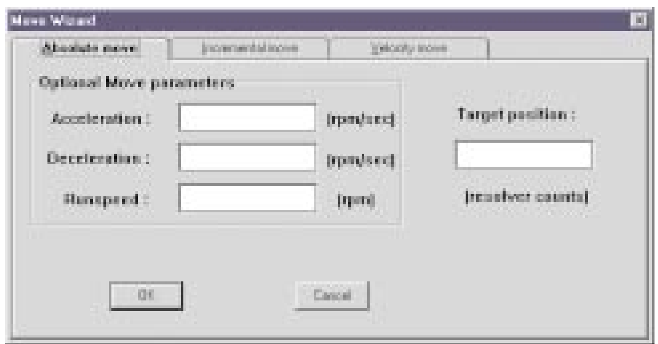
The following wizards have been added to the 950IDE to facilitate creating programs:

- Select Case
- Modbus and A-B DF1 Map
- Interrupt Handler
- Move Command
- Cam Setup

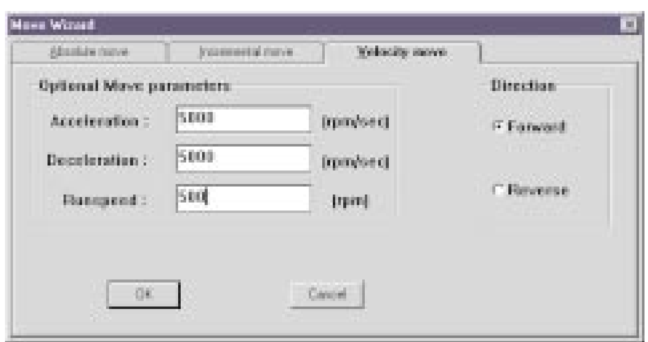
Procedure

To use a template to create a jog move, perform the following:

1. Select **Move Command** from the **Wizards** Menu and the following window will appear:

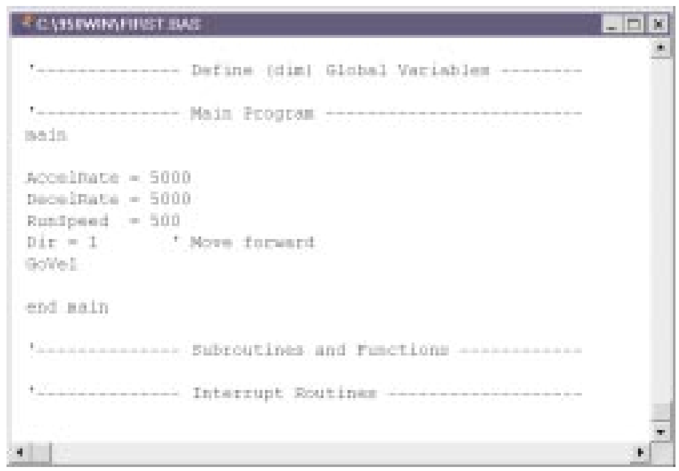


2. Select the Velocity Move tab and enter the appropriate parameters.



3. Click on **OK**.

The text will be inserted in the main program, below the cursor.



Enabling Drive

The controller can be enabled at this time by closing the switch between the $\overline{\text{Enable}}$ input (J4-6) and I/O RTN (J4-5). The commanded motor speed will be the power up default, set to 0 during configuration.

WARNING

Before proceeding, the motor may need temporarily clamped down to prevent inertial forces from displacing the motor.

2.7 Compiling a Program

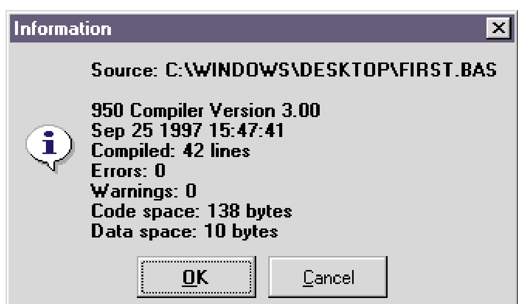
The SC950 requires a compiled program to be downloaded via serial communications. To compile a program:

Procedure

1. Select **Compile** from the **Compile** menu, press <**Ctrl - F5**> or click on the Run icon.

   Any error detected during compilation will abort further compilation (after the error is encountered), display the compilation status, and return to the program highlighting the program line containing the error.

2. Make necessary corrections until program is compiled successfully, and the Compiler Status indicates zero errors.



2.8 Run Program

To run the current program:

Procedure

1. Select **Run** from the **Compile** Menu, press <**F5**> or click on the Compile icon.

2.9 Working with Breakpoints

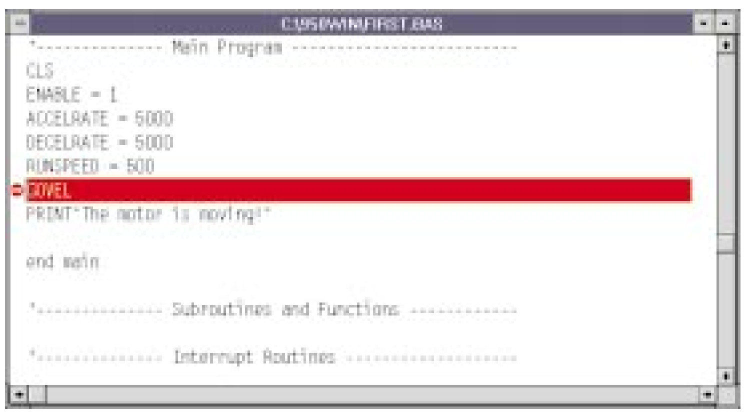
Introduction

Breakpoints allow you to halt program execution at a specific location. 950IDE can set up to nine breakpoints within a program. These should be set prior to running the program.

Setting breakpoints

To set breakpoints:

1. Place the cursor on the line of code in which you want a breakpoint set.
2. Select **Set Breakpoint** from the **Debug** menu or press <**F9**>.



Removing breakpoints

To remove a breakpoint:

1. Place the cursor on the line of code which contains a breakpoint that you want to clear.
2. Select **Clear Breakpoint** from the **Debug** menu or press <**F9**>.
3. To remove all breakpoints in a program, select **Clear All Breakpoints** from the **Debug** menu.

Continuing program execution

When a breakpoint is encountered during program execution, the SC950 will suspend program execution and await further instruction. After a breakpoint has been encountered, program execution can be continued by:

1. Selecting **Continue** from the **Compile** menu to continue program execution until the next breakpoint. **OR**

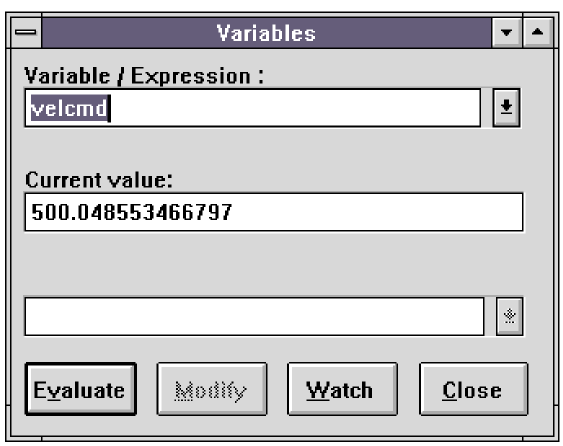
2. Selecting **Single Step** from the **Debug** menu, pressing <**F8**> to single step through the program line by line, or by clicking on the Single Step icon.

2.10 Using the Variables Window

Monitoring Motor Velocity

Select **Variables** in the **Compile** Menu. The Variables window allows all parameters, variables, and commands to be examined, changed, or actuated as appropriate.

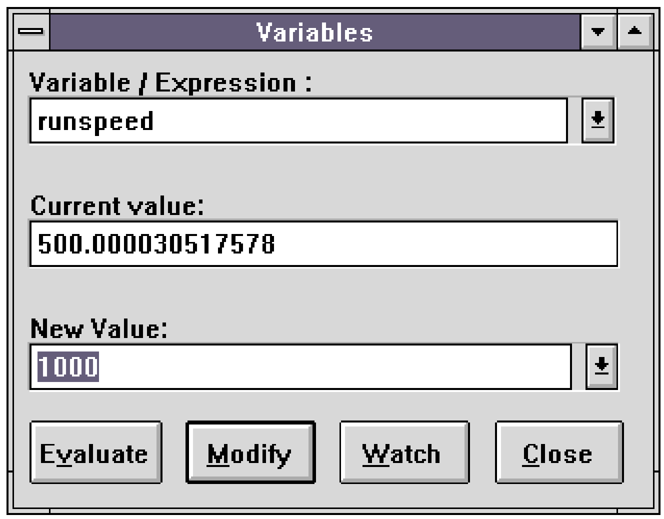
Type the variable name in the Variable/expression box. To examine the shaft velocity, type **velcmd** in the Variable/Expression box and press ↵. The present value of the motor velocity command in the drive will be displayed under Current Value.



To continuously read and update the measured velocity press the Evaluate button via the mouse or by using the <**Tab**> key to move the focus to that button.

Modifying variables

To modify the value of the runspeed, type RUNSPEED in the Variable/expression box and press ↵ .The current value of runspeed is displayed. To change the value <**Tab**> to the New Value box and enter 1000 and click on the Modify button.



Type GOVEL in the Variable/expression box and press ↵. The motor should now run at the new speed.

2.11 Viewing Variables

Introduction

Multiple variables can be monitored using a Watch window. Variables displayed in the watch window are updated as breakpoints are encountered or after each single step.

Opening the watch window

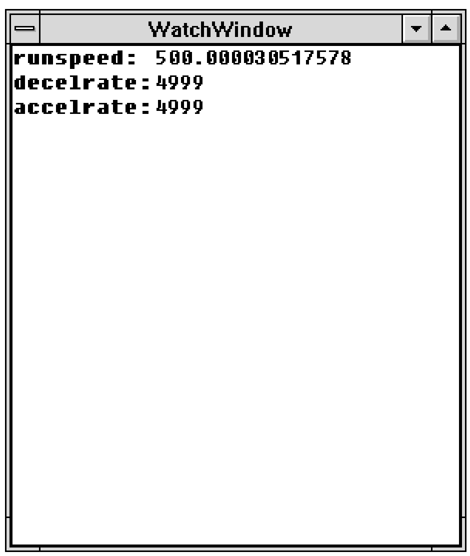
To open a Watch window:

1. Select **Watch Window** from the **Debug** menu.

Adding variables to the watch window

You can add variables to the watch window while a program is not executing.

1. To access the Variables window, select **Variables** from the **Compile** menu.
2. Type the name of the variable you which to watch in the Variable/Expression box and then click on the Watch button. The Variable is added to the Watch window.



Removing variables from Watch Window

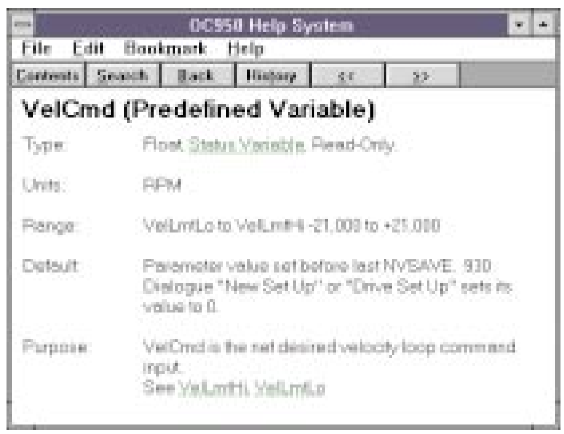
You can remove any variable that has been added to the Watch window. This can be done at any time.

To remove a variable from the Watch window:

1. Select the variable in the Watch window and press <**delete**>.

Getting Help

To get help information on a particular key word press the <**F1**> key while the cursor is located somewhere on that word in the Variable/Expression box. With VelCmd in that box <**F1**> should bring up the following help window.



2.12 Saving a Program

Select **Save As** from the **File** menu. Type the File Name **FIRST.BAS** and click on <**OK**>. The program will be saved on disk in a file named FIRST.BAS.

2.13 Opening a Program

Select **Open** from the **File** menu. Select **FIRST.BAS** from the list of file names and click on <**OK**>. The program FIRST.BAS, that you just saved to disk, will be read from the disk and displayed.

2.14 Exiting 950IDE

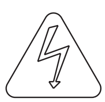
To exit 950IDE, select **Exit** from the **File** menu.

You should now know how to start and exit 950IDE, create a program, and monitor and change variable values.

3 OC950 Interfaces and Connections

Introduction

This chapter describes the OC950 interfaces and required connection information. The OC950 plugs into an SC900 servocontroller.



Warning!

Installation of the OC950 should be made with the servocontroller power off!

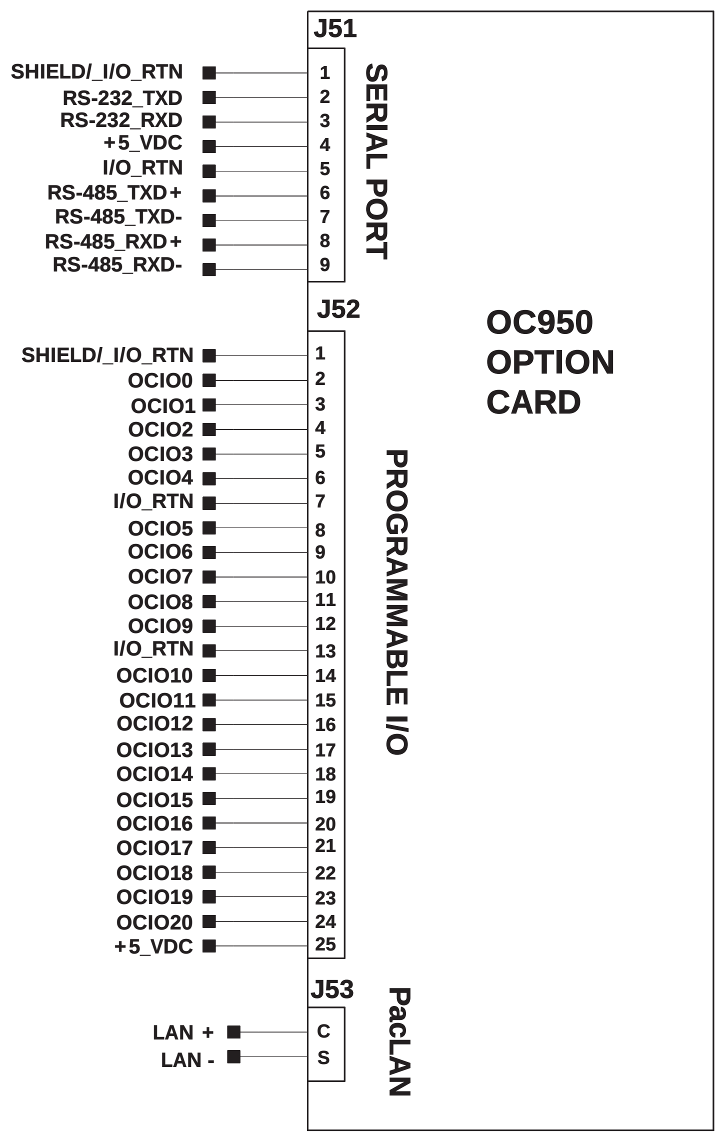
Two lock screws, located on the front panel of the OC950, should be turned until resistance is met, in order to properly seat the OC950 into the base unit. A communications address selection switch resides on the OC950 printed circuit board, accessible only when removed from the SC900 servo base unit.

The OC950 external connections include:

- Serial communications port (J51)
- Bi-directional I/O connections, OCIO0-OCIO20 (J52)
- Optional PacLAN™ communications port (J53)

Connection diagram

A connection diagram of the OC950 is shown below.



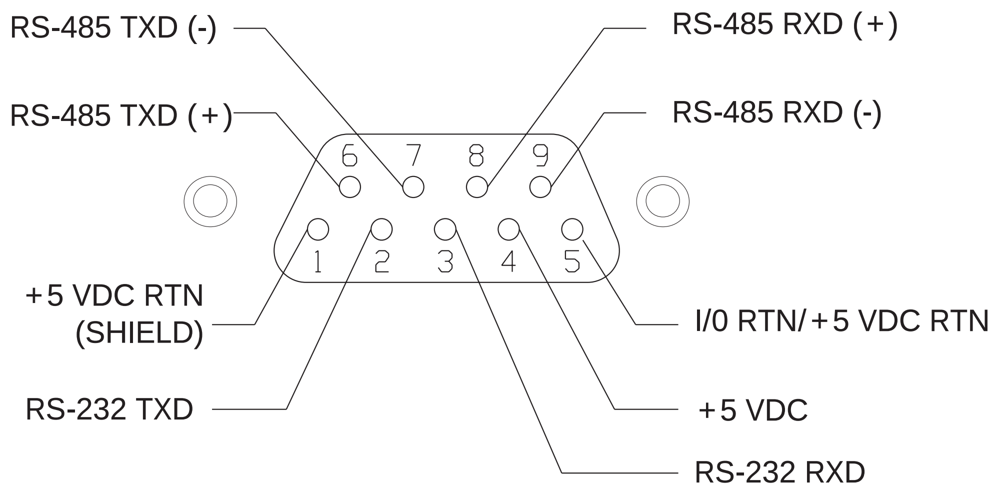
3.1 Serial Port J51

The serial port (J51), utilizes the 9 contact female D subminiature style connector shown below. A brief description of each signal is included in the J51 I/O table on following page. For additional information, please refer to the OC950 Serial Communications Transceiver Schematic at the end of this section.

I/O Table

Input/Output	Pin Number	Explanation
+5 VDC RTN/ Shield	J51-1	Common/shield
RS-232 TXD	J51-2	RS-232 transmitter output (from OC950)
RS-232 RXD	J51-3	RS-232 receiver input (to OC950)
+5 VDC	J51-4	+5 Vdc output (200 mA maximum between J51-4, J52-25 & J4-25)
I/O RTN/+5 VDC RTN	J51-5	Common
RS-485 TXD (+)	J51-6	RS-485 transmitter output (from OC950)
RS-485 TXD (-)	J51-7	
RS-485 RXD (+)	J51-8	RS-485 receiver input (to OC950)
RS-485 RXD (-)	J51-9	

J51 diagram



The information provided in this section should be used to connect the SC950 to your computer for use with the 950 Integrated Development Environment (950IDE) (due to the intelligent communications protocol utilized, it is not possible to configure or program the OC950 with a dumb terminal). Two communication links are available, RS-232 and RS-485. RS-485 allows a single computer to communicate with up to 32 SC950s in multi-axis configurations. A DIP switch on the OC950 selects the communications address for RS-485 communication. The 950IDE defaults to communicate with axis 255 upon start up.

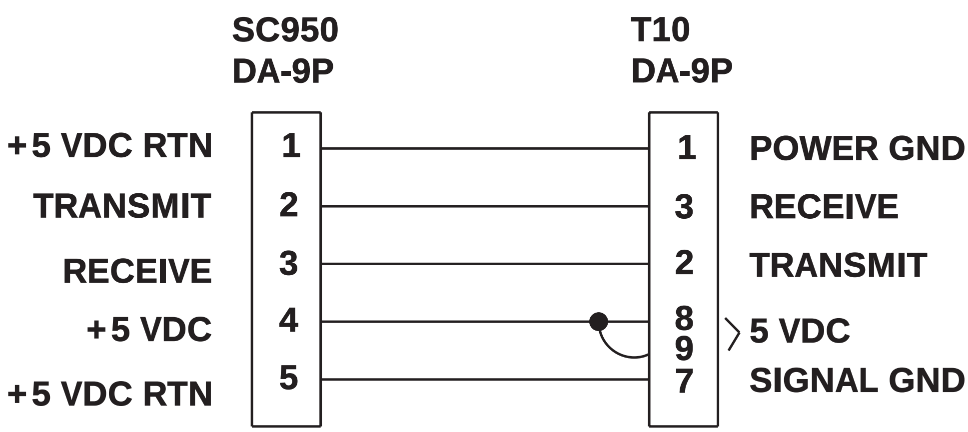
J51 Serial Port T10 Connections (RS-232)

Introduction

Connect the SC950 to your T10 terminal as follows:

The T10 terminal requires a DC-9P 9-pin plug-in male D connector.

Note: *The T10 terminal is a Burr-Brown TM2500.*



Procedure

1. Solder the cable leads to the SC950 mating connector and the T10 connector as shown.
2. Assemble the connector housings.
3. Plug the connectors into the SC950 and T10 and affix the connectors to the units.

Parameter setup

Set up the T10 terminal as follows:

1. Enter the setup mode by pressing the "." key while power is applied to the T10 (wait until the two letter prompt appears). Refer to the terminal manual for more information.
2. Enter the following values:

VW = 4	Default	LE = 0	Local Echo OFF
TM = 0	Character Mode	EN = 1	Line Terminator (CR)
TD = 0	Turnaround Delay	KC = 2	Key Click
BR = 2	Baud Rate (9600)	KR = 1	Key Repeat
DF = 4	8 Bits, No Parity	CU = 2	Cursor
HS = 0	Default (DTR asserted)		

Note: *The SC950 must be configured for 9600 baud (*`BaudRate`*= 9600), 8 bit, no parity data formats. Please see the MA950-LR Reference Manual for additional information on configuring baud rate.*

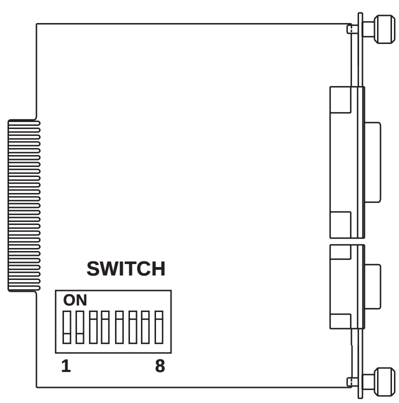
3.2 Setting Up Serial Addresses using Switch S1

Definition

The S1 switch sets the communication address for each OC950. The ability to select different addresses is used with RS-485 for multi-drop communications.

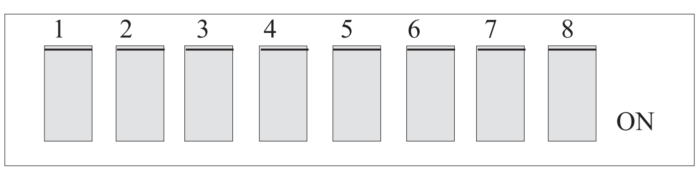
Procedure

Looking down at the top of the OC950, the following diagram shows the location of switch S1.



Note: Each *SC950 subsystem connected to an RS-485 multi-drop installation or PacLAN must have a unique serial address.*

The diagram below shows the S1 default switch settings. The OC950 factory default address is 255 (All switches off).



The switches are:

- On in the up position (away from number)
- Off in the down position (toward number)

Note: *When using RS-232 communications, it is recommended to leave the address set at 255.*

S1 Address table

Address	1	2	3	4	5	6	7	8
0	On	On	On	On	On	On	On	On
1	Off	On	On	On	On	On	On	On
2	On	Off	On	On	On	On	On	On
3	Off	Off	On	On	On	On	On	On
4	On	On	Off	On	On	On	On	On
5	Off	On	Off	On	On	On	On	On
6	On	Off	Off	On	On	On	On	On
7	Off	Off	Off	On	On	On	On	On
8	On	On	On	Off	On	On	On	On
9	Off	On	On	Off	On	On	On	On
10	On	Off	On	Off	On	On	On	On
11	Off	Off	On	Off	On	On	On	On
12	On	On	Off	Off	On	On	On	On
13	Off	On	Off	Off	On	On	On	On
14	On	Off	Off	Off	On	On	On	On
15	Off	Off	Off	Off	On	On	On	On
16	On	On	On	On	Off	On	On	On
17	Off	On	On	On	Off	On	On	On
18	On	Off	On	On	Off	On	On	On
19	Off	Off	On	On	Off	On	On	On
20	On	On	Off	On	Off	On	On	On

S1 Address table (cont'd)

Address	1	2	3	4	5	6	7	8
21	Off	On	Off	On	Off	On	On	On
22	On	Off	Off	On	Off	On	On	On
23	Off	Off	Off	On	Off	On	On	On
24	On	On	On	Off	Off	On	On	On
25	Off	On	On	Off	Off	On	On	On
26	On	Off	On	Off	Off	On	On	On
27	Off	Off	On	Off	Off	On	On	On
28	On	On	Off	Off	Off	On	On	On
29	Off	On	Off	Off	Off	On	On	On
30	On	Off	Off	Off	Off	On	On	On
31	Off	Off	Off	Off	Off	On	On	On
32	On	On	On	On	On	Off	On	On
33	Off	On	On	On	On	Off	On	On
34	On	Off	On	On	On	Off	On	On
35	Off	Off	On	On	On	Off	On	On
36	On	On	Off	On	On	Off	On	On
37	Off	On	Off	On	On	Off	On	On
38	On	Off	Off	On	On	Off	On	On
39	Off	Off	Off	On	On	Off	On	On
40	On	On	On	Off	On	Off	On	On

.

.

.

S1 Address table (cont'd)

Address	1	2	3	4	5	6	7	8
240	On	On	On	On	Off	Off	Off	Off
241	Off	On	On	On	Off	Off	Off	Off
242	On	Off	On	On	Off	Off	Off	Off
243	Off	Off	On	On	Off	Off	Off	Off
244	On	On	Off	On	Off	Off	Off	Off
245	Off	On	Off	On	Off	Off	Off	Off
246	On	Off	Off	On	Off	Off	Off	Off
247	Off	Off	Off	On	Off	Off	Off	Off
248	On	On	On	Off	Off	Off	Off	Off
249	Off	On	On	Off	Off	Off	Off	Off
250	On	Off	On	Off	Off	Off	Off	Off
251	Off	Off	On	Off	Off	Off	Off	Off
252	On	On	Off	Off	Off	Off	Off	Off
253	Off	On	Off	Off	Off	Off	Off	Off
254	On	Off	Off	Off	Off	Off	Off	Off
255*	Off	Off	Off	Off	Off	Off	Off	Off

* For RS-232 operation (factory default)

Procedure

1. Remove power from the SC950 servocontroller.
2. Refer to the table above to set the appropriate address.
3. Reconnect power to the SC950 servocontroller.
4. Repeat steps 1 through 4 for other SC950 units on the bus. Make sure to give the other units unique addresses.

OCIO +5VDC

A +5V DC supply resides on the SC900. This supply is connected to J51-4, as well as J52-25 (OCIO port), and J4-25 (command port on base unit). The **total** current draw from **all** connections must not exceed 200 milliamperes.

RS-232 Connections

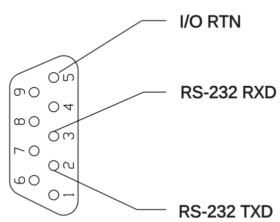
RS-232 connections on J51 are shown below. Cable wiring required for connecting to either 9 or 25 pin serial ports of most computers are also shown.

Note: *Pinouts vary among computer manufacturers. Check the hardware reference manual for your machine before wiring.*

Cabling diagram

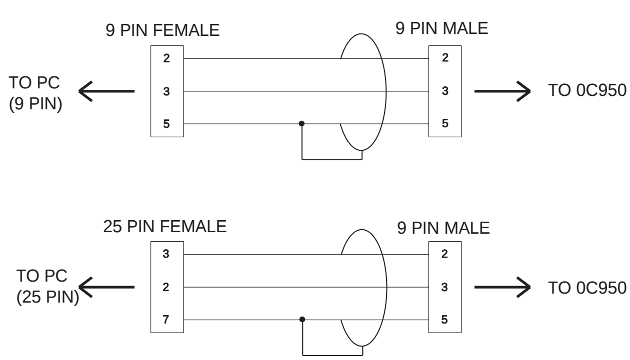
A 6 foot (1.8 m) RS-232 Cable with 9 pin connectors and a 9 pin to 25 pin adapter is available from Pacific Scientific. The Pacific Scientific order number is RS-232-750.

Note: *Shielded wiring is recommended for the serial communications cable to minimize potential errors from electrical noise.*



9 CONTACT FEMALE D SUBMINIATURE CONNECTOR

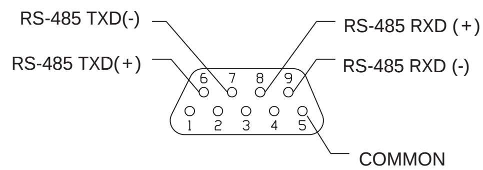
TYPICAL CABLES

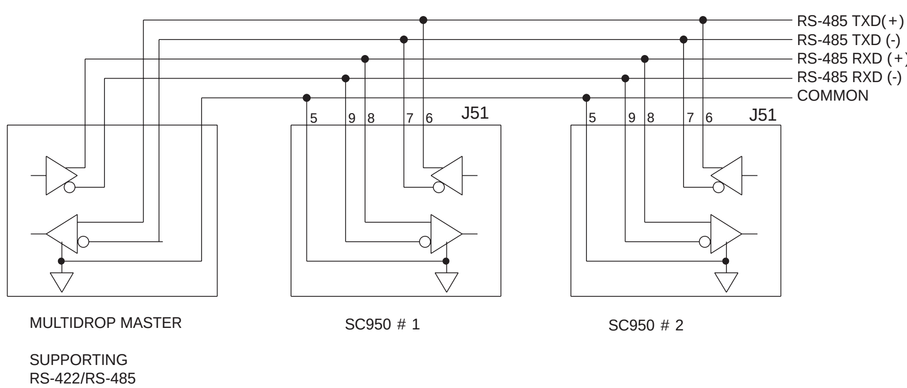


RS-485/RS-422 Connections

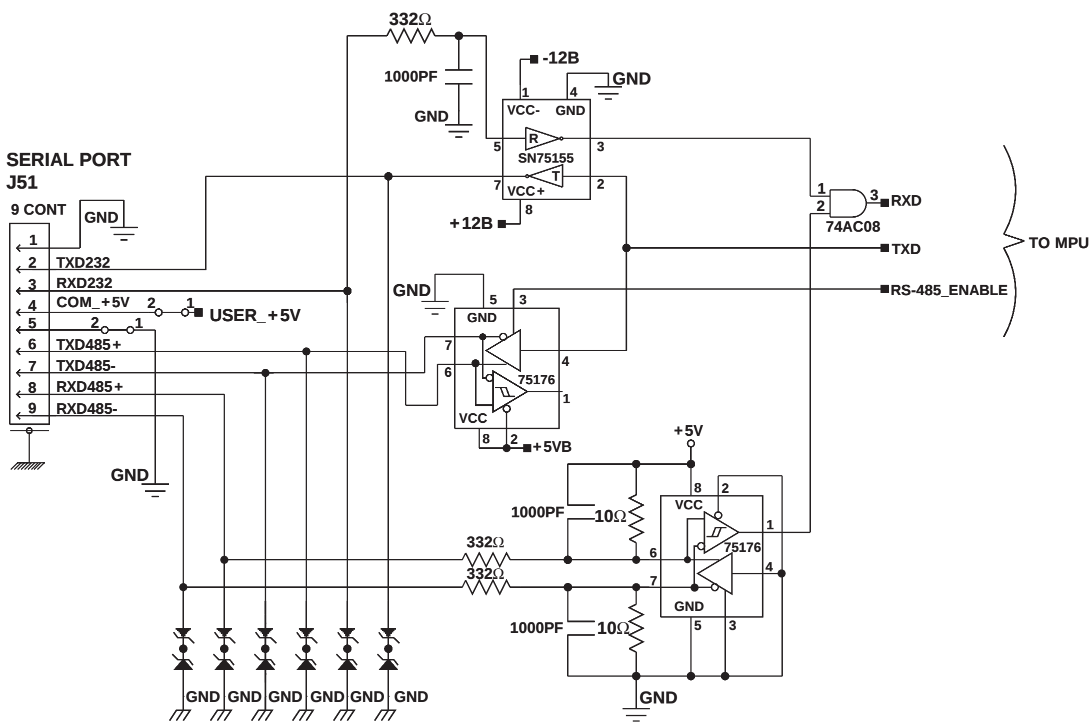
Up to 31 SC950s can be connected in parallel to a multidrop master. The SC950s must each have a unique address, set using DIP switch S1 as described in Section 3.2. RS-485/RS-422 connections to J51 are shown below. A multidrop interconnection diagram, showing multiple axes connected to a single host is also included.

RS-485/RS-422 Diagram





OC950 Serial Communications Transceiver Schematic



3.3 OCIO (Option Card I O) J52

The OC950 provides twenty-one channels of non-optically isolated I/O. J52 utilizes a standard 25 contact female D subminiature (DB25S) style connector. Each channel is bi-directional, and can be configured via software control as an input point or as an output point. The user program has complete access to monitor input or to set output channels.

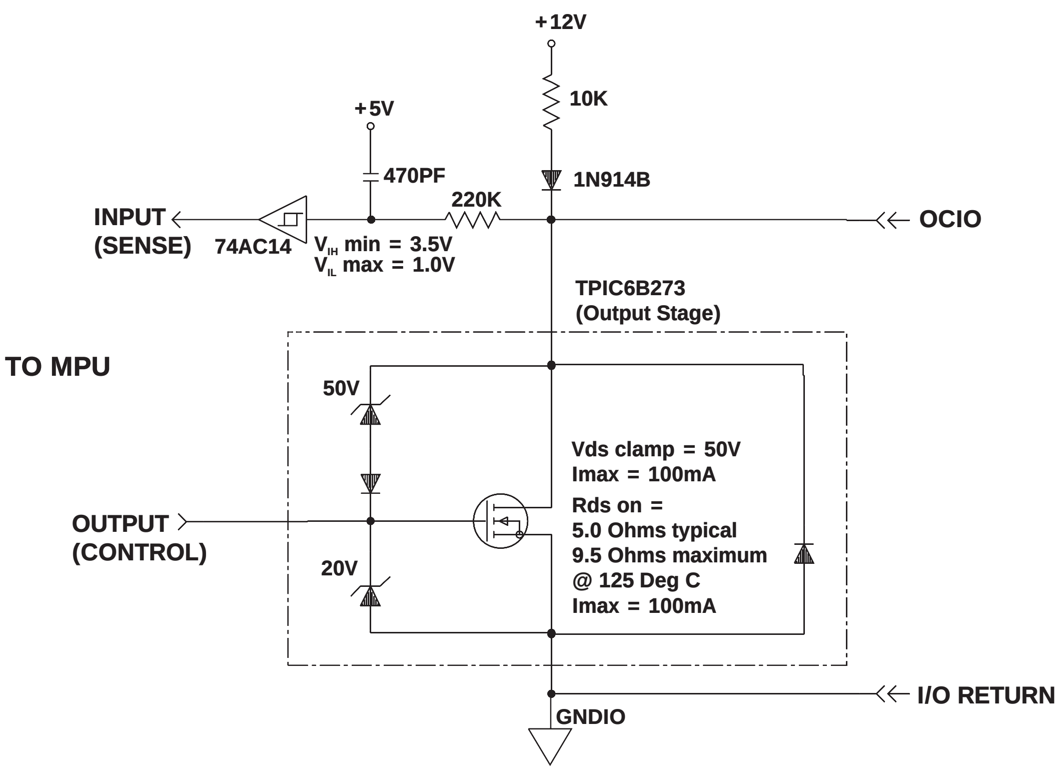
J52 I/O Table

The pinout of J52 is described in the following table:

Input/Output	**J52 Pin Number**	**Input/Output**	**J52 Pin Number**
I/O RTN	1	OCIO10	14
OCIO0	2	OCIO11	15
OCIO1	3	OCIO12	16
OCIO2	4	OCIO13	17
OCIO3	5	OCIO14	18
OCIO4	6	OCIO15	19
I/O RTN	7	OCIO16	20
OCIO5	8	OCIO17	21
OCIO6	9	OCIO18	22
OCIO7	10	OCIO19	23
OCIO8	11	OCIO20	24
OCIO9	12	+5 VDC	25
I/O RTN	13		

OC950 Bi-Directional I/O Channel

A schematic diagram of a single OCIO channel is shown below.



OCIO Inputs

Each channel has a 10KΩ pull-up resistor to +12VDC, connected in series with a current limiting diode. Inputs are sensed with a 74AC14 CMOS gate. Input logic thresholds are:

V_{IH} min:	3.5 V
V_{IL} max:	1.0 V

OCIO Outputs

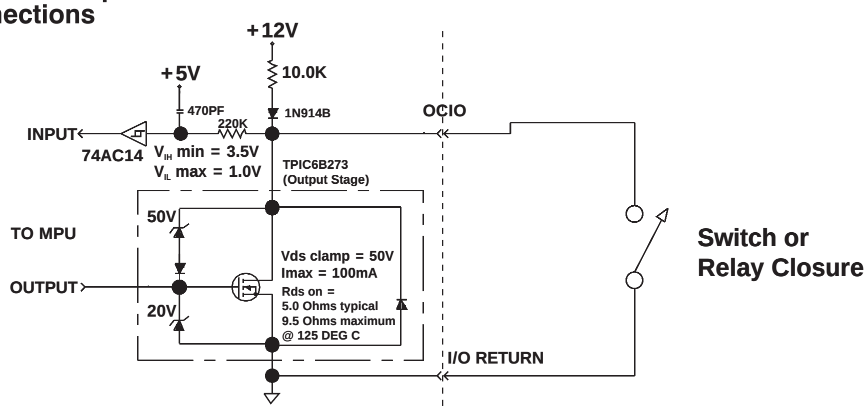
The OCIO output channel is a current-sinking (open drain) DMOS transistor. The power-on output state is off. Outputs are voltage clamped for inductive transient protection. The output transistor specifications are:

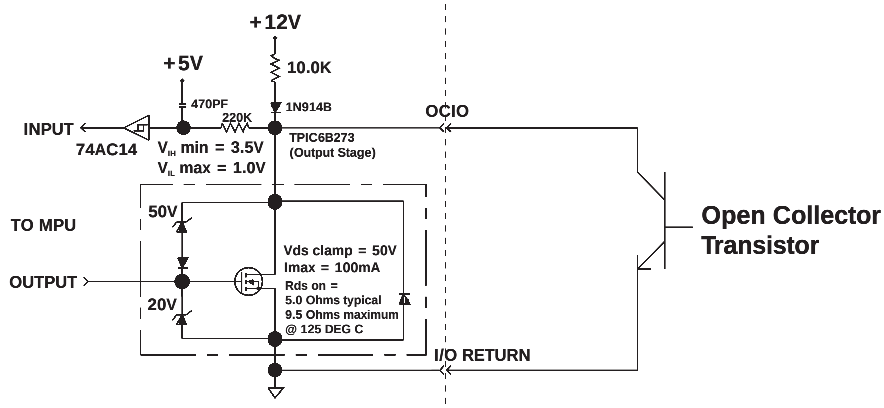
V_o max:	50 V
I_o max:	100ma
r_{ds}(on):	5Ω - typical @ 100 ma, 25°C 9.5Ω - maximum @ 100 ma, 125°C

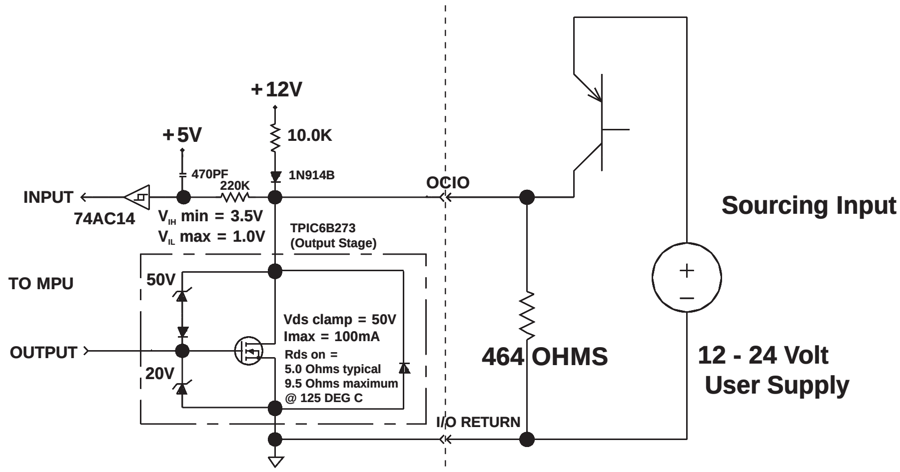
OCIO +5VDC

A +5V DC supply resides on the SC900. This supply is connected to J52-25, as well as J51-4 (serial port), and J4-25 (command port on base unit). The **total** current draw from **all** connections must not exceed 200 milliamperes.

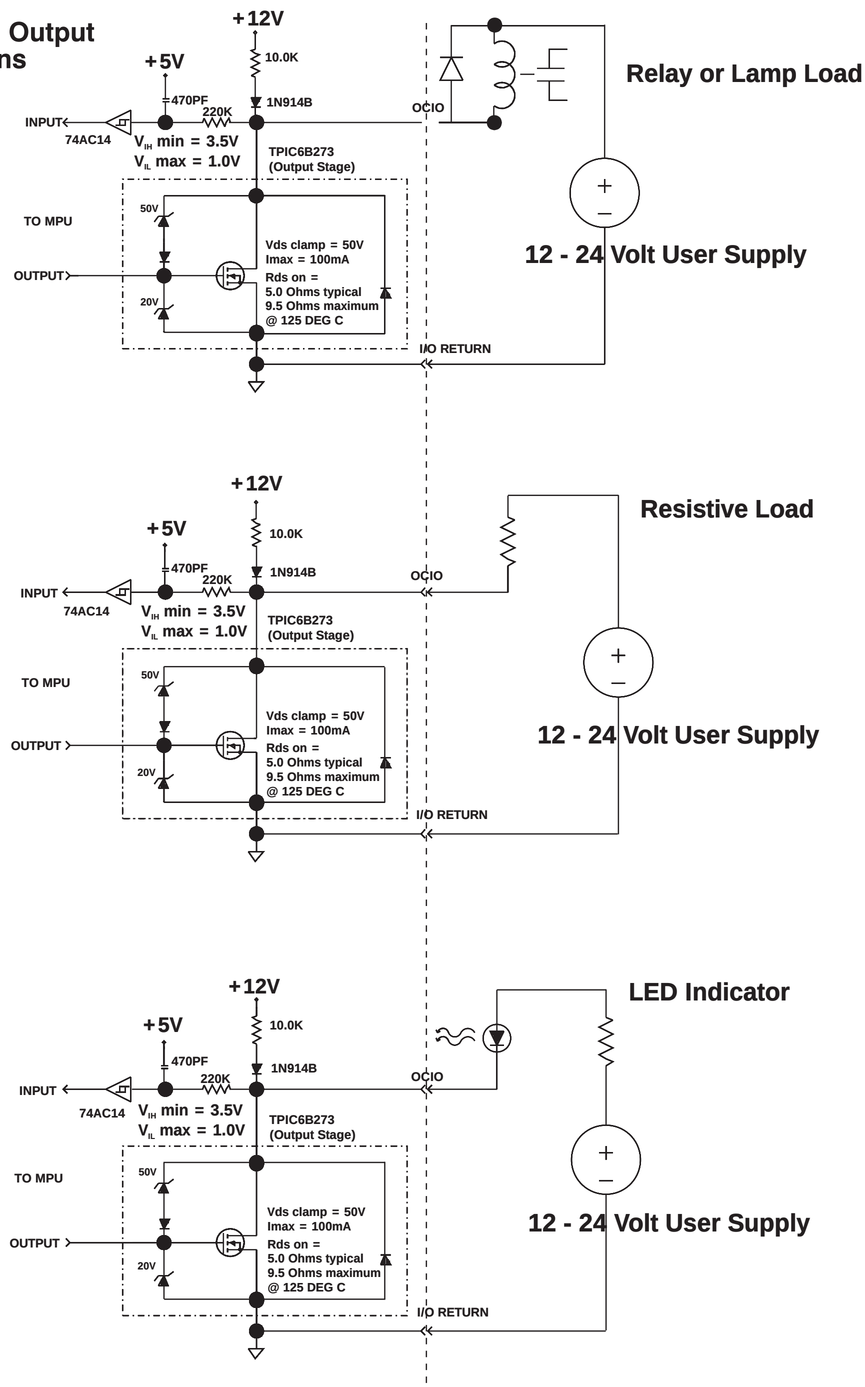
OC950 I/O Input Connections





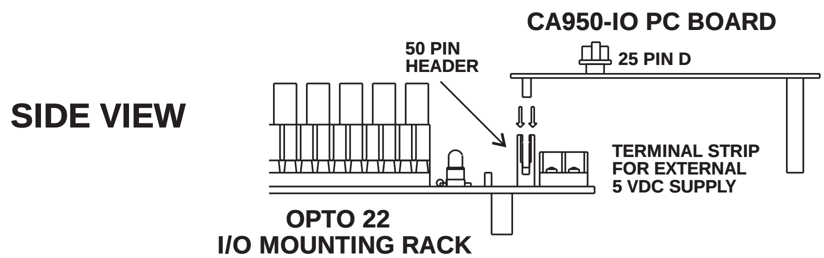


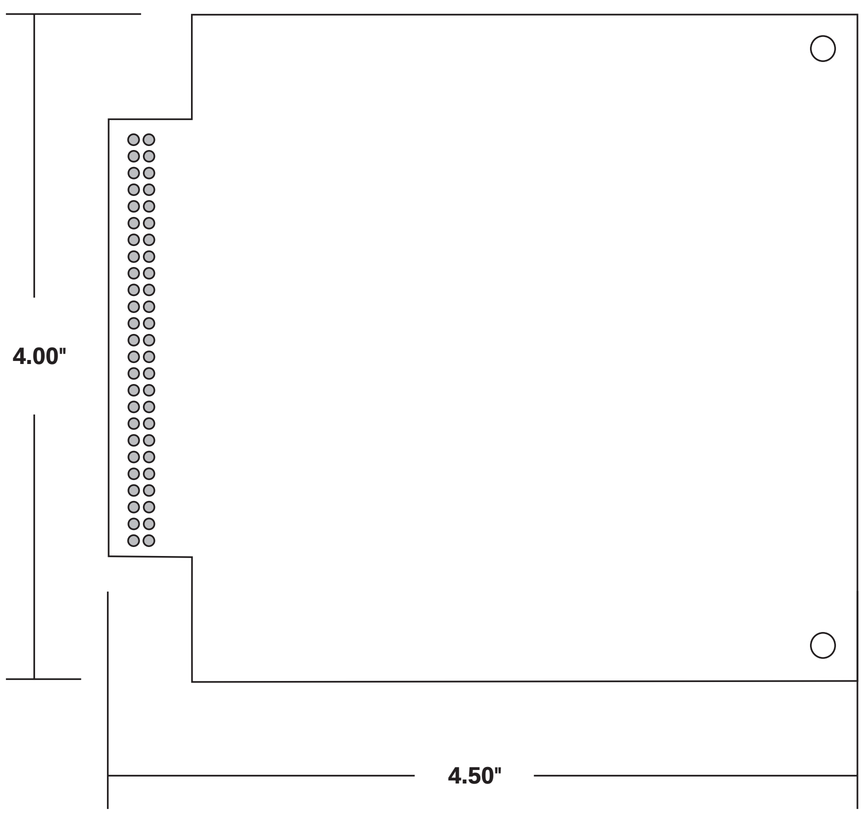
OC950 I/O Output Connections



3.4 Connecting to the Opto22 Mounting Rack

The CA950-IO PC Board allows for easy connections between the OC950 and optically isolated industrial I/O. The CA950-IO plugs directly into the 50 pin header of Opto 22's Generation 4 family of I/O Mounting Racks and other compatible products as shown below:





CA950-IO

Direct pin-to-pin connections between the male 25 contact D connector on the CA950-IO board to the female 25 contact D connector on the OC950 (J52) allows use of up to 21 I/O modules on the I/O mounting rack. This connection is best accomplished using a DB25 male to female shielded cable with EMI/RFI metallic hoods.

Note: *Recommended cable length is six feet or less.*

The OC950 I/O number corresponds to the Mounting Rack Module Position. For example, inp0/out0 connects to Module Position 0, inp1/out1 connects to Module Position 1, etc. Obviously, when a 16 module rack or smaller is used, the number of I/O is limited by the rack size. A schematic of the CA950-IO is given at the end of this section.

Inputs

OC950 discrete inputs are compatible with all Opto 22 Input Modules specified to operate with an Output Supply Voltage of 5 Vdc Nominal. These include:

- G4IDC5
- G4IDC5B
- G4IDC5D
- G4IDC5G
- G4IDC5K
- G4IAC5
- G4IAC5A

Driving the user's side of these input modules (so that the current flows in the emitting diode of the opto coupler) provides a low input to the corresponding OC950 discrete input. This will return a value of zero for the variable inpX, where X is the discrete input being used.

Outputs

OC950 discrete outputs are compatible with all Opto 22 Output Modules specified to operate with a Logic Pickup Voltage of 4 Vdc and Logic Dropout Voltage of 1 Vdc. These include:

- G4ODC5
- G4ODC5A
- G4OAC5
- G4OAC5A
- G4OACA5(NC)

Setting an OC950 discrete output to 0 using the 950BASIC assignment statement outX = 0 will drive current in the opto isolator's emitting diode which turns on the transistor or Thyristor switch at the user's side of the isolated interface.

OC950 discrete outputs can also be used to drive the G4OD5R and G4ODC5R5 Dry Contact Output Modules. Setting an OC950 output to zero will drive current in to the associated relay coil, closing the switch for the G4ODC5R and opening the switch for the G4ODC5R5.

Power Supply Considerations

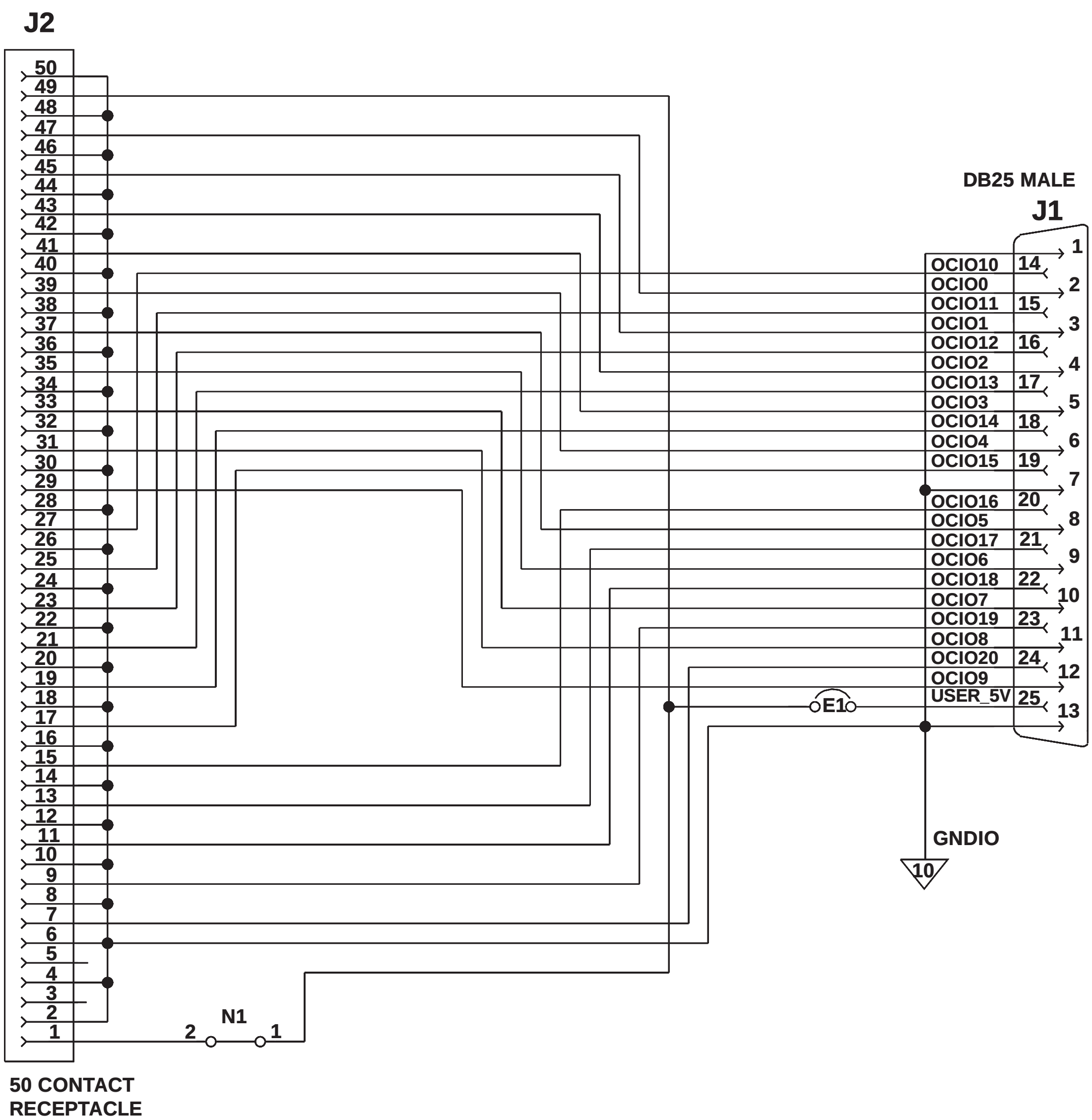
The OC950 +5 Vdc user power supply can source up to 200 ma. This can be used to power up to 16 I/O modules,[1] providing this supply is not used for other purposes (encoder or user's terminal). If the supply is used for other purposes or if more than 16 I/O modules are required, then the I/O mounting rack must be powered using a separate power supply. An isolated +5 Vdc power supply can be connected to the +5V and GND terminals on the Opto22 I/O Mounting Rack or the Rack can be powered using the Opto 22 PBSA power supply which operates from either 115 or 220 Vac. The PBSA supply fits underneath the CA950-IO board.

Important

If an external supply or the PBSA is used to power the I/O Mounting Rack, remove the plug-on jumper (E1) from the CA950-IO PC Board. Leave this jumper installed only if the I/O Mounting Rack is to be powered using the OC950's 5 Vdc supply.

1 Each I/O Module draws up to 12 mA @ 5Vdc. Therefore, the current available for other uses (encoder or user terminal) =200 ma - NumberOfModulesPowered * 12 mA.

CA950-IO Schematic



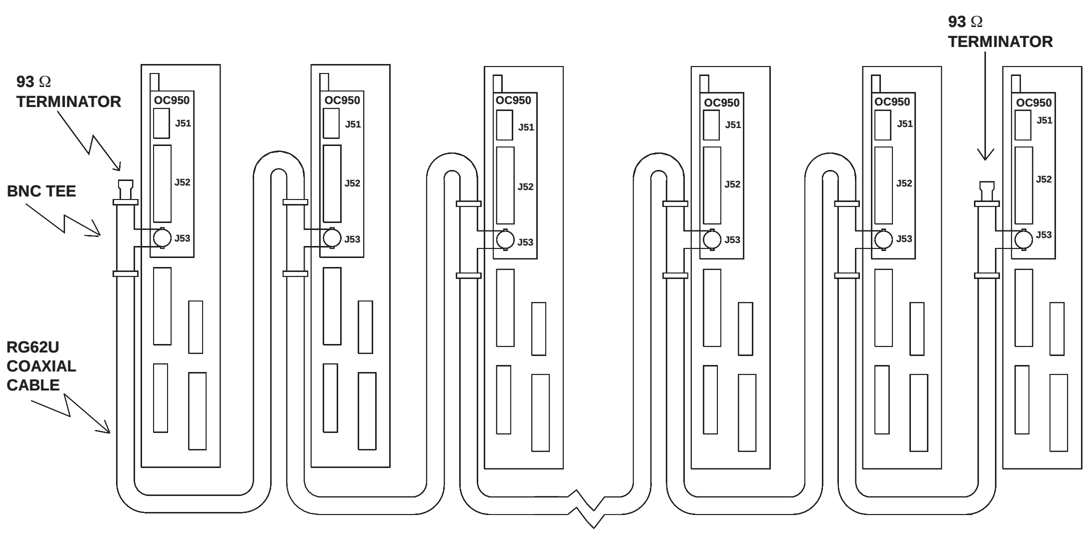
3.5 PacLAN Cabling & Hardware Connections - J53

Note: *PacLAN is only available on the following models:*

- OC950-503-01, OC950-603-01
- OC950-504-01, OC950-604-01

PacLAN connections

The electrical connections of the PacLAN interface between SC950 servocontrollers is illustrated in the diagram below.



Cabling for the PacLAN interface must conform to RG62 cabling specifications.

The most distant trunks of the cable should each be terminated in 93 ohm, 1/2 watt termination resistors. RG62 cable and 93 ohm terminators are available from Black Box Corporation and South Hills DataComm.

Note: *PacLAN operation will be supported only if the proper communications cabling is used.*

Coaxial cable requirements

The following table outlines the requirements for coaxial cable.

Summary of Coaxial Cable Requirements	
Minimum Spacing	1 m
Station impedance	$\geq$ 1.2KΩ
Segment termination value	93Ω
Maximum number of stations per segment	8
Maximum bus segment length	300 m

Active hubs

PacLAN permits direct connection of up to 8 SC950 servocontrollers. The maximum direct bus segment length is 1000 ft (305 meters). To achieve longer cable lengths, or to connect more than 8 controllers, ARCNET compatible active hubs must be used. These devices are capable of transmission lengths of up to 2000 feet. Active hubs are available from SMC, Contempory Control Systems, Inc., and South Hills DataComm.

PacLAN axis address setup

Each SC950 must have a unique PacLAN axis address. The PacLAN address selection is performed by configuring the S1 DIP switch.

Note: *The switch setting is read only when power is first applied to the controller.*

Each SC950 Servocontroller axis must have a unique port address. Please refer to Section 3.2 for additional information on setting the address.

Note: *PacLAN Address 0 is reserved. DO NOT USE.*

4 Servo Loop Parameters

Introduction

This chapter describes setting parameters associated with the velocity and position loops. In some cases the user must adjust control loop parameters due to large mismatches between motor and load inertia, mechanical resonance, backlash, etc. This chapter provides guidance for handling these situations.

Refer to Appendix A for a description of the control loop architecture and control loop block diagrams.

Note: *The two anti-resonant zeroes (ARZ0 and ARZ1) are assumed to both be off (set to zero) for this discussion.*

4.1 Velocity Loop

The velocity loop block diagram is shown in Figure 2 of Appendix A. Velocity loop bandwidth is the key indicator of system performance. Systems with fast settling time must have high velocity loop bandwidth. Conversely, if the velocity loop bandwidth is low, attempting to achieve fast settling time by increasing the position loop bandwidth, KPP, leads to overshoot and ringing.

Velocity loop bandwidth

The velocity loop bandwidth (f_{vc}) is given by the equation:

$$f_{vc}(Hz) = \frac{KVP * K_T \sqrt{3}/2}{2\pi * J_{TOT}} \approx 0.138 * KVP * \frac{K_T}{J_{TOT}}$$

where:

KVP is the velocity loop proportional gain in amps/(rad/sec)

K_T is the 0-peak line-line motor torque constant in lb-in/amp

J_{TOT} is the total inertia (motor total + load total) in lb-in-sec^2.

(Any consistent set of units for K_T, J_{TOT}, such as MKS, that yields K_T/J_{TOT} in rad/sec^2/amp will work)

The motor torque constant is the value of K_T peak published in the Pacific Scientific Motion Control Solutions catalog.

Note: *f_{vc} is the unity gain open-loop crossover frequency of the idealized rigid single mass system. See hardware specifications for maximum f_{vc} value.*

Default bandwidths

When creating a new program, the selections made set KVP to achieve the velocity loop bandwidths shown below, assuming there is no load on the motor shaft and the motor has no mechanical brake or other secondary devices installed.

Note: *The bandwidth depends upon the user's selection for desired system response:*

	Gentle	**Medium**	**Stiff**
f_{vc} Velocity Loop Bandwidth (Hz)	25	75	200

Load inertia

From the formula for bandwidth, it is seen that bandwidth changes inversely with total inertia. If the load inertia equals the motor plus resolver inertia, the velocity loop bandwidth will be half the values shown. If the load inertia is ten times the motor plus resolver inertia, the bandwidths will be one eleventh these values. Clearly KVP must be increased to compensate for increased load inertia if bandwidth is to be maintained. Typically, load inertia up to 3(motor + resolver) give acceptable performance without further optimization.

The most common servo setup problem is adding large load inertia without a corresponding increase in KVP.

The value of KVP to achieve a desired bandwidth can easily be calculated as follows:

$$KVP = \frac{2\pi * f_{vc} * J_{TOT}}{K_T \sqrt{3}/2} \approx 7.26 * f_{vc} * \frac{J_{TOT}}{K_T}$$

Example calculation

For example, to achieve 75 Hz bandwidth with an R32G motor having 20 to 1 load inertia = 0.011 lb-in-sec^2:

J_{TOT}[1] = 0.00055 + 0.011 = 0.01155 lb-in-sec^2

K_T[2] = 4.4 lb-in/amp

$$KVP = 7.26 * 75 * \frac{0.01155}{4.4} = 1.43$$

1 Motor plus resolver inertia (0.00055 lb-in-sec^2) for the R32G motor can be found in the catalog.

2 K_T can be found in the catalog as K_T peak (4.4 lb-in/amp).

950IDE can also be used to make the calculation. Simply enter the inertia ratio and 950IDE will calculate the appropriate value for KVP to achieve 25, 75 or 180 Hz bandwidth depending upon the choice made for system response.

There is no specific answer to the general question "What should the bandwidth be?" In general, the higher the velocity loop bandwidth, the faster the settling time will be and the better the rejection of torque disturbances (increased stiffness). Typically, velocity loop bandwidths range from 30 to 100 Hz. However, too high a bandwidth can lower the damping of resonance in mechanical linkages, causing excessive ringing and/or wear in coupled mechanics. Remember, it is the resulting motion at the end of any mechanical linkages that typically matters, not the response at the motor shaft.

Problems with high load inertia

It would seem from the above that setting KVP is simply a matter of increasing its value to compensate for load inertia. Unfortunately, the following problems often interfere, particularly when the load inertia is large compared with the motor's inertia:

1. Mechanical resonance between motor and load cause high frequency oscillation.
2. Backlash between motor and load effectively unload the motor over a small angle. Within this small angle the increased bandwidth results in oscillations.
3. Ripple in the velocity feedback signal results in large motor ripple current if KVP is large.

As a general rule, any system with KVP set higher than 5 times the medium bandwidth setting will require adjustment to the default ARF0 and ARF1 settings.

Resonance

Mechanical resonance is caused by springiness between motor inertia and load inertia. This may result from belts, flexible couplings, or the torsion stiffness of shafts. **In general, the stiffer the couplings, the higher the resonance frequency and the easier it is to tune the system for good performance.**

If the velocity loop breaks into an oscillation at a frequency well above the calculated velocity loop bandwidth, a resonance problem may well exist. A second symptom is that the frequency of oscillation is relatively constant in the presence of changes to `ARF0` and `ARF1`.

ARF0 & ARF1

Two digital anti-resonant low-pass filters `ARF0` and `ARF1` are included in the velocity loop. Their purpose is to lower the gain above f_{vc} and especially at any resonant frequency > f_{vc} so that oscillations do not occur. Default values, also a function of the selected system response, are shown below:

	Gentle	**Medium**	**Stiff**
ARF0 (Hz)	100	150	1500
ARF1 (Hz)	200	750	$1x10^5$

If the velocity loop bandwidth cannot be raised to an acceptable value without encountering a resonant oscillation, the procedure on the following page is recommended.

Procedure

1. Set both ARF0 and ARF1 to 400 Hz and set KVP low enough to prevent oscillation.
2. Increase KVP slowly until oscillation at the resonant frequency just begins. Then, reduce KVP slightly until the oscillation just stops. Compute the velocity loop bandwidth using the formula given at the beginning of this section. If the velocity loop bandwidth is less than .25 times the value of ARF0 and ARF1, then proceed to Step 3. Otherwise, go to Step 4.
3. Decrease both ARF0 and ARF1 by 20% and go back to Step 2.
4. The velocity loop bandwidth should now be approximately one quarter the value of ARF0 and ARF1. For margin, reduce KVP, ARF0, and ARF1 by 20%.

Backlash

Some backlash may be unavoidable, especially when gear reduction is used. If backlash is present, the inertia match must be good (load inertia should be roughly equal to motor inertia) for good servo performance. Gearing reduces the inertia reflected to the motor by the square of the gear reduction from motor to load. Therefore, select a gear ratio to give the required match.

Current ripple

The velocity feedback signal in standard SC900 Drives operating with the standard 20 arcmin resolver can have up to 3% p-p ripple. The resulting motor torque current ripple, with no ARF0/ARF1 filtering, can be calculated using the following formula:

$$\text{Current ripple (amps p-p)} = \frac{3}{100} * \text{Speed (RPM)} * \frac{2\pi}{60} * KVP$$

$$\approx 0.003 * \text{Speed (RPM)} * KVP$$

There can be cause for concern when this p-p number exceeds 40% of the drive's or motor's current rating. The motor current should be monitored using Dac Monitors on J4-3 to insure actual ripple current, with ARF0/ARF1 filtering, is not excessive.

Motor current ripple can often be reduced by lowering the ARF0, ARF1 low-pass filter break frequencies. This benefit is limited by velocity loop bandwidth and stability constraints. Velocity feedback ripple, and hence motor current ripple, can also be reduced by specifying a higher accuracy resolver.

KVI

The parameter KVI sets the so called "lag-break" frequency of the velocity loop. KVI is equal to the frequency in Hz where the velocity loop compensation transitions from predominantly integral characteristics to predominantly proportional characteristics. Drive rejection of torque disturbances increase as KVI increases. Default values for KVI are shown below:

	Gentle	**Medium**	**Stiff**
KVI (Velocity Loop Lag-Break Freq. (Hz))	1.7	5.0	13.3

If the Drive is to be used within a position loop (either with BlkType = 2 or when using an external position drive and BlkType = 1), KVI should be equal to or less than 0.1 times the velocity loop bandwidth. If no position loop is used, KVI can be set to 0.25 times the velocity loop bandwidth (or higher if some ringing can be tolerated). In general, the response to a velocity command step (or truncated ramp) will have velocity overshoot for non-zero values of KVI.

4.2 Position Loop

When BlkType is set equal to 2, a position loop is configured outside the velocity loop described in the previous section. Figure 3 in Appendix A illustrates the structure of the position loop. **The velocity loop must be set up and evaluated in terms of bandwidth before attempting to setup the position loop.**

KPP

The position loop proportional gain, KPP, determines the settling time of the position loop. KPP is the bandwidth of the position loop, in Hz, assuming an ideal velocity loop. Default values for KPP are shown below:

	Gentle	**Medium**	**Stiff**
KPP (Position Loop Bandwidth (Hz))	5	15	50

In general, the higher the value of KPP, the faster the settling time. However, **trying to set KPP to a high value with inadequate velocity loop bandwidth results in overshoot and ringing.** A good trade off is to set KPP to 0.2 times the velocity loop bandwidth. Slightly higher values can be used if overshoot can be tolerated.

KVFF

KVFF is the velocity feed forward gain. In the absence of velocity feed forward (KVFF = 0), the commanded velocity is proportional to the position (following) error. This means that the actual position will lag the commanded position by a value proportional to the speed. The error will be smaller for larger values of KPP.

The following table gives a feel for the following error magnitude.

Speed (rpm)	**KPP (Hz)**	**Following Error (revolutions)**
1000	10	0.27
2000	10	0.53
5000	10	1.33
1000	20	0.13
2000	20	0.27
5000	20	0.66

Note: *The following error can easily exceed one complete motor revolution. In many electronic gearing applications, such following errors are not acceptable (real gears don't have following errors!) Also, stepper systems don't have such errors.*

Feed forward takes advantage of the fact that the SC900 DSP knows the frequency of the encoder or step inputs and hence knows how fast the motor should be going at a given instant. All or part of this velocity can be added to the velocity command to reduce following error. If KVFF is set to 100 (%), then the steady state following error reduces to zero.

Overshoot

Setting KVFF equal to 100% can result in position overshoot. Somewhat lower values may be required if this is a problem. KVFF set to 70%-80% typically achieves the fastest step response with no overshoot. However, setting KVFF to less than 100% will give steady state following error when running at constant speed.

4.3 Advanced Velocity Loop Tuning

Continuous time transfer function approximation

The transfer function for the velocity loop compensation block is given below:

$$\frac{FVelErr}{VelErr}(s) = \frac{\left(\frac{s}{\omega_z}\right)^2 + \frac{1}{Q_z}\frac{s}{\omega_z} + 1}{\left(\frac{s}{\omega_f}\right)^2 + \frac{1}{Q_f}\frac{s}{\omega_f} + 1}$$

$$\frac{ICmd}{VelErr}(s) = \frac{\left(\frac{s}{\omega_z}\right)^2 + \frac{1}{Q_z}\frac{s}{\omega_z} + 1}{\left(\frac{s}{\omega_f}\right)^2 + \frac{1}{Q_f}\frac{s}{\omega_f} + 1}(KVP)\left(1 + \frac{2\pi(KVI)}{s}\right)$$

Definitions for the terms used in the equations above are shown on the following page.

For ARx0 > 0 both roots are real and:

$$\omega_x = 2\pi\sqrt{(ARx0)(ARx1)}$$

$$Q_x = \frac{\sqrt{(ARx0)(ARx1)}}{Arx0 + ARx1}$$

For ARx0 < 0 roots are a complex pair and:

$$\omega_x = -2\pi ARx0$$

$$Q_x = ARx1$$

Note: *When ARZ0 and ARZ1 are both zero, the numerator of* $\frac{FvelErr}{VelErr}(s)$ *reduces to 1. If* ARZ0 *or ARZ1 is individually 0 the numerator reduces to* $\frac{s}{2\pi ARZx} + 1$

Discrete time transfer function

The velocity loop compensation is actually implemented as a digital discrete time system function on the DSP. The continuous time transfer function is converted to the discrete time domain by a backward Euler mapping:

$$s \circledR \frac{1}{T_s}\left(1 - z^{-1}\right)$$

where T_s = 250 μsec.

5 Maintaining/Troubleshooting

In this Chapter

This chapter covers maintenance and troubleshooting of the SC950 servocontroller.

5.1 Maintaining the SC950 Servocontroller

Introduction

The SC950 series servocontrollers are designed for minimum maintenance. The following cleaning procedure, performed as needed, will minimize problems due to dust and dirt buildup.

Procedure

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

5.2 Troubleshooting the SC950 Series Servocontroller

Introduction

The system status display located on the front panel indicates unit status and is useful for troubleshooting.

0 means the drive is not faulted and not enabled, while 8 means the drive is not faulted and enabled. 0. means the drive is disabled and a program is running, 8. means the drive is enabled and a program is running. Alternating 8¬ means actively inhibiting CW motion and alternating 8r means actively inhibiting CCW motion.

System Status table

The following table lists the System Status LEDs for the OC950.

Status LED	Value	Fault Meaning
(Blinking) 1	1	Velocity feedback (VelFB) over speed
(Blinking) 2	2	Motor Over-Temp
(Blinking) 3	3	Drive Over-Temp
(Blinking) 4	4	Drive I*t
(Blinking) 5	5	l-n Fault (9x3)
(Blinking) 6	6	Control ±12 V supply under voltage
(Blinking) 7	7	Output over current or bus over voltage
(Blinking) 9	9	Shunt regulator overload
(Blinking) A	10	Bus OV detected by DSP
(Blinking) b	11	Auxiliary +5V Low
(Blinking) C	12	Not assigned
(Blinking) d	13	Not assigned

System Status table (cont'd)

Status LED	Value	Fault Meaning
(Solid) E*	14	Processor throughput fault
(Blinking) E*	14	Power Up Self Test Failure
(Alternating) E1	225	Bus UV, Bus Voltage VBusThresh
(Alternating) E2	226	Ambient Temp Too Low
(Alternating) E3	227	Encoder commutation align failed (Only CommSrc=1)
(Alternating) E4	228	Drive software incompatible with NV memory version
(Alternating) E5*	229	Control Card hardware not compatible with drive software version
(Alternating) E6	230	Drive transition from unconfigured to configured while enabled
(Alternating) E7	231	Two AInNull events too close together
(Alternating) F1	241	Excessive Position Following Error
(Alternating) F2*	242	Program Checksum Error (Memory Error)
(Alternating) F3*	243	Parameter Checksum Error (Memory Error)
(Alternating) F4	244	Run Time Error (Err)

*FaultReset cannot reset these faults.

See the table on the following page for further information on Blinking E, Blinking 1 and Alternating F2 and F3.

Extended Faultcodes

LED Display	Value of ExtFault	Description
1	1	|VelFB| < 21038
1	2	|VelFB| < 1.5*max(|VelLmtxx|)
E	0	No ExtFault information
E	1	Resolver calibration data corrupted
E	2	Excessive dc offset in current feedback sensor
E	3	DSP incompletely reset by line power dip
E	6	Excessive dc offset in Analog Command A/D
E	7	Unable to determine option card type
E	8	DSP stack overflow
E	10	Firmware and control card ASIC incompatible
E	11	Actual Model does not match value in nonvolatile memory
E	12	Unable to determine power stage
E	15	RAM failure
E	16	Calibration RAM failure
F2	13	Control card nonvolatile parameters corrupt
F3	14	Option card nonvolatile parameters corrupt

A table of problems, causes, and appropriate actions are on the pages that follow.

Troubleshooting table

Problem (and system status numbers)	Possible cause	Action
System status display not lit.	No control power.	Check that SC955 has AC switch set to INT **or** SC952, SC953 and SC954, have 115 V ac or 230 V ac applied to J3 pins 5,6.
	Blown control power fuse.	Replacement of fuses within the drive is not recommended and rarely useful. A blown fuse is a strong indication that the drive is defective and should be returned to the factory for repair.
Controller unconfigured (U/C).	New controller.	Perform controller setup.
Resolver conversion overspeed (1).	Incorrect resolver phasing.	Verify proper phasing.
	Open or intermittent resolver connection.	Check connector and cable.
Motor overtemperature (2).	Motor PTC pins open.	Connect PTC pins (J3-8 and J3-9).
	Temperature overload due to high motor ambient temperature or excessive RMS torque.	Lower ambient temperature. Operate within continuous torque rating.
	Motor PTC damaged.	Check motor PTC for resistance ≤ 310 ohm.

Troubleshooting table (cont'd)

Problem (and system status numbers)	Possible cause	Action
Controller overtemperature (3).	Temperature overload due to:	
	High ambient temperature.	Lower ambient temperature to below 50°C (60°C, if derated).
	Restriction of cooling air due to insufficient space around unit.	Provide sufficient cooling space.
	Operation above continuous power rating.	Operate within continuous power rating. Add fan option to controller to boost continuous rating.
	Fan inoperative.	Return to factory for fan replacement.
IT Fault (4).	Excessive time at peak current.	Check time spent at peak current. If excessive, change profile or load. (Use larger motor/drive.)
Line-neutral fault (5).	Motor cable short to ground. Motor winding short.	Check motor cable.
	Internal failure.	Contact distributor.
Servocontroller logic supply undervoltage (6).	External short on signal connectors.	Remove connectors and reapply power.
	Insufficient voltage on J1-5 and J1-6.	Check voltage with meter ($\geq$ 90 V).
	Internal failure.	Contact distributor.

Troubleshooting table (cont'd)

Problem (and system status numbers)	Possible cause	Action
Output overcurrent, or bus overvoltage (7).	Excessive ac input voltage.	Reduce ac input voltage to below 264 V ac.
	Output short circuit.	Check for short.
	Motor cabling wires shorted together.	Check for short.
	Motor cabling shorted to ground.	Check for short.
	Open or missing regen resistor.	Check regen wiring.
	Internal motor winding short circuit.	Check for short.
	Insufficient motor inductance. SC952 > 4mH SC953 > 2mH SC954 > 1mH SC955 > 0.5mH	Check current loop compensation parameters in software. Replace motor with motor of correct inductance.
Drive enabled (8)	Normal Operation	None.
Drive enabled, no faults (8.)	Program running.	None.
Enabled, inhibits active. (8 ⌐) CCW Inhibit active (8 ¬) CW Inhibit active (8 ∩) CCW&CW Inhibit active	Drive Inhibits are active.	Deactivate inhibit inputs.
Unit will not enable (does not display 8 when enable applied).	External enable not applied.	Apply enable input to pin J4-6.
	Internal fault.	Contact distributor.
	Software enable not on.	Set Enable variable to 1.
	No bus power. (9x4, 9x5 only)	Apply Bus power.

Troubleshooting table (cont'd)

Problem (and system status numbers)	Possible cause	Action
Shunt regulator overload (9)	Excessive regen in application	Increase cycle time. Reduce the inertia. Add external regen with higher wattage.
	Improper external regen wiring or components on J5.	Check connections on J5.
	Internal failure.	Return to factory for repair.
Bus overvoltage (A)	Bus AC input overvoltage.	Check AC voltage < 264.
	Internal failure.	Return to factory for repair.
+5 V low (b)	Short on pin J52-25.	Remove J52 connector. Reapply AC power.
	Short on pin J51-4.	Remove J51 connector. Reapply AC power.
	Short on pin J4-25.	Remove J4 connector. Reapply AC power.
	Internal failure.	Return to factory for repair.
Microprocessor throughput fault. (E)	Intermittent failure from environment noise.	Unconfigure drive and cycle AC power. Check ground connections. Check for excessive ground noise and loose AC connections.
	Internal failure or firmware version conflict.	Return to factory for repair.

Troubleshooting table (cont'd)

Problem (and system status numbers)	Possible cause	Action
Bus Undervoltage (E1)	VBus < VBusThresh	VBusThresh is set too high.
		VBus is low because bus AC power is low. Check J1-1, J1-2, and J1-3.
Ambient temperature too low (E2)	Ambient temperature < 0°C	Measure temperature in control cabinet. Condition if necessary.
	Internal failure.	Return to factory for repair.
Encoder commutation alignment failure (E3)	Not enough current to lock the motor shaft to proper position within 1 second.	Check IlmtPlus and IlmtMinus. If vertical load, reduce load or increase motor/drive. Verify motor brake releases properly.
	Internal failure.	Return to factory for repair.
Incompatible 950IDE (E4)	950IDE version is incompatible with drive's NV memory version.	Verify that FWV and 950IDE version are the same. If FWV = 200, then 950IDE should be version 2.0 or higher.
Control card hardware not compatible with drive software version (E5)	Resolver wiring error.	Remove J2 and J3 connectors. Cycle power. If fault is now a "2", then correct resolver excitation wiring.
	Internal failure.	Return to factory for repair.

Troubleshooting table (cont'd)

Problem (and system status numbers)	**Possible cause**	**Action**
Drive transition from UnConfigured to Configure while enabled (E6)	Hardware Enable input active.	Disable hardware Enable input. Re-configure drive then cycle power.
	Enable inactive during several configurations, drive still faults.	Return for repair.
Two AInNull events too close together. (E7)	Connection to AInNull input has switch bounces.	Use BDOutX to trigger the AInNull input. Replace switch if no fault occurs.
Following error overflow (F1)	CwInh and/or CcwInh held low while commanding motion.	Check CwInh and CcwInh inputs and variables.
	Loss of feedback information.	Disable drive and spin motor shaft by hand. Verify Position variable is changing. If not, swap out motor/cables or drive with know good spare to isolate defective part.
	Open motor phase.	Check wiring and motor winding for open circuit.
	Loss of motor power.	Bus voltage is low.
	Tuning Wizard incomplete.	No motor selected, therefore cannot respond.
User Program Checksum Fault (F2)	Checksum fault while reading user program memory during power up.	Cycle AC power. Download the program.
	Memory allocation error. Program is writing to a variable address location that has not been dimensioned.	Check Dim section of program for string variables and arrays.

Troubleshooting table (cont'd)

Problem (and system status numbers)	Possible cause	Action
Parameter checksum error (F3)	Checksum fault reading NV memory. Typically during power cycle or after program download.	Check for AC momentary glitch. Do not interrupt program or waveshape download with keystroke or power cycle.
		Download a new parameter set. Download a program with a valid parameters section.
	Internal failure.	Return to factory for repair.
Run time error (F4)	Programming error.	Read error number in Output Window or check value of Err in the variables window. Look up values in run-time error table and make necessary corrections to your program.
Processor failure (h)	Excessive noise in application or internal failure.	Cycle AC power. Return to factory for repair.
Unit is enabled, no system status faults (status code 8), but motor does not respond.	Seized load or excessive load friction.	Reduce load.
	Reset command on J4-7.	Remove reset from J4-7.
	No bus power.	Apply bus power.
	Open motor connections.	Connect motor.
Motor oscillates or runs erratically.	Improper shielding and grounding.	Shield cabling correctly per Section 2.5.1 or Section 2.5.2.
	Improper drive set up.	Review set up procedure.

Communications Troubleshooting Table

Symptom	Corrective Action
SC950 will not respond to commands over the serial link	Verify that SC950 is NOT running a program (8. or 0. on status display).
	Verify that axis address is set to 255 using switch S1.
	Verify that baud rate and COM port are set correctly in the 950 IDE.
	Check terminal transmit and receive lines from computer to receive and transmit lines on J51. (See Section 3.2)
	Verify that the serial cable is functioning properly. 1. Disconnect serial cable from J51. 2. Short pins 2 & 3 (See Section 3) 3. Type a character on the keyboard. 4. Verify that character entered echoes back to output window.
	Verify that three wire serial cable is connected to J51.
	Internal failure. Return to factory for service.
SC950 will not respond during RS-422/RS-485 operation	Verify that each unit has a unique serial address using switch S1. (See Section 3.2)

If problems continue to exist, please contact your local distributor for further assistance.

If the servocontroller is defective

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

Return procedure

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

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

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

3. Ship the servocontroller to:

   Pacific Scientific
   110 Fordham Road
   Wilmington, MA 01887
   Attn: Repair Department, RMA #________

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

   Pacific Scientific
   4301 Kishwaukee Street
   Rockford, IL 61105
   Attn: Brushless Repair Department, RMA#________

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

Appendix A Selecting Motor Control Functionality

Introduction

The SC900 family has three distinct modes of controlling the motor shaft and two distinct sources for the shaft command:

Modes

- Torque Control
- Velocity Control
- Position Control

Commands

- Analog Command
- Program Control

A.1 Torque Block Modes

A.1.1 Analog Command Torque Block (BlkType = 0)

This mode allows the differential analog voltage between terminals J4-1 and J4-2 to set the motor's terminal torque current amplitude. Since the actual motor current amplitude (`IFB`) times the motor's 0-peak line-line torque constant K_T times $\sqrt{3}/2$ is the shaft torque, then the analog input directly controls motor shaft torque. The overall gain of this block, i.e. the output current amplitude in amps per input volt, is set by the `CmdGain` parameter directly in Amp/V and should be set by the user to the desired value.

Command processing

Figure 1 shows the analog torque block mode has the same signal processing as a velocity loop except that the velocity error signal (`VelErr`) is set to `VelCmd` not to (`VelCmd - VelFB`) and that the `VelCmd` clamp is bypassed. Thus, the analog input goes through a number of signal processing steps before becoming the motor torque current command `ICmd`.

1. Analog differential amplifier with 1200 Hz low pass filter.
2. High resolution A/D sampled at the velocity loop update rate and added to the `ADOffset` parameter.
3. `ADF0` adjustable low pass filter to become `AnalogIn`.
4. Bypass the `VelLmtHi`, `VelLmtLo` clamp.
5. Velocity error variable `VelErr` is set equal to `VelCmdA`.
6. The anti-resonance second order velocity loop compensation block controlled by the `ARF0, ARF1, ARZ0`, and `ARZ1` parameters to become the `FVelErr` variable.
7. The proportional and integral velocity loop compensation block controlled by the `KVP` and `KVI` parameters respectively.
8. And finally through the `IlmtPlus` and `IlmtMinus` current command clamp to become the `ICmd` motor torque current command variable.

Although this looks like a large amount of processing, the options are only there to allow tailoring the response to fit a particular application.

Typically, most of the signal blocks are set to directly pass the signal so that `ICmd = CMDGain*(AnalogIn)` as directly as possible. The set of parameters below accomplish this result.

`ADF0` = 100,000 Hz to bypass, 1000 Hz by auto set up

`ARF0` = 100,000 Hz

`ARF1` = 100,000 Hz

`ARZ0` = 0 (not active)

`ARZ1` = 0 (not active)

`KVP` = 1 A/rad/sec

`KVI` = 0 Hz

IMPORTANT

The `KVP` parameter must be set to 1 A/rad/sec for the units on `CmdGain` to be correct. If `CmdGain` is set to 1 Amp/V and `KVP` to 2 A/rad/sec then an analog input of 1 volt will incorrectly give 2 amps of output torque current amplitude.

When changing the `BlkType` from something else to 0 to get an analog torque block you will generally need to additionally set `KVP` to 1, `KVI` to 0, and the other items in the above list to appropriate values to get the system working as desired.

A.2 Velocity Block Modes

A.2.1 Analog Command Velocity Block (BlkType = 1)

This mode allows the differential analog voltage between terminals J4-1 and J4-2 to set the motor's shaft velocity, also informally known as shaft speed. The overall gain of this block, i.e. the output shaft velocity per input volt, is set by the `CmdGain` parameter in kRPM/V and should be set by the user to the desired value.

Command processing

The analog input goes through a number of signal processing steps before becoming the actual motor velocity command `VelCmd` as shown by Figure 2.

1. Analog input differential amplifier with 1200 Hz low pass filter.
2. High resolution A/D sampled at the velocity loop update rate and added to the `ADOffset` parameter.
3. `ADF0` adjustable low pass filter to become the `AnalogIn` variable.
4. Range clamped by the `VelLmtHi, VelLmtLo` parameters.

Velocity loop compensation

The actual velocity command (`VelCmd`) is then combined with the measured shaft velocity (`VelFB`) and processed by the velocity loop compensation to create the motor torque current command (`ICmd`). The detailed signal processing steps to create ICmd are listed below and shown in Figure 1.

1. `VelErr` set equal to (`VelCmdA - VelFB`)
2. The anti-resonance second order velocity loop compensation block controlled by the `ARF0, ARF1, ARZ0,` and `ARZ1` parameters to become the `FVelErr` variable.
3. The proportional and integral velocity loop compensation block controlled by the `KVP` and `KVI` parameters respectively.
4. And finally through the `IlmtPlus` and `IlmtMinus` current command clamp to become the `ICmd` motor torque current command variable.

For more information on tuning the velocity loop see Chapter 4 Servo Loop Parameters.

A.3 Position Block Modes

A.3.1 Digital Command Position Block (BlkType = 2)

This mode is just a velocity block mode with the VelCmd coming from the position loop. See Figure 3. In particular,

`PosError = PosCommand` - Position Feedback

$$VelCmd = 2\pi * KPP * PosError + \frac{KVFF}{100} * \left[\frac{d}{dt} \left(PosCommand \right) \right]$$

where:

`PosCommand` is the position command in counts

`KPP` is the proportional position loop gain

`KVFF` is the velocity feed forward gain percentage.

When the SC900 is disabled and `BlkType = 2`, `PosCommand` is set to the position feedback value. This insures that when the drive is enabled, it picks up motion from its present position.

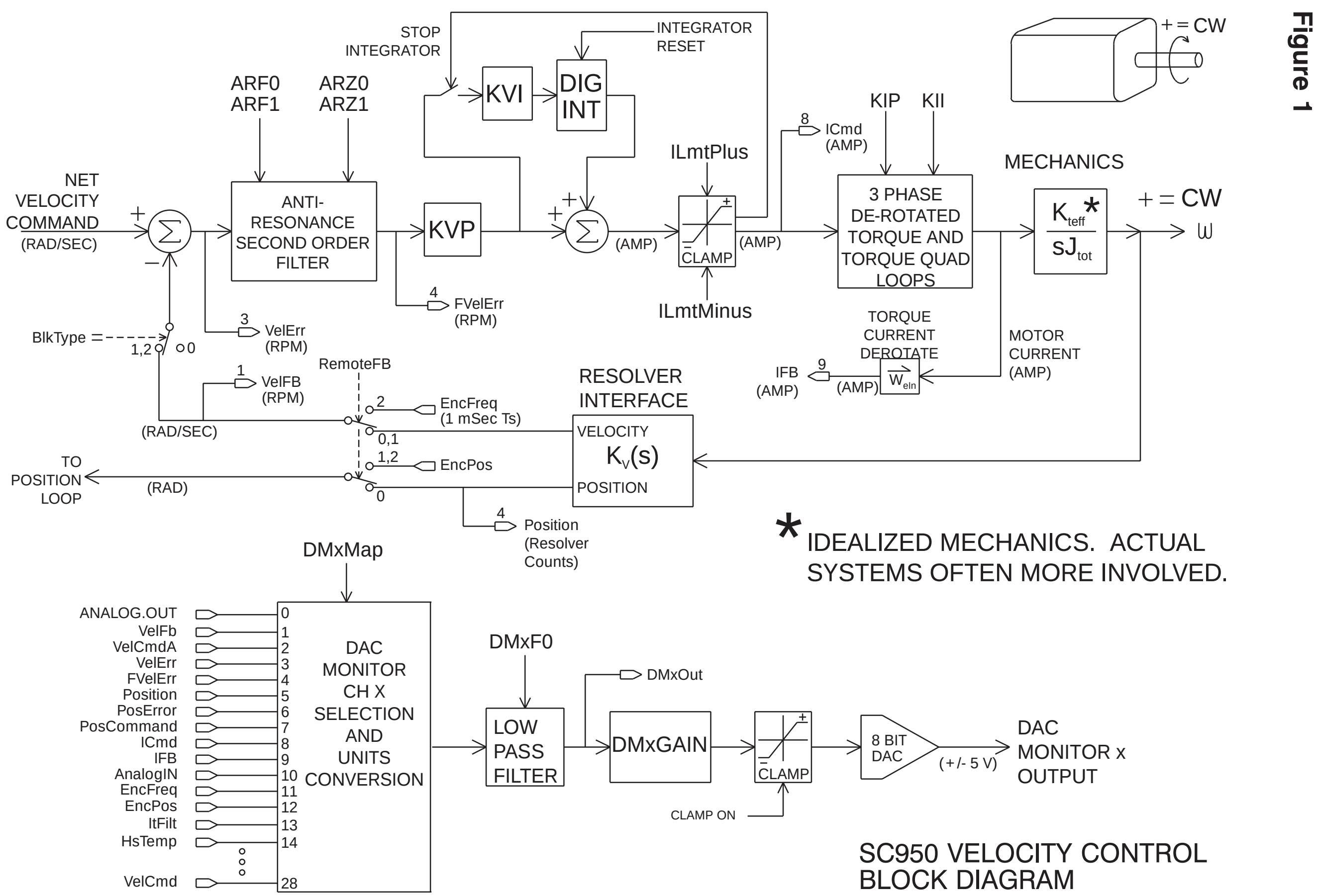
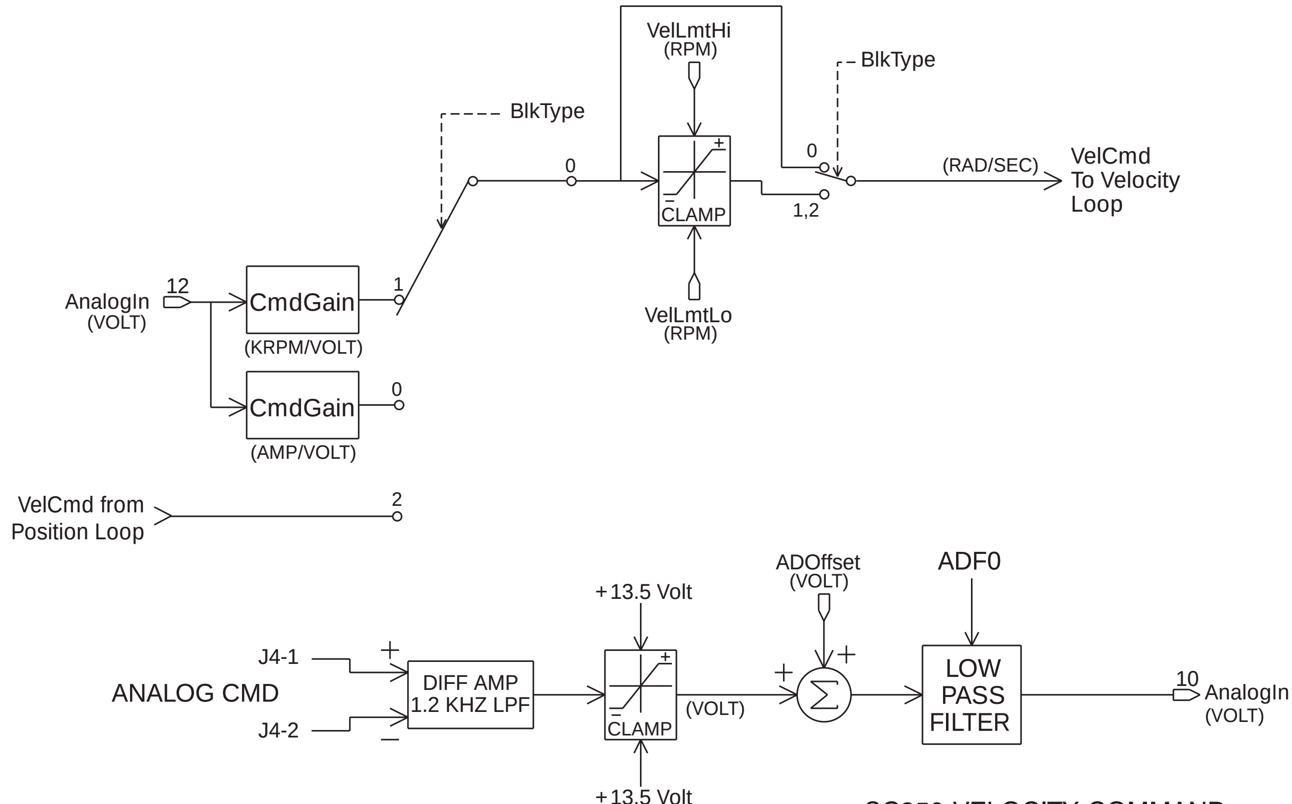

NET VELOCITY COMMAND
(RAD/SEC)
ARF0
ARF1
ARZ0
ARZ1
ANTI-RESONANCE SECOND ORDER FILTER
KVP
STOP INTEGRATOR
KVI
DIG INT
INTEGRATOR RESET
ILmtPlus
ILmtMinus
CLAMP
(AMP)
ICmd
(AMP)
KIP
KII
3 PHASE DE-ROTATED TORQUE AND TORQUE QUAD LOOPS
MECHANICS
+ = CW
ω
TORQUE CURRENT DEROTATE
MOTOR CURRENT (AMP)
IFB
(AMP)
VelErr
(RPM)
FVelErr
(RPM)
VelFB
(RPM)
BlkType =
1,2
0
RemoteFB
EncFreq
(1 mSec Ts)
0,1
1,2
EncPos
RESOLVER INTERFACE
VELOCITY
POSITION
(RAD/SEC)
TO POSITION LOOP
(RAD)
Position
(Resolver Counts)
* IDEALIZED MECHANICS. ACTUAL SYSTEMS OFTEN MORE INVOLVED.
DMxMap
ANALOG.OUT
VelFb
VelCmdA
VelErr
FVelErr
Position
PosError
PosCommand
ICmd
IFB
AnalogIN
EncFreq
EncPos
ItFilt
HsTemp
VelCmd
DAC MONITOR CH X SELECTION AND UNITS CONVERSION
DMxF0
LOW PASS FILTER
DMxOut
DMxGAIN
CLAMP
CLAMP ON
8 BIT DAC
(+/- 5 V)
DAC MONITOR x OUTPUT
SC950 VELOCITY CONTROL BLOCK DIAGRAM

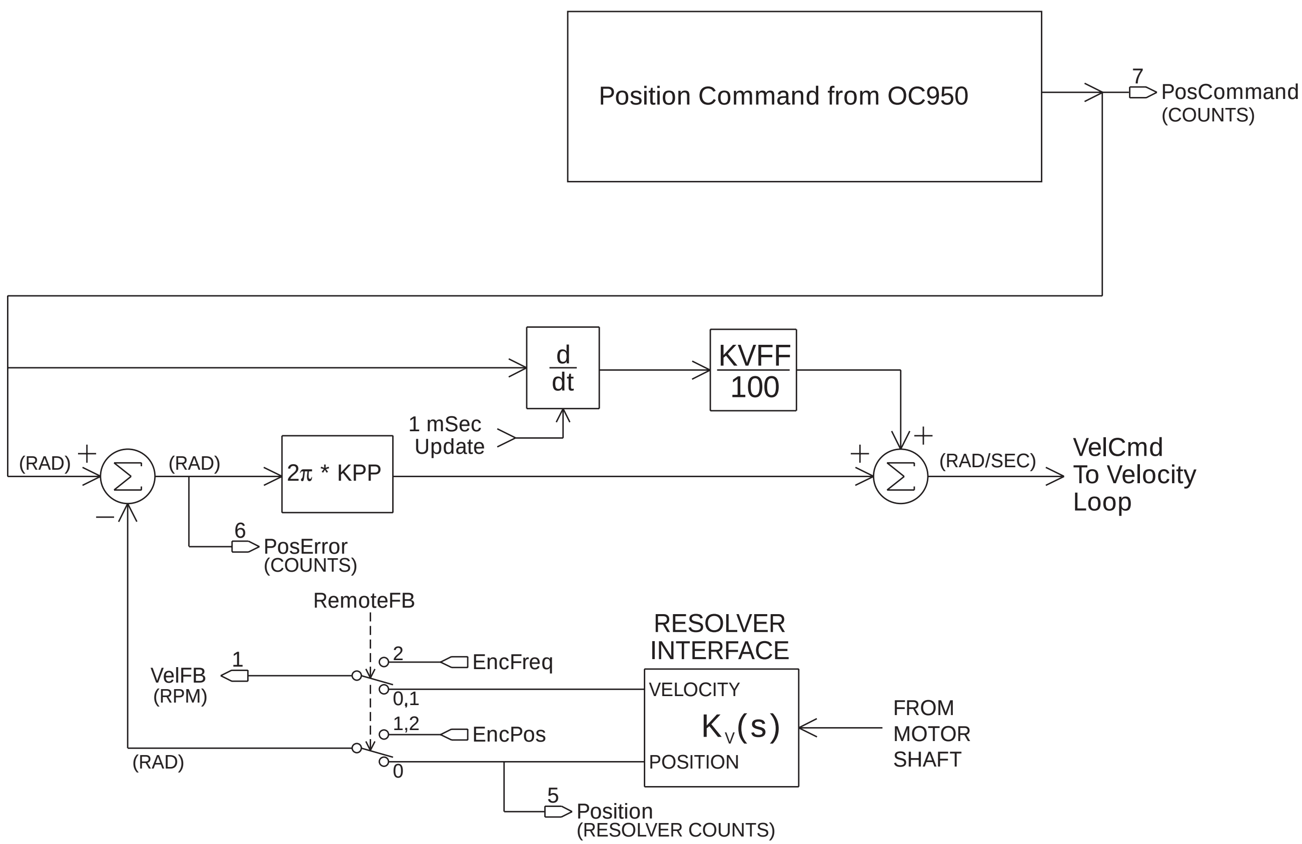
Figure 1

Figure 2



SC950 VELOCITY COMMAND
PROCESSING BLOCK DIAGRAM

Figure 3



SC950 POSITION CONTROL
BlkType = 2 BLOCK DIAGRAM

Appendix B
OC950 Specifications

OC950 J51 Serial Interface

Type	RS-232/RS485 (address selectable)
Baud Rate	19200 or 9600 (software selectable)
Parity	No parity
Data Word	10 bits: 1 start, 8 Data, 1 stop
Connector	9 Contact Female D-subminiature

OC950 J52 Programmable Inputs / Outputs

Input/Output Type & Quantity	21 Bi-Directional Channels
Logic Voltage	5 to 24 Volt, 40V Absolute Maximum
Input Channel	
Pull-up resistor	10K ohm to +12VDC
Low-to-High Logic Threshold	3.5 Volts maximum
High-to-Low Logic Threshold	1.0 Volts minimum
Hysteresis	0.5 Volts minimum, 1.5 Volts maximum
Filter RC Timeconstant	100 microseconds

Output Channel

Device	Open Drain DMOS Transistor
Sink Capability	100 milliamperes
Drain-to-Source - Impedance	5 ohms typical, 9.5 ohms maximum at 125° C
Connector	25 Contact Female D-subminiature

User +5V Power (OC950 J51-4, J52-25 and SC900 J4-25)

Output Voltage 5 Volts ± 5%

Output Current 200 milliamperes, Maximum
Note: *Output current is the combined current of all three connection points.*

OC950 J53 (Optional) PacLAN

Data Rate	2.5 MBaud
Termination Impedance	93 ohms
Cable Type	coaxial, RG62U
Maximum Bus Segment Length	305 meters
Maximum Stations/Segment	8
Connector	Standard BNC

NVRAM - Program Memory

Data Retention 10 Years

Index

!

950IDE	2-4
Compiling a Program	2-9
Configuring Your System	2-6
Creating a New Program	2-7
Exiting	2-14
Getting Around	2-5
Installing	2-4
On-Line Tuning	2-8
Opening a Program	2-14
Run Program	2-9
Saving a Program	2-14
Starting	2-4
Using Variables Window	2-11
Using Wizards	2-8
Viewing Variables	2-12
Working w/Breakpoints	2-10

A

ARF0	4-5
ARF1	4-5
Axis address	3-25
Switch S1	3-7

B

Backlash	4-6
Breakpoints	2-10
Continuing execution	2-11
Removing	2-10
Setting	2-10

C

CA950-IO	3-20
Configuring System	2-6
Applying AC Power	2-6
BaudRate	2-6
Serial Port	2-6
Connections	2-3
Diagram	2-3
Creating a New Program	2-7
Enabling Drive	2-8
Select Drive	2-7
Select Filename	2-7
Select Inertia Ratio	2-7
Select Motor Number	2-7
Current ripple	4-7

E

Enabling Drive	2-8

F

Faultcodes	5-1

H

Help	2-14

I

Interfaces and Connections 3-1
- Diagram 3-2
- OCIO J52 3-14
- PacLAN J53 3-24
- Serial Port J51 3-3

J

J51 3-3
- Diagram 3-4
- I/O Table 3-3
- Specifications B-1
- T10 Connections 3-5

J52 3-14
- +5VDC 3-16
- Connecting to Opto22 3-19
- I/O Table 3-14
- Inputs 3-16
- Outputs 3-16
- Schematic 3-15
- Specifications B-1

J53 3-24
- Active hubs 3-25
- Axis address 3-25
- Connections 3-24
- Specifications B-1

K

KPP 4-8

KVFF 4-9

KVI 4-7

KVP 4-3

L

Load inertia 4-3
- Problems with 4-4

M

Maintenance 5-1

N

NVRAM 1-2

O

OCIO 3-14
- I/O Table 3-14
- Inputs 3-16
- Outputs 3-16
- Schematic 3-15
- Specifications B-1

On-line Tuning 2-8

Opto22 Mounting Rack 3-19
- CA950-IO 3-20
- Inputs 3-20
- Outputs 3-21
- Power Supply 3-22
- Schematic 3-23

Overshoot 4-10

P

PacLAN 1-2, 3-24
- Active hubs 3-25
- Axis address 3-25
- Connections 3-24

Specifications B-1

Position Block A-5
  Digital Command A-5
Position Loop 4-8
  KPP 4-8
  KVFF 4-9
  Overshoot 4-10
Programmable Option Card 1-1
  Definition 1-1
  Hardware varieties 1-2

R

Resonance 4-5
Return procedure 5-13
RMA# 5-13
RS-232 3-11
  diagram 3-11
RS-485 3-12
  diagram 3-12

S

Serial Connection 3-3
Setting Up 2-1
  Hardware 2-1
  Serial Address 3-7
  Wiring connections 2-2
Specifications B-1
  J51 B-1
  J52 B-1
  J53 B-2
Status Displays 5-1
Switch S1 3-7
System Configuration 2-6

T

T10 Connections 3-5
Torque Block A-1
  Analog Command A-1
Transfer function 4-10
Troubleshooting 5-1
  Table 5-5
  RMA# 5-13
Tuning, Advanced 4-10

V

Velocity Block A-3
  Analog Command A-3
Velocity Loop 4-1
  Advanced Tuning 4-10
  ARF1 4-5
  Backlash 4-6
  Bandwidth 4-2
  Current ripple 4-7
  Example calculation 4-3
  KVP 4-3
  Load inertia 4-3
  Resonance 4-5

W

Warranty 1-2
Watch Window 2-12
  Adding variables to 2-13
  Opening 2-12
  Removing variables from 2-13
Wizards 2-8