

Sierra Radio Systems

# ***Sierra Radio Systems***

## Series 200 Control System Owners Manual

Version 2.3



High performance communications control systems

# **Table of Contents**

## ***Introduction***

## ***Hardware overview***

- Architecture
- Configuring and installing the hardware boards
- External connections

## ***Configuring the control system***

- Introduction
- Using DTMF
- Using a PC and the CONFIG program
- Basic configuration parameters
- Radio port configuration
- Setting audio levels
- Using macros and creating custom commands
- Saving the configuration to a file
- Downloading configuration to the control system
- Terminology and file system

## ***Installing new firmware***

## ***Control system operation***

- Status indication LEDs
- Command processor
- Unlock codes
- Site prefixing

## ***Command reference guide***

## ***System-wide Audio Level Setting***

## ***Final System Installation Checklist***

## ***Appendix***

- Glossary
- Receive qualification, squelch and transmit timing
- Telemetry tones and level control
- Software architecture and theory of operation

## Introduction

This manual provides all the basic information you need to know to get your control system installed, configured and on the air quickly. The control system is designed to work "out of the box" in a standard configuration.

In most installations, all that is required is making the physical connections to power and the radio equipment and setting a few basic configuration parameters such as call sign and unlock code (password) and you are on the air.

The control system can be tailored to your particular needs in many ways. This manual will give you a good overview of many of the parameters that can be changed to tailor your system to meet your specific needs. Each port can be customized for a particular behavior including repeaters, RF links, VOIP links, etc. Telemetry, command names, and many other behaviors can also be customized.

Additional application notes are available on the SRS support web site that provides detailed information on a variety of special topics.

### ☐ **Basic Checklist**

As you read through this document, from time to time you will see a check box. This is used to get your attention and indicates that this is something you really need to do to get your control system working. This is part of your "basic check list". Make sure you read these notes.

☐ You will also see open check boxes. These are for your to check them off as you go to make sure you follow all the important steps.

## Architecture Overview

The Series 200 control systems are very modular and scalable. The basic architecture includes a master CPU board, from 1 to 8 Radio Control Boards (RCBs), and a backplane to provide the interconnection between boards. Each RCB interfaces with an external device such as a repeater, RF link, VOIP link, or computer and provides a basic signaling interface that includes receive audio input, transmit audio output, COR (carrier present), CTCSS/DCS decode logic input, and PTT (transmit control) output, serial radio control I/O, and transmitter cooling fan output control.

The RCB can interface with many types of radio and computer equipment without modification. In some cases, additional signal processing is required. The Series 200 is designed to allow an additional signal interface board to be added to any of the RCB. Optional signaling boards include special boards for microwave radio equipment, E&M signaling, squelch detect, CTCSS tone generation, CTCSS detection, low pass filtering, high pass filtering, TTL to RS-232 conversion, etc. The optional signaling interface boards bolt on to the RCB to form a board pair or "module".

A control system may be as simple as a CPU board, a single RCB and a short backplane. A fully configured system may occupy an entire 19 inch card rack and contain a CPU, up to 8 RCBs each with its own special signaling interface board, optional system interface boards, and a large backplane and power supply board. In any configuration, the system uses exactly the same CPU and RCB boards so a small system can easily grow into a large system by simply adding more boards. Additionally, the software that runs on the smallest configuration is exactly the same as the software for the largest configuration.

The modular nature of the Series 200 control system make it not only easy to upgrade, but also very easy to maintain. Boards can be quickly replaced from the front of the card rack.

Very modular. Very scalable. Easy to maintain.

# System Installation Checklist

## ☐ **Basic Checklist Summary**

This is a reminder of the basic steps to get your control system configured and running.

### Hardware installation and external connections

- ☐ Configure all necessary jumpers on each board.
- ☐ Plug boards into the backplane in the correct slots.
- ☐ Connect the control system to an external 12 VDC power source.
- ☐ Connect the CPU's DB-9 connector to your PC.
- ☐ Connect each RCB's DB-9 connector to your external devices: radios (repeater, RF links, remote base radios), VOIP computers, etc.

### Software configuration

- ☐ Set basic configuration parameters
  - ☐ Call sign
  - ☐ Site prefix
  - ☐ Location ID
  - ☐ Unlock code
- ☐ Set radio port configuration
- ☐ Set audio levels
- ☐ Customize commands
- ☐ Save the configuration to a file
- ☐ Download configuration to the control system

## Configuring the CPU Board

Most of the boards in the control system have one or more jumpers that can be set to configure the board for various special purposes. The default configuration will be used in most cases. Refer to the SRS Hardware Reference manuals for more details on each board and the various jumper options.

Here are the default jumper settings for the CPU and RCB boards.

### **Basic Checklist**

Make sure all CPU board jumpers in properly set.

The CPU board executes the main control system functions including routing the audio, controlling the transmitter PTT lines, command decoding, and command response telemetry generation. The CPU board also provides a serial RS-232 connection to a PC for controller configuration, and 24 user defined I/O connections including digital inputs, analog inputs and buffered digital outputs.

## CPU Jumper Settings

### **JU1 - DVB Jumper**

The DVB jumper enables or disables audio from the DVB\_bus signal on the backplane. This signal is audio generated from the optional Digital Voice Board or DVB.

**"DIS"** In normal operation when not using the DVB, the jumper **MUST** be in the disabled or "DIS" position.

**"EN"**: When using the optional DVB in the system, JU1 **MUST** be in the enabled or "EN" position.

### **JU2 - Digital Pot Bypass Jumper**

**OUT:** Default. Set telemetry level under CPU control. This is the normal state for this jumper.

**IN:** Force telemetry pot to mid scale. This overrides the CPU level set for the master telemetry generator.

### **JU3 - ICD Power**

**OUT:** Default. For using Con 3, the 10 pin header connector for in-circuit programming.

**IN:** Install if using Con 5, the modular ICD jack for in-circuit programming.

**NOTE:** When using the ICD jack, you **MUST** cut the trace on the bottom of board to enable the ICD power.

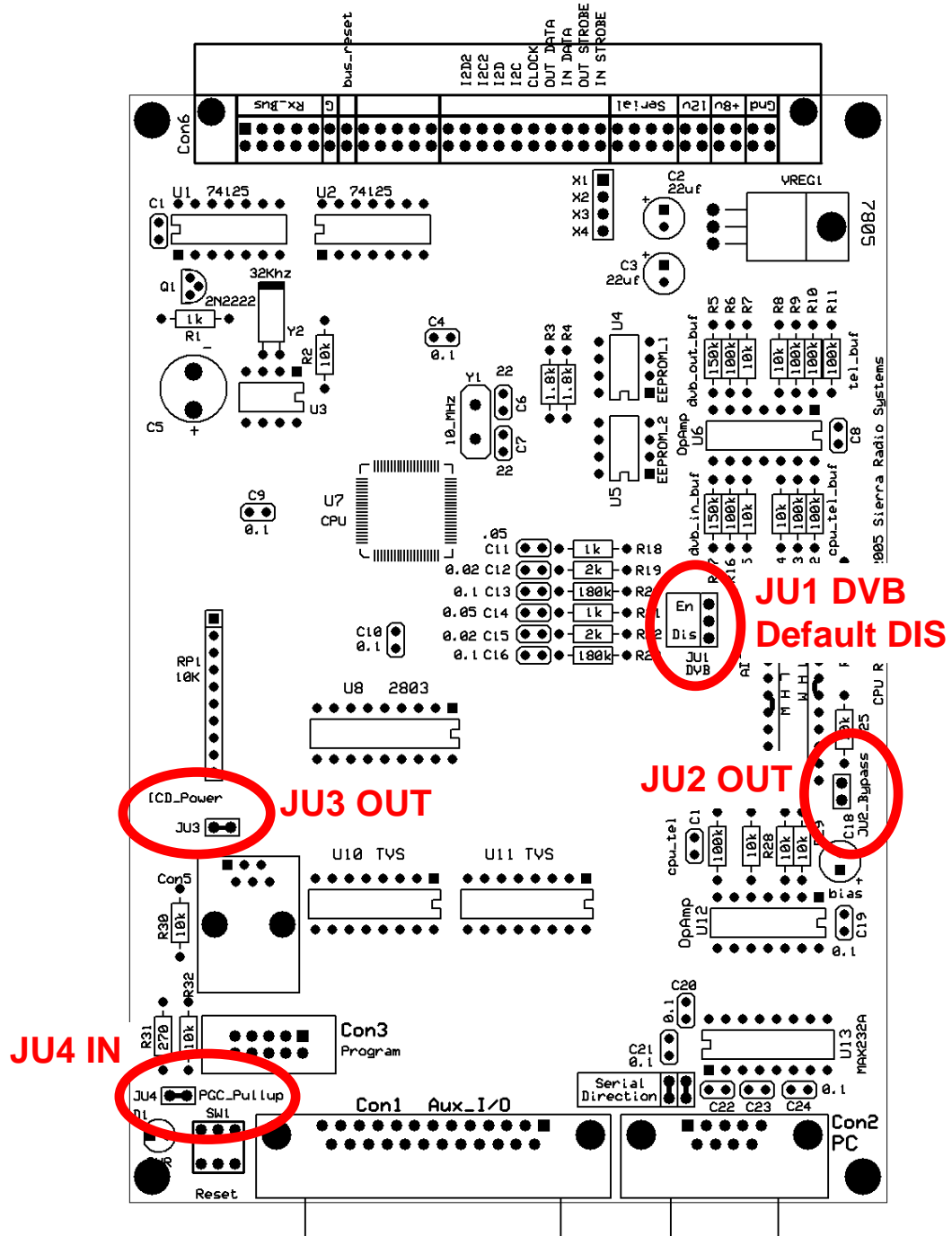
### **JU4 - PGC Pullup**

**IN:** Default. Install this jumper for using Con 3, the 10 pin header connector for in-circuit programming.

**OUT:** Remove the jumper when using Con 5, the ICD programming jack.

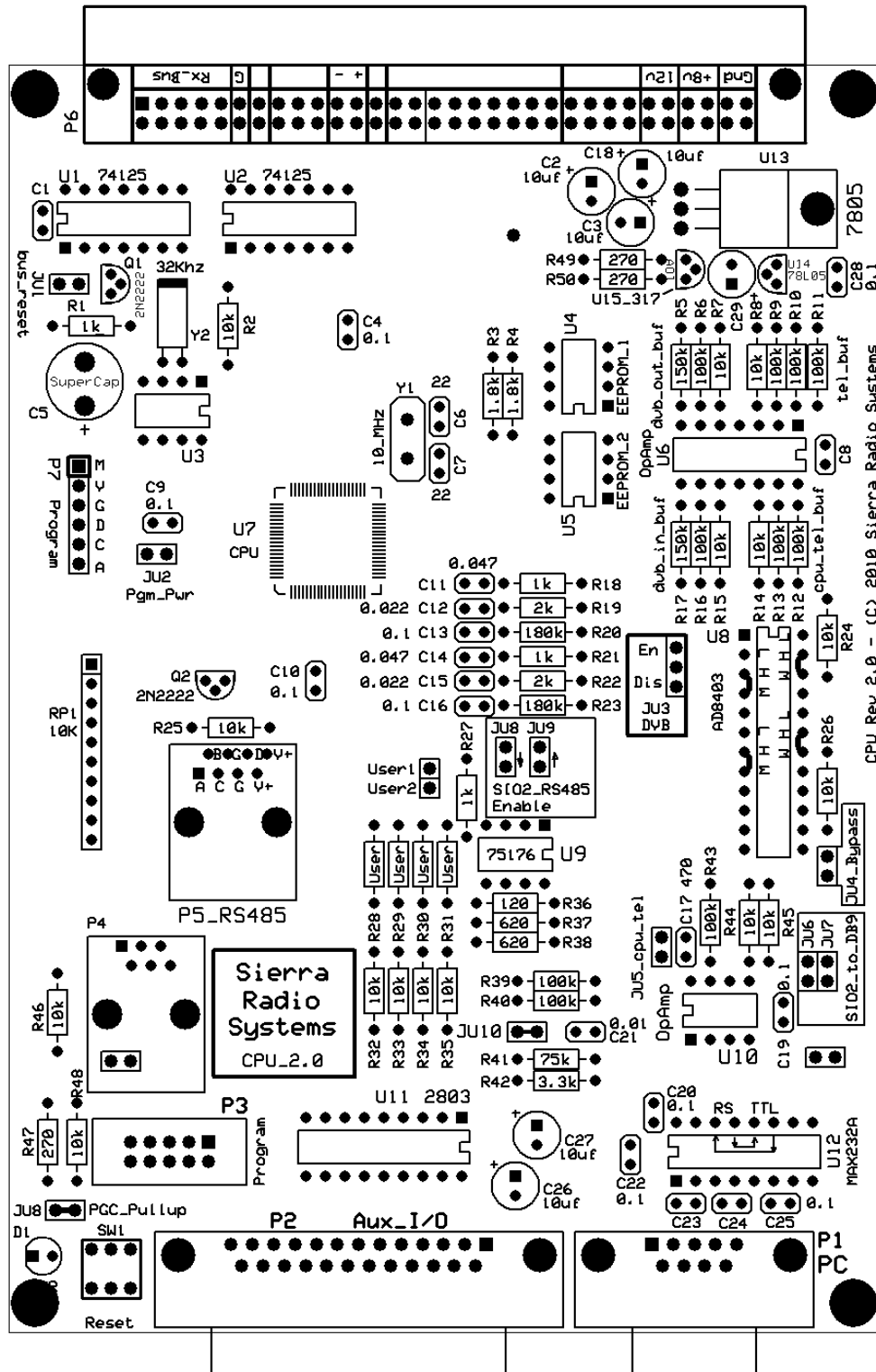
**NOTE:** When using the ICD jack, you **MUST** cut the trace on the bottom of board. Default CPU Jumper Configuration

## Version 1.0 CPU Board



All jumpers should be installed in the default positions indicated.

## Version 2.0 CPU Board





## Configuring the Radio Control Board – Version 1.0 and 1.1

The radio control board provides the audio interface, level set, audio mixing, local telemetry (key up and CW ID) generation, serial radio control interface, extra user programmable open collector outputs. One RCB is required for every radio or VOIP computer in the system. Each RCB is configured to have one of several "personalities" including a repeater, RF link, remotely controlled base station, or VOIP computer.

### ☐ **Basic Checklist**

Make sure all RCB board jumpers are properly set.

### ☐ ☐ **JU1 - RX Port Select**

You **MUST** have one (and only one) jumper installed in this block. It must correspond to the port you want to assign this board from 0 to 7.

No two boards can have the same rx port select value.

### ☐ ☐ **JU2 - CPU Address**

You **MUST** install jumpers in the address header to match the port assignment. This should be the same value as the jumper installed in the Rx Port Select information above.

<u>Port</u>	<u>Jumpers</u>			
	8	4	2	1
0	out	out	out	out
1	out	out	out	IN
2	out	out	IN	out
3	out	out	IN	IN
4	out	IN	out	out
5	out	IN	out	IN
6	out	IN	IN	out
7	out	IN	IN	IN

### ☐ ☐ **JU 3 - DTMF Filter Bypass**

**OUT: Default.** Takes the rx audio through a bandpass filter before it goes to the DTMF decoder.

**IN:** Bypass the DTMF filter and drive the audio directly into the DTMF decoder.

### ☐ ☐ **JU4 - Local (NO)**

**OUT: Default**

**IN:** Install the jumper to enable the open collector output of Q2 to drive the "local COR" input on the local CPU.

### ☐ ☐ **JU5 - PL\_Filter (NC)**

**OUT:** Remove jumper is using a companion signaling board like the microwave or PL encode / decode boards.

**IN: Default.** Install the jumper when operating the RCB without any signal conditioning

boards.

☐ ☐ **JU6 - DAC Pot Mid**

**OUT: Default.** Allows digital pots to be set by the local CPU.

**IN:** Sets all four digital pots to mid scale.

☐ ☐ **JU6 – Last Port Jumper (yes, there are two jumpers marked JU6)**

**OUT: Default.** Leave the jumper out for all RCB board except the last one.

**IN:** Insert a jumper in this position if this is the last RCB in the card rack. The Last Port jumper provides a termination on the high speed serial bus and will reduce noise on the bus.

**JU7 - ADEN**

**OUT: Default.**

Not used at this time.

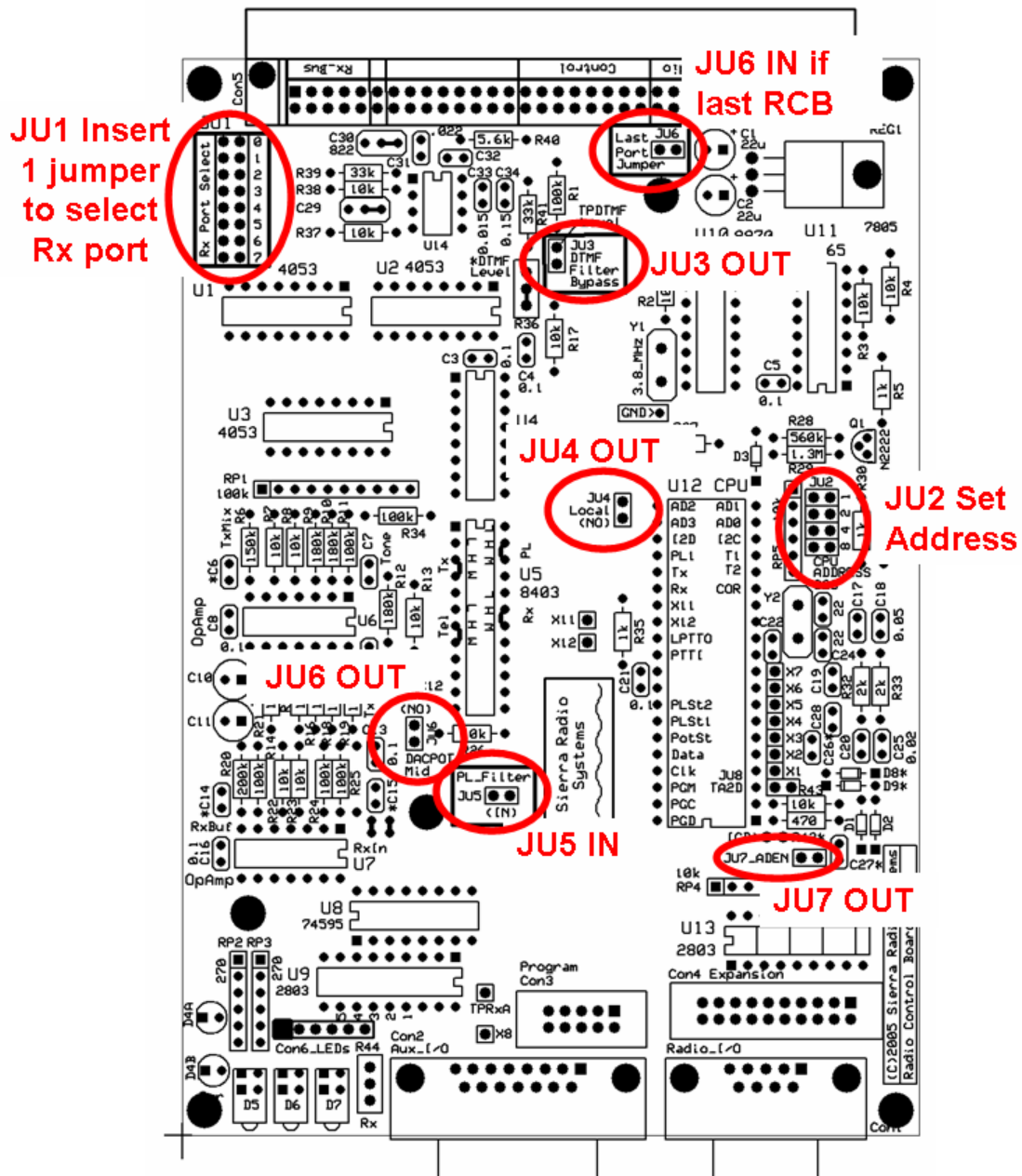
**JU8 - TA2D**

**OUT: Default**

Not used at this time.

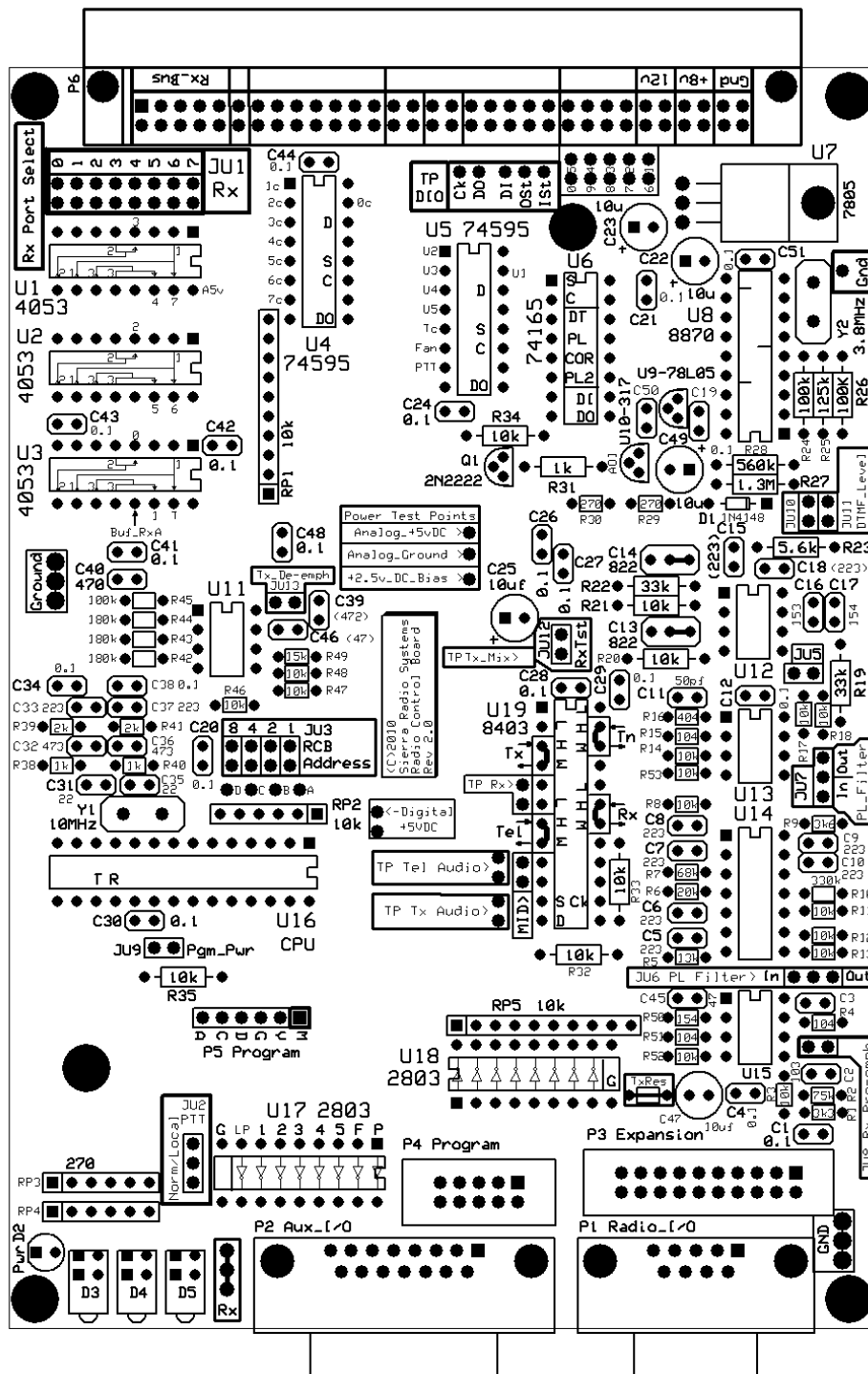
## Default RCB Jumper Configuration

### Version 1.0 Radio Control Board



\_\_\_\_\_ Jumper block \_\_\_\_\_ must be set with a single jumper to properly route the RX audio to the proper backplane bus signal.

# Version 2.0 Radio Control Board



## Hardware Orientation

### Power Supply Board

The power supply board provides a power switch and a power LED.

The power switch provides primary power to the CPU, RCB and accessory boards. The 12 VDC from the external power connector is routed to all boards in the card cage and is not switched.

The power LED indicates that power is being supplied to the rest of the control system.

### CPU Board

The CPU board provides a power LED, hardware reset switch, optional I/O connector (DB25) and the male DB9 RS-232 serial connection to a PC used for computer-based configuration and control.

To trigger a hardware reset, press the hardware reset button.

### Radio Control Board (RCB)

The RCB board provides a power LED, a set of 6 channel status LEDs, a female DB15 auxiliary I/O connector and a female DB9 radio I/O connector.

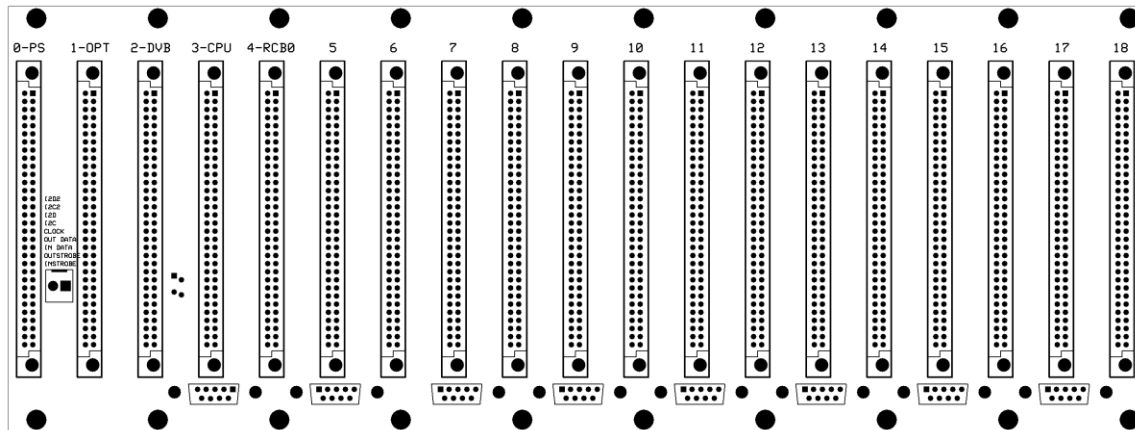
The channel status LEDs indicate COR (active Rx signal present), PTT (transmitter active), CTCSS/DCS decode, DTMF decode and two user programmable output bits.

The DB15 auxiliary connector provides many convenient functions including

## Configuring the Backplane

The system will come configured as ordered. You will need to plug the various boards into the correct slots in the backplane. If your system is already assembled as desired, you can skip this section.

The Power Supply, CPU and the first Radio Control Board (RCB) must be placed in **specific** slots. After the first RCB, all additional RCBs must be installed next to each other. There can be no gaps between the CPU and any of the RCBs. When optional microwave or CTCSS signaling boards are used, these boards are paired with each RCB and are installed in the slot to the right of the RCB.



The basic control system consists of the card rack with backplane, a power supply board, CPU board and one Radio Control Board (RCB) for each receiver/transmitter in the system. All controllers have the first 5 slots assigned as follows...

Slot	Assignment
0	Power supply
1	Reserved for future expansion
2	Reserved for future expansion
3	CPU
4	Radio control board - port 0

Radio Control Boards are installed starting in slot 4. Every radio requires a radio control board. Radio Control Boards must be plugged into the backplane next to each other and are named port 0 through port 7. There can be NO gaps between boards after port 0.

A basic 8 port controller will have the following configuration...

Typical 8 port controller

Slot	Assignment
0	Power supply
1	Reserved for future expansion
2	Reserved for future expansion
3	CPU
4	Radio control board - port 0
5	Radio control board - port 1
6	Radio control board - port 2
7	Radio control board - port 3
8	Radio control board - port 4
9	Radio control board - port 5
10	Radio control board - port 6
11	Radio control board - port 7
12-18	Empty

Each port is assigned a "personality" which defines the basic behavior of that RCB. Personalities can be either a repeater, RF link, VOIP link or remotely controlled base station. The different personalities of a system can be assigned to the ports in any desired order. The user level and many of the configuration commands refer to specific ports by a number, often referred to as the link number because most of them will be links. Configuration of the system is simpler if the number by which a port will be referred is same as the port number. A typical configuration would have a primary repeater in port 0, RF links in the next set of ports, VOID links, remote bases and then any additional repeaters, For example, 2 repeaters, 4 links, and 2 remote bases would be configured as...

Slot	Assignment
0	Power supply
1	Reserved for future expansion
2	Reserved for future expansion
3	CPU
4	Radio control board - port 0 – Repeater 1
5	Radio control board - port 1 – Link 1
6	Radio control board - port 2 – Link 2
7	Radio control board - port 3 – Link 3
8	Radio control board - port 4 – Link 4
9	Radio control board - port 5 – Remote base 1, link 5
10	Radio control board - port 6 – Remote base 2, link 6
11	Radio control board - port 7 – Repeater 2, Radio 7
12-18	Empty

**Special Note:** There must not be any empty slots between radio control boards. If there is a gap between boards, all ports up to the gap will work fine but all subsequent ports will not function properly. This breaks the high speed serial control bus which communicates the state of COR, PL decode, PTT, fan control, user outputs and DTMF signals to the master CPU.

### Using Optional Signaling Boards

If optional signaling boards such as the microwave radio board or PL encode / decode boards are used, they are placed in the slot immediately to the right of the radio control board. A typical 6 port microwave hub would be configured as...

Slot	Assignment
0	Power supply
1	Reserved for future expansion
2	Reserved for future expansion
3	CPU
4	Radio control board - port 0 - Microwave radio 1
5	Microwave E&M signaling interface board
6	Radio control board - port 1 - Microwave radio 2
7	Microwave E&M signaling interface board
8	Radio control board - port 2 - Microwave radio 3
9	Microwave E&M signaling interface board
10	Radio control board - port 3 - Microwave radio 4
11	Microwave E&M signaling interface board
12	Radio control board - port 4 - Microwave radio 5
13	Microwave E&M signaling interface board
14	Radio control board - port 5 - Microwave radio 6
15	Microwave E&M signaling interface board
16-18	Empty

Any combination of Radio Control Boards with or without optional signaling boards may be used. For example, a site with 1 repeater, 2 microwave radios, 2 narrowband RF links, one with PL encode/decode, one VOIP interconnect to the internet and one remote base radio.

Slot	Assignment
0	Power supply
1	Reserved for future expansion
2	Reserved for future expansion
3	CPU
4	Radio control board - port 0 - Repeater
5	Radio control board - port 1 - Microwave radio, link 1
6	Microwave E&M signaling interface board
7	Radio control board - port 2 - Microwave radio, link 2
8	Microwave E&M signaling interface board
9	Radio control board - port 3 - RF link 3
10	PL encode / decode signaling interface board for RF link 1
11	Radio control board - port 4 - RF link 4
12	Radio control board - port 5 - VOIP internet interconnect board
13	Radio control board - port 6 - remote base radio
14-18	Empty



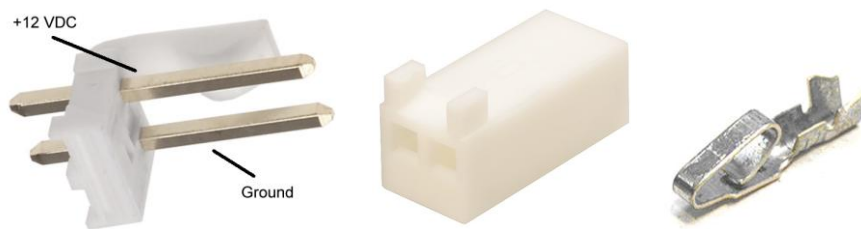
## External Connections

### Backplane DC Power Connection

#### ☐ **Basic Checklist**

Make sure you connect the control system to an external 12 VDC power source.

The control system typically uses +12 to +14 VDC and will operate down to 10 VDC with no problems. The external DC power is supplied through a 2 pin connector mounted on the back of the control system.



The alignment tab is on the top of the connector. As you look at the back of the backplane, the left pin is +12 VDC in and the right pin is ground.

Make sure the connector is plugged in correctly. The control system is internally fused on the power supply board just in case something gets shorted out. The fuse value should be 3 amps.

## CPU Board Connections

There are three types of external connections on the CPU board, the configuration serial port connector, the general purpose I/O connector and the in-circuit programming connectors.

### ☐ **Basic Checklist**

Make sure you connect the CPU's DB-9 connector to your PC so you can use the CONFIG program to configure your control system.

#### CPU Board Serial Port Connector

This connection is a standard RS-232 ASCII interface between the control system and your computer. It is used to configure the control system under computer control.

Connector pin outs...

- Pin 2 - Rx data in
- Pin 3 - Tx data out
- Pin 5 - Ground

The configuration serial port is the male DB-9 connector on the front edge of the board. This connection is not used in normal operation. When using the optional PC-based configuration



software, the PC's serial port is plugged in this connector.

#### PC Configuration Cable

This cable connects your computer to the main CPU board for upload and downloading the control systems configuration. The programming cable is a straight through cable. Connections required are pin 2 to 2, 3 to 3 and 5 to 5. The crossover of txd / rxd is done on the CPU board. Note that most female to female cable available in computer stores swap pins 2 and 3, so you may need to make your own cable.

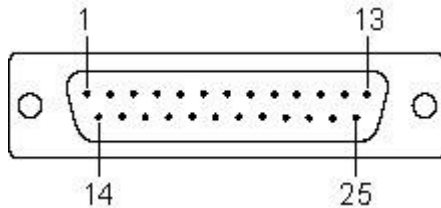
#### Con\_3 - In-Circuit Programming Connector

The in-circuit programming connector, Con\_3, is used to download new versions of firmware to the main CPU. This is a 10 pin header connector designed to be used for in-circuit programming with an ME Labs in-circuit programmer.

## General Purpose I/O Connector – CPU version 1.0

The GPIO connector provides 24 I/O signals that can be used for a variety of purposes. The signals include 8 TTL level digital inputs with internal logic pullups, 8 open collector buffered outputs, 8 analog to digital converter inputs and ground.

All 24 I/O signals are protected from over voltage conditions using transient voltage suppression devices.



<u>Pin</u>	<u>Use</u>	<u>Pin</u>	<u>Use</u>
1	Analog input	14	Analog input
2	Analog input	15	Analog input
3	Analog input	16	Analog input
4	Analog input	17	Analog input
5	Digital output	18	Digital output
6	Digital output	19	Digital output
7	Digital output	20	Digital output
8	Digital output	21	Digital output
9	Digital input	22	Digital input
10	Digital input	23	Digital input
11	Digital input	24	Digital input
12	Digital input	25	Ground
13	Digital input		

Digital inputs are pulled up with a 10k resistor and must be either ground or +5 VDC.

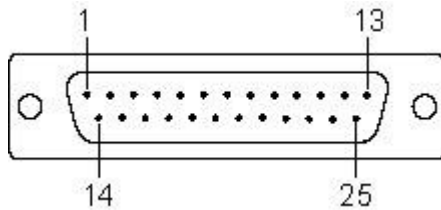
Analog inputs must be between 0 and +5V.

Digital outputs are buffered open collector outputs that can sink up to 500 ma.

## General Purpose I/O Connector – CPU version 2.0

The GPIO connector provides 24 I/O signals that can be used for a variety of purposes. The signals include 8 TTL level digital inputs with internal logic pullups, 8 open collector buffered outputs, 8 analog to digital converter inputs and ground.

All 24 I/O signals are protected from over voltage conditions using transient voltage suppression devices.



<u>Pin</u>	<u>Use</u>	<u>Pin</u>	<u>Use</u>
1	Aux audio input 1	14	Aux audio input 2
2	RS-485 B	15	RS-485 A
3	Analog input 4	16	Analog input 3
4	Analog input 2	17	Analog input 1
5	Digital output 1	18	Digital output 2
6	Digital output 3	19	Digital output 4
7	Digital output 5	20	Digital output 6
8	Digital output 7	21	Digital output 8
9	Digital input 8	22	Digital input 7
10	Digital input 6	23	Digital input 5
11	Digital input 4	24	Digital input 3
12	Digital input 2	25	Ground
13	Digital input 1		

Digital inputs are pulled up with a 10k resistor and must be either ground or +5 VDC.

Analog inputs must be between 0 and +5V. If additional “User” resistors R28, R29, R30 and R31 are added, the voltage input can be higher. These resistors form a voltage divider. The series input resistor is 10k. If the “User” resistor is 2k, that will scale the input by a factor of 5. In this case the input can accept a voltage range from 0 to 25 volts.

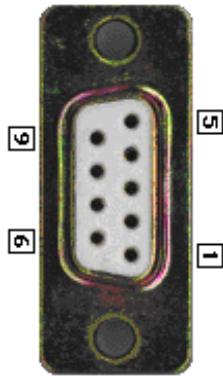
Digital outputs are buffered open collector outputs that can sink up to 500 ma.

## Radio Control Board (RCB) Connections

### ☐ **Basic Checklist**

Make sure you connect each radio, computer or other external device to the control system RCBs using the DB9 connector on the front of the RCB.

DB-9 radio connections



The control system has a female DB-9 connector which is used to connect to an external device including a repeater, link radio, remote base radio or computer system. The cable should have a male connector.

This picture is the DB-9 connector as seen on the front of the control system.

- |              |  |
|--------------|--|
| <b>Pin 1</b> | High impedance Rx audio input from receiver. Input can range from 0 to 2v peak-to-peak. Anything around 1 v p/p works great. Rx audio may be unsquelched.  |
| <b>Pin 2</b> | COR ("Carrier Operated Relay"), an active low signal. When the carrier is present, the COR pin must be pulled to ground.   |
| <b>Pin 3</b> | Ground   |
| <b>Pin 4</b> | PTT ("Push To Talk"), an active low signal. This is an open collector output that pulls to ground when the controller wants to key the transmitter. The output can sink a maximum of 250 ma.     |
| <b>Pin 5</b> | Low impedance audio output to the transmitter. Nominal 1 v p/p output. Can be adjusted from 0 to about 4v p/p.   |
| <b>Pin 6</b> | PL decode input active low. Pull this input pin low to indicate the presence of the proper PL or DPL tone being decoded.   |
| <b>Pin 7</b> | Serial data input. TTL level ASCII input used with serial programmable remote base radios.   |
| <b>Pin 8</b> | Serial data output. TTL level ASCII input used with serial programmable remote base radios.  |
| <b>Pin 9</b> | Fan control output. This is an open collector output which is pulled to ground to actuate a relay that will turn on the cooling fan on the transmitter. The output can sink a maximum of 250 ma. |

**Radio control cable.**

The radio control cable connects the control system to the individual radios. The cable is a 9 pin, shielded cable with all 9 connections wired straight through, with pin 1 to 1, 2 to 2, etc.

**In-circuit programming connector**

This is a 10 pin, dual row header connector. This connection is not used in normal operation. This is the connection where the in-circuit programmer is plugged in when downloading new versions of firmware to the RCB's local CPU.

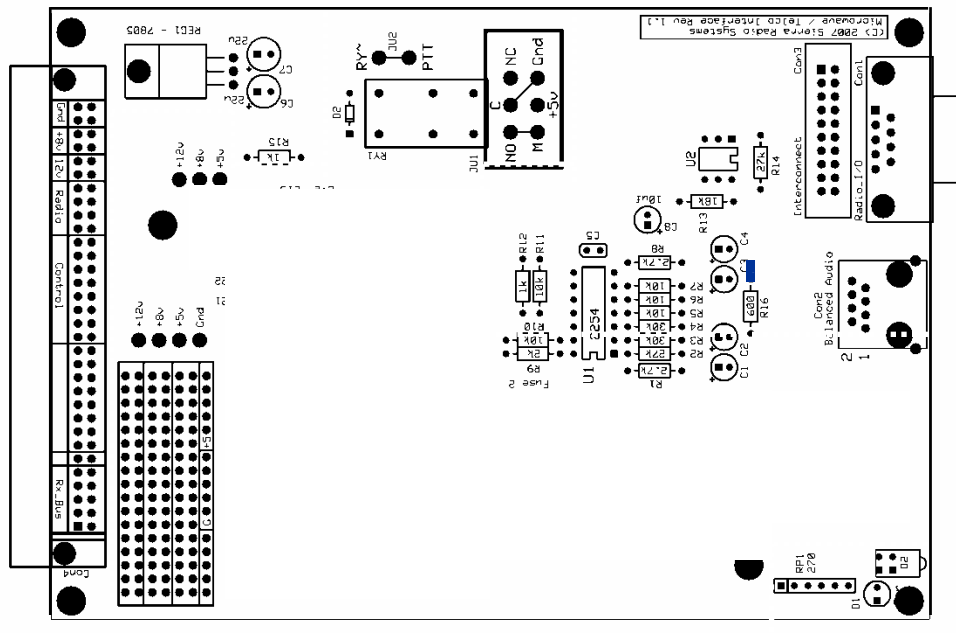
**RCB Accessory Connector "Aux\_IO"**

This is a female DB-15 which provides various optional connections for external device control, testing and alternate programming connections for the local CPU on the radio control board.

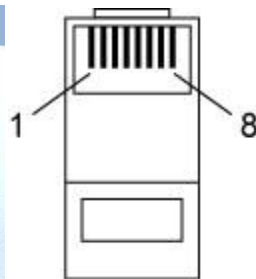
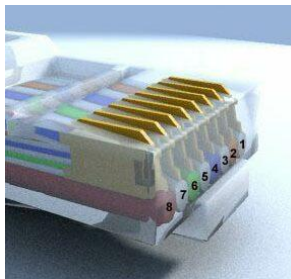
**Typical cable for external device control**

15	Open collector user output #1
8	Open collector user output #2
7	Open collector user output #3
14	Open collector user output #4
6	Open collector user output #5
11	Open collector PTT Output
12	Open collector Tx cooling fan output
13	Ground

## Microwave Radio Interface Board



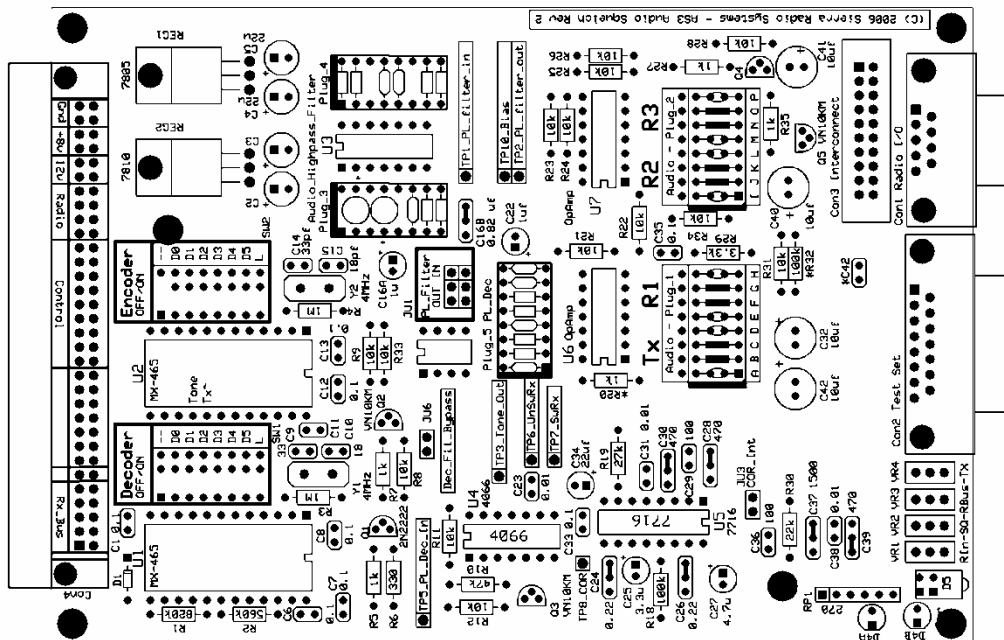
This board can be configured to drive a microwave radio that requires balanced audio and E & M signaling.



### Modular jack pin outs

Pin	Use
1	Balanced Tx audio out (600 Ohms)
2	Balanced Tx audio out (600 Ohms)
3	PTT "M" signaling
4	Ground
5	Negative voltage input for PTT keying circuit (nomially -24v)
6	COR input, active low
7	Balanced Rx audio input(600 Ohms)
8	Balanced Rx audio input(600 Ohms)

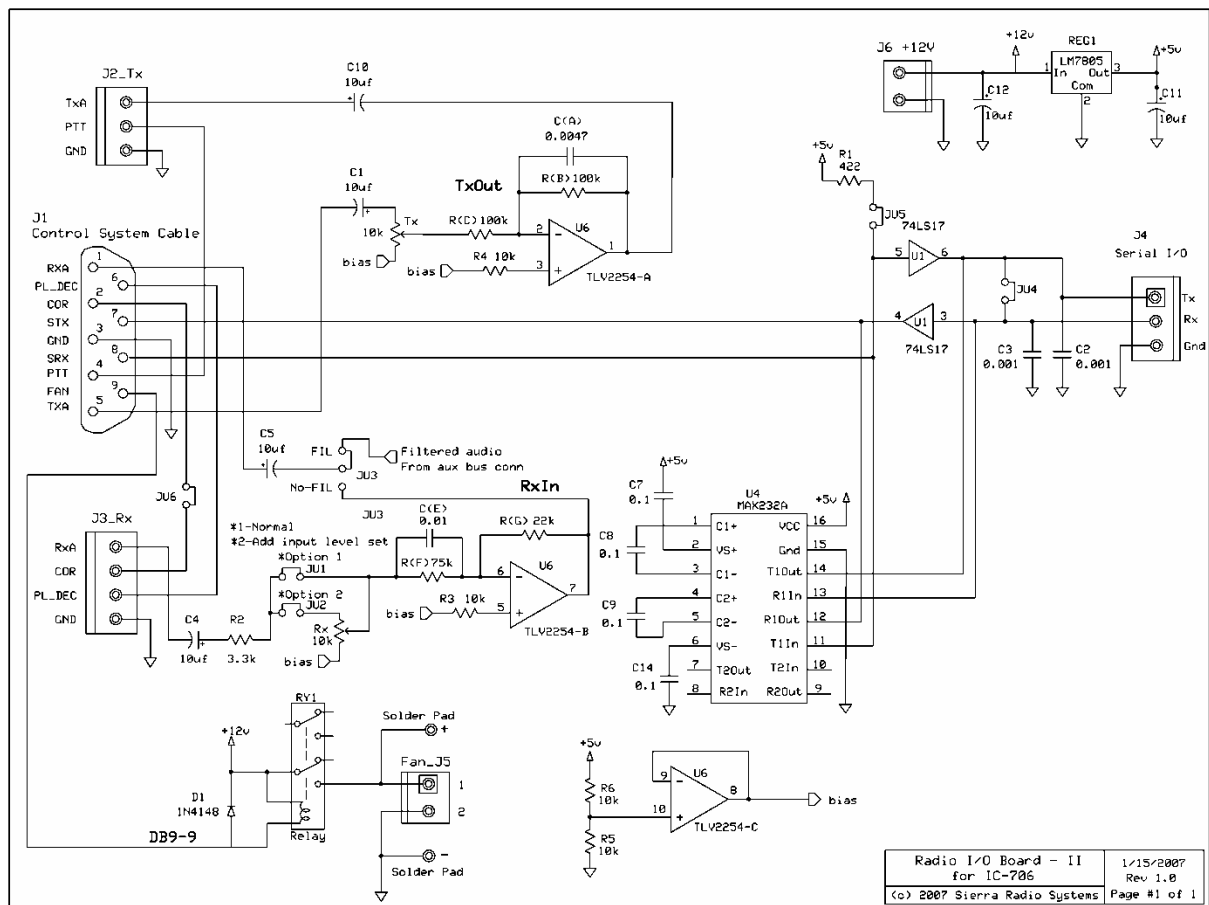
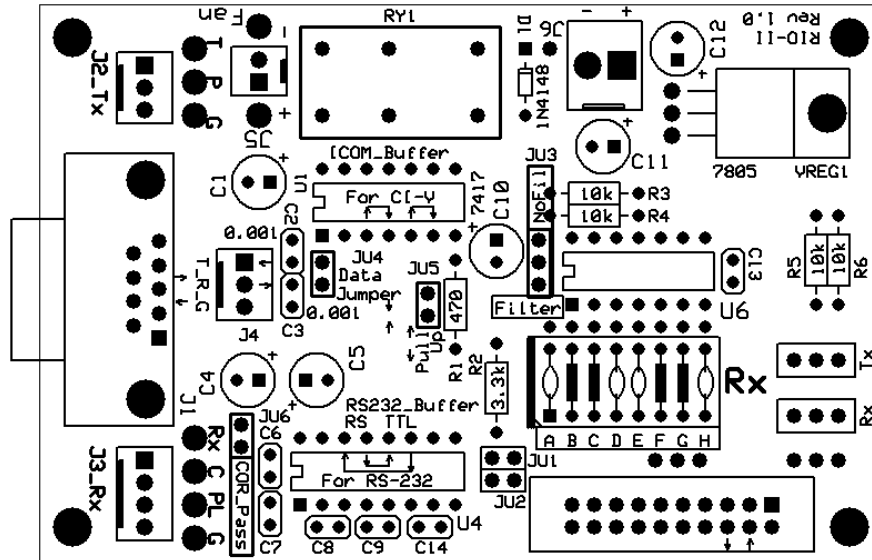
## AS-3 Audio Squelch Board



- Pin 1** High impedance Rx audio input from receiver. Input can range from 0 to 2v peak-to-peak. Anything around 1 v p/p works great. Rx audio may be unsquelched.
- Pin 2** COR ("Carrier Operated Relay"), an active low signal. When the carrier is present, the COR pin must be pulled to ground.
- Pin 3** Ground
- Pin 4** PTT ("Push To Talk"), an active low signal. This is an open collector output that pulls to ground when the controller wants to key the transmitter. The output can sink a maximum of 250 ma.
- Pin 5** Low impedance audio output to the transmitter. Nominal 1 v p/p output. Can be adjusted from 0 to about 4v p/p.
- Pin 6** PL decode input active low. Pull this input pin low to indicate the presence of the proper PL or DPL tone being decoded.
- Pin 7** Serial data input. TTL level ASCII input used with serial programmable remote base radios.
- Pin 8** Serial data output. TTL level ASCII input used with serial programmable remote base radios.
- Pin 9** Fan control output. This is an open collector output which is pulled to ground to actuate a relay that will turn on the cooling fan on the transmitter. The output can sink a maximum of 250 ma.



# Radio I/O – II (RIO-II) Board



Radio I/O Board – II for IC-706	1/15/2007 Rev 1.0
© 2007 Sierra Radio Systems	Page #1 of 1

## Configuring the Control System

# Control System Configuration Software

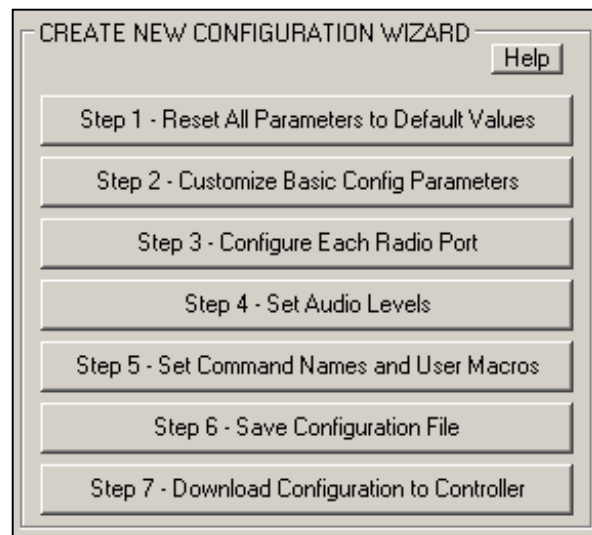
## Overview

Sierra Radio Systems supplies a software configuration program called “config” that allows the system administrator to configure the control system through a Windows-based PC. Config can be used in many different ways to configure a Sierra Radio Systems control system. Features include a graphic user interface to enter and modify control system parameters, run files control commands called scripts, reading configuration files into the program and writing the configuration back out to a file, interactively set audio levels, and more. The config program can be used “offline” to create and edit a configuration without a control system. Config uses the serial port on the master CPU board to transfer information to and from the control system. Config runs on a standard Windows PC that supports a serial port. A USB to serial adapter can be used if no serial port is available on the PC.

If you want a detailed understanding of the config program and data files, skip forward to section II. If you just want to jump right in and configure your control system then here we go...

## Section I - Setting up a New Configuration

The built in wizard helps you walk through the steps of creating a new configuration.



As you can see from the wizard the steps are...

1. Reset all parameters
2. Customize basic parameters
3. Configure each radio port
4. Set audio levels
5. Set command names and macros
6. Save configuration file
7. Download configuration to the controller

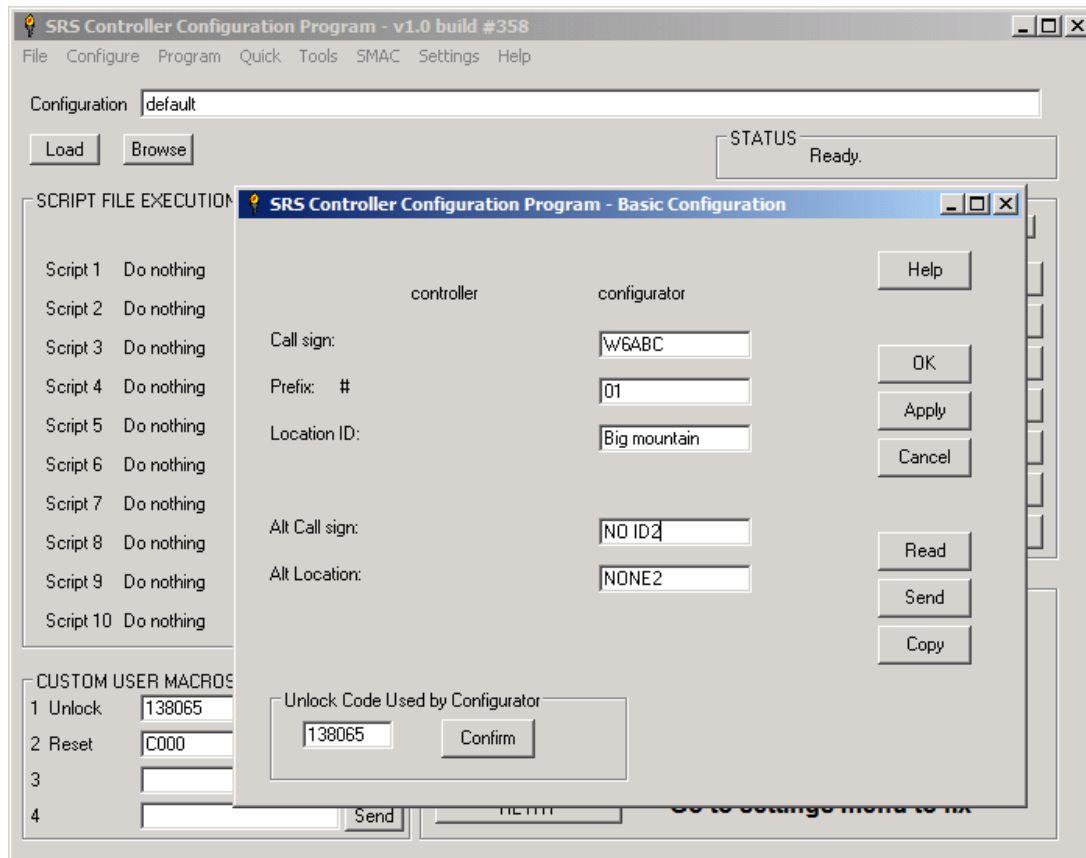
The steps are simple and in some cases some steps can even be skipped. Here is a description of what each step does...

## 1. Reset all parameters

This will set all the configuration parameters in the config program to their normal default values. Then the configuration file “default.cfg” will be loaded. This is the starting point for your new controller configuration.

## 2. Customize basic parameters

At a minimum, the controller should be personalized with its call sign, site prefix, location ID and unlock codes. For testing purposes, the defaults are fine but systems typically need to have their own unique call sign. Also for security reasons, the unlock code should be changed to something only the site administrator knows. If there are multiple control systems in a network, the site prefix and location ID should be changed to something unique to avoid confusion and proper network operation.



Call signs and location IDs can be up to 20 characters.

The site prefix should be 1, 2 or three unique digits. All site prefix codes are # followed by the parameter that you enter. If you enter 123, then your prefix will be #123.

Unlock codes should be something that only system administrators know. The minimum requirement is to specify unlock codes 1 and 2.

### 3. Configure Each Radio Port

Each port is assigned a personality depending on the type of equipment and desired behavior. Port types include repeaters, link, VIOP gateways, and remote bases.

The Port Configuration dialog box displays a table with 8 rows, each representing a radio port. The columns are: Port, Enabled checkbox, Description, Port type dropdown, Group dropdown, Access dropdown, Remote type dropdown, and Default checkbox. An OK button is located in the top right corner.

Port	Enabled	Description	Port type	Group	Access	Remote type	Default
Port 0	<input checked="" type="checkbox"/>	Repeater	1) Repeater	1) Group 0	2) PL & Carrier	1) None	<input checked="" type="checkbox"/> On link
Port 1	<input checked="" type="checkbox"/>	Link 1 North	2) RF Link	1) Group 0	2) PL & Carrier	1) None	<input checked="" type="checkbox"/> On link
Port 2	<input checked="" type="checkbox"/>	Link 2 East	2) RF Link	1) Group 0	2) PL & Carrier	1) None	<input checked="" type="checkbox"/> On link
Port 3	<input checked="" type="checkbox"/>	Link 3 South	3) VOIP Link	1) Group 0	2) PL & Carrier	1) None	<input checked="" type="checkbox"/> On link
Port 4	<input checked="" type="checkbox"/>	Link 4 West	4) Remote base	1) Group 0	2) PL & Carrier	1) None	<input checked="" type="checkbox"/> On link
Port 5	<input checked="" type="checkbox"/>	Port 5	4) Remote base	1) Group 0	2) PL & Carrier	1) None	<input checked="" type="checkbox"/> On link
Port 6	<input checked="" type="checkbox"/>	Port 6	1) Repeater	2) Group 1	2) PL & Carrier	1) None	<input checked="" type="checkbox"/> On link
Port 7	<input checked="" type="checkbox"/>	Port 7	4) Remote base	2) Group 1	2) PL & Carrier	1) None	<input checked="" type="checkbox"/> On link

#### Differences between port types

The table below describes the differences in behavior between ports configured as repeaters, links, irlp nodes, and remote bases. Some of these are just the default behavior that is created when the port type is defined, and can be changed, either by port type or by individual port. Others (such as ability to accept commands) are built into the programmed behavior of the port type.

	Repeater/ local	Link	IRLP	Remote base
Retransmits input on output	x			
Accepts commands preceded by *	x			
Accepts commands preceded by #prefix		x		
Transmits ID	"polite" algorithm	"link ID 2" algorithm		
Default CW ID frequency (Hz)	1064	1064		
Default CW ID speed (WPM)	20	24		
Default ID level (0..255)	80	40		
Default transmit carrier off delay (s)	4.0	4.0 0.1 with link delay off	0.1	0.1
Default COR/PL recognition delay (ms)	50	50	0	50
Default COR/PL drop recognition delay (ms)	150	5	0	150
Default PL status	PL on	PL on	PL off	PL off

### Repeater port type

Used for normal repeater operation. Loops repeater receiver audio to transmitter, uses transmitter carrier delay, mutes all DTMF audio to the transmitter but passes DTMF to the links, etc.

### Link port type

Used for RF or "private" internet VOIP point to point network links. Does not loop audio, passes all DTMF tones, has no carrier delay unless the signal comes from a repeater input, etc.

### VOIP port type

Used for IRLP, EchoLink and other "public" VOIP systems. Does not loop audio, does not pass DTMF or any locally generated telemetry, turns off carrier delay by default, etc.

### Remote base port type

Used for remotely controlled base stations including Icom, Yaesu, Kenwood serial programmed radios, BCD and channelized radios. Does not loop audio, does not pass DTMF or any locally generated telemetry, carrier delay turned off by default, etc

If the port type is "Remote Base" then the remote base type should be selected for the controller to generate the correct frequency and mode control data sent to the remote base radio.

### Groups

The control system can support up to 3 independent groups of ports. The default is for all ports to operate together as group 0. Additional groups can be created by assigning group #1 or #2 to

a port. Each group will act as if its members are the only radios in the system. All groups must use the same command set and will generate the same type of telemetry.

**Access**

This defines the combination of carrier detect (COR) and CTCSS/DCS signaling required for a valid signal present indication from each receiver.

**Enabled**

When checked, this makes that specific RCB (radio control board) available to the system. When unchecked, it is logically disconnected as if the board was pulled from the card cage.

**Default on link**

When checked, this port will default to be connected to the other ports in its group when the system is reset or a site normal command is executed. When unchecked, this port will be defaulted in the disconnected mode.

### 3. Set Audio Levels

Each radio control board has 4 digitally controlled audio level set potentiometers. The control the receiver and transmitter audio levels, telemetry level generated by the RCB that is sent only to that specific transmitter and an auxiliary audio input level. The local telemetry generator per transmitter is used for dial tone, busy signal, function complete, CW ID and other functions. The auxiliary audio input is typically used for sub-audible CTCSS and other externally generated signaling tones.

Audio level set dialog box

SRS Controller Configuration Program - Set Audio Levels

LEVEL SET

☒ Rx ☐ Tx ☐ Tel ☐ Tone

OK

PORT

☒ Port 0 - Repeater  
☐ Port 1 - Link 1 North  
☐ Port 2 - Link 2 East  
☐ Port 3 - Link 3 South  
☐ Port 4 - Link 4 West  
☐ Port 5 - Port 5  
☐ Port 6 - Port 6  
☐ Port 7 - Port 7  
  
☐ CPU

127	127	127	127
127	127	127	127
127	127	127	127
127	127	127	127
127	127	127	127
127	127	127	127
127	127	127	127
127	127	127	127

Up 1 Max Up 10

Mid

Down 1 Min Down 10

- DVB Telemetry -

127 127 127 127

Set all pots to mid scale

change the pot level.

#### Interactive, real time audio level setting from the computer.

This is an interactive dialog box. Most commands and parameters entered on most screens are not sent to the control system until the “download configuration to control system” is performed. In the case of the audio level set screen, as you change audio level values, they are sent real time to the control system. This allows you to set all audio levels in the control system while connected to actual radio and test equipment. This is the equivalent of adjusting up to 36 little audio pots but without having to bring your little green screwdriver.

Each audio level can be set between 0 and 255. 0 is minimum (“Min”) which shuts off the audio, 127 is Mid scale and 255 is maximum (“Max”) audio. Audio is adjusted by pressing one of the buttons on the right side of the screen. It Up 10 and Down 10 lets you move quickly across the range of the pot while Up 1 / Down 1 let you fine tune your levels.

### 4. Set Command Names and Macros

The control system comes with several dozen built in commands. In the simplest use mode, all you need to do is select your own personal unlock code and use all the generic built in commands and



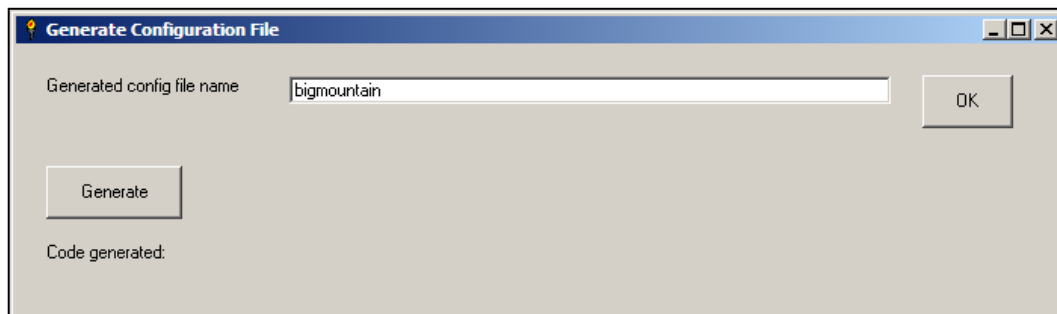
everything will work fine. This relies on the unique unlock code for security, which for most situations may be good enough. It is often desirable to have some or all of the user commands assigned unique “command names”. This is accomplished by using the user command macro features of the control system. Up to 250 user macros can be created. A macro can be a simple mapping of a custom command name to a single built in controller command. For example the built in command C310 will send the CW ID for the system. If you wanted to have your own command 123 send a CW ID you would create a macro with the name 123 and the macro string contents would be C310.

Macros allow you to string several commands together. You can create a command that will perform multiple actions with a single command name. For example you can create a command 234 which will execute the internal commands C3311 C3312 C3313 C3304 C351 which will turn on links 1, 2 and 3, turn off link 4 and set the repeater in CTCSS mode. Macros can even contain other macros.

**Special note: If you have any special script files that need to be run to finalize your configuration, this is the time to do it.**

## 5. Save Configuration File

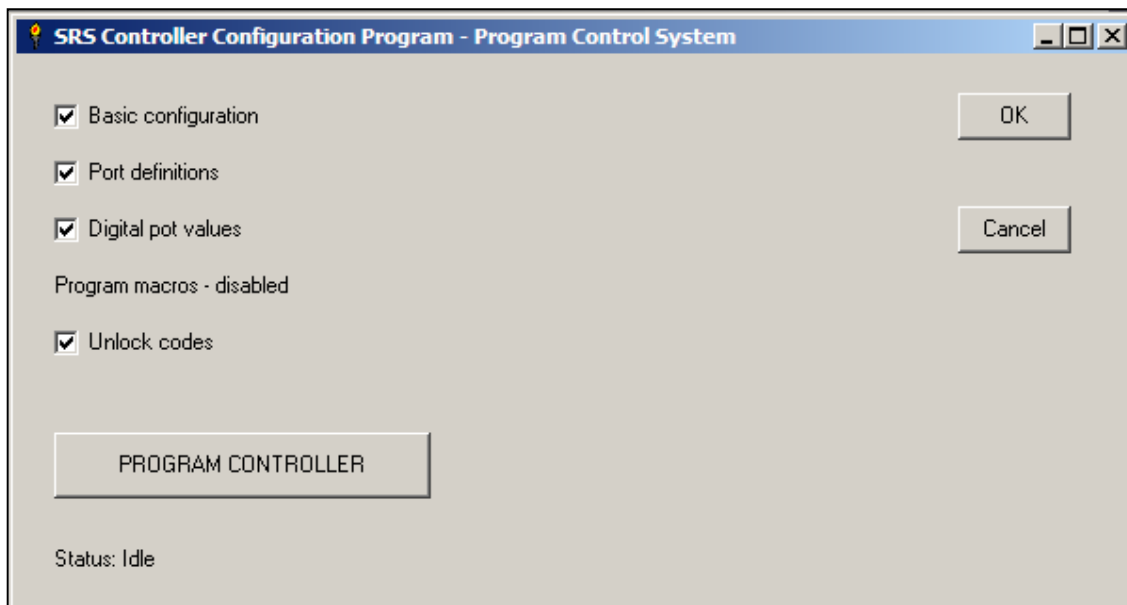
After you create your configuration, you should save it to a disk file so you can easily edit or re-download your configuration in the future into the same or another control system. To generate your configuration file, select File / Save As. You will see a dialog box like this...



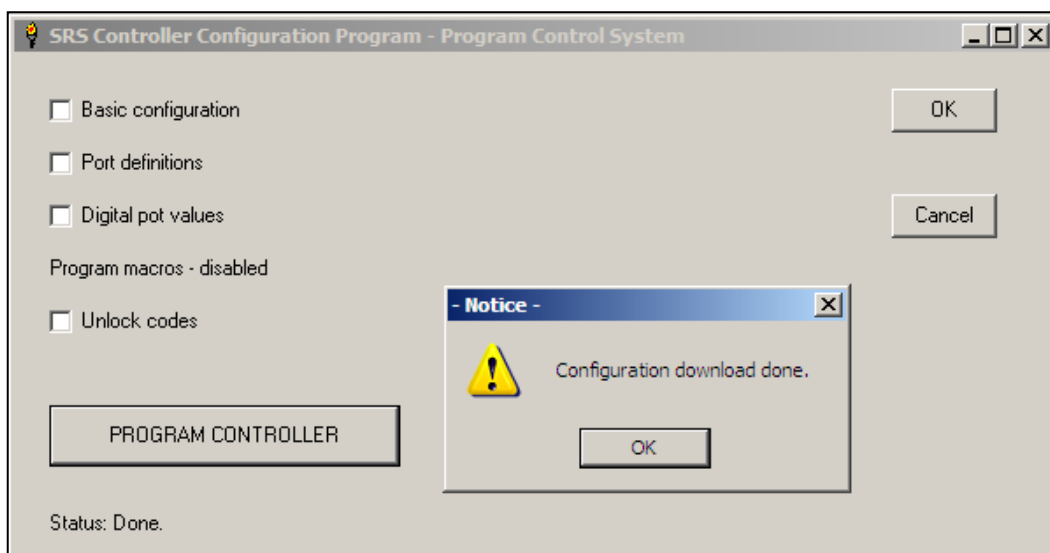
The dialog box will open with a default file name of “newfile”. It is recommended that you change the name to something more meaningful, like the name of the radio site or a version number of your own making. If you enter “bigmountain” for example, you will create a configuration file named bigmountain.cfg. To create your configuration file, just press the generate button and in a few seconds you are done. The newly created configuration file can be read or “loaded” into the config program at any time, edited or downloaded to your control system.

## 6. Download configuration to the controller

Now that your configuration file is done, you are ready to download it to your control system. Select menu Program / Write configuration to controller. The download dialog box looks like this...



This dialog box will allow you to select which parts of the configuration are to sent to the control system. You don't have to send the whole configuration. For example, if you only changed the callsign you might want to select just the “basic configuration” selection. Then click “Program Controller”. Only the checked sections of the configuration will be sent to the controller.



## Terminology and File System

### Files

The standard distribution of config comes with a set of support files as follows...

Config_xxx.exe	This is the main config program.
*.sll, *.dll, *.tkn	These files are part of the config program.
config.env	The config program environment file containing PC installation and configuration data.
default.cfg	The default control system configuration file.
scriptxx.scr	Script files. These are empty and are modified by the user.
My_reference.htm	An html file containing any reference information about your system such as control codes, site info etc.
SRS Command List.htm	This is the list of all built-in commands for the control system.

### Configuration file (\*.cfg)

An ascii text file that contains one or more of the configuration parameters for a particular control system. The configuration file is named *yourfilename.cfg* where *yourfilename* is whatever you like. This file can be hand generated but it is hundreds of lines long so it is generated by the config program automatically. When you select “file” then “save as” from the top menu, you have a dialog box that lets you generate a config date file. Enter the name you want in the config file box. Then, click “apply” then click “generate”. The config program will generate the complete configuration file for you based on the current settings you have entered in the various configuration windows.

To load in a saved config file, enter the config file name in the top left field on the main page and click “load”. You can also click “browse” to look for a file. This will read the configuration file and load it into all the fields. You can then either edit the configuration and save it again or download it to the control system.

You can have as many configuration files as you like. Typically one per control system but you may have various versions depending on what your needs are.

When you look at the contents of the config file, you will see three types of data: a comment, a configuration switch and configuration data.

All lines starting with a single quote ' are comments and are ignored by the config program.

Switches start with a forward slash / followed by a name such as port\_0\_type followed by a space then data. In this example the line assigns the port 1 personality type to be a 1 which means a repeater.

Example:

```
/port_1_type 1
```

Note that there are never any spaces inside the switch name. Also note there is always one, and only one space between the switch name and the data, in this case a "1".

The only exception are a few commands where spaces can be inserted in the data such as a CW ID string. In this case the switch is on the first line and the string with spaces is on the next line.

Example:

```
/cwid_1  
DE W1AW
```

Note the use of the space in the string. This is by far the exception.

Blank lines can be inserted anywhere and will be ignored by the config program.

If a subset of the available parameters are included in the config file, the last configuration that was loaded will become the default. So if you read in a new config file with just the call sign of the station, all other parameters that were present in the system will be used.

When the program initially starts, it will automatically read a default configuration file called "default.cfg". The software is supplied with a generic default.cfg file but you can modify it and save it off as your own default file.

## Script Files xxxxxx.scr

The control system accepts commands either from the DTMF decoder or through the serial port on the main CPU board. A simple program like HyperTerm can be connected to the serial port and you can simply type in commands and look at the results on your screen. Serial port settings are 9600 baud, N81. To make it more convenient to send several commands to the control system, you can create an ascii text file called a script file. The script file is simply raw commands that the control system will understand.

For example:

```
C2011D
C20212345
C000
```

Which will assign port 0 to be a repeater, port 1,2,3,4,5 to be links and then do a system reset. These are called "script" files because in addition to the raw command strings the control system understands, you can insert special commands that the script processor will interpret and not pass on to the control system. These are either comments or script processing directives.

Comments start with a single quote ' just like in a configuration file.

Script processing directives all start with a forward slash / like the switches in the configuration file.

Script processing directives set the rate at which data is transmitted to the controller, and various other parameters.

Scripts are executed from the first line to the last. There is no notion of flow control or other more advanced programming features.

You can load up to 10 of your favorite script files in the main window of the config program. You can simply click the edit button to change them or click the run button to start running them.

## Script File Control Directives

'Comments

Any line that starts with a single quote ' is a comment and the rest of the line is ignored by the config program.

<empty>

Any empty line is ignored by the config program.

/end

Stop processing script file and ignore all lines after /end.

Script files support the following config.env environmental variables. When these script directives are encountered, they override the default values supplied in the config.env file. See the section on config.env for details.

```
/command_delay 500
/digit_delay 20
```

Script files also support the following config file (xxxxxx.cfg) parameters. See the section on configuration file parameters for details.

```
/unlock_code
/prefix
```

/call\_sign  
/location\_id

#### Additional script processing directives

/set\_audio\_all xxx

Sets the value of all Rx, Tx, Telemetry and aux audio digital pots to a value of xxx, where xxx is between 0 and 255. 0 is no audio and 255 is maximum.

/set\_audio\_level\_port0\_rx xxx

/set\_audio\_level\_port0\_tx xxx

/set\_audio\_level\_port0\_te xxx

/set\_audio\_level\_port0\_to xxx

Sets the specific digital pot level to xxx, where the port is "port0" through "port7". The specific pot of that port is selected by the last two letters where...

rx is receiver audio level

tx is transmitter audio level

te is the locally generated telemetry audio level

to is the externally generated tone input, for example the PL encode level from an AS-3 audio squelch board.

The level xxx is between 0 and 255. 0 is no audio and 255 is maximum.

/set\_audio\_level\_cpu\_0

/set\_audio\_level\_cpu\_1

/set\_audio\_level\_cpu\_2

/set\_audio\_level\_cpu\_3

Sets the specific digital pot on the master CPU board level to xxx. The specific pot of that board is selected by the last number where...

0 is not used.

1 is audio from the DVB board.

2 is not used.

3 is the master CPU generated telemetry which drives the common telemetry bus that goes to all transmitters.

/fcg\_off

Turns off the function complete generator.

/fcg\_on

Turns on the function complete generator.

/wait xxxx

Causes the execution of the script file to stop and wait for xxxx milliseconds.

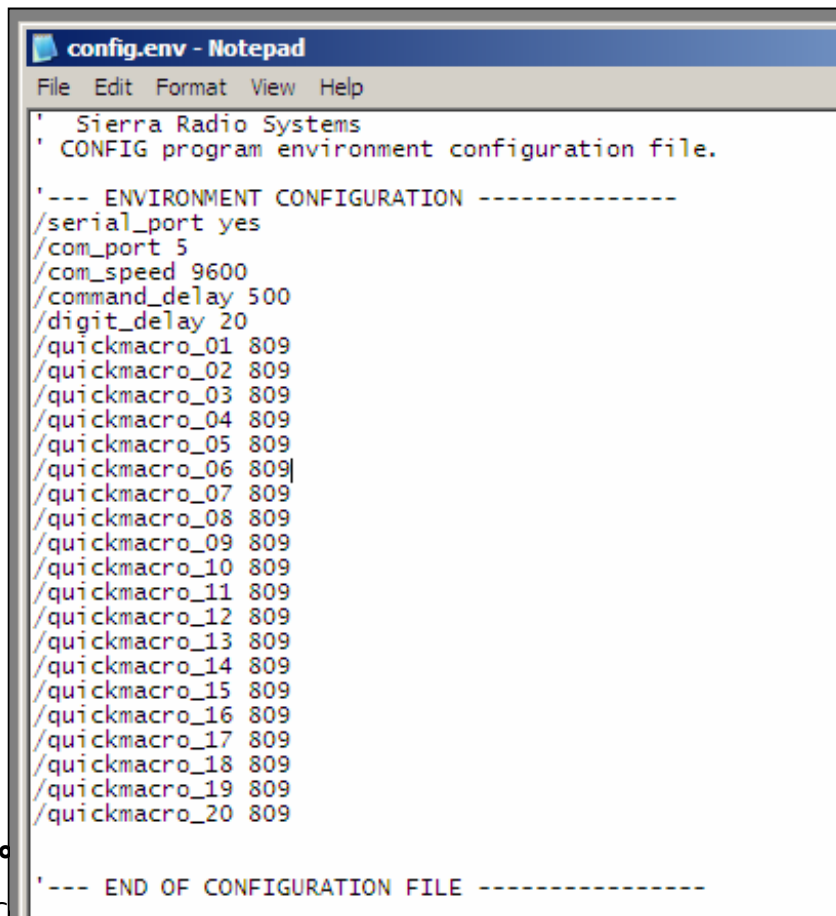
## Config Program Environment File config.env

Various environment variables are stored in the config.env file. This file is set up once per installation and does not move with the specific controllers configuration. The format of the config.env file looks similar to the .cfg and .scr files but always contains a specific set of parameters. These parameters define which com port is used, the speed, and other useful or necessary installation specific information.

For example, up to 20 “quick macros” are stored in the config.env file. These are essentially like a one line “script” where you can put up to 20 of your most commonly used commands.

The most important thing about the config.env file is to make sure the com port is set properly. If it is not, when the program starts up you will get a warning saying that it can not find the serial port. This is the most commonly made mistake in setting up your environment. Most computers will use com port 1 but your system may be different. If you are using a USB to serial adapter you will certainly have to check the hardware configuration in Windows first then edit the config.env file to make sure they match.

The typical config.env file looks like this...



```
config.env - Notepad
File Edit Format View Help

' Sierra Radio Systems
' CONFIG program environment configuration file.

'--- ENVIRONMENT CONFIGURATION -----
/serial_port yes
/com_port 5
/com_speed 9600
/command_delay 500
/digit_delay 20
/quickmacro_01 809
/quickmacro_02 809
/quickmacro_03 809
/quickmacro_04 809
/quickmacro_05 809
/quickmacro_06 809
/quickmacro_07 809
/quickmacro_08 809
/quickmacro_09 809
/quickmacro_10 809
/quickmacro_11 809
/quickmacro_12 809
/quickmacro_13 809
/quickmacro_14 809
/quickmacro_15 809
/quickmacro_16 809
/quickmacro_17 809
/quickmacro_18 809
/quickmacro_19 809
/quickmacro_20 809

'--- END OF CONFIGURATION FILE -----
```

Any line that starts with a single quote ' is a comment and the rest of the line is ignored by the config program.

<empty>

Any empty line is ignored by the config program.

/serial\_port yes

Yes indicates that the serial port on the computer is active and when the config program is launched it should attempt to connect to the control system.

No will prevent the config program from attempting to communicate with the control system through the serial port.

/com\_port 5

Sets the comp port that the config program should use to communicate with the control system.

/com\_speed 9600

Set the serial port's data rate. Should always be set to 9600

/command\_delay 500

The time in milliseconds that the config program will wait between sending commands to the control system.

/digit\_delay 20

The time in milliseconds that the config program will wait between characters when sending commands to the control system.

/quickmacro\_01 809

Up to 20 quick macros can be defined. These are single line command sequences that are available in the config program. Quick macro 1-4 are found on the main config screen and 1-20 are found on the Quick Macro pull down menu option.



## **Installing New Firmware**

## Installing New Firmware

To update the firmware on the main CPU or RCB boards, you must use an "in-circuit programmer". This device provides a connection between your computer and the control system boards. When installing new firmware, each board must be programmed individually. The only requirement is that the board must be powered and the in-circuit programmer must be connected. The in-circuit programmer we recommend is available from **ME Labs** and comes in many versions. There are parallel, serial and USB port versions. The serial version can also be used via RS-232 or through a USB to serial adapter. They all work fine so you should pick the unit that matches the type of interface available on your computer.

We prefer the USB version in the plastic enclosure because it provides protection from shorting out or physical damage.

Here are a few examples of what ME Labs offers...

**Option 1** - The EPIC™ Plus Programmer for the parallel port

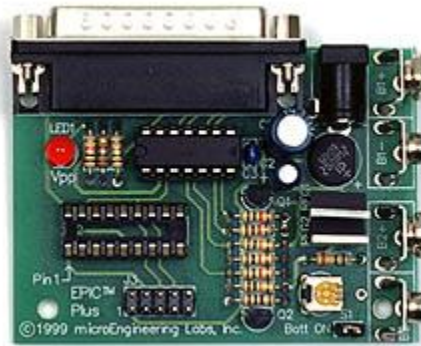
You must order the following...

EPIC™ Plus Programmer ICSP Board Only (Includes software)

AC adapter

25 pin cable

10 pin header cable



<http://www.melabs.com/products/epic.htm#Pricing>

**Option 2** - The Serial Programmer

You must order the following...

The melabs Serial Programmer ICSP board only

AC adapter

9 pin serial cable

10 pin header cable



<http://www.melabs.com/products/serprog.htm>

**Option 3** - The Serial Programmer in a plastic case

You must order the following...

**SRC** The melabs Serial Programmer ICSP board only

**ACAD** AC adapter

**10CAB** 10 pin header cable

9 pin serial cable



If you only have a USB connection you can use a USB to serial adapter. We suggest you order the adapter from ME Labs because we know it will work.  
Order the "USB 1.1 to Serial Converter Cable"



You can reach ME Labs on the web at  
[www.melabs.com](http://www.melabs.com)

or call them at (719) 520-5323

## Connecting to the Control System

The in-circuit programmer connects to the control system board with 4 signals including a clock, data, MCLR and ground. The connection at the programmer end is a 10 pin (2x5) header connector. The connection at the control system board end is either a 10 pin (2x5) pin header or a DB15 connector depending on which version of which board you are programming. This table shows which board support which kind of connector.

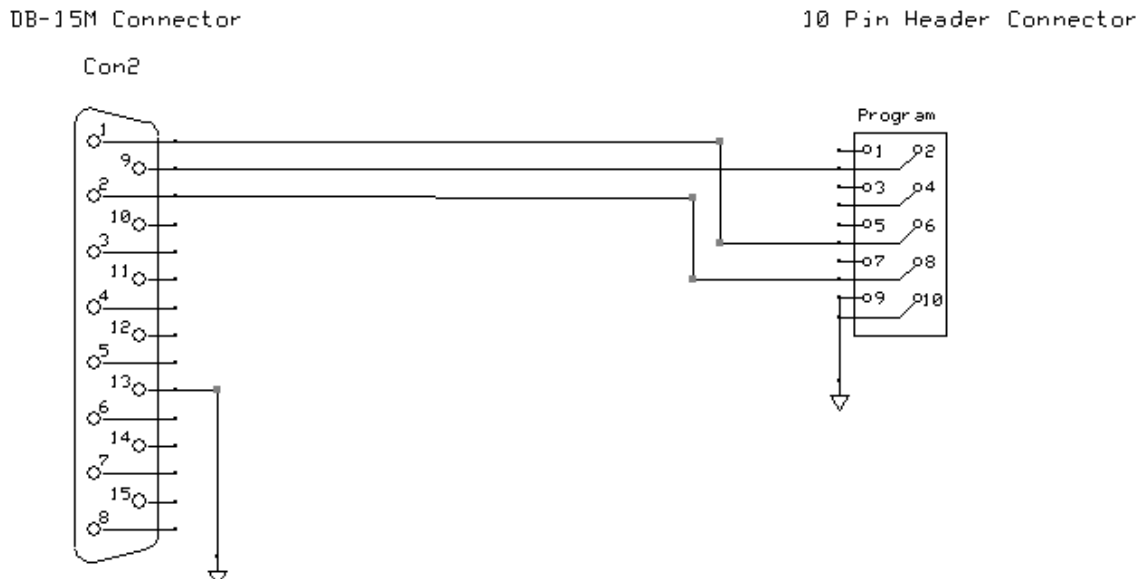
	10 Pin	
	Header	DB-15
CPU v1.0	<input type="checkbox"/>	
RCB pre-production	<input type="checkbox"/>	
RCB v1.0	<input type="checkbox"/>	<input type="checkbox"/>

The simple 10 pin header to 10 pin header ribbon cable can be used with all boards. The advantage of the DB15 connector on the RCB boards is that you can leave the boards plugged in the card cage while downloading new firmware.

### Optional cable for programming the local CPU on the RCB

Pins used for building an in-circuit programming interface:

Pin	Use
1	PGD
2	PGC
13	Ground
15	MCLR



Connection method #1 – 10 pin header and ribbon cable. Use this with the CPU or RCB boards.



## Downloading New Firmware to the Boards

Before you can install new firmware, you will need to install the ME Labs EPIC programmer software on your PC. This software is used to transfer the firmware image file from your PC to the flash memory of the CPU chips on the CPU, RCB or other boards. The software disk was included in the package when you got your EPIC programmer from ME Labs. You will also need the main CPU firmware or RCB firmware in the form of a .HEX file. The code files are available for download from the SRS web site.

When installing firmware the target board (CPU, RCB, etc) must be powered on. Although it is not necessary to plug the board into the card cage for programming, it is a convenient way to provide power to the board.



**SPECIAL NOTE:** When programming the Radio Control Boards, make sure the main CPU board is UNPLUGGED. The main CPU board provides a system wide reset signal which is sent to all RCBs. If the CPU board is plugged into the rack when programming the RCBs the system reset signal will prevent the EPIC programmer from taking control of the local CPU on the RCBs, thus preventing the RCB from being programmed. You know you have this situation when you see an EPIC programmer error message such as “Target not found” indicating that the EPIC programmer does not see the RCB’s CPU chip.

## ME Labs EPIC In-Circuit Programmer Settings

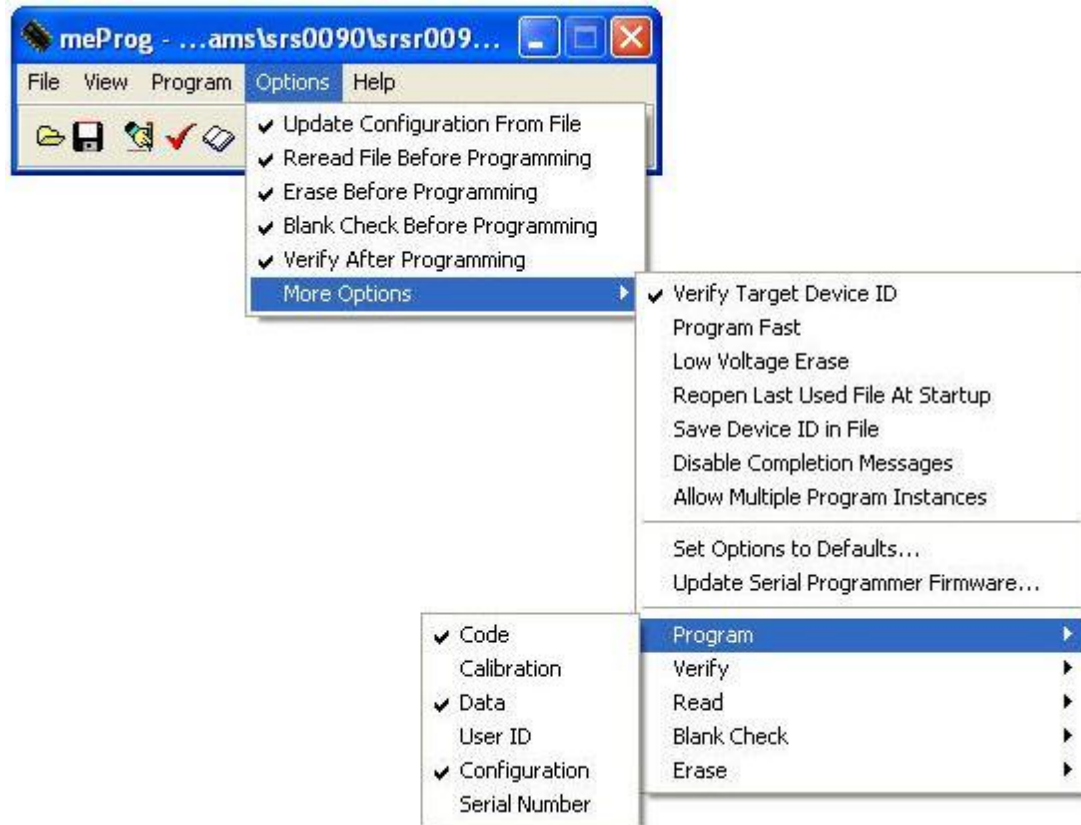
The programmer must be configured to program the correct type of processor, and set various other parameters. The EPIC software version should be v3.31 beta or later. Select the type of processor. For v1.0 production hardware use the following settings...

CPU board – Select the 18F8722 CPU type

RCB board – Select the 18F4620 CPU type



Next, set the Options menu parameters as follows...





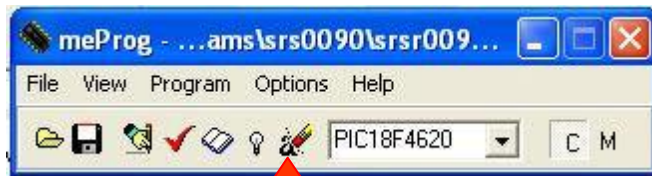
After opening the Radio Control Board firmware .hex file using the File menu, the program will show the first few items of the processor configuration as follows:



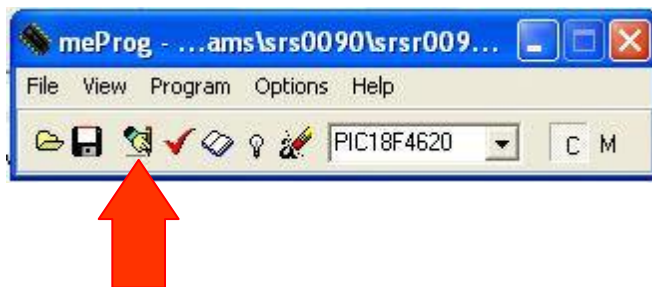
## Downloading New Firmware to the CPU Board

To download new firmware, you should follow these steps...

- 1 - Turn off the power to the control system.
- 2 - For the CPU board, plug the short 10 pin ribbon cable into the CPU board. For the RCB, you can either use the 10 pin connector or the more convenient DB15 connector on the front of the radio control board PCB.
- 3 - Plug the AC power adapter into the in-circuit programmer
- 4 - Plug the serial or parallel cable into your computer
- 5 - Turn on the power to the control system.
- 6 - Launch the EPIC software on your PC
- 7 - Select the controller firmware file to download to the CPU board.  
The CPU firmware is named srcxxxx.hex  
The RCB firmware is named srsrxxxx.hex  
xxxx is the firmware release number.
- 8 - Erase the CPU's flash memory

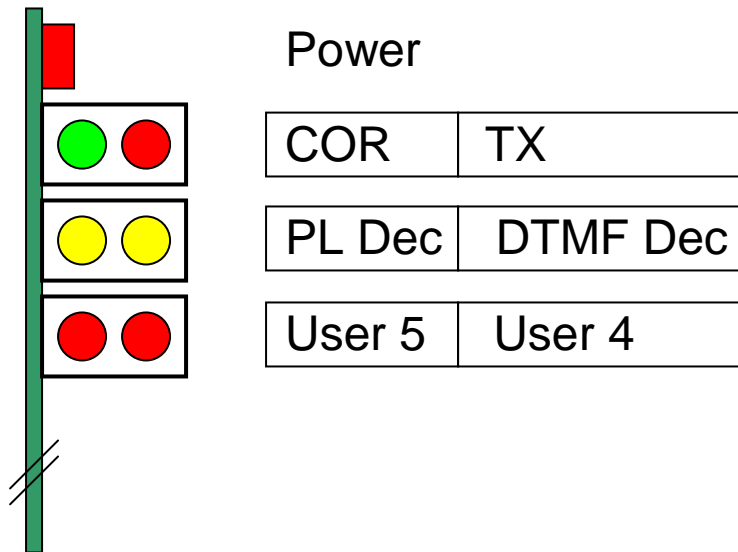


- 9 - Download the new firmware to the CPU board



## Control System Operation

## Status indication LEDs on the Radio Control Board



### LED Functions

- |                       |   |
|-----------------------|---|
| Red power LED         | - Power has been applied to the RCB board.                  |
| Green COR LED         | - A valid carrier signal is present                         |
| Red TX LED            | - The control system is keying the transmitter              |
| Yellow PL Dec LED     | - A valid PL / DPL tone has been decoded                    |
| Yellow DTMF LED       | - A DTMF digit is being decoded                             |
| Red User output 5 LED | - The open collector output #5 is active (pulled to ground) |
| Red User output 4 LED | - The open collector output #4 is active (pulled to ground) |

Note: there is no special indication when user output #1-#3 are active.

## Command Processor

The heart of the control system includes a powerful command processor which reads all commands that come in either through the DTMF decoders or through the CPU board's serial port, analyzes them and executes valid commands.

Here are some important concepts to understand in the behavior of the command processor...

1. Command strings entered through the DTMF decoders or the serial port are inserted into the command processor and treated the same.
2. Commands may consist of 2, 3, 4 or 5 characters.
3. No "terminator" or "enter" key is used. As digits are entered, the command processor scans the input buffer and if it detects a valid command, it will immediately execute the command and flush only those characters from the buffer. While this makes the command processor powerful in terms of command names and keeps commands names short, the limitation is that you can not have a command name that is a subset of another. For example you can have commands "123" and "456" but you can NOT have a command "12" and another command "123". In this case, the command processor will detect "12" as a valid command and execute it before it sees anything else.
4. The control system never uses an "unkey" or drop of COR signal to execute a command. This lets you run full time linked systems including VOIP and microwave systems and not rely on a loss of COR to execute.
5. Commands can be strung together. For example, commands 123, 456 and 789 can be run together as "123456789" and all three will be executed properly and again, no unkey is required.
6. When a valid command is decoded, the system will generate "function complete" telemetry. This indicates that a command was executed. You will hear a function complete tone for each valid function. The default tone set for function complete is "beep beep" of two 1064 Hz beeps of about 100 ms. duration.

### Entering commands on a local repeater port

When controlling from a repeater port, the controller prefix is a DTMF star “\*”. For example when entering command 123 you would key up and enter “\* 1 2 3”.

After successful decoding and execution of the 123 command, the control system will generate the function complete “beep beep”.

All DTMF digits are ignored until the “\*” is entered. The controller can be configured to either pass all non-decoded DTMF or mute all non-decoded DTMF. However, when a star is decoded the control system will immediately mute the repeater audio until the command executes or the command window timer expires.

### Entering commands on a remote site through a link port

When controlling from a remote site through a link, each control system on the network must have a unique address or “prefix”. All prefixes start with a pound “#” followed by 2 or 3 DTMF digits from the set 0,1,2,3,4,5,6,7,8,9,A,B,C,D.

Valid prefixes include #01, #1#, #AB, #123, #A9D, etc.

Of course, if you can not use a 3 digit prefix that overlaps with a 4 digit prefix. For example you can have #01 and #123 on the same network but you can not have #12 and #123 on the same network.

When controlling from the link you first send the prefix of the control system you want to control like #01. If controller #01 is listening on the network, it will respond with a “controller ready” telemetry tone. The default tone is a common dial tone (350 Hz and 440 Hz). At this point, you can enter your command, lets say “123”. Do not ever enter a star “\*” when controlling on the link. As with the normal repeater input, you can string together multiple commands. For example You can send commands 123, 456 and 789 in the following way...

Key up, send #01, unkey.

Hear the dial tone from site #01

Key up and send 1 2 3 4 5 6 7 8 9, unkey

You will hear “beep beep” “beep beep” “beep beep” indicating all three commands were executed on site #01

Note that the control system access timer, which allows commands to be received, is active while the “dial tone” is running. If no commands are entered, after about 10 seconds, the window will time out and the control system will ignore DTMF until it is prefixed again.

While the access window is active, as soon as a single DTMF digit is entered, it will stop the “dial tone” and keep the window open for a few seconds. If you keep entering digits, the window timer will refresh. If you wait too long, more than 4 seconds, the timer will time out and the input buffer will be flushed.

If you prefix the remote system, and another user is in the middle of controlling the system from another port, you will get a busy signal. The default is the telco “busy signal” (pulsing 440 Hz and 680 Hz). When the other user is done, you can then access the system.

### Entering commands on the local serial port on the main CPU board

When controlling from the serial port, simply enter the string of command digits without any star “\*” or prefix “#xx” digits. Enter commands to the serial port can be done with a simple dumb terminal program like HyperTerm or the SRS graphic user front end program called CONFIG.

## Unlock Codes

The control system can be configured to decode commands that require or do not require an unlock code. An unlock code is defined by the system administrator and can be any length from 3 to 20 digits. The system supports up to 20 different unlock codes. The first of these codes is the “master unlock” code.

The default SRS master unlock code is 138065.

☐

Change the master unlock code to your own personal code.

If a command requires an unlock code, the unlock code is entered between the access digit(s) and the command. For example if the unlock code is 138065 and the user command to be executed is 123 you would enter...

\* 1 3 8 0 6 5 1 2 3

The unlock feature has a smart timer. After the unlock code is entered, it resets a timer and allows commands that require an unlock code to be entered without an unlock code as long as the 5 minute unlock window is still open. This makes it easy for a system administrator to enter several unlock protected commands without having to enter the unlock command each time. Of course you can enter the unlock code with each command if you like.

## Control System Command Reference Guide



# Control System Command Reference Guide

The Series 200 control system comes with hundreds of built-in commands and command variations. Commands can be divided into 4 groups: common User Level commands, System Admin or “Super User” commands, System Configuration commands, and internal hardware commands.

Each system operation will consider who is a “user” differently. In some systems, no user has access to any commands while in other systems, all users have some degree of command execution authority. For the purpose of this introduction, we will consider a user to have limited access to the most commonly used commands and the system administrator would have access to all commands. (u) indicates that this command requires an unlock code to execute.

## User Level Commands

These commands are the most commonly used commands that are used to operate the system on a day to day basis.

- System normal commands
- Identification
- Repeater System on / off
- Repeater Port Commands
- Link Port Connection Commands
- Port Access Mode – Carrier / CTCSS/DCS / Etc.
- Port Monitor Commands
- Force transmitter on / off
- Loopback
- General Purpose I/O (GPIO) commands
- Remote Base Commands

## System Admin (Super User) Commands

These commands are reserved for the system administrator who need to reconfigure the system. System admin commands may perform the same function as a user level command. In this case, the system admin command takes priority. For example if a system admin command “123” turns off port 1 and the user level command “456” turns port 1 on, the system admin command will take priority.

- Test tone generator
- Link port CTCSS mode commands
- Port Enable / Disable Master Commands
- Command Enable / Disable Commands
- DTMF Generation Commands

## System Configuration Commands

These commands are used by the system administrator to set up the specific behavior of the system. Typically these commands are defined once and don't change. In most cases the default system configuration parameters supplied with the control system will be used unaltered by the system administrator.

- Unlock code management commands
- Configuring basic parameters – ID, prefix, location ID
- Macro programming commands
- CW Telemetry Commands
- Port personalities and link map
- Group configuration and management
- Digital Pot Controls
- Command Telemetry Routing
- Radio normal behavior
- Remote base radio configuration
- Basic configuration – ID, prefix, location ID

## Internal Hardware System Commands

- Reset processor
- Firmware code version
- Reset counter management
- Query number of ports
- Query firmware CRC values
- Save / restore configurations to / from EEPROM
- Low level EEPROM read / write commands
- Macro table block read / write

## System Normal Commands

These commands return the control system to a known, safe state. Some normal commands reset parts of the control system and other commands can reset all control system states to a known condition.

### C300 (u) Site normal

Sets normal conditions for the entire control system including all groups.

- Sets CTCSS / DCS required to on,
- Sets link delay on
- Connects or disconnects (link on/off) ports according to their normal link state as defined in the group definitions (all on in the default configuration)
- Turns on audio loopback on repeater ports only
- Turns off audio loopback on all other port types
- Turns remote monitor modes off
- Disables remote base transmitters
- Executes remote normal (C303) command.

### C301 (u) Link normal

Operates on the current group only.

- Connects link, VOIP/IRLP, and remote base ports according to their normal
- Link state as defined in the group definitions
- Disconnects (unlinks) all local radios.
- Disables remote base transmitters (use C380 to enable).

### C302 (u) Interface normal

Operates on the current group only

- Connects all ports which are defined as normally linked in the group definitions
- Not connected for VOIP and remote base ports.
- Disables the remote base transmitters (enable with C380).
- This can be used to reconnect the repeaters after a link normal (C301) command.

### C303 (u) Remote normal

- Clears unlocked conditions
- Turns off the force transmitter on test condition
- Turns off the test tones
- An unlock code will need to be reentered before executing subsequent commands which require an unlock code.

### C304 (u) Link reset

- Operates on the current group only

- Connects link ports according to their normal link state as defined in the group definitions.
- Does not change whether locals are connected.
- Turns off remote monitor
- Turns link delay on
- Set link PL to normal state
- Remote base transmitters disabled

#### C305 (u) Local reset

- Connects the normally connected local ports in the current group.
- Does not changed the link on/off status of non-locals.

#### C306 (u) Link reset flexible

Takes single byte options parameter to define command behavior, as for Inkrstdefs configuration parameter in C304.

Syntax is C306o, where o is the option byte, above.

Always turns off remote base transmitters.

#### C307 (u) Group normal

**Same as site normal, but only operates on the current group.**

#### C308 (u) Break all links

Disconnects all ports in the current group.

### Identification

#### C310 (u) Force CW ID

Set ID timer to 0 on all radios in current group, thereby forcing ID on all transmitters. Default 1064 Hz tone.

#### C311 (u) Force local CW ID

Same as C310, but forces only all the local radios in the current group.

C312 (u)    Send CW ID

Sends ID as telemetry. Currently goes only back to command source radio. This includes the primary local radio Default 800 Hz tone

C313 (u)    Send location ID in CW

Returns the location string.

## Repeater System on / off

C002 (u) Repeater system off

Disables all transmitters and command processing

C003 (u) Repeater system on

Inverse of C002 & only command recognized after C002 [] - command is always allowed.

## Repeater Port Commands

C321 (u) Repeater on

Enables repeat (loopback) on the local port from which the command comes, or the lowest logical radio in the same group as the source if it is a link,

Connects the link if it was linked before. [plre] - command is not blocked if it comes from a local local even if other conditions would block it.

C320 (u) Repeater off

Inverse of C321. Disables repeat.

C323 (u) Repeater connect

Connects the repeater to the rest of the group.

C322 (u) Repeater disconnect

Disconnects the repeater from the rest of the group. Acts on the local from which the command comes, if it is a local, or the lowest logical radio in the same group as the command source if the source is a link.

C324 (u) Repeater status

Provides on/off/disconnected status for local from which command comes, or lowest logical local in the same group as the command source if the source is a link.

## Link Port Connection Commands

### C330 (u) Link off

Unlinks specific radio in the current group.

Syntax is \*C330r, where r is the logical port number (linkmap is applied).

This works with any port, including port 0, not just link ports.

### C331 (u) Link on

Links specific radio to all other linked radios in group.

Link is mapped to radio number by linkmap. [plre] - command is allowed from a link that is off if it is to turn on the same link.

Syntax is \*C331r, as for C330.

This works with any port, including port 0, not just link ports.

### C332 (u) Send link status OK. [plre]

Command is allowed from a link that is off if it is to turn on the same link.

Syntax is \*C332r, as for C330. This works with any port, including port 0, not just link ports.

### C333 (u) Split group links

Syntax is C333r1r2..rnD.

This command takes the ports r1..rn and links them together, leaving the other currently linked ports in the group linked as is.

Link map is applied. If entered with no radio specified (C333D), or with an invalid parameter, it will clear the split link, rejoining the ports. This applied only to the current group.

Consider an example in which a group contains a repeater on port 0, and links on port 1, 2, and 3, and all are initially linked together. Issuing C33303D will link the repeater and port 3, and unlink both from ports 1 and 2. Ports 1 and 2 will remain linked together. The original state is returned by issuing C333D. The split link condition is only reset by C333D, the break all links command, C308, or by the global site normal, C300. It should be appended to user macros for other "normal" commands if it is desired for those commands to

reset this condition.

C334 (u) Link delay off

Sets link delay off for link radios in current group.

C335 (u) Link delay on

Sets link delay on for link radios in current group.

C336 (u) This link off

Turns off the port from which the command is entered

C337 (u) This link on

Turns on the port from which the command is entered [p]

C338 (u) This link status

Returns link on/off status for the port from which the command is entered. [p]



## Port Access Mode – Carrier / CTCSS/DCS / Etc.

### C350 (u) COR mode

COR only required for local repeater (command source port if it is a repeater, lowest logical numbered repeater port in the same group as the command source if the source is a link port). [pe]

Command is not blocked if coming from a local even if PL is on and missing.

### C351 (u) PL mode

PL required for local repeater (target port determined as for C350).

### C352 (u) Send PL status

Uses status of first local in group. [pe]

Command is not blocked if coming from a local even if PL is on and missing.

### C354 (u) COR mode for specific port

Set COR only for a specific radio. This is not sticky, it will be reset on site normal or processor reset.

Syntax is \*C354r, where r is the port to change (link map is applied). [] - command always allowed.

### C355 (u) PL mode for specific port

Set PL required for a specific radio. This is not sticky, it will be reset on site normal or processor reset.

Syntax is \*C355r, where r is the port to change (link map is applied).

### C356 (u) Send PL status for specific port

Returns PL status for specific radio.

Syntax is \*C356r, where r is the port to change (link map is applied in versions after srsc0090e). [pe]

Command is not blocked if coming from same port whose status it is requesting, even if PL is on and missing.

C357 (u)    Save PL mode for specific port

Save the PL/COR mode for specific radio to eeprom.

Syntax is C357r, when r is the port number (link map is applied).

After this command is executed, the current PL state for this port will be reloaded each time the processor is reset.

---

## Port Monitor Commands

### C340 (u) Remote monitors off

Turns off remote monitor for the command source, with exception given above for remote monitor on. If the local port is in remmonl mode, then all remmonl ports in the group will have remmon cleared. If it is in remmon mode, remmon will be turned off for only that port.

### C341 (u) Remote monitors on

Turns on remote monitor for the command source port, if that is a local (repeater). Otherwise, turn remote monitor on for the lowest numbered logical port local in the group of which the source port is a member. If the command comes from the serial port, set remote monitor on for the lowest numbered local logical port in group 0.

### C342 (u) Remote monitor on with local link enabled

Turns on remote monitor for all of the currently linked local (repeater) ports in the same group as the command source. The local repeater ports remain fully linked to each other.

### C344 (u) Remote monitor off, port specified

Turns off remote monitor as if the command came from the specified port. Syntax is \*C344n, where n is the logical port number

### C345 (u) Remote monitor on, port specified

Turns on remote monitor as if the command came from the specified port. Syntax is \*C345n, where n is the logical port number.

### C346 (u) Remote monitor on with local link enabled, port specified

Turns on remote monitor with local link enabled as if the command source came from the specified port. Syntax is \*C346n, where n is the logical port number.

## Force Transmitter on / off

### C298 (u) Force transmitter on

Keys up a specific transmitter. Syntax is \*C298n, where n is the physical port number. This will timeout after 5 minutes, even if it is not reset with C299.

C299      Cancel 298

Turns off force transmitter on. Syntax is \*C299.

## Loopback

C250 (u) Turn on loopback Turns on loopback (repeats input to output) for a specific port in the current group. Useful for testing links. Syntax is \*C250n, where n is a logical port number (linkmap applied).

C251 Turn off loopback Turns off loopback globally - all groups. It does not affect the normal loopback for repeater ports.

## Delay

C260 (u) Delay

Does nothing but delays for a while. Syntax is \*C260nnnD, which nnn is the delay in increments of 5 ms. The maximum value for nnn is 255 (1.275 sec).

## General Purpose I/O (GPIO) commands

There are 5 user programmable general purpose output signals available on each radio control board (RCB). These signals are available on the DB15 aux I/O connector. Each signal is an open collector output with an active pull to ground. The output buffer can sink up to 500 ma and will pull to approximately 0.6v when active. When inactive, the signal will float. This works well to control external DC relays.

### C370 (u) Set GPIO word

Syntax is C370nn

Sets all 5 general purpose output pins on a specific RCB board specified by r based on the value of nn.

A set bit (logic 1) will actively pull that bit to ground.

r is radio number 0..7

nn is a number from 0..31 to define the state of the 5 pins.

### C371 (u) Set individual GPIO bit

Sets or resets one general purpose output pin on a specific RCB board.

Syntax is C371rbv

r is the radio (0..7)

b is the bit (0..4)

v is the value (0 or 1)

A value of 1 will set the output bit meaning that it will pull the bit to ground.

Example: C371 0 0 1      Set radio port 0, user bit 0 to ground (active low on state)  
          C371 0 0 0      Sets radio port 0, user bit 0 to float (off state)

## Remote Base Commands

C380 (u) Enable remote base port transmit

Enables remote base port transmit.

Syntax is C380r, where r is the logical port number 0..7.

Will not do anything unless the port link status is on.

C381 (u) Disable remote base port transmit

Disables remote base port transmit.

Syntax is C381r, where r is the logical port number 0..7.

C382 (u) Disconnect all remote base ports

Disconnects all remote base ports (disables transmit and sets link on/off status to normal state).

Normally used be remote base timeout.

C401 (u) Set remote base emission mode

Syntax is C401rnnnn, where r is the logical port number and nnn is the emission type.

C402 (u) Set remote base radio frequency in MHz

Syntax is C402rffffff, where r is the logical port number and ffffff is the frequency in MHz.

C403 (u) Set remote radio base frequency in kHz

Syntax is C403rffffff, where r is the logical port number and ffffff is the frequency offset from the set band value, in kHz.

C404 (u) Set remote base radio frequency in Hz

Syntax is C404rffffffff, where r is the logical port number and fffffffff is the absolute frequency in Hz. The maximum value is  $2^{32}$  Hz, or a little over 4 GHz. For radios capable of higher frequencies, the frequency entered is

interepreted as 10's of Hz.

C405 (u) Set remote base radio frequency last digit of MHz and kHz

Syntax is C405rffff, where r is the logical port number and ffff is the frequency offset from the last set frequency value truncated to the nearest 10 MHz.

For example, C402146D sets the frequency to 146 MHz. However, the C405 value will be added to 140 MHz. A subsequent C40526520D sets the frequency to 146.52 MHz.

C406 (u) Set remote base radio frequency interactively Not yet implemented

C407 (u) Set remote base radio repeat mode

Syntax is C407rnnn, where r is the logical port number, and nnn designates the repeat mode, as follows:

C408 (u) Set remote base radio repeat offset

Syntax is C408rnnnnnn, where r is the logical port number, and nnnnn is the odd repeat split offset, in kHz. Plus or minus is indicated by the set repeat mode command.

C409 (u) Set remote base radio repeat reverse

Syntax is C409r, where r is the logical port number.

This command swaps transmit and receive from the normal setting (the previous set frequency value is the transmit frequency instead of the receive frequency). The state of this mode after entering any of the set frequency commands is radio type dependent.

NOT YET IMPLEMENTED

C400 (u) Set remote base radio type

This is considered a system administrator command because the remote base type is set once when the system is initially configured.

Syntax is C400rnnnn

r is the logical port number (0...7)



nnn is the radio type

C40A (u) Get remote base radio frequency

C410 (u) Set remote base radio repeat mode normal

Syntax is C410r, where r is the logical port number.

This command undoes C409

C411 (u) Set remote base radio transmit PL frequency

C412 (u) Set remote base radio receive PL frequency

## System Admin (Super User) Commands

### Test Tone Commands

C219 (u) Turn on test tone

Turns a 1 kHz tone on and off for test purposes.

Syntax is C219r

r is the physical port of the radio 0..8, where 8 is the CPU card

C220 (u) Turn off test tone

Turns off the test tone from C219

Syntax is C220r, as for C219.

C221 (u) Set the test tone frequency

Syntax is \*C221ffffD, where ffff is the frequency in Hz.

The value must be less than 4500 Hz. The default value is 1000 Hz, and is reset whenever the processor is reset. This value is not written to eeprom and is therefore not retained across resets.

C222 (u) Set the test tone amplitude

Syntax is C222aaaD

aaa is the amplitude, from 0..255.

The default value is 127, and is reset whenever the processor is reset.

C229 (u) Send test telemetry

Sends sound sequence from sound library.

Syntax is C229nnnD

nnn is the index of the sound sequence in the sound library. The tone amplitude is the same as that set for the test tone via command, C222.

## Link Port CTCSS/DCS Mode Commands

C358 (u)    Reset link PL mode to normal value

Reset the PL mode for all non-local radios in the current group to the eeprom stored normal values.

C359 (u)    Reset local PL mode to normal values

Reset the PL mode for all local radios in the current group to the eeprom stored normal values

## Port Enable / Disable Master Commands

### C360 (u) Disable specific port

Syntax is C360n, where n is the port number, with link map applied.

This command does not save the state to eeprom, so will be reset to its previously saved state on a processor reset. Use C363 to save it permanently. This command is designed to be allowed only in unlock or super unlock mode and therefore will work on any radio, independent of the radio's group. This command is designed stop operation of a port, and block commands from the port. The repeater off command uses this same function.

### C361 (u) Enable a disabled port

Syntax is as for C360.

This command does not save the state to eeprom, so will be reset to its previously saved state on a processor reset. Use C363 to save it permanently. This command is designed to be allowed only in unlock or super unlock mode and therefore will work on any radio, independent of the radio's group. [pe] - command is not blocked if coming from the same radio it is commanding.

### C362 (u) Send disabled status      Sends disabled status. Will not send status back to command source if the source port is disabled or super disabled. Syntax is as for C360. This command is designed to be allowed only in unlock or super unlock mode and therefore will work on any radio, independent of the radio's group. [pe] - command is not blocked if coming from the same radio it is commanding.

### C363 (u) Save disabled states      Saves the disabled status for one radio to eeprom, so that it will be retained across a processor reset. Syntax is as for C360.

### C364 (su) Super disable specific port      Super disable a radio. Syntax is C364n, where n is the port number, with link map applied. This command does not save the state to eeprom, so will be reset to its previously saved state on a processor reset. Use C367 to save it permanently. This command is designed to be allowed only in unlock or super unlock mode and therefore will work on any radio, independent of the radio's group. This is a more secure version of the C360 command. It is unlikely to ever be needed - it is designed to be used if you have a Jack Gerritson both using and trying to control your system over a link, and he has one of the unlock codes.

### C365 (su) Enable a super disabled port      Enable a super disabled radio. Syntax is as for C364. This command does not save the state to eeprom, so will be reset to its previously saved state on a processor reset. Use C367 to save it permanently. This command is designed to be allowed only in unlock or super unlock mode and therefore will work on any radio, independent of the radio's group. [pe] - command is not blocked if coming from the same radio it is commanding.

### C366 (u) Send super disabled status      Sends super disabled disabled status. Will not send status back to command source if the source port is disabled or super disabled. Syntax is as for C364. This command is designed to be allowed only in unlock or super unlock mode and therefore will work on any radio, independent of the radio's

group. [pe] - command is not blocked if coming from the same radio it is commanding.

C367 (su) Save super disabled state Saves the super disabled state for one radio to eeprom, so that it will be retained across a processor reset. Syntax is as for C364.

## Command Enable / Disable Commands

- C368 (u)    Command lockout specific radio        Set command lockout for a radio. Syntax is C368n, where n is the port number, with link map applied. This command does not save the state to eeprom, so will be reset to its previously saved state on a processor reset. Use C36A to save it permanently. This command is designed to be allowed only in unlock or super unlock mode and therefore will work on any radio, independent of the radio's group. After executing this command the port will still operate normally, but will not accept commands. It is designed to block a hacker trying to control the system from one of the links.
- C369 (u)    Enable a command locked out radio    Turn off command lockout for a radio. Syntax is as for C368. This command does not save the state to eeprom, so will be reset to its previously saved state on a processor reset. Use C36A to save it permanently. This command is designed to be allowed only in unlock or super unlock mode and therefore will work on any radio, independent of the radio's group. [pe] - command is not blocked if coming from the same radio it is commanding.
- C36A (u)    Save command lockout state    Save the command lockout condition for one radio to eeprom, so that it will be retained across a processor reset. Syntax is as for C368. [pe] - command is not blocked if coming from the same radio it is commanding.
- C36B (u)    Send command lockout status        Sends command lockout status for one radio. Will not send status back to command source if the source port is disabled or super disabled. Syntax is as for C368. This command is designed to be allowed only in unlock or super unlock mode and therefore will work on any radio, independent of the radio's group.
- C36C (u)    Send complete lockout status    Sends three number 0..255 providing the disabled, super disabled, and command lockout bit patterns for all radios. Port 0 is in the 1's place, port 1 is the 2's, place, etc. The ports are physical ports, not logical ports (linkmap NOT applied). For example, if 2 16 192 is returned, then physical port 1 is disabled, physical port 4 is super disabled, and physical ports 6 and 7 are command locked out. The normal condition would be 0 0 0. This command is designed to be allowed only in unlock or super unlock mode and therefore will work on any radio, independent of the radio's group. [p] - command is only blocked by missing PL.

## DTMF Generation Commands

C385 (u) Send dtmf string

Sends dtmf string to a specific radio port.

Syntax is C385rsssss...D

r is the radio port 0..7

s is a dtmf string of up to 16 characters

0...9 are entered as 0...9

A is an escape character for entering the remaining DTMF characters

AA sends a DTMF "A"

AB sends a DTMF "B"

AC sends a DTMF "C"

AD sends a DTMF "D"

A7 sends a DTMF "\*"

A9 sends a DTMF "#"

B inserts a space

C clears the input to start over

D is the termination for completion

### Examples

C385 3 123 D

Generates 123 out to the port 3 transmitter

C385 3 A7 555 AC D

Generates \*555C out to the port 3 transmitter

## System Configuration Commands

### Unlock Code Management Commands

C110 (su) Set unlock code Sets unlock code for enabling locked commands.

This command requires a super unlock code (code #0 or #1). The unlock code number

is returned in the telemetry if successful. No checking is done for conflicts with commands, so be careful that a command is not the same as the unlock code or is contained in the leading characters of the unlock code. The lowest index blank unlock code is set.

Syntax is C110sssssD

Up to 6 characters can be used (recommended). The terminating D is not used if 6 characters are entered. The unlock code characters must be numeric (0..9); ABCD are not allowed.

C111 (su) Delete unlock code Deletes the matching unlock code.

This command requires a super unlock code. Unlock code #0 cannot be deleted with this command. The unlock code index is returned if successful.

Syntax is C111sssssD

C112 (su) Delete unlock code index Deletes a specific unlock code number.

This command requires a super unlock code. Unlock code number 0 cannot be deleted. Confirms if an unlock code is deleted. Does not confirm if index is out of range, or no unlock code currently exists at the specified index.

Syntax is C112nnD

The system supports 20 unlock codes, with index 0..19

C113 (u) Check unlock code

Returns the index of a matched unlock code. Returns nothing if there is no match.

Syntax is C113sssssD.

C114 Set unlock code 0 (no radio control cards)

No unlock code is required, but all radio control cards must be removed from the controller first.

Syntax is C114sssssD

C115 (su0) Set unlock code 0

Sets unlock code 0. Requires unlock code 0 to be used to unlock the controller first.



Syntax is C115ssssD

## Configuring Basic Parameters – ID, prefix, location ID

C116 (u) Set the primary CW ID string

S116 (u)

Syntax is C116c1c1c2c2c3c3c4c4...D

Each character in the string is defined by two DTMF digits. This table represents all CW characters. For example, W6ABC would be represented by...

91 60 21 22 23  
W 6 A B C

		Seocnd Digit				
		0	1	2	3	4
First Digit	0	0	space	= BT	( KN	+ AR
	1	1	.	,	/	?
	2	2	A	B	C	@ AC
	3	3	D	E	F	#
	4	4	G	H	I	\$ SK
	5	5	J	K	L	~ AS
	6	6	M	N	O	% macro parm input
	7	7	P	Q	R	S
	8	8	T	U	V	*
	9	9	W	X	Y	Z

'2' is specified by "20", 'A' is specified by "21", 'B' by "22", etc.

If the C116 command is used from the serial port, spaces may be inserted between characters for clarity (eg. C116 91 60 21 22 23 <CR>)

Using the serial port version S116...

The S116 version is designed to be used via the RS232 port. It takes the characters directly. All spaces are included.

Return terminates the string, ! escapes.

For example, S116 W6ABC <CR> sets the ID to "W6ABC ".

This command does not download the new ID to the radio cards. Issue a processor reset (C000) to do this. One or two leading spaces should be included with the ID to give the transmitter enough time to come up before the CW starts; there is no delay in the program. The ID strings can be up to 19 characters long.

- C117 (u)    Get ID    Returns the primary ID string
- C118 (u)    Sets the secondary ID string.
- S118 (u)        This is used by radios for which the secondary ID flag bit is set.
- C119 (u)    Get ID2 Returns the secondary ID string

## Site Prefix Commands

- C120 (u)    Sets the prefix string. Syntax is the same as for Set ID. Beware that only 0..9,  
S120 (u)        A..D, and # can be used. The prefix can be up to 4 characters long.
- C121 (u)    Get prefix Returns the prefix
- C122 (u)    Sets the location string. Syntax is the same as for Set ID.
- S122 (u)        The location strings can be up to 9 characters long.

## Site Location ID Commands

- C123 (u)    Get location Returns the location string
- C124 (u)    Sets the secondary location string. Syntax is the same as for Set ID.
- S124 (u)        This location string will be appended to the ID for ports with the append location  
                    flag bit set (default is not set).
- C125 (u)    Get location 2        Returns location string 2

## Link Name

- C126 (u)    Sets the link name for one port.
- This is transmitted in the link status response along with the link number and status.
- Syntax is the same as the set macro command.
- The maximum length for the name string is 6 characters.

## Macro programming commands

C130 (u) Set macro

S130 (u)

Defines a macro. C130 is for dtmf input, S130 is for RS-232 input.

Syntax is C130c1c2c2, where the syntax is as for set ID, except the first few characters must define the macro number, followed by a space. For S130,

Syntax is S130nnn sssssss<CR>

C131 (u) Get macro

Returns the macro string.

Syntax is C131mD, where m is the macro number.

C132 (u) Set macro permission

Syntax C132mBpD, where m is the macro number is p is the permission byte

C133 (u) Get macro permission Returns the macro permission byte.

Syntax is C133mD

C134 (u) Turn off macro telemetry mute

Turns off the muting of function complete telemetry generation during macros.

Reset by remote normal or site normal.

C135 Macro site normal telemetry

Macros normally send the function complete telemetry associated with the last command in the macro when the macro exits. This command forces that telemetry to be the site normal telemetry if placed last in the macro. This command is useful if the group normal command, C307, is to be used as the site normal command. The command does nothing outside of a macro.

C136      Macro function complete telemetry

If placed last in a macro, this command forces function complete telemetry at macro end. This is useful if a configuration command is implemented by the macro, and it is desirable for the telemetry to be function complete instead. This command does nothing outside of a macro.

C137      Macro configuration complete telemetry

If placed last in a macro, this command forces configuration complete telemetry at the macro end. This command does nothing outside of a macro.

C138      No macro complete telemetry

If placed last in a macro, the macro will not issue function complete telemetry when it completes. This is useful if the macro calls a status command followed by a command which normally returns a function complete. In that case, it might be preferred that only the CW status be returned. This command does nothing outside of a macro.

## CW Telemetry Commands

C139 (u) Set CW WPM

Sets the CW speed in words-per-minute for command telemetry.

Syntax: C139 15 D

## Link Map

### C200 (u) Set link map

The control system performs operations on port numbers 0...7. When we refer to these ports they are “logical ports” which are normally mapped one-for-one to the physical ports. This means that the RCB assigned a hardware port number of 0 will map to logical port number 0.

With the link map you can assign any logical port to any physical port. In normal operation, all logical ports are mapped to their corresponding physical port. One example of why you may want to change the link map is the case where you have two physical repeaters on a single control system. One as a primary radio and one as a backup. Both can be plugged in at all times. Lets say you have many custom macros written to operate on logical port 0 which is assigned to physical port 0 and therefore the primary repeater. If the primary repeater should fail, you could easily remap the backup repeater on physical port 1 to behave as logical port 0. In this case, all commands that would normally act upon the primary repeater is now going to act upon the backup repeater with no other hardware or software configuration changes.

This command sets the link map which maps the logical port number to the physical port number.

Syntax is C200 p0 p1 p2 ... D

p0 is the physical port number (0..7) that will be recognized as port 0

p1 is the physical port number recognized as port 1, etc.

Any ports not listed will be assigned in increasing order following the last one specified. If there is a local repeater port, the first one should be logical and physical port 0 - not absolutely required, some of the commands which treat the lower numbered physical port which is a local radio as the primary repeater port will may not send telemetry to the correct port if this is not done.

## Port Type Assignment

Each port must be assigned a port type. A port type defines a set of characteristics that make the port exhibit a specific behavior. These characteristics typically include but are not limited to the following...

- Audio loop back on or off
- DTMF muting on or off
- DTMF command decoding enabled / disabled
- Transmitter carrier delay on or off
- Telemetry generation to port transmit output
- Etc...

## Port types

### Repeater

The repeater port type loops back audio, adds carrier delay, mute DTMF audio, etc. In general it behaves the way you would want a repeater to behave.

### Link

A link port type is used for all RF or VOIP point to point link circuits which are part of the network "backbone".

### Remote Base

The remote base port type is used for HF/VHF/UHF remotely controlled base stations. The remote base port will not accept DTMF commands from the outside world and will not generate transmit telemetry. The control system will make the remote base radio sound as much like a manually operated transceiver as possible. The remote base can be frequency and mode agile and takes additional parameters to assign the radio control format necessary to control the remote base radio.

### IRLP

The IRLP port type is similar to a remote base port but is used for public VOIP networks like IRLP, Echolink and others.

### VOIP Link

The VOIP Link port type is similar to the Link port type but adds additional conditions to handle radio over IP signaling.

## Syntax example

\*C201 r1 r2 ... D

r1 r2... is a list of physical ports from 0 to 7.

### Examples

\*C2010D sets the first port (physical port 0) as the only repeater port

\*C202123456D set ports 2,3,4,5,6 as link ports

\*C2037D set port 7 to be a remote base port

Any conflicts between these commands are resolved as the command is entered, in favor of values specified in the last command entered. The receive signal detect qualification and unqualification delays for each radio are set to the respective normal values for the radio type. The PL/COR required state is set to the defaults for



the radio type. Be very careful in using these commands remotely.

Note: The change takes place immediately and a mistake can make it difficult to control the system. Sets all group linked values for ports changed by this command to default values (linked for local and link ports, not linked for remote based and voip ports).

C201 (su) Sets the port type to repeater

C202 (su) Sets the port type to link

C203 (su) Sets the port type to remote base

C204 (su) Sets the port type to IRLP

C205 (su) Sets the port type to VOIP link

## Group Configuration and Management

The control system can support up to 4 independent groups. The default configuration uses one group (group #0) where all ports (0-7) are part of the same logical system. Ports can be grouped so that each logical group (#0 to #3) can operate independently. This means that ports in group 0 will not interact with ports in group 1, etc. The only limitation is that all groups must use the same system settings. This means they will use the same command set, telemetry tones, etc.

C210 (u)    Define group            Sets up a group membership.

Syntax is \*C210 g r1 r2 r3 D

g is the group number (0..3)  
r1, r2, ... are port numbers.

Example: C210 0 123 = Assign ports 1,2,3 to group 0

C211 (u)    Define group linked    Sets the group members which are linked by default.

Syntax is C211 gr1r2r3D

g is the group number (0..3)  
r1, r2, ... are port numbers

You do not need to define the group members first.  
Radios included here which are not group members will be ignored, but will be remembered if they are included in the group later.

C212 (u)    Create linked group

Used to create a temporary group to link groups together.

Syntax is \*C212rrrrrD

Where each r is a digit from 0 to 7 used to list the radios in the group. They are all link on by default. Radio types do not change. Linkmap.

This does not save to eeprom, so it reset when the processor is reset. It is also reset by site normal, C300, and restore groups, C213

C213 (u)    Restore groups

Restore the configuration to the programmed group configuration.  
Used to undo C212.

C214 (u)    Set current group

Explicitly set the current group.

Syntax is C214n  
Where n is the new current group number

C215 (u) Set current group, radio

Explicitly set the current group to the one to which the specified port is a member.

Syntax is C215n, where n is a port number.

C216 (u) Reset current group

Sets the current group to the normal value (group containing the command source).

## Digital Pot Controls for Audio Level Setting

The system contains multiple digital potentiometers used to set or “trim” various audio levels. There are two digital pots on the main CPU board and four digital pots on each of the radio control boards (RCB). Each pot can be set to a value between 0 and 255 where 0 is minimum, or no audio, 127 is midscale, and 255 is max audio.

There are two ways to set the digital pots: setting a specific value and setting a value interactively. Setting a specific value simply takes the value parameter and immediately sets the pot to that value. The interactive pot set will allow value increment and decrement without sending the whole command string. This is similar to the old way of adjusting a mechanical pot.

### CPU Board

- Pot 0
- Pot 1
- Pot 2
- Pot 3 CPU telemetry

### RCB Board

- Pot 0 External audio input (typically sub-audible tone encoders)
- Pot 1 Transmit audio
- Pot 2 Receive audio
- Pot 3 RCB local telemetry

C223 (u) Set radio pot      Sets one digital pot value.

Syntax is \*C223rpv

r is the physical port (0..8, where 8 is the cpu card)  
p is the pot (0..3),  
v is the value (0..255)

C224 (u) Set radio pot interactive      Sets the value of one pot, interactively.

Syntax is \*C224rp.....D

r is the physical port (0..8,) where 8 is the cpu card  
p is the pot (0..3)  
.... is a sequence of keystrokes to interactively set the pot value

1 and 7 increment and decrement the pot value by 1  
2 sets the pot value to 255  
5 sets the pot value to 127  
8 sets the pot value to 0  
3 and 9 increment and decrement the pot values by 10, respectively  
D saves the value  
\* escapes and returns the value to what it was before entering this command

## Setting Specific Digital Pot Values

Setting audio levels to a specific value between 0 (minimum) and 255 (maximum) level.

FUNCTION	COMMAND	USER NOTES
<b>CPU Board</b>		
CPU reserved	C 2 2 3 8 0 <0...255>	
CPU DVB bus level	C 2 2 3 8 1 <0...255>	Audio input level from digital voice board
CPU reserved	C 2 2 3 8 2 <0...255>	
CPU Telemetry bus	C 2 2 3 8 3 <0...255>	Common telemetry bus level
<b>Port 0</b>		
Port 0 Rx level	C 2 2 3 0 2 <0...255>	
Port 0 Tx level	C 2 2 3 0 1 <0...255>	
Port 0 Local telemetry	C 2 2 3 0 3 <0...255>	
Port 0 External tone	C 2 2 3 0 0 <0...255>	
<b>Port 1</b>		
Port 1 Rx level	C 2 2 3 1 2 <0...255>	
Port 1 Tx level	C 2 2 3 1 1 <0...255>	
Port 1 Local telemetry	C 2 2 3 1 3 <0...255>	
Port 1 External tone	C 2 2 3 1 0 <0...255>	
<b>Port 2</b>		
Port 2 Rx level	C 2 2 3 2 2 <0...255>	
Port 2 Tx level	C 2 2 3 2 1 <0...255>	
Port 2 Local telemetry	C 2 2 3 2 3 <0...255>	
Port 2 External tone	C 2 2 3 2 0 <0...255>	
<b>Port 3</b>		
Port 3 Rx level	C 2 2 3 3 2 <0...255>	
Port 3 Tx level	C 2 2 3 3 1 <0...255>	
Port 3 Local telemetry	C 2 2 3 3 3 <0...255>	
Port 3 External tone	C 2 2 3 3 0 <0...255>	
<b>Port 4</b>		
Port 4 Rx level	C 2 2 3 4 2 <0...255>	
Port 4 Tx level	C 2 2 3 4 1 <0...255>	
Port 4 Local telemetry	C 2 2 3 4 3 <0...255>	
Port 4 External tone	C 2 2 3 4 0 <0...255>	
<b>Port 5</b>		
Port 5 Rx level	C 2 2 3 5 2 <0...255>	
Port 5 Tx level	C 2 2 3 5 1 <0...255>	
Port 5 Local telemetry	C 2 2 3 5 3 <0...255>	
Port 5 External tone	C 2 2 3 5 0 <0...255>	
<b>Port 6</b>		
Port 6 Rx level	C 2 2 3 6 2 <0...255>	
Port 6 Tx level	C 2 2 3 6 1 <0...255>	
Port 6 Local telemetry	C 2 2 3 6 3 <0...255>	
Port 6 External tone	C 2 2 3 6 0 <0...255>	
<b>Port 7</b>		
Port 7 Rx level	C 2 2 3 7 2 <0...255>	
Port 7 Tx level	C 2 2 3 7 1 <0...255>	
Port 7 Local telemetry	C 2 2 3 7 3 <0...255>	
Port 7 External tone	C 2 2 3 7 0 <0...255>	

## Interactive Digital Pot Value Setting

Interactively set the value of one pot.

Syntax is \*129rp.....D, where r is the radio (0..8, where 8 is the cpu card), p is the pot (0..3), and .... is a sequence of keystrokes to interactively set the pot value.

Interactive "hot key" commands 1 and 7 increment and decrement the pot value by 1. 2 sets the pot value to 255, 5 sets the pot value to 127, 8 sets the pot value to 0, 3 and 9 increment and decrement the pot values by 10, respectively. D saves the value, \* escapes and returns the value to what it was before entering this command.

#### DTMF Key Pad "Hot Keys"...

(1) UP 1 unit	(2) Set to 255 MAX	(3) UP 10 units	(A)
(4)	(5) Set to 128 MID	(6)	(B)
(7) DOWN 1 unit	(8) Set to 0 MIN	(9) DOWN 10 units	(C)
(*) Escape	(0)	(#)	(D) Save value

FUNCTION	COMMAND	USER NOTES
<b>CPU Board</b>		
CPU reserved	C 2 2 4 8 0 <Hot_Keys>	
CPU DVB bus level	C 2 2 4 8 1 <Hot_Keys>	Audio input level from digital voice board
CPU reserved	C 2 2 4 8 2 <Hot_Keys>	
CPU Telemetry bus	C 2 2 4 8 3 <Hot_Keys>	Common telemetry bus level
<b>Port 0</b>		
Port 0 Rx level	C 2 2 4 0 2 <Hot_Keys>	
Port 0 Tx level	C 2 2 4 0 1 <Hot_Keys>	
Port 0 Local telemetry	C 2 2 4 0 3 <Hot_Keys>	
Port 0 External tone	C 2 2 4 0 0 <Hot_Keys>	
<b>Port 1</b>		
Port 1 Rx level	C 2 2 4 1 2 <Hot_Keys>	
Port 1 Tx level	C 2 2 4 1 1 <Hot_Keys>	
Port 1 Local telemetry	C 2 2 4 1 3 <Hot_Keys>	
Port 1 External tone	C 2 2 4 1 0 <Hot_Keys>	
<b>Port 2</b>		
Port 2 Rx level	C 2 2 4 2 2 <Hot_Keys>	
Port 2 Tx level	C 2 2 4 2 1 <Hot_Keys>	
Port 2 Local telemetry	C 2 2 4 2 3 <Hot_Keys>	
Port 2 External tone	C 2 2 4 2 0 <Hot_Keys>	
<b>Port 3</b>		
Port 3 Rx level	C 2 2 4 3 2 <Hot_Keys>	
Port 3 Tx level	C 2 2 4 3 1 <Hot_Keys>	
Port 3 Local telemetry	C 2 2 4 3 3 <Hot_Keys>	
Port 3 External tone	C 2 2 4 3 0 <Hot_Keys>	
<b>Port 4</b>		
Port 4 Rx level	C 2 2 4 4 2 <Hot_Keys>	
Port 4 Tx level	C 2 2 4 4 1 <Hot_Keys>	
Port 4 Local telemetry	C 2 2 4 4 3 <Hot_Keys>	
Port 4 External tone	C 2 2 4 4 0 <Hot_Keys>	
<b>Port 5</b>		
Port 5 Rx level	C 2 2 4 5 2 <Hot_Keys>	
Port 5 Tx level	C 2 2 4 5 1 <Hot_Keys>	
Port 5 Local telemetry	C 2 2 4 5 3 <Hot_Keys>	
Port 5 External tone	C 2 2 4 5 0 <Hot_Keys>	
<b>Port 6</b>		
Port 6 Rx level	C 2 2 4 6 2 <Hot_Keys>	
Port 6 Tx level	C 2 2 4 6 1 <Hot_Keys>	
Port 6 Local telemetry	C 2 2 4 6 3 <Hot_Keys>	
Port 6 External tone	C 2 2 4 6 0 <Hot_Keys>	
<b>Port 7</b>		
Port 7 Rx level	C 2 2 4 7 2 <Hot_Keys>	
Port 7 Tx level	C 2 2 4 7 1 <Hot_Keys>	

Port 7	Local telemetry	C 2 2 4 7 3	<Hot_Keys>	
Port 7	External tone	C 2 2 4 7 0	<Hot_Keys>	

## Command Telemetry Routing

- C225 (u) Command telemetry goes only to source.  
Responsiveness to RS232 serial port commands is much faster if CW telemetry is turned off while entering commands from the serial port.  
This command should normally be invoked before configuration via the serial port for that reason.
- C226 (u) Command telemetry goes to local (and source). This is the default.



## Basic Configuration – ID, prefix, location ID

### S118 Set ID2

Sets the secondary ID string. This is used by radios for which the secondary ID flag bit is set.

### S120 Set prefix

Sets the prefix string. Syntax is the same as for Set ID. Beware that only 0..9, A..D, and # can be used. The prefix can be up to 4 characters long.

### S122 Set location string

Sets the location string. Syntax is the same as for Set ID. The location strings can be up to 9 characters long.

### S124 Set location string 2

Sets the secondary location string. Syntax is the same as for Set ID. This location string will be appended to the ID for ports with the append location flag bit set (default is not set).

## Internal Hardware System Commands

C000 (u) Reset processor

C001 Send firmware version

Returns the CPU firmware code version

C021 (u) Send RCB firmware version

Returns the version string for one radio card.

Syntax is C021r

r is the physical port number.

C004 (u) Reset reset counters

Sets the reset counters to zero

C005 Reset counter status

Sends CW with reset count (# resets since \*C004). Six values are returned.

C006 (u) Exit safe mode

This command is obsolete for software versions 0090 and later.

C007 (u) Nradios status

Returns recognized number of radio cards

## Firmware CRC Commands

### C010 (u) Boot CRC

Returns CRC values determined at last configuration reset (first boot after programming CPU card processor, or first boot at eeprom valid byte reset, C100). The first value is the program ROM CRC, the second is the configuration parameter CRC in eeprom, the third is the macro table CRC in eeprom. This can be used to determine if the processor was properly programmed and if the default values were properly restored by the program

### C011 (u) Program CRC

Returns the current program ROM CRC. This takes about 7 seconds to execute.

### C012 (u) Eeprom parameter CRC

Returns the CRC of the current parameter space in the eeprom

### C013 (u) Eeprom macro CRC

Returns the CRC of the current macro storage area in eeprom

### C014 (u) Radio card program CRC

Computes the CRC of the program ROM in one radio card. Syntax is C014r, where r is the physical port number of the radio card. This several seconds to execute.

### C015 (u) Radio card CRC status

Returns the last computed value of the radio card CRC. Syntax is C015r. R is the physical port number.

## EEPROM Management Commands

The system configuration is stored in a serial eeprom external to the main processor on the CPU board. The serial eeprom can store 3 copies of the configuration parameters, including the table of macros. They are designated the working set, backup set, and deep backup set.

C100 (su) Force default configuration load on next processor reset.

Sets serial eeprom valid byte to 0, invalidating the eeprom. New eeprom values will be loaded from the hard coded defaults on the next reset.

C101 (u) Eeprom valid status

Tells if the eeprom contains valid data. This should normally return VALID, as it is fixed with the first eeprom write on the first boot after code load. It will return INVALID following successful execution of the C100 command and before the processor is reset. It is useful primarily for debugging.

C102 (u) Configuration backup

This command should be used after changes are made to the controller configuration and are verified to be working properly. Note that all of the configuration backup and restore commands except this one check for a correct eeprom valid status byte in the set to be copied before starting the copy.

C103 (su) Configuration restore

Restore configuration from backup set to working set.

Most controller operations use values of working parameters stored in RAM on the CPU and Radio Control board processors. To make the restored parameters active, a processor reset (C000) should be done after using this command.

C104 (su) Deep configuration backup

Copy backup configuration set to the deep backup set. This command should be used after the configuration backup command after a new configuration has been thoroughly tested.

C105 (su) Deep configuration restore

Copy deep backup configuration set to working set. Notice that this is NOT the inverse of the deep backup command which copies from the backup set to the deep backup set. This is changed from version of the program prior to 0094d.

C106 (u) Set eeprom byte

Sets a byte in the configuration eeprom.

Syntax C106aaaBnnnD

aaa address in the eeprom between 0 and 13823.

nnn byte value in the range of 0..255

C107 (u) Get eeprom byte

Returns one byte from eeprom as a decimal value.

Syntax is C107aaaaD

aaa is the address.

C108 (u) Set eeprom word

Sets a 16 bit unsigned word in the configuration eeprom.

Syntax is C108aaaBnnnnnD

Valid eepromaddress is between 0 and 13823

C109 (u) Get eeprom word

Returns one word from eeprom as decimal.

Syntax is **C109aaaaD**

Valid eeprom address is between 0 and 13823

C10A(su) Set eeprom bytes

Writes multiple sequential bytes in the configuration eeprom.

Syntax is C10A aaa Bnnn Bnnn Bnnn ... D

aaa    eeprom starting address  
nnn    byte values (in decimal) to be written to each byte

This command has an 80 character limit to the sequence. Up to 16 bytes can be written at a time.

Note: This command is super unlock code protected so that it cannot be used to set unlock codes using only an unlock code.

Valid address ranges are 0 ... 13823 and 41504 ... 65535.  
Nothing will be returned if the address is out of range.  
The super unlock code 0 will not be written, even if it is in the specified address range.

#### C10B (su) Get eeprom bytes

Reads and returns multiple sequential bytes in the configuration eeprom, as decimal numbers.

Syntax is C10B aaa Bnn D

aaa    is the starting address  
nn    is the number of bytes to be read, up to 16

Valid address ranges are 0 ... 13823 and 41504 ... 65535.

Note: If the super unlock code 0 addresses are contained in the address range, they will be returned as 0.

#### C10C (su) Get macro command table

Copies macro command table to serial eeprom, starting at address 41505.

Note: The number 12 is written to eeprom address 41504 to confirm that the macro table was copied.

#### C10D (su) Set macro command table

Writes macro command table from serial eeprom back to program memory (from where it is used). Checks for 12 in eeprom address 41504 before executing.

## Configuration Parameters

Timing Configuration						
Parameter Name	Description	Addr	Default Value	Type	Set Value	Get Value
dtmf_qual_delay	length of time dtmf tone must be detected by hardware to be recognized	48	1	byte	C106 48 B (0...255) D	C107 48 D
telemetrydelayl	time from key up to start of key up telemetry	49	8	byte	C106 49 B (0...255) D	C107 49 D
xmittimeouts_long	time from key up to dropping carrier	50	40	byte	C106 50 B (0...255) D	C107 50 D
xmittimeouts_short	time from key up to dropping carrier with link delay off	51	0	byte	C106 51 B (0...255) D	C107 51 D
dtmf timeouts	timeout between dtmf entry to abort command entry	52	30	byte	C106 52 B (0...255) D	C107 52 D
busytimeouts	dtmf timeout for busy signal on case	53	60	byte	C106 53 B (0...255) D	C107 53 D
id_idleto	time before ID is forced after transmit starts after long transmit gap (repeater port)	54	30	word	C108 54 B (0...255) D	C109 54 D
id_politeto	time remaining in absolute timeout when ID will be triggered if transmit is idle (repeat port)	56	120	word	C108 56 B (0...255) D	C109 56 D
id_absto	absolute timeout for sending ID (normally 10 minutes)	58	600	word	C108 58 B (0...255) D	C109 58 D
id_inkidleto	time before ID is forced after transmit starts after long transmit gap (link port)	60	5	word	C108 60 B (0...255) D	C109 60 D
cmdteldelays	delay after first command recognized until command telemetry starts	62	15	byte	C106 62 B (0...255) D	C107 62 D
serteldelays	delay after first command recognized until command telemetry starts, with rs232 command source and teltolocal=1. Value is 1 (defined by SERTEDELAYS0) if teltolocal=0.	63	1	byte	C106 63 B (0...255) D	C107 63 D
dialtonetimeout	timeout for dial tone after prefix recognized if not command is entered	64	10	byte	C106 64 B (0...255) D	C107 64 D
parmtimeout	timeout while waiting for a command parameter to be entered	65	5	byte	C106 65 B (0...255) D	C107 65 D
fantimeouts	timeout for fans after transmit stops	66	255	byte	C106 66 B (0...255) D	C107 66 D
rcvqdelays_loc	time local receiver needs to detect COR and/or PL until detected active, normal value	67	10	byte	C106 67 B (0...255) D	C107 67 D
rcvunqdelays_loc	local receiver delay to drop receiver active, normal value	68	30	byte	C106 68 B (0...255) D	C107 68 D
rcvqdelays_lnk	link receiver delay to receiver active detect, normal value	69	10	byte	C106 69 B (0...255) D	C107 69 D
rcvunqdelays_lnk	link receiver delay to drop receiver active, normal value	70	1	byte	C106 70 B (0...255) D	C107 70 D
rcvqdelays[]	specific radio values, set by set radio defaults	71	see above	byte	C106 71 B (0...255) D	C107 71 D
rcvunqdelays[]	specific radio values, set by set radio defaults	79	see above	byte	C106 79 B (0...255) D	C107 79 D

Strings						
Parameter Name	Description	Addr	Default Value	Type	Set Value	Get Value
	allocated size of prefix	5				
prefix_string	0 terminated string with prefix (0..9, A..B, # allowed)	87		special		
	allocated size of ID string	20				
id_string	0 terminated string to be sent as repeater ID	92		special		
id_string2	0 terminated string with alternate ID	112		special		
	allocated size of location string	20				
loc_string	0 terminated string with location	132		special		
loc_string2	0 terminated string with alternate location	152		special		
idselect	OBSOLETE	172	0	byte	C106 172 B (0...255) D	C107 172 D

## Configuration Parameters

Parameter Name	Description	Addr	Default Value	Type	Set Value
<b>CW Parameters</b>					
cwfreqs	frequency for cw telemetry	180	800	special	
cwamp	amplitude for cw telemetry	182	127	special	
cwwpm	cw speed for cw telemetry	183	18	special	
altidflags	for each port, bit=1 for radio to send id_string2 (0)	184		word, byte	C108 184 B (0...255) D
idlocflags	for each port, bit=1 for radio to append loc_string2 to id (0)	185		word, byte	C108 185 B (0...255) D
idlocalfreq	frequency for repeater and remote base ID (if enabled)	186	1064	special	
idlocalamp	amplitude for repeater and remote base ID	188	80	special	
idlocalwpm	cw speed for ID to repeater and remote base	189	20	special	
idlinkfreq	frequency for link ID	190	1064	special	
idlinkamp	amplitude for link ID	192	40	special	
idlinkwpm	cw speed for ID to links	193	24	special	
<b>Repeater port sounds with normal and radio specific values</b>					
nolnksnd	normal key up telemetry sound when nothing is linked	194		word, byte	C108 194 B (0...255) D
rnolnksnd[]	radio specific nolnksnd's	197		word, byte	C108 197 B (0...255) D
lnkintsnd	normal key up telemetry sound when some links active	221		word, byte	C108 221 B (0...255) D
rlnkintsnd[]	radio specific lnkintsnd's	224		word, byte	C108 224 B (0...255) D
lnkcardelsnd	normal key up telemetry sound with no link carrier delay	248		word, byte	C108 248 B (0...255) D
rlnkcardelsnd[]	radio specific lnkintsnd's	251		word, byte	C108 251 B (0...255) D
remmonsnd	normal key up telemetry sound when in remote monitor mode	275		word, byte	C108 275 B (0...255) D
rremmonsnd[]	radio specific remmonsnd's	278		word, byte	C108 278 B (0...255) D
remmonsnd	normal key up telemetry sound when in remote monitor mode	302		word, byte	C108 302 B (0...255) D
rremmonsnd[]	radio specific remmonsnd's	305		word, byte	C108 305 B (0...255) D
rbaseactsnd	added telemetry to repeater when remote base unkeys	329		word, byte	C108 329 B (0...255) D
rrbaseactsnd[]	radio specific rembaseactsnd's	332		word, byte	C108 332 B (0...255) D
<b>Repeater port sounds which are globally defined only</b>					
timeoutsnd	sent when timeout is about to reset the repeater to Site Normal	356		special	
alarmsnd	sent when hardware alarm is triggered (not implemented)	359		special	
<b>Link port sounds</b>					
accesssnd	valid prefix received from same port (def: dial tone)	362		special	
busysnd	valid prefix received, but controller busy (def: busy signal)	365		special	
alertsnd	valid prefix and link is off/disconnected (def: +950Hz)	368		special	
alert2snd	valid prefix and local is off/disconnected (i i @ def: +950Hz)	371		special	
lnkwrnsnd	function end telemetry when command blocked (def: 1300/1700Hz)	374		special	
rbaseInksnd	added telemetry to links when remote base unkeys	377		word, byte	C108 377 B (0...255) D
rrbaseInksnd[]	radio specific rmbaselInksnd's	380		word, byte	C108 380 B (0...255) D
fncmpsnd	function complete sound	404		word, byte	C108 404 B (0...255) D
progsnd	function parameter prompt sound (normally continuous tone), not currently used	407		word, byte	C108 407 B (0...255) D
wrnsnd	warning sound for invalid parameter input	410		word, byte	C108 410 B (0...255) D
reordersnd	invalidated password code entered	413		word, byte	C108 413 B (0...255) D
cnfsnd	confirm tone for configuration command	416		word, byte	C108 416 B (0...255) D
snrmsnd	site normal function complete sound, not currently used	419		word, byte	C108 419 B (0...255) D



## Methodology for System-wide Audio Level Setting

## Methodology for System-wide Audio Level Setting

Buffered audio from each of up to 8 receivers is distributed on the backplane. Each radio control board selects from 0 to 8 of the receive signals to mix and send to that particular transmitter. Each receiver and transmitter level can be adjusted with internal digital potentiometers.

The control system routes a standard internal signal level of 1v peak-to-peak for full system deviation from all receivers to all transmitters. Receive input signals to the RCB should be between a minimum of 250 mv p/p to a maximum of 4v p/p. The buffered output to each transmitter will be between 0 v to a maximum of 4v p/p. If for some reason you need more gain in either the receiver or transmitter buffers, you can change the values of the resistors on corresponding op amps or add audio squelch boards that provide a wide range of audio buffering and audio signal processing options. The opamps used in the RCB are low voltage devices that will handle a maximum of 4 v p/p.

The following procedure is used to set all audio levels. It is assumed that you have either two service monitors or one which can generate and receive on different frequencies at the same time. One service monitor is used as a signal generator to drive the receivers and the other service monitor is used to receive the transmitted signals. For illustration, assume you have a system with one repeater on port 0 and 4 link radios on ports 1 through 4.

The audio level set pots can be controlled in four ways:

- Directly setting a pot value through DTMF on a radio port
- Interactively setting a pot value through DTMF on a radio port
- Using a dumb terminal program connected to the serial port of the CPU board
- Using the SRS CONFIG program to interactively set all pot values.

While any of these techniques would work fine, the BEST method is #4. In this approach you have full control to adjust all digital pots from the graphic user interface. Once set, you can also store the configuration on your PC. This methodology explains how to set the audio level in a conventional 5 Khz FM radio system. Regardless of the method of setting each pot, here is the sequence of steps you would perform to set all levels properly...



### Step 1 – Set the Port 0 Rx level.

Set your service monitor to generate a signal that your repeater receiver frequency.

Generate a 1KHz tone with a RF deviation of 2.5 KHz on the receivers frequency.

Probe the DB-15 Con\_2 Aux I/O socket Pin 10 with an oscilloscope.

Using the config program or DTMF set the Port 0 receive level digital pot for 0.5v p/p.

This will be the reference point for adjusting all other audio levels.



### Step 2 – Set the Port 0 Tx level.

While still generating the signal used in step 1, set your service monitor to receive the signal now being transmitted by the radio on port 0.

Set the port 0 transmit level pot so that the port 0 transmitter deviation is set to 2.5 KHz deviation.

This will match the received signal to the transmitted signal.

☐ Step 3 – Set the link port Tx levels.

With the signal still being generated from step 1, move the service monitors receiver input through each of the link transmitters.

On each transmitter, set the Tx level digital pot for 2.5 KHz deviation.

☐ Step 4 – Set the link port Rx levels.

Set the receiving service monitor to the port 0 transmit frequency.

Set the generating service monitor to the port 1 receive frequency.

Generate a 1 KHz tone signal on the port 1 receive frequency that is set at 2.5 KHz deviation.

Adjust the port 1 receive level digital pot so that the port 0 transmitter generates the 1 KHz tone at 2.5 KHz deviation.

☐ Step 5 – Set the level of the rest of the receivers in the same way as port 1 was set in step 4.

☐ Step 6 – Set the system wide telemetry bus level.

Set the telemetry test tone parameters...

*C2211000D	Set the test tone frequency to 1000 Hz
*C222127D	Set test tone amplitude to mid scale (127)
*C2198	Turn on test tone generator

Now, set the CPU board's telemetry digital level set pot to 2.5 KHz deviation on the port 0 transmitter output.

☐ Step 7 – Turn off the built in test tone generator.

\*C220

☐ Step 8 – Set local telemetry level on each transmitter.

Now that the master telemetry bus is set, step through each RCB to set the telemetry level that is unique to each port.

Set the receiving service monitor to the port 0 transmitter frequency.

\*C220

Once the telemetry pots are set for max system deviation, you can adjust the actual telemetry level under software control. Each telemetry type has an associated level parameter. For example, this will allow you to set the CW ID to 20% of system deviation while the function complete and dialtone telemetry are set for 60% of system deviation.

Basic level sets are now done!





## Appendix

# Glossary

## **Radio Control Board**

One of the circuit boards plugged into the backplane, which provides the interface between the controller and a radio transmitter and receiver combination or any other audio signal source and destination, such as an IRLP node or telephone interface.

## **Port**

Generic term for one full duplex connection to the controller. This could be a repeater, a link to another site, and IRLP node, or a remotely controller transceiver (remote base).

## **Physical Port Number**

The Radio Control board farthest to the left in the card cage, and next to the CPU board is designated physical port number 0, the Radio Control board immediately to the right of physical port number 1, and so on.

## **Logical Port Number**

Users need to control the system by commanding specific ports, using the port number. For example, the user may want to disconnect Link 1 at the San Francisco Bay area Black Mountain site which links to another site at Crystal Mountain. He would issue the Link 1 OFF command to do this. There may be situations where the port number that a user associates with a particular link cannot be the same as the physical port number. As a system grows, or if there is a problem with the hardware for one of the links, it is undesirable to change the logical port number for a link to some particular other site, but it may be necessary to change the physical port number attached to that link. The controller provides a logical to physical port mapping that can be used to make the logical and physical port numbers different.

## **Repeater or Local Radio or Repeater Port**

A port which is configured to operate as a repeater. Signals coming in on its receiver will be repeated back out its transmitter.

## **Link or Link Radio or Link Port**

A port which is configured to operate as a link. Signals coming in on its receiver will not normally be repeated back out its transmitter, but only to other ports which are connected ("linked") to this port.

## **IRLP Port**

A port which is configured to operate as a restricted link, normally used for IRLP or other voice over IP connections into the site.

## **Remote Base Port**

A port which is configured to operate a remotely controlled transceiver. It invokes special telemetry, and is most commonly used in a simplex mode. The v0.90 software has limited support for this type of port.

## **Unlock code and Superunlock code or password**

Many controller commands cannot be executed until an unlock code is entered. This prevents unauthorized users from reconfiguring the system. There are up to 20 unlock codes, which are designated unlock code number 0



to 19. There are two levels of unlock code, designated unlock code and superunlock code. Unlock codes 0 and 1 are super-unlock codes. The rest are just unlock codes. A few commands require the super-unlock code, including the commands to change the unlock code table, commands to force reloading of default parameters, and commands to backup and restore the EEPROM parameter sets.

**Group**

A set of ports which are normally linked together, independently from ports which are members of a different group. Definition of groups allows the controller to manage several semi-independent repeaters. Group membership can be changed dynamically give great flexibility in connecting different radios together in different ways.

## Receive Qualification, Squelch and Transmit Timing

In most cases, the audio from the receiver is muted when no carrier is detected. In this case, the receiver muting via the audio switches is not critical in the controller. It's primary purpose is to avoid passing noise or other audio which opens receiver squelch when PL is not detected. In that case, if the audio is not muted, and the transmitters are switched on due to either telemetry or input from another receiver, then the undesired audio would also be passed to the linked transmitters.

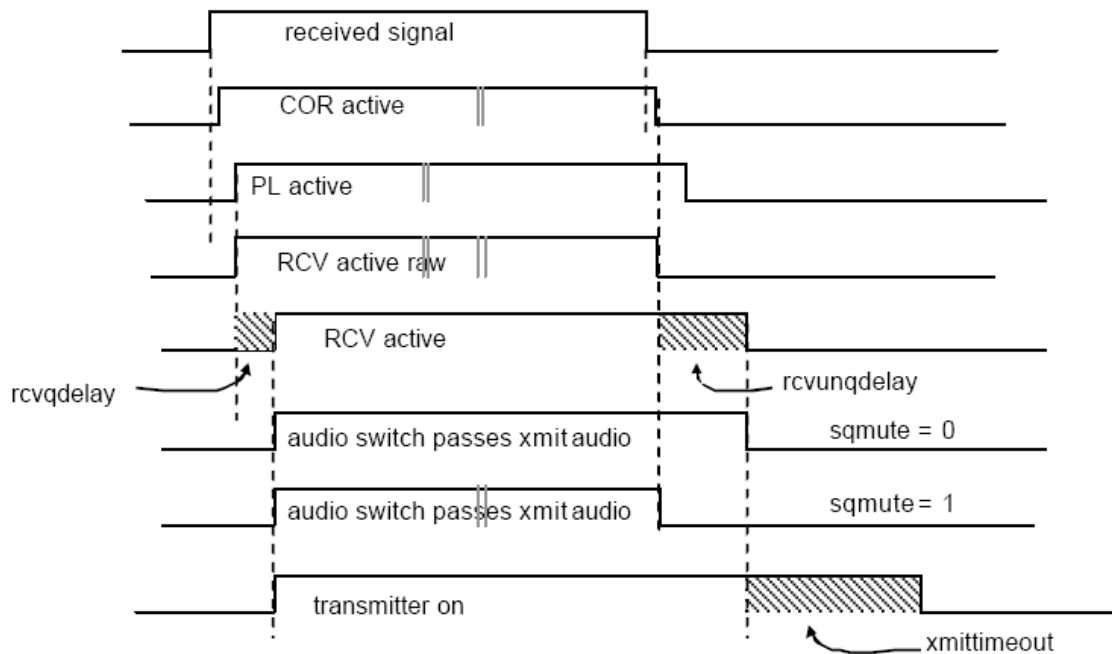
In other cases, the receiver and/or receiver interface provides COR and PL detect signals, but does not mute the receive audio. This means that the audio line from that receiver has a high level of broadband noise when there is no signal present. In this case, the switches in the controller radio card must provide the squelch function (even if only one receiver is on, the transmitter remains on for a few seconds after the incoming carrier is lost), and timing of the squelch operation is therefore critical to avoid passing high levels of audio noise.

It is desirable for the controller to have some mechanism for dealing with transient false COR or PL detect signals, which could be due to either noise, or short "kerchunking" of the repeater. This is implemented in the controller by requiring COR and PL to be present for a finite period of time before recognizing the received signal, switching on the audio and keying the transmitters. This delay is referred to as the receive qualify delay.

Several things happen when the received signal drops out. The audio switches are turned off to mute the audio from that receiver. The transmit key down delay is started, to turn off the transmitter after a delay of a few seconds. In the case of a local repeater input ("remote"), a key up telemetry delay is started to transmit key up telemetry after a delay of about a second. Incoming signals will often be marginal, due to rapid mobile flutter or weak signals from a distant handheld. It is desirable to avoid cutting off audio, transmitting key up telemetry, or cutting off the transmitters during a noisy transmission. This is handled by introducing a second delay after COR or COR and PL is lost before the controller recognizes that the received signal has dropped out. In the case of a link receiver, in which the transmitter and receiver are in a fixed location, with the link designed for high signal to noise, then this delay should be 0. In the case of a repeater input, it needs to be some fraction of a second, depending upon the behavior of the receiver COR, PL, and squelch circuits. This delay is referred to as the received unqualify delay.

Key up telemetry is sent following a delay after received signal from a local repeater drops out. An option is provided to abort the key up telemetry if the received signal comes back on before the delay timeout. This prevents momentary loss of signal during a mobile flutter from repeatedly sending key up telemetry, independent of the value of the receive unqualify delay. Both delays may be set independently for each receiver.

## Transmitter PTT Timing



A weak signal may not provide a reliable PL signal. If PL is on, this can cause the audio switches to mute the weak audio. The receive unqualify delay should be set to prevent this from happening, while still providing the protection from passing grunge or undesirable keying of the transmitter.

If the audio from the receiver is not squelched, a long squelch tail will be heard between the loss of carrier and the receiver unqualify delay timeout. There is an option to shut off the audio switch immediately on loss of carrier (squelch mute), to prevent this. Its undesirable side effect is that if the COR is not sufficiently sensitive, it may cut off the audio of a weak signal during fades. This option may be selected independently for each receiver.

## Telemetry Tones and Level Control

The amplitude of individual telemetry tones and CW can be adjusted after the digital pots levels are properly set. There are no specific commands to make these changes. The parameters must be changed by directly modifying the parameter value where it is stored in the serial eeprom.

As an example, lets change the amplitude of the CW ID for link ports. The default value is 40 in the range of 0 to 255. If the digital pot calibration procedure has been followed, with a system peak deviation of 5 kHz, then this tone will have a deviation of approximately  $40/255 * 5 \text{ kHz} = 784 \text{ Hz}$  deviation. Lets say you want to reduce the level to about 500 Hz deviation. To do that, the value should be  $255 * 500/5000 = 25$ . Lets also say the configuration eeprom address for the link ID CW amplitude is 192.

To read the current value, use the read eeprom byte command, C107. After entering an unlock code, you can verify the current value with

C107 192 D

This will return the current CW telemetry level between 0 and 255. To change the level to 25, use the set eeprom byte command C106.

C106 192 B25 D

The processor will then need to be reset (C000) for the new value to be downloaded to the Radio Control cards where the ID's are generated.

C000

Telemetry tone sequence amplitudes are slightly more complicated to adjust. To change the unkey telemetry "beep-boop" generated upon key up on a repeater port when there are links active and link delay is on, the "lnkintsnd" parameter needs to be changed. There are two forms of this parameter in the eeprom, a global default, which is used to simultaneously set the values for all of the ports, and individual values for each port.

Within the lnkintsnd parameter, there is a 16 bit integer which is an index into a table of sound sequences, and an 8 bit integer, which set the tone amplitude. To change the amplitude for all ports, use the C106 command to set the amplitude part of the lnkintsnd parameter (address 221 + 2) then issue the set rlnkintsnd command to transfer this value to all of the individual radio specific values, in rlnkintsnd. To set this amplitude (default is 127) to 80.

C106 223 B80 D

Then reset the processor to load it into RAM.

C000

C231

This is a somewhat complicated set of steps, which will be simplified in a future version of the PC based config software



## Software Architecture and Theory of Operation

This section of the manual is for those people interested in learning about the internal organization and behavior of the control system firmware. It is not necessary to understand all these details but will give you an deeper understanding of how it all works.

All primary repeater control functions are executed by the processor on the CPU board.

The processor on the Radio Control Board handles initialization of radio board hardware, such as gain control pots, and generation of radio specific tones (key up telemetry, CW ID, busy signal).

The software for the CPU board is compiled into a program with a name format of SRSCnnmms.hex. The main version number is nn. The minor version number is mm.

During development of any minor version, different passes of the code are given a letter, s. The v0.90 beta code is designated, SRSC0090. The binary executable to be loaded into the CPU board processor is in the file srscnnmms.hex. The software for the Radio Control board follows a similar naming scheme, SRSRnnmms. The binary executable to be loaded into the Radio Control board processor is in the file srsrnnmms.hex.

The controller has many parameters defining its configuration and operating behavior. Most of these can be easily changed by the site manager. These parameters have hard coded default values which are chosen so that most of the parameter values will be satisfactory for site operation without any further configuration.

The CPU and Radio Control boards have several forms of memory. Flash PROM on the processor chip contains the executable program. The size of this memory is 128kB, 64kB, and 32kB for the 18F8722, 18F4620, and 18F452 processors, respectively. Data RAM on the processor chip is used for working variables, program stack, etc. The size of this memory is 3982B for the 18F8722 and 18F4620, and 1536B for the 18F452. There is a nonvolatile data EEPROM on the processor chip. It is 1kB on the 18F8722 and 18F4620. Finally, there is an external serial EEPROM on the CPU board which is used to store configuration parameters. The size of this memory is 64kB, and the board has provisions for adding a second chip to double the memory. The hardware supports changing the contents of all of these forms of memory under program control.

### Memory usage for the CPU board

#### **Program ROM**

Executable code, library of sound sequences, command table

#### **On-chip data EEPROM**

Reset counters

#### **On-chip RAM**

Working values of configuration parameters, program working variables, stack, etc.

#### **Serial EEPROM**

Stored values of configuration parameters, stored values of unlock codes, stored and working table of macros.

### Memory usage for the Radio Control Board

#### **Program ROM**

Executable code and the library of sound sequences. digital pots settings, tone definitions and amplitudes, ID frequency, speed, and amplitude are downloaded from the CPU board when the system is reset. Most of the parameter set functions change the CPU board stored values, but do not download the values to the Radio Control Boards (digital pot setting is an exception, as it must be interactive).

Therefore, if the ID is changed, the system must be reset for it to be effective. There is space in the serial EEPROM for storing the working copy and two backups of the configuration parameters, unlock code table, and macro table. The parameter backup and restore commands are used to copy the data from one of these copies to the other.

### Configuring the System - Parameter Storage

Most parameters defining operation of the controller are represented by global variables of the CPU board program. They are stored in a serial EEPROM from which they are loaded on booting the program. The EEPROM contains three copies of the controller configuration parameters, designated working copy, backup, and deep backup. The working copy is generally synchronized with the RAM copy of the parameters. Some parameters can be set with an individual command to the CPU board via the RS232 serial port or DTMF received by one of the radios. These commands check parameters and most set both the RAM and EEPROM working value. Other parameters are set by writing directly to the corresponding address in the serial EEPROM. The working copy can be copied to and from the backup copy and the backup copy can be copied to and from the deep backup copy, via DTMF commands. Macros are stored only in the serial EEPROM, and not in microprocessor RAM, except as they are being executed. The commands for changing specific parameters do input range checking and change both RAM and eeprom, so the effect of the command may be determined immediately. Specific commands are not provided for changing many of the. There are commands to change the value of any address in EEPROM directly. This should not normally be invoked manually from DTMF, as a mistake can corrupt the EEPROM. When used over RS232 from a controlling program on an attached PC, it will provide a mechanism for efficiently configuring all parameters of the system. Some parameters are stored differently.

1. The command translation table, which translates a sequence of DTMF characters into a word length internal command, is stored in program ROM. Because the PIC processor is capable of reprogramming itself, this table will be able to be customized in the field. This capability is not available in program v0.90.
2. Macro strings are stored in the serial eeprom. They are included in the backups.
3. A tone sequence library is stored in program ROM. Pointers to the tone sequence library for particular telemetry sequences are stored in RAM and serial eeprom.

### **Initialization**

Default configuration parameter values are built into the CPU board program. The first time that the program is started after it is written to the processor chip ROM, it will load the default parameters into the serial EEPROM and into RAM. On subsequent resets of the system, the processor checks for a configuration valid byte in the EEPROM, and if it is correct, the configuration parameters are loaded from the serial EEPROM. A reset is caused by power cycling, hitting the reset button on the CPU board, by issuing the C000 command, or by the watchdog timer built into the processor detecting a program hangup.

A reload of the hard coded defaults can be forced on the next reset by resetting the EEPROM configuration valid byte. There is a command, C100, to do this.

The default parameter load takes about 12 seconds. A beep-beep confirmation is sent to port 0 when this is complete. If power is interrupted during the default load, it will start over again on the next reset. Subsequent resets take less than one second.

Any new program load will overwrite the configuration parameter working copy in serial EEPROM. If the program revision is a minor one with no changes to the EEPROM parameter memory layout, then previously configured parameters can be restored by using the restore command. For more significant program upgrades, this will not work. Therefore, a site manager should keep a record of changes made to the default configuration so that they can be quickly restored.



