The Centre for Research in Earth and Space Technology

A Centre of Excellence funded by a grant from the Ontario Technology Fund and other private, industrial, federal and international sources.

S2-PT User's Manual

Version 3.2 (133XT), October 1998

CRESTech-SGL-TR98-044

Table of Contents

List of Figures

List of Tables

1. INTRODUCTION

1.1 S2-PT User Documentation

The S2-PT User's Manual contains information required by users of the S2 Playback Terminal for installation, operation, maintenance and servicing of the system. This manual is for S2-PT systems which use an MVME133XT Recorder Control Computer board.

Please note the version number on the front cover. This manual is accurate only for S2-PT systems running the matching Recorder Operating System (ROS) software version number. To check which software version you are using, type `version` at the S2 console. For systems running older software (lower version numbers) you should use the appropriate older User's Manual, or upgrade the software. For systems running newer software (higher version numbers) you should check to see if a newer version of the User's Manual has been issued by CRESTech/SGL. If not, refer to the release notes for the software version you are using; they will contain documentation on additional features.

If you have any problems or questions related to the S2 please call CRESTech/SGL in Canada at (416) 665-5470 or send email to s2support@sgl.crestech.ca . We can be found on the World Wide Web at *http://www.sgl.crestech.ca/* . Up-to-date versions of this User's Manual are available at *ftp://ftp.sgl.crestech.ca/pub/s2/doc/userman/* for all four S2 system combinations (S2-RT & 162 CPU, S2-PT & 162 CPU, S2-RT & 133XT CPU, S2-PT & 133XT CPU). Note that the the name CRESTech was formerly ISTS (Institute for Space and Terrestrial Science) until it was changed in late 1997.

1.2 S2 Physical Description

The S2 consists of up to three modules: one or two Transport Array Modules (TAMs) containing up to four tape transports each, and one Data, Signal and Control Module (DSCM) containing most of the system electronics.

1.2.1 Transport Array Module (TAM)

The TAM, shown in Figure 1-1, consists of one to four S2 tape transports mounted in a custom rack which is designed either for tabletop operation or which may be mounted in a 19" rack. The transports are supported on sliders which are mounted to infill brackets within the rack. The sliders allow easy removal of each transport, and, with the use of extension cables, the operation of the transport in an easily-accessible position outside the rack. To lock the sliders in place for shipping and normal operation, a single removable "slider block" is used on the left-hand side infill bracket.

Figure 1-1: Transport Array Module (TAM)

The TAM racks have two user-removable panels. The front panel is held in place with four thumbscrews and is removed during normal operation. The rear vented cover is also secured by hand-removable thumbscrews, but is normally in place when the unit is operating. A fixed rear plate holds all power and signal connectors.

TAM Dimensions and Weight

Each TAM is approximately 40 (H) x 44 (W) x 46 (D) cm and weighs about 36 kg with four transports. 10 rack units are required for rack mounting.

1.2.2   Data, Signal and Control Module (DSCM)

The DSCM, shown in Figure 1-2, contains custom S2 electronics required to manage the system, format and control the data on record, recover the data on playback and control the transports. The DSCM is a semicustom VME cage with three standard VME-compatible slots for the system Recorder Control Computer (RCC), Static RAM board (SRAM) and optional Ethernet board, and 12 slots for electronics required to manage the data flow, format the data on record, decode the data on playback and interface to the user's hardware. These boards consist of an eight-channel Formatter (FORM) board, eight decoder or Data Recovery and Deformatter (DRDp) boards, and a User Interface (UI) board.

Figure 1-2: Data, Signal and Control Module (DSCM)

The DSCM is designed for tabletop or 19" rack mounting. Rack mounting may be made easier by removal of the two leg mounts screwed to the outside of the enclosure (replace the screws as they also hold on the fan tray).

DSCM Dimensions and Weight

The DSCM is 33 (H) x 44 (W) x 33 (D) cm and weighs approximately 17 kg. 7 rack units are required for rack mounting.

2. INSTALLATION

2.1 Introduction

The S2 recorder normally consists of three units requiring intermodule cable hookup as well as connection to external equipment for operation. Figures 2-1 and 2-2 illustrate the system with all signal and power cables connected.

To install the system, please follow these steps:

1. Configure the TAMs and DSCM in a 19" rack or on a bench in the desired arrangement. Please refer to Chapter 3 of this manual to establish environmental constraints on system configuration and operation.

2. Identify the cables and cable ports (described in the next two sections).
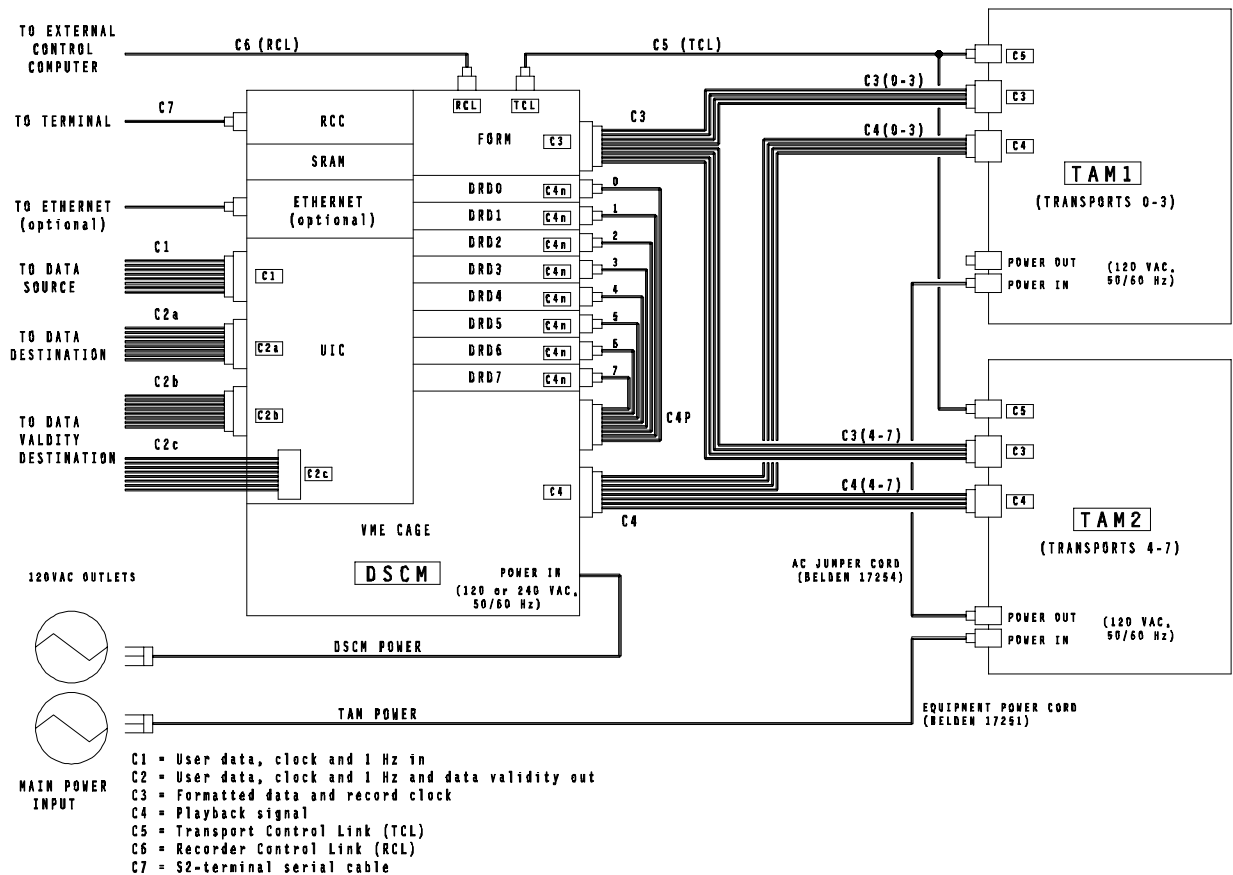
Figure 2-1: S2-PT Configuration, 120VAC Operation
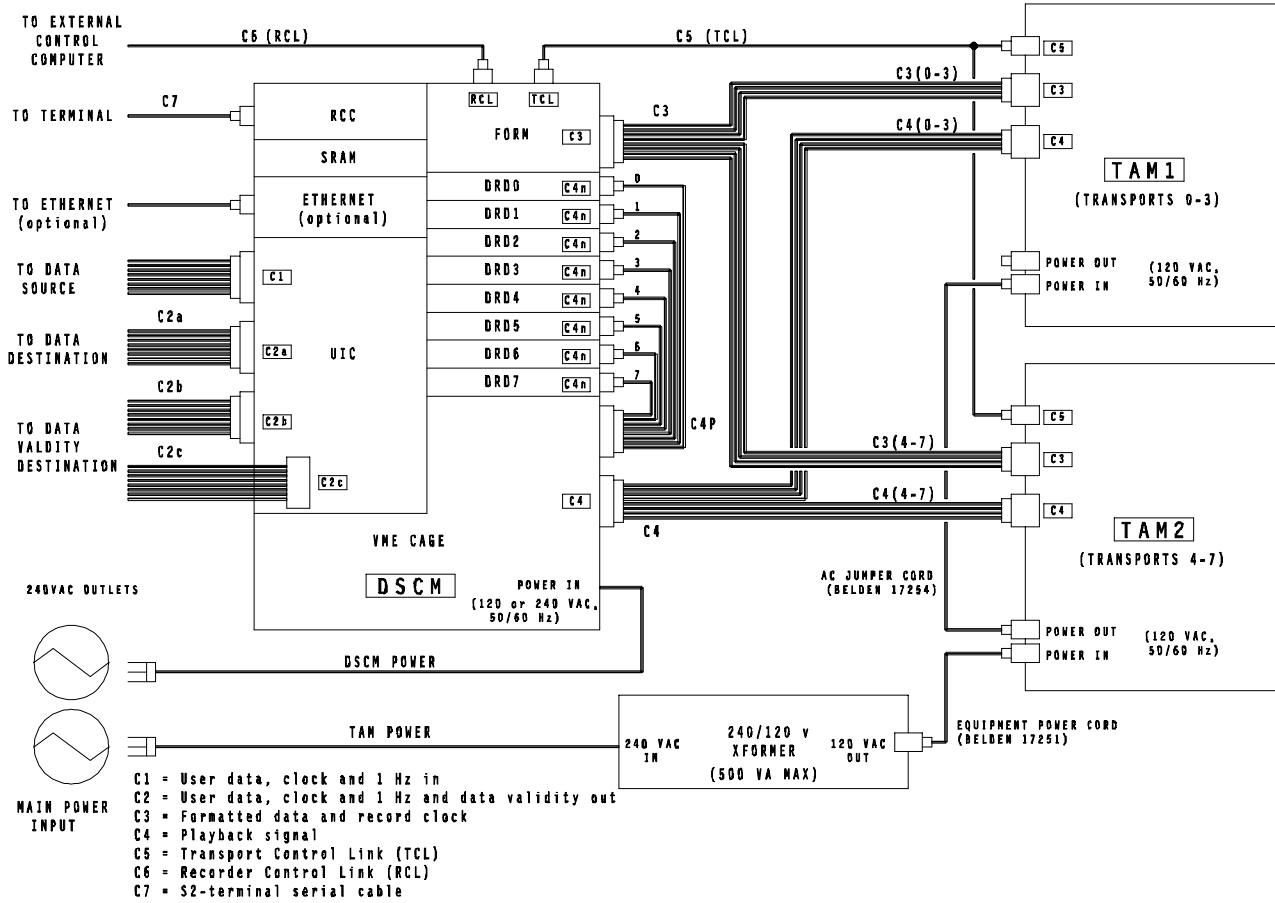
Figure 2-2: S2-PT Configuration, 240VAC Operation

3. Connect the cables as shown in Figure 2-1 for 120 VAC operation and Figure 2-2 for 240 VAC operation. Note that the DSCM power input port accepts either 120 VAC or 240 VAC, but the TAM power inputs accept **120VAC only**. Operation of the TAMs from a 240 VAC power source therefore requires 240 VAC to 120 VAC voltage conversion, shown in Figure 2-2 as a separate transformer.

4. Connect a console terminal such as DEC VT100 to the console port (C7). The terminal should be configured for 9600 baud, 8 data bits, one stop bit, no parity. An IBM PC or compatible computer can be used as a console by running the ROSCO terminal emulator program supplied with the S2 (see Section 6.2 for more information on ROSCO).

5. Power on the system and answer the installation questions which appear on the console (first time system turned on only). See Section 4.2.6 for more information on the installation questions.

6. Perform the post-installation checkout tests as described in Section 2.8.

2.2 Identifying Cable Ports

2.2.1 DSCM Ports

Figure 2-3 illustrates the cable ports on the front panels of the DSCM circuit boards. These are:

RCC Console Port: This port connects the RCC to a console terminal via the C7 serial cable. It is a female DB-25 connector configured as RS-232C DCE (Data Communications Equipment).

Ethernet Port: (Optional) BNC connector for thin Ethernet, or female DB-15 connector for Ethernet transceiver. These ports are not both active simultaneously; the chip in socket U4 of the Ethernet board must be moved to the adjacent socket U3, or vice versa, to enable the other Ethernet connector. The factory default setting is for the BNC (coaxial) connector to be enabled.

TCL Port: This DB-9 pin port is the DSCM input for the C5 Transport Control Link (TCL) connecting the DSCM and TAM.

RCL Port: This port is the DSCM input for the C6 Radioastronomy Control Link (RCL). It is a male DB-25 connector configured as RS-232C DTE (Data Terminal Equipment).
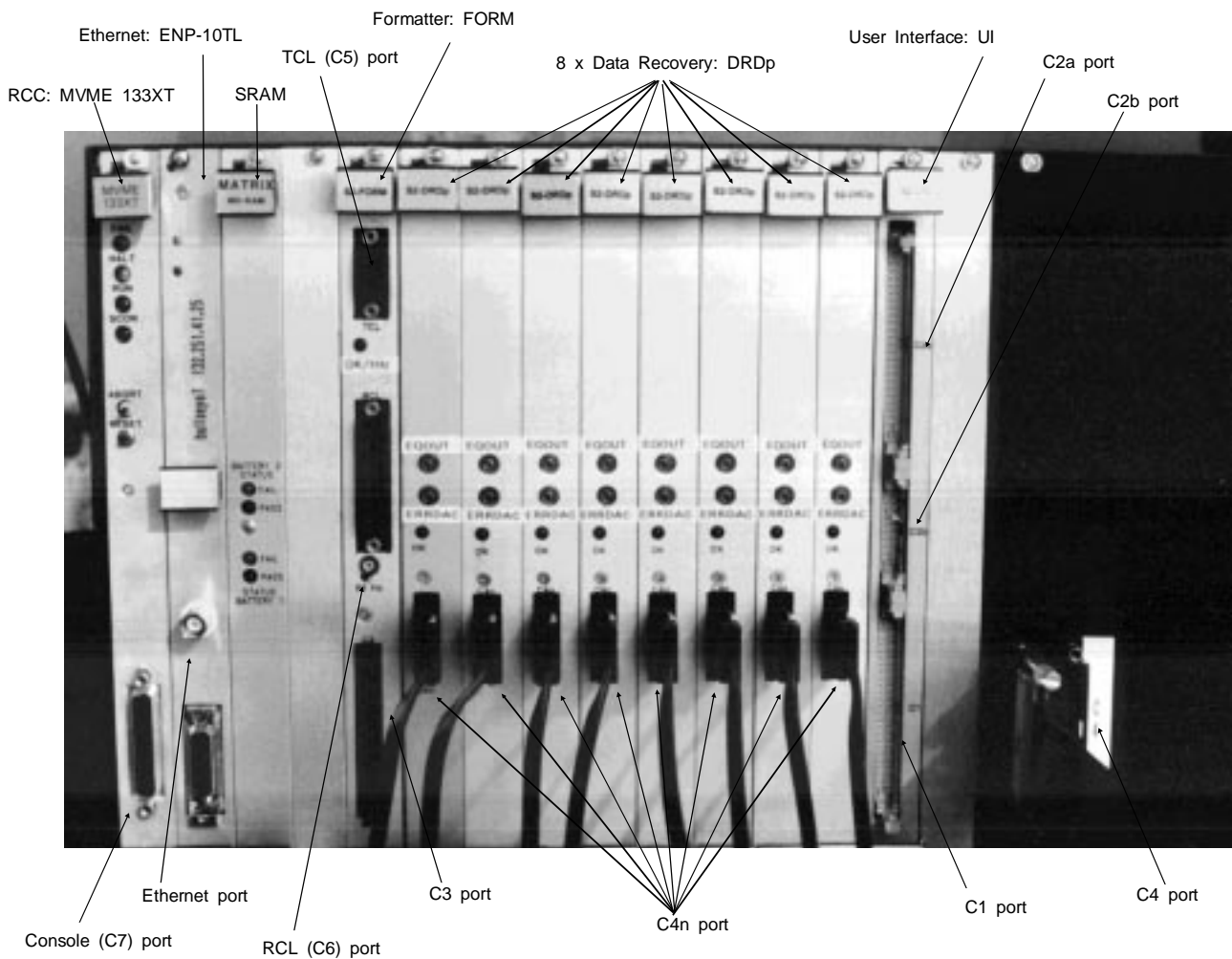
Figure 2-3: DSCM Ports

C1 Port: This port originates the C1 cable from the data source to the DSCM, bringing clock, data and 1 Hz to the User Interface.

C2a Port: This port originates the C2a cable from the DSCM, bringing clock, data and 1 Hz from the User Interface to the user data destination; it also carries optional clock and 1Hz from the user data destination to the User Interface.

C2b Port: This port originates the C2b cable from the DSCM, bringing data validity from the User Interface to the user data destination; this cable carries validity appropriate for 8-channel 1 or 2-bit quantized data from the S2 to the user.

C2c Port: This port originates the C2c cable from the DSCM, bringing data validity from the User Interface to the user data destination; this cable carries validity appropriate to the odd channels of 16 independent 1-bit quantized baseband channels. Note that the C2c port is not accessible from the UI front panel, but is available from a straight dual-row header near the front and center of the UI board itself.

C3 Port: This port originates the C3 cable from the DSCM, bringing clock and data from the DSCM Formatter to the TAMs.

C4 Port: This port originates the C4 cable from the DSCM, returning signal from the TAMs to the eight DSCM Data Recovery modules and bringing ±5 V from the DSCM to the custom transport electronics.
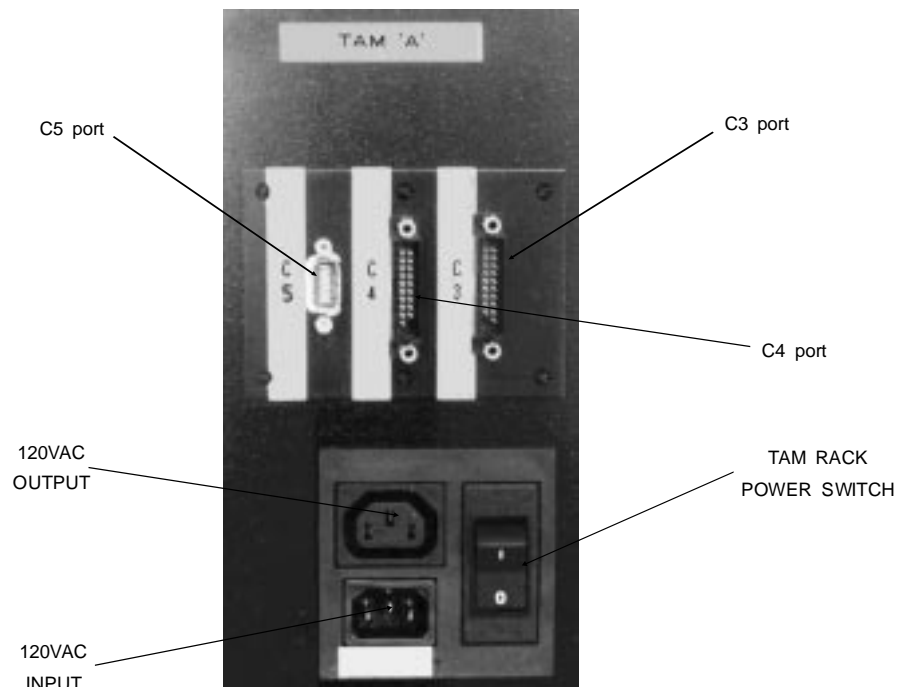
Figure 2-4: TAM Rack Connector Placement

2.2.2   TAM Ports

Figure 2-4 illustrates the cable ports on the fixed back panel of the TAM. These are:

C3 Port: This port terminates the C3 cable from the DSCM, bringing clock and data from the DSCM Formatter to the TAM.

Figure 2-5: DSCM Shipping Box with Cable Assemblies

C4 Port: This port terminates the C4 cable from the DSCM, returning signal from the TAM to the eight DSCM Data Recovery modules and supplying ±5 V from the DSCM to the TAM custom electronics.

C5 Port: This port terminates the serial Transport Control Link (TCL) cable from the DSCM.

Power In/Out: This port supplies power to the TAM, and includes a connector for feed through power to a second TAM. This port is rated for 120 volts (50-60 Hz) only.

2.3   Cables

Figure 2-5 shows the S2 cable assemblies as shipped in the DSCM container. They consist of S2-to-user I/O cables for transfer of signals and power to and from the user and S2 system cables linking S2 modules. Connect the cable assemblies as shown in Figure 2-1 or 2-2. See Appendix B for detailed pin assignments, timing and electrical specifications.

2.3.1   S2 User Cables

C1:   This cable consists of a 50 conductor twisted pair ribbon cable (maximum length = 10 m) carrying user data, clock and 1 Hz signals from the user data source to the S2.

C2a: This cable consists of a 50 conductor twisted pair ribbon cable (maximum length = 10 m) carrying recovered data, clock and 1 Hz from the S2 to the user.

C2b: This cable consists of a 20 conductor twisted pair ribbon cable (maximum length = 10 m) carrying recovered data validity for 1 or 2-bit quantized data (carries only even user channels for 1-bit data).

C2c: This cable consists of a 20 conductor twisted pair ribbon cable (maximum length = 10 m) carrying recovered data validity for the odd user channels in the case of 1-bit quantized data. It is needed only for 1-bit quantized modes that use odd user data channels. Note that the C2c cable is not accessible from the UI front panel, but is available from a straight dual-row header near the front and center of the UI board itself (this header is unlabeled). The C2c header is polarized to accept only the wider end of the physical C2c cable assembly. The blank panel on the right-hand side of the UI must be removed to install this cable.

C6: This cable, also known as the Radioastronomy Control Link (RCL), is a serial cable connecting the S2 to an external control computer. The S2 RCL port is configured for DTE operation, so the C6 cable needs to be a null cable (exchange pins 2 and 3) if the external computer is also acting as DTE. This is the case for IBM PCs and compatibles. The external computer should be configured for 19200 baud, 8 data bits, one stop bit, no parity. The RCL baud rate can be set up to 57,600 by editing the system defaults file (see Section 4.2.7).

C7: This cable is a serial cable connecting the S2 to a console terminal. The S2 console port is configured for DCE operation. The terminal should be configured for 9600 baud, 8 data bits, one stop bit, no parity. The console baud rate can be changed by editing the system defaults file, but the maximum allowed rate is 9600 baud.

2.3.2 S2 System Cables

C3: C3 is a 50 ohm ribbon cable transporting clock and formatted data from the DSCM to the TAMs. The cable is terminated via a single connector at the DSCM FORM board, and is split in the middle to allow the cable to service two TAMs.

C4: C4 is a 50 ohm ribbon cable transporting return signal from the TAMs to the DSCM. The cable is terminated via a single connector at the DSCM rear panel, and is split in the middle to allow the signals to come from two TAMs.

C4p: C4p is a continuation of the C4 cable required to split the signals to and from the TAMs into separate channels for the eight DRDp boards in the playback terminal. The eight split ends are labeled according to channel number and should be terminated on the DRDp with the corresponding channel label and address jumper setting.

C5: C5, the Transport Control Link (TCL), is a serial cable connecting the TAMs and the DSCM.

NOTE: Insertion keys are present on the C3 and C4 cables to prevent accidental interchange as this may result in damage. It is recommended that the C3 and C4 cables be inserted with the power off if possible. If power must remain on exercise care and do not force the connectors.

2.4 Power Requirements

Total S2-PT power requirements are 510 W: 230 W for the DSCM and 140 W for each of the TAMs. At least 700 VA should be available in order to supply sufficient current when the transports are most active, which normally occurs when tape is loaded onto the transport scanner.

2.5 Power cables: 120 Volt Operation

The S2 units may be powered from any standard 120VAC wall outlet capable of supplying the total current requirements of the S2. The system is supplied with three power cords; the recommended power cord arrangement is shown in Figure 2-1.

2.6 Power cables: 240 Volt Operation

For operation from a 240 VAC power source, an external 2:1 transformer must be used to supply the TAMs with 120 VAC, as indicated in Figure 2-2.

The DSCM contains an auto-ranging power supply and may be powered directly with 240V as shown. The DSCM is supplied with an incomplete power cable to which a suitable plug can be attached. Colour coding of these cables is as follows:

> Brown (or Black) - AC Line
> Blue (or White)   - AC Neutral
> Green   - Chassis Ground

2.7 External Transformer

A schematic for a 240/120 VAC transformer circuit is shown in Figure 2-6. It consists of a 500 VA 2:1 transformer. The 240 volt supply current is limited by a 3 A slow-blow fuse. The 120 volt output current is limited by a 4 A fuse.
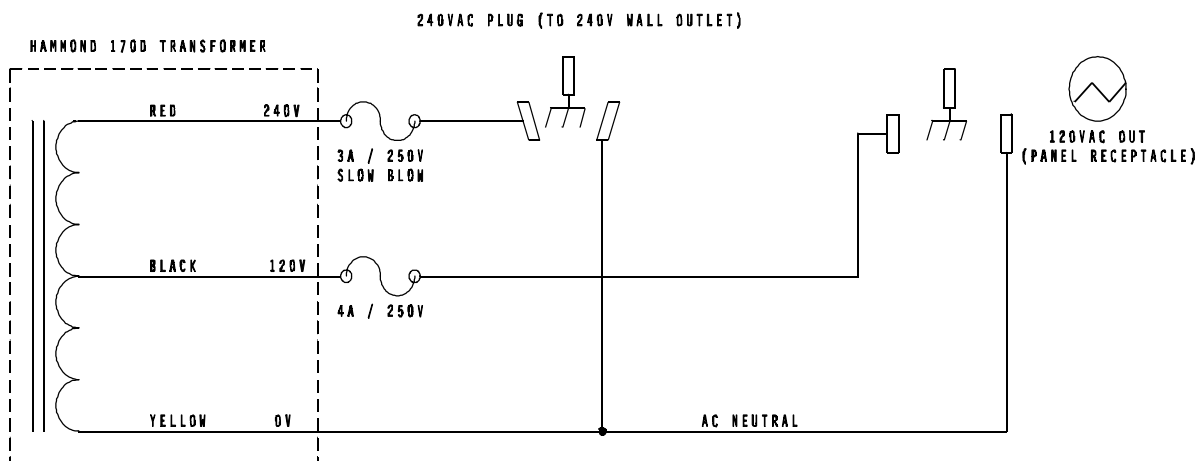
> **Warning!** Transformer circuit is designed to power one S2 (i.e. two TAM racks) only. Both TAM racks together require a supply of about 400 VA peak.

Figure 2-6: 240V to 120V Transformer Circuit

```
 S2 Recorder |        User Parms (playback)  | rec24     1998  217-19:18:56 ▓▓
 /\ Operating|                               |
    System   | Tape ID:                      | Recorder mode : 16x8-1 (br)
-------------|                               | Station delay : [no 1 Hz in]
 Transports  |                               | Diag dec 0 in : channel 0 play
0*    []     |                               | Diag dec time : XXX-XX:XX:XX.XX
1*    []     |                               | Diag dec sync : 0% (uncalib)
2*    []     |                               |
3*    []     | User data valid:              | Est err rate : >1.0e-2 (8 ch)
4*    []     | Tape speed:                   | % data valid : 0.00%   (128 Mb/s)
5*    []     | Tape position: 0:00:01 +-01 s | Decoder Xbar : 01234567
6*    []     |-------------------------------|----------------------------------
7*    []     |  <<    >    >>   ██[]   ||  REC|
-------------------------------------------|
                                           |
                                           |----------------------------------
                                           |
> █                                        |

--------------------------------------------------------------------------------
[help   ] [time   ] [mode   ] [tapeid ] [user   ] [record ] [play   ] [--ETC--]
```

Figure 2-7: ROS Normal Display on Bootup (PT)

2.8   Post-installation Checkout Tests

Following transportation of the S2, built-in test facilities may be used to verify the functionality of the system prior to connection with any external hardware other than a console terminal.

2.8.1   System Setup

Before testing, the system should be set up according to the installation instructions earlier in this chapter. No cables are to be connected to the C1 or C2 signal ports. The console terminal should be configured for 9600 baud, 8 data bits, one stop bit, no parity.

2.8.2   System Signal and Electrical Test

Power up the system. A self-test executes as part of the bootup and should take about one minute to complete (you may first be asked to enter your console terminal type and baud rate). Verify that the console display following the self-test is as shown in Figure 2-7 (there may be minor differences, such as ''no tape'' in the Transport window). Verify that the time in the upper right corner of the screen is incrementing and that there are no detected faults displayed in the status window (middle right).

Failure of the power-on self-test causes the test to terminate and print a descriptive message to the console interaction window. The status window will indicate ''Diagnostic test failed'' with a positive numeric diagnostic code or negative error code.

2.8.3   Record/Playback Performance Test

This test allows users to verify that the S2 is functioning as specified, i.e. that data is recorded and reproduced at a specified bit-error rate of less than 1×10^{-4} (speed LP) or 1×10^{-3} (speed SLP, average) on interchange.

For speed SLP (which must be used on systems with AG2550 transports) users should check that the record/playback performance test gives a bit-error rate of less than 1×10^{-3} averaged across eight transports, or 3×10^{-3} individually. Success of this test verifies that the record and playback signal systems are operating according to specification and that the mechanical alignment of the transports has not been disturbed during shipping. The time required to complete the test is about 90 minutes.

Before starting the record/playback test, insert writable tapes into all eight transports. For information on what kind of tape can be used in the S2 please refer to Chapter 5. You should ensure that the tape type setting is correct and matches the type of tape you are using for the test by entering the command

```
tapetype N default
```
where N is one of the tape type codes given in Chapter 5. Then run the batch file **berlp.tst** for speed LP or **berslp.tst** for speed SLP by typing one of the following commands

```
batch berlp.tst
batch berslp.tst
```

The batch file sets up the system as follows:

- S2 mode set to 16x8-1

- System Clock PLL mode set to **xtal**

- formatter diagnostic sequence on

The batch file then records a pseudo-random test data sequence for 30 minutes. Following the recording the tapes are rewound and played back, and bit-error rate measurements are made on each of the eight tape channels. These measurements along with any detected status errors are recorded in the S2 internal log file. When playback is complete, any status errors recorded in the log file during the record or playback phases are displayed. You will be prompted to press return, then the accumulated bit-error rates for three minutes per channel of playback data are displayed. There will be 8 lines of output of the form

```
     0022  E  189-12:15:23  174-10:51:31   1.23e-5   0   0   11520
```

The fourth column from the right shows the bit-error rate, followed by the data channel number and transport number. Verify that the error rate is less than 1.0e-4 for each transport if LP speed was used (**berlp.tst** batch file). If SLP speed was used (**berslp.tst** batch file) verify that the error rate is less than 3.0e-3 for each individual transport and that the average over all 8 transports is less than 1.0e-3. Not all 8 output lines fit on the screen at once so you will have to press return. It is a good idea to write down these bit-error rate numbers for future reference.

In order to test tape interchange, eject the tapes and rotate them by one position, i.e. the tape recorded in transport 0 is inserted into transport 1, the tape recorded in transport 1 is inserted into transport 2, and so on (the tape recorded in transport 7 is inserted into transport 0). Then run the batch file **inter.tst** by typing the command `batch inter.tst`. This batch file rewinds the tapes and repeats the playback phase of the **berlp.tst/berslp.tst** batch file. As above, verify that the error rate displayed for each channel and/or average is less than the specified value.

Failure to achieve specified performance may call for particular transports or other system components to be serviced; see Chapter 6 for maintenance information. In particular, poor performance in this test may indicate a misalignment of the transport tape guide posts (Section 6.4.6).

3. OPERATING ENVIRONMENT

Temperature and Humidity

Temperature Range:

> For transport operation in
> TAM rack: +10 degrees to +30 degrees Celsius[1]
> For specified performance: +15 degrees to +25 degrees Celsius[1]

Rate of Temperature Variation: Not more than 1 degree Celsius per minute, and
                                    not more than 10 degrees Celsius per 30 minutes

Relative Humidity: 35% to 80% non-condensing

Particulate Contamination

Particle size, μm	Particle count/m^3
< 1.0	4 x 10^7
1.0 to 1.5	4 x 10^6
1.5 to 5.0	4 x 10^5
5.0 to 10	4 x 10^4
> 10	4 x 10^3

---

[1] *This refers to the ambient temperature around the TAM rack. If the TAMs are mounted in a 19" rack or other enclosed space, ventilation must be provided to ensure that this condition is met. Temperature around each transport should be within +10 to +40 deg. C. Temperature inside each transport must be within +5 to +60 deg. C or else thermal shutdown software will be triggered.*

4. OPERATION

4.1 Overview

This chapter describes how to operate the S2 Playback Terminal. After some preliminary information and definitions, a summary of basic operational procedures is given in Section 4.2.4. Detailed descriptions of screen displays and console commands follow in Sections 4.2.8 and 4.2.9, using more of a reference-manual format. Ethernet network information is in Section 4.4, and all S2 status codes are listed and described in detail in Section 4.5.

The S2 Recorder can be operated using either one of two alternative interfaces: the console terminal, or the Radioastronomy Control Link (RCL). Both of these can use RS-232C serial links, but the RCL can operate as an RS-485 multi-drop link with the addition of an external conversion box, allowing more than one device to be controlled from the same serial port. On S2 systems equipped with an Ethernet board, console and RCL connections can be made using network rlogin/telnet and socket protocols, eliminating the need for serial cables. The console and the RCL may be used at the same time as long as operations don't interfere — for example the console display can be used to monitor what is happening when the S2 is under automatic control via the RCL.

The console allows high-level manual control of the S2 for regular operation, plus low-level access for testing and other unusual operating conditions. Any cursor-addressable terminal or terminal emulator, such as DEC VT100, can be used as a console (a VT100 emulator program for IBM PCs and compatibles called ROSCO is supplied with the S2). The console display is divided into multiple windows which are continuously updated. Command entry is aided by a soft-key system which uses function keys and command completion to make commands faster to type and easier to remember. There are approximately 50 console commands, of which only about 10 are needed for everyday operation. All of Section 4.2 is written from the point of view of console operation (the bulk of this chapter), but most of the principles described can be applied to RCL operation as well.

The Radioastronomy Control Link (RCL) is intended for automatic control of devices by an external computer. It can be used to control S2 recorders according to a correlation schedule running on the external computer. The RCL uses a reliable binary packet protocol to communicate requests from the external computer to the S2 and return responses. Source code for the packet assembly and disassembly routines needed to run on the external computer is provided with the S2 system, so the application programmer deals only with a simple function-call interface, and does not need to worry about the details of the RCL protocol. The RCL interface library routines are written in ANSI C and should be easy to port to many different types of computers. As shipped they work on IBM PCs and compatibles, or on the following flavours of Unix: SunOS, Solaris, and Linux. The commands available over the RCL are a subset of the console commands, and are intended mainly for high-level control during automated operation. The RCL tries to present a view of the S2 as a single tape recorder — not 8 individual transports. Consequently, most commands and responses make no mention of individual transports. Section 4.3 gives some additional general information on RCL operation, but most of the detailed information is in Appendix A, the RCL protocol specification.

4.1.1 Terminology

Because so many different kinds of control and status information are processed by the S2 software and hardware, it is necessary to use very precise terminology when describing things. Words that are normally used imprecisely and interchangeably have been given concrete definitions in the context of the S2 Recorder.

"Channel"   Channels are signal paths carrying sampled data. An important distinction is made between *user* channels and *internal* S2 channels. User channels carry data as sup-

plied by the user at one of several different data rates, with up to 16 user channels available depending on the current S2 mode. 2-bit quantized data occupies 2 user channels (bit streams), and is generally treated like two channels of 1-bit quantized data except that during playback only one Data Validity signal is used for both sign and magnitude. Internal S2 channels carry data at a fixed rate of 16 Mbits/s (or slightly higher when the data includes format overhead). There are 8 internal channels (one per transport) for a total of 128 Mbits/s.

"Mode"
The mode describes the current user data rate and channel allocation of the S2. It is sometimes called "observing mode" or "recorder mode". Each mode has a specific number and arrangement of active user data channels at a data rate of 4, 8, 16, or 32 Mbits/s per channel. The product of the number of channels and the channel data rate gives the total system data rate, which can be anywhere from 16 to 128 Mbits/s in multiples of 16 Mbits/s. Dividing the total data rate by 16 gives the number of transports needed for operation in a particular mode. In addition to regular observing modes, there are a number of test modes for checking C1/C2 cable connectivity and general system operation.

"Group"
In modes that require less than 8 transports, the active transports are normally selected in order starting at 0. For example, in a mode requiring 4 transports, transports 0–3 would be selected. This group of transports is called "group 0". Other group numbers can be specified, thus selecting a different set of transports. For example, group 1 would be transports 4–7. In this case there are just two groups (0 and 1), but in modes with lower total data rates there may be 4 or even 8 groups (see Figure 4-4, Page 4-12). Together the mode and group number determine which transports are selected for use.

"State"
This refers to the tape motion state of the S2. There are 11 states, the first 7 of which will be commonly encountered in everyday use: Record, Play, Position, Rewind, Fast-Forward, Stop, No-Tape (tapes ejected), Play-Pause, Record-Pause, Cue, Review. The S2's overall tape motion state is derived from the individual states of the currently selected transports. Under unusual circumstances it is possible that the individual transport states cannot be clearly summarized, in which case the overall state is considered *inconsistent*.

"Setup"
This describes the current internal diagnostic signal path configuration of the S2 Recorder. The setup is normally selected automatically by the S2 software, but can be manually specified using the console `setup` command when the S2 is in manual playback or record. The setup determines which of several internal diagnostic loopback paths is currently active (**play, record, bypass, ahook,** or **dhook** — see Figure 4-5 on Page 4-19), and which of the 8 transports/decoders is selected for diagnostic monitoring. In an S2-RT only one decoder (Data-Recovery Deformatter) is available so this must be switched among the active transports for complete diagnostic coverage. Even in an S2-PT with 8 decoders there is only one diagnostic signal path because of other hardware and software design factors. The decoder in the current diagnostic signal path is referred to as the "diagnostic decoder" or "diagnostic DRD". The setup also controls whether the system timing is set for playback or record. Playback timing is used when playback data from the transports is being decoded (**play**), and record timing is used at all other times (**bypass, record, ahook, dhook**). The setup used when the S2 is stopped (or rewinding, etc.) is known as the *stop setup* and is treated specially in that it is variable. The stop setup is normally **bypass** on an S2-RT and **play** on an S2-PT. This ensures that the system timing does not get disturbed unnecessarily when going between play/record and stop, which may be desirable if external systems are locked to the output 1 Hz tick.

"Configuration"
This term refers to all other hardware switch settings (and most software switch settings) besides setup.

"Error"   All commands and actions have an associated error return code which describes the success or failure of the particular operation. Error codes are negative numbers in the range -1 to -128, each associated with a mnemonic and a one-line text description. Normally only the text description is shown, but in certain cases where there is not enough room to display the message the numeric code is used. Error codes can be manually translated to the appropriate message with the console `error decode` command, or the RCL ERROR_DECODE command. For a complete list of error codes see Page A-32.

"Status"   The S2 software continuously gathers and reports status information which describes the overall health and operational condition of the system. On the console, this information is displayed in the 3-line status window to the right of the center of the screen. Over the RCL, it is obtained via the STATUS command and is the principal method of monitoring system activity. Status is distinct from errors in that it represents conditions that arise after a command has executed or result from interactions between multiple commands, so that error returns cannot be used to convey the necessary information. It is sometimes called "online status" to emphasize that it is gathered continuously and not the result of a specific diagnostic test or procedure. Status codes are positive numbers in the range 1 to 127, each associated with a mnemonic and a text description. The console `status decode` command or RCL STATUS_DECODE command can be used to translate a status code to its corresponding text message. Three status severity levels are defined: *informational*, *error*, and *fatal error*. In addition, status codes representing transient or instantaneous conditions are classified as *clear-on-read*, meaning that they are no longer relevant after they have been read. Each status code has a "short" and a "long" version of the message. The short message is limited to 34 characters and is typically used on the console. The long message is limited to 399 characters (but is usually shorter) and is typically used on the RCL. For a complete listing and explanation of all status codes see Section 4.5.

"Diagnostic"   The S2 software has certain diagnostic sequences, such as the power-on self-test, which evaluate system functionality and whose primary result is a single numeric code. This code either indicates that the diagnostic test passed (DIAG_OK) or that it failed, with the code number indicating the type of failure. Diagnostic codes are positive numbers in the range 1 to 255 and each is associated with a mnemonic and a text description. The console `diag decode` command can be used to translate a code to its corresponding text message. When a diagnostic code appears as part of a status message, the diagnostic message is included in the long-form status message from `status decode`.

```
S2 Recorder |       User Parms (playback)   | rec24    1998  217-19:18:56 ▓▓
/\ Operating|                               |
   System   | Tape ID:                      | Recorder mode : 16x8-1 (br)
------------|                               | Station delay : [no 1 Hz in]
 Transports |                               | Diag dec 0 in : channel 0 play
0*   []     |                               | Diag dec time : XXX-XX:XX:XX.XX
1*   []     |                               | Diag dec sync : 0% (uncalib)
2*   []     |                               |
3*   []     | User data valid:              | Est err rate : >1.0e-2 (8 ch)
4*   []     | Tape speed:                   | % data valid : 0.00%   (128 Mb/s)
5*   []     | Tape position: 0:00:01 +-01 s | Decoder Xbar : 01234567
6*   []     |-------------------------------|--------------------------------
7*   []     | <<   >   >>   ■[]■  ||   REC   |
------------------------------------------- |
                                            |
                                            --------------------------------
> ■



-------------------------------------------------------------------------
[help    ] [time    ] [mode    ] [tapeid ] [user    ] [record ] [play    ] [--ETC--]
```

Figure 4-1: Sample ROS Screen, User (stopped)

4.2 Console Operation

The following sections describe operation of the S2 Recorder using the console. General information and procedures are given first, and should be read carefully when first learning about the S2. Detailed descriptions of screen displays and console commands follow in Sections 4.2.8 and 4.2.9.

4.2.1 Connecting the Console

The S2 console port is wired as a DCE (Data Communication Equipment) device and should be connected to a DTE (Data Terminal Equipment) device. Any cursor-addressable terminal or terminal emulator, such as DEC VT100, can be used. For information on setting the terminal type and baud rate when first getting started, see Section 4.2.6. On S2 systems that have an Ethernet board it is possible to connect to the console over the network using rlogin or telnet, although a regular terminal is still required for initial configuration the first time you turn on the S2 (see Section 4.4.2).

4.2.2 Display Organization

The console display is normally divided into 8 different regions or windows (see Figure 4-1). The large area at the bottom of the display is the *interaction* window, which shows user commands as they are being typed. The window contents scroll up and off the top in the usual manner. The line at the very bottom of the screen contains eight *soft-keys*. These correspond to function keys f1 through f8 on terminals which have function keys, and show the current command and parameter alternatives at any time when typing a command. The remaining console windows contain information in various categories and are continuously updated at a rate of once per second. These windows are described in general here — for detailed information refer to Section 4.2.8.

At the upper right-hand corner of the display is the *Crucial Parms* window. It contains important parameters such as the system time and channelization mode, and various results from the diagnostic decoder (DRD). Below the Crucial Parms window, to the right of the middle of the screen, is the *Status* window. It contains one-line messages describing overall system health. The word FAIL appears highlighted on the bottom window boundary if any fatal error conditions exist, providing a single OK/FAIL indication for the entire system (the green front-panel LED

also acts as an overall OK/FAIL indicator when set to OK mode, i.e. when not blinking). Although there is room for only 3 messages in the window, status conditions are ranked by importance so that the most relevant information is usually at the top. If necessary, you can page through the status information with the `status pagedn` and `status pageup` commands, or use the entire screen for the status display with `status display fullscreen`.

The window at the upper middle of the screen is the *User Parms* window. It shows user-specified operational parameters such as the tape speed, the tape ID, the current tape position, and other tape information. All of these parameters except tape position change depending on whether the S2 is in record or playback. During playback (or when the S2 is stopped), the User Parms window shows information as extracted from tape. During record, the information shown describes what is being recorded on tape. The area of the screen normally occupied by the User Parms window can be switched to show several other displays, mostly containing diagnostic information. The alternate display of most interest to regular users is the *User I/O* window, selected by typing `userio`. It gives details about the current input/output interface configuration, including exactly which user data channels are active and what system clocking options are currently in effect.

Below the User Parms window, in the middle of the screen, is a small window which indicates the current overall tape motion state of the S2. Highlighted symbols are used to indicate whether the S2 is stopped, playing, rewinding, etc. This state indication is a summary of the states of the individual transports shown in the transport window to the left of the middle of the screen. During tape positioning the word "POSITIONING" will appear on the bottom boundary of the state window. If the state is playback and the tapes are aligned, the word "ALIGNED" will appear.

The *Transport* window shows the state of each transport in the system. An asterisk ('*') marks those transports selected for use under the current mode and group settings. During record and playback, an 'L' indicates that the transport head drum servo is properly locked to the system timing reference. If a transport is playing and locked, and the decoder has been able to extract playback tape time, then the tape time is shown. In addition to transport state and tape time, the Transport window can be switched to show many other parameters related to the transports or extracted from tape during playback (see Page 4-24). If the system status indicates one or more transport failure conditions, affected transports are marked with a highlighted 'F' on the right-hand boundary of the Transport window.

The last window is a small one in the top-left corner of the screen . Its main feature is an activity indicator that changes each time the screen is updated. This can be used to check that the S2 software is still running and that the console terminal is properly connected. The station number, when non-zero, is shown in this window just below the activity indicator (set it with the `station` command).

4.2.3   Console Command Entry

S2 console commands follow the conventional format of a command word followed by parameters. However, several productivity enhancement features are provided to make commands quicker and easier to enter. These features include soft-keys, command completion, input-line editing, command history, and batch files.

4.2.3.1   Soft-keys

At any point while typing a command, the current keyword and parameter alternatives are shown in the 8 soft-key labels at the bottom of the display. Keywords are shown in lower-case, and can be entered in one step simply by pressing the function key (f1–f8) corresponding to the soft-key position. If you choose to type the keyword, after each letter only those keywords which match what has been typed so far will remain on the display. When only one remains, pressing the space bar or tab key will complete it. In addition to keywords, numeric and string parameters appear in the soft-key display as "NUM" and "STR" (or sometimes "NNN" and "SSS"). Most such

parameters include a word in angle brackets that describes the quantity to be entered, e.g. "NUM<year>" indicates a year number.

Whenever there are more than 8 possibilities, soft-key 8 will show "--ETC--", indicating that further alternatives can be obtained by pressing function key f8. If there is a syntax error anywhere in the command the soft-key display will show "**ERR**". When a command is potentially finished, "-DONE-" will be shown in one of the soft-keys. Unless "-DONE-" appears, it is not possible to press return to execute the command.

Because of the command completion feature provided by the soft-key system, it has not been necessary to shorten or abbreviate commands to make them easier to type. In fact, commands are often longer than necessary to make them more descriptive. The soft-key labels are limited to 7 characters so occasionally a keyword will be truncated to fit on the display. To see the hidden part, press function key f10 (or Cntl-G) repeatedly and all over-length soft-keys will scroll to the left. On some terminals there may be too few function keys, or the function keys may not work. In this case a simple work-around is provided: press escape followed by a number from 0 to 9 to simulate pressing the corresponding function key (esc-0 is function key f10). If desired, almost all commands can be entered using function keys and numbers alone. When *expert* mode is off (`expert off`), non-essential and diagnostic commands not needed for regular S2 operation do not appear in the softkey display and cannot be entered. This is useful to reduce confusion when first learning about the S2 or to prevent inappropriate commands from being used during regular operation.

Key	Operation	termcap code
Backspace Cntl-H or Del	Delete character to the left of cursor	kb
f1 – f8	Enter softkey	k1 – k8
f9 or Cntl-X	Delete previous word	k9
f10 or Cntl-G	Scroll long softkeys to the left	k0
Esc 1–9, 0	Simulate pressing f1–f9, f10	
Tab or Space	Complete command	
Cntl-P	Cursor up (**P**revious history)	ku
Cntl-N	Cursor down (**N**ext history)	kd
Cntl-B	Cursor left (**B**ackward)	kl
Cntl-F	Cursor right (**F**orward)	kr
Delete or Cntl-D	Delete character under cursor	kD
Esc Esc	Clear entire line	
Cntl-A	Move to beginning of line	
Cntl-E	Move to end of line	
Cntl-W	Delete to beginning of line	
Cntl-T	Toggle insert mode	
Cntl-L	Re-draw entire screen	
Cntl-C	Interrupt current operation	
Cntl-_	Re-boot the system	

Table 4-1: Console Special Function Keys

4.2.3.2 Command Editing and History

The S2 command-line editing and history features are similar to the Unix **tcsh** and MSDOS **4DOS** programs. Commands can be edited as you type them, or you can recall any of the last 100 commands and edit or re-enter them. Use the left/right cursor keys to move around within a command, and the up/down cursor keys to move around within the command history. Characters typed are inserted at the current cursor position, and the backspace key removes the character to the left of the cursor. The f9 key (or Cntl-X) deletes the previous word, and pressing escape twice in a row clears the entire line. You can search for specific commands by typing the first few letters of the command and then moving up or down in the history. The Cntl-B/Cntl-F and Cntl-P/Cntl-N keys are alternate cursor controls for terminals without proper cursor keys. Refer to Table 4-1 for a complete list of special function keys.

Re-drawing the Screen

Sometimes the display may become garbled because of lost characters or other problems with the terminal. Since only screen areas that change are normally updated, this condition will not fix itself, so press Cntl-L to re-draw the entire screen.

4.2.3.3 Batch Files

Batch files are lists of console commands prepared in advance and then executed using the `batch` command. They can be created by capturing a sequence of console commands (`batch log`) or from scratch using a built-in full-screen editor similar to the Unix vi editor (`batch edit`). The S2 stores batch files in non-volatile SRAM memory, so they will not be lost when the power is turned off. Some console commands are meant exclusively for use within batch files, such as `sleep` which performs absolute or relative time delays and `echo` which writes messages to the screen or waits for the user to press return (`echo prompt`). Batch files can be chained (but not nested) simply by executing one from within another. A variation on this is to make a batch file loop continuously by calling itself. Batch files can be interrupted at any time by pressing Cntl-C once. For more information, see the description of the `batch` command on page 4-31.

```
 S2 Recorder |         User Parms (playback)      | rec24     1998   217-20:19:06
 \/ Operating|                                     |
    System   | Tape ID: CA-3-0028813              | Recorder mode : 32x4-2 (br)
-------------|                                     | Station delay : [no 1 Hz in]
 Transports  | Station    : USUDA64              | Diag dec 6 in : channel 6 play
0*03:06:41.00| Source     : 3C273               | Diag dec time : 211-03:06:41.00
1*03:06:41.00| Scan Start: 211-03:00:00          | Diag dec sync : 100% (track cal)
2*03:06:41.00|                                     |
3*03:06:41.00| User data valid: yes              | Est err rate : 4.9e-5  (8 ch)
4*03:06:41.00| Tape speed: slp                   | % data valid : 99.99%  (128 Mb/s)
5*03:06:41.00| Tape position: 2:40:24 +-02 s     | Decoder Xbar : 01234567
6*03:06:41.00|------------------------------------|-------------------------------
7*03:06:41.00|  <<   ▮ >   >>    []    ||    REC  |
-------------------------------ALIGNED------------|
                                                  |
> stop                                            |----------------------------------
> position 2:40:00
> play
> ▮

--------------------------------------------------------------------------------
[help    ] [time    ] [mode    ] [tapeid ] [user    ] [record ] [play    ] [--ETC--]
```

Figure 4-2: Sample ROS Screen, User (playback)

4.2.4   Basic Procedures

Although there are many possible commands, only a few of them are actually needed for everyday S2 operation. This section outlines the commands and procedures that will typically be used during regular operation of an S2 Playback Terminal to play back and/or record tapes. It is written from the point of view of the console, but can easily be applied to the RCL as well. When first learning to use the S2 we recommended that you turn off "expert" mode by typing expert off. This hides low-level diagnostic commands that may cause confusion. To see an actual demonstration of a playback procedure, run the S2-PT demo batch file as follows: First place 8 scratch tapes in the transports, then type batch demo.pt. The demo will not work on S2 systems with fewer than 8 transports.

4.2.4.1   Playing Back Tapes

Playing back previously recorded tapes involves the following basic steps:

1. Select a mode with the correct number of transports (mode command).

2. Select the desired group number (group command).

3. Position the tapes to the appropriate starting location using the position command.

4. Start playback by typing play.

The user will typically have available information logged at record time to help decide what mode, group, and position to use in steps 1 to 3. In addition, several S2 parameters such as the recorder mode and barrel-roll settings are automatically set to the right value once playback starts. Although the correct playback mode will be set automatically, step 1 above is still necessary because the required mode cannot be determined until the transports are put into motion, and for that the S2 must be told how many transports to use. Any mode can be chosen in step 1 as long as it has the same total data rate as the mode used for recording. The same group number as used at record time should be chosen.

Figure 4-2 shows what the console display might look like during playback. The tape time of each transport, if known, is shown in the Transport window on the left side of the screen. The Tape ID and User Info fields entered at record time are shown in the User Parms window, as well as the tape speed and overall position. Tape position

is expressed as the number of hours:minutes:seconds of recorded data since the beginning of tape, e.g.

```
4:58:57  +-05 s
```

means 4 hours and 58 minutes into the tape. The position may optionally be shown as an integer number of seconds by changing the **posdisformat** parameter in the defaults file, in which case the previous position would be displayed as

```
17937  +-5
```

The overall position display includes a linear variance value that indicates the maximum absolute deviation of any individual transport position from the value shown. Usually all tapes in a group will remain close enough together so that a single position value can be used, but if necessary it is possible to enter individual positions for each tape in the `position` command, and to display them with `transport display position`. The tape counters on the transport front-panel displays should not be used since they do not show the correct tape position (they have a 64/60 multiplicative error, as well as other uncertainties).

Tape Alignment

The S2 automatically aligns the tapes to each other shortly after playback starts. Playback data is not valid until tape alignment completes. The user can then use the `align` command to adjust the overall tape alignment and synchronize two or more playback terminals. For example,

```
align  -00:01:27.0893017
```

will move the tapes backward in time by 1 minute, 27 seconds, and 89.3017 milliseconds. Alignment times can also be given in absolute terms (see the `align` command on page 4-29).

Handling Bad or Lost Tapes tapes

In the event that one or more tapes in a set is lost or damaged, or was never recorded due to a failure at record time, users may still wish to play back the remaining good tapes. In this situation the following procedure should be used:

1. Insert the available tapes.

2. Set the correct mode & group as if the bad/missing tape(s) were present.

3. Type `transport autoselect-N` where N is the transport corresponding to the bad/missing tape. This de-selects that transport. If more than one tape is missing you can specify additional transport numbers with minus signs. Note that there is no RCL equivalent for this command but RCL users can still enter it using CONSOLECMD.

4. Perform positioning and playback as usual. You will get the following ''fatal'' status messages which may be ignored:
   STAT_BADTRANSEL: Bad transport selection for mode/group
   STAT_CHIDMISS: Tape with channel ID N is missing

5. When done, restore the normal transport selection by resetting the group number with the `group` command.

The effect of a missing tape depends on whether barrel roll was on or off at record time. If one of 8 tapes is missing and barrel roll was on, one out of every 8 bits in each user data channel will be invalid on playback. If barrel roll was off at record time, those user data channels which correspond to the missing tape(s) will contain invalid data while the remaining channels will be intact. To see exactly which user data channels on the C2 cable are still valid in this situation, switch to the user I/O screen by typing `userio`. In both cases (barrel roll on and off), the data obtained for the invalid bits is undefined.

When finished with playback, `stop`, `rewind`, and `eject` can be used to rewind and remove the tapes as usual.

```
 S2 Recorder |       User Parms (record)     | rec24     1998  211-03:07:08
 \/ Operating|                                |
    System   | Tape ID: CA-3-0028813          | Recorder mode : 32x4-2 (br)
------------|                                 | Station delay : 0 us
 Transports | Station   : USUDA64            | Diag dec 0 in : channel 0 rec
0*    > REC L| Source    : 3C273              | Diag dec time : 211-03:07:08.00
1*    > REC L| Scan Start: 211-03:00:00       | Diag dec sync : 100% (uncalib)
2*    > REC L|                                |
3*    > REC L| User data valid: yes           | Est err rate : <1.0e-6 (1 ch)
4*    > REC L| Tape speed: slp    Type: 3     | % data valid : 100.00% (16 Mb/s)
5*    > REC L| Tape position: 2:40:34 +-01 s  | Decoder Xbar : 01234567 (fthru)
6*    > REC L|-------------------------------|---------------------------------
7*    > REC L| <<   >   >>   []   ||   REC  |
------------|-------------------------------|
            |                                |
            |                                 ---------------------------------
> █         |
            |
            |
            |
            |
            |
-------------------------------------------------------------------------------
[help   ] [time   ] [mode   ] [tapeid ] [user   ] [record ] [play   ] [--ETC--]
```

Figure 4-3: Sample ROS Screen, User (record)

4.2.4.2   Recording Tapes

The record function is useful on Playback Terminals to (for example) make test tapes. If desired, S2-PTs can also be used like regular S2-RTs. However to use an S2-PT as a Record Terminal and record real user data you will probably need to change some of the defaults file parameters. At minimum the following defaults parameters should be as shown:

    scpllmode = refclk
    clkinsrc = c1
    clkinrate = [AS REQUIRED]
    stopsetup = bypass

Making a recording involves the following basic steps:

1. Set the time

2. Set the recorder mode and group

3. Enter the Tape ID and other user-defined information fields

4. Set the tape speed (if different from the default, AG2530 and earlier only)

5. Set the tape type (if different from the default)

6. Switch on the diagnostic test vector data (if making test tapes)

7. Start recording

Each of the above steps is described in detail below. Figure 4-3 shows what the console display might look like during record. For information on what kind of tape can be used in the S2 please refer to Chapter 5.

Time Setting

Before making a recording, the S2 system time must be set, typically to UTC time. Only the seconds and higher-order digits need to be set since the fractional seconds portion of the time is precisely determined by the S-1 Hz hardware sync signal (S1HZ_REC input on the C1 cable). At power-up, the initial system time is taken from a battery backed-up clock on the RCC, and will usually be correct to within a few seconds. However, the time is not considered "validated" until it has been set or approved by the user. As long as the time is not validated, two highlighted question marks appear beside the time on the console display and a warning is issued when the S2

is commanded to record. When setting the time the station delay measurement displayed in the upper right of the console screen should be 0, otherwise the time setting may be inaccurate. If some value other than 0 is shown, type `delay 0` to zero the station delay.

The console `time` command can be used to enter an absolute time as in the following example:

```
time  093-23:18:00
```

Here 093 is the day number (from 001 to 365, 366 for leap years) and the following fields are hours, minutes, and seconds. The day number can be omitted, in which case the current day is assumed. Absolute time commands such as this one should always be issued immediately *after* the 1 Hz tick they represent, i.e. above you would press return just after the reference clock's seconds digits flipped to 00. Setting the year requires a separate command as follows:

```
time  y 1996
```

This will almost never be necessary since the year should already be correct. It is possible to individually set any time field, not just the year, in both absolute and relative terms. For example, to increment the seconds by 2:

```
time  s+2
```

This form of time setting will probably be common since, as already mentioned, the default time at boot-up should be correct to within a few seconds. If the default time happens to be perfectly correct, or you are not particularly interested in setting the exact time, the time can be approved without changing it by typing `time validate`.

If the S2 system is connected to a network, the time can be obtained from one of the other hosts on the network using `time setremote`. This can also be set to occur automatically every time the system is turned on (see the description of the time command on Page 4-50 for more details).

The `time` command also has a useful facility for converting between day numbers as used by the S2 and the more usual month/day representation. Enter `time convert` to display the current time and date in month/day format, or specify a date in one format and it will be converted to the other format.

Mode and Group Setting

The S2 recorder mode is set using the `mode` command. The correct mode to use depends on the number of user data channels and data rate required for your particular application. The S2 is capable of individual channel data rates of 4, 8, 16, or 32 Mbits/s at 1 or 2 bit quantization, with either 1, 2, 4, 8, or 16 active data channels, subject to the restriction that the total data rate must be from 16 to 128 Mbits/s in multiples of 16. Under the S2 mode naming scheme, the most straightforward modes have names of the form **bxu-q** where **b** is the user channel bit-rate in Mbits/s and **u** is the number of user data channels. The **q** indicates either 1 or 2-bit quantization[1]. By convention two S2 user data channels are occupied by each 2-bit-quantized sampler channel, i.e. sign and magnitude are counted separately. For example, mode 16x8-1 is 8 channels of 16 Mbits/s data at one-bit quantization. Mode 4x16-2 is 16 channels of 4 Mbits/s data at two-bit quantization (and therefore represents 8 2-bit sampler channels). With regular S2 modes ('x' letter code), active user data channels are allocated sequentially starting at channel 0. Other modes exist in which the active channels follow different patterns — for example the 'i' and 'p' modes, initially added to support RadioAstron. To see exactly which user data channels are active, switch to the user I/O screen by typing `userio` (repeat the command to switch back). Table B-5 also shows the active user channels for each mode. Note that some modes have the same active user data channels and data rate as other modes but differ in the way channels are mapped internally (e.g. 16x8-2 and 16v8-2).

Setting the mode implicitly selects how many transports will be used for subsequent operations. Transports are normally selected in order starting at address 0, including as many as is necessary to accommodate the total data rate. For modes which do not require all available transports, it is further possible to specify which *group* of transports will be used. For example, in mode 16x2-1 (which requires 2 transports), transports 0–1 are "group 0",

---

[1] *The quantization setting currently only affects the data validity signals during playback on a PT, and has no effect in any other situation.*

Mode 16x8-1 (128 Mb/s)   Mode 8x8-2 (64 Mb/s)

Mode 16x2-1 (32 Mb/s)   Mode 4x4-2 (16 Mb/s)
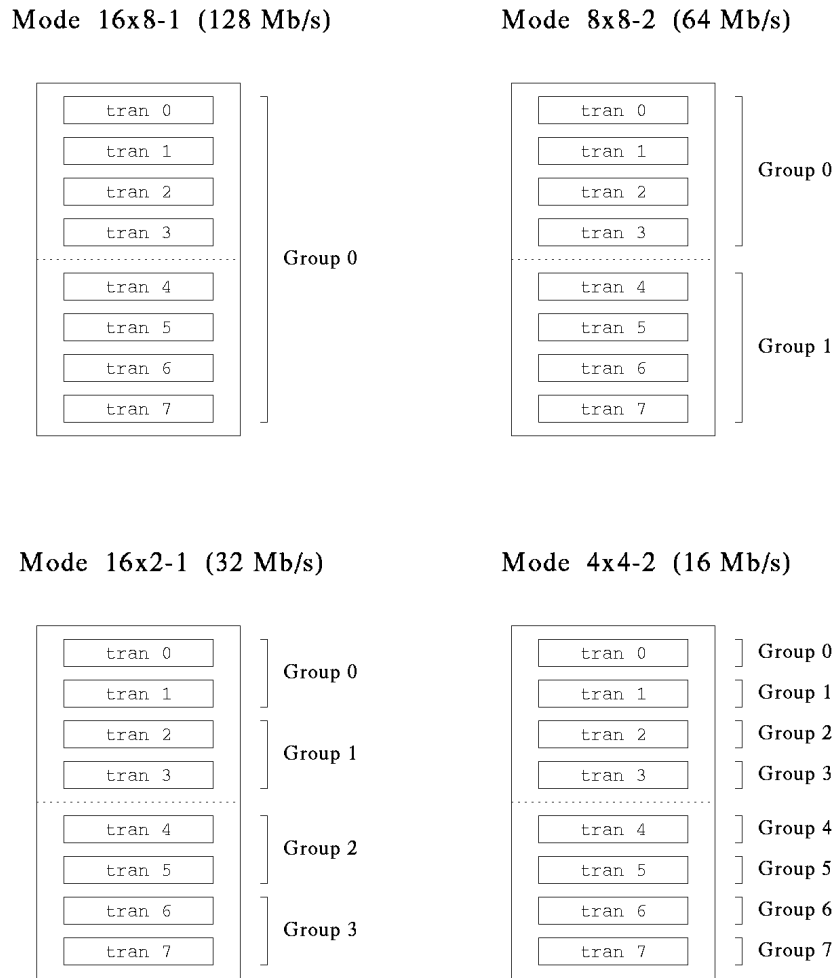
/home/s2/doc/userman/working/dwg/group.xpic

Figure 4-4: Transport Selection based on Mode & Group

transports 2–3 are "group 1", and so on (Figure 4-4). The console `group` command or RCL GROUP_SET command is used to set the group number. Together the mode and group settings determine exactly which transports will be used. Group switching can be used to extend unattended recording time in low-bandwidth modes, and will typically be performed using the RCL GROUP_SET command as part of an automated schedule running on an external control computer. Note that group switching is not allowed during regular (automatic) play or record — the S2 must be stopped and then restarted.

See the description of the `mode` command on page 4-40 for additional information, including a complete list of valid mode designators, and the `group` command on page 4-38.

Tape ID and User Info

The *tape ID* and *user info* are user-assignable text fields that get recorded on tape as part of the S2 auxiliary data. All 8 transports receive the same auxiliary data, so this information is common to all tapes in a set. The

tape ID is a text string up to 20 characters long assigned using the `tapeid` console command. It should be chosen so that it uniquely identifies the set of tapes being recorded, and should remain the same for the entire duration of the tapes. Information contained in the tape ID is entirely up to the user, however CRESTech and the Canadian correlator group have defined an S2 tape labeling convention with tape IDs of the form XX-T-YYYYYZZ, where XX is the country code, T is the tape type code, YYYYY is the tape set index number, and ZZ is a checksum. Example: CA-3-0012307

The physical tape label includes an extra suffix which indicates the tape number within the set from 0 to 7. More information on the Canadian tape labeling and user info conventions can be found in the document DRAO-RPCS3001, available at *http://www.drao.nrc.ca/web/vlbi/docs/doc3001.ps.gz* .

User info consists of 4 arbitrary text fields of varying lengths, each with a 16-character user-assignable label. The field label should describe the contents of the user info field. For example, the label might be "Source Name" and the contents of the field might be "3C273". The console commands to set info field 2 as just described would be as follows:

```
user info 2 label Source Name
user info 2 3C273
```

After setting the field label for the first time, it does not need to be set again for the rest of the session. User info can be changed at any time (but should not be changed more often than every few seconds). The Canadian convention is to put the name of the recording station in info field 1, and the source name in field 2 (fields 3 and 4 are not specified). The maximum lengths of the four fields are 16, 16, 32, and 48 characters respectively.

The tape ID and the first three user info fields are displayed in the User Parms window located in the upper middle of the console screen. To display the value of the fourth (and longest) user info field, type `user info 4`. On a PT the stop setup will usually be set to **play**, so the screen display will continue to show the (absent) playback parameters even when the S2 is stopped. To display the record parameters, change the stop setup to **bypass** by typing `setup bypass`.

Tape Speed

On S2 systems with AG2530 and earlier model transports, either of two available tape speeds may be selected using the console `speed` command. For LP (long play), enter `speed lp`, and for SLP (super long play) enter `speed slp`. LP should be used when optimum bit-error rate performance is desired. SLP gives 50% longer recording times at somewhat higher bit-error rates (for more information see Chapter 5). SLP is sometimes also called EP. With AG2550 and later model transports only the SLP speed is supported for record (tapes recorded in LP can still be played back). In many applications the tape speed will always be the same, so a default value can be set in the system defaults file using the **recordspeed** parameter. As shipped the default tape speed is SLP. The defaults file is described in Section 4.2.7.

Tape Type

Several different types of tapes are approved for use in the S2, and the S2 needs to be informed which type is being used due to differences in magnetic properties. Tape type codes are numbers from 1 to 9 or letters from A to Z. For a complete list see Chapter 5 or type `tapetype typecodes`. In many applications the tape type will always be the same due to an established tape pool, so a default value can be set in the system defaults file using the **tapetype** parameter.

Enabling Test Vector Data

(If you are not recording test tapes, skip ahead to "Starting the Recording".) Two types of pseudo-random test vectors can be produced by the S2 to record test tapes: the Formatter test vector and the UI test vector. Facilities for using the Formatter vector are slightly more versatile so it is usually preferred. The Formatter test vector is enabled by typing `form diag on` before making a recording. Then use `form berc` during playback to take bit-error rate measurements. For an average across all active playback channels use `form berc rotate`. See Page 4-37 for more information on these commands.

The UI test vector is different from the one produced by the Formatter. It is used for the C1 and C2 cable test modes, as well as the UI diag modes. To make a recording with the UI test vector attach an appropriate external test data generator to the C1 cable or switch to one of the UI diag modes (e.g. `mode diag8`). Then use `uic berc` during playback to take bit-error rate measurements. See Page 4-54 for more information.

Starting the Recording

Once all of the above parameters have been set, starting the recording is easy: just type `record`. During record, the S2 automatically scans the transports and monitors the data being recorded, including tests for proper signal level, recovery of sync, and correct extraction of auxiliary data. Although the transports are not capable of read-after-write, the loopback signals being monitored are tapped off as close to the record heads as possible. Any unusual conditions will be reported immediately in the console status window.

When the recording is complete, type `stop` to stop the tapes. If any of the tapes should reach the end (EOT), the S2 will stop *all tapes* automatically. Type `rewind` to wind the tapes to the beginning if desired, and then type `eject` to extract them.

"Eject-All" Feature

A special convenience feature allows all of the tapes to be manually ejected without using the console: press and hold any transport's eject button for two seconds (or more) and all the tapes will eject. This feature should preferably be used only when all transports are stopped. No other transport front-panel switches or buttons should be touched. The **ejectalldisable** defaults file option can be used to disable the eject-all feature.

4.2.5   System Bootup

The S2 system goes through a boot procedure approximately 90 seconds long every time the power is applied or the system is reset. The boot procedure loads and initializes the software, configures the hardware, and performs a number of integrity checks to ensure correct system functionality. This section may be a bit technical for some users, so you may wish to skip ahead to "ROS Bootup".

The first code to run is part of what's called the *boot monitor*, which resides in EPROMs located on the RCC (MVME133XT CPU board, VME slot 1). This performs phase 1 of the boot sequence. The boot monitor first displays the following message:

```
    Press space bar to enter boot monitor, 'n' for network boot...
```
It then enters a short (4-second) count-down sequence. Normally, the count-down is allowed to expire and the boot procedure continues to phase 2, VxWorks[2] initialization. If however you hit the space bar, the boot sequence is interrupted and the CPU enters the boot monitor, whose command prompt is "`[S2 Boot]:`". The monitor supports a small number of low-level commands used only for CPU/SRAM diagnostics and maintenance functions. The boot sequence will also drop into the boot monitor if it detects data corruption in certain critical portions of the SRAM ('VxWorks' and 'root' partitions). See Section 6.2 for information on how to re-load a corrupted SRAM. Type `go` to continue the boot sequence from inside the boot monitor. Back at the count-down sequence, you can type 'n' to cause a network boot. This is only for maintenance operations at CRESTech/SGL.

Phase 2 in a normal boot sequence is VxWorks system initialization and Static RAM checkout. The boot monitor first copies the VxWorks system code from SRAM into the DRAM on the CPU board and then begins executing it. A series of messages like the following accompanies the creation and checkout of SRAM partitions:

---

[2] *VxWorks*[TM] *software was developed by Wind River Systems, Inc.*
   Curses software was developed by the University of California, Berkeley.

```
    Creating SRAM disk /sram/bin/ at 0x4012c00, length 819200 bytes
```
Next, ROS and related software is loaded from SRAM into DRAM. If any problems are encountered, the console will beep and display an error message, and the CPU will drop into the VxWorks shell at the end of phase 2. The VxWorks shell (prompt "->") is used for software checkout and debugging by technical personnel. You can also force the shell to start up by pressing the space bar during the software load portion of the bootup, but this is not normally done.

If no problems are encountered in phase 2, the boot procedure continues on to phase 3, ROS bootup.

4.2.6   ROS Bootup

The Recorder Operating System (ROS) boot procedure comes after the VxWorks system portions of the bootup, and includes software initialization, hardware configuration and initialization, and the power-up self-test (known as *self-test 1*).

If this is the first time you are running this S2 system, ROS will first ask some installation questions, including the terminal type and baud rate you wish to use. The default terminal type is VT100 and the default baud rate is 9600, but these can be changed at this time. Enter your console terminal type and baud rate (max. 9600). If your terminal type is not one of those recognized by the S2, you will be given the opportunity to add a terminal definition to the terminal capability database, but to do this you must understand the Berkeley Unix termcap file format (see Appendix D). Table 4-2 shows some popular terminal types recognized by the S2 software as shipped. After you have answered the installation questions, they will not appear again. If you ever wish to re-run the installation procedure, press space during phase 2 of the boot sequence to enter the VxWorks shell, and then type `install` at the shell prompt ("->"). All installation options can also be set by editing the system defaults file, or by using the `default` command (see Section 4.2.7).

Most ROS hardware and software initialization happens before the main display screen appears. Each major step outputs a message to the console. These messages can usually be ignored, but are useful for debugging if problems arise. Any serious problems will result in status messages following the boot phase, so there is no particular need to pay attention to the bootup messages. As part of DRD board initialization, the OK LEDs are sequenced from left to right for visual verification that the address jumpers correctly match the address labels on the front of the boards (although any errors will also be caught by the self-test).

The ROS power-on self-test starts just after the main console display appears. The self-test requires all necessary transports and other system hardware modules to be present and connected. Normally all 8 transports and DRDs are tested, but the **diagnumtran** option in the system defaults file can be changed to reduce the number of transports/DRDs to test (1, 2, 4 or 8). This is useful for scaled-down S2-PTs with fewer than 8 transports/DRDs (the defaults file is described in Section 4.2.7). No console commands can be typed while the self-test executes (about 45 seconds), but you may choose to interrupt the test by typing Cntl-C once. Most RCL commands which affect tape motion or system switch settings are also not allowed during the self-test. Any faults detected are reported in the interaction window, as well as an overall diagnostic failure indication in the status window. For more information on the self-test see Section 4.6.

Code	Terminal Description	Code	Terminal Description
adm31a	LSI adm31a	**adm3**	LSI adm3
adm3a	LSI adm3a	**amiga**	Amiga ANSI
ampex210	Ampex a210	**ansi25**	IBM PC/NANSI.SYS
ansi	Ansi	**apollo**	Apollo
d800	Direct 800/A	**dialogue80**	Ampex dialogue 80
dm1520	Datamedia 1520	**dm2500**	Datamedia 2500
dm3025	Datamedia 3025a	**dm3045**	Datamedia 3045a
go140	Graphon go-140	**h19a**	Heathkit h19 ansi mode
hp150a	HP 150a	**hp2621a**	HP 2621a
hp2640a	HP 2640a	**hp2645**	HP 2645
hp2648a	HP 2648a graphics terminal	**ibm3101**	IBM 3101-10
ibmpc	IBM PC PC/IX	**ibmpcx**	IBM PC xenix console display
ISC8001	ISC 8001	**kaypro2**	Kaypro II
macintosh	Macintosh with MacTerminal	**masscomp**	Masscomp
regent100	Adds Regent 100	**regent20**	Adds Regent 20
regent25	Adds Regent 25	**regent40+**	Adds Regent 40+
regent40	Adds Regent 40	**regent60**	Adds Regent 60
st	Atari ST	**sun-cmd**	Sun Workstation cmdtool
sun1	Sun1 (old Sun Workstation console)	**sun**	Sun Workstation
tek4105	Tektronix 4105	**tek4115**	Tektronix 4115
tek4404	Tektronix 4404	**tvi924**	Televideo Model 924
tvi925	TeleVideo Model 925	**v3220**	LANPAR Vision II model 3220/3221/3222
vc404	Volker-Craig 404	**vc6220**	Volker-Craig 6220,
vis200	Visual 200	**vision1000**	Vision 1000 (dim reg standout)
vt100	DEC vt100 (or ROSCO)	**vt102**	DEC vt102
vt125	DEC vt125	**vt132**	DEC vt132
vt200	DEC vt200	**vt220**	DEC vt220
vt52	DEC vt52	**wyse50**	Wyse-50 in Wyse mode
wyse75	Wyse-75	**wyse85**	Wyse-85
wyse925	Wyse-50 emulating tvi925	**xterm**	X windows terminal emulator
yterm10	yterm 1.0 UCB ascii.kbd	**yterm11**	yterm 1.1 UCB ascii.kbd
z50	Zephyr 50		

Table 4-2: Terminal Types Supported by ROS

After the self-test completes (successfully or not), the system automatically begins executing the batch file called 'autoexec', if such a file exists. This can be used to perform any actions that are required every time the S2 is turned on. For example, you might put a command in the autoexec file to change to the User I/O screen automatically at bootup, or to set other conditions or switches as desired. However, please don't set switches in the autoexec file for which there is a provision in the system defaults file, otherwise the ROS boot procedure may not work properly.

When the autoexec batch file is finished, the ROS console prompt appears (">") and the system is ready for you to start typing commands or to start issuing control requests over the RCL. If the system is functioning properly and no serious problems were discovered by the self-test or online status, the green "OK" LED on the front of the VME card cage will be lit (**ledmode** defaults-file parameter must be set to 'ok'). The LED on the Formatter board has the same function. If there are serious problems with any of the DRD boards the "OK" LED on the

affected boards will fail to light, making it easy to tell which ones to swap. Any failure conditions indicated by LEDs will also be shown in the console Status window.

4.2.7   ROS Configuration

Two configuration files used by the Recorder Operating System can be edited by the user: the **defaults** file and the **termcap** file.

The defaults file contains default boot-up parameters such as the console terminal type, tape type, serial port baud rates, and input/output clock rates and modes. It also contains the S2 station number and "nickname", and many other operational parameters you may wish to set. Appendix C contains an example defaults file. Each default parameter is documented right in the file and should be self-explanatory. The defaults file can be edited using the `edit` command (see below), or individual parameters may be displayed and updated directly using the `default` command (described on Page 4-32). Note that changes to the defaults file take effect only after the next system boot. A few parameters such as the terminal type, station number and nickname are changed as side-effects of other console commands, and will automatically be updated in the file.

The termcap file is the terminal capabilities database in standard Berkeley Unix termcap format. It may be necessary to edit this file to add new terminal types or fix the behaviour of existing terminals, particularly with respect to function keys. This requires knowledge of the Berkeley termcap format, so an extract of the Unix reference manual on the subject of termcaps has been included in Appendix D. To help define termcap entries for special keys such as function and cursor keys, a useful utility is provided to decode ASCII character sequences. Type `shell` to enter the VxWorks shell and then type `showkey`. All subsequent characters typed will be displayed in hex format. Hit return twice to get back to the normal shell prompt.

To modify either of the above files, use the console `edit` command and give the file's name. This starts a full-screen editor which is very similar to the Unix vi editor. For those unfamiliar with vi, online help can be obtained from within the editor by typing ":help" and pressing return. Briefly, vi has two modes: a command mode where letter and number keys have special functions, and an insert mode where text is inserted into the file. You are normally in command mode, and can use the cursor keys to position the cursor to the point where you would like to make a change (the 'h','j','k', and 'l' keys also function as cursor keys). To start inserting text, press 'i'. Then press the escape key when you are done to return to command mode. Use 'x' to delete characters while in command mode. When you are finished editing, hit the upper-case 'Z' twice to save the file and exit the editor. Type ":q!" and press return to quit without making any changes.

If the terminal type is not set correctly, it may be difficult or impossible to edit. In this case use the console `term` command to set the correct terminal type. If the terminal you are using is not supported by the S2 you will need to add a definition for it to the termcap file *without* using the editor. This can be done via the S2 install procedure, which is invoked from the VxWorks shell by typing `install`. You can use the console `shell` command or press the space bar at the appropriate part of the bootup to enter the shell.

4.2.8   Console Screen Descriptions

This section describes each console display parameter, organized by window. Some low-level and diagnostic display parameters are not fully documented. More important display parameters are marked with "*". Those that are important only for record applications are marked "*R", and for playback applications "*P".

4.2.8.1   Crucial Parms Window (upper right, Figure 4-1)

(Top line, left):   The system nickname. This is the conventional name assigned to this S2 system, up to 8 characters long. The nickname is stored in the system defaults file, and can be set using the `station` command, or by editing the defaults file (`edit defaults`). This space will be blank if no nickname is defined.

(Top line, right)*:   The system time. This is the current reference time, written on tape during record. The time is synchronized to the S-1/C-1 Hz sync input (S1HZ_REC on C1 cable or C1HZ on C2a cable), possibly with a delay offset (see "Station delay" below). Time is usually set to UTC using the `time` command. The display format is "YYYY DDD-HH:MM:SS" where YYYY is the year, DDD is the day number (1–365, 366 for leap years), and HH:MM:SS is the 24-hour time. The display will be highlighted in the event of a severe failure of S2 internal system timing (and the clock will stop). There is a very short delay between the actual 1 Hz tick and the screen time display update, but it should be short enough so as not to cause problems. Most other display parameters have a somewhat longer delay ($\frac{1}{4}$ of a second or less), since it's not possible to update everything on the screen at the same time.

Recorder mode:*   This shows the current S2 channelization mode, also called recorder mode or observing mode. The mode determines the user data rate and the number of active user data channels, and is set using the `mode` command. If the barrel roller is on, "(br)" is shown beside the mode.

Station delay:*   Shows the amount of clock offset between the externally supplied 1 Hz reference (S1HZ_REC on C1 cable or C1HZ on C2 cable) and the S2's internal 1 Hz tick. This is a true measurement of the difference, with positive numbers indicating that the internal 1 Hz tick is later than the external reference, and negative numbers indicating that the internal tick is earlier. During record, the station delay determines the offset between the user's 1 Hz reference and the time used to label the bits on tape. During playback, the station delay determines the offset between the user's 1 Hz timing reference and when the corresponding data bit is output on the C2 cable. The displayed station delay can range from -0.5 seconds to +0.5 seconds less 31.25 nanoseconds. Display may be in units of seconds (s), milliseconds (ms), microseconds (us), nanoseconds (ns), or user data bits (bits), depending on the units used when the delay was set (the display units can also be changed without changing the delay).

    When the System Clock PLL is in **refclk** mode (used for recording or fixed-rate playback), the delay measurement should exactly match the delay setting from the last `delay` command. Glitches on the high-rate reference clock, or a shift in the external or internal 1 Hz timing can cause a mismatch, in which case the delay measurement display will be highlighted and the status message "Stn delay measurement != setting" (STAT_BADSTNDELAY) will appear. Corrections should be made manually by resetting the appropriate delay value. The S2 can also be instructed to automatically "fix" station delay mismatches by setting the defaults-file parameter **scpllrefclkfix** to 1. In this case any station delay mismatch that arises while in SC PLL **refclk** mode and remains stable for 2 seconds will be auto-

DSCM   TRANSPORTS

/home/s2/doc/userman/working/dwg/diag.xpic
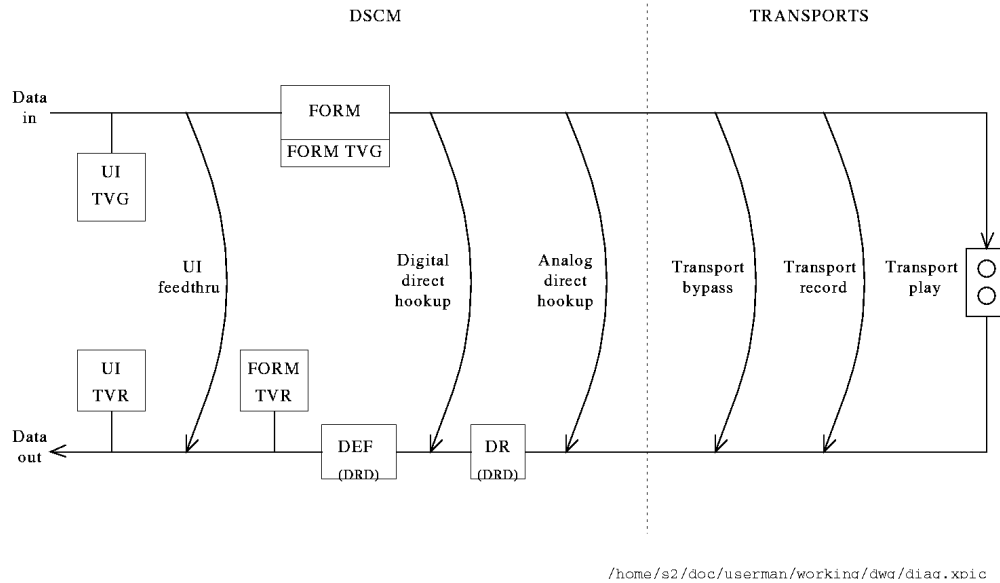
Figure 4-5: S2 Diagnostic Loopback Paths

matically corrected by re-asserting the last entered station delay setting. The informational status message STAT_DELAYFIX will appear, giving a count of the total number of such station delay auto resets since the last `status reset`. Beware that this momentarily disturbs the entire system timing, and may cause a significant glitch during both recording and playback, resulting in loss of several seconds of data. Users should decide whether station delay mismatches should be corrected at record time, with the possibility of glitching, or simply logged and then corrected at correlation (playback) time. To make it easier to establish later what the actual station delay was at record time, the delay measurement is continuously recorded in the tape auxiliary data channel and is one of the quantities available at playback time via the `tapeinfo` command.

In System Clock PLL **1hz** mode (used for playback), the delay measurement will only be approximately the same as the delay setting (within a few hundred bits). The display will be highlighted whenever the delay error is greater than about 60 microseconds (1000 bits at a data rate of 16 Mbits/s) and status message STAT_SCPLLUNLOCK will appear, indicating that the System Clock Phase-Lock Loop is not locked. If no user 1 Hz reference tick is present, "[no 1 Hz in]" will be displayed.

In System Clock PLL **errmes** mode (used for playback), the user 1 Hz reference is not used so there will normally be no delay measurement. However should a 1 Hz tick be supplied the delay measurement will still appear even though it is used for nothing except display.

Diag dec N in: Identifies which decoder (DRD) is currently acting as the *diagnostic* decoder, and the source of the data it is decoding. There are 5 possible input sources corresponding to 5 of the 6 diagnostic data paths (Figure 4-5): **play** (transport playback), **rec** (transport record), **byp** (transport bypass), **ahook** (analog direct-hookup), and **dhook** (digital direct-hookup). For each path it is possible to select a channel num-

ber (0 to 7) and a transport number (0 to 7). The channel number identifies which S2 internal data channel is being decoded and sent on to the UI for output. The channel number is normally the same as the transport number except when a transport other than the one corresponding directly to that channel is being used, e.g. during playback when the tapes are inserted in the transports out of sequence, or any time a group number higher than 0 is selected. The transport number is always the same as the diagnostic decoder number because each transport is directly connected to its corresponding decoder. Note that the transport number is only shown when it is different from the channel number.

The decoder input source is synonymous with the current recorder setup, as controlled by the console `setup` command (see page 4-45). Most of the time the setup is automatically changed as required, e.g. `record` selects the **record** path and `play` selects the **play** path. The `stop`, `rewind`, and `ff` commands select the *stop setup*, which is normally **bypass** on an RT and **play** on a PT. The stop setup can be changed by using the `setup` command while the S2 is stopped. The **stopsetup** defaults-file parameter determines the default stop setup to use at bootup.

Diag dec time: This shows the time extracted from the auxiliary data being decoded by the current diagnostic decoder. During playback, this will be the tape time from the transport feeding the diagnostic decoder (i.e. the one with the same address). During record the decoders are fed with diagnostic loopback data, so the time shown should be the same as the current system time. If time could not be decoded for the particular frame at the moment of display, "XXX-XX:XX:XX.XX" is shown. The time display update is synchronized to the system internal 1 Hz tick, with a very short delay between the actual 1 Hz tick and the display update. Most other display parameters, in particular the playback tape times in the transport window, have a somewhat longer delay (about $\frac{1}{4}$ of a second or less).

Diag dec sync: This shows the percentage of L2 sync achieved by the diagnostic decoder over the previous one-second period. It is computed by counting the number of data blocks with valid L2 sync (there are 250 data blocks per frame). L2 (level 2) sync is acquired at the beginning of each frame upon the detection of two consecutive sync words followed by 3 consecutive block IDs. The amount of sync is displayed as an integer percentage with fractions rounded down, so that only perfect sync over a 1 second period will show up as 100%. Remember that this value comes from a single decoder, so it represents only one internal data channel of 16 Mbits/s. The word "(uncalib)" appears beside the percentage if data-recovery calibration for the diagnostic decoder has not yet reached a sufficiently stable state where it is considered calibrated. "(track cal)" during playback indicates that phase-2 tracking calibration is in progress[3]. In some cases when the system is having difficulty playing back a particular tape, a special phase-1b algorithm may be triggered by ROS that performs large tracking jumps of 20 units or more.

Est error rate:[*] This gives an estimate of the raw bit-error rate (BER) of the data being decoded, averaged over the last 6 seconds. During playback, the estimate includes all decoded data, i.e. up to 8 internal data channels, 128 Mbits/s total. The exact number of channels included is shown in brackets beside the error rate, and may vary depending on the current mode or other factors. The error rate estimate is computed not only during playback but also during record, where the decoders are fed with diagnostic loopback data and thus the error rate will typically be negligible. When

---

[3] *There are two tracking algorithms: phase-1 is performed by the transport (green DIGITAL TRACKING light flashes, if present), and phase-2 is performed by ROS (track cal message appears)*

recording there is only one active diagnostic signal path so the estimated error rate represents only one internal data channel of 16 Mbits/s, however the automatic record monitor continuously scans active transports for complete diagnostic coverage. To see individual channel error estimates type `transport display esterr`.

The estimated error rate is obtained by counting the number of incorrect 16-bit sync words found when decoding the data, and extrapolating to encompass all data bits using an empirically derived formula. We assume that the error rate is low, so if a sync word is incorrect it is likely that in fact only a single bit is in error. We also assume a uniform distribution of random errors. These assumptions break down for very low and very high error rates, and for extended burst errors. In practice, however, the estimate has been found to be quite close to the true bit-error rate in most cases.

Because sync words represent only a small proportion of the data, a relatively long integration period is needed to obtain a useful estimate. The integration period used (6 seconds) gives a minimum measurement resolution of around 1.0×10^{-6}. There is also an artificial upper limit of 1.0×10^{-2}, imposed to avoid the inaccurate upper estimation range. System performance may be acceptable even with relatively high error rates (e.g. up to 3.0×10^{-3}) since much of the bad data would likely be flagged as invalid. This could, for example, be caused by a longitudinal tape scratch that destroys a particular sync word in each frame. Nonetheless, investigative action should probably be taken — see Section 6.4, "TAM Transport Replacement and Servicing".

% data valid:[*]   This shows the percentage of valid decoded data over the previous one-second period, obtained by integrating the data validity (DV) signal of the currently active internal data channels. During playback, the measurement includes all decoded data, i.e. up to 8 internal data channels, 128 Mbits/s total. The exact amount of data included is the same as for estimated error rate (above) and is shown in brackets beside the measurement. % DV is computed not only during playback but also during record, where the decoders are fed with diagnostic loopback data and thus the percentage of valid data will usually be 100%. When recording there is only one active diagnostic signal path so the % DV represents only one internal data channel of 16 Mbits/s, however the automatic record monitor continuously scans active transports for complete diagnostic coverage. To see individual channel % DV measurements type `transport display dv`. The % DV measurement normally recognizes the user DV flag, so it will drop to zero whenever user DV is set to false (`user dv no`). Type `user dv playback disable` to ignore the user DV flag.

Decoder Xbar: This shows the current mapping of decoder outputs onto S2 internal data channels. In general, there are up to 8 decoders in an S2 system. Each decoder input is physically connected to a particular transport. The crossbar switch at the decoder outputs allows tapes to be inserted in any order and allows switching between different groups of transports during playback. Setting of the crossbar is normally totally automatic, so the display is for reference only. The crossbar setting is represented by an 8-digit string where each digit position represents an internal data channel from 0 to 7. The value of the digit in a particular position indicates which decoder/transport is outputting on that channel ('x' indicates no output). For example, in the crossbar string "51234760" decoder/transport 5 is outputting data on internal channel 0, and decoder/transport 7 is outputting on channel 5. The string "01234567" represents the identity mapping, which means the tapes are in the right order. The crossbar can be set manually using the `def xbar` command, but this should not normally be necessary. The display shows "(fthru)" beside the crossbar setting if UI feed-through mode is on to emphasize the fact that the outputs of the decoder(s) are not being used. If the Mark III formatter is enabled, "(Mk3)" is shown to indicated that decoded output is being reformatted. If both feed-through and the Mark III formatter are on, "(Mk3fthru)" is shown.

```
S2 Recorder  |        User Parms (playback)      | rec24      1998  217-20:51:49
 \/ Operating|                                   |
    System   | Tape ID:                          | Recorder mode : 4x16-2 (br)
-------------|                                   | Station delay : [no 1 Hz in]
 Transports  |                                   | Diag dec 4 in : chan 0/tr 4 play
0    []    L|                                    | Diag dec time : XXX-XX:XX:XX.XX
1    []    L|                                    | Diag dec sync : 0% (uncalib)
2    []    █|
3    []    L| User data valid:                   | Est err rate : >1.0e-2 (4 ch)
4*   []    L| Tape speed:                        | % data valid : 0.00%   (64 Mb/s)
5*   []    L| Tape position: 0:10:02 +-01 s | Decoder Xbar : 45670123
6*   []    L|-------------------------------|--------------------------------
7*   []    L|  <<   >   >>   [█]   ||   REC |Tran 2 failure (reason follows)
-------------------------------------------|Tran 2 +5V/-5V supply bad (C4)
                                           |Tran 2 32 Hz sync errs (155)
                                           ---page-1/2-----FAIL----------MORE-
> position 0:10:00
> mode 4x16-2
Recorder mode set to: 4x16-2
> group 1
Transport group 1 is selected
> █
-----------------------------------------------------------------------------
[help    ] [time    ] [mode    ] [tapeid ] [user    ] [record ] [play    ] [--ETC--]
```

Figure 4-6: Sample ROS Screen, Bad Status

4.2.8.2   Status Window (middle right, Figure 4-6)

(Body of window)[*]:   Contains one-line status messages summarizing the health of the system. "Error" and "fatal error" messages are highlighted, but "informational" messages are not. Transient status entries (known as *clear-on-read* conditions) stay on the display until you type `status reset`. Refer to the list of status codes in Section 4.5 for the meaning and possible causes of each status message. If there are more than 3 status conditions, the status window can be paged down and back up using `status pagedn` and `status pageup`. To show all current status conditions at once with slightly more verbose descriptions use `status display fullscreen`. This in particular includes the status code numbers, not shown in the status window, which are useful to quickly find the right entry in the list of status codes in Section 4.5. You can also dump the current status information to the S2 internal log file by typing `status display tolog`. This provides an easy way to save a status snapshot for later examination.

(Bottom boundary line)[*]:   The current status page number is shown on the left side of the bottom window boundary, and the word "MORE" is displayed on the right side if there is another page below this one. If any fatal status errors exist, the word "FAIL" appears highlighted in the middle. Fatal errors indicate that the current operation is probably not proceeding successfully, and corrective action should be taken. The green "OK" LED on the front of the VME card cage will go out when FAIL appears in the status window, provided the **ledmode** defaults-file parameter is set to 'ok'. Note that fatal clear-on-read status errors cause the FAIL condition to persist until status is manually reset using `status reset`, even if the fatal condition is no longer true. The "OK" LED reacts to clear-on-read errors slightly differently. It will go back on either when `status reset` is typed or when the RCL status is read (implicitly clearing clear-on-read conditions), while the console FAIL indicator is unaffected by the RCL.

4.2.8.3   User Parms Window (upper middle, Figure 4-1)

(Top line):   Gives the name of the window, since this area of the display can also show other windows (type `user` to switch to the User Parms window if it is not currently displayed). If the word "record" appears on the top line in brackets, the rest of the window shows information that is being (or would be) recorded on tape. If the word "playback" appears in brackets, the rest of the window shows information being extracted from tape, specifically the tape in the transport feeding the current diagnostic decoder. When the S2 is stopped (or rewinding/fast-forward winding/positioning) the *stop setup* determines whether record or playback parameters are shown. If the stop setup is **play** playback parameters are shown, otherwise record parameters are shown. The stop setup is normally **bypass** on an RT and **play** on a PT, but can be changed in the defaults file or by using the `setup` command while the S2 is stopped.

Tape ID:[*]   Shows the tape identifier string, up to 20 characters long, as set by the `tapeid` command.

(User info 1)[*]:   Shows user info field 1. The label (text to the left of the colon) can be changed by the user. Depending on the length of the label, the entire 16-character info field may not fit on the display. If so, it will be truncated on the right. User info is set with the `user info` command.

(User info 2)[*]:   Shows user info field 2, similar to user info 1.

(User info 3)[*]:   Shows user info field 3, similar to user info 1 except that the maximum length is 32 characters so this field is more likely to be truncated. User info field 4 is not shown. Type `user info N` to display any user info field in its entirety, where N is from 1 to 4.

User data valid:[*]   Shows the setting of the user data valid flag. The user DV flag is recorded on tape as part of the auxiliary data and recovered at playback time, where it is used to invalidate data (if false) by lowering the data validity signals on the C2b/c cable. This facility provides a way to indicate at record time that invalid data is being recorded, e.g. the telescope is slewing or off-source (see the `user dv` command on page 4-55). During playback, the display shows what the user DV flag was set to during record. The user DV flag can be ignored at playback by entering `user dv playback disable`, indicated by the word "(ignored)" on the display. Then data marked invalid by the user at record time will be treated as valid.

Tape speed:[*]   Indicates the record or playback tape speed. Legal speeds for recording are **lp** and **slp** on AG2530 or earlier model transports, and **slp** only on AG2550 and later transports[4]. All transport types can play back at either speed, but playing **slp** tapes on AG2510 transports is not recommended due to poor performance. During playback, the unused speed **sp** may sometimes appear for blank spots on the tape — this is normal. All active transports must report the same speed during playback or this display will be blank.

Tape position:[*]   Indicates the current overall tape position. Position is measured as the number of seconds of recorded data since the beginning of tape, and is normally displayed as hours:minutes:seconds, e.g. `3:18:57`. This format is similar to a tape time except that the hours portion is shown with only a single digit. Alternately, the

---

[4] *The SLP speed is sometimes also called EP*

position may be shown as an integer number of seconds by changing the **posdisformat** defaults file parameter. Although there may be up to 8 tapes and therefore 8 different positions, it is usually sufficient to deal with a single position value since the group of currently active transports will typically be in almost the same position. The overall position is defined as the mid-point of the individual transport positions, and includes a linear variance value that indicates the maximum absolute deviation of any individual position from the mid-point, e.g. `+-12 s` means that the latest transport is 12 seconds ahead and the earliest transport is 12 seconds behind the overall position. If the variance exceeds 60 seconds it is shown in hours and minutes, e.g. `+-1h29` means one hour and twenty-nine minutes. To show each transport's individual position use `transport display position`. Positions are normally positive but can take on small negative values due to anomalies around BOT. If any individual position is unknown, the overall position is also considered unknown. See the `position` command on Page 4-42 for more information.

4.2.8.4 State Window (middle, Figure 4-1)

(Body of window)[*]: Contains highlighted symbols to represent the overall S2 tape motion state based on the states of the individual transports. Only the states of currently selected transports (those marked with '*') are used in determining the overall state. The symbol meanings are as follows:

      `<<` — rewind
      `>` — play
      `>>` — fast-forward
      `[]` — stop
      `||` — pause
      `REC` — record

Sometimes the symbols will light in combination to indicate states such as cue and review. An 'M' after the record or play symbol indicates manual record/play. In some unusual situations no consistent overall tape motion state can be derived, e.g. after certain transport errors or low-level console commands. In this case no state is indicated.

(Bottom boundary line)[*]: The word "POSITIONING" indicates that tape positioning is in progress. The word "ALIGNED" indicates that all tapes are aligned during playback. The words "NO TAPE" indicate that one or more required tapes have not yet been inserted.

4.2.8.5 Transport Window (middle left, Figure 4-1)

(Top line): Indicates what transport-related parameter is being displayed. Normally, the window shows the transport state and switches to tape time during playback if time can be extracted. In this case the window is simply labeled "Transports". For other display parameters, more specific labels are used.

(Body of window:) For each transport (0–7) shows one of several parameters as selected with the `transport display` command. Transports which do not respond to communication attempts are shown as "`--`". Currently selected transports are marked '*', and transports for which system status contains failure conditions are marked with a highlighted 'F'. A list of possible display parameters is given below. The word in square brackets is the transport display command code that selects the

parameter. Note that when playback tape times are shown the displayed values are typically delayed by about $\frac{1}{4}$ of a second due to display update latencies (use the "Diag dec time" display for a more instantaneous tape time value).

State/Tape time: Shows the transport state, or playback tape time if available (hours/mins/secs/frames). Includes servo lock indicator ('L'). This is the parameter normally displayed. [time]

State/Tape day: Like the first display above, but the tape time includes day#/hours/mins. [day]

State/Tape year: Like the first display above, but the tape time includes year/day#. [year]

Tape position: Gives the individual transport tape positions in hours:minutes:seconds of recorded data since BOT, or as an integer number of seconds depending on the **posdisformat defaults file parameter. [position]**

Est error rate: Shows the estimated bit-error rate for each transport/DRD (includes all transports in auto record/play only). [esterr]

% data valid: Shows the % data valid for each transport/DRD (includes all transports in auto record/play only). [dv]

AGC level: Indicates the playback/record/bypass signal strength. [agclevel]

Tape channel ID: Shows the channel ID extracted from tape during playback. This corresponds to the address of the transport used to record the tape. [chanid]

Transport temp: Shows the temperature inside the transports in degrees Celsius. [temp]

In-service time: Shows the total number of hours the transport has spent in play, record, pause, cue, or review states since the last service event (as recorded using transport N service lastserv). Also shows the total in-service time since manufacture. [servtime]

Head-use time: Shows the total number of hours the transport has spent in play, record, pause, cue, or review states since the last head replacement (as recorded using transport N service lasthead). Also shows the total head-use time since manufacture. [headtime]

On time: Shows the total amount of time the transport has had power applied since manufacture in hours and minutes. [ontime]

Transport tracking: Displays the current transport tracking control value, relative to the internal (phase-1) optimized position. Includes manual and pending flags. [tracking]

TCP serial #: Shows the transport (TCP) serial number. [serial]

Playback TCP serial #: Shows the transport serial number extracted from tape. This corresponds to the serial number of the transport (TCP) used to record the tape. [serialpb]

Transport state: Shows the transport state, but does not switch to tape time. [state]

4.2.8.6   User I/O Window (upper middle, Figure 4-7)

This window summarizes the current input/output configuration. It replaces the User Parms window when you type userio. Most of the parameters shown in the User I/O window are controlled using the userio and scpll console commands.

(Top line):   Gives the name of this window, "User I/O".

Data rate: Shows the user channel data rate in Mbits/s, and the number of active user channels. These are determined by the current recorder mode setting (see the mode command). The product gives the total system data rate.

```
 S2 Recorder |         User I/O          | rec24    1998  217-20:58:44
 /\ Operating|                           |
    System   | Data rate  : 32 Mb/s × 4  | Recorder mode : 32x4-2 (br)
-------------| C1 data in : 0-3          | Station delay : [no 1 Hz in]
 Transports  | C2 data out: 0-3          | Diag dec 0 in : channel 0 play
0*    []     | C2 DV out  : 0,2          | Diag dec time : XXX-XX:XX:XX.XX
1*    []     |                           | Diag dec sync : 0% (uncalib)
2*    []     | SC PLL lock mode: errmes  |
3*    []     | Ref clk/1 Hz in : (unused)| Est err rate : >1.0e-2 (8 ch)
4*    []     | Ref clk in rate : (unused)| % data valid : 0.00%   (128 Mb/s)
5*    []     | Ref clk out rate: 32 MHz  | Decoder Xbar : 01234567
6*    []     |---------------------------|-----------------------------------
7*    []     |  <<   >   >>   [] ||  REC |
-------------|---------------------------|
             |                           |
 > userio    |                           |
 > ■         |                           -----------------------------------
             |
             |
-------------------------------------------------------------------------
[help   ] [time   ] [mode   ] [tapeid ] [user   ] [record ] [play  ] [--ETC--]
```

Figure 4-7: Sample ROS Screen, User I/O

C1 data in:[*R]   Indicates which of the 16 user data input channels on the C1 cable are active in the current mode. User data channels are numbered from 0 to 15, corresponding to the input signal names IN 0 to IN 15 listed in Table B-1. Certain sequences of channels are abbreviated to fit on the screen, e.g. 0,2...14 means 0,2,4,6,8,10,12,14. Refer to Table B-5 for a list of which user channels are active in each recorder mode. If the S2 is in C1-cable test mode, "(C1test)" is shown. A switch can be set to disable the C1 data inputs for diagnostics, in which case the display shows "(disabled)".

C2 data out:[*P]   Indicates which of the 16 user data output channels on the C2a cable are active in the current mode and setup. The numbers shown correspond to the output signal names OUT 0 to OUT 15 listed in Table B-2. Unless the S2-PT is in play (or set up to play), only a subset of the potential C2 outputs are active since there is only one diagnostic signal path. When the UI is set to feed-through mode all potential C2 data outputs are active and contain feed-through data from the C1 input cable. When the Mark III formatter is enabled (mk3 form enable), the C2a cable switches to Mark III-compatible mode and "(Mk3)" is displayed. The cable pin-outs are as shown in Table B-4, but in this case the channel numbers shown in the display do not correspond directly to the TRACK numbers shown in the table but rather should be multiplied by 2 and then incremented by either 1 or 2 depending on whether the odd or even Mark III track number convention is being used. When the S2 is in a C2-cable test mode "(C2test)" is shown (or "(C2tMk3)" if the Mark III formatter is also on). A switch can be set to disable the C2 data and validity outputs for diagnostic purposes, in which case the display shows "(disabled)". See the userio command on Page 4-55 for more information.

C2 DV out:[*P]   Indicates which of the 16 user data validity outputs on the C2b and C2c cables are active in the current mode and setup. The numbers shown correspond to the output signal names DV OUT 0 to DV OUT 15 listed in Table B-3. Note that even channel numbers (0, 2, 4 etc.) appear on C2b and odd channel numbers (1, 3, 5 etc.) appear on C2c. The C2c cable is needed only for 1-bit quantized modes with names of the form **bxu**-1, since all other modes use only even validity channels. To avoid the need for the C2c cable substitute equivalent "interleaved" 1-bit modes for regular S2 modes, e.g. use 16i8-1 instead of 16x8-1. Unless the S2-PT is in play (or set up to play), only a subset of the potential C2 DV outputs is

active since there is only one diagnostic signal path. When the UI is set to feed-through mode all C2 DV outputs are forced high.

SC PLL lock mode:[*] This shows the current System Clock PLL lock mode, one of **refclk, 1hz, errmes, xtal,** or **manual** indicating the SC PLL lock reference source. This is normally **refclk** for a Record Terminal and **1hz** or **errmes** for a Playback Terminal. The **scpllmode** defaults-file parameter determines the default SC PLL mode to use at bootup. See the description of the `scpll mode` command on Page 4-44 for further details.

Ref clk/1 Hz in: Shows the source of the high-rate and 1 Hz input reference clocks, which can come either from the C1 or C2 cable. The C1 cable is normally selected on a Record Terminal, and the C2 cable on a Playback Terminal. In System Clock PLL modes other than **refclk** and **1hz** the clock inputs are not used, so this display shows "(unused)". In System Clock PLL **xtal** mode, this display shows "(int Xtal)" indicating that the high-rate reference is an internal crystal oscillator located on the UI board. The **clkinsrc** defaults file parameter determines the default input clock source to use at bootup.

Ref clk in rate:[*R] Shows the expected frequency of the high-rate input reference clock in MHz. This can be set to 4, 8, 16, or 32 MHz using the `userio clkin` command. It can also be set to follow the user channel data rate ("followdata"), in which case the display shows "(fd)" and the expected input frequency tracks the user data rate on mode changes. The input ref clock is used only in System Clock PLL **refclk** mode, otherwise "(unused)" is displayed. In SC PLL **xtal** mode, 32 MHz is shown because this is the frequency of the internal crystal oscillator. The **clkinrate** defaults file parameter determines the default input clock frequency to use at bootup.

Ref clk out rate:[*P] Shows the frequency of the high-rate reference clock output on the C2 cable. This can be set to 4, 8, 16, or 32 MHz (independent of the input clock rate) using the `userio clkout` command. It can also be set to follow the user channel data rate ("followdata"), in which case the display shows "(fd)" and the output clock rate tracks the user data rate on mode changes. If the Mark III formatter is active the output clock rate increases by a factor of 9/8 due to parity insertion, and always follows the user data rate regardless of the output rate setting. The **clkoutrate** defaults-file parameter determines the default output clock frequency to use at bootup.

4.2.8.7   Mark III Window (upper middle, Figure 4-8)

This window summarizes Mark III/IV related activity and switch settings. It replaces the User Parms window when you type `mk3`. Note that entries in this window are only active when the Mark III formatter is enabled (`mk3 form enable`).

(Top line): Gives the name of this window, "Diagnostic Parms (Mk3)".

Form time: Shows the Mark III time being encoded, similar to a Mark III decoder's front-panel display. The Mark III time is based on the time decoded by the current S2 diagnostic DRD when in playback (or set up for play), or the current system time otherwise. All 0's will be used if the S2 diagnostic decoder cannot extract time from the playback auxiliary data.

Form aux: Shows the Mark III auxiliary data being encoded, similar to a Mark III decoder's front-panel display. The auxiliary data consists of 12 user-specified BCD digits, followed by two BCD zeroes, followed by a two-digit hex hardware ID code. The

```
S2 Recorder |     Diagnostic Parms (Mk3)    | rec24     1998  217-21:09:46
 /\ Operating|                               |
    System   | Form time: 8 194 200243.000   | Recorder mode : 8x8-1 (br)
-------------| Form aux : bbbbbbbbbbbb0052    | Station delay : [no 1 Hz in]
 Transports  | Form CRC : cc7                 | Diag dec 3 in : channel 3 play
0*20:02:43.00|                               | Diag dec time : 194-20:02:43.00
1*20:02:43.00| Form output: play 0-7         | Diag dec sync : 100%
2*20:02:43.00| Frame rate : 400 Hz           |
3*20:02:43.00| Bad CRCs    : 68              | Est err rate : 7.8e-4   (4 ch)
4    []      | Test chan  : 0                | % data valid : 99.84%   (64 Mb/s)
5    []      | Sync detect: 100.00%   [0]    | Decoder Xbar : 01234567 (Mk3)
6    []      |-------------------------------|-------------------------------
7    []      |  <<   ■>   >>   []   ||   REC  |
-------------------------------ALIGNED-------------|
                                                   |
                                                    -------------------------------
> mk3 form enable
Mk3 formatter is enabled
> play
> mk3
> ■
--------------------------------------------------------------------------
[help   ] [time   ] [mode   ] [tapeid ] [user    ] [record ] [play   ] [--ETC--]
```

Figure 4-8: Sample ROS Screen, Mark III

12 user digits are taken from the first 12 characters of user info field 1. Only digits from 0 to 9 are allowed, any other character is mapped to hex b. It is advisable to avoid too many zeroes in a row in the auxiliary data. The hardware ID code at the end of the auxiliary data string is set at record time using the mk3 hwid command. The default value can be set in the defaults file using the **mk3hwid** parameter (initially set to 52 for "S2"). The Mark III auxiliary data will be all 0's if the S2 diagnostic decoder cannot properly extract playback auxiliary data.

Form CRC: Shows the Mark III 12-bit cyclical redundancy check being encoded, similar to a Mark III decoder's front-panel display.

Form output: Shows which S2 user data channels are active at the Mark III formatter output. This is similar to the "C2 data out" display in the User I/O window. Also indicates whether playback or record (feed-through) data is being used.

Frame rate: Shows the Mark III/IV formatter frame rate, which depends on the S2 user channel data rate: 200 Hz for 4 Mbit/s data (Mark III), 400 Hz for 8 Mbit/s data, and 800 Hz for 16 Mbit/s data (Mark IV).

Bad CRCs: Shows the number of deliberately incorrect CRCs generated since the last status reset command. The S2 software deliberately generates an invalid CRC whenever the S2 diagnostic decoder is not able to properly extract playback time or auxiliary data. (The S2 hardware also automatically invalidates the Mark III parity bits whenever the data valid signal is low.)

Test chan: This is the user data channel currently being tested by the Mark III sync detector (next line) and data extractor. Only one channel is tested at a time.

Sync detect: Shows the percentage of valid sync detected over the previous one-second period by a special Mark III sync detect circuit. This circuit counts valid syncs found in one output channel of the Mark III formatter. The number displayed in square brackets gives a cumulative count of the number of Mark III frames with missing/excess sync since the last status reset command.

4.2.9   Console Command Descriptions

This section explains each console command and its parameters. Commands are listed alphabetically, or to find a specific command use the index. Virtually all commands that set some kind of switch, register, or mode follow a general convention that the command with parameters sets the value of the switch, and the same command without parameters displays its value without changing it. For example, the `time` command with parameters sets or adjusts the time, while without parameters it displays the current time. Of course, the time is also shown on the console display.

Command syntax is indicated by listing keywords in lower-case, and numeric/string parameters in upper-case (NUM/STR) with a descriptive word in angle brackets. This is the same as the conventions used by the soft-key display. Optional parameters are enclosed in square brackets '[]' and lists of alternatives are separated by vertical bars '|'. Alternatives where exactly one must be chosen are enclosed in brace-brackets '{}'. An ellipsis '...' means that a portion of the syntax description has been omitted. Note that a number of low-level and diagnostic commands are not fully documented.

Many console commands are not needed for regular S2 operation. Most of the "extra" commands are for system configuration, testing, and unusual operation modes. Important commands are marked with "*" below. Those that are important only for record applications are marked "*R", and for playback applications "*P".

Command: **align** *P

Syntax: align {[+|-|NNN<day>-]NN<hour>:NN<min>:NN<sec>[.NUM<fracsec>]|re-align|selfalign}

Description: This is used during playback to align tapes to an absolute time or shift by some relative amount. Tape alignment works by slewing the transports and (if necessary) adjusting the station delay setting. Unlike positioning, tape alignment is a precise operation, accurate down to the data bit. Alignment is performed relative to a moving reference frame (playback), whereas positioning is performed relative to a stationary reference frame (stop). A fractional-seconds portion can be specified as part of the alignment time, and will be rounded to the nearest bit (sample). When playback first begins, a form of alignment called *self-alignment* is performed automatically. In self-alignment all transports are aligned to each other by choosing either the earliest, latest, or middle tape time as a reference — which method to use is determined by the **selfaligntarg** defaults file parameter. All transports must be within a limited distance of the target time chosen, where the maximum allowed time difference is given by the **selfaligntime** defaults-file parameter (normally 30 minutes), otherwise the alignment operation is aborted and status code STAT_ALIGNFAIL with error ERR_ALBADTIME appears. Self-alignment can also be initiated manually using the `selfalign` command option (in which case the time restriction does not apply). To skip the automatic self-alignment step, manually issue an absolute alignment or self-alignment request immediately after starting playback.

In absolute tape alignment, the user specifies a time to which the S2 should align its tapes, e.g.

        align 14:59:02.173

This tape time is referenced to the second it is received by the S2, i.e. it is the tape time that would appear if the procedure was instantaneous, but in fact alignment may take several seconds or minutes, depending on the distance to slew. Because of the asynchronous nature of the console, the user should issue the absolute align command immediately following a 1 Hz (output) clock tick, and that tick is taken as the reference for the absolute time contained in the command. While alignment is in progress the current tape time is undefined and playback data should be considered invalid. If it takes 20 seconds to align, then the first valid data will be

tagged with the requested alignment time plus 20 seconds. If a fractional-seconds portion is included in the time, the 1 Hz output tick (S1HZ_PB on C2a cable) will shift backwards by the indicated amount, e.g. in the above example the S1HZ_PB tick would start coming out 173 ms earlier.

Relative tape alignment applies a signed relative offset to the current tape time. Unlike absolute alignment, there are no special synchronization requirements for the command itself. However the tapes must already be aligned to the same tape time or relative alignment is not allowed (if necessary type `align selfalign` first). For example, to shift the tapes ahead by 29.1795453 seconds, use:

```
align  +00:00:29.1795453
```

The `realign` option can be used to re-align the tapes to the reference time chosen during the previous alignment operation should one or more transports become un-aligned, but this form of alignment normally happens automatically.

The most common reason for a tape alignment operation to fail is that the data for the time being aligned to does not actually exist on one or more tapes. In this case the affected transports might encounter blank tape, or an old or different recording, or they might run into the beginning or end of tape. In each case status code STAT_ALIGNFAIL will appear with an appropriate error code (see Page 4-73). Blank tape encountered while aligning will result in an End of Data (EOD) indication, and playback will stop. EOD is defined as approximately 30 seconds or more of blank tape encountered during positioning or alignment (slewing) in the forward direction. The affected transports will be left positioned a few seconds before the actual end of data.

The `align` command should not be used to implement delay and delay-rate corrections at a VLBI correlator, as this may interfere with the ROS automatic playback monitoring — use the `delay` command instead. `delay` should be used for any regular delay adjustments that fall within the delay-rate and acceleration limits given on Page 4-33.

Command: **barrelroll**

Syntax: barrelroll [on|off]

Description: This command is used to turn barrel-roll on and off. When on, barrel-roll rotates user data over all available transports, and un-rotates on playback, so that the possible effect of a marginal transport is averaged over all channels. Barrel roll has no effect and is considered inactive in any modes which require only one transport, e.g. 16x1-1, 8x2-2. Barrel roll must be turned off for both record and playback to obtain correct output data when playing back tapes on an RT, since an RT has only one decoder and therefore cannot reconstruct barrel-rolled data. During auto playback, barrel roll is automatically set to the same condition as at record, but during manual playback it must be set manually (use `tapeinfo` to show the required barrel roll setting). The **barrelroll** defaults-file parameter gives the default barrel roll setting to use at bootup.

Command: **tapetype**

Syntax: tapetype [STR<type> [default]|typecodes]

Description: This command is used to set which type of tape is being used. Several different types of tapes are approved for use in the S2, and the S2 needs to be informed which tape is being used due to differences in magnetic properties. Tape type codes are numbers from 1 to 9, and if additional codes are necessary the letters from A to Z will be used. See Chapter 5 for a complete list of tape types and their codes. You can get the most up-to-date list of type codes directly from the S2 by typing `tapetype typecodes` (assuming you have the latest software version). The **tapetype** parameter in the system defaults file gives the default tape type

setting to use at bootup. Note that the tape type will revert to its default value as in the defaults file next time the system is rebooted unless you specify default, which causes the defaults file to be automatically updated.

Command: **batch**

Syntax: batch [[verify] STR<filename>|edit {STR<filename>|autoexec}|delete STR<filename>|log {STR<filename>|end}|ls [long]]

Description: Begins execution of a batch file, or initiates other functions such as batch file editing or verification. e.g.

        batch myfile

executes the batch file "myfile". Use

        batch ls

to obtain a list of available batch files. If long is specified the listing includes file sizes and modify dates. The edit option calls up a full-screen editor similar to Unix 'vi' for creating new files or editing existing batch files. See Section 4.2.7 for more information on the editor. Use the verify option after editing to quickly check the syntax of the entire batch file. An alternative to editing is the log option, e.g.

        batch log common

This copies all subsequent console commands to the file "common", until you type batch log end. Then the same sequence of commands can be re-executed any time later using batch common. The delete option is used to erase batch files. Batch files can be chained (but not nested) simply by executing one from within another. To make an endless loop, have a batch file call itself. To interrupt a batch file, press Cntl-C once.

Command: **beep**

Syntax: beep [NUM|on|off]

Description: Causes the console terminal to beep, or turns beeping on and off. May be useful in batch files.

Command: **boxmode**

Syntax: boxmode [s2|c16|bita NUM|STR<filename>]

Description: For diagnostic use only. Downloads a Xilinx device connected to the UI board external download port.

Command: **clear**

Syntax: clear

Description: Clears the console interaction window.

Command: **config**

Syntax: config {{save|restore|verify|delete} STR<name>|ls [long]}

Description: Saves or restores the current S2 configuration, which includes most system switch settings except the transport tape motion state. This is useful to take a "snapshot" of the system so that it can be restored later. Configurations can be named, or if no name is given the name 'default' is used. The verify option can be used to test the integrity of a saved configuration. Software updates, in particular, may render existing configurations invalid. Use the ls option to list existing configu-

rations, with `long` to include file sizes and dates. Use `delete` to erase configurations.

    The configuration save area includes files containing transport service information that were dumped using the `transport service dumpmem tofile` command, or automatically by the system (see Section 4.).

Command: **cue**

Syntax: cue

Description: Puts the currently selected transports into a forward movement state that is about 9 times faster than playback, but slower than fast-forward. This command serves no significant purpose in normal operation and should be avoided. Use the `po-sition` command instead to perform tape positioning. The S2 must currently be in playback otherwise `cue` is not allowed. Also must not currently be aligning or positioning. Use `uncue` to stop cueing and return to normal play.

Command: **def**

Syntax: def {auxsnap|statsnap|accumulate|syncerrlocate [histogram
[NUM<startblk> NUM<endblk> [NUM<time> {s|m}]] [cumulative
[NUM<decay>]] [log10]|enable|disable|errtrigger
[NUM<errs1sec>|off]]|xbar [SSSSSSSS]|{drand|dhook|dprec|usrdv}
[on|off]|reset}

Description: Sets/displays low-level deformatter control options and switches. Strictly for diagnostic use only. The `auxsnap`, `statsnap`, and `accumulate` options display raw auxiliary data and deformatter status in various formats for the current diagnostic DRD. The `syncerrlocate` option is a data gathering and graphing facility which shows how errors are distributed across tape frames. It must first be enabled, then error location histograms can be displayed. Note: You should use manual playback and be setup to the transport in question when using the syncerrlocate facility. The histogram has various options such as display range limits, running time limit, cumulative display, exponential averaging, and logarithmic display. Cumulative values are reset each time the histogram is displayed except when `cumulative` is used alone without exponential averaging or a time limit, allowing data accumulation in the "background". See Section 6.4.6 for an example use of the syncerrlocate histogram.

    `drand` and `dprec` control the de-randomizer and de-precoder. `xbar` sets the DRD crossbar, which determines the mapping of transports/DRDs onto internal S2 data channels 0–7. The `dhook` option can be used to select digital direct-hookup, but use `setup dhook` instead. `usrdv` controls the low-level switch used to implement user data-valid (see `user dv`).

Command: **default**

Syntax: default {STR<parameter> [STRA<value>|blank] | fromfile STR<file-name>|tofile STR<filename>}

Description: This command can be used in place of the `edit defaults` command to directly view or modify parameters in the system defaults file. This may be desirable if you don't feel comfortable using the editor (which is similar to Unix vi). The defaults file contains many system startup options and parameters which allow users to customize S2 behaviour for their environment. Appendix C contains an example defaults file. Each default parameter is documented right in the file and should be self-explanatory. Entering the name of a default parameter without anything after it displays its current value, e.g.

    `default tapetype`

Specifying a new value for a parameter updates the defaults file accordingly, e.g.

    `default tapetype 3`

To clear a parameter's value, use the word `blank`. Note that you cannot add new parameters or delete existing parameters in this way (use `edit defaults` for that).

The `fromfile` and `tofile` options allow you to transfer the entire defaults file from, or to, a remote host over the network by specifying an appropriate remote file name. The remote host must be configured to accept RSH requests (or FTP logins) from the S2 system, and the host name/address and RSH userid (or FTP userid & passwd) must be entered in the S2 boot parameters (see Section 4.4.1). Changes to the defaults file take effect only after the next system boot.

Command:	**delay**[*]
Syntax:	delay [NUM] [s|ms|us|ns|bits] [relative]
Description:	Sets the station delay in absolute or relative terms. In absolute terms, this provides a positive or negative offset of up to 1/2 second between the external 1 Hz reference and the S2 internal 1 Hz timing. Relative delay setting is indicated by adding the word ''relative'', and does not require the presence of an external 1 Hz reference, whereas absolute setting does. During recording, the station delay determines the offset between the user's 1 Hz reference (S1HZ_REC on C1 cable or C1HZ on C2 cable) and the time used to label the bits on tape. During playback, the station delay determines the offset between the user's 1 Hz reference and when the corresponding data bit is output on the C2 cable. BNC jacks labeled 1Hz-REC and 1Hz-PB are provided on the S2 DSCM front panel for user measurement of the S2 internal 1 Hz tick (the appropriate jack should be used whether in record or playback). The S2 performs its own measurement of the station delay, which is displayed in the Crucial Parms window.

The delay amount can be expressed in units of seconds, milliseconds, microseconds, nanoseconds, or bits (samples) at the current user data rate, and can be entered as a floating-point number. The exact delay range limits are from -0.5 seconds to one bit short of +0.5 seconds for absolute delays, and one bit short of -1.0 second to one bit short of +1.0 second for relative delays. All delay settings are rounded off to the nearest bit. The value displayed by the `delay` command indicates the actual absolute delay implemented, after rounding. Since the station delay value shown in the Crucial Parms window is a measurement, it may not match the latest delay setting in the event of a timing glitch or other similar event (see ''Station delay'', Page 4-18). The two values also will not match in System Clock PLL **1hz** lock mode, but should be within a few hundred sample times. To show the latest delay setting, use the `delay` command without any parameters.

Non-zero station delays can be used during record operation to implement clock offsets and corrections. The System Clock PLL mode must be **refclk** (lock to external high-rate clock). Delay offsets should be entered prior to starting recording, since changing the station delay while recording has a de-stabilizing effect on the entire system timing, and may result in several seconds of lost data.

During playback the `delay` command (or its RCL equivalent DELAY_SET) can be used to implement delay and delay-rate tracking if the System Clock PLL is set to **1hz** mode (lock to external 1 Hz). Small delay changes will result in a smooth, gradual movement to the new value with all system timing and data remaining continuously valid. A typical application would be to have several S2-PTs at a correlation center fed by a common fixed 1 Hz reference, with DELAY_SET commands sent over the RCL to each PT for delay tracking. System timing and data remains valid provided that the following limits are met:

Max. delay rate: $\pm 1 \times 10^{-3}$ s/s

Max. delay acceleration: $\pm 1 \times 10^{-6}$ s/s^2

Max. delay step at 0 delay rate: $\pm 1 \times 10^{-5}$ s

You should use the `align` command instead of `delay` for large delay changes that would exceed these limits. Exceeding the limits may cause STAT_SCPLLSLIP

and/or STAT_SCPLLUNLOCK status messages indicating that the SC PLL is considered unlocked (difference between delay setting and delay measurement greater than about 6 x 10^{-5} s). Smooth delay tracking is still attempted, although internal timing problems resulting in loss of playback data become more likely in this situation. Smooth delay tracking is suspended for delay errors or steps greater than about 3 x 10^{-3} s, at which time phase-offsetting is used to adjust the delay (hence there will definitely be a glitch in the playback data). Phase-offsetting can be disabled by setting the SC PLL type to `purephase`.

For detailed information on S2 timing and playback rate control facilities please refer to the following document:
ISTS-SGL-TR94-030, "Timing and Synchronization in S2 Record and Playback Terminals" (available at *ftp://ftp.sgl.crestech.ca/pub/s2/doc/timing/timing.ps.Z*). See also the ERRMES RCL command for an alternate delay tracking method (Page A-22), and the description of the STAT_SCPLLSLIP status message on Page 4-69.

Command: **diag**

Syntax: diag {{self1|syncaux|formberc|uicberc|dcbias|drdxbar|mk3sync} [verbose]|pgstress|decode NUM<diagcode>}

Description: Initiates S2 diagnostic sequences or decodes diagnostic result codes. The `decode` option can be used to translate numeric diagnostic codes, sometimes found in the status window following the power-on self-test. The remaining options initiate specific diagnostic sequences. The `self1` option can be used to manually re-execute the power-on self-test (type `diag self1 verbose`). Without the `verbose` option, diagnostic sequences display a formal diagnostic result message (single-fault only). With `verbose`, different output is produced that is more suitable for interactive use. Most diagnostic sequences are not allowed during automatic record or playback. All of the sequences except `dcbias` and `mk3sync` run as part of the power-on self test.

The `syncaux` test checks for the presence of sync and auxiliary data in all channels of the current diagnostic loopback path. It tests internal 16 Mbit/s data channels, not user data channels, and works in any of the following setups: **dhook, ahook, bypass, record**. In paths that go through the transports, channels for which the transport is not currently selected are not tested. This test is roughly equivalent to manually entering `setup N` and `group M` for each available channel and group, and checking the "Diag dec time" and "Diag dec sync" displays.

The `formberc` test uses the Formatter diagnostic test vector (form diag mode) and bit-error counter to check that all channels of the current diagnostic loopback path contain (near-) perfect data. It tests internal 16 Mbit/s data channels, not user channels, and works in any of the following setups: **dhook, ahook, bypass, record**. In paths that go through the transports, channels for which the transport is not currently selected are not tested. This test assumes that `diag syncaux` has already been run successfully. `diag formberc` is roughly equivalent to manually entering `setup N` and `group M` for each available channel and group, and checking the bit-error rate with `form berc`. However, it is not able to spend enough time on each channel to detect anything except gross failures.

The `uicberc` test uses the UI bit-error counter to check that all active or *potentially* active user data channels contain (near-) perfect data. It requires that the S2 is currently being fed and/or is outputting the UI diagnostic test vector. To inject the UI test vector, attach an appropriate external test data generator or switch to one of the UI diagnostic modes (e.g. `mode diag8`) or one of the C2 cable test modes. `diag uicberc` tests user data channels on the UI board, not just internal data channels, and works in any of the following setups: **dhook, ahook, bypass**. The UI feedthru switch may be on or off — if on, the current setup doesn't matter. The Mk3 formatter must not be active. Note that no 32 Mbit/s modes will work unless feedthru is on because each 32 Mbit/s channel requires two internal 16 Mbit/s channels, and there is always only a single internal diagnostic path (even

in PTs), hence it is not possible to reconstruct 32 Mbit/s data. For similar reasons, this test also won't work with barrel roll on. diag uicberc is roughly equivalent to manually checking the bit-error rate with uic berc N for each potentially active user data channel, switching the setup using setup N if necessary to adjust the coverage of user output channels. Like diag formberc, diag uicberc is not able to spend enough time on each channel to detect anything except gross failures.

The dcbias test uses the UI DC bias counter to check that all active or *potentially* active user data channels contain approximately random data (between 40% and 60% DC bias). In other ways, this test is similar to diag uicberc except that it also works with the Mk3 formatter enabled. It is roughly equivalent to manually checking the DC bias with uic dcbias N for each potentially active user data channel, switching the setup using setup N if necessary to adjust the coverage of user output channels.

The drdxbar test checks all combinations of the (distributed) DRD crossbar switch to ensure that each DRD is capable of driving all 8 internal data channels (DDP). This test detects problems that would otherwise be discovered only when tapes are inserted in a certain unusual order. The drdxbar diagnostic is valid only on a PT (on an RT diag formberc covers this test, so it is unnecessary).

The mk3sync test checks all 14 Mark III output channels for valid sync using the Mark III sync detector. The Mark III formatter must be enabled (mk3 form enable).

The pgstress test is a special diagnostic which plays back reference tapes and determines the allowable range of playback PGSHIFT offsets. Before running this test you should select the desired transports and insert official PGSHIFT reference tape(s).

Command:	**dnld**
Syntax:	dnld {{form|drd|uictxdc|uicdddv|uicdqa|box} STR<filename>|ls [long]}
Description:	Explicitly downloads the indicated Xilinx chip chain. Strictly for diagnostic use only.

Command:	**dr**
Syntax:	dr [cal...|reset [all] [flat]|setdefault... |pbsel [NUM<transport>]|emulin [NUM<drd>|off] |fdbkout [NUM<transport>|off]|decsel [mlsd|bitby-bit]|vco|tdacsel [eye|error]|...]
Description:	Sets/displays low-level analog data-recovery control options and switches. For diagnostic use only. Most of these switches are set automatically as a result of other commands during regular operation. dr reset flat is sometimes useful to help stabilize the data-recovery if it is having trouble calibrating. This can happen after updating software or re-loading the Static RAM. The tdacsel switch selects one of two alternative output signals for the ERRDAC test point (SMA jack) on the DRD board front panel: eye selects the eye pattern (decoded filtered data) while error selects an error signal showing deviation from the ideal eye pattern. The default is eye.

Command:	**echo**
Syntax:	echo {off|on|prompt|[STR<txt>]}
Description:	Writes a text message to the console interaction window, useful from batch files. Also controls the echoing of commands from a batch file with the on/off options. For example, to make a batch file execute without echoing, add echo off as the first command. This will suppress both the printing of commands and their output, so if you want to see a particular command's output you will have to turn

echoing back on. Exceptions are the echo command itself, log display, and status display, whose output always appears. The prompt option of echo writes out "[Hit return to continue]" and waits for the user to press return. For examples of how to use the echo command, examine the supplied demo.pt batch file. See also the wait command.

Command: **edit**

Syntax: edit {defaults|termcap|startup.cmd}

Description: Use this to edit one of three system configuration files: **defaults**, **termcap**, and **startup.cmd**. The **defaults** file contains many system startup options which allow users to customize S2 behaviour for their environment. The **termcap** file contains terminal capability definitions in Berkeley termcap format, allowing users to add support for their console terminal type if it does not already exist. See Section 4.2.7 for more information on these two files. The **startup.cmd** file contains VxWorks shell commands to execute at boot time and might be used occasionally for advanced network operation or to implement binary software patches. You must use only VxWorks shell commands (not ROS console commands), one per line. Do not use the exit shell command anywhere in the **startup.cmd** file.

   The editor invoked by the edit command is a full-screen editor similar to Unix vi (for online help type :help while in the editor). In the event that your terminal type is not set correctly, it may be difficult or impossible to edit. In this case use the console term command to set the correct terminal type before editing.

Command: **eject**[*]

Syntax: eject

Description: Ejects the tapes from all currently selected transports. Stops the S2 first if it is not already stopped. The S2 must not be recording or record-paused. Another way to eject all the tapes is to press and hold any transport's eject button for two seconds (or more). This feature may be convenient if the console terminal is not located near the TAM racks, or the S2 is under computer (RCL) control. Eject-all should preferably be used only when all transports are stopped. No other transport front-panel buttons or switches should be touched. The **ejectalldisable** defaults file option can be used the disable the eject-all feature.

Command: **encrypt**

Syntax: encrypt STR<passwd>

Description: This command encodes a password to be used when connecting to the console using rlogin or telnet. Type the desired password and the encrypted version of the password will be displayed. Then enter the encrypted string as the value of the **loginpasswd** parameter in the defaults file. Along with **loginuserid**, this defines the user-ID and password that must be entered when connecting via rlogin or telnet.

Command: **error**

Syntax: error decode NUM<err>

Description: Used to translate numeric error codes which may occasionally appear in the status window to their corresponding text message. For a complete list of error codes see Page A-32. All ROS error codes are negative numbers. Positive numbers are not error codes, but may be status codes or diagnostic codes — see also status decode and diag decode.

Command: **expert**

Syntax: expert [on|off]

Description: Sets *expert* mode which controls whether or not non-essential and diagnostic console commands are suppressed. When expert mode is off (`expert off`), commands and options not needed for regular S2 operation do not appear in the console softkey display and may not be entered. This is useful when first learning to operate the S2 or to prevent inappropriate commands from being entered during regular use. When expert mode is on, all S2 console commands and options are available. Expert mode does not affect batch files or the RCL. The **expert** defaults-file parameter determines the default expert mode setting to use at bootup.

Command: **ff**

Syntax: ff

Description: Fast-forward winds the tapes. Stops playback first if necessary. The S2 must be in stop, play, play-pause, or rewind state otherwise fast-forward is not allowed. Also must not currently be aligning or positioning.

Command: **form**

Syntax: form [diag [on|off]|berc [NUM<chan>|rotate] [NUM<time> {frms|s|m|h}] [undetected] [log]|rand [on|off]|prec [on|off]|insw [1|2|4|8]|outdis [{NUM<chan>|all} {enable|disable}]|stuck ...|ddpsel [NUM<chan>]|frdsel [NUM<chan>]|rclk [on|off]|fdbksel [transpt|form]|fsync [rec|pb]|...]

Description: Controls the Formatter Test Vector Generator and Test Vector Receiver, and sets other low-level Formatter control options and switches. The `form` command by itself switches to the Formatter display window (replaces User Parms window, repeat the command to switch back).

The `diag` option selects Formatter "diag" mode which causes user data to be replaced by a pseudo-random noise sequence produced by a Test Vector Generator on the Formatter board. The test vector sequence can be verified with the `form berc` command which uses the Formatter Test Vector Receiver to perform a true bit-error rate measurement on one 16 Mbit/s internal S2 data channel. During playback, when up to 8 channels contain active data, the `rotate` option can be used to measure the average bit-error rate across all active channels by rapidly switching between them (time between switches is $\frac{1}{8}$ of a second). This is the next best thing to actually testing all 128 Mbits/s of data simultaneously, which is not supported in hardware. The Formatter test vector is a 127 bit maximal-length pseudo-random sequence that gets reset synchronous to the system 1 Hz tick, and like all data it is randomized before being written to tape. It can be used directly to check S2 internal data loopback paths or to record and verify test tapes. Note that `form berc` will automatically turn on Formatter diag mode for the duration of the measurement unless the S2 is in record or playback, so the `form diag` command is really only needed in preparation for recording test tapes. The Formatter test vector tests only internal S2 data paths from the Formatter to the DRD outputs, in particular it does not test UI functionality. A second type of test sequence is available on the UI board which can perform bit-error rate measurements of either user-supplied or internally generated UI test vector data (see the `uic berc` command).

The `rclk` option activates a crystal oscillator on the Formatter, thus allowing the system to run without a UI board. This should never be used for regular operation.

Command: **group**

Syntax: group [NUM]

Description: Selects which group of transports should be used in modes that don't require all 8 transports. For example, in mode 16x2-1 (which requires only 2 transports) there are 4 groups. Transports 0–1 are "group 0", transports 2–3 are "group 1", and so on (Figure 4-4). Together the mode and group settings determine which transports will be used for subsequent operations. Group switching allows the unattended recording time to be increased at lower bandwidths, e.g. with ST-182 tape in mode 16x2-1 (32 Mbits/s) at speed LP the unattended recording time is 22.8 hours. The group command is not allowed to change the group number during regular (automatic) playback or record (stop the S2 between group changes). This command is typically used following a mode command. For information on how to play back the remaining tapes when one or more tapes in a group are damaged or lost, see Page 4-9.

Command: **help**

Syntax: help [STR<cmd>]

Description: Lists all available console commands, or provides a brief one-line description of a specific command.

Command: **log**

Syntax: log [clear|stop|start|echo [STR<txt>]|display {[STR<types>] [NUM<line>|tail] [fullscreen|tofile STR<filename>]|typecodes}|auto {dv|timing} [enable|disable] |entry verifier]

Description: Controls the ROS internal log facility. The log is a time-stamped list of status changes and other events accumulated during all modes of operation. It is a circular list so that when the log is full the oldest entries are gradually overwritten. When using the console, the log is useful to keep a concrete record of unusual events without the need for continuous operator vigilance. It is a diagnostic facility only and does not replace the normal VLBI observation log. Logging would typically be used as follows. First, before starting a new sequence of S2 operations, the log may optionally be cleared using

      log clear

If you don't clear the log you may have to look through a lot of old log entries later. At any time during or after a sequence of operations use

      log display

to show the contents of the log in time order. Entries are coded using upper-case letters, and these type codes can be used for filtering the display, e.g. use

      log display S

to show only messages derived from system status, or

      log display ^ST

to show everything *except* status and timing-related messages. Enter log display typecodes for a list of the different log type codes. The tail option is useful to show just the most recent part of the log, or you can enter a line number at which to start displaying the log. With the fullscreen option the log display occupies the entire console screen instead of just the interaction window. This can be used to capture the log to a file if your terminal emulator has that capability (e.g. in ROSCO press shift-F9 followed by F2 to begin capturing incoming characters to a file). It is also possible to transfer log files over the network (see the tofile option, below). Use the stop and start options to temporarily halt and continue logging. The log command by itself shows whether logging is active and gives the current size of the log file. The maximum log file size is 350K or roughly 6000 average-size log entries. If this limit is exceeded the oldest log

entries will be overwritten in a circular fashion. The log file is kept in battery backed-up SRAM, so it will survive crashes and power failures. Note that `log clear` automatically restarts logging if it is stopped. The `tofile` option can be used to transfer log files (or parts thereof) over the network to a remote host by specifying an appropriate destination file name if your S2 system is equipped with an Ethernet board. The remote host must be configured to accept RSH requests (or FTP logins) from the S2 system, and the host name/address and RSH userid (or FTP userid & passwd) must be entered in the S2 boot parameters (see Section 4.4.1). A command to transfer the log file might look something like this:

    log display tofile host.name:/local/s2/logs/log941214

The host name with a colon is VxWorks' syntax for remote files.

Command: **logout**

Syntax: logout

Description: Closes the connection when accessing the console via the optional ethernet board using **rlogin** or **telnet**. S2 operation is not disturbed in any way, and display output reverts to the console serial port. A typical use of this is to monitor one or more S2 systems from a networked workstation, calling up their console displays only when needed using rlogin or telnet. You can force an rlogin or telnet connection to be closed from the serial console by typing cntl-C cntl-C cntl-] in quick succession (do not use this control sequence at any other time).

Command: **mk3**[*P]

Syntax: mk3 [form [enable|disable|errinduce {sync|crc} NUM<errs> NUM<frm>|showframe [NUM<frm>]|clkinv [posmidcell|negmidcell]]|hwid [SS]|dataextract [synctrig|1hztrig] NUM<offset> NUM<uchan>|dcbias NUM<uchan> [NUM<time> {s|m|h} [NUM<frms>]] [level|edge]|testchan [NUM]|nrzm [on|off]]

Description: Sets/displays Mark III control options and switches. The S2 UI board includes a Mark III/IV formatter which provides the capability to output S2 data in Mark III format, allowing direct playback of S2 tapes at Mark III/IV compatible correlators. Note that this only reformats data as it is output, i.e. the recorded data does not contain Mark III format. The command

        mk3 form enable

turns on the Mark III formatter and switches the C2a cable output to Mark III/IV compatible mode. Depending on the current S2 mode the channel data rate will either be 4.5, 9, or 18 Mbits/s, with corresponding Mark III/IV frame rates of 200, 400, and 800 Hz. The output data is NRZM encoded just like a real Mark III playback terminal. The `mk3` command by itself switches to the Mark III formatter display window (replaces the User Parms window, repeat the command to switch back). The `form clkinv` option can be used to switch the phase of the output clock to have either the positive clock edge or the negative clock edge appear at data mid-cell, depending on the requirements of the destination equipment (default is positive edge at mid-cell).

The `hwid` option is used to set the two-hex-digit Mark III hardware-ID code encoded on tape and to display its value on playback. The hardware-ID facility is provided for compatibility with Mark III VLBI operations and should be used at record time to label the tapes with the correct hardware ID, even though the Mark III formatter itself may be off, so that the correct hardware ID gets used at playback. The default hardware ID is 52, for "S2", but may be set as desired in the system defaults file. When finished, use

        mk3 form disable

to turn off the Mark III formatter and return to regular S2 operation.

The remaining command options are for low-level diagnostic use. The `dcbias`

and `dataextract` options are similar to the same options of the `uic` command, except they examine formatted Mark III/IV data. The `errinduce` option is useful when testing the interface to external Mark III/IV equipment such as a correlator or decoder. A given number of consecutive frames with sync-word or CRC errors can be introduced, starting at a particular frame number. With `errinduce sync` the usual Mark III sync word 0xffffffff is replaced by the invalid sync word 0x55555555, and with `errinduce crc` the 12-bit CRC field is replaced by an invalid CRC value. Note that the S2 also invalidates the Mark III CRC field during playback whenever the auxiliary data extracted from S2 tapes is invalid.

Command: **mode**[*]

Syntax: mode [32x4-1|...|4x4-2|c1test16|c2test8|diag4|...]

Description: Sets the S2 recorder mode, which controls the user data channelization and bit rate (sample rate). The S2 supports up to 16 channels of user data at 4, 8, 16, or 32 Mbits/s with a maximum total data rate of 128 Mbits/s. The mode setting, in conjunction with the group setting, determines which transports will be active. The mode should be set prior to recording or playback. For playback, any mode can be chosen as long as it has the same total data rate as the mode used for recording, so that the correct number of transports is selected. The precise mode will be set automatically after playback starts. There are 51 modes in total, 12 of which are test modes. The complete list is as follows:

S2 modes:

4x4-1, 4x8-1, 4x16-1,
4x4-2, 4x8-2, 4x16-2,
8x2-1, 8x4-1, 8x8-1, 8x16-1,
8x2-2, 8x4-2, 8x8-2, 8x16-2,
16x1-1, 16x2-1, 16x4-1, 16x8-1,
16x2-2, 16x4-2, 16x8-2,
32x1-1, 32x2-1, 32x4-1, 32x2-2, 32x4-2,
4i8-1, 8i4-1, 8i8-1, 16i4-1, 16i8-1,
4p8-2, 8p8-2, 16p8-2,
8d16-2, 8n16-2, 16v8-2, 32a4-2, 32b4-2

Test modes:

c1test4, c1test8, c1test16, c1test32,
c2test4, c2test8, c2test16, c2test32,
diag4, diag8, diag16, diag32

S2 mode designators are a string of the form **btu-q** where **b** is the user channel bit-rate in Mbits/s, **u** is the number of user data channels, and **q** is the quantization (1 or 2 bit). The designator **t** is a letter such as 'x', 'i', 'p', 'a', or others. 'x' indicates a regular S2 mode with user channels assigned consecutively starting at 0. 'i' stands for interleaved (every second channel used), and 'p' stands for partitioned (channels divided into 2 groups: 0-7 and 8-15). Other letters have other interpretations. Table B-5 shows which user channels are active in each S2 mode. Some S2 modes exist specifically to support certain VLBI equipment combinations, including the VSOP and RadioAstron Space VLBI missions. Additional documentation on modes for specific applications is available upon request from CRESTech/SGL, and from our FTP site *ftp://ftp.sgl.crestech.ca/* .

In addition to the numerous regular operational modes, there are 3 types of test modes, each available at various data rates. The **c1test** modes are used in record applications to verify proper connection of the C1 cable. The user's external equipment must be capable of generating the prescribed UI test vector sequence. The presence of STAT_UICVERFAIL status errors while in C1 test mode indicates a C1 cable fault. The **c2test** modes are used in playback applications to verify proper connection of the C2 cable. The user's external equipment must be capable of verifying the UI test vector sequence generated by the S2. Finally, the **diag**

modes inject the UI test vector at the UI data inputs for S2 internal testing purposes, replacing any user C1 data. (This is similar to the `form diag on` command, but uses a different test sequence). Use the `uic berc` command to verify the UI test vector. Unlike regular modes, the test modes do not have a firm notion of total bandwidth, so the defaults file parameter **diagnumtran** is used to control the associated bandwidth, i.e. how many transports to test. **diagnumtran** should be set to the number of transports being used with a particular S2 system (normally 8).

The `mode` command is not allowed during regular (automatic) playback, and can only be used to change to modes with the same total data rate during automatic record. For any other mode changes, the S2 must be stopped first. The **dfltmode** defaults-file parameter determines the initial recorder mode to use at bootup.

Command: **pause**

Syntax: pause

Description: Puts the S2 recorder into play-pause or record-pause state. For normal operation this provides no appreciable benefits and should not be used — use `stop` instead. The S2 must currently be recording or playing (and must not be aligning or positioning). Unlike `stop`, time spent in pause counts against head lifetime. To prevent excessive head and tape wear, the transports automatically stop after pausing for more than about 3 minutes.

Command: **play**[*]

Syntax: play [manual]

Description: Starts playback. The recorder must be stopped, rewinding, or fast-forwarding, otherwise the `play` command is not allowed. Tape alignment (self-alignment) is performed automatically to align the individual transports to each other and to obtain a single, coherent tape time. After this the user may issue `align` commands to further align the recorder to a desired epoch, or `delay` commands (or RCL ERRMES commands) to perform VLBI delay tracking. During playback the software monitors active transports and scans for abnormal conditions. Anything unusual is reported in the console Status window. Some of the diagnostic checks include:

- Playback sync and auxiliary data valid, including tape time
- Tape alignment maintained (automatically re-align if alignment lost)
- Auxiliary data consistent, no missing/duplicate channel IDs
- Current mode matches mode at record time (if not, switch)
- Current barrel roll setting matches record setting (if not, switch)

As indicated by the last two items above, during playback the correct mode and barrel roll settings are selected automatically. In special circumstances it may be desirable to play back using a mode that is different from the mode used at record time. The **playmodemap** defaults-file parameter provides this capability, with the restriction that both the original and the new playback mode must have the same total data rate.

The `manual` option suppresses most of the regular playback monitoring and scanning, including automatic actions such as tape alignment and mode setting. It is useful to manually examine particular playback channels, but should not be used for regular operation.

Note: When automatic playback first starts, one of the first steps is to scan all transports for auxiliary data. If for any reason auxiliary data cannot be obtained from one or more transports, automatic tape-alignment will not be attempted. To force tape alignment issue a manual alignment request, e.g. `align selfalign`. See also the general description of how to play back tapes in Section 4.2.4.1.

Command: **position**

Syntax: position [[+|-]NUM<hours>:NN<mins>:NN<secs> ...|reestablish]

Description: This command is used before playback or record to position the tapes to a specific location. Tape position is measured as the number of seconds of recorded data since the beginning of tape, and is normally specified as hours:minutes:seconds, e.g.

```
position  3:18:57
```

This format is similar to a tape time except that the hours portion normally includes only a single digit. Alternately, the position may be entered as an integer number of seconds (regardless of the setting of the **posdisformat** defaults file parameter), e.g.

```
position  11937
```

The current position is considered *unknown* after a tape is inserted or after the transports are first turned on. The position becomes known when a transport encounters beginning-of-tape (BOT), where the position is defined to be 0 (i.e. 0:00:00)[5]. Positions are normally positive for purposes of the position command, but the current position reading can take on small negative values due to anomalies around BOT. It is important to realize that sections of blank tape *do not count* in position determination. For example, on a tape which has only one hour of data recorded on it, the position 2:00:00 (2 hours) does not exist. An attempt to reach that position will result in an End of Data (EOD) indication, and the actual position will be left a few seconds before the end of data (around 1:00:00 for this example). EOD is defined as approximately 30 seconds or more of blank tape encountered during positioning or alignment (slewing) in the forward direction.

During positioning the State window shows POSITIONING. Although there may be up to 8 tapes and therefore 8 different positions, we try to work with a single position value whenever possible since the group of currently active transports will usually be in almost the same location. If it becomes necessary to specify different positions for individual transports, more than one position value can be given on the command line. Each successive position applies to the next selected transport, and if not enough positions are given the first value is re-used. e.g. In mode 16x4-1, group 1 (transports 4 through 7 selected) if three position values are given they apply to transports 4, 5 and 6 respectively, and transport 7 re-uses the first value.

There are two ways to perform positioning: *absolute* and *relative*. In addition, there are two special positioning functions *reestablish* and *preset*[6]. Absolute positioning moves the tapes to a specific position regardless of their current location. If the current position is not known, the tapes are first automatically rewound to the beginning. Relative positioning moves the tapes a given amount relative to their current position, and is indicated by a plus or minus sign. The current position does not need to be known for relative positioning to proceed. Positive position values ("+") move forward, and negative values ("-") move backward. Relative positioning may not work for small values (5 seconds or less), and repeated relative offsets may introduce cumulative errors and thus not add up exactly to the expected total.

The position reestablish option allows an unknown position to be determined without moving the tapes to a specific absolute location. The tapes are rewound and then automatically returned to their original position, as computed

---

[5] *The position ''preset'' feature will also make the position known.*

[6] *The preset function is not available with this command, see* transport position.

from the distance covered during rewind, and the position is again considered known. This may be useful following a power failure when the S2 has lost track of the current tape position, or when non-rewound tapes are inserted. It can also be used to accurately re-determine the current position if cumulative errors have built up in the position measurement after a long period of repetitive tape activity.

Command: **powermon**

Syntax: powermon

Description: Displays measurements of the VME DC power supply voltages, nominally +5 V, -5.2 V, +12 V, and -12 V. The measurements are made using an A/D converter on the Formatter board. If any of these voltages should ever exceed ±5% of nominal, status code STAT_VMEPOWFAIL will appear ("'VME xxx power supply bad").

Command: **record**^{*R}

Syntax: record [manual]

Description: Starts recording. The S2 must currently be stopped otherwise the record command is not allowed. The physical write-protect tabs on the VHS cassettes must be intact for recording to be possible. If any tabs are broken, the write-protected tapes will be automatically ejected by the transports (model AG2510) or the transports will beep 7 times and refuse to go into record (models AG2520 and AG2530) or the transport tape display will flash and they will refuse to go into record (model AG2550). In all cases the S2 status will indicate STAT_RPMONILSTATE, "Unexpected transport state change during play/record". Since breaking the record tabs will not normally be part of VLBI operational procedures, an alternate way to prevent accidental erasure of tapes is provided by setting the **writeprotect** option in the system defaults file to "on".

While recording the S2 monitors the active transports and scans for abnormal conditions. Anything unusual is reported in the system Status window. Some of the diagnostic checks include:
 – Transport servos locked
 – Transport record signal levels within allowed range
 – Record loopback data contains valid sync and auxiliary data, incl. time
The manual option suppresses most of the automatic monitoring and scanning, allowing manual examination of particular record channels. See also the general description of how to record tapes in Section 4.2.4.2.

Command: **review**

Syntax: review

Description: Puts the currently selected transports into a reverse movement state that is about 9 times faster than playback, but slower than rewind. This command serves no significant purpose in normal operation and should be avoided. Use the position command instead to perform tape positioning. The S2 must currently be in playback otherwise review is not allowed. Also must not currently be aligning or positioning. Use unreview to stop reviewing and return to normal play.

Command: **rewind***

Syntax: rewind

Description: Rewinds the tapes. Stops playback first if necessary. The S2 must be in stop, play, play-pause, or fast-forward state otherwise rewind is not allowed. Also must not currently be aligning or positioning.

Command: **scan**

Syntax: scan [aux [nomap]|time]

Description: This command is for use during manual playback only. It scans all active transports to acquire valid tape time and/or auxiliary data. If no options are specified, aux is the default, which means to scan for complete time and auxiliary data. aux nomap is similar but doesn't try to establish the DRD crossbar mapping and may work better in difficult situations. The time option is not useful on a PT since time, unlike the rest of auxiliary data, is extracted from all channels simultaneously.

Command: **scpll**

Syntax: scpll {mode [refclk|1hz|errmes|xtal|manual]|type [phase|pure-phase|freq]|dac [NUM<cv>]|div [NUM<freq>]}

Description: Sets/displays parameters associated with the System Clock Phase-Lock Loop (SC PLL). This phase-lock loop is located on the UI board and drives all S2 system timing based one of several possible types of user lock-reference signals. Only the mode option of the scpll command should be used in regular operation — the others are for testing and diagnostics. The System Clock PLL mode controls the PLL's lock reference, which can be one of **refclk, 1hz, errmes, xtal,** or **manual**. The SC PLL is normally locked to an external high-rate reference clock during record (**refclk**), or to a 1 Hz or software reference for playback rate control at a correlator (**1hz** or **errmes**). The PLL can also run independently based on an internal crystal oscillator (**xtal**) or be controlled manually for testing (**manual**). The SC PLL mode should be set as needed with the system defaults file **scpllmode** parameter and thereafter should not normally need to be changed. Although it is not an option of this command, **auto** can be used in the defaults file to automatically select **refclk** mode if an external high-rate reference clock is present at bootup and **xtal** otherwise. See also userio clkin, which selects the source of the high-rate and 1 Hz input reference clocks (C1 or C2 cable) and sets the expected frequency of the high-rate clock (4, 8, 16, 32 MHz, or followdata). Note: the high-rate reference is used only in SC PLL **refclk** mode. For detailed information on S2 timing and playback rate control facilities please refer to the document ISTS-SGL-TR94-030, "Timing and Synchronization in S2 Record and Playback Terminals" (available at *ftp://ftp.sgl.crestech.ca/pub/s2/doc/timing/timing.ps.Z*).

The type option controls the behaviour of the SC PLL when in **1hz** mode *only*. Normally the type is phase which indicates phase-locking with "phase-offsetting" enabled. Phase offsetting is simply a sudden jump in the PLL phase implemented via a station delay setting sequence when the difference between the current and desired phase is large (greater than about 3×10^{-3} s). This improves initial locking times dramatically, but causes a timing glitch and possible loss of data around the phase jump. Setting the SC PLL type to purephase disables phase-offsetting, turning the PLL into a conventional phase-lock loop. freq turns the SC PLL into a frequency-lock loop (FLL), which locks very quickly but can drift and have arbitrary phase errors. Frequency locking is employed automatically as a preliminary stage in the phase-offsetting sequence described above, and should not be used for regular operation.

The dac option of the scpll command allows manual setting of the PLL VCO control voltage when in **manual** mode. It can also be used to read the current control voltage setting in **1hz** and **errmes** modes (but not **refclk** mode).

div gives low-level access to the high-rate reference clock divisor, active only in **refclk** mode. This should normally be controlled using userio clkin instead. See also the delay command and the RCL ERRMES command.

Command: **serial**

Syntax: serial

Description: Displays the serial number assigned to this S2 system. This number can be used to keep track of S2 systems in the field, and to match tapes to the system they were recorded on. Each S2 encodes its system serial number onto tape when recording — to see the serial number extracted from tape during playback use the tapeinfo command. If you need to check a specific tape (e.g. you suspect not all tapes have the same system serial number), make sure you are in manual playback (play manual), select the desired transport using setup transport N, and then type tapeinfo. Note that each transport has its own serial number, which is also encoded on tape. You can display transport serial numbers using transport display serial to show the serial numbers for your S2 system, or transport display serialpb to show the serial numbers of the transports used to record the currently playing tapes. Thus you can trace tapes not only to specific S2 systems but also to individual transports. The S2 system serial number is stored in the system defaults file, **serialnum** parameter, and on rare occasions may become incorrect. For example, if the Static RAM board is replaced (since that's where the defaults file is stored), or if you perform a complete SRAM reload from external media. After such an operation, always check that the serial number matches the number labeled on the VME card cage, and if not edit the defaults file accordingly.

Command: **setup**

Syntax: setup [play|record|bypass|ahook|dhook] [NUM<chan>] [transport NUM<tran>]

Description: Configures the flow of data through the S2 recorder. During normal operation, the setup is controlled automatically as part of commands such as play, record, and stop, so there is no need to use this command. setup controls *internal* data channels, not user data channels. It is for diagnostic and manual operation only.

There are 6 diagnostic data paths in the S2 recorder (Figure 4-5), 5 of which can be manually selected using setup (the remaining one is selected with uic feedthru). In each path there are 8 channels numbered 0 to 7. However, during record data channels can map onto several transports, and during playback each transport can potentially map onto any data channel. Therefore the setup incorporates separate concepts of *transport number* and *channel number*. The channel number is the internal data channel on which the data belongs, from which there is a direct mapping onto one or more user channels. The transport number is the transport which the data happens to be going to or coming from. The channel and transport chosen in the current setup are referred to as the *diagnostic channel* and *diagnostic transport*. In a playback terminal the decoder (DRD) connected to the diagnostic transport is the *diagnostic decoder*. The effect of changing the setup is to change the diagnostic loopback path and/or the current diagnostic decoder. The "Decoder input" display in the Crucial Parms window shows what the current setup is. The setup also controls whether the system timing is set for playback or record. Playback timing is used when playback data from the transports is being decoded, and record timing is used at all other times.

There are four typical ways the setup command might be used. First, you can change the diagnostic loopback path, e.g.

        setup   ahook

for "analog direct-hookup" or

        setup   bypass

for "transport bypass" path. Other options are dhook, record, and play. If the S2 is stopped, setting bypass, record, or play also selects the *stop setup*. The stop setup is the setup used when the S2 is stopped (or rewinding, etc.) and

is treated specially in that it is variable. The stop setup is normally **bypass** on an S2-RT and **play** on an S2-PT. This ensures that the system timing does not get disturbed unnecessarily when going between play/record and stop, which may be desirable if external systems are locked to the output 1 Hz tick (S1HZ_PB on C2a cable).

Another way to use `setup` is to change the current diagnostic channel or transport. For example, during manual playback,

        setup 3

selects channel 3, automatically locating the correct transport. Similarly,

        setup transport 1

selects transport 1, automatically choosing the correct channel. The active range of channel and transport numbers is controlled by the `mode` and `group` commands. Finally, you can specify both the channel and transport, effectively forcing a particular decoder output mapping, e.g.

        setup 5 transport 2

The above command may generate status messages such as "Tape channel ID does not match setup", indicating that the setup is not correct for the data currently being decoded. This is considered an inconsistent setup, and can be remedied by typing `setup` with no parameters. Besides displaying the current setup, this will fix it if it's inconsistent. The `setup` command is not allowed during automatic record or playback.

Command: **shell**

Syntax: shell

Description: Escape to the VxWorks shell without interrupting S2 operation. This is strictly for diagnostic use only and requires knowledge of the VxWorks shell syntax. Type `exit` when finished to return to ROS.

Command: **shutdown**

Syntax: shutdown [reboot|rcl]

Description: Terminates the execution of ROS and enters the VxWorks shell, or with the `reboot` option causes an immediate reboot. This should be used for diagnostic and debugging purposes only, or as the last command before turning off power to the system. Warning: this instantly disables the S2 recorder, halting any record, playback or other operation that may be in progress. The only recommended way to restart the system following `shutdown` (without the reboot option) is a full system reset. There are several ways to do this: turn the power off and on, press the small red "reset" button on the RCC board, or type Cntl-_ on the console (control-underscore, may require shift as well). Typing `reboot` in the VxWorks shell will also work. Use the `shell` command instead of `shutdown` if you just want to issue a few VxWorks shell commands and then return to ROS (for experts only!).

The `rcl` option closes any currently active network RCL connection, useful to clean up hung RCL connections. It does not shut down the rest of ROS.

Command: **sleep**

Syntax: sleep {NUM<seconds>|[+][NNN<day>-]NN<hour>:NN<min>:NN<sec>}

Description: Delays for a certain amount of time or waits until a specific time. This is useful in batch files to introduce delays and make commands execute in a precisely scheduled manner. There are 3 general formats of the sleep command. The simplest is to give a number of seconds to delay, e.g.

        sleep 5

causes a delay of 5 seconds. There is also a long-form relative format, e.g.

```
        sleep  +03:45:30
```
delays for 3 hours, 45 minutes, and 30 seconds. The third format is a long-form absolute sleep command, e.g.
```
        sleep  189-23:00:00
```
which waits until 11:00 pm on day number 189. The time to wait is not allowed to exceed 30 days, and a warning is printed if it is more than 1 day. The day number can be omitted, so just
```
        sleep  23:00:00
```
will wait until 11:00 pm today (or tomorrow if the current time is between 11:00 and midnight). See also the `wait` command.

Tape	LP	SLP
ST-126/SE-180 (258m)	4 h 01 m	6 h 02 m
ST-182/SE-260 (371m)	5 h 47 m	8 h 41 m

Command: **speed**[*R]

Syntax: speed [lp|slp]

Description: Selects the tape speed to use for recording (playback tape speed is determined automatically). S2 systems with AG2530 and earlier model transports support two speeds: **lp** and **slp**, corresponding to the standard VHS tape speeds of the same name (but sped-up by a factor of 64/60). Systems with AG2550 and later model transports can record in **slp** only, but will play back tapes recorded in either speed. If **lp** is selected on a system with one or more AG2550 transports it is automatically mapped to **slp**. Note that although AG2510 transports can record at either speed, using AG2510s to play back **slp** tapes is not recommended due to poor performance. The total recording time per tape is given above for both speeds and two different available tape lengths. Refer to Chapter 5 for additional information.

Command: **station**

Syntax: station [NUM [STR8<nickname>|blank]]

Description: Sets the station number and nickname of this S2 system. The station number is an integer from 0 to 255 which serves to identify different S2 systems in applications where more than one S2 is used. Non-zero station numbers are displayed on the console screen just above the Transport window and below the activity indicator, and on the transport front-panels beside the transport address (values 1 to 6 only). The station number is encoded on tape so that it can be recovered during playback (see the `tapeinfo` command). The number is remembered across power-downs since it is stored in the system defaults file (this command automatically updates the defaults file). Select the value 0 if you do not wish to assign a station number (no number will be displayed).

In addition to the station number, each system can also be assigned a "nickname" of up to 8 characters. It appears in the console window to the left of the system time. Like the station number, the nickname is automatically updated in the system defaults file and thus persists across power-downs, but it is not encoded onto tape. Enter the word `blank` to clear the nickname. Both the station number and nickname can also be set by editing the **station** and **nickname** entries in the defaults file (see Section 4.2.7).

Command: **status**

Syntax: status {reset|pagedn|pageup|decode NUM<statcode> [short]|display [fullscreen|tofile STR<filename>] [short]}

Description: Provides several options for controlling the console status display. `status reset` is the form of this command you will probably use most often. It clears "transient" (clear-on-read) conditions from the console status window and zeroes all miscellaneous error counters. Note that `status reset` affects only the console status indications, not the RCL status (for the RCL, clear-on-read status conditions are cleared each time status is read). The `pagedn` and `pageup` options allow you to page through the status messages in case there are more than 3 of them, although the most important conditions will usually be shown first. As a convenience feature, when running the ROSCO terminal emulator program the PgUp and PgDn keyboard keys automatically generate the commands `status pageup` and `status pagedn`. Another way to show all status information is with the `display` option, which writes all current status conditions to the interaction window (or to the log with `tolog`). Normally the "long" (RCL) version of the status messages is given, but `short` can be specified to see the "short" form which is the form used in the console status window. The numeric status code and mnemonic are both shown, making it easy to look up the code in the status descriptions (Section 4.5). By specifying `fullscreen`, the information can occupy the entire terminal screen to get around size limitations of the interaction window. This can be used to capture status information to a file if your terminal emulator has that capability (e.g. in ROSCO press shift-F9 followed by F2 to begin capturing incoming characters to a file). The `tolog` option writes the current status information to the S2 internal log file for later examination. Alternatively with the `tofile` option you can write status information over the network directly to a file on a remote computer if your S2 system is equipped with an Ethernet board and the remote host has been configured appropriately (see the description of the `tofile` option under the `log` command for more information).

     The `decode` option is used to translate numeric status codes to their corresponding text message, either long or short format. This may occasionally be useful if a status code appears in numeric form due to space limitations, but `status display` is normally preferred as long as the status condition is still active since `status decode` does not show occurrence-specific information such as channel or transport numbers which may be inserted into the message. Places where such information gets inserted are marked by "xxx". For a complete listing and explanation of all status codes see Section 4.5.

Command: **stop**[*]

Syntax: stop

Description: Stops the S2 if it is currently in record, playback, rewind, fast-forward, or any other tape motion state including alignment and positioning. Automatically switches to the stop setup, which is normally **bypass** on an RT and **play** on a PT. The stop setup can be changed simply by selecting a new diagnostic signal path with the `setup` command while the S2 is stopped. The **stopsetup** defaults-file parameter can be used to define the stop setup to use at bootup.

Command: **tapeid**[*]

Syntax: tapeid [STR20|blank]

Description: Sets the Tape ID, shown at the top of the console User Parms window. During record the Tape ID is written continuously to the S2 auxiliary data channel, and during playback it is recovered and displayed along with other parameters such as User Info. The Tape ID should be chosen so that it uniquely identifies the set

of tapes being recorded, and should not exceed 20 characters. For example, the Canadian tape labeling convention defines the Tape ID to contain a two-letter country code, the tape type code, and a tape set index number with checksum as follows:

CA-3-0012307

(More information on the Canadian tape labeling conventions can be found in the document DRAO-RPCS3001, available at *http://www.drao.nrc.ca/web/vlbi/docs/doc3001.ps.gz*). Specify `blank` to enter an empty Tape ID (the `tapeid` command with no parameters displays its current value). The Tape ID should be entered prior to the start of recording and thereafter should not be changed for the remainder of the tape. Following this rule will make it easy to determine if all tapes belong to the same set at playback, regardless of their current positions. If this rule is not followed, status code STAT_INTAPEID may appear during playback indicating that two transports have different tape IDs, but this will have no adverse effect on system operation.

Command: **tapeinfo**

Syntax: tapeinfo

Description: During playback, `tapeinfo` displays information extracted from tape auxiliary data describing various conditions in effect at record time. The information applies to the tape in the current diagnostic transport, as shown in the "Diag dec in" display. Most of the tape info applies equally to all tapes assuming that they were all recorded together and are currently aligned. The following items are displayed:

1. The Tape ID (also shown in the User Parms window).
2. The channel ID, which is the address of the transport the tape was recorded on.
3. The recorder mode at record time.
4. The S2 system serial number, station number, and transport serial number where the tape was recorded. Useful for tracing tapes back to the systems they were recorded on.
5. The current tape time, and an indicator showing if time was not validated during record.
6. The tape type setting in effect at the time of recording (type codes 1-9 and A-Z only, no extended digit strings).
7. The elapsed time since the start of recording.
8. The record start time.
9. The record stop time, if known (not implemented).
10. The measured station delay at that point in the recording (shown as "unknown" if there was no 1 Hz input at record time).
11. The barrel roll setting (on/off).
12. The Formatter randomizer and precoder settings (both normally always on).
13. The Formatter diagnostic mode setting (on/off). On only for test tapes, indicates tape contains Formatter TVG pseudo-random noise sequence.

Command: **term**

Syntax: term [STR<type> [default]]

Description: Sets the console terminal type. If the terminal type is not set correctly the console display may appear garbled, and editing with the `edit` or `batch edit` commands may be almost impossible. Note that the terminal type will revert to its default value as entered in the system defaults file next time the system is rebooted unless you specify `default`, which causes the defaults file to be automatically updated.

Terminal types entered with this command must appear in the terminal capability (termcap) file or they will not be recognized. Table 4-2 shows some popular terminal types recognized by the S2 software as shipped. The termcap file can

be edited by typing `edit termcap`, but if the terminal you are using is not supported by the S2 you will need to add a definition for it to the termcap file *without* using the editor. This can be done via the S2 install procedure, which is invoked from the VxWorks shell by typing `install` (use the console `shell` command or press the space bar at the appropriate part of the bootup to enter the shell). See Section 4.2.7 and Appendix D for more information on termcaps.

Command: **time**[*]

Syntax: time [[NNN<day>-]NN<hour>:NN<min>:NN<sec>
|{s|m|h|d|y}[+|-]NUM|validate
|convert [NUM<day#> [NUM<year>]
    |{jan|feb|...|dec} NUM<day> [NUM<year>]]|setremote [STR<host-nameaddr>]]

Description: Sets and displays the S2 system time. The system time is used to time-tag data recorded on tape, and is normally set precisely to UTC before recording. For playback the time need not be set precisely, however setting the time is still recommended since it is used by the `sleep` command and the ROS internal log facility. Without parameters `time` displays the current system time as well as the total elapsed time since the S2 was booted (the "uptime"). If tapes are currently playing and aligned the complete playback tape time is shown also (this provides a quick way to check the year of the recording, which is not normally shown on the display). By specifying parameters the time can be set to a specific value, e.g.

        time 159-11:47:00

Here 159 is the day number (where day 1 is Jan 1), 11 is the hours, 47 is the minutes, and 00 is the seconds. When setting absolute times in this way the command should be issued immediately *after* the S-1/C-1 Hz input tick (S1HZ_REC on C1 cable or C1HZ on C2a cable). For the above example you would press return just after the reference clock's seconds digits flipped to 00. The year must be set separately as follows (but this should rarely be necessary):

        time y 1996

Each time field can also be adjusted in absolute or relative terms, e.g.

        time m 23

sets the minutes field to 23, or

        time s+1

increments the time by one second. While the S2 power is off time is maintained by a battery backed-up clock chip (MK48T02) on the RCC board, so it should usually remain correct within a few seconds. Following system bootup (and remote time setting and certain severe system timing glitches) two question-marks "??" are shown beside the system time indicating that it may not be accurate. Use the `time` command to set it, or type `time validate` if it is actually correct. The `convert` option is useful to convert between day numbers and the more familiar month/day representation. The `setremote` option can be used to set the time from a remote network host using the internet RDATE protocol (IP port number 37). Your S2 system must be equipped with an Ethernet board and connected to a network from which the host to obtain the time from is accessible. The name or internet address of the time host can be specified in the command, or if no name is given the value entered for the **timehost** parameter in the system defaults file will be used. If the **timehost** defaults parameter is missing or blank, the host name and address from the boot parameters will be used instead. When specifying the time host it will usually be necessary to use the internet address and not the host name since the required name translation information will not be present (unless you have taken steps to add it). See the general description of time setting on Page 4-10 for more information.

Command: **transport**

Syntax: transport [[{all|autoselect|NUM<addr>}[{+|-}NUM<addr>...[{+|-}NUM<addr>]]] [status {version|serial|state|position|time|volt|recsig|pgshift|wrlev} [log]|stop|play|...|position {[+|-]NUM<hours>:NN<mins>:NN<secs>|preset {NUM<hours>:NN<mins>:NN<secs>|unknown}|reestablish}|tracking [NUM<amount>|center|recalibrate [phase2only]|{auto|manual} [phase1|phase2]]|wrlev [NUM|optimize ...]|program {[alternate] STR<filename>|ls [long]}|caltemp NUM<degC>|pgshift {recval|playval} NUM<us>|wrlevoffset NUM|options {takeupsense [enable|disable]|wrlevoffset [{enable|disable} [default]]}|service {lastserv|lasthead|edithist|dumpmem {tofile|...}|loadmem NUM<oldserial>|cleartimes}]|display [time|day|year |position|esterr|dv|agclevel|chanid|temp|servtime|headtime|ontime|tracking|serial|serialpb|state]|assignaddr NUM<serial> NUM<addr>]

Description: Sets/displays low-level transport control options and switches. Most of these are for diagnostic and system configuration use. Many functions provided here are low-level versions of regular operational commands and should *never* be used for regular operation. The command transport play (for example) is *not* the same as play, and should never be used in its place. The following descriptions include only the most commonly used command options.

The display option selects what to display in the Transport window. Possibilities include individual positions, estimated error rate, AGC signal level, tape channel ID, transport temperature, and head use time. For a complete list see the description of the transport window on Page 4-24.

Another function of the transport command is to select sets of transports that are different from the automatic selections normally made by the mode and group commands. This may be useful in unusual situations such as when one of the tapes in a tape set has been lost but you still wish to play back the remaining ones. For example,

        transport 3+6+7
selects only transports 3, 6, and 7, and

        transport all-7
selects all transports that are not dead or off, except number 7. After such a command, type

        transport autoselect
to return to the set of transports that were automatically selected.

The status option shows internal information about the transports. For example,

        transport status volt
shows measurements of the ±5 V supplies from the C4 cable, plus the current temperature and AGC (signal) level.

        transport status version
shows the current software version running on the 8051 custom processor inside the transports. Most other status information is a duplicate of information available in the Transport window via transport display.

The assignaddr option is used to set or change a transport's address, which must be done every time a transport or TCP board is replaced. The address is what distinguishes control messages to different transports, and must match the C3/C4 cable channel the transport is connected to (depends on its position in the TAM racks). The transport's TCP serial number must be known to assign its address. For example,

        transport assignaddr 2053 3
assigns the address 3 to the transport with serial number tcp2053. The serial number is printed on a label attached to the back of the transport. See Section 6.4 for more information on how to assign addresses.

The `program` option is used to re-load the software which runs on the 8051 processor inside the transports. This software is stored in electrically-erasable memory (EEPROM) and may need to be re-loaded following a system software update (Section 6.2). The normal command to re-load the software of all transports is

        `transport  all  program`

This will first display the version number and date of the software currently loaded in the transports, followed by the version number and date of the new transport software about to be loaded, and prompt for confirmation. If the response is "yes", the transports will be re-loaded one after the other. Each is first placed in a special programming mode where it recognizes nothing but program data, then the data is transferred. After the last data record is sent the transport reboots itself. If any transport fails to respond within 30 seconds of being programmed the entire operation will abort with an error message. In this case some transports may not yet have been re-loaded, and one transport may be dead or only partially programmed. After successfully programming all eight transports you should see this message: "Transports 0-7: Programmed prog & all".

The transport software is actually broken down into a number of different modules, and it is possible to give the name of a specific module to re-load. This should be done by expert service personnel only. A combination of all the modules is called **all.ag25x0**, a different version of which exists for each transport type. Two identical programming modules are used for downloading the EEPROM, with one being used to download the other when the programming modules themselves need to be updated; their names are **prog.tcp** and **progalt.tcp** for the "standard" and "alternate" programming modules, respectively. Two other modules exist, **main.ag25x0** and **xil.ag25x0**. These two modules and **progalt.tcp** are not normally included in S2 software distributions to save space (in any case they are part of **all.ag25x0**). Care must be taken when programming individual transport modules, especially when using **all.ag25x0**, because it is possible to make a mistake and kill a transport. Only use **all.ag25x0** if you are sure the old and new programming modules are exactly identical. If not, you must download one of the programming modules first. Of course you can avoid these problems by using the standard programming sequence `transport N program` described above, since this automatically loads the standard programming module and then loads **all.ag25x0**. In the event that the EEPROM is corrupted a transport may appear dead (will not beep and set its front panel address display a few seconds after power-on), making it impossible to issue the `transport program` command to repair the damage. In this case it is sometimes possible to recover using a special "emergency recovery" feature: Hold the eject button on the front panel while applying power to the transport. This will force it into program mode (standard programming module), after which you should use `transport N program all.ag25x0`. You will get a warning ("assuming transport already in program mode"), but this is normal.

The `position` option is a low-level version of the `position` command, and should not normally be used. However, it does support the position preset feature which is intentionally left out of the regular `position` command. Normally position preset will be done via the RCL.

The `tracking` option can be used to manually adjust the tracking control on the transports. Each integer tracking unit is equivalent to pressing a tracking button once. Tracking adjustments are always relative, but the absolute range is approximately -50 to +50. Use the `recalibrate` sub-option to trigger automatic tracking calibration. There are two automatic tracking algorithms: phase-1 is performed by the transport (green DIGITAL TRACKING light flashes, if present), and phase-2 is performed by ROS ("track cal" message appears in Diag dec sync display). Each phase can be individually enabled and disabled for particular transports using `transport tracking auto` and `transport tracking manual`. There is also a special phase-1b algorithm which performs large tracking jumps of 20 units or more that may be triggered by ROS when the system is

having difficulty with a particular tape. Phase-1b can be globally disabled using `transport tracking phase1b disable`. Note: The tracking `center` option will not actually center the tracking, but rather acts as a low-level trigger for the phase-1 tracking algorithm. It is equivalent to pressing the two tracking buttons simultaneously, and should not normally be used.

The `caltemp` option is used to calibrate the temperature sensors inside the transports. This is normally done only once during manufacture, but in some circumstances might need to be done again in the field. The current correct transport internal temperature in degrees Celsius must be specified (the measured ambient temperature can be used immediately after turning on the S2 system, before the transports have had a chance to warm up; for the complete temperature calibration procedure see Section 6.4.5). The value 0 has a special meaning: it clears the temperature calibration factor so that the raw, uncalibrated temperature reading is shown. Type `transport display temp` to display the transport temperatures. The transports automatically turn off above 60 degrees or below 5 degrees C. In rare cases if a transport has shut itself off but is known not to exceed the temperature and condensation limits, it could be that the temperature calibration factor has been corrupted and needs to be re-entered.

The `wrlev` option allows you to explicitly set the write current level used during record, however the write current is normally set according to the tape type and should not be changed. By using the word 'optimize' (`transport wrlev optimize`) you can run a special optimization routine that automatically determines the best individual write current offset for each transport. This can significantly improve bit-error rates compared to operating without offsets. You can also set the offset manually using the `wrlevoffset` option, but this is not recommended. To display the write current value and offset for one or more transports use `transport status wrlev`. See Section 6.4.4 for more information on write current optimization.

The `options` option allows you to turn on and off specific software features of the transport. This should not normally be necessary except in rare cases to work around unusual problems. Under `options`, the `takeupsense` option switch lets you turn off the TCP takeup reel sensing software and prevent tape motion fail errors (status code STAT_TRANTPMOTFAIL) in the event that these errors are occurring erroneously. The `wrlevoffset` option switch lets you turn off per-transport write current offsets, effectively making the per-transport offset zero. The defaults file **wrlevoffoption** parameter controls the state of the `wrlevoffset` option switches at bootup.

The `service` option has a number of sub-options to be used by service personnel for various tasks related to maintaining and repairing transports. They will be described briefly here; for more information see Section 6.4.1.

```
transport N service lastserv
transport N service lasthead
```

These two commands zero the time since last service (*in-service* time) and time since last head replacement (*head-use* time), respectively, which are stored in the EEPROM inside each transport. A *service* is defined as any time the transport case is opened for inspection or maintenance such as cleaning/greasing, and should be indicated by typing `transport N service lastserv` as soon as the transport is back in operation. A head replacement is defined as any time the upper head cylinder is replaced, and by definition also implies a service. It should be indicated by typing `transport N service lasthead` after the transport is back in operation. Both the in-service and head-use times are measured in hours and minutes of active head use, and can be displayed by typing `transport N status time`. In addition to the in-service and head-use times, this also shows the total cumulative head use and total on time since manufacture.

```
transport N service edithist
```

This command allows you to edit a text file containing service history information

that is stored inside each transport (in the EEPROM). This file should be updated each time a transport is serviced, cleaned, repaired, etc. Follow the existing format and include the current total head time and nominal service hours as shown. The maximum allowed file size is about 2 kilobytes.

```
transport  N  service  dumpmem  tofile
transport  N  service  loadmem  NNNN
```

These two commands should be used when replacing or swapping a transport's TCP board. They allow all of the service and calibration information stored in the EEPROM to be transferred from the old TCP board to the new board (the TCP board holds the EEPROM itself). Otherwise important information such as head use time, service history, and pgshift and write current calibrations would be lost. The `transport N service dumpmem tofile` command should be issued before replacing the TCP, and the `transport N service loadmem NNNN` command should be issued after replacing the TCP and reassigning its address, where NNNN is the *old* TCP serial number (swapping the TCP means that the serial number changes!). Note that if a TCP board has failed and it is not possible to carry out the initial dumpmem step, you can still transfer recent values for the EEPROM parameters since the S2 performs a dumpmem for each transport on every system boot (just in case).

```
transport  N  service  cleartimes
```

This command can be used to zero all head use and service time counters, *including* the totals. This should only be done if an older TCP is installed in a brand new transport, or the counter values have become corrupted for some reason.

Command: **uic**

Syntax: uic [dcbias NUM<uchan> [NUM<time> {s|m|h} [NUM<frms>]]]
[level|edge]|berc NUM<uchan> [NUM<time> {s|m|h} [NUM<frms>]] [unde-
tected] [log]|{dataextract|validityextract} NUM<offset>
NUM<uchan>|feedthru [on|off]|quant [1|2]|...]

Description: Sets/displays low-level UI (User-Interface) board control options and switches. Most of these are for testing and diagnostics, but a few may be useful during regular operation. The `dcbias` option performs a DC bias measurement on a specific user data channel, and the `berc` option performs a true bit-error rate measurement, provided that the current user output data contains the UI diagnostic test vector. The UI test vector is a 32767 bit maximal-length pseudo-random sequence that gets reset synchronous to the system 1 Hz tick. It is not the same as the Formatter test vector tested by `form berc`. To inject the UI test vector in place of user input data, attach an appropriate external test data generator or switch to one of the UI diag modes (e.g. `mode diag8`) or one of the C2 cable test modes. Note that unlike `form berc`, `uic berc` will not automatically enable UI diag mode.

The `dataextract` and `validityextract` options take actual 24-bit snap-shots of the user data or validity signals. Extractions are synchronous to the 1 Hz tick (frame 0), with an optional positive offset of up to 255 bits. The data extractor can be used to check proper connection to a data acquisition system before re-cording. For example, the data acquisition hardware could be configured to generate a sine wave of appropriate frequency at the sampler inputs, which should appear as a square wave in the S2 data extractor display (UI feed-through, described next, should be turned on). The `feedthru` option selects the UI feed-through diagnostic path, which results in all C1-cable input data being duplicated on the C2 cable. In feed-through mode, the DC bias, UI BERC, and data-extract facilities examine feed-through data instead of data from the decoders (DRDs). This ensures that all user data channels are accessible and provides the most direct way to examine data from the C1 cable.

Command: **uncue**

Syntax: uncue

Description: Returns to regular playback from cue or review. Currently selected transports must be cueing or reviewing. Should not be used for regular operation.

Command: **unpause**

Syntax: unpause

Description: Returns to regular playback or record from pause. Currently selected transports must be in play-pause or record-pause. Should not be used for regular operation.

Command: **unreview**

Syntax: unreview

Description: Returns to regular playback from cue or review. Currently selected transports must be cueing or reviewing. Should not be used for regular operation.

Command: **user**[*]

Syntax: user [info {1 [STR16|blank|label {STR16|blank}]|2 [STR16|blank|label {STR16|blank}]|3 [STR32|blank|label {STR16|blank}]|4 [STR48|blank|label {STR16|blank}]}|dv {[yes|no]|playback [enable|disable]}]

Description: Sets/displays the value of one of 4 user-defined information fields and labels, or the user data-valid flag. User info is shown in the console User Parms window (except field 4). During record this information is written continuously to the S2 auxiliary data channel. During playback it may be recovered at any position on the tape. The format of the user info fields is totally up to the user as long as the overall length does not exceed 16, 16, 32, 48 characters for fields 1 through 4 respectively, and 16 characters for each label. Specify blank to clear an info field or label (the command with no parameters displays its current value). User info should be entered prior to the start of recording and thereafter may be changed at any time, but should not be changed more often than every few seconds. See "Tape ID and User Info" on page 4-12 for more information, including some examples.

The dv option of the user command is used at record time to set a boolean flag which indicates whether data is valid or invalid. Then at playback time the hardware data valid (DV) signals will mark the data invalid whenever the user DV flag was false. User DV is just one of the factors contributing to the final playback data validity decision. This feature can be used whenever it is known that bad data is being recorded, e.g. the telescope is slewing or off-source. To use the so-called bad data anyway, the user DV flag can be ignored during playback by typing user dv playback disable. This may be useful if data was erroneously marked invalid at record time. The playback enable/disable function has no effect during record.

Command: **userio**[*]

Syntax: userio [clkin [c1|c2] [32|16|8|4|followdata]|clkout [32|16|8|4|followdata|syncerrinduce [on|off]]|{c1in|c2out} [on|off]]

Description: Without any parameters, this command switches to the User I/O window, which replaces the User Parms window in the upper middle of the screen. It gives useful information about the current User Interface input/output configuration, including exactly which user data channels are active and what clocking options are currently in effect. See Section 4.2.8.6 for a description of the User I/O window.

The parameters of the `userio` command set/display various input/output options. `clkin` selects the source of the high-rate and 1 Hz input reference clocks (C1 or C2 cable) and sets the expected frequency of the high-rate input reference. This can be fixed at 4, 8, 16, or 32 MHz, or set to `followdata` which means track the user channel data rate. These input clocking parameters are normally set as desired in the system defaults file using the **clkinsrc** and **clkinrate** options. Note that the high-rate reference clock is used only in System Clock PLL **refclk** mode, and the 1 Hz reference is used only in SC PLL modes **refclk** and **1hz**.

The `clkout` option sets the frequency of the high-rate reference clock output on the C2 cable (SCLK_PB), either 4, 8, 16, or 32 MHz, or `followdata`. If the Mark III/IV formatter is active, the output clock can't be set to anything except `followdata`, and the output clock rate increases by a factor of 9/8 due to parity insertion. The default output clock rate is set by the **clkoutrate** defaults file parameter.

The `clin` option is used to enable and disable the C1-cable data inputs, while `c2out` enables and disables the C2-cable data and validity outputs. These may be useful when testing the interface between the S2 and a correlator or between the S2 and a data acquisition system. Note that in both cases the timing signals (clock and 1 Hz) are unaffected.

The `syncerrinduce` option may be used to test the correlator timing interface. Turning syncerrinduce on introduces a timing offset error in the 1 Hz output (S1HZ_PB on C2a cable) relative to the data (unlike the `delay` command which shifts data and clock together when in playback). The amount of error offset is 250 ns in 4, 8, and 16 Mbit/s modes and 62.5 ns in 32 Mbit/s modes. Don't forget to turn it off when you are done.

For other commands which affect the S2's input/output configuration, see the `mode` command, which sets the user data rate and channelization, and the `scpll mode` command, which selects the System Clock PLL lock reference.

Command:	**verbose**
Syntax:	verbose [NUM<level>|countall [on|off]]
Description:	Sets/displays the verbose level, which controls the amount of information displayed in the console status window by filtering out certain messages. The valid range of verbose levels is from 0 to 3, where 3 gives the most information and 0 the least. Level 1 is appropriate for normal operation. Level 0 eliminates some important status messages, and therefore is not recommended. The verbose level also filters a small number of RCL status conditions, but in general has less effect on the RCL than on the console. The verbose level should be set to 1 for normal RCL operation. The initial verbose level is defined in the system defaults file by the **verbose** parameter.

Command:	**version**
Syntax:	version
Description:	Displays the Recorder Operating System software version number, including CPU type and compilation date/time. This can be used along with the boot-EPROM and VxWorks version numbers (displayed during bootup) and the transport software version (displayed with `transport all status version`) to build a complete picture of the software revision status.

Command: **wait**

Syntax: wait {stop|aligndone|drcal}

Description: Waits until a specified condition is true. This command is useful in batch files where `sleep` commands of uncertain length would otherwise be required. Currently the following options are supported: `stop`, `aligndone`, `drcal`.

`wait stop` waits until the overall tape motion state changes to stop. This can be used in a number of different situations, e.g.

— Wait for rewinding to complete.

— Wait for positioning to complete.

— Wait for playback or record to hit end-of-tape.

— If there are no tapes in the transports, wait for user to insert tapes.

`wait aligndone` waits until tape alignment completes during automatic play (successfully or not). It is needed because in automatic play tape alignment happens in the "background", i.e. the alignment command returns immediately. In manual play alignment commands do not return until alignment is complete, so `wait aligndone` is not needed (but no harm is done, it will just return immediately). Sometimes `wait aligndone` may not work as desired immediately after a `play` command: since no alignment operation has started yet, it returns right away. The solution is to issue an `align selfalign` command first.

`wait drcal` waits until the data recovery parameters of the current diagnostic transport/DRD have been calibrated, including phase-2 tracking if applicable. Intended for use in manual play. Should be avoided in automatic play/record.

Note: Use `echo prompt` to wait for the user to hit return.

4.3 RCL Operation

The basic approach to RCL operation is similar to console operation except that there is no console display to "look" at. Instead, desired display quantities must be requested using specific RCL commands. The only request that needs to be made regularly (about every 1 to 5 seconds) is the STATUS command, which is the RCL equivalent of the console status window. Other quantities can be requested whenever desired to verify current operation, e.g. STATE_READ to read the recorder state, or TIME_READ to verify the system time (TIME_READ_PB for playback tape time). The RCL is not meant to be used when performing system configuration or debugging, since it deliberately has a somewhat simplified interface. It tries to present the view of a single recorder, with little mention of the fact that there are actually 8 individual transports. In rare instances this may be too restrictive, so there is a special RCL facility to execute single console command strings, thus allowing low-level access. Refer to the RCL protocol specification, given in Appendix A, for more information.

The S2 RCL port is wired as an RS-232C DTE device and should be connected to a DCE device (this is the opposite of the console port). Some computers, such as IBM PCs, have only DTE serial ports and will require a null modem cable. Multi-drop RS-485 connection is possible with the use of an external conversion box. On S2 systems that have an Ethernet board an equivalent RCL connection can be made over the network using internet sockets (see Section 4.4.3). The RCL can be tested using the RCLCO program, provided with the S2 on floppy disks in ANSI C source code format. RCLCO supports both serial and network operation and can serve as the basis for user application programs that control the S2. It includes the RCL interface library, a set of function calls which provide the programmer's interface to the RCL. As shipped RCLCO runs either on IBM PC or compatible computers (serial port version) or under Unix (network version). Read the file called "1README" on the first floppy disk for further instructions.

4.4 Ethernet Usage

The optional S2 Ethernet board allows access to the console and RCL ports over the network using rlogin/telnet for the console and internet TCP sockets for the RCL. The Ethernet board can also be used to upgrade S2 software or restore the 3 MB SRAM from data files on a remote host, but for this the host computer must have the necessary files available and be configured to recognize the S2 target system. More information on network software reloads is available from CRESTech/SGL upon request. Note that when adding an Ethernet board to an S2 system you must remove the VME IACK (interrupt acknowledge) backplane jumper for that slot (VME slot 2). When removing the Ethernet board the IACK jumper must be put back in place.

4.4.1 Configuring Network Parameters

Before using the S2 on a network you must set boot parameters that define network-related information. These parameters are stored in a small (2K) battery backed-up Static RAM located on the RCC MK48T02 SRAM/clock chip (not the 3 MB SRAM board), and are retained while the power is off. You must obtain an internet address and name for each S2 system connected to the network. To do this contact your local network administrator. Then set the S2 boot parameters as follows: Enter the boot monitor by resetting the S2 and hitting the space bar before the countdown completes (to reset turn the power off and on, or press the small red "reset" button on the RCC board). Then type "c" <return>, and a series of prompts will appear:

boot device: Enter the type of Ethernet interface being used. You must always enter "enp" here.

processor number: Always enter 0 for this field.

host name: Enter the internet name of your remote host computer system, e.g. sgl.crestech.ca. The remote host may be used for transferring files to and

from the S2, software upgrades, or for VxWorks network boots. This field is not required if you will only be using the Ethernet for console rlogin/telnet or RCL socket connections, but *must not be left blank*. Enter x if you do not wish to supply an actual host name. For more information on configuring a remote host for special operations like software upgrades and file transfers see: *ftp://ftp.sgl.crestech.ca/pub/s2/doc/appnotes/HostSetup* .

file name: This is only required if you plan to perform software upgrades over the Ethernet. If not, enter x here since this field also *must not be left blank*. Users planning to perform network software upgrades should enter the absolute pathname of the VxWorks boot file on their host computer system, e.g. /local/s2/vxWorks. Even if this file does not actually exist, entering its name establishes a root directory for S2-related files and data. For the above example, the files used for network software upgrades would be placed in the directory /local/s2/ros/data/.

inet on ethernet (e): Enter the internet address assigned to this S2 system, e.g. 132.251.42.94. This field is required. It is also possible to include a subnet mask, which you may need to do if the default mask (ffff0000) doesn't work. Specify the subnet mask after the address separated by a colon, e.g. 132.251.42.94:ffffff00. Note that the internet address and name are a property of the ethernet board, *not* the RCC or the S2 system itself, so if you swap ethernet boards you must enter a new address here.

inet on backplane (b): Always enter "." to clear this field. It is not used, and should always be blank.

host inet (h): If you have entered the name of a remote host computer at the "host name" prompt above, enter its internet address here, e.g. 132.251.42.1. Otherwise enter "." to clear this field.

gateway inet (g): Normally the S2 system is accessible only on the local subnet where it is connected. To allow network access from distant network sites separated by one or more gateways, enter the internet address of the local gateway connecting your network to the outside world. Only one gateway can be chosen, and this becomes the default route for the S2, e.g. 130.155.192.2. The correct netmask must be specified at the "inet on ethernet" prompt for this to work. To allow access only from the local network, enter "." to clear this field.

If you wish to use more than one gateway, or you do not wish to establish a default route for security reasons but still want to enable outside access from certain places, you must leave this parameter blank (".") and manually add entries to the routing tables by typing routeAdd "DESTINATION","GATEWAY" in the VxWorks shell, where DESTINATION is the internet address of the destination host or network, and GATEWAY is the internet address of the gateway. Note: Execution of the routeAdd commands can be automated to occur each time the system boots up by adding them to the startup.cmd file using the edit startup.cmd console command (see Page 4-36).

user (u): This is only required if you plan to perform software upgrades over the network, or to transfer log files across the network via RSH (or FTP) using log display tofile, or to boot VxWorks over the network. Otherwise enter "." to clear this field. Users who wish to do so should enter the user name with which the S2 will access the remote host computer. The named user should have read access to the VxWorks boot file specified at the "file name" prompt above, and that user's account should allow access to the S2 system by placing appropriate entries in the **.rhosts** or **/etc/hosts.equiv** files.

ftp password (pw):   Enter "." to clear this field. It should normally be blank. Advanced network users who wish to use FTP instead of RSH for remote file copy operations (e.g. log and batch files) may enter the password corresponding to the userid chosen above.

flags (f):   Always enter 0 for this field.

target name (tn):   Enter the internet name assigned to the S2 system. This field is optional. If you do not wish to set it enter "." to clear this field.

startup script (s): Enter "." to clear this field. It is only relevant when booting VxWorks over the network.

other (o):   This field is not used. Enter "." to clear it.

To exit the boot monitor and continue the boot type `go`, or simply reset the S2. All of the above information can also be entered from the VxWorks shell by typing `bootChange`. This is especially useful if you are already logged on using rlogin or telnet and therefore cannot access the boot monitor. After changing any parameters type `reboot` to reboot the system (your connection will be broken and you will have to log in again).

4.4.2   Console access via rlogin/telnet

To access the console display over the network, use either **rlogin** or **telnet** depending on which is available or which one you prefer. Give the internet name or address assigned to the target S2 system, e.g.
```
telnet  132.251.42.94
```
If console password protection is enabled you will have to enter a user-ID and password (see below for more information). Otherwise you will be connected immediately. While the network console connection is active the S2 will no longer output information to or accept commands from the console serial port. Only one simultaneous network console connection is allowed — additional rlogin or telnet requests will be rejected. Type `logout` to close the rlogin/telnet connection and return control to the serial port. You can also force control to revert to the serial port by typing cntl-C cntl-C cntl-] in quick succession on the serial console's keyboard (do not use this control sequence at any other time). After connecting using rlogin or telnet you may need to change the terminal type if it is not correct. To do so use the `term` console command, e.g. `term xterm`. To change the default terminal type used by the S2 at bootup include the word `default`, or edit the defaults file and set the **term** parameter as appropriate.

Console rlogin/telnet access may be protected by a user-ID and password prompt, although these are not enabled when the S2 is shipped. To enable password protection, set the desired user-ID and password using the **loginuserid** and **loginpasswd** parameters in the defaults file. You must first encrypt the password using the `encrypt` console command, then use the encrypted version of the password in the defaults file (see Page 4-36). Additional userid/password pairs can be defined by adding more lines in the same format (each new userid line must be followed by a passwd line). If desired, network console connections can be disabled completely by setting the **consolelock** defaults parameter to 1. This prevents unwanted rlogin/telnet connections from interrupting serial console operation.

Note that in rare cases you may find yourself in the VxWorks shell after connecting with rlogin or telnet. This usually means that the previous user logged out while in the shell, or that an error occurred during bootup which prevented ROS from starting. If an error occurred you should see a brief error message which gets reprinted every time you press return (in fact it is part of the prompt to be sure you don't miss it). The best thing to do in this case is to connect the serial console and reboot the system, watching for any error messages.

4.4.3   RCL access via Internet sockets

User software can access the RCL over the network using internet TCP sockets and specifying port number 1025. The underlying RCL packet format is exactly the same as when using serial connections. The RCLCO program and RCL interface library software supplied with the S2 support network operation when compiled under Unix (preprocessor macro UNIX defined). You can use these programs as an example of how to write your own networked RCL control code.

Before RCL commands can be sent to a particular S2 system a network connection must be opened by calling **rcl_open()** (RCLCO `open` command), specifying the desired internet name or address. The open operation returns a *reference address* which is an integer used to subsequently refer to that connection. The reference address should be substituted for the RCL device address when issuing RCL commands. In network operation the RCL device address is not used since internet addresses are sufficient to identify different S2 systems — in fact network RCL packets contain the broadcast address to be sure to be accepted by the target S2 system. More than one network connection can be opened simultaneously to control multiple S2 systems. While a connection is open the target S2 system will not respond to RCL commands sent over the regular RCL serial link. If desired, the **rcllock** defaults-file parameter can be used to prevent unwanted network connections from interrupting serial RCL operation. When you are finished, use **rcl_close()** (RCLCO `close` command) to close each active network connection. For more information refer to the RCLCO and RCL Interface Library source code and 1README files.

4.5 Status Codes and Interpretations

The S2 recorder has approximately 80 different status codes that describe its current state of well-being. These are ranked and divided into 3 severity categories: *informational*, *error*, and *fatal error*. Each status code may additionally be designated *clear-on-read*, which means it is a transient indication that persists only long enough to be read. On the RCL, clear-on-read conditions are cleared following each status read request, but on the console there is no way to tell when the operator has read the message so status must be cleared manually using `status reset`. Status information is gathered continuously and is not the result of any specific diagnostic test or procedure (it is sometimes called "online status" to emphasize this fact).

Each status code is associated with an explanatory text message, and there is both a "short" and a "long" version of the message. The short messages are limited to 34 characters and are typically used on the console. The long messages are limited to 399 characters (but are usually much shorter) and are typically used with the RCL. In the list of status codes below only the short (console) messages are shown; for a list of the long (RCL) messages see the table starting on page A-40. Many status messages have occurrence-specific fields where information such as transport or channel numbers may appear. Places where such information gets inserted are marked by "xxx" or "nnn".

Status can be summarized by checking if any error or fatal error conditions exist. On the console display, the presence of any highlighted messages in the Status window indicates an error, and the word FAIL indicates a fatal error. Similar summaries are provided for the RCL. Overall system status can be considered perfect if there are no errors of either kind. Non-fatal errors should be noted by the user and probably investigated, but usually pose no threat to system operation. If any fatal errors appear, there is a good chance that the system is not operating correctly, and immediate corrective action should be taken. Informational messages (non-highlighted) are of course always benign. The green front-panel LED on the VME card cage remains lit as long as no fatal status conditions exist (must be set to OK mode, i.e. not blinking). Note that the S2 will not automatically abort any operation, even in the presence of fatal errors, unless there is no possible way to continue or there is imminent danger of hardware damage.

The S2 status codes are listed on the following pages, each with an explanation of what it means and what corrective action might be necessary. They are ranked by code number in rough order of severity as they would appear in the console status window. To help locate a status code quickly, use the `status display` command to determine its code number (the numbers are not shown in the status window due to space limitations).

0)	Mnemonic:	STAT_OK
	Type:	Informational
	Console Msg:	(none)
	Interpretation:	Indicates system status OK, no other status conditions exist.

1)	Mnemonic:	STAT_DIAGIP
	Type:	Informational
	Console Msg:	**"Diagnostic test xxx in progress"**
	Interpretation:	A system diagnostic test, such as the power-on self-test (self1), is currently running. Upon completion, either STAT_DIAGDONE or STAT_DIAGFAIL will appear.

2)	Mnemonic:	STAT_DIAGDONE
	Type:	Informational (clear-on-read)
	Console Msg:	**"Diagnostic test completed OK"**
	Interpretation:	A system diagnostic test, such as the power-on self-test, has completed successfully. Requires verbose level 3 to appear on the console.

3)   Mnemonic: STAT_DIAGFAIL

          Type: Fatal error (clear-on-read)

    Console Msg: **"Diagnostic test failed: code nnn"**

    Interpretation: A system diagnostic test, such as the power-on self-test, has detected a problem. Due to space limitations, the console status window shows only a numeric diagnostic code. This can be expanded using `status display` or `status display fullscreen`. It is also possible to perform a direct translation of the status code using `status decode`, but occurrence-specific message fields such as channel and transport numbers will be filled with "xxx" instead of real information. To confirm the results of the power-on self test, the test can be re-run using `diag self1 verbose`.

4)   Mnemonic: STAT_VMEPOWFAIL

          Type: Fatal error

    Console Msg: **"VME xxx power supply bad"**

    Interpretation: One or more VME power supplies are outside the acceptable range (±5% of nominal), as measured by an A/D converter on the Formatter board. The exact measured voltages can be displayed by typing `powermon`. See also STAT_TRANBADVOLT.

5)   Mnemonic: STAT_NOFORM

          Type: Fatal error

    Console Msg: **"Formatter board marked not present"**

    Interpretation: The Formatter board cannot be accessed by the Recorder Control Computer (VME read attempt gives bus error). It is either not present or has failed.

6)   Mnemonic: STAT_NODRD

          Type: Fatal error

    Console Msg: **"DRD nnn marked not present"**

    Interpretation: One or more required Data-Recovery/Deformatter boards cannot be accessed by the Recorder Control Computer (VME read attempt gives bus error). They are either not present or have failed. The **diagnumtran** parameter in the system defaults file determines how many DRD boards are required to be present, usually 8 except in scaled-down S2 systems.

7)   Mnemonic: STAT_NOUIC

          Type: Fatal error

    Console Msg: **"UI board marked not present"**

    Interpretation: The User Interface board cannot be accessed by the Recorder Control Computer (VME read attempt gives bus error). It is either not present or has failed.

8)   Mnemonic: STAT_FORMFAIL

          Type: Fatal error

    Console Msg: **"Formatter failure (reason follows)"**

    Interpretation: The Formatter board is operating incorrectly, probably due to a hardware failure. This is a general OK/FAIL indicator for the Formatter board, and is a summary of other status conditions which appear later in the display.

9)   Mnemonic: STAT_DRDFAIL

          Type: Fatal error

    Console Msg: **"DRD nnn failure (reason follows)"**

    Interpretation: One or more Data-Recovery/Deformatter boards are operating incorrectly, probably due to a hardware failure. This is a general OK/FAIL indicator for the DRD boards,

and is a summary of other status conditions which appear later in the display. Each DRD board also has a green OK LED on its front panel which will go out when a failure is detected.

11)　Mnemonic:　STAT_UICFAIL

　　　　　Type:　Fatal error

　Console Msg:　**"UI board failure (reason follows)"**

　Interpretation:　The User Interface board is operating incorrectly, probably due to a hardware failure. This is a general OK/FAIL indicator for the UI board, and is a summary of other status conditions which appear later in the display.

12)　Mnemonic:　STAT_TRANFAIL

　　　　　Type:　Fatal error

　Console Msg:　**"Tran nnn failure (reason follows)"**

　Interpretation:　One or more transports are operating incorrectly, probably due to a cable fault or hardware failure. This is a general OK/FAIL indicator for the transports, and is a summary of other status conditions which appear later in the display. Each failed transport listed here will also be marked with a highlighted 'F' in the Transport window.

13)　Mnemonic:　STAT_ROSSWFAIL

　　　　　Type:　Fatal error

　Console Msg:　**"ROS s/w failure (reason follows)"**

　Interpretation:　The Recorder Operating System software is operating incorrectly. This is a general OK/FAIL indicator for the S2 software, and is a summary of other status conditions which appear later in the display.

15)　Mnemonic:　STAT_SRAMCHKSUM

　　　　　Type:　Fatal error

　Console Msg:　**"SRAM board bad checksum (xxx)"**

　Interpretation:　One or more partitions of the 3 MB SRAM board have become corrupted and are failing a checksum comparison test. Try rebooting the S2 system to see the results of similar checksum tests performed during bootup. The corrupted partition, or the entire 3 MB SRAM, probably needs to be reloaded. See Section 6.2 for information on how to re-load a corrupted SRAM.

16)　Mnemonic:　STAT_NOSRSYR

　　　　　Type:　Fatal error

　Console Msg:　**"64 Hz frame sync is absent!"**

　Interpretation:　The Formatter 64 Hz frame sync signal (SRSYR), which normally drives most of the S2 internal hardware and software, is absent. This is a severe internal timing fault, and causes the S2 system clock to stop. It can be caused by a failure of the timing or interrupt generation circuitry on the Formatter board, or a failure of the System Clock PLL on the UI board. This does *not* indicate the absence or presence of user-supplied clock signals — see STAT_NOREFCLK and STAT_NOREF1HZ.

17)　Mnemonic:　STAT_NOFG1

　　　　　Type:　Fatal error

　Console Msg:　**"System 1 Hz tick is absent!"**

　Interpretation:　The Formatter 1 Hz reference tick (FG1) is absent. This is a severe internal timing fault. It can be caused by a failure of the timing circuitry on the Formatter board.

This does *not* indicate the absence or presence of user-supplied clock signals — see STAT_NOREF1HZ.

18)   Mnemonic: STAT_TASKFAIL

        Type: Fatal error

Console Msg: **"Task not running: xxx"**

Interpretation: One or more required ROS software tasks are not running. A task may die if it encounters a severe error condition or detects an internal inconsistency, resulting in this message. The S2 system should be rebooted by cycling the power or pressing the RCC "reset" button.

19)   Mnemonic: STAT_TRANSWFAIL

        Type: Fatal error

Console Msg: **"Tran nnn s/w fail: xxx"**

Interpretation: One or more transport internal processors have detected a software-related failure. The possible types of failure are as follows:

    EEPROM chksum bad: The transport internal EEPROM has been corrupted. This contains the TCP program code and TCB Xilinx configuration, plus calibration parameters and service history information. The transport must be completely re-programmed using `transport N program`.

    stack overflow: The transport software stack has grown past a predefined limit. This indicates a software bug and should be reported to CRESTech. The transport may continue to operate normally since there is a built-in safety margin before the stack actually overflows.

    speed set fail: The transport could not set its speed or sense the correct speed. This indicates a probable hardware failure in the speed sensing circuitry.

20)   Mnemonic: STAT_TRANTPMOTFAIL

        Type: Fatal error (clear-on-read)

Console Msg: **"Tran nnn tape motion fail"**

Interpretation: A transport has detected that the tape motion state is not as required via a sensor on the takeup reel. The transport automatically issues a stop command to itself (without affecting other transports) to return to a consistent state. This situation can arise when a transport fails to respond to a tape motion command ("missed remote control code"), or if the tape jams or there is some mechanical failure. There can be a delay of up to 30 seconds before the transport recognizes a tape motion failure. For example, suppose a `rewind` command is issued for all 8 transports but one of them, transport 2, fails to respond due to a missed remote control code. Initially the console display for transport 2 will indicate that it is rewinding although it is in fact stopped, having never physically started rewinding. After approximately 30 seconds a tape motion fail status message will appear and transport 2 will be instructed to stop, after which the display will correctly show that it is stopped. Users should react to the STAT_TRANTPMOTFAIL condition by stopping the current operation and restarting it from the beginning. Repeated occurrences of STAT_TRANTPMOTFAIL indicate a tape jam or hardware failure within the transport. In the rare event that a transport repeatedly stops on erroneous tape motion fail conditions, yet appears to be working properly, you can disable the tape motion fail behaviour using the command `transport N options takeupsense disable`. This command may be put in the autoexec batch file to remain effective even after rebooting.

22) Mnemonic: STAT_TRANBADTEMP

      Type: Fatal error

    Console Msg: **"Tran nnn temp out of range!"**

Interpretation: One or more transport internal temperature sensors show an unsafe operating temperature (greater than 60 or less than 5 degrees C). The transport automatically turns itself off and cannot be turned on until the temperature returns to a safe level. Check the TAM rack fans and air filters for proper operation. To display the temperature measurements, type `transport display temp` or `transport status volt`. If a transport's temperature reads out of range but it is known not to actually exceed the temperature limits, the temperature sensor may need to be recalibrated (see the `caltemp` option of the `transport` command on Page 4-53).

23) Mnemonic: STAT_TRANWARNTEMP

      Type: Error

    Console Msg: **"Tran nnn temp almost out of range"**

Interpretation: One or more transport internal temperature sensors show a temperature greater than 55 or less than 8 degrees C, which is close to the unsafe operating limits. The transport will continue to operate normally unless the temperature rises or falls past the limits defined for safe operation in STAT_TRANBADTEMP, above. Check the TAM rack fans and air filters for proper operation. To display the actual temperature measurements, type `transport display temp` or `transport status volt`.

24) Mnemonic: STAT_TRANDEAD

      Type: Fatal error

    Console Msg: **"Req'd transpts nnn are off/dead"**

Interpretation: One or more currently selected transports are off or dead. Dead transports show up as "--" in the transport display. This means that the Recorder Control Computer is unable to communicate with the transport, possibly due to breaks in the TCL serial cable or failure of the Transport Control Processor.

        Transports which are off show up as "off" in the Transport window. Some reasons for a transport to turn itself off include excessive condensation, temperature out of range (see STAT_TRANBADTEMP), and internal hardware failures. You can try turning it on again using `transport N power on` or the front-panel power button (labeled "operate" on AG2530s).

25) Mnemonic: STAT_TRANBADVOLT

      Type: Fatal error

    Console Msg: **"Tran nnn +5V/-5V supply bad (C4)"**

Interpretation: One or more transports are not receiving proper ±5 V supplies over the C4 cable. The acceptable voltage ranges are +4.1 to +5.5 V for the +5 V supply and -4.1 to -5.5 V for the -5 V supply. Check the C4 cable connections and the interface circuits on either side (transport, DRD board). In particular, the transport ±5 V supplies pass through chokes on the DRD board which tend to act as fuses and can blow. To display the actual supply voltage measurements, type `transport N status volt`.

26) Mnemonic: STAT_TRANBADRECSIG

      Type: Fatal error

    Console Msg: **"Tran nnn record clk/data bad (C3)"**

Interpretation: One or more transports are not receiving proper clock and data signals from the C3 cable. Check the C3 cable connections and the interface circuits on either side

(transport, Formatter board). To determine which of the clock/data signals are missing, type `transport status recsig`.

27)   Mnemonic: STAT_ILMODE
            Type: Fatal error
   Console Msg: **"Recorder mode is illegal"**
   Interpretation: The recorder mode is not properly set. Caused by a failure in setting the mode, and indicates a probable hardware fault on the UI board (DDDV Xilinx chip download chain).

28)   Mnemonic: STAT_NOREFCLK
            Type: Fatal error
   Console Msg: **"User reference S-Clk/C-Clk/X-tal is absent"**
   Interpretation: The System Clock PLL is currently set to lock to an external high-rate clock (**refclk** mode), but no clock is being supplied over the C1 cable (SCLK_REC) or C2 cable (CCLK). If the System Clock PLL mode is set to **xtal**, this message indicates a failure of the UI board on-board 32 MHz crystal oscillator (X-tal).

29)   Mnemonic: STAT_NOREF1HZ
            Type: Fatal error
   Console Msg: **"User ref S-1/C-1 Hz sync is absent"**
   Interpretation: The System Clock PLL is currently set to lock to an external high-rate clock (**refclk**) and/or 1 Hz tick (**1hz**), but no 1 Hz tick is being supplied over the C1 cable (S1HZ_REC) or C2 cable (C1HZ).

32)   Mnemonic: STAT_BADSTNDELAY
            Type: Error
   Console Msg: **"Stn delay measurement != setting"**
   Interpretation: The station delay measurement does not match the setting from the last `delay` command (System Clock PLL **refclk** mode only). This is most likely to be caused by a glitch in the user's high-rate reference clock. For example, if the clock rate is 32 MHz, a glitch might cause more than (or fewer than) 32,000,000 clock cycles to be registered over a one-second period, resulting in a shift of the S2's internal 1 Hz tick with respect to the external 1 Hz reference. A station delay mismatch can also be caused by an actual shift in the external or internal 1 Hz timing, or by instability of the System Clock Phase-Lock Loop. Delay mismatches may persist once they have arisen because the S2 internal 1 Hz tick is not automatically re-synchronized unless **scpllrefclkfix** is set to 1 in the defaults file (see below). Delay corrections can be made manually by resetting the appropriate delay value (usually 0) with the `delay` command. If the condition persists, check the input clock frequency selection (`userio clkin`). Ensure that the external clock and 1 Hz sync signals conform to S2 interface specifications given in Appendix B. If after this the delay still cannot be set, there is probably a hardware fault in the UI timing circuitry.

The S2 can be instructed to automatically "fix" station delay mismatches by setting the defaults-file parameter **scpllrefclkfix** to 1. In this case any station delay mismatch that arises and remains stable for 2 seconds will be corrected by re-asserting the last entered station delay value. See the station delay measurement description on page 4-18 for more information.

33) Mnemonic: STAT_UICVERFAIL

        Type: Error (clear-on-read)

  Console Msg: **"UI verifier errors: nnn"**

Interpretation: Indicates one or more UI board verifier errors have occurred since the last status read (RCL)/reset (console). In C1 cable test mode, this shows a C1 cable connection fault. During record, this indicates a failure of the distributor or de-distributor on the UI board. Note: UI verify errors are not considered fatal because continuous verifier errors activate STAT_UICFAIL, which is fatal.

34) Mnemonic: STAT_SYNCERR1HZ

        Type: Error (clear-on-read)

  Console Msg: **"1 Hz sync errors: nnn"**

Interpretation: Indicates one or more internal 1 Hz sync errors have occurred since the last status read (RCL)/reset (console). A 1 Hz sync error is defined as two consecutive system internal FG-1 Hz ticks that are not exactly 64 frames apart. Excessive 1 Hz sync errors can indicate a failure of the Formatter or User Interface boards.

35) Mnemonic: STAT_NESTFRINT

        Type: Error (clear-on-read)

  Console Msg: **"Nested frame interrupts: nnn"**

Interpretation: Indicates one or more nested frame-sync interrupts have occurred since the last status read (RCL)/reset (console). This can be caused by frame syncs (SRSYR) occurring too close together, such as might happen during an internal timing glitch.

36) Mnemonic: STAT_LATEFRINT

        Type: Error (clear-on-read)

  Console Msg: **"Late frame interrupts: nnn"**

Interpretation: Indicates one or more frame sync interrupts have been late or missed since the last status read (RCL)/reset (console). A late frame interrupt is defined as a period longer than $\frac{1}{60}$th of a second without the occurrence of a frame sync interrupt. This can be caused by a system internal timing glitch or absent SRSYR signal (see STAT_NOSRSYR).

37) Mnemonic: STAT_SCPLLBADMODE

        Type: Error

  Console Msg: **"Bad SC PLL mode (should be xxx)"**

Interpretation: Appears during record if the System Clock PLL mode is not **refclk** and external input data is being recorded. The S2 must lock to an external high-rate reference clock for correct record operation, therefore the SC PLL mode must be **refclk**. The SC PLL mode can be set using the `scpll mode` command or by changing the **scpllmode** parameter in the defaults file (`edit defaults`).

38) Mnemonic: STAT_SCPLLUNLOCK

        Type: Fatal error

  Console Msg: **"System clk PLL is unlocked"**

Interpretation: Indicates that the System Clock PLL is not locked. This message is triggered by continuous SC PLL sync errors (see STAT_SCPLLSLIP). It is normal when lock is first being acquired in SC PLL **1hz** and **errmes** modes. If the SC PLL mode is **refclk**, check the input clock frequency selection (`userio clkin`) and ensure that the external clock signal conforms to S2 interface specifications given in Appendix B. If problems persist, there may be a fault in the System Clock PLL hardware on the UI board. If the SC PLL mode is **xtal**, there may be a problem with the UI on-board 32 MHz crystal oscillator.

39) Mnemonic: STAT_SCPLLSLIP

      Type: Error (clear-on-read)

Console Msg: **"System clk PLL sync errors: nnn"**

Interpretation: Indicates that the System Clock PLL is not locked, or is having difficulty locking to its reference. Occurrence of this message is normal when lock is first being acquired, particularly in SC PLL **1hz** and **errmes** modes. Continuous sync errors mean the SC PLL is unlocked, and will activate the STAT_SCPLLUNLOCK status message. If the SC PLL mode is **refclk**, ensure that the external clock signal conforms to S2 interface specifications given in Appendix B. If problems persist, there may be a fault in the System Clock PLL hardware on the UI board. If the SC PLL mode is **xtal**, there may be a problem with the UI on-board 32 MHz crystal oscillator.

In SC PLL **1hz** mode the SC PLL is considered unlocked whenever the difference between the external and internal 1 Hz ticks is more than about about 60 microseconds from its intended value. This is equivalent to the difference between the current station delay setting and the delay measurement shown on the console display. A similar definition of lock applies in SC PLL **errmes** mode, except that the user-supplied error measurement is used. This lock definition has been chosen somewhat arbitrarily based on the requirements of space VLBI and the capabilities of the S2 hardware. The occurrence of STAT_SCPLLSLIP or STAT_SCPLLUNLOCK status errors does not in itself indicate a problem with S2 internal timing, only that the synchronization with the external reference has exceeded the 60 us threshold. However internal timing problems resulting in loss of data become more likely in this situation. See the description of the `delay` command for more information.

40) Mnemonic: STAT_DRDMISSIG

      Type: Error (clear-on-read)

Console Msg: **"DRD nnn missed/extra sigs: nnn/nnn"**

Interpretation: Indicates one or more missed or extra signatures were detected by the indicated DRD board(s) since the last status read (RCL)/reset (console). The Recorder Control Computer and the DRD 8051 control processors synchronize by exchanging signature codes, and any missing or extra signatures mean a problem with that synchronization. Isolated missed sigs can be caused by timing glitches, but continuous missed sigs indicate a probable failure of the DRD control processor. If more than one DRD board is listed, the number of missed/extra sigs shown is the maximum of all the affected boards.

41) Mnemonic: STAT_FORMDFIFOTERR

      Type: Error (clear-on-read)

Console Msg: **"Form d-FIFO timing errs: nnnnnnnn"**

Interpretation: Indicates internal timing errors in one or more S2 Formatter channels, specifically the data FIFOs. Isolated Formatter data-FIFO errors can be caused by glitches in system timing, such as System Clock PLL sync errors (STAT_SCPLLSLIP) — check your external timing reference. Continuous data-FIFO errors probably mean a hardware fault on the Formatter board. The error condition is sampled only once per second so there can be at most one data-FIFO error reported per second, but in actual fact there may have been more. The display shows the number of errors in one of two formats: if there have been fewer than 10 errors in all 8 Formatter channels, a string of 8 digits is shown where each gives the number of errors in the corresponding Formatter channel (0–7). If any channel has had 10 or more errors, only the channel with the most errors is shown. The impact of a Formatter data-FIFO error event during record is that at least one frame of data will be lost in all channels, even if only one channel registered an error (there are 64 frames

per second). During playback there should be no adverse effect since the formatter is not used, assuming no other timing-related problems occurred.

42)   Mnemonic: STAT_MK3PLLSLIP

         Type: Error (clear-on-read)

Console Msg: **"Mk3 PLL sync errors: nnn"**

Interpretation: During Mark III/IV-compatible operation, indicates the Mark III PLL on the UI board is having difficulty locking. Isolated Mark III PLL errors can be caused by changes in system timing, such as when the setup is changed between **play** and **bypass/record**, or when the recorder mode is changed. They can also appear when switching the Mark III formatter on or off. If system timing is otherwise stable, this message indicates a probable failure of the UI Mark III timing circuitry.

43)   Mnemonic: STAT_MK3MISSYNC

         Type: Error (clear-on-read)

Console Msg: **"Mk3 sync detect misses: nnn"**

Interpretation: During Mark III/IV-compatible operation, indicates that the Mark III formatted data contains missing or excess syncs, as determined by the Mark III sync detector. The sync detector is a diagnostic circuit intended to help verify correct Mark III formatter operation by checking for the sync word pattern. Spurious sync misses can be caused by changes in system timing, such as when the setup is changed between **play** and **bypass/record**, or when the recorder mode is changed. They can also appear when switching the Mark III formatter on or off. If system timing is otherwise stable, this message may indicate a failure of the Mark III formatter on the UI board.

44)   Mnemonic: STAT_MK3DFIFOTERR

         Type: Error (clear-on-read)

Console Msg: **"Mk3 d-FIFO timing errs: nnn"**

Interpretation: Indicates internal timing errors in one or more Mark III formatter channels, specifically the data FIFOs. Spurious Mark III data-FIFO errors can be caused by changes in system timing, such as when the setup is changed between **play** and **bypass/record**, or when the recorder mode is changed. They can also appear when switching the Mark III formatter on or off. Continuous Mark III data-FIFO errors probably mean a hardware fault on the UI board.

45)   Mnemonic: STAT_TRANSYNCERR32

         Type: Error (clear-on-read)

Console Msg: **"Tran nnn 32 Hz sync errs (nnn)"**

Interpretation: Indicates one or more transports have detected 32 Hz sync errors since the last status read (RCL)/reset (console). A sync error is declared whenever the 32 Hz sync signal from the Formatter appears more then 5 microseconds from its expected location, or is missing. The 32 Hz sync is carried on the C3 cable multiplexed with the tape clock signal. The number in brackets shows the highest number of sync errors of all the affected transports. Isolated 32 Hz sync errors can be caused by glitches in system timing such as System Clock PLL sync errors (STAT_SCPLLSLIP), and are usually not serious as long as no other status errors exist. Continuous 32 Hz sync errors probably mean a C3/C4 cable connection fault or a hardware fault in the transport or the Formatter.

46)   Mnemonic: STAT_REFCLKOUTLOW

        Type: Fatal error

Console Msg: **"C2 output ref clock rate too low"**

Interpretation: The C2 output reference clock rate is set to a value lower than the current user
                channel data rate. Set the correct clock rate using the `userio clkout` console
                command.

47)   Mnemonic: STAT_ERRRCL

        Type: Fatal error (clear-on-read)

Console Msg: **"RCL error nnn: xxxxxx"**

Interpretation: The RCL task encountered an internal software error. This should not normally
                appear. If the complete error message does not fit in the status window, `error
                decode` can be used to translate the numeric error code to the full text message.

48)   Mnemonic: STAT_ERRRPMON

        Type: Fatal error (clear-on-read)

Console Msg: **"RpMon error nnn: xxxxxx"**

Interpretation: The record/playback monitor task encountered an unexpected software error. This
                can happen if conditions are not correct for automatic record or playback, in which
                case this message will be accompanied by other more descriptive status messages.
                If the complete error message does not fit in the status window, `error decode`
                can be used to translate the numeric error code to the full text message.

49)   Mnemonic: STAT_ERRMK3SW

        Type: Fatal error (clear-on-read)

Console Msg: **"Mk3 form s/w error nnn: xxxxxx"**

Interpretation: The Mark III/IV related software encountered an internal error. This should not
                normally happen. If the complete error message does not fit in the status window,
                `error decode` can be used to translate the numeric error code to the full text
                message.

51)   Mnemonic: STAT_RPMONBOTEOT

        Type: Fatal error (clear-on-read)

Console Msg: **"Hit BOT/EOT/EOD during auto play/record"**

Interpretation: One of the active transports encountered the beginning of tape (BOT) or end of
                tape (EOT) during automatic record or playback. The S2 has automatically stopped
                the remaining transports. This message will also appear when a transport encounters
                the end of recorded data (EOD) during playback tape alignment. EOD is defined
                as approximately 30 seconds or more of blank tape encountered during positioning
                or alignment (slewing) in the forward direction.

52)   Mnemonic: STAT_RPMONILSTATE

        Type: Fatal error (clear-on-read)

Console Msg: **"Unexp tran state chg during play/record"**

Interpretation: The current overall tape motion state unexpectedly changed to something other
                than play/record during automatic playback/record. This can happen if the transport
                front-panel buttons are pressed, or if certain commands such as `pause` or `cue`
                are used — these commands should be avoided during automatic playback and
                record. This message will also appear if an attempt is made to record on tapes
                with the write-protect tabs broken, since write-protected tapes will be automatically
                ejected by the transports (model AG2510) or the transports will beep 7 times and
                refuse to go into record (models AG2520 and AG2530).

53)　Mnemonic: STAT_TRANLKFAIL

　　　　　Type: Fatal error

　Console Msg: **"Tran xxx servo failed to lock"**

Interpretation: One or more transport head-drum servos remain unlocked after 13 seconds in automatic record or playback. This usually indicates a transport failure or marginal transport PLL performance, or problems with the 32 Hz servo sync reference signal on the C3 cable. This status code will not appear during manual record/play.

54)　Mnemonic: STAT_TRANLKLOSS

　　　　　Type: Error (clear-on-read)

　Console Msg: **"Tran nnn servo lock loss (glitch)"**

Interpretation: One or more transport head-drum servos momentarily became unlocked in automatic record or playback, and have (probably) regained lock. If the servo is unable to regain lock after 13 seconds, STAT_TRANLKFAIL will appear. When more than one transport loses lock around the same time, some of them may not be listed in this status message. STAT_TRANLKLOSS appears only during automatic record/play (i.e. not during manual record/play, but the event will always be recorded in the ROS internal log). Servo lock glitches can be caused by damaged tapes or system timing glitches such as STAT_TRANSYNCERR32, or by irregularities in the movement of tape past the heads. For example, in the erase-gap between two different recordings the transport may momentarily switch speeds, which can cause loss of lock. Transports with repeated unexplained loss of lock problems on different tapes should be serviced and checked for hardware faults.

55)　Mnemonic: STAT_TRANTRK1FAIL

　　　　　Type: Error (clear-on-read)

　Console Msg: **"Tran nnn stuck in phase1 trk cal"**

Interpretation: The phase-1 tracking algorithm performed by the transports is taking too long. This message indicates a possible hardware fault or software bug in the Transport Control Processor. It may be caused by a broken transport control track sensor which is erroneously reporting blank tape.

56)　Mnemonic: STAT_ILSETUP

　　　　　Type: Error

　Console Msg: **"Bad setup: xxxx"**

Interpretation: The current internal diagnostic signal path configuration is illegal or inconsistent. This normally appears only during manual operation, and indicates that something is wrong with the switch settings that define the setup. The specific problem is given in place of the x's in the message, e.g.

　　　"ch filt on, shld be off": The channel emulation filter is being used when it shouldn't be

　　　"ch filt off, shld be on": The channel emulation filter is not being used when it should be

　　　"diag transpt is dead": The current diagnostic transport is dead

　　　"diag transpt is off": The current diagnostic transport is off

　　　"diag DRD N is dead": The current diagnostic DRD is marked not present

57)　Mnemonic: STAT_INSTATE

　　　　　Type: Fatal error

　Console Msg: **"Tape motion state is inconsistent"**

Interpretation: The overall tape motion state cannot be consistently determined based on the states of currently selected transports. For example, some transports are recording and some are stopped. This can also arise if some or all transports are in a state that

is not legal for system operation, such as "off" or "dead". This status message is usually accompanied by other messages which clarify the problem.

58)   Mnemonic: STAT_BADTRANSEL
          Type: Fatal error
   Console Msg: **"Bad tran selection for mode/group"**
Interpretation: The set of selected transports (marked with '*') is not correct for the current mode and group settings. Normally the `mode` and `group` commands automatically select the active transports as required. This status message appears if a different selection is in effect during automatic record or playback. Stop the S2 and select the correct transports by (re)setting the correct mode or group, or by typing `transport autoselect`.

59)   Mnemonic: STAT_CHIDDUP
          Type: Fatal error
   Console Msg: **"Tran N & N tapes have same chan ID"**
Interpretation: During playback, (at least) two different tapes have the same channel ID, which means they were recorded on transports with the same address. The tapes probably come from different tape sets. Stop the S2, insert the correct tapes, and restart playback.

60)   Mnemonic: STAT_ALIGNFAIL
          Type: Fatal error
   Console Msg: **"Unable to align tapes: error nnn"**
Interpretation: A playback tape alignment operation has failed under circumstances for which alignment *will not* be automatically retried. This includes any user-initiated alignment operation as well as automatically initiated alignment operations that fail for certain reasons as outlined below. The reason is given by the error code (nnn) embedded in the message — use `error decode` to translate the numeric error code to a text message. Alignment must be explicitly re-initiated using the `align` command, or by stopping and re-starting playback. Causes of STAT_ALIGNFAIL include:

   ERR_ALCRSBOUND — Crossed a boundary from one recording into another. Tape alignment can only be performed within a single continuous recording.

   ERR_ALBADTIME — Requested alignment time is out of range. It is either more than 10 hours away from the current tape time, or in the case of automatic self-alignment the difference between some transport's tape time and the target time exceeds the value given for the **selfaligntime** defaults-file parameter, normally 30 minutes.

   ERR_BOTEOT — One or more transports ran into beginning/end of tape during alignment. All transports have been stopped.

   ERR_EOD — One or more transports encountered EOD, the end of recorded data, during alignment. All transports have been stopped.

61)   Mnemonic: STAT_ERRALIGN
          Type: Error (clear-on-read)
   Console Msg: **"Err aligning tapes: nnn (retrying)"**
Interpretation: A playback tape alignment operation has failed under circumstances for which alignment will be automatically retried. The reason is given by the error code (nnn) embedded in the message — use `error decode` to translate the numeric error code to a text message. The user may wish to take action if this condition persists for a long time. Some causes of STAT_ERRALIGN include:

   ERR_NOPBAUX — Couldn't decode time from one or more transports (possibly ran past end of recorded material).

   ERR_TRANLKFAIL — Transport head-drum servo failed to lock.

62)   Mnemonic: STAT_CHIDMISS
      Type: Fatal error
    Console Msg: **"Tape with channel ID N is missing"**
    Interpretation: During playback one or more of the required tapes is missing. This status condition is usually accompanied by STAT_BADTRANSEL or STAT_CHIDDUP. If one or more tapes in a set are ever lost, it is still possible to play back the remaining tapes but this status message will appear. You should manually de-select the unused transport before starting playback by typing `transport autoselect-N` where N is the address of the unused transport.

63)   Mnemonic: STAT_BADGROUP
      Type: Fatal error
    Console Msg: **"Tran N tape/data not part of curr group"**
    Interpretation: One or more of the group of currently playing transports contains a tape that is not part of the current group. For example, suppose the mode is 16x4-1 and we have chosen group 0 (transports 0–3). If during playback the tape in transport 3 is found to contain channel 7 data (i.e. channel ID 7), the message "Tran 3 tape not part of curr group" will appear, since that tape (channel ID 7) was actually recorded as part of group 1. The same group setting as used at record time must be used for playback. This status message can also appear with non-playback data, e.g. suppose the situation is as above but the S2 is stopped and **bypass** path is selected. If we enter `setup 3 transport 7`, we will get the message "Tran 7 data not part of curr group" since bypass data from transport 7 is part of group 1, not 0. During automatic record or playback, this message usually appears along with other status conditions such as bad transport selection (STAT_BADTRANSEL) or missing tapes (STAT_CHIDMISS).

64)   Mnemonic: STAT_CHIDMISMATCH
      Type: Error
    Console Msg: **"Tape/Decoded chan ID does not match setup"**
    Interpretation: The decoded channel ID does not match the diagnostic channel selected in the current setup. In other words, decoded data is being mapped onto the wrong internal S2 channel. Although each decoder (DRD) is capable of outputting data onto any internal data channel leading to the UI, the "right" channel is the one which matches the channel used at record time. For example, if transport 4 bypass data is being mapped to channel 5 (`setup 5 transport 4`), this status message will appear. However, if we are in playback and transport 4 contains the tape with channel 5 data, then mapping it onto internal channel 5 is correct. During automatic record or playback, this status message usually appears along with other status conditions such as bad transport selection (STAT_BADTRANSEL) or tape not part of current group (STAT_BADGROUP).

65)   Mnemonic: STAT_NOPBAUX
      Type: Error
    Console Msg: **"Can't decode aux data/time from tran N"**
    Interpretation: During playback, auxiliary data or time could not be decoded from the indicated transport after trying for 15 seconds. Although time is part of the auxiliary data, it is easier to decode since it is treated specially and has its own CRC. Only one transport is listed, but the condition may apply to others as well. The tape could be blank, or there may be a problem with the transport or the DRD. This message appears only during automatic playback.

66) Mnemonic: STAT_BADPBTRANSEL

        Type: Fatal error

    Console Msg: **"Can't set mode xxx, wrong # tran"**

    Interpretation: During automatic playback, the S2 is unable to switch to the mode that the tapes were recorded in because the wrong number of transports is currently active. This means that the current mode has a total data rate that is different from the mode used for recording. You must always select a mode with the same total data rate as used at record time before starting playback. Stop the S2, select an appropriate mode, and restart playback. The simplest way to choose a mode is to use 16xN-1 where N is the same number of transports as used at record time. The S2 will automatically switch to the correct mode after playback starts.

67) Mnemonic: STAT_INTAPEID

        Type: Error

    Console Msg: **"Tran N & N have different tape IDs"**

    Interpretation: During automatic playback, (at least) two tapes have different Tape ID strings. This could mean they come from different tape sets. If so, stop the S2, insert the correct tapes, and restart playback. This message may appear momentarily if the recorded Tape ID changes, but its presence will have no adverse effects on system operation. Nonetheless we recommend that the same Tape ID be used throughout the entire length of the tape.

68) Mnemonic: STAT_BADPBESTERR

        Type: Error

    Console Msg: **"Tran N estimated err rate >nnn"**

    Interpretation: During automatic playback this indicates that the estimated error rate for one or more transports is above a certain "acceptable" threshold. The threshold is currently 5.0×10^{-4} for speed LP and 3.0×10^{-3} for SLP. This message should just be treated as a warning, any more severe conditions will cause additional status messages to appear. To see the actual estimated error rates for each transport type `transport display esterr`.

69) Mnemonic: STAT_RECBADSYNC

        Type: Fatal error

    Console Msg: **"Tran nnn rec data has bad sync"**

    Interpretation: The diagnostic loopback data from one or more transports has missing or marginal sync during automatic record. This may mean that the data is not being recorded properly, and should be investigated immediately. Possible causes include Formatter failure, transport failure, DRD failure, or marginal C3/C4 cable connections.

70) Mnemonic: STAT_RECBADAUX

        Type: Fatal error

    Console Msg: **"Tran nnn rec data has bad aux"**

    Interpretation: The diagnostic loopback data from one or more transports has excessive errors in the decoded auxiliary data during automatic record. This may mean that data is not being recorded properly, and should be investigated immediately. Possible causes include Formatter failure, transport failure, DRD failure, or marginal C3/C4 cable connections.

71) Mnemonic: STAT_RECBADCHID

       Type: Fatal error

   Console Msg: **"Tran nnn rec data has wrong chid"**

  Interpretation: The diagnostic loopback data from one or more transports contains the wrong channel ID during automatic record. This indicates incorrect wiring of the C3 or C4 cables, and should have been caught by the power-on self-test.

72) Mnemonic: STAT_BADAGCLEV

       Type: Fatal error

   Console Msg: **"Tran nnn AGC levels out of range"**

  Interpretation: One or more transport AGC levels are outside the "acceptable" range during automatic record or play (the AGC level indicates record or playback signal strength). During record this probably means that the data is not being recorded properly, and could be due to a Formatter failure, transport failure, or C3 cable connection fault. During playback this may mean that data cannot be properly decoded, and is probably due to a failure in the indicated transport(s). The acceptable AGC ranges are from 100 to 254 for record and 1 to 254 for playback.

73) Mnemonic: STAT_MARGAGCLEV

       Type: Informational

   Console Msg: **"Tran nnn AGC levels are marginal"**

  Interpretation: One or more transport AGC levels are outside the "ideal" range during automatic record or play (the AGC level indicates record or playback signal strength). This will probably not affect operation, but should eventually be investigated. Check the indicated transports. If recording also check the Formatter board and C3 cable. The ideal AGC ranges are from 150 to 249 for record and 6 to 249 for playback.

74) Mnemonic: STAT_ERRLOG

       Type: Error (clear-on-read)

   Console Msg: **"Log file error nnn: xxxxxx"**

  Interpretation: An error was encountered maintaining the ROS internal log. To re-initialize the log you may manually clear it using `log clear`.

75) Mnemonic: STAT_ERRDRCONFIG

       Type: Error (clear-on-read)

  Console Msg: **"Error in DRD nnn DR config: nnn"**

  Interpretation: The data-recovery calibration software encountered an internal error. This should not normally happen, and can usually be ignored.

76) Mnemonic: STAT_ERRDRTRACK

       Type: Error (clear-on-read)

   Console Msg: **"Error in tran nnn tracking cal: nnn"**

  Interpretation: The data-recovery phase-2 tracking calibration software encountered an internal error. This is displayed only at verbose level 3 for both the console and the RCL, and can usually be ignored.

77) Mnemonic: STAT_DELAYFIX

       Type: Informational (clear-on-read)

   Console Msg: **"Station delay auto resets: nnn"**

  Interpretation: Gives a count of how many times the station delay has been automatically corrected after becoming incorrect (glitching) in SC PLL **refclk** mode. This correction can only happen if the **scpllrefclkfix** defaults-file parameter is set to 1. See also the explanation of the STAT_BADSTNDELAY status code.

81)   Mnemonic: STAT_NOALIGN

         Type: Error

Console Msg: **"Tapes playing but not aligned"**

Interpretation: This message appears during playback when the individual tapes are not aligned to each other. In particular, it always appears when playback starts and then goes away once the automatic self-alignment procedure has aligned the tapes. While this status condition exists playback data should be treated as invalid. Since there is no single coherent tape time no "overall" tape time can be defined (the RCL TIME_READ_PB command will return an error, and relative tape alignment is not allowed). If for some reason alignment is lost during playback, STAT_NOALIGN will reappear and the S2 will attempt to re-align to the current running aligned-tape time. Certain errors such as ERR_ALCRSBOUND ("Tape alignment crossed recording boundary") will result in the fatal status code STAT_ALIGNFAIL which means that no further attempts will be made to realign. Other less-severe errors produce STAT_ERRALIGN, which means that alignment encountered an error but is being re-attempted.

84)   Mnemonic: STAT_ALIGNIP

         Type: Informational

Console Msg: **"Tape alignment in progress"**

Interpretation: A playback tape alignment operation is in progress, initiated either automatically or by request from the user. Upon completion, either STAT_ALIGNDONE or STAT_ALIGNFAIL will appear, and STAT_ALIGNIP will disappear. During alignment, STAT_ERRALIGN may appear but in this case attempts to align will continue.

85)   Mnemonic: STAT_ALIGNDONE

         Type: Informational (clear-on-read)

Console Msg: **"Tape alignment completed OK"**

Interpretation: Playback tape alignment has completed successfully and the overall tape time is now coherent and valid. Requires verbose level 3 to appear on the console, verbose level 1 or greater on the RCL.

89)   Mnemonic: STAT_TAPESYNCERR

         Type: Error (clear-on-read)

Console Msg: **"Tape-time sync errors: nnnnnnnn"**

Interpretation: The time decoded from tape was not as expected. In each successive frame the time should advance by exactly one unit. This message indicates an unexpected timing shift of one or more frames by the transport, such as might be caused by a tape jam. Tape-time sync errors can also occur during transitions between tape motion states, and during pause, so normally they are counted only if the tapes are currently aligned. Another possible cause is a transition between two different recordings. Any significant tape-time sync error is always accompanied by other status conditions such as STAT_NOALIGN, so STAT_TAPESYNCERR is not shown on the console or the RCL unless the verbose level is 2 or higher (1 is the level for normal operation). The display shows the number of errors in one of two formats: if there have been fewer than 10 sync errors in all 8 transports, a string of 8 digits is shown where each gives the number of errors in the corresponding transport (0–7). If any transport has had 10 or more errors, only the one with the most errors is shown.

92)   Mnemonic: STAT_SCPLLDACRAIL

         Type: Informational

  Console Msg: **"System clk PLL DAC at/near rail"**

 Interpretation: The System Clock PLL software-controlled DAC value is within 5% of its maximum limits. This can occur only in SC PLL modes **1hz** and **errmes** (i.e. playback), and is normal during initial lock acquisition. At any time afterwards, it means that the S2 is operating at or near the bounds of its playback speed (delay rate) control range.

93)   Mnemonic: STAT_SCPLLACCRAIL

         Type: Informational

  Console Msg: **"System clk PLL accum at/near rail"**

 Interpretation: The System Clock PLL software accumulator value is within 5% of its maximum limits. This can occur only in SC PLL modes **1hz** and **errmes** (i.e. playback), and is normal during initial lock acquisition. At any time afterwards, it means that the S2 is operating at or near the bounds of its playback acceleration control range.

94)   Mnemonic: STAT_DRTAPRAIL

         Type: Informational.

  Console Msg: **"DR nnn taps at/near rail [nnn]"**

 Interpretation: One or more data-recovery FIR filter taps on one or more DRD boards are at or near their maximum limits. This could indicate poor or degenerate DR calibration, possibly caused by marginal components on the DRD board, but can usually be ignored. If this condition is also accompanied by poor signal channel performance, you might try `dr reset flat` to reset the filter taps. You should first do `setup transport N` where N is the address of the DRD board given in the message so that the right board gets reset.

95)   Mnemonic: STAT_DREQSETRAIL

         Type: Informational.

  Console Msg: **"DR nnn EqSet value at/near rail"**

 Interpretation: The data-recovery equalization value on one or more DRD boards is at or near its maximum limits. This could indicate poor or degenerate DR calibration, but can usually be ignored. Requires verbose level 2 or higher to appear on both the console and the RCL.

96)   Mnemonic: STAT_DRFSETRAIL

         Type: Informational.

  Console Msg: **"DR nnn FSet value at/near rail"**

 Interpretation: The DR clock-recovery PLL center frequency (FSet) on one or more DRD boards is at or near its maximum limits. This could indicate poor or degenerate DR calibration. Requires verbose level 2 or higher to appear on both the console and the RCL.

97)   Mnemonic: STAT_TRANDIEDBORN

         Type: Error (clear-on-read)

  Console Msg: **"Transports died/born: nnn/nnn"**

 Interpretation: Shows the number of transports which have been declared "dead" and/or subsequently come back to life since the last status read (RCL)/reset (console). A transport is declared dead after 3 unsuccessful communication attempts (2 retries). A dead transport is indicated by "--" in the console Transport window. This message will appear when transports are swapped while the S2 is powered on, or

when transports are programmed. When you are finished programming, or have connected the new transport and assigned its address, type `status reset` to clear the status message.

98) Mnemonic: STAT_TRANPKTRETRY

Type: Informational (clear-on-read)

Console Msg: **"Transport packet retries: nnn"**

Interpretation: Shows the number of Transport Control Link communications retries since the last status read (RCL)/reset (console). This is a measure of the rate of communication errors over the TCL link. A non-zero retry rate is not normally a problem unless it leads to consecutive retries and transports being declared "dead". Excessive retries could indicate a problem with the TCL cable, or a partial failure of the RS-485 balanced line driver/receiver on the Formatter or one of the transports. This message requires verbose level 3 to appear on both the console and the RCL. To see a debugging message on each retry showing which transport was being addressed, set the verbose level to 1 and type `transport debug on`.

99) Mnemonic: STAT_DRFSETFADJ

Type: Informational (clear-on-read)

Console Msg: **"DR nnn Fine FSet adjustments: nnn"**

Interpretation: Counts the number of incremental FSet adjustments following clock-recovery lock acquisition. If more than one DRD board is affected only the maximum count value is shown. Excessive counts coupled with unstable data-recovery calibration indicate a possible failure on the DRD board related to the VCO read-back ADC. Requires verbose level 3 to appear on both the console and the RCL.

4.6 Power-on Self Test

The power-on self-test procedure, self-test 1, is a comprehensive diagnostic designed to report a single fault or fault class, with the limitation that it will not initiate tape motion and therefore does not perform a complete record/playback test (see Section 2.8 if you wish to test record and playback). Multiple faults will correctly cause self-test 1 to fail but may not be completely reported (typically only one of the faults is shown). The test makes no assumptions about user inputs on the C1/C2 cables except when the default System Clock PLL mode is **refclk**, in which case the user reference clock and 1 Hz signals must be present. Self-test 1 runs automatically as part of the S2 startup procedure, unless the **skipselftest1** defaults file parameter is set to 1. If desired the self-test may be executed manually from the console by typing `diag self1 verbose`. While the test is running most RCL commands which affect tape motion or system switch settings are prohibited. The following checks are performed by self-test 1:

1. Prerequisite Checks
    - [ ] Ensures UI, FORM, and DRD boards are present and successfully downloaded. The number of DRD boards expected to be present is determined by the **diagnumtran** defaults-file parameter (usually 8).
    - [ ] Ensures the required number of transports is present, as given by the **diagnumtran** defaults-file parameter (usually 8).
    - [ ] If the System Clock PLL is in **refclk** mode, checks that the PLL is properly locked (i.e. no SC PLL sync errors detected). If in any other SC PLL mode, self-test 1 switches to **xtal** mode for the duration of the test.

    Failure at any of the above steps causes self-test 1 to terminate and print a descriptive message to the console interaction window. The status window will indicate ''Diagnostic test failed'' with a negative numeric error code.

2. On-line Status Check

    At this point the on-line status is checked for any fatal non-clear-on-read conditions. The on-line status operates continuously regardless of what operation the S2 is currently performing (even during the self-test). Some of the relevant status conditions include:
    - [ ] VME +5/-5.2/+12/-12 V power supply bad
    - [ ] System internal 64 Hz frame sync (SRSYR) absent
    - [ ] Transport software failure (including TCP EEPROM checksum test)
    - [ ] Transport +5V/-5V supply bad
    - [ ] Transport record clk/data bad
    - [ ] UI verifier shows failure condition (indicates distributor/de-distributor failure)

    If any fatal non-clear-on-read status conditions are found the self-test terminates. The status window will indicate ''Diagnostic test failed'' with diagnostic code DIAG_ONLINEFAIL.

3. Signal Path Checks

    This series of tests verifies the integrity of S2 internal data paths.
    - [ ] Checks transport AGC levels are between 100 and 254 in bypass.
    - [ ] Checks all N bypass channels have correct sync & auxiliary data, and show acceptable FORM BERC error rates (using MLSD decoding). N is determined by the **diagnumtran** defaults-file parameter (usually 8).
    - [ ] Checks all N bypass channels show acceptable FORM BERC error rates using bit-by-bit decoding (reduced signal margin test, produces warning message only, does not fail self-test).
    - [ ] Checks C4 cable/8-way splitter (C4p) wiring corresponds to DRD addresses. User can also visually check left-to-right sequencing of DRD ''OK'' lights during bootup to ensure addresses are assigned in the proper order.
    - [ ] Checks transport addresses correspond to C3/C4 cable channel numbers.

☐ Using mode diag8 (or diag4 if fewer than 8 transports are present) to inject the UI diagnostic sequence, checks that all 16 user data channels show acceptable UI BERC error rates (bypass path).

☐ Checks all permutations of the DRD distributed crossbar switch to ensure that each DRD board can output on all 8 internal data channels.

Failure at any of the above steps causes the self-test to terminate and print a descriptive message to the console interaction window. The status window will indicate ''Diagnostic test failed'' with a positive numeric diagnostic code.

In addition to self-test 1, some batch files are supplied with the S2 to test record and playback operation:

berlp.tst records 30 minutes of FORM test vector data at speed LP and plays it back, measuring the bit-error rates obtained.

berslp.tst records 30 minutes of FORM test vector data at speed SLP and plays it back, measuring the bit-error rates obtained.

inter.tst repeats the playback portion of either of the above tests after the tapes have been interchanged.

These batch files are intended to verify correct S2 operation following shipment. See Section 2.8 for further details.

5. TAPE

5.1 Choice of Tape and Tape Speed

The following types of tape are approved for use in the S2-RT and S2-PT. Other types of tape should be avoided.

Type

1. Scotch$^{®}$ ST-120/SE-180 (20 micron) SVHS tape manufactured by 3M, now no longer in production. This was available in North America as ST-120 and in Europe/Australia as SE-180 (about 5% longer than ST-120).

2. Maxell ST-182/SE-260 (13 micron) SVHS tape. Also available as the very slightly shorter Maxell ST-180. May be designated ''XR-S Black'' or ''Professional Broadcast Quality''. Note: These thin tapes will not work in systems with AG2510 transports.

3. Ampex 289 ST-126/SE-180 (20 micron) SVHS tape. Some cassettes may be labeled ST-120 but are actually the same length as the ST-126.

4. Fuji H471S ST-120/SE-180 (20 micron) SVHS tape. Available in North America as ST-120 and in Europe/Australia as SE-180 (about 5% longer than ST-120).

Refer to Table 5-1 for a summary of tape types and recording times. The ''fast'' S2 tape speed is LP (29 micron track pitch), and gives 4 hours 1 minute total recording time per ST-126/SE-180 tape at a specified bit-error rate of less than 1×10^{-4}. Typical bit-error rates of around 1×10^{-5} can be expected in LP. Note that AG2550 transports cannot record in LP, but can play back LP tapes recorded using other transports. Operating the system at the slower SLP speed (20 micron track pitch) gives 50% longer recording times, or 6 hours 2 minutes on ST-126/SE-180 tapes. SLP bit-error rate is specified at less than 1×10^{-3} averaged across 8 transports, with typical bit-error rates around 2×10^{-4}. Note that AG2510 transports can record in SLP but should not be used for SLP playback due to inadequate performance.

With Maxell ST-182/SE-260 "thin" tapes the SLP recording time is extended to 8 hours 41 minutes, and although thin tapes have not been formally specified BER performance should be similar to thick tape. Note that thin tapes (those longer than ST-126/SE-180) do not work in AG2510 transports, but are fine in AG2520 and later model transports.

The type of tape being used in the S2 must be indicated using the `tapetype` console command or TAPETYPE_SET RCL command. A default setting to use at bootup can be defined using the **tapetype** parameter in the system defaults file (the defaults file is described in Section 4.2.7). The tape type is a one-digit code indicating one of several approved varieties of tape as listed above and in Table 5-1. The tape type code is used by the S2 to determine transport write-current levels, so technically it does not need to be set for playback but setting the correct tape type is still recommended. For unlisted tape types it is possible to enter digits NNNMMM where NNN is the write current for LP and MMM is the write current for SLP. These numbers must be determined empirically and will typically be supplied by CRESTech/SGL when a new tape type is approved for use in the interim until the next offical software release (which will recognize the new type).

Tape Type	Description	Length (meters)	Thick/ Thin	LP Time	SLP Time	Notes
1	3M Scotch ST-120	246	thick	3h 50m	5h 45m	discontinued
1	3M Scotch SE-180	258	thick	4h 01m	6h 02m	discontinued
2	Maxell ST-182/SE-260	371	thin	5h 47m	8h 41m	not for AG2510s
2	Maxell ST-180	367	thin	5h 43m	8h 35m	not for AG2510s
3	Ampex 289 ST-120/ST-126/SE-180	258	thick	4h 01m	6h 02m	recommended
4	Fuji H471S ST-120	246	thick	3h 50m	5h 45m	
4	Fuji H471S SE-180	258	thick	4h 01m	6h 02m	

Table 5-1: S2 Approved Tape Types

5.2 Tape Shipping and Storage

Tapes must be shipped and stored in closed containers. Canadian Space Agency (CSA) tapes use a low-cost cardboard sleeve designed to hold a set of 8 tapes, and a padded cardboard shipping container that holds 4 tape sets or 32 tapes total. Whenever tapes are not in the transports they should always be placed in their containers. If no CSA or equivalent tape containers are available standard VHS dust jackets may be used.

For maximum data integrity, tapes should be stored vertically in the fully rewound position. Tapes should be fully rewound when shipped.

Tape Shipping Environment

The ideal shipping environment for tape is 19–22 degrees Celsius, at a relative humidity of 45% to 55%. Environmental limits are to be determined.

6. MAINTENANCE AND SERVICING

This chapter provides information and procedures for servicing and maintaining the S2-PT. For additional or updated service procedures please see the following directory on our web site: *ftp://ftp.sgl.crestech.ca/pub/s2/doc/service/*

6.1 Troubleshooting Hints

The following are explanations for some commonly encountered problems.

1. **Q**: When issuing a record command, one or more tapes are ejected (model AG2510 transports) or the transports beep several times and refuse to go into record (model AG2520 or AG2530 transports) or the transport tape display flashes and they refuse to go into record (model AG2550 transports).
    A: The write-protect tab on the edge of the tapes is broken off. See the `record` command on Page 4-43.

2. **Q**: The AG2520 transports make unexpected clicking noises after they have been stopped for about 5 minutes.
    A: Don't worry, this is just the sound of the tapes automatically unwrapping. It is normal.

3. **Q**: The system reports spurious interrupt errors after adding or removing the Ethernet board.
    A: The IACK backplane jumper for the Ethernet board's VME slot (slot 2) must be put in place when the Ethernet board is removed, and taken out when the board is inserted. See Section 4.4.

4. **Q**: One or more transports are off and refuse to turn on.
    A: The transports automatically shut off when the internal temperature is above 60 or below 5 degrees Celsius. They can also shut down if excessive internal condensation is detected. In this case the console display will show ''off'' for the affected transports. First unplug the transport power cords and wait for conditions to stabilize. If a transport still refuses to turn on, it could be that the temperature calibration factor has been corrupted and needs to be re-entered (see Section 6.4.5). If this is not the problem or the transport appears dead (shows ''--'' on the console display), the transport EEPROM could be corrupted. You might be able to reload it using the eject-button emergency procedure outlined at the end of Section 6.2.

5. **Q**: During bootup the S2 reports ''Error: Checksum incorrect -- SRAM corrupted''.
    A: Something has caused the contents of the 3 MB Static RAM board to be corrupted. You will need to reload the affected partitions or the entire SRAM. See Section 6.2 for details.

6. **Q**: I can't log on using rlogin/telnet or connect an RCL socket from outside the local network.
    A: You must specify a gateway inet address in the S2 boot parameters. Also the subnet mask must be correct in the ''inet on ethernet'' field of the boot parameters for gateways to work. See Section 4.4.1 for more information.

6.2 Software Reloads/Upgrades

The S2 Recorder Operating System software resides on the battery backed-up Static RAM board, and can be reloaded via the console serial port in order to upgrade software or recover from an SRAM failure. To do so you must operate the S2 from the ROSCO terminal emulator program. ROSCO runs on IBM PC and compatible computers, and also on PCFS Linux computers using the 'dosemu' DOS emulator. The data file for reloading the S2 SRAM as well as the ROSCO program and other utilities are supplied with the S2 on floppy diskettes. These files are also available from the following FTP directory: *ftp://ftp.sgl.crestech.ca/pub/s2/swload/*
This is the distribution directory for the latest S2 ROS software versions. Check the README file in this directory for downloading instructions. Also read the release notes supplied with each new software version. For information

and files necessary to run ROSCO on PCFS Linux computers using 'dosemu', see:
ftp://ftp.sgl.crestech.ca/pub/s2/pcfs/dosemu/

To reload the S2 software follow these basic steps:

- Connect a serial cable between the S2 console port (C7) and COM port 1 of your PC. Note: COM 2 can be used if necessary by changing the port selection in ROSCO (press shift-f5 then f1).
- While running ROSCO reboot the S2 (one way is to turn the power off and on). Immediately enter the boot monitor by hitting the space bar during the initial boot countdown.
- In the boot monitor type `restore`
- Select the ROSCO file menu by pressing shift-f9. If desired press f3 to specify a partition name, then press f5 to transfer data to the Static RAM. You will be prompted for the file name of the SRAM image to transfer; enter the name of the .bxo file for the appropriate software version, e.g. `b970207.bxo`

The transfer should start immediately, and will show its progress by displaying the number of bytes transmitted so far. Reloading the entire 3-Megabyte Static RAM takes about one hour at the default console baud rate of 9600 baud. Reloading using PCFS and 'dosemu' takes longer, up to 3 hours, due to the slow speed of the emulator. Note that a full SRAM reload will *delete any batch files and configuration changes you may have made*. It is sometimes possible to avoid this problem by reloading only specific parts of the SRAM: Before pressing f5 to start the transfer, press f3 to select a particular partition (**root, bin, exo, dr_config, tcpbin, vxworks** etc.) or **all** for the entire SRAM. For some software upgrades, only a few of the partitions may need to be updated (which ones will be indicated in the release notes supplied with the upgrade), and for certain SRAM failures, only a single partition may need to be reloaded. Important batch or configuration files can also be saved and reloaded individually over the network to a suitably configured remote host (use the full remote pathname syntax in the editor's :w or :e commands, e.g. `:w boot.host:/s2dir/file.bat`, or in the VxWorks shell's `copy` command).

SRAM reloads are possible, and indeed much faster, using the network instead of the serial port, but this requires a properly configured host computer and a different version of the SRAM data file. For more information see *ftp://ftp.sgl.crestech.ca/pub/s2/doc/appnotes/HostSetup* .

Transport Programming

Following a ROS software upgrade, it may be necessary to perform an extra step to upgrade the software which runs on the custom 8051 processor inside the transports. This software is stored in electrically-erasable memory (EEPROM) and can only be changed with the S2 booted and running normally. Reboot the S2 after completing the SRAM reload and verify that all 8 transports are responding properly[1] (the transport window should show all transports are stopped). Then type the following console command:

```
transport all program
```

This will first display the version number and date of the software currently loaded in the transports, followed by the version number and date of the new transport software about to be loaded. The S2 will then wait for user confirmation by asking "Are you sure (yes/no)?". If the version dates match perfectly *there is no need to upgrade the transport software*, so type `no`. Otherwise if the version dates are different type `yes` and wait for transport programming to complete (about 5 minutes). If everything is successful the S2 will report "Transports 0-7: Programmed prog & all". You should ignore most status messages which come up during programming such as

---

[1] *If your S2 system has fewer than 8 transports, all transports present should be responding*

"Tran N failure", "Req'd transpts N are off/dead", "Tape motion state is inconsistent", and "Transports died/born: N/N".

Note that if your S2 system has a mixture of transport types you will need to program transports of each type separately. For example, if your TAM 'A' contains 4 AG2520 transports while TAM 'B' contains 4 AG2530 transports, you must type the following:

```
transport 0+1+2+3 program
transport 4+5+6+7 program
```

If any transport fails to respond within 30 seconds of being programmed the entire operation will abort with an error message. In this case some transports may not yet have been reprogrammed, and one transport may be dead or only partially programmed. If you have a transport that is apparently dead (does not beep and set its front panel address display a few seconds after power-on), it is sometimes possible to recover using a special "emergency recovery" feature: Hold the eject button on the front panel while applying power to the transport. This will force it into program mode, after which you can try to re-program it using `transport N program all.ag25x0` where ag25x0 is the type of transport. This method works *only* for problems related to a (partially) corrupted EEPROM — use it only when all other possibilities, such as a bad TCL cable, have been thoroughly checked. For more information on programming transports see the `transport` command on page 4-51.

6.3 DSCM Electronics Replacement and Servicing

The DSCM electronics have no manual adjustments and should require servicing only to replace long-life lithium batteries located on the SRAM and RCC boards (see below). Removing DSCM PC boards should otherwise only be required in the event of a fault in the electronics which requires substitution of a board.

DSCM PC boards are to be positioned as shown in Figure 6-1.

1. The RCC (MVME133XT) must be positioned in slot 1.

2. The ETHERNET board, if present, should be positioned in slot 2. If no Ethernet board is being used, the IACK (interrupt acknowledge) backplane jumper for slot 2 must be in place. If an Ethernet board is present the IACK jumper must be removed.

3. The 3MB SRAM board should occupy slot 3.

4. Slot 4 should be empty, with the IACK jumper in place.

5. The Formatter (FORM) must be positioned in slot 5.

6. The DRDp boards should be positioned sequentially from left to right starting with DRDp 0 in slot 6.

7. The UI may occupy any of the slots to the right of the DRDp's.

SRAM and RCC Battery Replacement

The SRAM board uses two lithium batteries (type LTC-7PN, 3.5 V, 750 mAH) to retain memory contents while the power is off. These have a rated lifetime of 10 years and should be replaced when the red FAIL LEDs on the SRAM board come on. The MVME133XT RCC uses an SGS-Thomson MK48T02 2K SRAM/clock chip with built-in battery, lifetime 3 to 6 years. It should be replaced when the RCC no longer retains the system time or network boot parameters while the power is off. Note that the S2 may be operated normally after the SRAM/clock chip battery dies, except that the system time will not be retained while the power is off and users requiring the network facilities will have to re-enter the boot parameters each time they turn on the system (see Section 4.4.1).
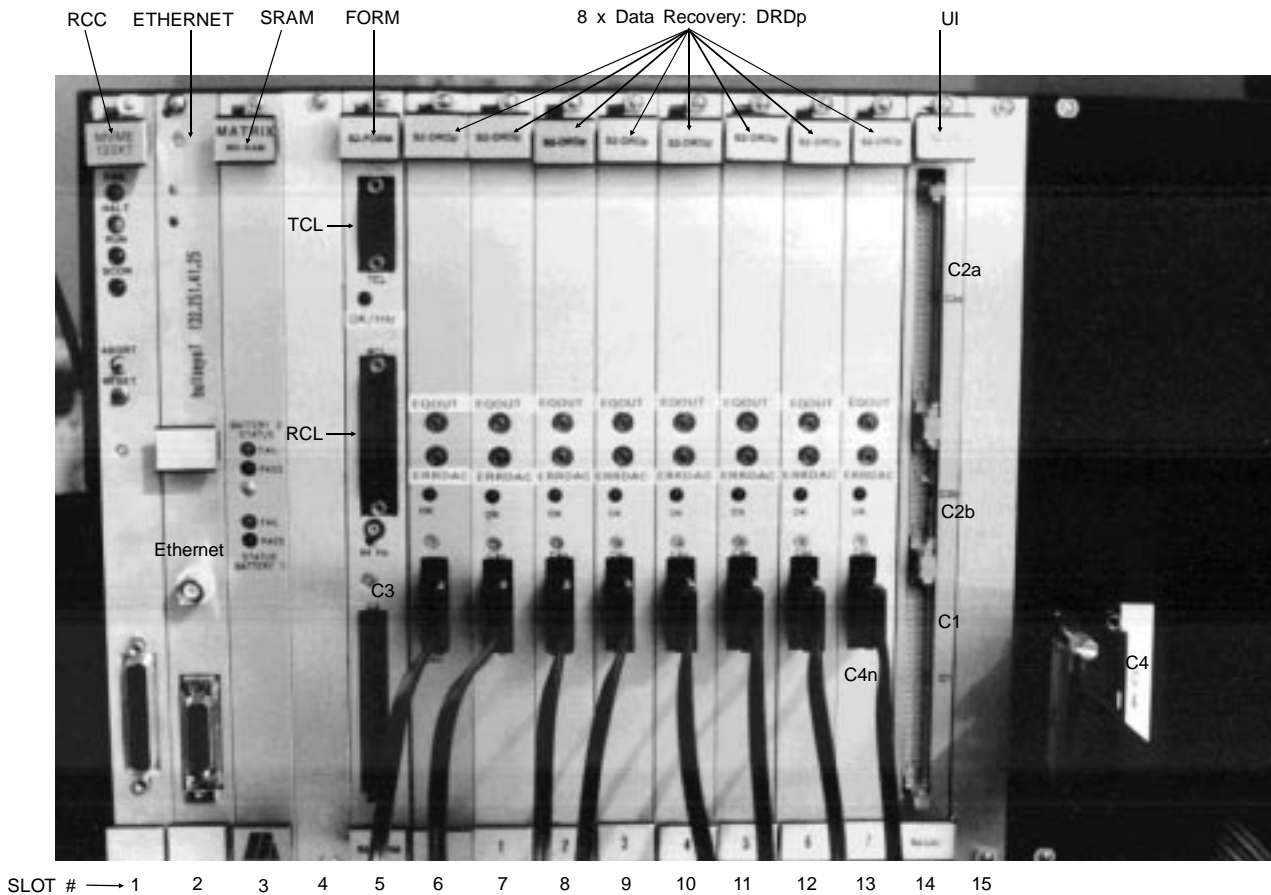
WARNING! PC boards should only be removed or inserted when DSCM power is off. Be careful not to snag components on either side of the board against adjacent boards or panels. Where there is a blank panel to the left of the board being removed, the blank panel should be removed first.

The VME boards (RCC, Ethernet and SRAM) must not be positioned in slot 4 or higher for any reason. The S2 custom boards (FORM, DRD and UI) must not be positioned in slots 1-3 for any reason. Failure to follow this directive will result in damage to the system.

Figure 6-1: DSCM PC Board Configuration

DSCM FORM and DRD Replacement

When replacing DRDp boards the address jumpers on the boards should be set from 0 to 7 corresponding to the slot position the boards occupy, and must match the labels on the ends of the C4p 8-way splitter cable. The jumpers are located near the back of the DRDp board, to the right of the silkscreen label 'ADDRESS'. The three jumper contacts are labeled 0, 1 and 2, corresponding to the binary address of the board (least significant bit = 0). Connecting these pins corresponds to setting the address bit to 0, and disconnected pins correspond to setting the address bit to 1. For example, setting the DRDp to address 6 means only the jumper at position 0 should be installed. After setting the jumpers replace the label plate located in the board extractor on the bottom front of the board so that it correctly identifies the board's address. The plates should slide in and out easily. Note that if the jumpers are installed correctly, when the system is turned on the front panel "OK" LEDs will blink on in sequence from left to right (DRDp at channel 0 to DRDp at channel 7).

Following insertion of a new FORM or DRD board or re-loading/replacement of the SRAM board, the self-test may fail or take unusually long on first power-on due to software recalibration lead time. If the self-test fails

with high error rate or bad sync messages, re-run the test one or two times by typing `diag self1 verbose` on the console. If the test still fails, try the following:

1. Select bypass signal path: `setup bypass`

2. For each internal signal channel 0 thru 7, type `setup N` where N is the channel number, and then wait until the display indicates `Diag dec sync: 100% and Est error rate: <1.0e-6. If this does not occur within one minute, try entering the command dr reset flat`. If this still does not help there may be a real hardware failure.

3. Reset the S2 by turning it off and on or typing `shutdown reboot` on the console. The self-test should now pass.

6.4 TAM Servicing

Other than the transports themselves, the TAMs have no mechanical components requiring servicing except the fan filters. These filters are accessible via the back panel of the TAMs and should be replaced every three months of continuous operation or if the S2 issues a transport temperature warning (transport internal temperature over 55 degrees Celsius).

6.4.1 Transport Replacement And Servicing

The mechanics of the TAM transports are to be serviced according to the recommendations of the manufacturer as defined in the AG2510/AG2520/AG2530/AG2550 service manual and summarized in Table 6-1. The service manual is available from CRESTech/SGL as part of the optional Transport Servicing Kit. Service intervals can be identified using the service and head-use time counters built into the transport software and available for readout using S2 console commands. Two measured time intervals are available: the time since last service, also known as in-service time, and the time since last head replacement, also known as head-use time. A *service* is defined as any time the transport shell is opened for inspection or maintenance such as cleaning/greasing, and should be indicated by typing the following console command after the transport is back in operation:

```
transport N service lastserv
```

A head replacement is defined as any time the upper head cylinder is replaced, and by definition also implies a service. It should be indicated by typing the following console command after the transport is back in operation:

```
transport N service lasthead
```

These two commands clear the time since last service and the time since last head replacement, respectively, which are stored in EEPROM inside each transport. Both of these times are measured in hours and minutes of active head use, and can be displayed by typing

```
transport N status time
```

In addition to the in-service and head-use times, this also shows the total cumulative head use and total on time since manufacture, as in the following example:

Transport 2: On 5764h 51m, Head use 518h 44m/1662h, Serv 178h 02m/1662h

This indicates that

–The transport has been powered on for 5764 hours and 51 minutes since manufacture.

–There have been 518 hours and 44 minutes of active head use since the last head replacement.

–There have been 178 hours and 2 minutes of active head use since the last service.

–There have been 1662 hours of active head use since manufacture.

These values can also be displayed in the transport window using the following three console commands:

```
transport display ontime
transport display headtime
transport display servtime
```

Ref. No. IN P/L	Parts Name	500	1000	1500	2000	2500	3000	3500	4000	4500	5000	Ref. No. IN P/L	Parts Name	500	1000	1500	2000	2500	3000	3500	4000	4500	5000
								Hour													Hour			
7 (1)	Upper Cylinder U	♦	♣	♦	♣	♦	♣	♦	♣	♦	♣	24 (1)	Supply Main Brake U				♣				♣		
6 (1)	Cylinder U	♦	♦	♦	♦	♦	♦	♦	♦	♦	♣	27 (1)	Takeup Main Brake U				♣				♣		
34 (1)	A/C Head	♦	♦	♦	♦	♦	♦	♦	♣	♦	♦	55 (1)	Mode Switch								♣		
17 (1)	FE Head	♦	♣	♦	♣	♦	♣	♦	♣	♦	♣		Tape Transport	♦	♦	♦	♦	♦	♦	♦	♦	♦	♦
45 (1)	Pressure Roller U	♦	♦	♣	♦	♦	♣	♦	♦	♣	♦		Loading Base				♥				♥		
14 (1)	Supply Reel Table U	♦	♦	♦	♦	♦	♦	♦	♣	♦	♦		Cam Gear Section				♥				♥		
71 (1)	Takeup Reel Table U	♦	♦	♦	♦	♦	♦	♦	♣	♦	♦	132 (2)	Timing Belt				♣				♣		
	Supply Reel Shaft				♠				♠			25 (1)	Supply Soft Brake (1) U				♣				♣		
	Takeup Reel Shaft				♠				♠			13 (1)	Tension Band U				♣				♣		
133 (2)	Capstan Rotor U										♣	70 (1)	Main Brake Spring				♣				♣		
	Capstan Shaft	♦	♦	♦	♦	♦	♦	♦	♦	♦	♦	92 (1)	Cleaner Roller U		♣		♣		♣		♣		♣

* Note:

Symbol	Maintenance	Requirement	Remark
♦	Cleaning	Ethyl-alcohol or Cleaning Liquid (purchased locally)	Wipe dirt from the parts using soft cloth impregnated with Ethyl-Alcohol. **Note:** When cleaning rubber parts, avoid using excessive alcohol since it may accelerate deterioration of these parts. After cleaning with alcohol, wipe the alcohol quickly and thoroughly.
♣	Replacement		
♠	Lubrication	High Quality Spindle Oil (purchased locally)	Supply one or two drops of oil.
♥	Greasing	Molytone Grease (MOR265)	Wipe the old grease and apply new grease.

Table 6-1: Maintenance Chart for AG2520 Transports

Transport Service History

Each transport stores a text file within its EEPROM memory which lists details of service steps performed on that transport in the past. This file is useful for record-keeping by service personnel and should be updated each time a transport is serviced, cleaned, repaired, etc. Type the following command to bring up the service history file in the editor:

```
transport N service edithist
```
An example service history file is shown below.

```
Service history for AG2550 transport [s/n K7KD00164].
Currently installed TCP is tcp4102.
     Date        Name      TotHead*   Work Description**
     ------      -------   ----------  ---------------------
Jul 15, 1998   ROS         0019       Service history started.
Jan 21, 1999   P. Newby    0627       S0500 Cleaned tape path.


-----------------------------------------------------------------
* Actual total head time (hours).
** Include service code S[NNNN], where NNNN = nominal service hours
   (as per AG25X0 service chart), plus additional notes.
```

Transport Replacement

Replacement or servicing of a transport requires the removal of the transport from the TAM rack, as follows:

a) Transport Removal

1. Power down the system. This is a precautionary step only. If operational requirements make it necessary to do so, the transports can be removed while the power is on.

2. Unscrew and remove the slider block from the rack infill brackets on the front of the rack. Remove the TAM back panel. Unplug the ribbon 50 ohm cable, serial and power cables from the back of the transport to be removed.

3. Slide the transport forward until the slider catch is encountered. Remove the transport by depressing the catches on both sides of the extended sliders while pulling forward on the transport. The transport will slide free of the rack. AG2550 transports have no sliders and are removed by pulling them forward from the front. In some cases it may first be necessary to push the transport forward a few inches from the back of the TAM.

b) Transport Insertion

1. Power down the system. This is a precautionary step only. If operational requirements make it necessary to do so, transports can be inserted while the power is on.

2. Make a note of the TCP serial number of the transport(s) being inserted (number is printed on a label on the back of the transport). This will be needed later.

3. Engage the sliders and slide the transport backwards until the slider catch is encountered. Depress the catches on both sides of the extended sliders while pushing backward on the transport. The transport will slide into the rack. Some pressure may need to be applied to the front of the rack to overcome the resistance of the extended sliders. AG2550 transports have no sliders and are inserted by simply pushing them into the appropriate slot as far as they will go.

4. Screw in the slider block to the rack infill brackets on the front of the rack. The slider block does not need to be tightened excessively. Plug in the ribbon 50 ohm cable, serial and power cables into the back of the transport. Screw on the TAM back panel. Note that thumbscrews should be *FINGER-TIGHTENED ONLY* when securing S2 transports and other TAM hardware.

c) Transport Address Assignment

Replacement of a transport in the TAM rack, or replacement of the TCP board within the transport, may require reassignment of the transport address. The address is what distinguishes control messages to different transports and must match the C3/C4 cable channel the transport is connected to, which depends on its position in the TAM racks. Address reassignment will normally be accomplished through the console. On the command line, enter

```
transport assignaddr [SERIAL NUMBER] [ADDRESS]
```
where `SERIAL NUMBER` is the four-digit serial number found on a label on the back of the transport of the form `TCP[SERIAL NUMBER]`, and `ADDRESS` is the channel number of the transport (0 - 7).

If at any time two (or more) transports have the same address, both will appear ''dead'' in the transport display. However, the `transport assignaddr` command should still work properly provided the serial number used is correct. If for some reason a transport's serial number label is lost or becomes unreadable, the number can be obtained using `transport N status serial` where N is the transport's address. If at the same time the transport's address is invalid (front panel display shows 69) or conflicts with other transports, proceed as follows: Disconnect the TCL serial cable (C5) from all transports except the one in question, then issue the console command `transport 255 status serial`. Re-label the transport serial number immediately.

Transport TCP Board Replacement

Occasionally it may be necessary to replace or swap the TCP board inside a transport due to failures. Users and service personnel must be careful when replacing the TCP board since this board contains the EEPROM which
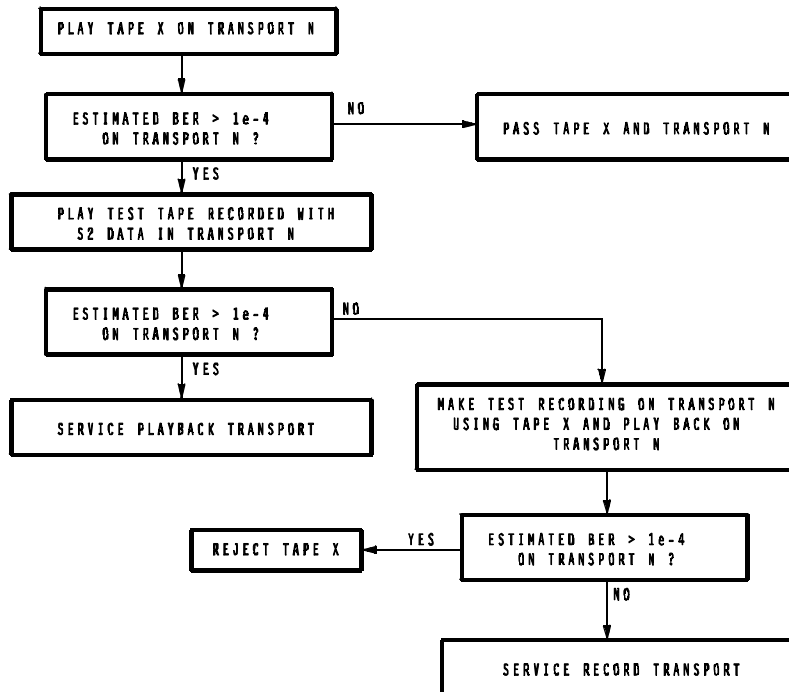
Figure 6-2: Transport/Tape Diagnostics (LP)

holds important information such as head use time, service history, and pgshift and write current calibration parameters. This service information must be transferred to the new TCP board. Therefore before you remove the original TCP board type the following console command:

```
transport N service dumpmem tofile
```

After replacing the TCP board and reassigning the correct address as explained under "Transport Replacement" type this second command:

```
transport N service loadmem NNNN
```

Here NNNN is the *old* TCP serial number, since the new board will have a different serial number. Be sure to update the label on the back of the transport to show the new TCP serial number, and update the service history file.

If a TCP board is being replaced because it has failed you may not be able to carry out the initial dumpmem step. To guard against this problem the S2 performs a dumpmem for each transport on every system boot, so you should still be able to transfer recent values for the EEPROM parameters just by carrying out the second step. The command `config ls long` can be used to check the date of the latest service info dump for a particular transport (look for a dump file name containing the old TCP serial number). In the event that the configuration save area becomes full you may start to see error messages when dumping service info; if so you should type `config delete all` to erase all dump files and start fresh.

6.4.2 System Service Calls

The S2 console displays an estimate of bit error rate (BER) during playback (see Section 4.2.8.1). In normal operation this will indicate an estimated BER of 1×10^{-4} or less (tape speed LP), or 1×10^{-3} or less (tape speed SLP). Although these values may occasionally be exceeded for short periods, in the event of regular occurrence

of BER estimates over spec investigation of the cause should be initiated. This investigation consists of the steps indicated in Figure 6-2.

6.4.3 Transport Setup For Off-Line Tests

Normal operation of the transport requires that it be supplied with clock and synchronization signals from the C3C4 cable connection on the back of the transport. In a test configuration, these signals are supplied by the DSCM via C3C4/C5 extender cables, available from CRESTech/SGL as part of the Transport Servicing Kit or Tape Guide Adjustment Kit.

6.4.4 Transport Write Current Optimization

Write current optimization is the process where the best write current level for recording is determined individually for each transport. Any differences from the nominal write current setting are entered as an offset (wrlevoffset) stored in the transport EEPROM. A significant improvement in bit-error rates — up to a factor of two or more on average — can be obtained as compared to operating with a single global write current setting (all offsets zero).

As of ROS version 3.2 a new fully automatic procedure is provided to optimize write currents and set the per-transport offsets. Users should run the procedure each time the rotary heads are replaced (1000 hour service interval). We also recommend running it at the 500 hour service interval due to possible effects of head wear. It's a good idea to run the post-installation record/playback test (Section 2.8) before and after optimizing write currents to ensure that there is no unexpected degradation in system performance; bit-error rates should improve or stay roughly the same.

The steps required to perform write current optimization are as follows:

1. Using the S2 console, select which transports you want to optimize by typing
   ```
   transport N+M+...
   ```
   To optimize write currents on all transports type
   ```
   transport all
   ```

2. Insert writable tapes in the selected transports. You should use the type of tapes you use most often in your S2 system.

3. Rewind the tapes, or position them to a point at least half an hour away from the end.

4. Set the tape type which corresponds to the tapes you are using and select speed SLP:
   ```
   tapetype N
   speed slp
   ```

5. Enter the following command to begin the optimization procedure:
   ```
   transport wrlev optimize
   ```
   This will iteratively record and play back tape, and may take several hours to complete (more than 5 hours when doing all eight transports), so you may wish to let the procedure run overnight. Here is an example of the output you will see in the interaction window:

   ```
   Now performing write current optimization:
   transports 0-7, speed 'slp', min # of iter=10, step size=10/6, tape type=3
   Sit back. This will take about 5 hours and 15 minutes, possibly a bit longer.
   (Press Cntl-C at any time to quit and display results up to that point)
   Iteration 1: Record phase.
   Write currents: 0: 70   1: 70   2: 70   3: 70   4: 70   5: 70   6: 70   7: 70
   Transport temp: 0: 37   1: 38   2: 37   3: 35   4: 38   5: 37   6: 36   7: 38
   Write currents: 0: 55   1: 55   2: 55   3: 55   4: 55   5: 55   6: 55   7: 55
   Write currents: 0: 65   1: 65   2: 65   3: 65   4: 65   5: 65   6: 65   7: 65
   Write currents: 0: 75   1: 75   2: 75   3: 75   4: 75   5: 75   6: 75   7: 75
   ```

```
Write currents: 0: 85  1: 85  2: 85  3: 85  4: 85  5: 85  6: 85  7: 85
Iteration 1: Record phase complete, now repositioning & playing back.
Triggering phase-1 tracking calibration (will also do phase-2)

     -- Raw Bit-Error Rates For Iteration 1 --
tran 0: wrlev  55: 354    65: 721    75: 528    85: 825    new  wrlev: 65
tran 1: wrlev  55: 2999   65: 3409   75: 3893   85: 4430   new  wrlev: 55
...
tran 7: wrlev  55: 555    65: 705    75: 766    85: 1057   new  wrlev: 55
Iteration 2: Record phase.
Write currents: 0: 65   1: 55   2: 55   3: 55   4: 65   5: 65   6: 75   7: 55
```

The procedure will continue for ten iterations, possibly more if the write current hasn't converged yet for some transports. When done, it will display the final write current from the last iteration, and the ''best'' write current for each transport. The best write current is an average of the results from iterations following convergence (the number of iterations included in the average is shown on the second-last line, after the slash).

```
Write current optimization ended, final & "best" average write currents:
tran 0: 60  1: 72  2: 48  3: 38  4: 78  5: 72  6: 70  7: 54
#iter 10/3   10/6    10/3    10/3    10/6    10/5    10/4    10/5
best 0: 61  1: 71  2: 49  3: 40  4: 81  5: 73  6: 72  7: 54
[Hit return to continue]
```

At this point you should press return. After confirming the tape type, the new optimum per-transport write current offsets are calculated and then you will be asked to confirm before the new offsets are actually written into the transport EEPROMs:

```
Tape type is 3: Ampex 289 ST-126/SE-180  (70/70)
Is this the correct tape type for the tapes now in use (y/n/q)? y
Here are the old and new optimum per-transport write current offsets:
Old: 0:-12  1:-4    2: 33  3: 25  4:-22  5:-10  6: 4    7: 0
New: 0:-21  1:-3    2: 12  3:-5    4:-11  5:-7    6: 6    7:-16
Enter "y" or "yes" to write the new per-transport write current offsets into
the transport EEPROMs, "no" to skip: y
Transport 0-7 write current offsets updated to new optimum values.
```

All displayed information is also written to the S2 internal log and can be reviewed later by typing `log display E fullscreen`. Offset values should not normally be larger than about 40 or 50, although almost any value can be fine if the transport performs well. If you suspect a problem with the resulting write-current offsets you should not write them into the transport EEPROMs (answer ''no''). Try repeating the optimization, or contact s2support for advice (include a copy of the log output).

Note that if you have AG2550 transports you should try to perform write current optimization under the same temperature conditions as your normal record operations, since AG2550 write currents are somewhat temperature dependent.

6.4.5 Transport Temperature Calibration

Calibration of the transport temperature sensors should be performed if there is reason to suspect that one or more of the sensors are out of calibration. It may be convenient to do this whenever the TAM fan filters are replaced. The calibration procedure is as follows:

1. Power down the system for eight hours in a temperature-stable environment so that the transports can cool to room temperature.

2. While the system is off, measure the ambient temperature near the TAMs in degrees Celsius. If the TAMs are not exposed to localized temperature or air flow conditions, the temperature of the room should be adequate for calibration.

3. Disconnect the C3 and C4 cables between the TAMs and DSCM and turn on the power of the DSCM only so as not to disturb the temperature of the transports.

4. Following the power-on self-test (which has failed owing to the disconnected cables), command the system to display transport temperatures on the console:

```
transport display temp
```
(The temperatures will not be shown yet since the TAMs are still off.)

5. Reconnect the C3 and C4 cables and then turn on the power to the TAMs. As soon as all the transports are responding (as observed in the console transport window), calibrate the temperature as follows:

```
transport all caltemp [REF]
```
where [REF] is the temperature in degrees Celsius measured in step 2 above. If any transports are off due to (erroneous) under- or over-temperature conditions you should use ''0+1+2+3+4+5+6+7'' instead of ''all'' in the command above.

6. If any transports are off due to previous under- or over-temperature conditions, type

```
transport 0+1+2+3+4+5+6+7 power on
```
to turn them back on.

For additional information see the description of the `caltemp` option of the `transport` command on Page 4-53.

6.4.6   Transport Tape Guide Adjustment

Poor transport performance (as shown by high estimated error rates or error rates above spec in the post-installation record/playback test, Section 2.8) may indicate that the transport P2 and P3 tape guide posts require adjustment. In particular, this adjustment has sometimes been found necessary following shipment of S2 systems.

Required Tools

The tools required to perform tape guide adjustment are supplied as part of the Transport Tape Guide Adjustment Kit. They are also part of the larger Transport Servicing Kit. The required tools are as follows:

- Post adjustment screwdriver, Matsushita/Panasonic part# VFK0329
- Transport C3C4/C5 extender cable (TRXTNDR)
- SMB-to-BNC converters (2)
- Matsushita/Panasonic NTSC alignment tape, part# VFM8080HQFP
- "S2 alignment tape" (best tape recorded during post-installation test, not supplied, see below)

Basic Steps

Stripped to its essentials, the procedure is described by the following sequence. Detailed instructions are given later below.

A. Measure baseline performance using the "S2 alignment tape". Record the baseline data reported by the S2 console display.

B. Adjust the tape guide posts using an oscilloscope and the Matsushita/Panasonic NTSC alignment tape. The guide posts are adjusted by determining the adjustment range which produces an acceptable waveform and setting each post to the mid-point of this range. If no oscilloscope is available, the adjustment may be carried out using S2 console screen displays instead.

C. Re-evaluate the performance using the "S2 alignment tape" as in part A. Hopefully the adjustment results in improved performance.

Users with access to the appropriate Matsushita/Panasonic service manual may wish to read over the section on tape guide post adjustment for reference only (AG2510 Service Manual section 3-4-2, AG2520 Service Manual section 3-4-2-4, AG2530 Service Manual section 2-2-4, or K-Mechanism Service Manual sections 3-1 and 3-2 for AG2550).

Part A: Measure Baseline Performance

1. Insert an "S2 alignment tape". This is a tape recorded by an S2 transport known to be in good calibration. A good way to obtain an "S2 alignment tape" is to use the tape from the post-installation checkout record/playback test (Section 2.8) which had the best overall performance (lowest combined interchange and non-interchange bit-error rates). You should use a tape recorded by the same type of transport as the one being adjusted. For AG2510 transports only ST-126/SE-180 or shorter tapes may be used, and the record speed should not be SLP.

2. The transport under test should be the only transport selected, type:
   ```
   transport N
   ```
   Here N is the number of the transport to be adjusted.

3. Put the transport into manual play, rewinding first if necessary:
   ```
   rewind
   play manual
   align selfalign
   ```

4. Wait for tracking calibration to complete (about 45 seconds; watch for ''track cal'' changing to ''uncalib'', finally disappearing):

```
wait drcal
```

5. Measure bit-error rate:

```
setup transport N
form berc 0 30 s
```

Write down the average bit-error rate obtained after 30 seconds.

6. Measure the flatness profile:

```
def syncerrlocate enable
def syncerrlocate histogram 0 249 30 s cumulative
```

This measures the tape error distribution across the frame for 30 seconds. Write down the flatness and scale values obtained.

7. Stop and eject the tape:

```
stop
eject
```

Part B: Adjust Posts

An oscilloscope is normally required to carry out the guide post adjustment. If no oscilloscope is available, the adjustment may be carried out using the alternate "no-scope" method given later below.

Setup

Remove the transport to be adjusted from its TAM as described on Page 6-7 and position it in an accessible location. Take off the top cover by removing the four black screws (two on each side) and lifting the cover starting at the back of the transport. The slider brackets do not need to be removed. Reconnect the transport using the transport extender cable (TRXTNDR) and a spare power cable.

Connect the EQOUT test port on the DRD board to oscilloscope CH1. Connect the FSYNC test port on the Formatter board to oscilloscope CH2. SMB-to-BNC converters will likely be required (insert carefully). Set oscilloscope CH1 to 50 ohm coupling, 2ms/div. Trigger on CH2 rising edge, also 50 ohm coupled. If 50 ohm couplers are not available regular high-impedance inputs may be used.

A special alignment tool is required to adjust the tape guide posts, Matsushita/Panasonic part# VFK0329. (In a pinch, a flat screwdriver can be modified for the task by filing or grinding a notch into the middle of the blade, and reducing the screwdriver blade thickness as required to fit the guide posts.) The two posts which move, wrapping the tape around the head drum when the tape is played, are designated P2 and P3. The P2 post is the one on the supply-reel side; P3 is on the takeup-reel side — see Figure 6-3. In general, problems at the beginning of a frame are corrected by adjusting P2, while end-of-frame problems are corrected by adjusting P3.
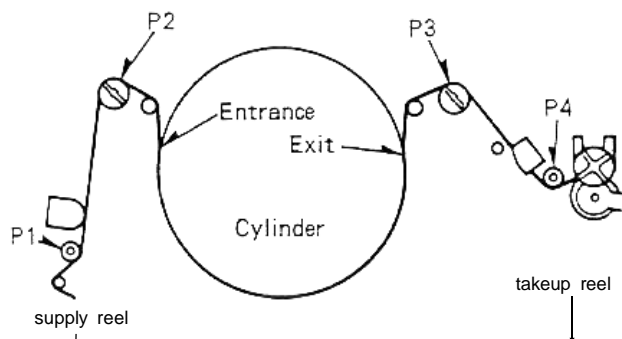
Figure 6-3: Location of Guide Posts

Using some kind of ink pen, place one dot on the perimeter of each tape guide post to serve as a reference.
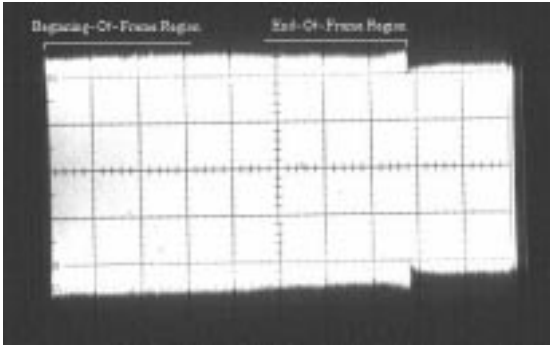
Figure 6-4: Proper tape guide adjustment
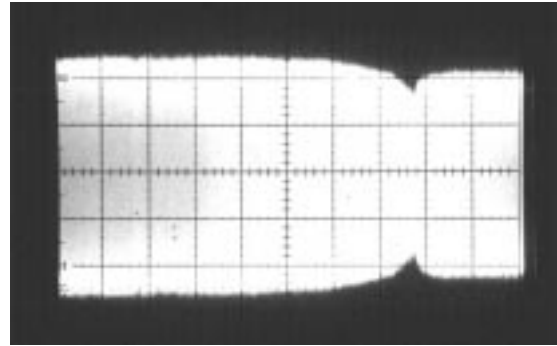
Figure 6-5: End-of-frame distortion example1
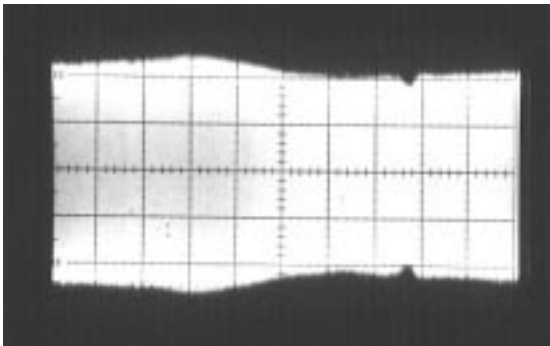
Figure 6-6: End-of-frame distortion example2
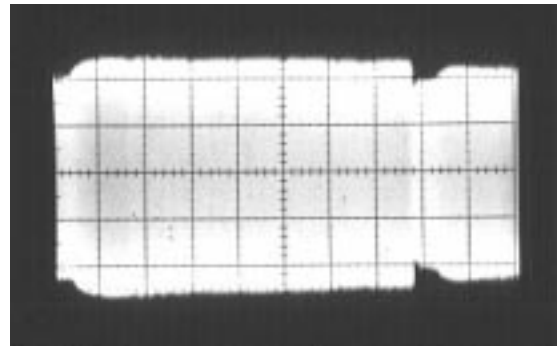
Figure 6-7: Beginning-of-frame distortion example

Procedure

1. Place the NTSC alignment tape (Matsushita/Panasonic part# VFM8080HQFP) in the transport and rewind it. The portion of tape which is relevant to this adjustment is the first section, ''Envelope Interchange''. Note that this is a special tape; it will auto-stop at the end of the first section, about 9 minutes. If you need more time you should rewind the tape and then *repeat steps 2 and 3*. Should you find the alignment tape is in the wrong section it will need to be manually advanced between sections (in either direction). Mark the spine label to indicate the number of uses. Replace the tape after 25 uses, or if it suffers edge damage as could be caused by a severely misaligned transport. If no Matsushita/Panasonic alignment tape is available and it would take too long to obtain one, you may use the ''S2 alignment tape'' from part A.

2. Put the transport into manual play:
```
transport N
play manual
```

3. Turn off gain controls and tracking calibration by running the **agcoff** batch file, supplied with ROS software version 3.1 and later. If you do not have this batch file, refer to the listings of batch files at the end of this section and type it in.
```
batch agcoff
```

4. Adjust the P3 post. Turn the post slowly clockwise until the end-of-frame portion of the displayed waveform is visibly distorted (see Figures 6-5 and 6-6). Note the position of the P3 post. Then, turn the post counter-clockwise until the waveform first improves and then shows distortion to a similar degree. Note the position of the P3 post, and turn the post clockwise again to the midpoint of the previous two adjustment positions. Always turn the post slowly, and try not to press down with the tool while observing the waveform. At the midpoint position the end-of-frame portion of the waveform should be relatively flat and look something like Figure 6-4.

5. Adjust the P2 post, using the same procedure as for P3. Figure 6-7 shows the beginning-of-frame distortion you might see at the extreme adjustment positions. After adjusting to the midpoint position the beginning-of-frame portion should be relatively flat and look something like Figure 6-4. In cases of extreme initial misadjustment it may be beneficial to repeat the adjustment of the P3 post.

6. Eject the alignment tape and restore software switch settings:
   ```
   stop
   eject
   batch agcon
   ```
   (Instead of running the **agcon** batch file you may simply wish to reboot.)

Alternate "No-Scope" Adjustment Method

It is possible to align the tape guide posts without an oscilloscope using S2 console displays, however this is slightly more difficult. The Matsushita/Panasonic alignment tape cannot be used, you must use the "S2 alignment tape" as described in part A.

1. Place the "S2 alignment tape" in the transport and rewind it. If you need to rewind the tape again in the middle of the procedure do so and then *repeat steps 2 and 3*.

2. Put the transport into manual play:
   ```
   transport N
   play manual
   ```

3. Set gain controls and tracking calibration as required by running the **agc_noscope** batch file, supplied with ROS software version 3.1 and later. If you do not have this batch file, refer to the listings of batch files at the end of this section and type it in. It takes about 45 seconds to complete.
   ```
   batch agc_noscope
   ```

4. Bring up the error distribution histogram with the following command:
   ```
   def syncerrlocate histogram 0 249 cumulative 2
   ```
   Here exponential averaging is used with decay time '2'.

5. Adjust the P3 post. You should turn the post very slowly in small steps, waiting a few seconds between steps for the histogram to settle. Also keep an eye on the estimated error rate value "Est err rate" in the Crucial Parms window. Do not press down with the tool while observing the histogram. Turn the post clockwise until the end-of-frame portion of the histogram display shows significant errors (see Figures 6-9 and 6-10). The estimated error rate should be around 2e-4 or higher for LP, or around 2e-3 for SLP. Note the position of the P3 post. Then, turn the post counter-clockwise until the histogram first improves and then shows errors to a similar degree. Note the position of the P3 post, and turn the post clockwise again to the midpoint of the previous two adjustment positions. At the midpoint position the end-of-frame portion of the histogram should have relatively few errors with no significant bumps or peaks, something like Figure 6-8.

6. Adjust the P2 post, using the same procedure as for P3. Figure 6-11 shows the beginning-of-frame errors you might see at the extreme adjustment positions. After adjusting to the midpoint position the beginning-of-frame portion should have relatively few errors with no significant bumps or peaks, something like Figure 6-8. In cases of extreme initial misadjustment it may be beneficial to repeat the adjustment of the P3 post.

7. Type 'q' to quit the error histogram display, then eject the tape and restore software switch settings:
   ```
   stop
   eject
   batch agcon
   ```
   (Instead of running the **agcon** batch file you may simply wish to reboot.)

Part C: Re-evaluate Performance

1. Reboot the S2 to clear any previous diagnostic switch settings:
   ```
   shutdown reboot
   ```

2. Re-insert the "S2 alignment tape" from part A and initiate manual playback, rewinding first if necessary:

```
S2 Recorder |      User Parms (playback)   | RT 407    1997  127-21:15:42
 \/ Operating|                              |
    System   | Tape ID: undefined           | Recorder mode : 16x8-1 (br)
-------------|                              | Station delay : [no 1 Hz in]
 Transports  | Info field 1: undefined      | Decoder input : channel 1 play
0  no tape   | Info field 2: undefined      | Decoded time  : 100-14:30:01.30
1*14:30:01.30| Info field 3: undefined      | Decoder sync  : 100% (cal hold)
2  no tape   |                              |
3  no tape   | User data valid: yes         | Est err rate : 4.4e-5
4  no tape   | Tape speed: lp               | % data valid : 100.00%
5  no tape   | Tape position: 0:12:34 +-00 s| Decoder Xbar : 01234567
6  no tape   |------------------------------|--------------------------------
7  no tape   |  <<   ▶ M ▶▶   []   ||   REC |Tape chan ID does not match setup
-------------------------------------------|Tapes playing but not aligned
Errors in Blocks    0 - 249 :    12
   Scale: 6         Flatness: 0.002855      --------------------------------

         :
         :                                                       :
 :       :                    ::        :  :::          :
--------------------------------------------------------------------------
[     ] [      ] [      ] [      ] [      ] [      ] [      ] [         ]
```
Figure 6-8: Proper tape guide adjustment, no scope

```
S2 Recorder |      User Parms (playback)   | RT 407    1997  127-21:36:02
 /\ Operating|                              |
    System   | Tape ID: undefined           | Recorder mode : 16x8-1 (br)
-------------|                              | Station delay : [no 1 Hz in]
 Transports  | Info field 1: undefined      | Decoder input : channel 1 play
0  no tape   | Info field 2: undefined      | Decoded time  : 100-14:31:28.28
1*14:31:28.28| Info field 3: undefined      | Decoder sync  : 100% (cal hold)
2  no tape   |                              |
3  no tape   | User data valid: yes         | Est err rate : 1.1e-4
4  no tape   | Tape speed: lp               | % data valid : 99.99%
5  no tape   | Tape position: 0:14:00 +-00 s| Decoder Xbar : 01234567
6  no tape   |------------------------------|--------------------------------
7  no tape   |  <<   ▶ M ▶▶   []   ||   REC |Tape chan ID does not match setup
-------------------------------------------|Tapes playing but not aligned
Errors in Blocks    0 - 249 :    31
   Scale: 17        Flatness: 0.006398      --------------------------------
                                                                         :
                                                                         :
                                                                         :
                                                                        ::
                                    :                                   ::
--------------------------------------------------------------------------
[     ] [      ] [      ] [      ] [      ] [      ] [      ] [         ]
```
Figure 6-9: End-of-frame distortion example1, no scope

```
S2 Recorder |      User Parms (playback)   | RT 407    1997  127-21:41:27
 /\ Operating|                              |
    System   | Tape ID: undefined           | Recorder mode : 16x8-1 (br)
-------------|                              | Station delay : [no 1 Hz in]
 Transports  | Info field 1: undefined      | Decoder input : channel 1 play
0  no tape   | Info field 2: undefined      | Decoded time  : 100-14:29:27.44
1*14:29:27.44| Info field 3: undefined      | Decoder sync  : 100% (cal hold)
2  no tape   |                              |
3  no tape   | User data valid: yes         | Est err rate : 3.8e-4
4  no tape   | Tape speed: lp               | % data valid : 99.85%
5  no tape   | Tape position: 0:12:00 +-00 s| Decoder Xbar : 01234567
6  no tape   |------------------------------|--------------------------------
7  no tape   |  <<   ▶ M ▶▶   []   ||   REC |Tape chan ID does not match setup
-------------------------------------------|Tapes playing but not aligned
Errors in Blocks    0 - 249 :    129
   Scale: 22        Flatness: 0.1095        --------------------------------
                                                :
                                                :             :
                                         :     :   :
                                         ::  :: : :
                                   :: ::: :::: :::: :::::::::::: :    ::
--------------------------------------------------------------------------
[     ] [      ] [      ] [      ] [      ] [      ] [      ] [         ]
```
Figure 6-10: End-of-frame distortion example2, no scope

```
S2 Recorder |      User Parms (playback)   | RT 407    1997  127-21:47:37
 /\ Operating|                              |
    System   | Tape ID:                     | Recorder mode : 16x8-1 (br)
-------------|                              | Station delay : [no 1 Hz in]
 Transports  |                              | Decoder input : channel 1 play
0  no tape   |                              | Decoded time  : XXX-XX:XX:XX.XX
1*14:30:54.62|                              | Decoder sync  : 96% (cal hold)
2  no tape   |                              |
3  no tape   | User data valid:             | Est err rate : 5.4e-3
4  no tape   | Tape speed: lp               | % data valid : 95.45%
5  no tape   | Tape position: 0:13:26 +-00 s| Decoder Xbar : 01234567
6  no tape   |------------------------------|--------------------------------
7  no tape   |  <<   ▶ M ▶▶   []   ||   REC |Tapes playing but not aligned
-------------------------------------------|
Errors in Blocks    0 - 249 :    1300
   Scale: 400       Flatness: 6.299         --------------------------------
:
::
::
::::
:::::
::::::
--------------------------------------------------------------------------
[     ] [      ] [      ] [      ] [      ] [      ] [      ] [         ]
```
Figure 6-11: Beginning-of-frame dist. example, no scope

```
transport N
rewind
play manual
align selfalign
```

3. Wait for data-recovery and tracking calibration to complete (about 45 seconds)
    ```
    wait drcal
    ```

4. Measure bit-error rate:
    ```
    setup transport N
    form berc 0 30 s
    ```
    Write down the average bit-error rate obtained.

5. Measure the flatness profile:
    ```
    def syncerrlocate enable
    def syncerrlocate histogram 0 249 30 s cumulative
    ```
    Write down the flatness and scale values obtained. Compare these values, as well as the bit-error rate, to the values obtained in part A. In all cases lower numbers are ''better''.

6. Stop and eject the tape:
    ```
    stop
    eject
    ```

To verify overall system performance we recommend you repeat the post-installation checkout record/playback test (Section 2.8).

Required Batch Files

The following three batch files are used for tape guide adjustment:
    agcoff agcon agc_noscope
These come standard with ROS software version 3.1 and later, but users with older software can easily type them

in. This can be done using the `batch edit` command. Alternatively they may simply be typed line for line wherever the batch file is called.

- agcoff
  ```
  # Set conditions for tape guide post adjustment (scope method)
  transport  tracking  manual  phase2
  transport  tracking  auto  phase1
  dr  cal  all  hold
  transport  tracking  recalibrate
  transport  agc  off
  dr  agc  all  disable
  dr  agc  all  gainset  25
  ```

- agcon
  ```
  # Undo settings made by agoff or agc_noscope
  transport  all  tracking  auto
  transport  all  agc  on
  dr  agc  all  enable
  dr  cal  all  continue
  def  syncerrlocate  disable
  ```

- agc_noscope
  ```
  # Set conditions for tape guide post adjustment (no-scope method)
  dr  agc  all  enable
  dr  cal  all  continue
  transport  agc  on
  transport  tracking  auto
  transport  tracking  recalibrate
  def  syncerrlocate  enable
  # Wait while tracking recalibrates
  wait  drcal
  dr  cal  all  hold
  ```

8. S2 SPECIFICATIONS SUMMARY

Operation

Format:	Non-data-replacement
User data rates:	16 to 128 Mbits/s (16 Mbits/s per transport)
LP Bit-error rate:	1×10^{-4} (maximum), 1×10^{-5} (typical)
SLP Bit-error rate[†]:	1×10^{-3} (maximum), 2×10^{-4} (typical)

Tape: SVHS Cobalt Iron Oxide, approved types only

Record/Play Time
 at 128 Mbits/s: 3 hrs 50 min (ST-120 tape, LP)
4 hrs 1 min (ST-126/SE-180 tape, LP)
6 hrs 2 min (ST-126/SE-180 tape, SLP)
5 hrs 47 min (ST-182/SE-260 tape, LP)
8 hrs 41 min (ST-182/SE-260 tape, SLP)

Total Data Capacity: 1.85 Tbits (ST-126 tape, LP)
3.94 Tbits (ST-182 tape, SLP)

User Data Density: 675 Tbits/m^3 (ST-126 tape, LP)
1440 Tbits/m^3 (ST-182 tape, SLP)

Data Shipping Weight: 1.1 Kg/Tbit (ST-126 tape, LP)
0.6 Kg/Tbit (ST-182 tape, SLP)

Timebase Deviation
 Tolerance: Max. delay rate of external reference clock: $\pm 10^{-3}$ s/s
 (corresponds to input frequency deviation of $\pm 0.1\%$)
Max. delay acceleration of external ref. clock: $\pm 10^{-6}$ s/s^2
Max. delay step at 0 delay rate: $\pm 10^{-5}$ s

Physical

Configuration: 1 or 2 Transport Array Modules (TAMs)
1 Data, Signal and Control Module (DSCM)
19" rack-mount or table-top operation

Dimensions in cm: TAM: 40 H x 44 W x 46 D (10 U)
DSCM: 33 H x 44 W x 33 D (7 U)

Weights: 1 TAM: 36 Kg
DSCM: 17 Kg

Power Consumption: 1 TAM: 140 W
DSCM: 230 W

† Averaged across eight transports

8. S2 SPECIFICATIONS SUMMARY

Operation Format: Non-data-replacement
       User data rates: 16 to 128 Mbits/s (16 Mbits/s per transport)
       LP Bit-error rate: 1×10^{-4} (maximum), 1×10^{-5} (typical)
       SLP Bit-error rate[†]: 1×10^{-3} (maximum), 2×10^{-4} (typical)

       Tape: SVHS Cobalt Iron Oxide, approved types only

       Record/Play Time
         at 128 Mbits/s: 3 hrs 50 min (ST-120 tape, LP)
                  4 hrs 1 min (ST-126/SE-180 tape, LP)
                  6 hrs 2 min (ST-126/SE-180 tape, SLP)
                  5 hrs 47 min (ST-182/SE-260 tape, LP)
                  8 hrs 41 min (ST-182/SE-260 tape, SLP)

       Total Data Capacity: 1.85 Tbits (ST-126 tape, LP)
                  3.94 Tbits (ST-182 tape, SLP)

       User Data Density: 675 Tbits/m^3 (ST-126 tape, LP)
                  1440 Tbits/m^3 (ST-182 tape, SLP)

       Data Shipping Weight: 1.1 Kg/Tbit (ST-126 tape, LP)
                  0.6 Kg/Tbit (ST-182 tape, SLP)

       Timebase Deviation
         Tolerance: Max. delay rate of external reference clock: $\pm 10^{-3}$ s/s
                  (corresponds to input frequency deviation of $\pm 0.1\%$)
                  Max. delay acceleration of external ref. clock: $\pm 10^{-6}$ s/s^2
                  Max. delay step at 0 delay rate: $\pm 10^{-5}$ s

Physical Configuration: 1 or 2 Transport Array Modules (TAMs)
                  1 Data, Signal and Control Module (DSCM)
                  19" rack-mount or table-top operation

       Dimensions in cm: TAM: 40 H x 44 W x 46 D   (10 U)
                  DSCM: 33 H x 44 W x 33 D   (7 U)

       Weights: 1 TAM: 36 Kg
                  DSCM: 17 Kg

       Power Consumption: 1 TAM: 140 W
                  DSCM: 230 W

[†] Averaged across eight transports

APPENDIX A

S2 RCL SERIAL COMMUNICATIONS PROTOCOL

A.1 Overview

This section describes the communications protocol of the S2 Radioastronomy Control Link (RCL), a serial communications link used for computerized control of the S2 recorder, and other devices such as the S2 Data Acquisition System. Here and in the rest of this section the word "recorder" refers to both S2 Record Terminals (RTs) and Playback Terminals (PTs). These specifications are subject to change at any time. All numbers are decimal unless preceded by 0x which indicates hexadecimal.

The S2 RCL port is a male DB25 connector wired as an RS-232C DTE (Data Terminal Equipment) device. It should be connected to a DCE (Data Communication Equipment) device (this is the opposite of the console port). Some computers, such as IBM PCs, have only DTE serial ports and will require a null modem cable which exchanges the receive and transmit data lines (pins 2 & 3). The normal RCL communications parameters are 19200 baud, no parity, 8 data bits, 1 stop bit. On S2 systems that have an Ethernet connection equivalent RCL access can be performed over the network using Internet TCP sockets (port number 1025).

The RCL uses a binary request-response packet protocol. In this scheme, the external control computer issues requests using a specific command format, and waits for a response from the S2. Only one request may be active at any time, and for each request there is exactly one response. Responses are immediate or almost immediate. Those operations for which there would be a significant delay return an immediate acknowledgement and then indicate their progress in the S2 system status information (STATUS command). Each S2 or other RCL device is assigned an address for communication, and each RCL packet contains a destination address, allowing multiple devices to be connected together and controlled simultaneously. When there is more than one non-networked RCL device, RS-485 multi-drop serial connections may be used. In this case the S2 requires an external RS-232C to RS-485 conversion box. This box should switch on its RS-485 tri-state line drivers only when the RTS (Request-To-Send) signal from the S2 is high (TRUE). The S2 sets the RTS signal high just before each RCL message is transmitted, and then sets RTS low (FALSE) following the transmission.

Please note that CRESTech/SGL will provide the source code required to implement packet assembly/disassembly for all RCL commands and responses described here to all S2 users. This comes in the form of an interface library written in ANSI C. Therefore it is not necessary for users to understand or implement the detailed packet protocol, only how to use the individual RCL commands.

A.2 General Packet Format

SOT	pkt len		addr	cmd/resp.code	seq#	...data...	checksum		EOT
	MSB	LSB					MSB	LSB	

bits	8	16	8	8	8		16	8

SOT　　　　　"Start of Transmission", 01 decimal. Defines the start of the packet. When a byte with the same code as the SOT character appears elsewhere in the message body, it is repeated to distinguish it from a true SOT. When the receiver detects two consecutive SOTs it interprets

them as a single byte with value 1. If a single SOT is detected in the middle of an incoming packet then a new packet has started and previously received bytes are discarded.

pkt len — The total number of bytes in the packet including everything except the SOT and EOT characters, offset by adding 505. Each packet is at least 7 bytes long, so the minimum length code value is 512. Doubled SOT data bytes within the message body are counted as 1 byte, not 2. Length is a 16-bit (2-byte) quantity with the most significant byte (MSB) transmitted first. The offset of 505 is added to ensure that the MSB of the length cannot be an SOT. Lengths between 0 and 511 (0 and 0x01FF) are illegal and will cause the receiver to discard the packet. The maximum packet length is artificially set at 2054 bytes (before offsetting) to limit transmission times and buffer sizes.

addr — The address of the device for which the packet is intended. When using RS-485 multi-drop operation with more than one RCL device controlled by a single master, each device must be assigned a unique address from 0 to 253. If there is only one RCL device, regular RS-232C can be used and the address assigned to the device should be 0. Address 254 (RCL_ADDR_MASTER) indicates the controlling computer or master. Address 255 (RCL_ADDR_BROADCAST) is a special broadcast code intended for diagnostic and configuration use only. All RCL slave devices will respond to code 255.

cmd/resp — A command or response code, as described in Section A.4 below. Codes in the range 0 to 99 decimal are for commands sent from the external control computer (master) to the S2 (slave), and codes 100 to 199 decimal are for responses sent from the S2 to the external computer. 255 is an expansion code to allow the number of commands to exceed the 8-bit limit if necessary. The remaining codes are reserved. Generic RCL commands and responses should use the same code numbers in all types of RCL devices, but commands which are specific to a particular device type may use any legal code numbers.

seq# — An 8-bit sequence number that is incremented for every new command sent by the external computer to the S2 recorder. If the external computer re-transmits a command because no response was detected then the re-transmitted packet has the same seq# as the previous (identical) packet. Whenever the S2 receives a command with the same sequence number as the previous command it repeats the last response without re-executing the command. Responses use the same sequence number as the command which the response is for. Example:

1. External computer $\longrightarrow$ S2: "stop" command   seq# 101
   (...S2 executes "stop"...)

2. S2 $\longrightarrow$ External computer: "ERR_NONE" response   seq# 101

3. External computer $\longrightarrow$ S2: "rewind" command   seq# 102
   (...S2 executes "rewind"...)

4. S2 $\longrightarrow$ External computer: "ERR_NONE" response   seq# 102
   (...response gets garbled or lost in transmission...)
   (...external computer times out...)

5. External computer $\longrightarrow$ S2: "rewind" command   seq# 102

6. S2 $\longrightarrow$ External computer: "ERR_NONE" response   seq# 102

data — Some number of data bytes. The theoretical maximum number of data bytes that may be transmitted is 65535-512 = 65023, however in practice packets are never that long because transmission time would be excessive (over 30 sec at 19200 baud) and too much memory would be required for buffering. In practice we limit the data portion to 2047 bytes, and the entire packet to 2054 bytes (not including SOT and EOT).

chksum — A 16-bit unsigned binary checksum (MSB first). Calculated by taking the 16 least-significant bits of the sum of all bytes except: SOT, EOT, and the checksum itself. Doubled SOT bytes in the message body are counted only once, not twice. Received packets whose checksum does not match the calculated checksum are discarded.

EOT — "End of Transmission", 04 decimal. Defines the end of the packet. The receiver only checks to see if it is there, since the length field determines how many bytes are in the packet. If EOT is not found at the end of the packet then the entire packet is discarded.

The external control computer must use timeouts when waiting for replies from the S2. Incorrect packets received by either end are always discarded, causing the external computer (master) to time out and re-send the command. The repeated packet must use the same sequence number as the previous packet. After a certain number of retries (2 recommended, for a total of 3 failed transmissions), the external computer should flag a communication error and declare that the S2 recorder is "dead" (not functioning/power off/RCL cable disconnected). The external computer can use 'ping' commands (described below) to detect when communications resume.

A.3   Command/Response Summary

Commands are numbered in the range 0 to 99 decimal. They are always sent from the external control computer (master) to the RCL device (slave). Most commands listed below are specific to the S2 recorder, but a few of the commands are generic and must be implemented in the same way by all types of RCL devices. Examples of generic RCL commands: PING, VERSION, and IDENT. Except for the generic commands, other types of RCL devices can have different command sets by defining the same command code numbers differently.

Code	Cmd Name	Function
00	STOP	Stop transports.
01	PLAY	Start playback (use STATUS to monitor progress).
02	RECORD	Start recording (use STATUS to monitor progress).
03	REWIND	Rewind tapes (use 'state_read' to test completion, recorder will auto-stop).
04	FF	Fast-forward tapes (use 'state_read' to test completion, recorder will auto-stop).
05	PAUSE	Temporarily pause recording or playback (not recommended, use STOP).
06	UNPAUSE	Resume recording or playback after pause.
07	EJECT	Eject tapes.
08	STATE_READ	Read current tape motion state (stop, record, play, rewind, ff, no_tape etc).
09	SPEED_SET	Set record tape speed lp or slp (lp not supported on AG2550 transports).
10	SPEED_READ	Read record tape speed.
11	SPEED_READ_PB	Read playback tape speed.
12	TIME_SET	Set system time; must be sent right after S-1/C-1 Hz input tick (S1HZ_REC on C1 cable or C1HZ on C2a cable) if SC PLL locked to refclk.
13	TIME_READ	Read current system time; reply returned right after next S-1/C-1 Hz input tick.
14	TIME_READ_PB	Read playback tape time; reply returned right after next S-1 Hz output tick (S1HZ_PB on C2a cable). Tapes must be aligned. This command can be used to test if tapes are aligned, or use STATUS.
15	MODE_SET	Set recorder mode (S2 mode string, or c1test16 for C1 cable test mode (test sequence check), c2test16 for C2 cable test mode (test sequence generate).
16	MODE_READ	Read current recorder mode.
17	TAPEID_SET	Set Tape ID (record), ASCII string up to 20 chars.
18	TAPEID_READ	Read Tape ID (record).
19	TAPEID_READ_PB	Read playback Tape ID. Value must be consistent across all tapes.
20	USER_INFO_SET	Set user-defined auxiliary data fields & labels (record) Field 1: ASCII or binary string up to 16 chars Field 2: ASCII or binary string up to 16 chars Field 3: ASCII or binary string up to 32 chars Field 4: ASCII or binary string up to 48 chars All 4 fields: 16 character ASCII user-defined label
21	USER_INFO_READ	Read user-defined auxiliary data fields & labels (record).
22	USER_INFO_READ_PB	Read playback user-defined aux data fields & labels. Values must be consistent across all tapes.
23	USER_DV_SET	Set user data-valid flag, boolean (record).

(continued on next page)

Code	Cmd Name	Function (continued)
24	USER_DV_READ	Read user data-valid flag (record).
25	USER_DV_READ_PB	Read user data-valid flag. Value must be consistent across all tapes.
26	GROUP_SET	Set transport group number.
27	GROUP_READ	Read transport group number (also gives number of groups).
28	TAPEINFO_READ_PB	Return tape-related information during playback for all 8 individual transports.
29	DELAY_SET	Set the station delay offset.
30	DELAY_READ	Read the current station delay offset setting.
31	DELAYM_READ	Read the current station delay offset measurement.
32	BARRELROLL_SET	Turn barrel roll on or off.
33	BARRELROLL_READ	Read the current barrel roll setting.
34	ALIGN	Align tapes (pb) (use STATUS to test completion).
35	POSITION_SET	Position tapes (use STATE_READ to test completion).
36	POSITION_READ	Read the current tape position.
37	ERRMES	Supply error measurement for playback rate control in SCPLL mode **errmes**.
38	ESTERR_READ	Read estimated error rates for all transports.
39	PDV_READ	Read % data valid values for all transports.
42	SCPLL_MODE_SET	Set the System Clock PLL mode.
43	SCPLL_MODE_READ	Read the current System Clock PLL mode.
44	TAPETYPE_SET	Set the tape type.
45	TAPETYPE_READ	Read the current tape type setting.
50	MK3_FORM_SET	Enable/disable Mark III formatter and Mark III output on C2a cable.
51	MK3_FORM_READ	Read current Mark III formatter enable/disable state.
55	TRANSPORT_TIMES	For all 8 transports read head-use time, total power-on time etc.
60	STATION_INFO_READ	Read station-related information assigned to S2 system including station number.
70	CONSOLECMD	Execute an arbitrary console command (caution required!). This is a generic cmd.
71	POSTIME_READ	Returns both the playback tape time and tape position for a particular transport. For SGL internal use only.
80	STATUS	Return brief status report, suitable for polling. Contains overall "error" summary bit and overall "fatal error" summary bit. Also contains count of active status conditions and list of status code numbers in approximate order of priority. Each status code includes "error", "fatal", and "clear-on-read" type indicator bits. Clear-on-read status entries are automatically cleared by the STATUS command. This is a generic command.
81	STATUS_DETAIL	Return detailed text descriptions for one or all active status conditions, suitable for logging. Includes only conditions in effect at the time of the last STATUS command. Optionally, re-reads current conditions (i.e. implicitly performs STATUS). This is a generic command
82	STATUS_DECODE	Translate numeric status code to text message. Suitable for building translation table of status codes and mnemonics/messages. This is a generic command.
83	ERROR_DECODE	Translate negative numeric error code to text message. This is a generic cmd.
90	DIAG	Initiate internal diagnostic self-test 1 (use STATUS to check completion). Note that most other RCL commands which affect tape motion or system switch settings are not allowed during the self-test. This is a generic command.
91	BERDCB	Perform a bit-error rate measurement using one of two test sequences (FORM or UI), or perform a DC-bias measurement on user data.
92	DATA_EXTRACT_READ	Extract and return three bytes of data or validity from a particular user data channel in S2 or Mark-III format.
97	IDENT	Return device-type information string. This is a generic command
98	PING	Perform no action, just return an "ERR_NONE" response. Useful for polling a dead S2 to tell when it comes alive. This is a generic command.
99	VERSION	Return ROS software version information string. This is a generic command.

Responses are numbered from 100 to 199 decimal, and are usually derived by adding 100 to the request command code. Where more than one request returns the same response type, the lowest numbered request has been used to derive the response code. Responses are always sent from the S2 recorder (slave) to the external control computer. Similar to generic commands, certain responses are generic and should be implemented in the same way by all types of RCL devices.

Code	Resp Name	Function
100	RESP_ERR	General error condition/OK response, contains error code, 0 if no error. Use ERROR_DECODE command to translate error code into descriptive text message. This is a generic response.
108	RESP_STATE	Tape motion state
110	RESP_SPEED	Transport record or playback tape speed
113	RESP_TIME	System UTC time or playback tape time
116	RESP_MODE	Recorder mode (record or playback)
118	RESP_TAPEID	Tape ID (record or playback)
121	RESP_USER_INFO	User-defined auxiliary data fields (record or playback)
124	RESP_USER_DV	User data-valid flag (record or playback)
127	RESP_GROUP	current transport group number (record or playback)
128	RESP_TAPEINFO	tape-related information for all 8 individual transports (playback only)
130	RESP_DELAY	Station delay (setting or measurement)
133	RESP_BARRELROLL	Barrel roll setting
136	RESP_POSITION	Tape position
138	RESP_ESTERR	Estimated error-rate from all transports
139	RESP_PDV	Percent data valid from all transports
143	RESP_SCPLL_MODE	Current System Clock PLL mode
145	RESP_TAPETYPE	Current tape type setting
151	RESP_MK3_FORM	Current Mark III formatter enable/disable state
155	RESP_TRANSPORT_TIMES	Head-use and in-service times from all transports
160	RESP_STATION_INFO	Station number, system serial number, nickname
171	RESP_POSTIME	Playback tape time and tape position. For SGL internal use only.
180	RESP_STATUS	System status summary. This is a generic response.
181	RESP_STATUS_DETAIL	Detailed system status. This is a generic response.
182	RESP_STATUS_DECODE	Translated text status message. This is a generic response.
183	RESP_ERR_DECODE	Translated text error message. This is a generic response.
191	RESP_BERDCB	Bit-error rate or DC-bias measurement result
192	RESP_DATA_EXTRACT	3-byte data extract or validity extract result
197	RESP_IDENT	RCL device type string. This is a generic response.
199	RESP_VERSION	ROS software version information. This is a generic response.

A.4   Command/Response Formats

This section contains detailed format descriptions of each command and response. Each format description has the following entries:

Command: The name of the command or response. The entry is labeled "Response" for responses. The word "generic" is included in brackets if this is a generic command or response.

Msg Code: The hex command/response code.

Data: The contents of the packet data field.

Expected Resp: (commands only) The proper response if the command is successful (unsuccessful commands return RESP_ERR).

Timer Duration: (commands only) The time in which the sender of the message should get a reply. If a reply is not received after this duration then the message should be re-sent. This information is for setting timeout intervals only — in practice most commands will complete much faster than the time indicated. The RCL Interface Library incorporates the timer duration values documented here and performs timeouts and retries as required. Therefore users calling the RCL Interface Library functions (in rcl_cmd.c) do not usually need to retry commands. The *typical* response time when calling RCL Interface Library functions will be less than the timer duration value, but the *maximum* response time is 3 times the timer duration value since up to 2 retries are automatically performed for most commands. The following commands are special and will not be automatically retried, so their maximum response time is the timer duration value:

rcl_ping()
rcl_time_set()
rcl_time_read()
rcl_time_read_pb()
rcl_align_abs()

User software should perform retries for these commands as needed.

Eq Console Cmd: (commands only) The equivalent console command used to manually implement the same function.

Description: The purpose and function of the command or response.

Commands

0) Command: STOP

Msg Code: 00

Data: none

Expected Resp: RESP_ERR [ERR_NONE]

Timer Duration: 1.5 sec

Eq Console Cmd: stop

Description: Stops the transports and/or associated system operation. This includes stopping playback, recording, rewinding, fast-forwarding, as well as alignment and positioning.

1) Command: PLAY

Msg Code: 01

Data: none

Expected Resp: RESP_ERR [ERR_NONE]

Timer Duration: 1.5 sec

Eq Console Cmd: play

Description: Starts playback. Automatically initiates a tape alignment operation (self-align) to align the individual transports to each other and to obtain a single, coherent tape time. After reading this tape time, the user may issue an RCL ALIGN command to further align the recorder to a desired epoch. The S2 must be currently stopped, rewinding, or fast-forwarding, otherwise the PLAY command is not allowed. Also must not currently be positioning. The PLAY command is normally issued after the recorder has been positioned using PO-

SITION_SET. In an S2-RT, playback is restricted to 16 Mbit/s total bit rate (out of either 1, 2 or 4 channels) instead of the usual 128 Mbit/s.

2) Command: RECORD
Msg Code: 02
Data: none
Expected Resp: RESP_ERR [ERR_NONE]
Timer Duration: 1.5 sec
Eq Console Cmd: record
Description: Starts recording. Recorder must currently be stopped.

3) Command: REWIND
Msg Code: 03
Data: none
Expected Resp: RESP_ERR [ERR_NONE]
Timer Duration: 1.5 sec
Eq Console Cmd: rewind
Description: Rewinds the tapes. Stops the recorder first if necessary (except when recording). Recorder must be in stop, play, play-pause, or fast-forward state, otherwise REWIND is not allowed. Also must not currently be aligning or positioning.

4) Command: FF
Msg Code: 04
Data: none
Expected Resp: RESP_ERR [ERR_NONE]
Timer Duration: 1.5 sec
Eq Console Cmd: ff
Description: Fast-forward winds the tapes. Stops the recorder first if necessary (except when recording). Recorder must be in stop, play, play-pause, or rewind state, otherwise FF is not allowed. Also must not currently be aligning or positioning.

5) Command: PAUSE
Msg Code: 05
Data: none
Expected Resp: RESP_ERR [ERR_NONE]
Timer Duration: 1.5 sec
Eq Console Cmd: pause
Description: Puts the S2 recorder into play-pause or record-pause state. Recorder must currently be recording or playing (and must not be aligning or positioning). This differs from STOP in that PAUSE is sometimes able to re-enter play or record more quickly. Unlike STOP, time spent in PAUSE counts against head lifetime. To prevent excessive head and tape wear, the recorder automatically stops after pausing for more than about 3 minutes. For normal record or playback operation, PAUSE provides no significant benefits and should not be used — use STOP instead.

6) Command: UNPAUSE
Msg Code: 06
Data: none
Expected Resp: RESP_ERR [ERR_NONE]
Timer Duration: 1.5 sec

Eq Console Cmd: unpause

    Description: Un-pauses the S2 recorder after a PAUSE command and resumes playback or recording, as the case may be. The recorder must currently be in play-pause or record-pause state.

7)   Command: EJECT

    Msg Code: 07

    Data: none

    Expected Resp: RESP_ERR [ERR_NONE]

    Timer Duration: 1.5 sec

Eq Console Cmd: eject

    Description: Ejects the tapes from all currently selected transports. Stops the S2 recorder first if it is not already stopped. The recorder must not be recording or record-paused. The recorder state changes to STATE_NOTAPE.

8)   Command: STATE_READ

    Msg Code: 08

    Data: none

    Expected Resp: RESP_STATE

    Timer Duration: 0.5 sec

Eq Console Cmd: (none, state displayed on console screen)

    Description: Reads the current recorder state, which refers to the status of the recorder with respect to tape motion (e.g. stop, record, play, rewind, ff, notape etc.)

9)   Command: SPEED_SET

    Msg Code: 09

    Data: byte1: Speed code
            0x01 LP
            0x02 SLP
    (total data bytes = 1)

    Expected Resp: RESP_ERR [ERR_NONE]

    Timer Duration: 1.5 sec

Eq Console Cmd: speed {lp|slp}

    Description: Sets the record tape speed. On systems with AG2530 and earlier model transports the two allowed speeds are LP (long play) and SLP (super-long play). Systems with AG2550 and later transports support recording in SLP speed only, although they will play back tapes recorded in LP. SLP is sometimes also called EP. The speed SP is not used. Recording times are 50% longer in SLP than LP, with some degradation in bit error-rate. The SPEED_SET command is not allowed during record. It is not possible to set the playback speed (playback speed is automatically determined).

10)   Command: SPEED_READ

    Msg Code: 10

    Data: none

    Expected Resp: RESP_SPEED

    Timer Duration: 0.5 sec

Eq Console Cmd: speed (also displayed on console screen)

    Description: Reads the currently selected record tape speed, either LP or SLP. If the S2 is not currently recording, this is the speed that would be used if a

RECORD command was sent. The SPEED_READ command may be sent at any time.

11)   Command: SPEED_READ_PB

        Msg Code: 11

             Data: none

   Expected Resp: RESP_SPEED

  Timer Duration: 0.5 sec

 Eq Console Cmd: (none, playback speed displayed on console screen)

      Description: Reads the current playback tape speed. This may be LP, SLP, SP, or 'unknown' (see RESP_SPEED). The S2 must currently be playing back tapes. The unused speed SP may appear for blank tapes or when tapes were not recorded by an S2. An 'unknown' response may appear when playback speed has not yet been determined or when tapes recorded at different speeds are played at the same time.

12)   Command: TIME_SET

        Msg Code: 12

             Data: byte1, byte2: Absolute UT year number (1901–2099), byte1=MSB
                    byte3, byte4: UT day number (1 to 365/366), byte3 = MSB
                    byte5: UT hours (0 to 23)
                    byte6: UT minutes (0 to 59)
                    byte7: UT seconds (0 to 59)
                    (total data bytes = 7)

   Expected Resp: RESP_ERR [ERR_NONE]

  Timer Duration: 0.5 sec

 Eq Console Cmd: time {[NNN-]NN:NN:NN|{s|m|h|d|y}[+|-]NUM|validate}

      Description: Sets the S2 system time, which is the time encoded onto tape during record. It is also used for other miscellaneous functions such as time-tagging internal log entries and implementing absolute time delays in batch files. If the System Clock PLL is in **refclk** mode, then the TIME_SET command *must* be synchronized to the S-1/C-1 Hz input tick (S1HZ_REC on C1 cable or C1HZ on C2a cable) by sending it immediately following a particular tick. The data field should contain the time of the tick that just passed. This protocol is necessary because it is not possible to communicate exact times using the RCL alone. With the SC PLL in **refclk** mode, the station delay must be set to 0 during time setting and the external S-1/C-1 Hz input signal must be present otherwise an error is returned. In other SC PLL modes no errors are returned since only **refclk** mode can be used for recording, and for anything else the system time only needs to be approximate.
                       Note: Unlike most other commands, TIME_SET is not subject to automatic communications retries within the RCL interface library, otherwise the wrong time might be set. The rcl_time_set() function call should be re-issued with an updated time if a communications timeout occurs.

13)   Command: TIME_READ

        Msg Code: 13

             Data: none

   Expected Resp: RESP_TIME

  Timer Duration: 1.25 sec

 Eq Console Cmd: time   (also displayed on console screen)

      Description: Reads the current S2 system time (this is not the playback tape time, see next command). If the external S-1 Hz or C-1 Hz input is being used

(S1HZ_REC on C1 cable or C1HZ on C2a cable is connected and System Clock PLL mode is **refclk** or **1hz**), then the response (RESP_TIME) is sent immediately following the next S-1/C-1 Hz *input* tick, and gives the time of the tick that just passed. If the 1 Hz input tick is not present then the error ERR_NOU1HZ will be returned. This protocol is necessary because it is not possible to communicate exact times using the RCL alone. Note that if the S2 station delay is non-zero it will not be reflected in the returned time, even though it affects the time encoded on tape. This is done on purpose to make time reading symmetrical with time setting. The external control computer should obtain the station delay using DELAYM_READ and add it to the time returned by TIME_READ to obtain the correct S2 system time with respect to the input S-1/C-1 Hz.

In System Clock PLL modes other than **refclk** and **1hz**, the RESP_TIME response is sent immediately following the next S-1 Hz *output* tick (S1HZ_PB on C2a cable) since the 1 Hz input is not used. The response gives the time of the tick that just passed, and should not be adjusted by adding the station delay.

The TIME_READ response includes a "time validated" flag. This flag is always FALSE following system bootup (and certain severe system timing glitches), and is set to TRUE following a TIME_SET command. Thus it gives an indication of whether the time is likely to be correct.

Note: Unlike most other commands, TIME_READ is not subject to automatic communications retries within the RCL interface library, otherwise the time returned would be uncertain. The rcl_time_read() function call should be re-issued if a communications timeout occurs.

14)	Command:	TIME_READ_PB
	Msg Code:	14
	Data:	none
	Expected Resp:	RESP_TIME
	Timer Duration:	1.25 sec
	Eq Console Cmd:	(none, tape time displayed on console screen)
	Description:	Reads the playback tape time. The S2 must currently be playing back properly recorded tapes and the tapes must be aligned. The response (RESP_TIME) is sent immediately following the next S-1 Hz *output* tick (S1HZ_PB on C2a cable), and gives the time of the tick that just passed. The external control computer may require special hardware to measure the S-1 Hz output tick against its own reference. This is necessary because it is not possible to communicate exact times using the RCL alone[1]. If the tapes are playing but not (yet) aligned, response RESP_ERR with error code ERR_NOALIGN will be returned (should it be necessary, you can use the TAPEINFO_READ_PB command to read the different individual tape times).

The TIME_READ_PB response includes a "time validated" indicator which shows the state of the time validated flag at the time the tapes were recorded. It will be TRUE if the time was set before making the recording.

Note: Unlike most other commands, TIME_READ_PB is not subject to automatic communications retries within the RCL interface library, otherwise the time returned would be uncertain. The rcl_time_read_pb() function call should be re-issued if a communications timeout occurs.

---

[1] *Precise playback tape time reading is a requirement in a PT, but not in an RT when playback capability is used for diagnostic purposes only.*

15) Command: MODE_SET

    Msg Code: 15

        Data: byte1 ... : Mode designator in the form of an ASCII string terminated by a NULL (0) character. See the description for a list of possible modes.
        (total data bytes = length of string + 1)

Expected Resp: RESP_ERR [ERR_NONE]

Timer Duration: 3.0 sec

Eq Console Cmd: mode ...

Description: Sets the recorder mode, which controls data channelization and bit rate (sample rate). The S2 supports up to 16 channels of data at 4, 8, 16, and 32 Mbits/s with a maximum total data rate of 128 Mbits/s. There are 51 modes in total, 12 of which are for diagnostics. The complete list is as follows:

S2 modes:

    4x4-1, 4x8-1, 4x16-1,
    4x4-2, 4x8-2, 4x16-2,
    8x2-1, 8x4-1, 8x8-1, 8x16-1,
    8x2-2, 8x4-2, 8x8-2, 8x16-2,
    16x1-1, 16x2-1, 16x4-1, 16x8-1,
    16x2-2, 16x4-2, 16x8-2,
    32x1-1, 32x2-1, 32x4-1, 32x2-2, 32x4-2,
    4i8-1, 8i4-1, 8i8-1, 16i4-1, 16i8-1,
    4p8-2, 8p8-2, 16p8-2,
    8d16-2, 8n16-2, 16v8-2, 32a4-2, 32b4-2

Test modes:

    c1test4, c1test8, c1test16, c1test32,
    c2test4, c2test8, c2test16, c2test32,
    diag4, diag8, diag16, diag32

Note that mode names always use lower-case letters — the S2 will map any upper-case letters to lower-case if necessary. The mode should be set prior to recording or playback. For playback, any mode can be chosen as long as it has the same total data rate as the mode used for recording, so that the correct number of transports is selected. The precise mode will be set automatically after playback starts. In addition to the numerous regular operational modes, there are several test modes used to verify proper connection of the C1 or C2 cable, and to perform other system diagnostics. Under certain conditions the playback mode may be different from the record mode as specified by the **playmodemap** defaults file parameter. The MODE_SET command is not allowed during playback, and can only be used to change to modes with the same total data rate during record. For any other mode changes, the S2 must be stopped first. See also the GROUP_SET command which should be used after MODE_SET to select the appropriate group of transports to use in low-bandwidth modes.

16) Command: MODE_READ

    Msg Code: 16

        Data: none

Expected Resp: RESP_MODE

Timer Duration: 0.5 sec

Eq Console Cmd: mode (also displayed on console screen)

Description: Reads the currently selected recorder mode. During (auto) playback the correct mode is set automatically based on information encoded on the tape. However, the initial mode in effect when playback starts must have the same total data rate as the mode used to record the tape, otherwise the mode will not be changed and MODE_READ will *continue to read the old mode*. The fatal status message STAT_BADPBTRANSEL will appear. To correct this situation, stop playback, set a mode with the correct total data rate using MODE_SET, then restart playback (you will probably have to reposition). You can determine what mode the tapes were recorded in either by parsing the mode string from the detailed STAT_BADPBTRANSEL status message, or by using the TAPEINFO_READ_PB command.

17) Command: TAPEID_SET

Msg Code: 17

Data: byte1 ... : ASCII Tape ID string terminated by a NULL (0) character, maximum 20 characters (not incl. NULL). Avoid using unprintable characters.
(total data bytes = length of string + 1, max. 21)

Expected Resp: RESP_ERR [ERR_NONE]

Timer Duration: 0.5 sec

Eq Console Cmd: tapeid [STR|blank]

Description: Sets the Tape ID. This information is recorded continuously as part of the S2 auxiliary data (which is the same for all individual tapes in the tape set). At playback, the Tape ID can be recovered at any position on the tape. It should be chosen so that it uniquely identifies the set of tapes being recorded. It may include an experiment number, site code, tape set index number, or other useful information. The format is totally up to the user as long as the overall length does not exceed 20 characters. The Tape ID should be set prior to the start of recording and thereafter should not be changed for the remainder of the tape. Following this rule will make it easy to determine if all tapes belong to the same set at playback, regardless of their current positions. If this rule is not followed, status code STAT_INTAPEID may appear during playback indicating that two transports have different tape IDs, but it will have no adverse effect on system operation.

18) Command: TAPEID_READ

Msg Code: 18

Data: none

Expected Resp: RESP_TAPEID

Timer Duration: 0.5 sec

Eq Console Cmd: tapeid (also displayed on console screen)

Description: Reads the current Tape ID setting from the last TAPEID_SET command. This is the Tape ID to be recorded, not the playback tape ID.

19) Command: TAPEID_READ_PB

Msg Code: 19

Data: none

Expected Resp: RESP_TAPEID

Timer Duration: 0.5 sec

Eq Console Cmd: tapeid (also displayed on console screen)

Description: Reads the playback Tape ID. The S2 must currently be playing back properly recorded tapes and all of them must indicate the same Tape ID value (this should automatically be true if the tapes are aligned, and may or may not be true if they are not). Since the Tape ID can be changed at any time

during record, the value returned by TAPEID_READ_PB reflects its instantaneous value around the current tape time. If the tapes are playing but not "sufficiently" aligned so that all tape IDs are the same, response RESP_ERR with error code ERR_INPBAUX will be returned (should it be necessary, you can use the TAPEINFO_READ_PB command to read the different individual tape IDs). If auxiliary data cannot be extracted from one or more of the tapes, ERR_NOPBAUX will be returned (this error can be expected to occur during tape alignment and just after playback has started).

20) Command: USER_INFO_SET

　　Msg Code: 20

　　　　Data: byte1: Info field number (1 to 4)
　　　　　　　byte2:　Boolean,
　　　　　　　　　　0x00　Set field value
　　　　　　　　　　0x01　Set field label
　　　　　　　byte3 ... : ASCII info field or label string terminated by a NULL (0) character. Avoid using unprintable characters. Maximum field sizes (not incl. NULL) are as follows:
　　　　　　　　　　Field 1: 16 chars
　　　　　　　　　　Field 2: 16 chars
　　　　　　　　　　Field 3: 32 chars
　　　　　　　　　　Field 4: 48 chars
　　　　　　　　　　Label 1-4: 16 chars
　　　　　　　(total data bytes = length of string + 3, max. 51)

　Expected Resp: RESP_ERR [ERR_NONE]

　Timer Duration: 0.5 sec

Eq Console Cmd: user info {1|2|3|4} {STR|blank|label {STR|blank}}

　　Description: Sets the value of one of 4 user-defined information fields or its label. For example, two separate commands could set field 1 label to "Source Name" and field 1 to "3C273". This information is recorded continuously as part of the S2 auxiliary data (which is the same for all tapes) and recovered at playback time. The user info format is totally up to the user as long as the overall length does not exceed the limits given above. The user info fields and labels should be entered prior to the start of recording and thereafter may be changed at any time, but should not be changed more often than every few seconds (otherwise some user info may be lost at playback).

21) Command: USER_INFO_READ

　　Msg Code: 21

　　　　Data: byte1: Info field number (1 to 4)
　　　　　　　byte2: Boolean,
　　　　　　　　　　0x00　Read field value
　　　　　　　　　　0x01　Read field label
　　　　　　　(total data bytes = 2)

　Expected Resp: RESP_USER_INFO

　Timer Duration: 0.5 sec

Eq Console Cmd: user info {1|2|3|4}　(also displayed on console screen)

　　Description: Reads the current setting of one of the user information fields or labels, as set using USER_INFO_SET. This is the user info to be recorded, not the playback user info.

22) Command: USER_INFO_READ_PB

　　Msg Code: 22

Data: byte1: Info field number (1 to 4)
byte2: Boolean,
    0x00   Read field value
    0x01   Read field label
(total data bytes = 2)

Expected Resp: RESP_USER_INFO

Timer Duration: 0.5 sec

Eq Console Cmd: user info {1|2|3|4}   (also displayed on console screen)

Description: Reads one of the playback user info fields or labels. The S2 must currently be playing back properly recorded tapes and all of them must indicate the same user info value (this should automatically be true if the tapes are aligned, and may or may not be true if they are not). Since the user info can be changed at any time during record, the value returned by USER_INFO_READ_PB reflects its instantaneous value around the current tape time. If the tapes are playing but not "sufficiently" aligned so that all user info is the same, response RESP_ERR with error code ERR_INPBAUX will be returned. If auxiliary data cannot be extracted from one or more of the tapes, ERR_NOPBAUX will be returned (this error can be expected to occur during tape alignment and just after playback has started).

23)   Command: USER_DV_SET

Msg Code: 23

Data: byte1: Boolean user DV value (record only),
    0x00   Set record user data-valid FALSE
    0x01   Set record user data-valid TRUE
byte2: Boolean playback-user-DV-enable value (playback only),
    0x00   Ignore user data validity flag during playback
    0x01   Recognize user data validity flag during playback
(total data bytes = 2)

Expected Resp: RESP_ERR [ERR_NONE]

Timer Duration: 0.5 sec

Eq Console Cmd: user dv {yes|no|playback {enable|disable}}

Description: Sets the value of the record user data-valid (DV) flag. This flag is recorded continuously on the S2 auxiliary data channel (which is the same for all tapes) and recovered at playback time, where it is used to invalidate data (if FALSE) by lowering the data validity signals on the C2b/c cable. This facility provides a way to indicate at record time that invalid data is being recorded, e.g. the telescope is slewing or off-source. The user DV value should be set as desired prior to the start of recording and thereafter may be changed at any time, but should not be changed more often than every few seconds. The user DV setting (first data byte) is only relevant during record, and is not used during playback.

The second data byte controls a playback option to ignore the user DV, so that data incorrectly marked invalid at record time can still be processed. This second byte is only relevant during playback, and is not used during record (pass 0x01 if just setting the user DV).

24)   Command: USER_DV_READ

Msg Code: 24

Data: none

Expected Resp: RESP_USER_DV

Timer Duration: 0.5 sec

Eq Console Cmd: user dv [playback]   (user DV also displayed on console screen)

Description: Reads the current setting of the record user data-valid (DV) flag and playback DV-enable flag, as set using USER_DV_SET.

25)   Command: USER_DV_READ_PB

          Msg Code: 25

               Data: none

     Expected Resp: RESP_USER_DV

     Timer Duration: 0.5 sec

     Eq Console Cmd: user dv [playback]   (user DV also displayed on console screen)

          Description: Reads the playback user data-valid (DV) flag, as set by the user at record time. Like USER_DV_READ, the reply contains 2 bytes, but the second byte is always 0. The S2 must currently be playing back properly recorded tapes and all of them must indicate the same user DV value (this should automatically be true if the tapes are aligned, and may or may not be true if they are not). Since the user DV value can be changed at any time during record, the value returned by USER_DV_READ_PB reflects its instantaneous value around the current tape time. If the tapes are playing but not "sufficiently" aligned so that all user DV flags are the same, response RESP_ERR with error code ERR_INPBAUX will be returned. It is not usually necessary to read the user DV flag at playback since the S2 will automatically set the hardware data validity signals accordingly, combined with other sources of playback data validity. If the playback DV-enable flag is set to FALSE, the playback user DV flag will not be used as a factor in determining data validity.

26)   Command: GROUP_SET

          Msg Code: 26

               Data: byte1: transport group number, 0–7
                        (total data bytes = 1)

     Expected Resp: RESP_ERR [ERR_NONE]

     Timer Duration: 1.5 sec

     Eq Console Cmd: group NUM

          Description: Selects which group of transports should be used in modes that don't require all 8 transports. There can be from 1 to 8 different groups, depending on the total data rate of the current mode. Groups are numbered starting at 0. For example, in mode 16x2-1 there are 4 groups numbered 0 to 3. Transports 0–1 are group 0, transports 2–3 are group 1, and so on. The GROUP_SET command is not allowed to change the group number during (automatic) playback or record — the S2 must be stopped and then restarted to change groups. GROUP_SET is typically used following a MODE_SET command.

27)   Command: GROUP_READ

          Msg Code: 27

               Data: none

     Expected Resp: RESP_GROUP

     Timer Duration: 0.5 sec

     Eq Console Cmd: group

          Description: Reads the current transport group number (0–7), as set using GROUP_SET. The reply also includes the total number of groups available (1–8), since this changes depending on the current mode. See RESP_GROUP for the reply format.

28)   Command: TAPEINFO_READ_PB

          Msg Code: 28

Data: none

Expected Resp: RESP_TAPEINFO

Timer Duration: 0.5 sec

Eq Console Cmd: tapeinfo

Description: Returns tape-related information during playback for each of the 8 individual transports. This information is extracted from the auxiliary data channel and includes the following for each tape:

Transport address: The number of the transport containing the tape, 0–7.

Channel ID: The S2 internal channel number of the tape's data, which is equivalent to the transport address at record time. Can be used to determine the ordering of tapes in transports. If auxiliary data could not be extracted (or tape is not playing, ejected, etc.) the channel ID will be given as 0xFF. In this case all other fields will appear as zeroes *except* possibly the tape time since it can be independently extracted (has its own CRC). Note than if auxiliary data cannot be extracted for a particular transport (channel ID is 0xFF), other transports may also show bad auxiliary data even though their data is in fact good, particularly those with higher transport addresses. This is because the S2 "looks" at the transports in order and may not scan past the bad one. This does *not* affect the tape time field.

Tape ID: The max. 20-character tape identifier string as returned by the TAPEID_READ_PB command. Useful to determine individual tape IDs in case the tape IDs are not all the same, causing TAPEID_READ_PB to return error ERR_INPBAUX.

Recorder Mode: The max. 9-character recorder mode string indicating what mode was in effect when the tape was recorded.

Tape time: The playback tape time as returned by the TIME_READ_PB command. Useful to determine individual tape times in case the tapes are not aligned, causing TIME_READ_PB to return error ERR_NOALIGN. Note that times obtained here may be ambiguous to within 1 to 3 seconds since special response synchronization is not performed as with TIME_READ_PB. However, relative time differences between tapes will be accurate. The year value 0 indicates that time could not be extracted from aux data (remaining time bytes will be 0). In general, the time field is much more likely to be valid when playback is marginal because it is decoded separately (i.e. channel ID may be 0xFF while the tape time is valid).

Record start time: The tape time at which this recording started (time RECORD command was issued).

Record S2 serial num: The serial number of the S2 system used to record this tape. Useful for problem tracing.

Record transport serial num: The serial number of the transport (TCP) used to record this tape. Useful for problem tracing.

Record station delay: The measured station delay at record time. Useful to determine if uncorrected delay glitches or any other unreported delay offsets existed at record time.

The above information may not be valid during tape alignment or immediately after playback starts (channel ID will be 0xFF). See RESP_TAPEINFO for detailed field information.

29) Command: DELAY_SET

Msg Code: 29

Data: byte1: type, 0 = absolute setting, 1 = relative setting
byte2 ... byte5: 32-bit signed 2's complement delay value in
                 nanoseconds, byte2 = MSB.
                 Range: -500000000 to +(500000000-1bit) (abs),
                 -(1000000000-1bit) to +(1000000000-1bit) (rel).
(total data bytes = 5)

Expected Resp: RESP_ERR [ERR_NONE]

Timer Duration: 4.0 sec

Eq Console Cmd: delay NUM [s|ms|us|ns|bits] [relative]

Description: Sets the station delay. This provides a positive or negative offset of up to 1/2 second between the external 1 Hz reference and the S2 internal timing. This can be used during record to implement clock offsets and corrections, or during playback to implement smooth delay and delay-rate corrections if the system-clock PLL is in **1hz** mode.

Delay setting can be performed in either absolute or relative terms. For absolute delay setting, the external 1 Hz sync signal (S1HZ_REC on C1 cable or C1HZ on C2a cable) is used as the reference and must be present, otherwise error code ERR_NOU1HZ is returned. Relative delay changes do not require the 1 Hz signal. All delay settings are rounded off to the nearest bit (sample). Use DELAY_READ to obtain the actual delay value implemented, after rounding, and DELAYM_READ to obtain the current true station delay measurement. The allowed absolute delay range is -0.5 sec to one bit-time less than of +0.5 sec, where the bit time is one over the current user channel data rate (e.g. 62.5 ns in mode 16x8-1). The allowed relative delay range is one bit-time short of -1.0 sec to one bit-time short of +1.0 sec. See also ALIGN (below), and the descriptions of the delay, align, and scpll console commands.

If the System-Clock PLL mode is **1hz** (used for playback), small station delay changes involve a smooth, gradual movement to the new delay value (in any other SC PLL mode besides **1hz**, delay changes glitch the output clocks and data). A typical application would be to have several S2-PTs at a correlation center fed by a common fixed 1 Hz reference, with DELAY_SET commands sent over the RCL to each PT for delay tracking. All system timing and data remains continuously valid provided that the following limits are met: max. delay rate $\pm 1 \times 10^{-3}$ s/s, max. delay acceleration $\pm 1 \times 10^{-6}$ s/s^2, max. delay step at 0 delay rate $\pm 1 \times 10^{-5}$ s. You should use the ALIGN command instead of DELAY_SET for large delay changes that would exceed these limits. Exceeding the limits may cause STAT_SCPLLSLIP and/or STAT_SCPLLUNLOCK status messages indicating that the SC PLL is considered unlocked (difference between delay setting and delay measurement greater than about 6×10^{-5} s). Smooth delay tracking is still attempted, although internal timing problems resulting in loss of playback data become more likely in this situation. Smooth delay tracking is suspended for delay errors or steps greater than about 3×10^{-3} s, at which time phase-offsetting is used to adjust the delay.

30) Command: DELAY_READ

Msg Code: 30

Data: none

Expected Resp: RESP_DELAY

Timer Duration: 0.5 sec

Eq Console Cmd: delay

Description: Returns the current station delay setting, from the last DELAY_SET command. For various reasons including movement or glitching of the 1 Hz reference or finite system-clock PLL VCO precision when not in **refclk** mode, this may not be the same as the true measured delay value (see next command). The station delay is expressed in absolute terms, even if the last DELAY_SET command was relative, and will be in the range -0.5 sec to one bit-time short of +0.5 sec. See RESP_DELAY for the exact reply format.

31) Command: DELAYM_READ

Msg Code: 31

Data: none

Expected Resp: RESP_DELAY

Timer Duration: 0.5 sec

Eq Console Cmd: (none, delay measurement displayed on console screen)

Description: Returns the latest station delay measurement. This is the true clock offset between the externally supplied 1 Hz reference and the S2's internal 1 Hz tick, which should match the current delay setting if the system-clock PLL is in **refclk** mode, or be within a few hundred bits of it when in **1hz** mode. The station delay measurement is always in the range -.5 sec to one bit-time short of +.5 sec. See RESP_DELAY for the exact reply format. If the user external 1 Hz reference is not present, RESP_ERR with code ERR_NOU1HZ will be returned.

32)   Command: BARRELROLL_SET

Msg Code: 32

Data: byte1: Boolean barrel roll setting,

    0x00   Turn barrel roll OFF

    0x01   Turn barrel roll ON

(total data bytes = 1)

Expected Resp: RESP_ERR [ERR_NONE]

Timer Duration: 0.5 sec

Eq Console Cmd: barrelroll {on|off}

Description: Turns barrel-roll on or off. When on, barrel-roll rotates user data over all active transports, and un-rotates on playback, so that the possible effect of a marginal transport is averaged over all channels.

33)   Command: BARRELROLL_READ

Msg Code: 33

Data: none

Expected Resp: RESP_BARRELROLL

Timer Duration: 0.5 sec

Eq Console Cmd: barrelroll

Description: Reads the current barrel-roll setting, as set using BARRELROLL_SET. During (auto) playback the barrel roll is automatically set to the same condition as at record time based on information encoded on the tape.

34)   Command: ALIGN

Msg Code: 34

Data: byte1: alignment type code:

    0x00: align to absolute tape time

        byte2, byte3: tape UT absolute year (1901–2099), byte2= MSB

            If 0, correct year and day will be inferred.

            If 1, just correct year will be inferred.

        byte4, byte5: tape UT days (1-365/366), byte4= MSB

            Ignored if year is 0 (correct day will

            be inferred).

        byte6: tape UT hours (0-23)

        byte7: tape UT minutes (0-59)

        byte8: tape UT seconds (0-59)

        byte9 ... byte12: tape UT ns (0-999999999), byte9 = MSB

            (32-bit unsigned integer)

    0x01: relative alignment

        byte2: sign,

                                      0 = positive (forward in time)
                                      1 = negative (backward in time)
                          byte3: hours (0-23)
                          byte4: minutes (0-59)
                          byte5: seconds (0-59)
                          byte6 ... byte9: nanoseconds (0-999999999), byte6 = MSB
                                      (32-bit unsigned integer)
                  0x02: "re-align": Re-align to latest alignment reference time
                          (no additional data bytes)
                  0x03: "self-align": Align the transports to each other using the
                          earliest, latest, or middle tape time as the reference,
                          as per the **selfaligntarg** defaults file option
                          (no additional data bytes)
          (total data bytes = 12 max.)

Expected Resp: RESP_ERR [ERR_NONE]

Timer Duration: 0.5 sec

Eq Console Cmd: align [[+|-]NN:NN:NN[.NNNNNNNNN]]

Description: This is used during automatic playback to align tapes to an absolute time or
by some relative amount. It works by slewing the transports and (if neces-
sary) adjusting the station delay setting. Alignment is normally done after po-
sitioning is complete. Unlike positioning, tape alignment is a precise
operation, accurate down to the bit. All alignment times are rounded off to
the nearest bit (sample) time at the current user channel data rate. Any
ALIGN command where the nanoseconds portion is non-zero *implicitly per-
forms a relative station delay adjustment* in the same way as the
DELAY_SET command. However, ALIGN should not be used to implement
delay and delay-rate tracking, as this may interfere with the automatic play-
back monitoring — use DELAY_SET instead. DELAY_SET (or ERRMES)
should be used in place of ALIGN for any regular delay adjustments that
fall within the delay-rate and acceleration limits given in the description of
DELAY_SET.

Two kinds of tape alignment are supported: *absolute* and *relative*. In ab-
solute alignment, the external control computer specifies a time to which the
S2 should align its tapes. This time is referenced to the second it is received
by the S2, i.e. it is the tape time that would appear if the procedure was in-
stantaneous, but in fact alignment may take several seconds or minutes, de-
pending on the distance to slew. Because of the asynchronous nature of the
Radioastronomy Control Link, the external computer must issue the absolute
align command immediately following an S-1 Hz output tick (S1HZ_PB on
C2a cable), and that tick is taken as the reference for the absolute time con-
tained in the command. If it takes 20 seconds to align, then the first valid
data will be for the requested alignment time plus 20 seconds. When it is
not convenient to include the year and/or day number in the absolute align-
ment time, these can be automatically inferred based on the year and day
number extracted from tape: pass the year as 0 to infer both the year and
day, or 1 to infer just the year.

Relative alignment is performed by applying a signed relative offset to
the current tape time. Unlike absolute alignment, there are no special synchro-
nization requirements for the command itself. However the tapes must al-
ready be aligned to the same tape time or error code ERR_NOALIGN will
be returned (if necessary use the self-align option first, see below). The exter-
nal control computer may read the tape time with TIME_READ_PB in order
to calculate the required relative offset.

Two special alignment options are provided that may be useful in certain
cases. "re-align" (type code 2) causes the tapes to re-align to the previously
chosen reference time should one or more transports ever become un-aligned.

This is normally done automatically by the S2's regular playback monitoring, but a way to initiate this manually may sometimes be useful following a "fatal" align error (STAT_ALIGNFAIL), where no further automatic attempts are made to re-align. The "re-align" option should not be used unless some other type of alignment has already been performed for the current set of tapes.

The "self-align" option (type code 3) causes the transports to align to each other by chosing a target reference time based on the current tape times. Three selection methods are available according to the defaults file option **selfaligntarg**: the "first" tape time, the "last" tape time, or the "middle" tape time half-way between the two. Self-alignment is the type of alignment normally performed automatically each time playback starts. However, the user may explicitly initiate self-alignment to force tape alignment following certain status conditions such as STAT_ALIGNFAIL (a fatal error in the previous alignment attempt, see below). To prevent the automatic self-alignment at the start of playback, issue a *relative* ALIGN command with a zero offset (i.e. hours, minutes, seconds, nsec all 0) *immediately* after playback starts. This will result in a STAT_ALIGNFAIL message with error code ERR_NOALIGN, which can be ignored, but no tape slewing will take place.

While alignment is in progress the current tape time is undefined and playback data, including auxiliary data, is invalid. The STATUS command should be used to monitor the progress of the alignment procedure. STAT_ALIGNIP indicates that tape alignment is in progress, and STAT_ALIGNDONE will appear momentarily when alignment successfully completes. STAT_ALIGNFAIL indicates that the alignment operation aborted due to a fatal error and will persist until another alignment operation is explicitly started or playback is terminated. STAT_ERRALIGN appears momentarily to indicate non-fatal errors which cause the alignment operation to be automatically restarted. Finally, STAT_NOALIGN appears whenever tapes are not aligned and playback data should be considered invalid. It will disappear following a successful alignment operation and will re-appear if any unusual condition causes the tapes to go out of alignment.

Note that absolute ALIGN commands (but not any other type) are not subject to automatic communications retries within the RCL interface library, otherwise the wrong alignment time might be sent. The rcl_align_abs() function call should be re-issued with an updated time if a communications timeout occurs. See also the DELAY_SET command.

35)   Command: POSITION_SET

    Msg Code: 35

        Data: byte1: positioning type code:

            0x00: absolute tape positioning

                byte2: number of positions being specified (1 or 8). Use 1 when commanding all currently selected transports to the same position, 8 when commanding transports to different positions.

                byte3 ... : List of positions. Each entry is a signed 32-bit integer representing the absolute position in seconds since BOT, MSB first. Values should be restricted to the range 0–43199, or 0x7FFFFFFF if byte2==8 and transport not selected (up to 32 bytes total)

            0x01: relative tape positioning

                byte2: number of positions being specified (1 or 8)

                byte3 ... : List of positions. Each entry is a signed 32-bit integer representing the relative number of seconds to move, MSB first. Values should be restricted to the range -43199–43199, or 0x7FFFFFFF if byte2==8 and transport

                        not selected (up to 32 bytes total)
            0x02: position preset
                        byte2: number of positions being specified (1 or 8)
                        byte3 ... : List of positions. Each entry is a signed 32-bit
                                integer representing the absolute position in seconds
                                since BOT, MSB first. Values should be restricted to the
                                range 0–43199, or -0x80000000 to mark a position as
                                unknown, or 0x7FFFFFFF if byte2==8 and transport not
                                selected (up to 32 bytes total)
            0x03: "re-establish": for all currently selected transports,
                        rewinds the tape to the beginning and then returns to
                        the original position, which is then known (no
                        additional data bytes).
        (total data bytes = 34 max.)

Expected Resp: RESP_ERR [ERR_NONE]

Timer Duration: 1.5 sec

Eq Console Cmd: position

Description: The position command is used before playback or record to position the
        tapes to a specific location. Position is measured as the number of seconds
        of recorded data since the beginning of tape, and is normally displayed as
        hours/minutes/seconds, e.g. 3:18:57. This format is similar to a tape time ex-
        cept that the hours portion is normally shown with only a single digit. Sec-
        tions of blank tape do not count in position determination. During positioning
        the recorder state changes to RSTATE_POSITION (use STATE_READ to
        test). Although there may be up to 8 tapes and therefore 8 different posi-
        tions, we try to work with a single position value whenever possible since
        the group of currently active transports will usually be in almost the same lo-
        cation. The current position is considered *unknown* after a tape is inserted, or
        following application of power to the transports. The position becomes
        known when a transport encounters beginning-of-tape (BOT), where the posi-
        tion is defined to be 0 (i.e. 0:00:00)[2]. Although positions are normally posi-
        tive for purposes of POSITION_SET, the current position reading can take
        on small negative values due to anomalies around BOT.

        There are two ways to perform positioning: *absolute* and *relative*. In addi-
        tion, there are two special positioning functions *preset* and *re-establish*. Abso-
        lute positioning moves the tapes to a specific position regardless of their
        current location. If the current position is not known, the tapes are first auto-
        matically rewound to the beginning to determine the position. Either 1 or 8
        individual position values may be specified. If only 1 is specified ("number
        of positions" = 1), all transports are moved to the same position. When 8
        values are specified, each corresponds to a transport in order from 0 through
        7. Transports which are not selected under the current recorder mode and
        group settings will ignore the position command. When 8 position values are
        given the entries for unselected transports should be filled in as 0x7FFFFFFF
        (the maximum 32-bit signed integer value). To determine which transports are
        currently unselected, use POSITION_READ and check for position entries
        with the value 0x7FFFFFFF. Note that using 0x7FFFFFFF for transports
        which *are* currently selected is not allowed, and will result in an error.

        Relative positioning moves the tapes a given amount relative to their cur-
        rent position. The current position does not need to be known for relative
        positioning to work. Positive position values move forward, and negative val-

---

[2] *The position "preset" feature will also make the position known.*

ues move backward.

The position re-establish feature allows an unknown position to be determined without moving the tapes to a specific absolute location. The tapes are rewound and then automatically returned to their original position, as computed from the distance covered during rewind, and the position is again considered known. This may be useful following a power failure when the S2 has lost track of the current tape position, or when non-rewound tapes are inserted. It can also be used to accurately re-determine the current position if cumulative errors have built up in the position measurement after a long period of repetitive tape activity.

The position preset feature allows users to set the current position without actually moving the tapes. This should only be used when the current position is considered unknown by the S2 and the user has detailed knowledge of what the position should be, and there is not enough time to rewind the tapes to determine the position in the normal way. Position preset can also be used to explicitly mark the position as unknown by using the "unknown" position -0x80000000 (the minimum 32-bit signed integer value).

36)	Command:	POSITION_READ
	Msg Code:	36
	Data:	byte1: type of request,
		0x00 return overall (mid-point) position and linear variance
		0x01 return 8 individual transport positions
		(total data bytes = 1)
	Expected Resp:	RESP_POSITION
	Timer Duration:	0.5 sec
	Eq Console Cmd:	transport display position (also displayed on console screen)
	Description:	Reads the current S2 tape position in one of two formats, *overall* or *individual*. The overall position is a single position value defined as the mid-point (middle) of the individual transport positions, and includes a linear variance value which indicates the maximum absolute deviation of any individual tape position from the mid-point. Only the transports selected under the current mode and group settings are counted in the overall position. If the variance is small enough (depending on the user's application) then it will be possible to later return to the same position using only the overall position value. If for some reason the variance is large, it may be necessary to obtain the individual transport tape positions to be able to return to exactly the same position. The individual position information obtained should be treated as a unit for purposes of subsequently commanding the S2 to return to that position with POSITION_SET. Any individual position value that is unknown will be returned as -0x80000000 (the minimum 32-bit signed integer value). Transports which are not selected for use under the current recorder mode and group settings will return their positions as 0x7FFFFFFF. Although normally positive, position readings can take on small negative values due to anomalies around BOT. See the description of the response RESP_POSITION for more information.

37)	Command:	ERRMES
	Msg Code:	37
	Data:	byte1 ... byte4: 32-bit signed 2's complement integer indicating the FIFO error in samples at the current user channel data rate.
		(total data bytes = 4)
	Expected Resp:	RESP_ERR [ERR_NONE]
	Timer Duration:	0.5 sec
	Eq Console Cmd:	(none)

Description: Used to synchronize the S2 recorder's playback rate when the System-Clock PLL is in **errmes** mode. The parameter indicates the difference between the actual playback tape position and the desired position as a number of bits (data samples) at the current user channel data rate. In practice, this is the relative distance between the center of the correlator's internal data FIFO and the read pointer. If the sign is positive then the FIFO read pointer is between center and overflow. If the sign is negative then the pointer is between center and underflow. When using **errmes** mode, an ERRMES command should be sent by the external control computer about once per second (occasionally missing one is not serious). The time between taking the FIFO error measurement and transmitting the command should be kept as short as possible for optimum SC PLL performance. The following limits should be observed:

     Max. delay rate: $\pm 1 \times 10^{-3}$ s/s

     Max. delay acceleration: $\pm 1 \times 10^{-6}$ s/s^2

     Max. delay step at 0 delay rate: $\pm 1 \times 10^{-5}$ s

Exceeding these limits may cause STAT_SCPLLSLIP and/or STAT_SCPLLUN-LOCK status messages indicating that the SC PLL is considered unlocked (FIFO error greater than about 6×10^{-5} s). In this situation, i.e. ERRMES values greater than 6×10^{-5} s, smooth delay tracking continues as well as possible, although internal timing problems resulting in loss of playback data become more likely in extreme cases where the SCPLL VCO is pushed near its rail values.

38)   Command: ESTERR_READ

     Msg Code: 38

         Data: byte1: ordering of returned list,

               0x00   in order of transports (0..7)

               0x01   in order of internal recorder channels (0..7)

         (total data bytes = 1)

Expected Resp: RESP_ESTERR

Timer Duration: 0.5 sec

Eq Console Cmd: transport display esterr

Description: Returns a list of estimated bit error rates. Each entry in the list corresponds to a transport and/or S2 internal data channel number, although this fact can be ignored. The number of entries in the list depends on how many channels/transports are currently active, as determined by the current mode. The list can be further processed externally to compute an average or select the highest & lowest entries. The ordering of the list can be either by transport or by internal recorder channel. If by transport, the order corresponds to the physical transports and the tapes they contain. If by channel, the order is independent of playback tape ordering and corresponds to the ordering of transports and tapes at record time. The default should usually be ordering by transport (data byte1 = 0x00). Estimated error rates are encoded as ASCII strings instead of as binary floating-point quantities to avoid problems with incompatible floating-point representations.

     Error rates are estimated by counting the number of missed sync words in the formatted internal data channels and extrapolating to encompass all data bits. We assume that the error rate is low, so if a sync word is incorrect we presume that in fact only a single bit is in error. These assumptions break down for high error rates and for extended burst errors, but in practice the estimate has been found to be very close to the true bit error-rate in almost all cases. Because sync words represent only a small proportion of the data, a relatively long integration is needed to measure useful error rates. The integration period used is 6 seconds, giving a minimum measurement resolution of around 1.0×10^{-6}. There is also an artificial upper limit of

1.0 x 10^{-2}, imposed to avoid the inaccurate upper estimation range. The strings returned in the two boundary cases are "<1.0e-6" and ">1.0e-2", although the exact numbers may vary (external software should check only for the initial "<" or ">" sign).

Estimated error rates are available in both record and playback states. In record, the error rate reflects the results of decoding data from the "record" diagnostic loopback path, and will usually be below the minimum resolution threshold. Because it has only one decoder, an S2-RT may take between 1 and 3 minutes to build up a complete list of error rate estimates when first starting playback or record. Until then error estimates are considered unknown, and appear as empty strings. The same is true when recording on a PT (because there is only one diagnostic path) but not during playback. This also means that at any time individual estimates might be several minutes old. In addition, certain status conditions can interfere with automatic playback/record scanning and prevent estimates from being computed or cause them to stop being updated. Also remember that the error rates are calculated for internal recorder channels and do not apply directly to user data channels unless the user data rate is 16 Mbits/s (e.g. mode 16x8-1). The estimated error rate values are for operator reference only, and do not by themselves indicate correct system functionality (system status is used for that).

39)   Command: PDV_READ

Msg Code: 39

Data: byte1: ordering of returned list,
          0x00   in order of transports (0..7)
          0x01   in order of internal recorder channels (0..7)
(total data bytes = 1)

Expected Resp: RESP_PDV

Timer Duration: 0.5 sec

Eq Console Cmd: transport display dv

Description: Returns a list of percent data valid. Each entry in the list corresponds to a transport and/or S2 internal data channel number, although this fact can be ignored. The number of entries in the list depends on how many channels/transports are currently active, as determined by the current mode. The list can be further processed externally to compute an average or select the highest & lowest entries. The ordering of the list can be either by transport or by internal recorder channel. If by transport, the order corresponds to the physical transports and the tapes they contain. If by channel, the order is independent of playback tape ordering and corresponds to the ordering of transports and tapes at record time. The default should usually be ordering by transport (data byte1 = 0x00). Percent data valid is encoded as ASCII strings instead of as binary floating-point quantities to avoid problems with incompatible floating-point representations.

The % DV value is obtained by integrating a channel's data validity signal over a one second period, where 100.00% indicates all data was valid and 0.00% indicates no data was valid. % DV is available in both record and playback states. In record, the value reflects the results of decoding data from the "record" diagnostic loopback path, and will usually be 100.00%. Because it has only one decoder, an S2-RT may take between 8 and 90 seconds to build up a complete % data valid list when first starting playback or record. Until then entries are considered unknown, and appear as empty strings. The same is true when recording on a PT (because there is only one diagnostic path) but not during playback. This also means that at any time individual values might be up to 90 seconds old. In addition, certain status conditions can interfere with automatic playback/record scanning and prevent % data valid from being computed or cause it to stop being updated. Remem-

ber that % data valid is calculated for internal recorder channels and does not apply directly to user data channels unless the user data rate is 16 Mbits/s (e.g. mode 16x8-1). The % data valid values are for operator reference only, and do not by themselves indicate correct system functionality (system status is used for that).

42) Command: SCPLL_MODE_SET
    Msg Code: 42
        Data: byte1: SC PLL mode code
                    0x00 **xtal**
                    0x01 **manual**
                    0x02 **refclk**
                    0x03 **1hz**
                    0x04 **errmes**
              (total data bytes = 1)
Expected Resp: RESP_ERR [ERR_NONE]
Timer Duration: 0.5 sec
Eq Console Cmd: scpll mode {refclk|1hz|errmes|xtal|manual}
    Description: Sets the System Clock Phase-Lock Loop mode. The SC PLL, located on the UI board, drives all of the S2 system timing. Its mode determines what the PLL uses as its lock reference. The SC PLL is normally locked to an external high-rate reference clock during record (**refclk**), or to a 1 Hz or software reference for playback rate control at a correlator (**1hz** or **errmes**). The PLL can also run independently based on an internal crystal oscillator (**xtal**) or be controlled manually for testing (**manual**). The initial SC PLL mode is determined by the **scpllmode** option of the system defaults file.

43) Command: SCPLL_MODE_READ
    Msg Code: 43
        Data: none
Expected Resp: RESP_SCPLL_MODE
Timer Duration: 0.5 sec
Eq Console Cmd: scpll mode   (also displayed on console 'userio' screen)
    Description: Reads the current System Clock Phase-Lock Loop mode, which determines what is being used as the system timing reference.

44) Command: TAPETYPE_SET
    Msg Code: 44
        Data: byte1 ... : ASCII tape type string terminated by a NULL (0) character, maximum 6 characters (not incl. NULL). Note that all normal tape type codes are one character long (0–9, A–Z) but ''unlisted'' tape types are six digits long of the form NNNMMM where NNN is the write current for LP and MMM is the write current for SLP.
              (total data bytes = length of string + 1, max. 7)
Expected Resp: RESP_ERR [ERR_NONE]
Timer Duration: 0.5 sec
Eq Console Cmd: tapetype [STR]
    Description: Sets the tape type. This indicates what type of tape is loaded in the S2 using the type codes 0–9 and A–Z as listed in Chapter 5. The tape type controls how much write current should be used when recording. As new tape types are approved software upgrades will be provided to allow recognition of the new type codes. For unlisted tape types it is also possible to specify a 6-digit numeric string which indicates the LP and SLP write-current levels

explicitly. For example, '100070' sets LP write current to 100 and SLP write current to 70. The required 6-digit tape type strings will be supplied by CRESTech/SGL when new tape types are approved. Note that the tape type reverts to the default value given in the defaults file when the S2 reboots. The TAPETYPE_SET command is not allowed during record.

45)   Command: TAPETYPE_READ
        Msg Code: 45
            Data: none
    Expected Resp: RESP_TAPETYPE
    Timer Duration: 0.5 sec
    Eq Console Cmd: tapetype
        Description: Reads the current tape type setting. See the description of RESP_TAPETYPE for more information.

50)   Command: MK3_FORM_SET
        Msg Code: 50
            Data: byte1: Boolean value,
                        0x00 Disable Mark III formatter
                        0x01 Enable Mark III formatter
                 (total data bytes = 1)
    Expected Resp: RESP_ERR [ERR_NONE]
    Timer Duration: 5.0 sec
    Eq Console Cmd: mk3 form {enable|disable}
        Description: Turns the S2 internal Mk3 format generator on and off. When turned on, the C2a cable output switches to Mk3-compatible mode.

51)   Command: MK3_FORM_READ
        Msg Code: 51
            Data: none
    Expected Resp: RESP_MK3_FORM
    Timer Duration: 0.5 sec
    Eq Console Cmd: mk3 form
        Description: Reads the current Mark III formatter enable/disable state, as set with MK3_FORM_SET.

55)   Command: TRANSPORT_TIMES
        Msg Code: 55
            Data: none
    Expected Resp: RESP_TRANSPORT_TIMES
    Timer Duration: 0.5 sec
    Eq Console Cmd: transport status time
        Description: For all transports reads the total power-on time since manufacture, total head-use time since manufacture, time since last head replacement (head-use), and time since last service (in-service). The head-use and in-service times give the amount of active head use since the last head replacement or service operation was performed, respectively, as entered using the transport N service console command. The reply includes TCP serial numbers for identification. See RESP_TRANSPORT_TIMES for more information.

60)   Command: STATION_INFO_READ
        Msg Code: 60

Data: none
Expected Resp: RESP_STATION_INFO
Timer Duration: 0.5 sec
Eq Console Cmd: station, serial
Description: Reads information which identifies the S2 system, including its station number, serial number, and nickname. All three of these values come from the system defaults file (**station**, **nickname**, and **serialnum** options). See RESP_STATION_INFO for more information.

70) Command: CONSOLECMD (generic)
Msg Code: 70
Data: byte1 ... : ASCII console command string terminated by a NULL (0) character, maximum 255 characters (not incl. NULL).
(total data bytes = length of string + 1, max. 256)
Expected Resp: RESP_ERR [ERR_NONE]
Timer Duration: 3.0 sec
Eq Console Cmd: —
Description: Executes an arbitrary console command. There are some severe limitations. First, no command output is returned except the error return code, so commands which just display information are of no use. Second, it is up to the caller to ensure that the chosen command is *guaranteed* to complete in less than 3 seconds. For example, during playback, `setup 0 transport 5` is fine but `setup transport 5` is bad because it can take more than 3 seconds to decode channel information from transport 5. Commands such as `form berc` are way out — you will permanently hang the RCL and probably eventually crash the S2 software. Any command already supported by the RCL should not be executed with CONSOLECMD, even apparently innocuous commands such as `record` or `stop`. Remember: use this feature at your own risk.

71) Command: POSTIME_READ
Msg Code: 71
Data: byte1: Transport number (0–7)
(total data bytes = 1)
Expected Resp: RESP_POSTIME
Timer Duration: 0.5 sec
Eq Console Cmd: (none)
Description: This command is for SGL internal use when running tests to check and calibrate the internal transport positioning software. The response RESP_POSTIME includes both the playback tape time and tape position for a particular transport. When test tapes are recorded in such a way that the tape time matches the tape position, this can be used to determine cumulative position sensing errors. The response from the S2 recorder is sent immediately (unlike TIME_READ_PB where the response is synchronized to the 1 Hz tick). If the playback tape time is unknown or invalid, the 'year', 'day', 'hour', 'min', 'sec', and 'frame' fields will all be 0. If the transport position is unknown, 'position' will be -0x80000000.

80) Command: STATUS (generic)
Msg Code: 80
Data: none
Expected Resp: RESP_STATUS
Timer Duration: 0.5 sec

Eq Console Cmd: (displayed on console screen)

Description: Reads the S2 recorder brief status report. The brief report includes only numeric status codes, no text messages. For a detailed description of all status codes see Section 4.5. This command should be sent approximately every 1 to 5 seconds since status conditions accumulate in-between reads and will come out all at once on the next read. Status entries are returned in order of severity from most to least severe, which means in order of increasing status code number, not in order of occurrence. Reading S2 status with the RCL STATUS command causes all clear-on-read status entries to be cleared for the RCL, similar to typing 'status reset' on the console for the console status. These entries will not appear in the next STATUS request unless the condition they represent has occurred again. Non-clear-on-read status entries will persist until the condition they represent has gone away. Note that RCL status is not necessarily the same as the console status window, and is not affected by console commands such as 'status reset'. Similarly, reading the RCL status will not affect the console status window. Use STATUS_DETAIL to obtain detailed information messages for status conditions returned here. See RESP_STATUS for more information.

81)   Command: STATUS_DETAIL   (generic)

Msg Code: 81

Data: byte1: status code to return detailed information for, 0 = all active status codes

byte2:   Boolean, "re-read" flag:

  0x00 Don't re-read status, use conditions from last STATUS request

  0x01 Re-read status (implicitly performs STATUS first)

byte3:   Boolean, "short" option:

  0x00 Use regular long-form messages (one or more lines, up to 399 characters, but often less than 80 chars)

  0x01 Use shortened messages (like console, max. 34 chars)

(total data bytes = 3)

Expected Resp: RESP_STATUS_DETAIL

Timer Duration: 0.5 sec

Eq Console Cmd: status display

Description: Reads S2 recorder detailed status report for 1 or all active status conditions. If the "re-read" flag (data byte 2) is not set, this lists only conditions in effect at the time of the last STATUS command. Thus you can obtain detailed information after detecting an unusual status condition in the brief status response (RESP_STATUS), without having the active status conditions change. The detailed status messages are suitable for logging or other error reporting purposes. If the "re-read" flag is set, this command lists new status conditions, i.e. implicitly performs STATUS first. The "short" option allows access to shorter versions of the detailed status messages, useful to save space when displaying status. However the regular (long) form should still be used whenever possible since the short messages are intended for the console and are not as accurate or complete. The reason that two versions of the status command (STATUS and STATUS_DETAIL) are provided is for efficiency so that you can use the "fast" one for polling and the slower one (STATUS_DETAIL) after problems are found if you need the text messages. If you always ask for the text messages regardless, you can skip sending STATUS and send STATUS_DETAIL directly with the "re-read" flag TRUE. See RESP_STATUS and RESP_STATUS_DETAIL for more information.

82)   Command: STATUS_DECODE   (generic)

Msg Code: 82

Data: byte1: numeric status code from 1 to 127 (RCL_STATCODE_MAX)
byte2: Boolean, ''short'' option:
0x00 Return regular long-form message (one or more lines, up to 399 characters)
0x01 Return shortened message (like console, max. 34 chars)
(total data bytes = 2)

Expected Resp: RESP_STATUS_DECODE

Timer Duration: 0.5 sec

Eq Console Cmd: status decode

Description: Translates a numeric status code to its corresponding mnemonic/text message. This is useful for building a run-time table of status codes and mnemonics so that the mnemonics, and not the changeable codes themselves, can be used easily in software condition tests. For error reporting or logging purposes use STATUS_DETAIL instead since that includes occurrence-specific information such as channel numbers in the returned messages, while STATUS_DECODE fills in such variable information with "xxx".

83)   Command: ERROR_DECODE   (generic)

Msg Code: 83

Data: byte1: 8-bit signed 2's complement error code
(total data bytes = 1)

Expected Resp: RESP_ERROR_DECODE

Timer Duration: 0.5 sec

Eq Console Cmd: error decode

Description: Translates a negative numeric error code (ERR_*) to the corresponding text message (includes mnemonic). See also RESP_ERR and the list of S2 error codes on Page A-32.

90)   Command: DIAG   (generic)

Msg Code: 90

Data: byte1: Diagnostic type:
0x01 Perform self-test 1
(total data bytes = 1)

Expected Resp: RESP_ERR [ERR_NONE]

Timer Duration: 0.5 sec

Eq Console Cmd: diag self1

Description: Initiates S2 internal diagnostic sequences. Currently the only diagnostic sequence which can be run here is self-test 1 (the power-on self test). While self-test 1 is running most RCL commands which affect tape motion or system switch settings are not allowed (they will return ERR_SELF1INPROG). Users should poll S2 status to determine when the diagnostic test completes: STAT_DIAGIP indicates that the test is in progress, STAT_DIAGDONE indicates that the test completed successfully, and STAT_DIAGFAIL indicates that the test has found a system fault or aborted due to an error.

91)   Command: BERDCB

Msg Code: 91

Data: byte1: Desired operation type:
   0x01 Perform Formatter bit-error rate measurement
   0x02 Perform UI bit-error rate measurement
   0x03 Perform UI DC-bias measurement

byte2: Channel number to use (internal data channel 0 through 7 for FORM BER, or user data channel 0 through 15 for UI BER and UI DC-bias).

byte3: Measurement time in seconds (1–255). *This directly affects the length of time for this command to execute and thus determines the required timer duration value (see below).* Recommended typical value: 1 sec, recommended maximum value: 10 sec.

(total data bytes = 3)

Expected Resp: RESP_BERDCB

Timer Duration: Measurement time + 1 sec

Eq Console Cmd: form berc, uic berc, uic dcbias

Description: Performs one of 3 types of statistical measurements on a given data channel: Formatter bit-error rate, UI bit-error rate, or UI DC-bias. All of these measurements can only be done on one channel at a time. The FORM BER measurement performs a true bit-error rate measurement on one 16 Mbit/s internal S2 data channel (0–7) by comparing against the Formatter test vector sequence. The UI BER measurement performs a true bit-error rate measurement on one user data channel (0–15) by comparing against the UI test vector sequence. Both BER measurements always count both detected and undetected errors (i.e. data validity is ignored). In the case of FORM BER the formatter Test Vector Generator is automatically turned on for the duration of the measurement if required.

The UI DC-bias measurement counts the number of bits with value 1 in the specified user data channel (0–15). This is useful to determine the ratio of 1-bits to 0-bits to check if data is reasonable. Note that both the UI DC-bias and UI BER measurements are made on UI output data (similar to the data at the C2a cable port). To make measurements of UI input data (similar to the data at the C1 cable port) you should turn UI feed-through mode on. Since there is currently no RCL command to set UI feed-through, you will have to use the RCL CONSOLECMD feature with the console command strings "uic feedthru on" or "uic feedthru off".

The typical recommended measurement time is 1 second to keep RCL response times quick. It is important to understand that while the BERDCB command is in progress no other RCL commands are allowed, therefore the maximum recommended measurement time is 10 seconds. In the case of FORM BER times of up to 255 seconds are actually possible, but for UI BER and DC-bias times are limited to a few seconds because of limited resolution in the internal counters.

92) Command: DATA_EXTRACT_READ

Msg Code: 92

Data: byte 1: type of extraction (0 to 3),
   0 - extract S2 user data (format-less)
   1 - extract S2 data validity
   2 - extract MkIII-format data (trigger on sync detect)
   3 - extract MkIII-format data (trigger on 1 Hz tick)

byte 2: bit offset (0 to 255, must be 0 for 32 Mbit modes 32xU-Q)

byte 3: user channel number (range 0-15 for S2 data/validity or 0-13 for MkIII data)

(total data bytes = 3)

Expected Resp: RESP_DATA_EXTRACT

Timer Duration: 1.25 sec

Eq Console Cmd: uic dataextract, uic validityextract, mk3 dataextract

Description: Extracts a small "snapshot" of data, consisting of 24 consecutive bits or three bytes of data, from a particular S2 user data channel. Can also be used to extract S2 data validity, or Mark III formatted data when the Mark III formatter is enabled. The extracted data "snapshot" starts synchronous to the next 1 Hz tick, or optionally synchronous to the next detected sync word for Mark III format data (extraction type code 2). A bit offset may be specified to delay the start of extraction for up to 255 bits relative to the 1 Hz tick (or sync word trigger). Bit offsets are not supported in 32 Mbit modes (offset value must be 0). For Mark III data-extract the three least-significant bits of the fourth response byte indicate parity errors for each of the 3 data bytes. The contents of the data or validity channel used for extraction depends on the current data path configuration as set by other RCL commands such as PLAY, STOP, MK3_FORM_SET, and console commands such as `setup`, `uic feedthru`, `mk3 form`, and others. The DATA_EXTRACT_READ command is intended for diagnostic and test purposes. It can be used, for exampled, to help verify the correct connection of the data acquisition system or sampler supplying data to the S2 recorder.

97)   Command: IDENT   (generic)
       Msg Code: 97
           Data: none
  Expected Resp: RESP_IDENT
 Timer Duration: 0.5 sec
Eq Console Cmd: (none)
    Description: Returns the RCL device type as a string, which determines what command set it responds to. All RCL devices should implement this command. Device types defined so far are "S2-RT", "S2-PT", and "S2-DAS" for the S2 Data Acquisition System. To obtain additional information from S2-RTs and S2-PTs use the STATION_INFO_READ and TRANSPORT_TIMES commands.

98)   Command: PING   (generic)
       Msg Code: 98
           Data: none
  Expected Resp: RESP_ERR [ERR_NONE]
 Timer Duration: 0.5 sec
Eq Console Cmd: (none)
    Description: Performs no action, just returns an "ERR_NONE" response. All RCL devices should implement this command. PING is useful for polling a device to test if it is alive or to tell when it comes up. Unlike most other commands, PING is not subject to automatic communications retries within the RCL interface library (rcl_ping()), so response will be quick even if the device is currently dead.

99)   Command: VERSION   (generic)
       Msg Code: 99
           Data: none
  Expected Resp: RESP_VERSION
 Timer Duration: 0.5 sec
Eq Console Cmd: version
    Description: Returns ROS (Recorder Operating System) software version information as a string, including the version number, CPU type, and compilation date/time.
                Example response:
                3.0d [133] (compiled Fri May 16 12:59:08 EDT 1997)
                See also RESP_VERSION.

Responses

100)   Response: RESP_ERR   (generic)
          Msg Code: 100
                Data: byte1: 8-bit signed 2's complement error code (see description).
                      (total data bytes = 1)
         Description: Response code RESP_ERR is returned any time a request cannot be com-
                      pleted. The data portion of this response contains an error code byte which
                      can be translated into a descriptive error message using the ERROR_DE-
                      CODE command. Requests which complete successfully and don't need to re-
                      turn any additional information (e.g. PLAY) respond with RESP_ERR error
                      code ERR_NONE. Those successful requests which provide specific informa-
                      tion return the appropriate response packet (e.g. the TIME_READ command
                      returns RESP_TIME packet).
                      ROS error codes and messages are shown below. Note that error codes are
                      negative decimal numbers, encoded as 8-bit 2's complement integers. This
                      list is subject to change at any time, typically for the purpose of adding new
                      error codes. All efforts will be made to avoid changing existing error code
                      numbers, but such changes cannot be ruled out. The mnemonics are unlikely
                      to change. If possible, user software should not rely on the numeric error
                      codes, but use the error mnemonic instead. To interpret an error code, user
                      software should decode it using ERROR_DECODE, and then take action
                      based on the error mnemonic (a string of the form ERR_*). The mnemonic
                      appears at the start of the error message and can be extracted by reading all
                      characters up to but not including the first colon. The message which follows
                      can be written to the screen or recorded in a log file.

ROS Error Return Codes		
0	ERR_NONE	No error has occurred
-1	ERR_OPFAIL	Operation failed (non-specific error)
-2	ERR_ILCMD	Unrecognized command
-3	ERR_NUNIQ	Non-unique items
-4	ERR_BADPARM	Incorrect parameter format
-5	ERR_BADLEN	Incorrect or illegal length
-6	ERR_BADVAL	Value is illegal or out of range
-7	ERR_TIMEOUT	Operation timed out
-8	ERR_CHKSUM	Checksum was bad
-9	ERR_DPTOOSOON	Xilinx D/P pin went high too soon
-10	ERR_DPNOTHI	Xilinx D/P pin did not go high
-11	ERR_BADFORM	Bad format encountered in operation
-12	ERR_VXIO	VxWorks I/O error
-13	ERR_VXSYS	VxWorks system error
-14	ERR_DEVNP	Required device not present
-15	ERR_ILMODE	Recorder mode is illegal for requested operation
-16	ERR_ILCONFIG	Recorder configuration is illegal for requested operation
-17	ERR_TOLEXCD	Measurement error tolerance exceeded
-18	ERR_MISSIG	Operation encountered missed DRD signatures
-24	ERR_NOTIMP	Operation not yet implemented
-25	ERR_HWCONFIG	Not possible with this hardware configuration
-26	ERR_CNTLC	Operation interrupted by cntl-C

	ROS Error Return Codes	(continued)
-28	ERR_ILSETUP	Recorder setup is illegal for requested operation
-29	ERR_INSETUP	Recorder setup is inconsistent across selected transports
-30	ERR_ILSTATE	Recorder state is invalid for requested operation
-31	ERR_NOPBAUX	Could not decode playback auxiliary data
-32	ERR_CHIDMISS	Tape with required channel ID is missing
-33	ERR_CHIDDUP	Two tapes have the same channel ID!
-34	ERR_BADCRC	Operation failed due to bad CRC
-35	ERR_DRNOTINIT	DR configuration is not initialized
-36	ERR_MK3SYNC	Mk3 frame sync error
-37	ERR_LOGBADLEN	Log file full, logging aborted
-38	ERR_LOGCLOSED	Log file is closed
-39	ERR_NOU1HZ	User 1 Hz sync signal not present
-40	ERR_ECHOOFF	Interactive command echoing must not be turned off
-41	ERR_NOALIGN	Tapes playing but not aligned
-42	ERR_INPBAUX	Tapes have inconsistent aux data
-43	ERR_CURSWIN	Error returned by curses windowing routines
-44	ERR_BOTEOT	Encountered beginning/end of tape (BOT/EOT)
-45	ERR_ALBADTIME	Tape-alignment target time is out of range
-46	ERR_ALCRSBOUND	Tape alignment crossed recording boundary
-47	ERR_ILFNAME	Illegal file name (max. length 6-dot-3, alphanum only)
-48	ERR_DELNONZER	Station delay is non-zero, must be zero for this operation
-49	ERR_RPMONPRMT	Operation preempted due to rpmon state change
-50	ERR_BADGROUP	Transport is not part of current group
-51	ERR_OPINPROG	Requested operation is already in progress
-52	ERR_TRANLKFAIL	Transport servo failed to lock
-53	ERR_TRANNOSEL	No transports selected
-54	ERR_TRANBADSEL	Transport selection incorrect for current mode & group
-55	ERR_ILAUTORP	Operation not allowed during automatic record/play
-56	ERR_INSTATE	Recorder state is inconsistent or illegal
-57	ERR_SCPLLBADMODE	System-clock PLL is set to the wrong mode
-58	ERR_NOTAUTORP	Operation valid only during automatic record/play
-59	ERR_DRDBADRTPT	Invalid address/combination of DRD board(s)
-60	ERR_WRITEPROT	Software write-protect is on, cannot record (edit defaults & reboot)
-61	ERR_ALSLEWMANY	Too many slewing iterations performed -- check transports
-62	ERR_EOD	Encountered end of data (EOD)
-63	ERR_SELF1INPROG	Operation not allowed while self-test1 in progress
-90	ERR_TCPUNEX	Unexpected response code from transport(s)
-91	ERR_TCPBADLEN	Wrong returned packet length from transport(s)
-92	ERR_TCPTIMEOUT	Transport read attempt timed out
-93	ERR_TCPDEAD	One or more transports are dead
-94	ERR_TCPOFF	One or more transports are off
-101	ERR_TCPOPFAIL	Transport command failed (non-specific error)
-104	ERR_TCPILCMD	Command not recognized by transport(s)?!
-105	ERR_TCPNOTIMP	Command not yet implemented by transport(s)
-106	ERR_TCPILSTATE	Transport(s) in illegal state for this operation

ROS Error Return Codes (continued)		
-110	ERR_TCPPGCHKSUM	Transport reported checksum error in program mode
-111	ERR_TCPPGFORMAT	Transport reported format error in program mode

108)   Response: RESP_STATE

    Msg Code: 108

        Data: byte1: recorder state code, one of:
                0x01   RSTATE_PLAY
                0x02   RSTATE_RECORD
                0x03   RSTATE_REWIND
                0x04   RSTATE_FF
                0x05   RSTATE_STOP
                0x06   RSTATE_PPAUSE (play-pause)
                0x07   RSTATE_RPAUSE (record-pause)
                0x08   RSTATE_CUE
                0x09   RSTATE_REVIEW
                0x0A   RSTATE_NOTAPE
                0x0B   RSTATE_POSITION
        (total data bytes = 1)

  Description: Indicates the current recorder state, which describes the S2's status with respect to tape motion. Not to be confused with the STATUS response.

110)   Response: RESP_SPEED

    Msg Code: 110

        Data: byte1: Speed code, one of:
                0x00   SP
                0x01   LP
                0x02   SLP
                0xFF   unknown
        (total data bytes = 1)

  Description: Indicates transport record or playback tape speed.

113)   Response: RESP_TIME

    Msg Code: 113

        Data: byte1, byte2: Absolute year number, byte1 = MSB
           byte3, byte4: Day number (1 to 365/366), byte3 = MSB
           byte5: Hours (0 to 23)
           byte6: Minutes (0 to 59)
           byte7: Seconds (0 to 59)
           byte8: Time validated flag
                0x00   time not validated/set
                0x01   time set
           (total data bytes = 8)

  Description: System or tape time and time validated flag (see TIME_READ, TIME_READ_PB commands).

116)   Response: RESP_MODE

    Msg Code: 116

        Data: byte1 ... : Mode designator in the form of an ASCII string terminated by a NULL (0) character, maximum 20 characters (not incl. NULL).
           (total data bytes = length of string + 1, max. 21)

  Description: Recorder mode designator (see MODE_SET, MODE_READ commands).

118)   Response: RESP_TAPEID

    Msg Code: 118

        Data: byte1 ... : ASCII Tape ID string terminated by a NULL (0) character, maximum 20 characters (not incl. NULL).
(total data bytes = length of string + 1, max. 21)

    Description: Tape ID value (see TAPEID_SET, TAPEID_READ, TAPEID_READ_PB commands).

121)   Response: RESP_USER_INFO

    Msg Code: 121

        Data: byte1 ... : ASCII info field or label string terminated by a NULL (0) character, maximum sizes (not incl. NULL) as follows:
        Field 1: 16 chars
        Field 2: 16 chars
        Field 3: 32 chars
        Field 4: 48 chars
        Label 1-4: 16 chars
(total data bytes = length of string + 1, max. 49)

    Description: User info field or label (see USER_INFO_SET, USER_INFO_READ, USER_INFO_READ_PB commands).

124)   Response: RESP_USER_DV

    Msg Code: 124

        Data: byte1: Boolean value,
        0x00   Record/playback user data-valid FALSE
        0x01   Record/playback user data-valid TRUE
byte2: Boolean value (always 0 for USER_DV_READ_PB),
        0x00   User data-valid flag ignored at playback
        0x01   User data-valid flag recognized at playback
(total data bytes = 2)

    Description: See USER_DV_READ and USER_DV_READ_PB commands.

127)   Response: RESP_GROUP

    Msg Code: 127

        Data: byte1: Transport group number (0–7)
byte2: Number of valid groups in current mode (1–8)
(total data bytes = 2)

    Description: See GROUP_READ command.

128)   Response: RESP_TAPEINFO

    Msg Code: 128

        Data: byte1 ... byte416: Table of information values for 8 tapes, with 52 bytes (= 1 row) per tape. For each tape (row of the table) the following information is given:

        rbyte1: address of transport containing tape, range 0–7 (numbers will be consecutive)

        rbyte2: channel ID (S2 internal channel number) of the tape's data, range 0–7. Equivalent to the transport address used at record time. If auxiliary data could not be extracted for this tape the channel ID will be 0xFF and all following fields (except possibly playback tape time) will contain zeroes.

        rbyte3 ... rbyte23: 20-character tape ID string, terminated by a null for 21 bytes total. Tape IDs shorter than 20 characters will leave trailing bytes unused (set to null) but will still occupy 21 bytes total.

rbyte24 ... rbyte33: 9-character recorder mode string, terminated by a null for 10 bytes total. Indicates what mode was in effect when the tape was recorded. Modes shorter than 9 characters will leave trailing bytes unused (set to null) but will still occupy 10 bytes total. In the rare case of an illegal mode at record time the mode string will be empty.

rbyte34 ... rbyte41: Playback tape time, 8 bytes total as follows (may be present even if channel ID is 0xFF):

rbyte34, rbyte35: Absolute year number (1901 to 2099), rbyte34 = MSB. A year value of 0 indicates time could not be extracted from aux data (remaining time bytes will be 0).

rbyte36, rbyte37: Day number (1 to 365/366), rbyte36 = MSB

rbyte38: Hours (0 to 23)

rbyte39: Minutes (0 to 59)

rbyte40: Seconds (0 to 59)

rbyte41: Frames (0 to 63, each frame is 1/64th of a second)

rbyte42 ... rbyte44: Record start time, 3 bytes total as follows:

rbyte42: Hours (0 to 23)

rbyte43: Minutes (0 to 59)

rbyte44: Seconds (0 to 59)

rbyte45, rbyte46: Record S2 serial number (0 to 65535, 0 = undefined), rbyte45 = MSB

rbyte47, rbyte48: Record transport serial number (1 to 65535), rbyte47 = MSB

rbyte49 ... rbyte52: 32-bit signed 2's complement measured record station delay value in nanoseconds, rbyte49 = MSB. Range: -500000000 to +(500000000-1bit), or 0x7FFFFFFF if unknown (no 1 Hz input).

(total data bytes = 416)

Description: Gives tape-related information during playback for each of the 8 individual transports. See the TAPEINFO_READ_PB command for further description.

130)   Response: RESP_DELAY

Msg Code: 130

Data: byte1 ... byte4: 32-bit signed 2's complement delay value in nanoseconds, byte1 = MSB. Range: -500000000 to +(500000000-1bit)

(total data bytes = 4)

Description: See DELAY_SET/DELAY_READ/DELAYM_READ.

133)   Response: RESP_BARRELROLL

Msg Code: 133

Data: byte1: Boolean value,

0x00   Barrel roll is OFF

0x00   Barrel roll is ON

(total data bytes = 1)

Description: See BARRELROLL_READ command.

136)   Response: RESP_POSITION

Msg Code: 136

Data: byte1: type of response,

0x00   overall (mid-point) position and linear variance

byte2 ... byte5: overall absolute position as a 32-bit signed integer number of seconds since BOT (MSB first, range -99–43199, -0x80000000 if unknown)

byte6 ... byte9: linear variance as a 32-bit signed integer
number of seconds (MSB first, range 0–43199,
-0x80000000 if unknown)

0x01 individual transport positions
byte2: number of positions being returned (normally 8)
byte3 ... byte34: List of positions. For each transport there is
a 32-bit signed integer representing the
absolute position in seconds since BOT
(MSB first, range -99–43199, -0x80000000 if
unknown, 0x7FFFFFFF if transport not selected)

(total data bytes = 9 (overall), 34 (individual))

Description: This is the response to the POSITION_READ command. There are two formats
as indicated by the first data byte: "overall" position, or "individual" positions.
See also POSITION_READ, POSITION_SET, ALIGN.

138) Response: RESP_ESTERR

Msg Code: 138

Data: byte1: number of strings in list (always 8 if ordered by transport, 1, 2, 4, or
8 depending on recorder mode if ordered by channel)
byte2 ...: list of up to 8 variable-length strings, each terminated by null, with
next string following immediately. Each string max. 12 bytes (incl. null).
(total data bytes = variable, max. 97)

Description: See ESTERR_READ command.

139) Response: RESP_PDV

Msg Code: 139

Data: byte1: number of strings in list (always 8 if ordered by transport, 1, 2, 4, or
8 depending on recorder mode if ordered by channel)
byte2 ...: list of up to 8 variable-length strings, each terminated by null, with
next string following immediately. Each string max. 12 bytes (incl. null).
(total data bytes = variable, max. 97)

Description: See PDV_READ command.

143) Response: RESP_SCPLL_MODE

Msg Code: 143

Data: byte1: SC PLL mode code
0x00 **xtal**
0x01 **manual**
0x02 **refclk**
0x03 **1hz**
0x04 **errmes**
(total data bytes = 1)

Description: Indicates the current System Clock PLL mode.

145) Response: RESP_TAPETYPE

Msg Code: 145

Data: byte1 ... : ASCII tape type string terminated by a NULL (0) character, maximum
6 characters (not incl. NULL). Note that all normal tape type codes are one character
long (0–9 or A–Z) but "unlisted" tape types are six digits long of the form
NNNMMM where NNN is the write current for LP and MMM is the write current
for SLP.
(total data bytes = length of string + 1, max. 7)

Description: Indicates the current tape type setting.

151) Response: RESP_MK3_FORM

Msg Code: 151

Data: byte1: Boolean value,

0x00   Mark III formatter disabled

0x01   Mark III formatter enabled

(total data bytes = 1)

Description: Indicates whether the S2's internal Mark III format generator is turned on or off. See also MK3_FORM_SET, MK3_FORM_READ.

155)   Response: RESP_TRANSPORT_TIMES

Msg Code: 155

Data: byte1: number of transports included in list (always 8).

byte2 ...: Table of transport times corresponding to transports 0 through 7 respectively. The set of times for each transport consists of 18 bytes and includes one 16-bit unsigned integer and four 32-bit unsigned integers as follows:

byte1 ... byte2: 16-bit unsigned transport serial number. A value of zero indicates the transport is dead or off or the serial number is unknown.

byte3 ... byte6: 32-bit unsigned total on-time since manufacture in minutes. A value of zero indicates unknown.

byte7 ... byte10: 32-bit unsigned total head-use time in minutes since manufacture. A value of zero indicates unknown.

byte11 ... byte14: 32-bit unsigned active head use time in minutes since the last head replacement. A value of zero indicates unknown.

byte15 ... byte18: 32-bit unsigned active head use time in minutes since the last service operation. A value of zero indicates unknown.

(total data bytes = 145)

Description: Response from TRANSPORT_TIMES command.

160)   Response: RESP_STATION_INFO

Msg Code: 160

Data: byte1: Station number, 0 to 255

byte2, byte3: System serial number (0 to 65535), byte2 = MSB

byte4 ... : ASCII nickname string terminated by a NULL (0) character, maximum 8 characters (not incl. NULL).

(total data bytes = length of string + 4, max. 12)

Description: Provides information which identifies the S2 system.

171)   Response: RESP_POSTIME

Msg Code: 171

Data: byte1, byte2: Absolute year number, byte1 = MSB

byte3, byte4: Day number (1 to 365/366), byte3 = MSB

byte5: Hours (0 to 23)

byte6: Minutes (0 to 59)

byte7: Seconds (0 to 59)

byte8: Frame number (0 to 63)

byte9 ... byte12: absolute tape position as a 32-bit signed integer number of seconds since BOT (MSB first, range -99–43199, -0x80000000 if unknown)

(total data bytes = 12)

Description: Response from POSTIME_READ command. Includes playback tape time and tape position of a particular transport.

180)   Response: RESP_STATUS   (generic)

Msg Code: 180

Data: byte1: Summary byte,

bit0: 1 = one or more error conditions in report
(always set if bit 1 is set)

0 = no error conditions in report

bit1: 1 = one or more fatal error conditions in report,
recommend abort current operation

0 = no fatal error conditions in report

bit2: 1 = one or more clear-on-read conditions in report

0 = no clear-on-read conditions in report

bit3 ... 7: reserved

byte2: Number of status entries (max. 32)

byte3 ... : List of status entries. For each entry there are 2 bytes, as follows:

1st byte: Status code number from 1 to 127 (RCL_STAT-
CODE_MAX), as defined in table below.

2nd byte: Status type,

bit0: 1 = error condition ("E" in table),

0 = non-error condition (always set if bit 1 is set)

bit1: 1 = fatal error condition ("F" in table),

0 = non-fatal condition

bit2: 1 = clear-on-read condition ("C" in table),

0 = non-clear-on-read.

bit3 ... 7: reserved

(total data bytes = # of status entries * 2 + 2, max. 130)

Description: Brief status report from the STATUS command. This lists the currently active S2 status conditions by their numeric codes. To obtain additional information such as text messages you should send a STATUS_DETAIL RCL command. There are 3 types of status codes: *informational, error,* and *fatal error*. Each of these may additionally be designated *clear-on-read*, which means they are automatically cleared immediately following the status request. Conditions of any type which appear and disappear between status requests are not lost; they are latched until the next status request. Status should be read regularly (every 1 to 5 seconds) to ensure that information is up to date.

The status summary byte indicates whether any errors or fatal errors exist. Overall system status can be considered "OK" if the summary error bit is clear, "cautionary" if the error bit is set but the fatal-error bit is clear, and "failed" if the fatal-error bit is set (fatal errors imply that the error bit is set too). If any fatal error conditions appear at any time (including clear-on-read conditions), there is a good chance that the system is not operating correctly, and immediate corrective action should be taken. Note that the S2 will not automatically abort any operation, even in the presence of fatal errors, unless there is no possible way to continue or there is imminent danger of hardware damage.

Status codes, mnemonics, and messages are given in the table below. Code numbers range from 1 to 127 (RCL_STATCODE_MAX). The text message shown is the "long" (RCL) version of the status message as would be returned by STATUS_DETAIL (or STATUS_DECODE) without the 'short' option. For a list of the short (console) messages see Section 4.5 of the User's Manual. This information is subject to change at any time, typically for the purpose of adding new status entries. All efforts will be made to avoid changing existing status code numbers, but such changes cannot be ruled out, especially since status codes are ranked by priority and some renumbering may be necessary when new codes are added. The mnemonics are unlikely to change. If possible, user software should not rely on the numeric status codes, but use the status *mnemonic* instead. To interpret a status code, user software should decode it using STATUS_DETAIL or STATUS_DECODE, and then take action based on the mnemonic (a string of the form STAT_*). The mnemonic appears at the start of the status message and can be ex-

tracted by reading all characters up to but not including the first colon. The message which follows can be written to the screen or recorded in a log file. To avoid having to issue extra RCL commands whenever testing status code mnemonics, user software should build a translation table in advance using repeated calls to STATUS_DECODE.

			Status Response Codes ("E" indicates error, "F" indicates fatal error, "C" indicates clear-on-read)		
1	STAT_DIAGIP	---	Diagnostic test 'xxx' is in progress.
2	STAT_DIAGDONE	--C	Diagnostic test completed successfully.
3	STAT_DIAGFAIL	EFC	Diagnostic test failed, diag/error code xxx: xxx
4	STAT_VMEPOWFAIL	EF-	VME xxx power supply voltage out of range.
5	STAT_NOFORM	EF-	Formatter board marked not present.
6	STAT_NODRD	EF-	DRD xxx marked not present.
7	STAT_NOUIC	EF-	UI board marked not present.
8	STAT_FORMFAIL	EF-	Formatter board failure (reason follows).
9	STAT_DRDFAIL	EF-	DRDxxx failure (reason follows).
11	STAT_UICFAIL	EF-	UI board failure (reason follows).
12	STAT_TRANFAIL	EF-	Transport xxx failure (reason follows).
13	STAT_ROSSWFAIL	EF-	ROS software failure (reason follows).
15	STAT_SRAMCHKSUM	EF-	SRAM board bad checksum for partition xxx.
16	STAT_NOSRSYR	EF-	System internal 64 Hz frame sync (SRSYR) is absent!
17	STAT_NOFG1	EF-	System internal 1 Hz tick (FG1) is absent!
18	STAT_TASKFAIL	EF-	Following required software tasks are not running: xxx
19	STAT_TRANSWFAIL	EF-	Transport xxx software failure: xxx
20	STAT_TRANTPMOTFAIL	EFC	Transport xxx tape motion sanity check failed
22	STAT_TRANBADTEMP	EF-	Transport xxx internal temperature is too high or too low for safe operation. Transport has shut down.
23	STAT_TRANWARNTEMP	E--	Warning: Transport xxx internal temperature is almost too high or too low for safe operation, will shut down above 60/below 5 deg C (check TAM fans and filters).
24	STAT_TRANDEAD	EF-	Required transports xxx are off or dead.
25	STAT_TRANBADVOLT	EF-	Transport xxx +5V/-5V supply is bad (check C4 cable).
26	STAT_TRANBADRECSIG	EF-	Transport xxx record clk/data is bad (check C3 cable).
27	STAT_ILMODE	EF-	Recorder mode is illegal, probably due to error during last mode change.
28	STAT_NOREFCLK	EF-	User reference high-rate clock (xxx) is absent.
29	STAT_NOREF1HZ	EF-	User reference xxx sync is absent.
32	STAT_BADSTNDELAY	E--	Station delay measurement does not match setting.
33	STAT_UICVERFAIL	E-C	UI verifier shows failure condition. In c1test mode, this indicates C1 cable connection fault.
34	STAT_SYNCERR1HZ	E-C	1 Hz sync error has occurred.
35	STAT_NESTFRINT	E-C	Nested frame interrupt has occurred.
36	STAT_LATEFRINT	E-C	Late frame interrupt has occurred.
37	STAT_SCPLLBADMODE	E--	System-clock PLL is set to the wrong mode (currently 'xxx', should be 'xxx').
38	STAT_SCPLLUNLOCK	EF-	System-clock PLL is unlocked.
39	STAT_SCPLLSLIP	E-C	System-clock PLL sync error has occurred.
40	STAT_DRDMISSIG	E-C	DRDxxx one or more missed/extra signatures detected.
41	STAT_FORMDFIFOTERR	E-C	Formatter data-FIFO timing error has occurred.

		Status Response Codes (continued)	
		("E" indicates error, "F" indicates fatal error, "C" indicates clear-on-read)	
42	STAT_MK3PLLSLIP	E-C	Mk3 PLL sync error has occurred.
43	STAT_MK3MISSYNC	E-C	Mk3 sync-detect miss has occurred.
44	STAT_MK3DFIFOTERR	E-C	Mk3 data-FIFO timing error has occurred.
45	STAT_TRANSYNCERR32	E-C	Transport(s) xxx have had 32 Hz sync errors.
46	STAT_REFCLKOUTLOW	EF-	C2 output ref clock rate is too low for current mode.
47	STAT_ERRRCL	EFC	RCL error xxx: xxx
48	STAT_ERRRPMON	EFC	Record/Playback monitor error xxx: xxx
49	STAT_ERRMK3SW	EFC	Mk3 formatter software error xxx: xxx
51	STAT_RPMONBOTEOT	EFC	Encountered xxx during automatic xxx (transport xxx). All tapes have been stopped.
52	STAT_RPMONILSTATE	EFC	Encountered unexpected transport state change during automatic xxx (state: xxx).
53	STAT_TRANLKFAIL	EF-	Transport xxx head-drum servo failed to lock.
54	STAT_TRANLKLOSS	E-C	Transport xxx servo lost lock or glitched momentarily.
55	STAT_TRANTRK1FAIL	EF-	Transport xxx stuck in phase-1 tracking calibration.
56	STAT_ILSETUP	E--	Bad setup: xxx
57	STAT_INSTATE	EF-	Tape motion state is inconsistent or illegal.
58	STAT_BADTRANSEL	EF-	Transport selection is incorrect for current mode & group.
59	STAT_CHIDDUP	EF-	Transport xxx and xxx tapes both have the same channel ID (xxx). Tapes are probably from different tape sets.
60	STAT_ALIGNFAIL	EF-	Unable to align tapes, fatal error xxx: xxx
61	STAT_ERRALIGN	E-C	Error aligning tapes, code xxx: xxx\n Trying alignment again.
62	STAT_CHIDMISS	EF-	Tape with channel ID xxx is missing.
63	STAT_BADGROUP	EF-	xxxxxxxxx is not part of current group.
64	STAT_CHIDMISMATCH	E--	xxx channel ID does not match current setup.
65	STAT_NOPBAUX	E--	Can't decode xxx from transport xxx.
66	STAT_BADPBTRANSEL	EF-	Can't switch to required playback mode xxx, wrong # of transports selected.
67	STAT_INTAPEID	E--	Transports xxx and xxx have different tape IDs.
68	STAT_BADPBESTERR	E--	Transport xxx estimated error rate greater than xxx.
69	STAT_RECBADSYNC	EF-	Transport xxx record data has bad sync.
70	STAT_RECBADAUX	EF-	Transport xxx record data has bad auxiliary data.
71	STAT_RECBADCHID	EF-	Transport xxx record data has wrong channel ID.
72	STAT_BADAGCLEV	EF-	Transport xxx AGC levels are outside acceptable range.
73	STAT_MARGAGCLEV	---	Transport xxx AGC levels are marginal but still within acceptable range.
74	STAT_ERRLOG	E-C	Log file error xxx: xxx
75	STAT_ERRDRCONFIG	E-C	Error xxx in DRDxxx data-recovery config: xxx
76	STAT_ERRDRTRACK	E-C	Error xxx in tran xxx DR tracking calibration: xxx
77	STAT_DELAYFIX	--C	Station delay auto resets: xxx
81	STAT_NOALIGN	E--	Tapes playing but not aligned (playback data not valid).
84	STAT_ALIGNIP	---	Tape alignment operation in progress.
85	STAT_ALIGNDONE	--C	Tape alignment operation completed successfully.
89	STAT_TAPESYNCERR	E-C	Tape-time sync error has occurred.
92	STAT_SCPLLDACRAIL	---	System clock PLL DAC at or near rail value.
93	STAT_SCPLLACCRAIL	---	System clock PLL accumulator at or near rail value.
94	STAT_DRTAPRAIL	---	DRDxxx data-recovery taps at or near rail value [xxx].

			Status Response Codes (continued) (**"E" indicates error, "F" indicates fatal error, "C" indicates clear-on-read**)
95	STAT_DREQSETRAIL	---	DRDxxx data-recovery EqSet at or near rail value.
96	STAT_DRFSETRAIL	---	DRDxxx data-recovery FSet at or near rail value.
97	STAT_TRANDIEDBORN	E-C	Transports died/born (xxx/xxx)
98	STAT_TRANPKTRETRY	--C	Transport packet retries (xxx)
99	STAT_DRFSETFADJ	--C	DRDxxx data-recovery fine FSet adjustments (xxx)

181)   Response: RESP_STATUS_DETAIL   (generic)

Msg Code: 181

Data: byte1: Summary byte,

bit0: 1 = one or more error conditions in report
(always set if bit 1 is set)
0 = no error conditions in report

bit1: 1 = one or more fatal error conditions in report,
recommend abort current operation
0 = no fatal error conditions in report

bit2: 1 = one or more clear-on-read conditions in report
0 = no clear-on-read conditions in report

bit3 ... 7: reserved

byte2: Number of status entries (max. 32)

byte3 ... : List of status entries. For each entry there are 2 bytes followed by a variable-length string, as follows:

byte1: Status code number from 1 to 127 (RCL_STATCODE_MAX).

byte2: Status type,

bit0: 1 = error condition ("E" in table),
0 = non-error condition (always set if bit 1 is set)

bit1: 1 = fatal error condition ("F" in table),
0 = non-fatal condition

bit2: 1 = clear-on-read condition ("C" in table),
0 = non-clear-on-read.

bit3 ... 7: reserved

byte3 ... : Status message in the form of an ASCII string terminated by a NULL (0) character, maximum 399 chars (not incl. NULL, may contain "\n" to indicate line breaks), or maximum 34 chars with "short" option.

(total data bytes = variable, max. 2000)

Description: Returns a detailed status description. This is like RESP_STATUS, but includes a text message for each status entry. Unlike RESP_STATUS_DECODE, this message may include information specific to this particular occurrence of the status condition, such as a list of affected transports or an error code. If the "short" option was used in the command, all messages are shortened to a single line of max. 34 characters (not incl. NULL). See RESP_STATUS response and STATUS_DETAIL command for more information.

182)   Response: RESP_STATUS_DECODE   (generic)

Msg Code: 182

Data: byte1 ... : Status message in the form of an ASCII string terminated by a NULL (0) character, maximum 399 characters (not incl. NULL), or max 34 characters for "short" option. Occurrence-specific string substitution information is marked by "xxx". May contain "\n" to indicate line breaks (long version only).
(total data bytes = length of string + 1, max. 400 (35 with short option))

Description: Returns a message describing a status code. For currently active status codes, STATUS_DETAIL should be used instead since it gives more specific information.

183) Response: RESP_ERROR_DECODE (generic)

Msg Code: 183

Data: byte1 ... : Error message in the form of an ASCII string terminated by a NULL (0) character, maximum 100 characters (not incl. NULL).
(total data bytes = length of string + 1, max. 101)

Description: Returns a message describing an error code.

191) Response: RESP_BERDCB

Msg Code: 191

Data: byte1 ... byte4: Number of incorrect bits during the measurement interval in the case of FORM BER and UI BER, or number of 1-bits in the case of UI DC-bias. This is a 32-bit unsigned integer value (MSB first).
byte5 ... byte8: Total number of bits measured in the time interval as a 32-bit unsigned integer (MSB first).
(total data bytes = 8)

Description: Gives the results of a measurement performed by the BERDCB command. To calculate the bit-error rate or the DC-bias ratio, divide the first 32-bit integer by the second 32-bit integer.

192) Response: RESP_DATA_EXTRACT

Msg Code: 192

Data: byte1 ... byte3: Extracted data or validity bytes.
byte4: Parity information for MkIII data extract only,
    where bit value 0 means NO parity error,   value 1 means that
a parity error occurred.
      bit 0: parity bit for byte1
      bit 1: parity bit for byte2
      bit 2: parity bit for byte3
    For S2 data/validity extract byte4 is always 0.
(total data bytes = 4)

Description: Returns three bytes of extracted data or validity from a DATA_EXTRACT_READ command. A fourth byte gives the parity bit for each of the three data bytes in its lowest three bits (Mark III data extract only). The time ordering of the extracted data or validity bits is from the most-significant bit of byte1 (bit 7), to the least-significant bit of byte1 (bit 0), followed by the most-significant bit of byte2 (bit 7), to the least-significant bit of byte2 (bit 0), followed by the most-significant bit of byte3 (bit 7), to the least-significant bit of byte3 (bit 0).

197) Response: RESP_IDENT (generic)

Msg Code: 197

Data: byte1 ... : RCL device type identifier string terminated by a NULL (0) character, maximum 10 characters (not incl. NULL).
(total data bytes = length of string + 1, max. 11)

Description: Gives the RCL device type, which determines what set of commands it responds to (see the IDENT command).

199)   Response: RESP_VERSION   (generic)

Msg Code: 199

Data: byte1 ... : ROS software version information string, terminated by a NULL
(0) character, maximum 60 characters (not incl. NULL).
(total data bytes = length of string + 1, max. 61)

Description: Gives ROS software version information. Includes CPU type in square brackets.
Example string:
  3.0d [162] (compiled Fri May 16 12:51:58 EDT 1997)
See the description of the VERSION command for more information.

APPENDIX B

S2 SIGNAL AND CONTROL INTERFACE SPECIFICATIONS

B.1   Overview

This appendix describes the physical, electrical, and timing characteristics of the signal and control cables connecting the S2 record and playback system to the user's hardware.

B.1.1   S2 Signal Interface

The S2 record and playback signal interface consists of up to four cables connecting to ports on the UI board:

1. C1: Input user data on 16 signal lines, input clock and 1Hz.

2. C2a: Output user data on 16 signal lines, output clock and 1Hz; and (optional) input clock and 1Hz for playback timing control.

3. C2b: 8 signals of output data validity representing the even user data channels (0,2,4,6,8,10,12,14) in modes with 1-bit quantization or pairs of user channels in modes with 2-bit quantization. In the case of 2-bit quantization, each signal represents combined data validity for both sign and amplitude channels of the same and next higher channel number, e.g. DV OUT 4 contains combined data validity for both user data channels 4 and 5.

4. C2c: 8 signals of output data validity representing the odd user channels (1,3,5,7,9,11,13,15) in modes with 1-bit quantization. Required only for 1-bit modes that use odd user data channels, which includes only S2 modes of the form **b**x**u**-1 where **u** is greater than 1. Note that the C2c port is not located on the UI board's front panel, instead it is a 20-pin header located on the PC board just behind the C2b port.

B.1.2   S2 Control Interface

The S2 control interface consists of two cables:

1. C6: RS-232C RCL serial control cable, connecting to a port on the Formatter board.

2. C7: RS-232C console terminal cable, connecting to a port on the RCC.

Equivalent control connections can be made over the Ethernet port using network rlogin/telnet and socket protocols, eliminating the need for serial cables (see Section 4.4).

B.2   S2 Signal Interface

B.2.1   Signal Assignments

The input signal interface consists of a single multiconductor twisted pair cable C1 terminating in a connector P1 at the data source. The C1 cable carries input data and clock signals as listed in Table B-1. Table B-5 shows which of the 16 available input data channels IN 0 through IN 15 are active in each S2 mode.

The output signal interface consists of three multiconductor twisted-pair equal-length cables C2a, C2b, and C2c (jointly designated as cable C2) terminating in connectors P2a, P2b, and P2c at the data destination. The C2 cable carries signals listed in Tables B-2 and B-3. Cable C2a (Table B-2) carries the output clock and 1Hz, and reproduced data, while cables C2b and C2c (Table B-3) carry output data-validity (DV) signals. In the S2's "Mark III/IV compatible" mode, the C2a cable delivers NRZM-encoded signals compatible with Mark III/IV equipment (Table B-4). Use the `mk3 form enable` command to switch the C2a cable into Mark III/IV compatible mode. Table B-5 shows which of the 16 available output data/DV channels OUT 0 through OUT 15 and DV OUT 0 through DV OUT 15 are active in each S2 mode. In Mark III/IV compatible operation the same C2a data output pins are active, but channels 14 and 15 are not available.

B.2.1.1   Signal Characteristics

The timing of signals on cable C1 as measured at the signal destination (S2 input) with appropriate termination is shown in Figures B-2 and B-3 for fixed 32 MHz clock and data-rate clock cases, respectively. Timing of signals on cable C2 as measured at the signal destination (Correlator input) with appropriate termination is shown in Figures B-4, B-5, and B-6 for fixed 32 MHz clock, data-rate clock, and MkIII/IV output mode cases, respectively. Note that in the MkIII/IV output case the phase of the generated clock (S-CLK_PB) can be selected using the `mk3 form clkinv` console command depending on the requirements of the destination equipment.

B.2.1.2   Electrical Characteristics

All signals are to be sourced by terminated ECL drivers; i.e. all signals listed in Tables B-1 through B-4 are to be transmitted through C1 and C2 in both true and inverted polarity along twisted pairs. In the receiver, each twisted pair is terminated differentially in 100 ohms as shown in Figure B-7.

B.2.1.3   Physical Characteristics

The C1 cable is to be Belden type 9V28050 50-conductor twisted pair cable or equivalent, and connector P1 is to be 50-pin Scotch header part number 3433-5302; socket part number 3425-6600 with strain relief clips part number 3448-3050 or equivalent.

The C2a cable is to be Belden type 9V28050 50-conductor twisted pair cable or equivalent, and connector P2a is to be 50-pin Scotch header part number 3433-5302; socket part number 3425-6600 with strain relief clips part number 3448-3050 or equivalent. The C2b and C2c cables are to be Belden type 9V28020 20-conductor twisted pair cable or equivalent and connectors P2b and P2c are to be 20-pin Scotch header part number 3428-5302; socket part number 3421-6600 with strain relief clips part number 3448-3020 or equivalent. Signal pin assignments are shown in Table B-2. The conductors in C2 not used for signal are to be grounded near the signal source. The length of the cable should not exceed 10 meters.

B.2.1.4   Input/Output Cable Connection Test Modes

In C1 and C2 test modes, cables C1 and C2 carry test data generated by the test vector generator circuit illustrated in Figure B-1.

The C1 test data is to be sourced by hardware resident in the user's system and should be substituted for normal record data during C1 test. The user should select S2 mode **c1test16** (16 Mbit/s data over all 16 input signal

lines) and examine system status to ensure that message STAT_UICVERFAIL is not present. The C1 test modes are available at various data rates (4, 8, 16, 32 Mbits/s). The (identical) C2 test data is substituted by the S2 for normal playback data when the S2 mode is set to **c2test16** (16 Mbit/s data over all 16 output signal lines). The C2 test modes are available at various data rates (4, 8, 16, 32 Mbits/s).

The test vector generator is driven by the data rate clock and initialized internally by a (data rate width) 1Hz signal. Table B-7 shows the first 20 words of test vector data following a synchronous 1Hz preset at sample clock period t = 0. Although the S2 supports C1 and C2 test modes at data rates of 4, 8, 16, and 32 Mb/s per data line, normally only the 16 Mb/s data rate test modes will be used. Figures B-2 and B-4 illustrate the timing of the test vectors relative to the other signals on the C1 and C2 cables.

B.3   S2 Control Interface

B.3.1   RCL Serial Link (C6 Cable)

The RCL serial interface is standard RS-232C. The link between the S2 and the external control computer may be either RS-232C, or multi-drop RS-485 with the addition of an RS-232C to RS-485 conversion box (supplied by the user) external to the S2. Use of RS-232C requires a dedicated serial port on the external control computer for each S2 controlled, while multi-drop RS-485 operation allows many S2's to be controlled from a single external control computer serial port. Alternatively Internet socket connections can be used instead of serial cables if the S2 is connected to an Ethernet network.

The S2 RCL port is a male DB-25 connector configured for DTE (Data Terminal Equipment) operation. The baud rate is selectable at standard values between 2400 and 57600 baud (default 19200 baud). The maximum distance between the S2 and the external control computer should not exceed 15 meters. Table B-6 shows the pin assignments for the DB-25 connector. Note that the RTS signal is asserted high (TRUE) just before each response message transmitted by the S2 and asserted low following the transmission. Thus the RTS signal can be used in multi-drop RS-485 interfacing to switch on and off the RS-485 tri-state line drivers. The CTS, DCD and DTR signals are not used.

B.3.2   RCC Console Cable (C7 Cable)

C7 is a standard RS-232C serial cable which connects to a female DB-25 connector on the RCC configured as DCE (Data Communications Equipment). Refer to the MVME133XT User's Manual for pin assignments.

PIN#	SIG. NAME	I/O	DESCRIPTION	PIN#	SIG. NAME	I/O	DESCRIPTION
1	GND		Ground	26	$\overline{\text{IN 9}}$	I	S2 Input Data (ECL)
2	GND		Ground	27	IN 10	I	S2 Input Data (ECL)
3	IN 0	I	S2 Input Data (ECL)	28	$\overline{\text{IN 10}}$	I	S2 Input Data (ECL)
4	$\overline{\text{IN 0}}$	I	S2 Input Data (ECL)	29	IN 11	I	S2 Input Data (ECL)
5	IN 1	I	S2 Input Data (ECL)	30	$\overline{\text{IN 11}}$	I	S2 Input Data (ECL)
6	$\overline{\text{IN 1}}$	I	S2 Input Data (ECL)	31	GND		Ground
7	IN 2	I	S2 Input Data (ECL)	32	GND		Ground
8	$\overline{\text{IN 2}}$	I	S2 Input Data (ECL)	33	IN 12	I	S2 Input Data (ECL)
9	IN 3	I	S2 Input Data (ECL)	34	$\overline{\text{IN 12}}$	I	S2 Input Data (ECL)
10	$\overline{\text{IN 3}}$	I	S2 Input Data (ECL)	35	IN 13	I	S2 Input Data (ECL)
11	GND		Ground	36	$\overline{\text{IN 13}}$	I	S2 Input Data (ECL)
12	GND		Ground	37	IN 14	I	S2 Input Data (ECL)
13	IN 4	I	S2 Input Data (ECL)	38	$\overline{\text{IN 14}}$	I	S2 Input Data (ECL)
14	$\overline{\text{IN 4}}$	I	S2 Input Data (ECL)	39	IN 15	I	S2 Input Data (ECL)
15	IN 5	I	S2 Input Data (ECL)	40	$\overline{\text{IN 15}}$	I	S2 Input Data (ECL)
16	$\overline{\text{IN 5}}$	I	S2 Input Data (ECL)	41	GND		Ground
17	IN 6	I	S2 Input Data (ECL)	42	GND		Ground
18	$\overline{\text{IN 6}}$	I	S2 Input Data (ECL)	43	SCLK_REC	I	Input Reference Clock (ECL)
19	IN 7	I	S2 Input Data (ECL)	44	$\overline{\text{SCLK\_REC}}$	I	Input Reference Clock (ECL)
20	$\overline{\text{IN 7}}$	I	S2 Input Data (ECL)	45	GND		Ground
21	GND		Ground	46	GND		Ground
22	GND		Ground	47	S1HZ_REC	I	Input 1Hz Reference (ECL)
23	IN 8	I	S2 Input Data (ECL)	48	$\overline{\text{S1HZ\_REC}}$	I	Input 1Hz Reference (ECL)
24	$\overline{\text{IN 8}}$	I	S2 Input Data (ECL)	49	GND		Ground
25	IN 9	I	S2 Input Data (ECL)	50	GND		Ground

Table B-1: C1 Input Cable Signal Assignments

PIN#	SIGNAL NAME	I/O	DESCRIPTION
1	GND		Ground
2	GND		Ground
3	OUT 0	O	S2 Output Data (ECL)
4	OUT 0	O	S2 Output Data (ECL)
5	OUT 1	O	S2 Output Data (ECL)
6	OUT 1	O	S2 Output Data (ECL)
7	OUT 2	O	S2 Output Data (ECL)
8	OUT 2	O	S2 Output Data (ECL)
9	OUT 3	O	S2 Output Data (ECL)
10	OUT 3	O	S2 Output Data (ECL)
11	OUT 4	O	S2 Output Data (ECL)
12	OUT 4	O	S2 Output Data (ECL)
13	OUT 5	O	S2 Output Data (ECL)
14	OUT 5	O	S2 Output Data (ECL)
15	OUT 6	O	S2 Output Data (ECL)
16	OUT 6	O	S2 Output Data (ECL)
17	SCLK_PB	O	Playback Sample 32 MHz Clock (ECL)
18	SCLK_PB	O	Playback Sample 32 MHz Clock (ECL)
19	OUT 7	O	S2 Output Data (ECL)
20	OUT 7	O	S2 Output Data (ECL)
21	OUT 8	O	S2 Output Data (ECL)
22	OUT 8	O	S2 Output Data (ECL)
23	OUT 9	O	S2 Output Data (ECL)
24	OUT 9	O	S2 Output Data (ECL)
25	OUT 10	O	S2 Output Data (ECL)
26	OUT 10	O	S2 Output Data (ECL)
27	OUT 11	O	S2 Output Data (ECL)
28	OUT 11	O	S2 Output Data (ECL)
29	OUT 12	O	S2 Output Data (ECL)
30	OUT 12	O	S2 Output Data (ECL)
31	OUT 13	O	S2 Output Data (ECL)
32	OUT 13	O	S2 Output Data (ECL)
33	NC		No Connection
34	NC		No Connection
35	OUT 14	O	S2 Output Data (ECL)
36	OUT 14	O	S2 Output Data (ECL)
37	OUT 15	O	S2 Output Data (ECL)
38	OUT 15	O	S2 Output Data (ECL)
39	GND		Ground
40	GND		Ground
41	GND		Ground
42	GND		Ground
43	CCLK	I	Correlator 32 MHz Reference Clock (ECL)
44	CCLK	I	Correlator 32 MHz Reference Clock (ECL)
45	C1HZ	I	Correlator 1Hz Reference (ECL)
46	C1HZ	I	Correlator 1Hz Reference (ECL)
47	S1HZ_PB	O	Playback 1Hz Reference (ECL)
48	S1HZ_PB	O	Playback 1Hz Reference (ECL)
49	NC		No Connection
50	NC		No Connection

Table B-2: S2 C2a Output Cable Signal Assignments

PIN#	SIGNAL NAME	I/O	DESCRIPTION
1	GND		Ground
2	GND		Ground
3	DV OUT 0	O	S2 Output Data Validity (ECL)
4	$\overline{\text{DV OUT 0}}$	O	S2 Output Data Validity (ECL)
5	DV OUT 2	O	S2 Output Data Validity (ECL)
6	$\overline{\text{DV OUT 2}}$	O	S2 Output Data Validity (ECL)
7	DV OUT 4	O	S2 Output Data Validity (ECL)
8	$\overline{\text{DV OUT 4}}$	O	S2 Output Data Validity (ECL)
9	DV OUT 6	O	S2 Output Data Validity (ECL)
10	$\overline{\text{DV OUT 6}}$	O	S2 Output Data Validity (ECL)
11	GND		Ground
12	GND		Ground
13	DV OUT 8	O	S2 Output Data Validity (ECL)
14	$\overline{\text{DV OUT 8}}$	O	S2 Output Data Validity (ECL)
15	DV OUT 10	O	S2 Output Data Validity (ECL)
16	$\overline{\text{DV OUT 10}}$	O	S2 Output Data Validity (ECL)
17	DV OUT 12	O	S2 Output Data Validity (ECL)
18	$\overline{\text{DV OUT 12}}$	O	S2 Output Data Validity (ECL)
19	DV OUT 14	O	S2 Output Data Validity (ECL)
20	$\overline{\text{DV OUT 14}}$	O	S2 Output Data Validity (ECL)

(a) C2b

PIN#	SIGNAL NAME	I/O	DESCRIPTION
1	GND		Ground
2	GND		Ground
3	DV OUT 1	O	S2 Output Data Validity (ECL)
4	$\overline{\text{DV OUT 1}}$	O	S2 Output Data Validity (ECL)
5	DV OUT 3	O	S2 Output Data Validity (ECL)
6	$\overline{\text{DV OUT 3}}$	O	S2 Output Data Validity (ECL)
7	DV OUT 5	O	S2 Output Data Validity (ECL)
8	$\overline{\text{DV OUT 5}}$	O	S2 Output Data Validity (ECL)
9	DV OUT 7	O	S2 Output Data Validity (ECL)
10	$\overline{\text{DV OUT 7}}$	O	S2 Output Data Validity (ECL)
11	GND		Ground
12	GND		Ground
13	DV OUT 9	O	S2 Output Data Validity (ECL)
14	$\overline{\text{DV OUT 9}}$	O	S2 Output Data Validity (ECL)
15	DV OUT 11	O	S2 Output Data Validity (ECL)
16	$\overline{\text{DV OUT 11}}$	O	S2 Output Data Validity (ECL)
17	DV OUT 13	O	S2 Output Data Validity (ECL)
18	$\overline{\text{DV OUT 13}}$	O	S2 Output Data Validity (ECL)
19	DV OUT 15	O	S2 Output Data Validity (ECL)
20	$\overline{\text{DV OUT 15}}$	O	S2 Output Data Validity (ECL)

(b) C2c

Table B-3: S2 C2b/c Output Cable Signal Assignments

PIN#	SIGNAL NAME		I/O	DESCRIPTION
	ODD TRACKS	EVEN TRACKS		
1	GND	GND		Ground
2	GND	GND		Ground
3	TRACK 1	TRACK 2	O	MK III Format NRZM data output (ECL)
4	$\overline{\text{TRACK 1}}$	$\overline{\text{TRACK 2}}$	O	MK III Format NRZM data output (ECL)
5	TRACK 3	TRACK 4	O	MK III Format NRZM data output (ECL)
6	$\overline{\text{TRACK 3}}$	$\overline{\text{TRACK 4}}$	O	MK III Format NRZM data output (ECL)
7	TRACK 5	TRACK 6	O	MK III Format NRZM data output (ECL)
8	$\overline{\text{TRACK 5}}$	$\overline{\text{TRACK 6}}$	O	MK III Format NRZM data output (ECL)
9	TRACK 7	TRACK 8	O	MK III Format NRZM data output (ECL)
10	$\overline{\text{TRACK 7}}$	$\overline{\text{TRACK 8}}$	O	MK III Format NRZM data output (ECL)
11	TRACK 9	TRACK 10	O	MK III Format NRZM data output (ECL)
12	$\overline{\text{TRACK 9}}$	$\overline{\text{TRACK 10}}$	O	MK III Format NRZM data output (ECL)
13	TRACK 11	TRACK 12	O	MK III Format NRZM data output (ECL)
14	$\overline{\text{TRACK 11}}$	$\overline{\text{TRACK 12}}$	O	MK III Format NRZM data output (ECL)
15	TRACK 13	TRACK 14	O	MK III Format NRZM data output (ECL)
16	$\overline{\text{TRACK 13}}$	$\overline{\text{TRACK 14}}$	O	MK III Format NRZM data output (ECL)
17	Clock Out	Clock Out	O	MK III Tape clock (ECL)
18	$\overline{\text{Clock Out}}$	Clock Out	O	MK III Tape clock (ECL)
19	TRACK 15	TRACK 16	O	MK III Format NRZM data output (ECL)
20	$\overline{\text{TRACK 15}}$	$\overline{\text{TRACK 16}}$	O	MK III Format NRZM data output (ECL)
21	TRACK 17	TRACK 18	O	MK III Format NRZM data output (ECL)
22	$\overline{\text{TRACK 17}}$	$\overline{\text{TRACK 18}}$	O	MK III Format NRZM data output (ECL)
23	TRACK 19	TRACK 20	O	MK III Format NRZM data output (ECL)
24	$\overline{\text{TRACK 19}}$	$\overline{\text{TRACK 20}}$	O	MK III Format NRZM data output (ECL)
25	TRACK 21	TRACK 22	O	MK III Format NRZM data output (ECL)
26	$\overline{\text{TRACK 21}}$	$\overline{\text{TRACK 22}}$	O	MK III Format NRZM data output (ECL)
27	TRACK 23	TRACK 24	O	MK III Format NRZM data output (ECL)
28	$\overline{\text{TRACK 23}}$	$\overline{\text{TRACK 24}}$	O	MK III Format NRZM data output (ECL)
29	TRACK 25	TRACK 26	O	MK III Format NRZM data output (ECL)
30	$\overline{\text{TRACK 25}}$	$\overline{\text{TRACK 26}}$	O	MK III Format NRZM data output (ECL)
31	TRACK 27	TRACK 28	O	MK III Format NRZM data output (ECL)
32	$\overline{\text{TRACK 27}}$	$\overline{\text{TRACK 28}}$	O	MK III Format NRZM data output (ECL)
33	NC	NC		No Connection, +read after write Ch. A (MKIII)
34	NC	NC		No Connection, -read after write Ch. A (MKIII)
35	HIGH	HIGH	O	HIGH (ECL)
36	LOW	LOW	O	LOW (ECL)
37	HIGH	HIGH	O	HIGH (ECL)
38	LOW	LOW	O	LOW (ECL)
39	GND	GND		Ground
40	GND	GND		Ground
41	GND	GND		Ground
42	GND	GND		Ground
43	NC	NC	I	No Connection, ECL 100 ohm input termination
44	NC	NC	I	No Connection, ECL 100 ohm input termination
45	NC	NC	I	No Connection, ECL 100 ohm input termination
46	NC	NC	I	No Connection, ECL 100 ohm input termination
47	HIGH	HIGH	O	HIGH (ECL)
48	LOW	LOW	O	LOW (ECL)
49	NC	NC		No Connection, +read after write clock Ch. A (MKIII)
50	NC	NC		No Connection, -read after write clock Ch. A (MKIII)

Table B-4: MkIII C2a Output Cable Signal Assignments

S2 Mode		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
		Active User Data and Validity Channels (D=C1/C2a, DV=C2b/C2c)															
32x1-1	D	x															
	DV	x															
32x2-1	D	x	x														
	DV	x	x														
32x2-2	D	x	x														
	DV	x	0														
32x4-1	D	x	x	x	x												
	DV	x	x	x	x												
32x4-2	D	x	x	x	x												
	DV	x	0	x	0												
32a4-2	D	x	x			x	x										
	DV	x	0			x	0										
32b4-2	D			x	x			x	x								
	DV			x	0			x	0								
16x1-1	D	x															
	DV	x															
16x2-1	D	x	x														
	DV	x	x														
16x2-2	D	x	x														
	DV	x	0														
16x4-1	D	x	x	x	x												
	DV	x	x	x	x												
16x4-2	D	x	x	x	x												
	DV	x	0	x	0												
16i4-1	D	x		x						x		x					
	DV	x		x						x		x					
16x8-1	D	x	x	x	x	x	x	x	x								
	DV	x	x	x	x	x	x	x	x								
16x8-2	D	x	x	x	x	x	x	x	x								
	DV	x	0	x	0	x	0	x	0								
16i8-1	D	x		x		x		x		x		x		x		x	
	DV	x		x		x		x		x		x		x		x	
16p8-2	D	x	x	x	x					x	x	x	x				
	DV	x	0	x	0					x	0	x	0				
16v8-2	D	x	x	x	x	x	x	x	x								
	DV	x	0	x	0	x	0	x	0								
8x2-1	D	x	x														
	DV	x	x														
8x2-2	D	x	x														
	DV	x	0														
8x4-1	D	x	x	x	x												
	DV	x	x	x	x												
8x4-2	D	x	x	x	x												
	DV	x	0	x	0												
8i4-1	D	x		x						x		x					
	DV	x		x						x		x					
8x8-1	D	x	x	x	x	x	x	x	x								
	DV	x	x	x	x	x	x	x	x								
8x8-2	D	x	x	x	x	x	x	x	x								
	DV	x	0	x	0	x	0	x	0								
8i8-1	D	x		x		x		x		x		x		x		x	
	DV	x		x		x		x		x		x		x		x	
8p8-2	D	x	x	x	x					x	x	x	x				
	DV	x	0	x	0					x	0	x	0				
8x16-1	D	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
	DV	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
8x16-2	D	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
	DV	x	0	x	0	x	0	x	0	x	0	x	0	x	0	x	0
8d16-2	D	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
	DV	x	0	x	0	x	0	x	0	x	0	x	0	x	0	x	0
8n16-2	D	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
	DV	x	0	x	0	x	0	x	0	x	0	x	0	x	0	x	0
4x4-1	D	x	x	x	x												
	DV	x	x	x	x												
4x4-2	D	x	x	x	x												
	DV	x	0	x	0												
4x8-1	D	x	x	x	x	x	x	x	x								
	DV	x	x	x	x	x	x	x	x								
4x8-2	D	x	x	x	x	x	x	x	x								
	DV	x	0	x	0	x	0	x	0								

Table B-5: S2 Modes with active C1/C2 signal channels
(Continued on next page)

S2 Mode		Active User Data and Validity Channels (D=C1/C2a, DV=C2b/C2c)															
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
4i8-1	D	x		x		x		x		x		x		x		x	
	DV	x		x		x		x		x		x		x		x	
4p8-2	D	x	x	x	x					x	x	x	x				
	DV	x	0	x	0					x	0	x	0				
4x16-1	D	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
	DV	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
4x16-2	D	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
	DV	x	0	x	0	x	0	x	0	x	0	x	0	x	0	x	0

Note: x = active signal, 0 = driven low, blank = floating

Table B-5: S2 Modes with active C1/C2 signal channels (Continued from previous page)

PIN#	SIG. NAME	I/O	DESCRIPTION
2	TxD	O	Transmit Data
3	RxD	I	Receive Data
4	RTS	O	Request to Send
5	CTS	I	Clear to Send
7	GND		Signal Ground
8	DCD	I	Data Carrier Detect
20	DTR	O	Data Terminal Ready

Table B-6: C6 (RCL) Cable Signal Assignments

C2a CABLE TEST VECTOR SEQUENCE
(FIRST 20 WORDS FOLLOWING 1Hz RESET AT t=0)

Data Rate Clock Period	OUT[0]	OUT[1]	OUT[2]	OUT[3]	OUT[4]	OUT[5]	OUT[6]	OUT[7]	OUT[8]	OUT[9]	OUT[10]	OUT[11]	OUT[12]	OUT[13]	OUT[14]	OUT[15]
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
2	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
3	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0
4	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
5	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0
6	0	0	0	0	1	1	0	0	1	1	0	0	0	0	0	0
7	0	0	1	0	1	0	1	0	1	0	0	0	0	0	0	0
8	1	1	1	1	1	1	1	1	0	0	0	0	0	0	1	0
9	0	0	0	0	0	0	1	0	0	0	0	0	1	1	0	0
10	0	0	0	0	1	1	0	0	0	0	1	0	1	0	0	0
11	0	0	1	0	1	0	0	0	1	1	1	1	0	0	0	0
12	1	1	1	1	0	0	1	0	0	0	1	0	0	0	1	0
13	0	0	1	0	1	1	0	0	1	1	0	0	1	1	0	0
14	1	1	1	0	1	0	1	0	1	0	1	0	1	0	1	0
15	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1
16	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0
17	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0
18	0	0	0	0	0	0	0	0	1	1	0	1	1	0	0	0
19	0	0	0	0	0	0	1	0	1	1	1	0	0	0	0	0
20	0	0	0	0	1	1	1	0	1	1	1	0	0	0	0	0

Note: Shown are the first 20 words of test data following synchronous 1 Hz reset at sampler clock period t=0.

Table B-7: C2a Cable Test Vector Sequence

C2b/C2c CABLE TEST VECTOR SEQUENCES
(FIRST 20 WORDS FOLLOWING 1Hz RESET AT t=0)

Data Rate Clock Period	C2b OUT[0]	OUT[2]	OUT[4]	OUT[6]	OUT[8]	OUT[10]	OUT[12]	OUT[14]	C2c OUT[1]	OUT[3]	OUT[5]	OUT[7]	OUT[9]	OUT[11]	OUT[13]	OUT[15]
1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	1	0		0	0	0	0	0	0	1	0
3	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
4	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	0
5	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0
6	0	0	1	0	1	0	0	0	0	0	1	0	1	0	0	0
7	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0
8	1	1	1	1	0	0	0	1	1	1	1	1	0	0	0	0
9	0	0	0	1	0	0	1	0	0	0	0	0	0	0	1	0
10	0	0	1	0	0	1	1	0	0	0	1	0	0	0	0	0
11	0	1	1	0	1	1	0	0	0	0	0	0	1	1	0	0
12	1	1	0	1	0	1	0	1	1	1	0	0	0	0	0	0
13	0	1	1	0	1	0	1	0	0	0	1	0	1	0	1	0
14	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
15	0	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1
16	0	0	0	0	0	1	1	1	0	0	0	0	0	0	1	0
17	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0
18	0	0	0	0	1	0	1	0	0	0	0	0	1	1	0	0
19	0	0	0	1	1	0	0	0	0	0	0	0	1	1	0	0
20	0	0	1	1	1	1	0	0	0	0	1	0	1	0	0	0

Note: Shown are the first 20 words of test data following synchronous 1 Hz reset at sampler clock period t=0.

Table B-8: C2b/C2c Cable Test Vector Sequence

TEST VECTOR GENERATOR SYNCHRONOUS PRESET

SYNCHRONOUS PRESET

CLOCK

OUTi   t=0   t=1   t=2   t=3   t=4   t=5
                  TVG1   TVG2   TVG3   TVG4   TVG5

Q(all)=1 FOLLOWING SYNCHRONOUS PRESET.

FOR LISTING OF TVG1 to TVG20 VECTORS, REFER TO TEST VECTOR SEQUENCE TABLE.

Q(all)=1 FOLLOWING SYNCHRONOUS PRESET

- Q16 / 4LS = OUT14
- Q15 / 4UA = OUT13
- Q14 / 4US = OUT12
- Q13 / 3LA = OUT11
- Q12 / 3LS = OUT10
- Q11 / 3UA = OUT9
- Q10 / 3US = OUT8
- Q9 / 2LA = OUT7
- Q8 / 2LS = OUT6
- Q7 / 2UA = OUT5
- Q6 / 2US = OUT4
- Q5 / 1LA = OUT3
- Q4 / 1LS = OUT2
- Q3 / 1UA = OUT1
- Q2 / 1US = OUT0
- Q1 / 4LA = OUT15

CLOCK

SYNCHRONOUS PRESET (PRESET TO ALL 1'S)

RADIOASTRON IF SYSTEM INTERFACE DESIGNATION

Figure B-1: C1/C2 Test Vector Generator Logical Circuit
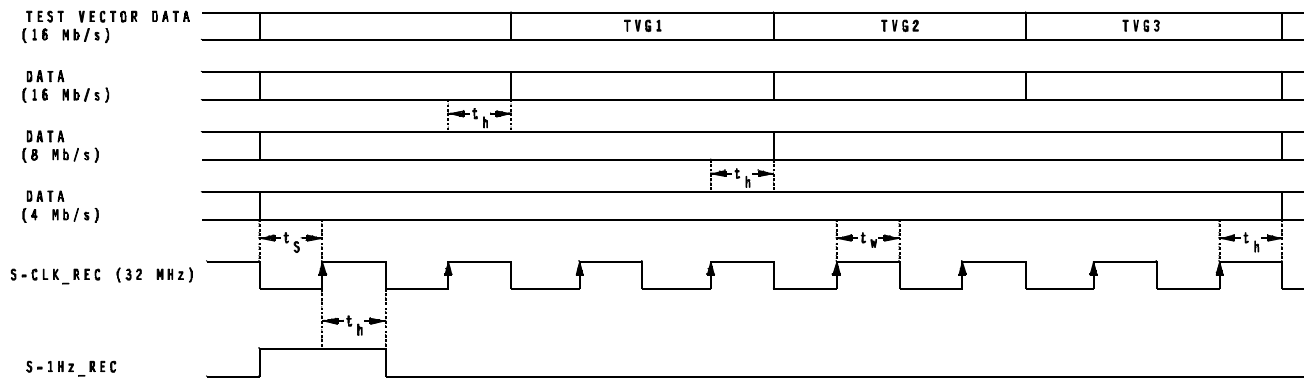
Figure B-2: C1 Input Timing Interface (32 MHz clk)
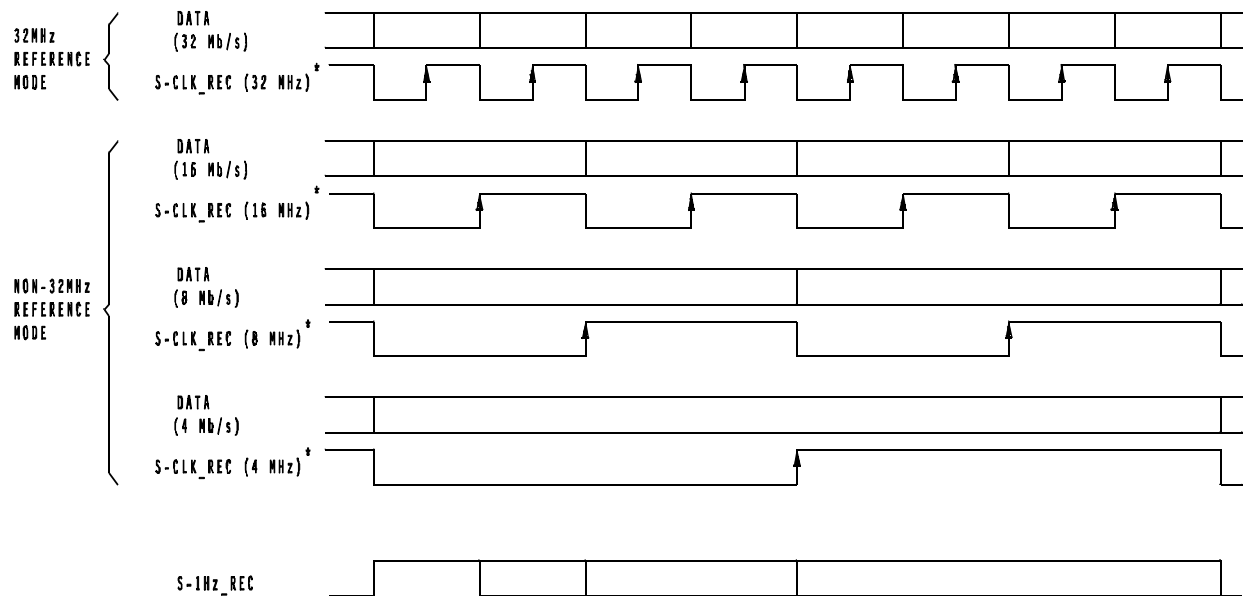
Figure B-3: C1 Input Timing Interface (clk = data rate)
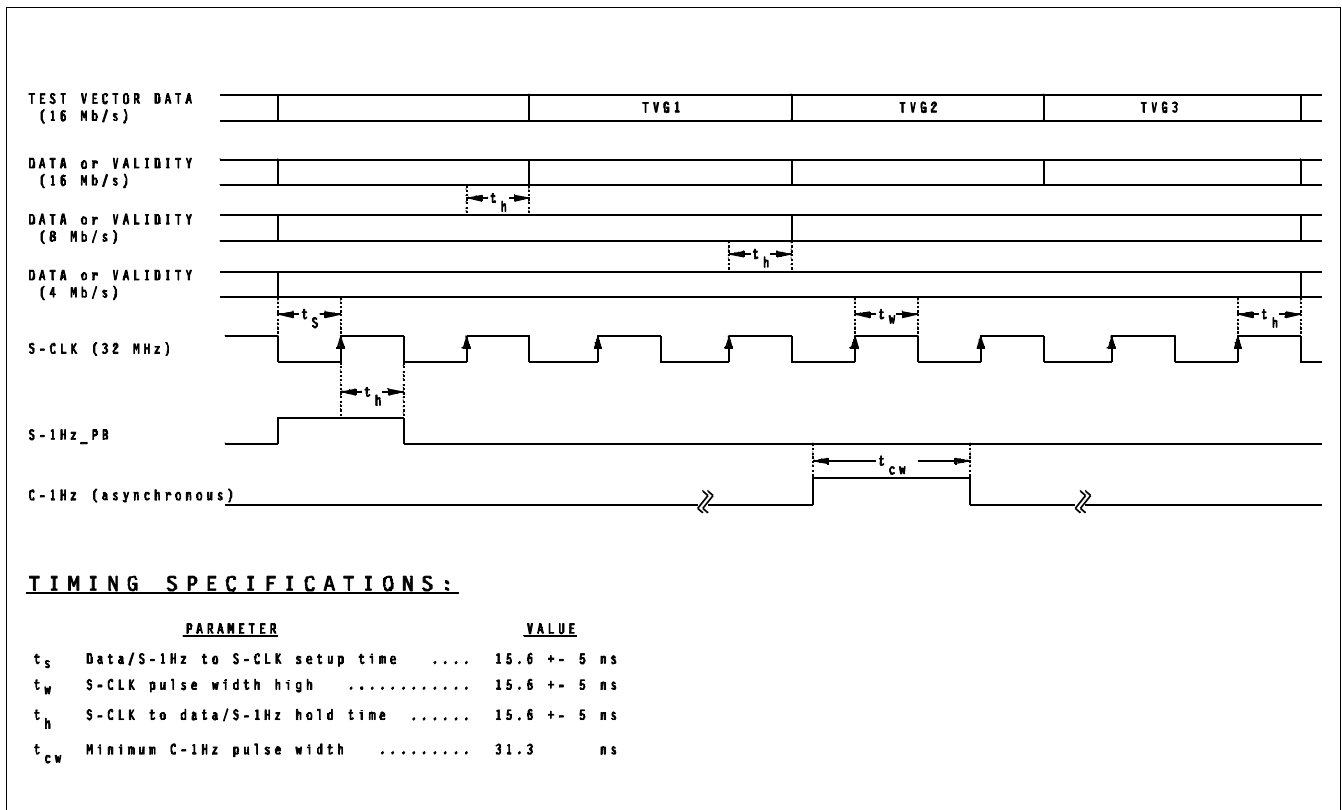
Figure B-4: C2 Output Timing Interface (32 MHz clk)
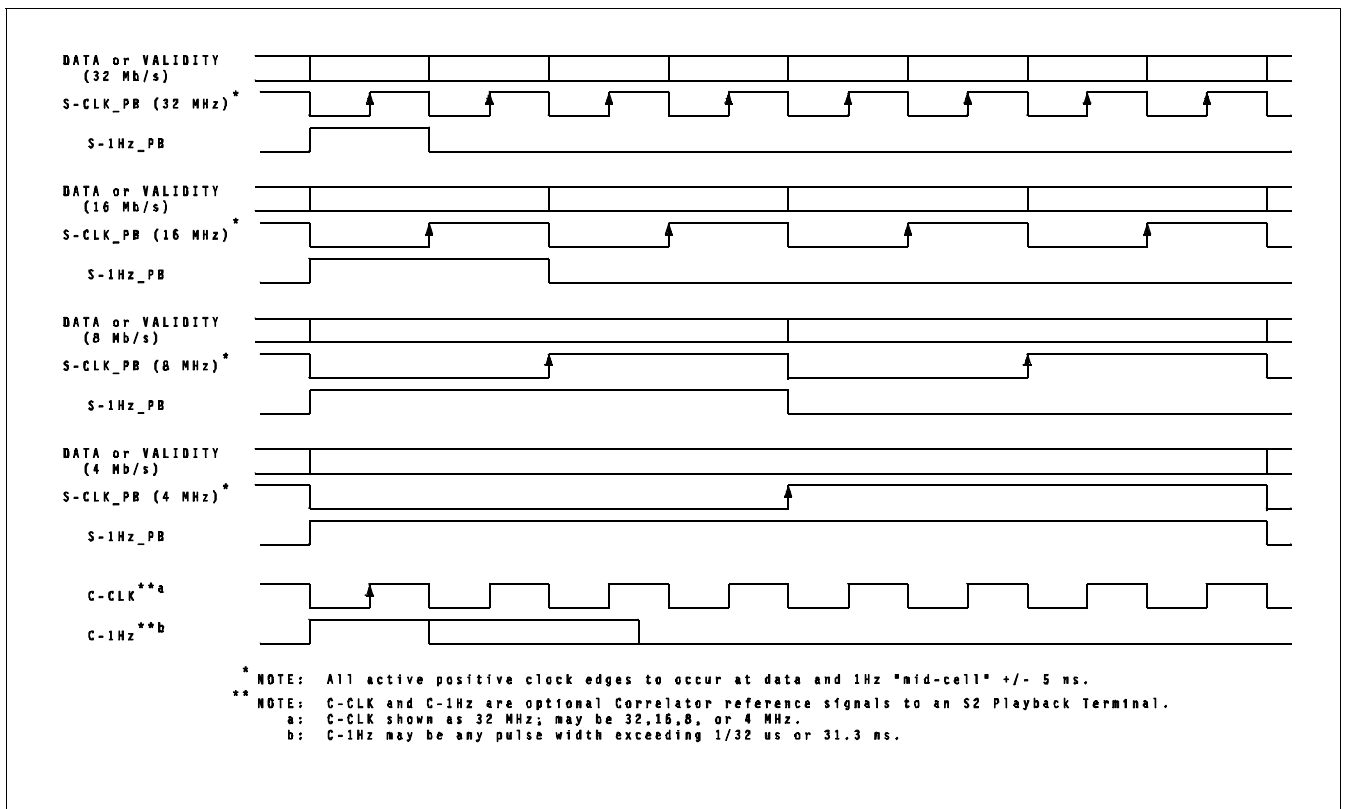
Figure B-5: C2 Output Timing Interface (clk=data rate)
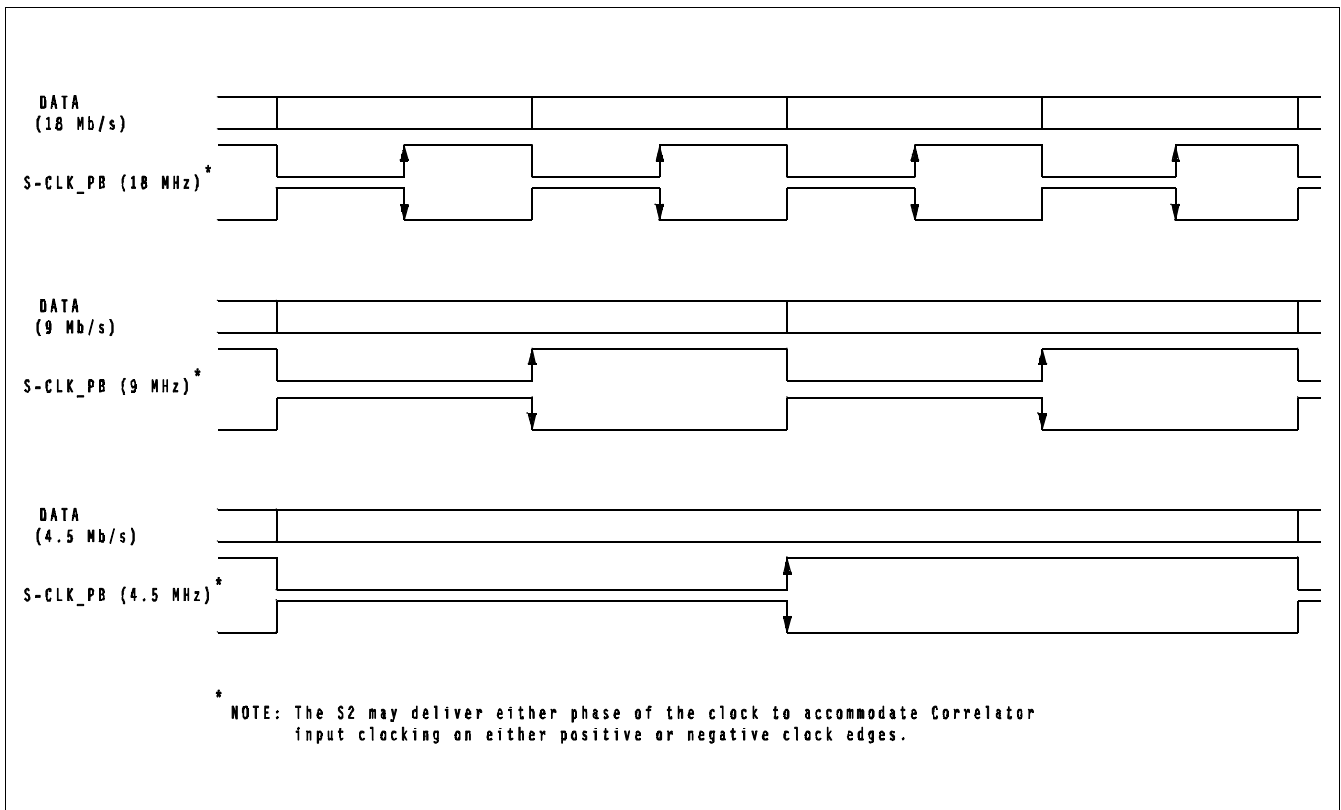
Figure B-6: C2 Output Timing Interface (MkIII format)
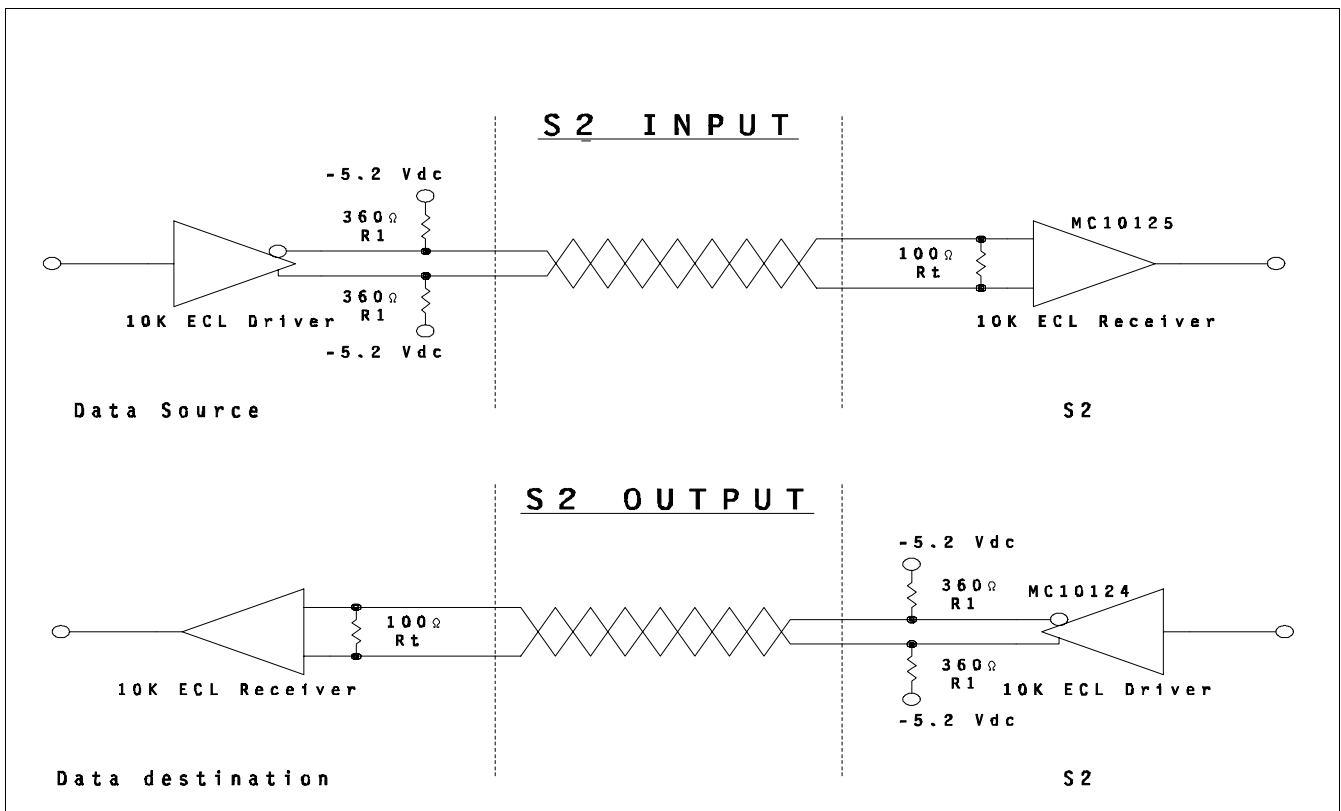
Figure B-7: C1 & C2 Electrical Specifications

APPENDIX C

SAMPLE ROS DEFAULTS FILE

```
####################################################
# S2 Recorder Operating System startup defaults file v2.3 #
####################################################

# Console terminal type (e.g. vt100).
term = vt100

# Console baud rate (normally 9600, max. 38400).
baud = 9600

# Default recorder mode.
dfltmode = 16x8-1

# Default barrel roll setting ('on' or 'off', normally 'on'). Should be
# off to record data that can be usefully played back on an S2-RT.
barrelroll = on

# Default record speed ('lp' or 'slp').
recordspeed = slp

# Tape type code, one of:
# 1 == Scotch ST-120/SE-180
# 2 == Maxell ST-180/ST-182/SE-260
# 3 == Ampex 289 ST-120/ST-126/SE-180
# ...or additional typecodes as listed by the 'tapetype typecodes' command.
# Used to determine write current levels. For unlisted tape types enter
# digits NNNMMM where NNN is write current for LP and MMM is current for SLP.
tapetype = 3

# Number of transports connected to this S2 system (1, 2, 4, or 8, normally 8).
# This determines the number of transports to check during general diagnostics
# and self-test. This also determines the number of transports to select and
# use in the test modes c1testXX, c2testXX, and diagXX.
# In a PT determines the minimum number of DRD boards that must be present.
diagnumtran = 8

# Default system clock PLL lock mode, one of: refclk, 1hz, errmes, xtal, auto,
# or manual. Normally 'refclk' for record terminal. 'auto' selects 'refclk'
# if user ref clock present, 'xtal' otherwise (on both RT and PT).
scpllmode = auto

# Source selection for input ref clock and 1 Hz signals, 'c1' for C1 (record)
# cable and 'c2' for C2 (playback) cable. 'default' means select C1 on an S2-RT
# and C2 on an S2-PT.
clkinsrc = c1
# Default input ref clock rate in MHz, one of: 4, 8, 16, 32, or followdata.
clkinrate = 32
# Default output ref clock rate in MHz, one of: 4, 8, 16, 32, or followdata.
clkoutrate = 32

# Disable the ability to record tapes ('on' or 'off', normally 'off' to allow
```

# recording). Useful in a PT to ensure that no data is accidentally destroyed.
writeprotect = off

# Station number from 0 to 255 (only 1-6 displayed by transports).
# 0 means "undefined", don't display on console.
station = 0
# System "nickname", what it's commonly known as (up to 8 characters).
nickname =
System serial number from 1 to 65535 (must match s/n label on DSCM card cage).
serialnum =

# Userid(s) and encrypted password(s) for rlogin/telnet access to console.
# If either of these are blank or omitted, no userid or password prompt will
# appear. The password entered here must be encrypted using the console
# 'encrypt' command. The default password is 'security' (encrypted RdebRSdzRc).
# Multiple userid/password pairs can be defined by adding more lines in the
# same format (each new userid line must be followed by a passwd line).
loginuserid = vxworks
#loginpasswd = RdebRSdzRc

# Default status verbose level (1-3) (normally 1)
# Higher values show some additional non-essential status messages.
verbose = 1

# Default "expert" mode setting ('on' or 'off').
# Suppresses some non-essential console commands if off, otherwise all
# commands available.
expert = on

# Select whether the VME cage front-panel LED (and formatter LED) should act
# as an overall system OK/FAIL light ('ok') or show the formatter hardware
# 1 Hz signal ('1hz').
ledmode = ok

# Recorder setup to use when S2 is stopped (or in rewind, ff etc.), either
# 'bypass' or 'play'. 'default' means select bypass on an S2-RT and play
# on an S2-PT. Record applications will typically use 'bypass', while playback
# applications will use 'play' which ensures a consistent S1HZ_PB output tick
# on the C2 cable (except during mode changes).
stopsetup = default

# Target-time selection method in automatic playback self-alignment.
# 'first' chooses the earliest transport's tape time, 'last' chooses the
# latest transport's tape time, and 'middle' chooses the time half-way
# between the two.
selfaligntarg = first

# Maximum time difference allowed during automatic playback self-alignment,
# in minutes from 1 to 600 (default 30 min, 0 means no limit). If the difference
# between any transport tape time and the selected target time exceeds this
# value, status code STAT_ALIGNFAIL with error ERR_ALBADTIME will appear.
# This limit does not apply to user 'align selfalign' requests.
selfaligntime = 30

SC PLL 'refclk' glitch fix flag. Automatically resets correct station delay
after a clock glitch in refclk mode (1) or does nothing (0) (normally 1).
Delay measurement must be different from setting and stable for two seconds.

scpllrefclkfix = 1

# Position display format. S2 tape positions are normally displayed as
# hours/minutes/seconds of recorded data since BOT, e.g. 5:47:29 (format 1).
# Tape positions may also be displayed as an integer number of seconds,
# e.g. 20849 (format 2). Note that console commands such as 'position' always
# accept both position formats.
posdisformat = 1

Mapping for mode selection on playback. This is used in special situations
where the mode selected on playback should be different from the mode used
at record time. Give pairs of modes in brackets, maximum 230 characters total.
Both modes in a pair must have the same total data rate.
e.g. (16x8-1 16i8-1) (8x16-1 8i16-1)
playmodemap =

# Default setting for per-transport write current offset option switch,
# as set by the 'transport options wrlevoffset' console command (offset set
# by 'transport wrlevoffset' command).
# 0 means disable per-transport write current offsets at bootup.
# 1 means enable per-transport write current offsets at bootup.
# 'default' means enable offsets at bootup for AG2550 and later transports only
wrlevoffoption = 1

# Disable the "eject all" feature (1) or eject all tapes as usual when any
# transport eject button is held longer than 2 seconds (0) (normally 0).
ejectalldisable = 0

# Disable the console "reboot" key (1) or reboot the S2 software and hardware
# as usual when cntl-_ (control-underscore) is typed (0) (normally 0).
rebootkeydisable = 0

# Skip self-test 1 at bootup (1) or run the self test as usual (0) (normally 0).
skipselftest1 = 0

# RCL baud rate (normally 19200, max. 57600).
rclbaud = 19200

# This S2's RCL address, 0-253. Use 0 unless connecting multiple RCL devices
# using RS485. Address is ignored during RCL connection via network sockets.
rcladdr = 0

# Mark III hardware-ID code, identifies recording station in Mark III header.
# Two hex digits, FF not allowed.
mk3hwid = 52

# Backspace character for VxWorks shell (ASCII code, normally 8 or 127).
backspace = 127

# The following parameter supplies stevie (vi) editor startup options.
# Use 'nobackup' to prevent use of backups, recommended when editing
# large files (remaining SRAM partition space), e.g. termcap file.
exinit = set nobackup showmode

#
The following parameters are used for network operation only.
#

```
# Lock the console against access by rlogin and telnet (1), or allow access
# as usual (0). As a special case, 2 allows access during the boot sequence
# but not after ROS starts running.
consolelock = 0

# Lock the RCL against access by network sockets (1), or allow access
# as usual (0).
rcllock = 0

# During bootup set system time from remote network host using RDATE
# protocol (1) or from built-in battery backed-up clock (0) (normally 0).
remtime = 0

# Internet name or address of host to use for network time setting.
# If omitted or blank will use host name and inet address from boot parameters.
# Host names entered here must appear in the VxWorks hosts table, so it may be
# easier to use the inet address.
timehost =

#
# The following parameters are for diagnostics only.
#

# Boot with the formatter RCLK enabled (1) or disabled (0) (normally 0).
formrclk = 0

# Enable automatic data-recovery defaults saves (1) or disable (0) (normally 1).
drsetdefaultauto = 1

# Flag used by install procedure, "done" indicates install done.
install = not done
```

APPENDIX D

UNIX TERMCAP FILE FORMAT

D.1   Overview

Termcap is a data base describing the capabilities of termi-
nals.   Terminals   are   described in termcap source descrip-
tions by giving a set of capabilities which they have, by
describing how operations are performed, by describing pad-
ding requirements,   and   by   specifying   initialization
sequences.   This database is used by applications programs
such as vi(1), and libraries such as curses(3X), so they can
work with a variety of terminals without changes to the pro-
grams.

Each termcap entry consist of a number of colon-separated
(:) fields.   The first field for each terminal lists the
various names by which it is known, separated by bar (|)
characters.   The first name is always two characters long,
and is used by older (version 6) systems (which store the
terminal type in a 16-bit word in a system-wide database).
The second name given is the most common abbreviation for
the terminal (this is the one to which the environment vari-
able TERM would normally be set).   The last name should
fully identify the terminal's make and model. All other
names are taken as synonyms for the initial terminal name.
All names but the first and last should be in lower case and
contain no blanks; the last name may well contain upper case
and blanks for added readability.

Terminal names (except for the last, verbose entry) should
be chosen using the following conventions:

+ The particular piece of hardware making up the terminal
   should   have a root name chosen; for example, for the
   Hewlett-Packard 2621, hp2621. This name should not   con-
   tain hyphens.

+ Modes that the hardware can be in or user preferences
   should   be indicated by appending a hyphen and an indica-
   tor of the mode. Thus, a vt100 in 132-column mode would
   be given as: vt100-w. The following suffixes should be
   used where possible:

Suffix	Meaning	Example
-w	wide mode (more than 80 columns)	vt100-w
-am	with automatic margins (usually default)	vt100-am
-nam	without automatic margins	vt100-nam
-n	number of lines on the screen	aaa-60
-na	no arrow keys (leave them in local)	concept100-na

-np	number of pages of memory	concept100-4p
-rv	reverse video	concept100-rv

Terminal entries may continue onto multiple lines by giving a \ as the last character of a line, and empty fields may be included for readability (here between the last field on a line and the first field on the next). Comments may be included on lines beginning with #.

D.2 Types of Capabilities

Terminal capabilities each have a two-letter code, and are of three types:

boolean   These indicate particular features of the terminal. For instance, an entry for a terminal that has automatic margins (an automatic RETURN and LINEFEED when the end of a line is reached) would contain a field with the boolean capability am.

numeric   These give the size of the display of some other attribute. Numeric capabilities are followed by the character '#', and a number. An entry for a terminal with an 80-column display would have a field containing co#80.

string   These indicate the character sequences used to perform particular terminal operations. String-valued capabilities, such as ce (clear-to-end-of-line sequence) are given by the two-letter code, followed by the character '=', and a string (which ends at the following : field delimiter).

A delay factor, in milliseconds may appear after the '='. Padding characters are supplied by tputs after the remainder of the string is sent. The delay can be either a number, or a number followed by the character '*', which indicates that the proportional padding is required, in which case the number given is the amount of padding for each line affected by an operation using that capability. (In the case of an insert-character operation, the factor is still the number of lines affected; this is always 1 unless the terminal has in and the software uses it.)

When a * is specified, it is sometimes useful to give a delay of the form 3.5 to specify a delay per line to tenths of milliseconds. (Only one decimal place is allowed.)

D.2.1   Comments

To comment-out a capability field, insert a '.' (period) as
the first character in that field (following the :).

D.2.2   Escape Sequence Codes

A number of escape sequences are provided in the string-
valued capabilities for easy encoding of characters there:

>   \E   maps to ESC
>   ^X   maps to CTRL-X for any appropriate character X
>   \n   maps to LINEFEED
>   \r   maps to RETURN
>   \t   maps to TAB
>   \b   maps to BACKSPACE
>   \f   maps to FORMFEED

Finally, characters may be given as three octal digits after
a backslash (for example, \123), and the characters ^
(caret) and \ (backslash) may be given as \^ and \\ respec-
tively.

If it is necessary to place a : in a capability it must be
escaped in octal as \072.

If it is necessary to place a NUL character in a string
capability it must be encoded as \200. (The routines that
deal with termcap use C strings and strip the high bits of
the output very late, so that a \200 comes out as a \000
would.)

D.2.3   Parameterized Strings

Cursor addressing and other strings requiring parameters are
described   by   a parameterized string capability, with
printf(3S)-like escapes (%x) in it; other characters are
passed through unchanged. For example, to address the cur-
sor, the cm capability is given, using two parameters: the
row and column to move to. (Rows and columns are numbered
from zero and refer to the physical screen visible to the
user, not to any unseen memory. If the terminal has
memory-relative cursor addressing, that can be indicated by
an analogous CM capability.)

The % escapes have the following meanings:
>   %%   produce the character %
>   %d   output value as in printf %d
>   %2   output value as in printf %2d
>   %3   output value as in printf %3d
>   %.   output value as in printf %c

%+x   add x to value, then do '%.'
%>xy   if value > x then add y, no output
%r   reverse order of two parameters, no output
%i   increment by one, no output
%n   exclusive-or all parameters with 0140 (Datamedia 2500)
%B   BCD (16*(value/10)) + (value%10), no output
%D   Reverse coding (value - 2*(value%16)), no output
         (Delta Data)

Consider the Hewlett-Packard 2645, which, to get to row 3
and column 12, needs to be sent \E&a12c03Y padded for 6
milliseconds. Note: the order of the row and column coordi-
nates is reversed here and that the row and column are sent
as two-digit integers.   Thus   its   cm   capability   is
':cm=6\E&%r%2c%2Y:'.   Terminals that use '%.' need to be
able to backspace the cursor (le) and to move the cursor up
one line on the screen (up). This is necessary because it
is not always safe to transmit \n, ^D, and \r, as the system
may change or discard them. (Programs using termcap must
set terminal modes so that TAB characters are not expanded,
making \t safe to send. This turns out to be essential for
the Ann Arbor 4080.)

A final example is the Lear Siegler ADM-3a, which offsets
row and column by a blank character, thus it requires
':cm=\E=%+ %+:'.

Row or column absolute cursor addressing can be given as
single-parameter capabilities ch (horizontal position abso-
lute) and cv (vertical position absolute). Sometimes these
are shorter than the more general two-parameter sequence (as
with the Hewlett-Packard 2645) and can be used in preference
to cm. If there are parameterized local motions (for exam-
ple, move n positions to the right) these can be given as
DO, LE, RI, and UP with a single parameter indicating how
many positions to move. These are primarily useful if the
terminal does not have cm, such as the Tektronix 4025.

D.2.4 Delays

Certain capabilities control padding in the terminal driver.
These are primarily needed by hardcopy terminals and are
used by the tset (1) program to set terminal driver modes
appropriately.   Delays embedded in the capabilities cr, sf,
le, ff, and ta will set the appropriate delay bits in the
terminal driver. If pb (padding baud rate) is given, these
values can be ignored at baud rates below the value of pb.
For 4.2BSD tset, the delays are given as numeric capabili-
ties dC, dN, dB, dF, and dT instead.

D.2.5   Similar Terminals

If there are two very similar terminals, one can be defined as being just like the other with certain exceptions. The string capability tc can be given with the name of the simi-lar terminal.   This capability must be last, and the com-bined length of the entries must not exceed 1024. The capa-bilities given before tc override those in the terminal type invoked by tc. A capability can be canceled by placing xx@ to the left of the tc invocation, where xx is the capabil-ity. For example, the entry

    hn|2621-nl:ks@:ke@:tc=2621:

defines a 2621-nl that does not have the ks or ke capabili-ties, hence does not turn on the function key labels when in visual mode. This is useful for different modes for a ter-minal, or for different user preferences.

D.3   Capabilities

The characters in the Notes field in the next table have the following meanings (more than one may apply to a capabil-ity):

    N   indicates numeric parameter(s)
    P   indicates that padding may be specified
    *   indicates that padding may be based on the number of lines affected
    o   indicates capability is obsolete

Obsolete capabilities have no terminfo equivalents, since they were considered useless, or are subsumed by other capa-bilities. New software should not rely on them.

Name	Type	Notes	Description
!1	str		sent by shifted save key
!2	str		sent by shifted suspend key
!3	str		sent by shifted undo key
#1	str		sent by shifted help key
#2	str		sent by shifted home key
#3	str		sent by shifted input key
#4	str		sent by shifted left-arrow key
%0	str		sent by redo key
%1	str		sent by help key
%2	str		sent by mark key
%3	str		sent by message key
%4	str		sent by move key
%5	str		sent by next-object key
%6	str		sent by open key
%7	str		sent by options key
%8	str		sent by previous-object key
%9	str		sent by print or copy key

%a	str		sent by shifted message key
%b	str		sent by shifted move key
%c	str		sent by shifted next-object key
%d	str		sent by shifted options key
%e	str		sent by shifted previous-object key
%f	str		sent by shifted print or copy key
%g	str		sent by shifted redo key
%h	str		sent by shifted replace key
%i	str		sent by shifted right-arrow key
%j	str		sent by shifted resume key
&0	str		sent by shifted cancel key
&1	str		sent by ref(erence) key
&2	str		sent by refresh key
&3	str		sent by replace key
&4	str		sent by restart key
&5	str		sent by resume key
&6	str		sent by save key
&7	str		sent by suspend key
&8	str		sent by undo key
&9	str		sent by shifted beg(inning) key
*0	str		sent by shifted find key
*1	str		sent by shifted cmd (command) key
*2	str		sent by shifted copy key
*3	str		sent by shifted create key
*4	str		sent by shifted delete-char key
*5	str		sent by shifted delete-line key
*6	str		sent by select key
*7	str		sent by shifted end key
*8	str		sent by shifted clear-line key
*9	str		sent by shifted exit key
5i	bool		printer will not echo on screen
@0	str		sent by find key
@1	str		sent by beg(inning) key
@2	str		sent by cancel key
@3	str		sent by close key
@4	str		sent by cmd (command) key
@5	str		sent by copy key
@6	str		sent by create key
@7	str		sent by end key
@8	str		sent by enter/send key (unreliable)
@9	str		sent by exit key
AL	str	(NP*)	add n new blank lines
CC	str		terminal settable command char in prototype
CM	str	(NP)	memory-relative cursor motion to row m, col n
DC	str	(NP*)	delete n characters
DL	str	(NP*)	delete n lines
DO	str	(NP*)	move cursor down n lines
EP	bool	(o)	even parity
F1-F9	str		sent by function keys 11-19
FA-FZ	str		sent by function keys 20-45
Fa-Fr	str		sent by function keys 46-63
HC	bool		cursor is hard to see
HD	bool	(o)	half-duplex
IC	str	(NP*)	insert n blank characters
K1	str		sent by keypad upper left
K2	str		sent by keypad center

K3	str		sent by keypad upper right
K4	str		sent by keypad lower left
K5	str		sent by keypad lower right
LC	bool	(o)	lower-case only
LE	str	(NP)	move cursor left n positions
LF	str	(P)	turn off soft labels
LO	str	(P)	turn on soft labels
MC	str	(P)	clear left and right soft margins
ML	str	(P)	set soft left margin
MR	str	(P)	set soft right margin
NL	bool	(o)	\n is NEWLINE, not LINEFEED
NP	bool		pad character does not exist
NR	bool		ti does not reverse te
Nl	num		number of labels on screen (start at 1)
OP	bool	(o)	odd parity
RA	str	(P)	turn off automatic margins
RF	str		send next input character (for ptys)
RI	str	(NP)	move cursor right n positions
RX	str	(P)	turn off xoff/xon handshaking
SA	str	(P)	turn on automatic margins
SF	str	(NP*)	scroll forward n lines
SR	str	(NP*)	scroll backward n lines
SX	str	(P)	turn on xoff/xon handshaking
UC	bool	(o)	upper-case only
UP	str	(NP*)	move cursor up n lines
XF	str		x-off character (default DC3)
XN	str		x-on character (default DC1)
ac	str		graphic character set pairs aAbBcC - def=VT100
ae	str	(P)	end alternate character set
al	str	(P*)	add new blank line
am	bool		terminal has automatic margins
as	str	(P)	start alternate character set
bc	str	(o)	backspace if not ^H
bl	str	(P)	audible signal (bell)
bs	bool	(o)	terminal can backspace with ^H
bt	str	(P)	back-tab
bw	bool		le (backspace) wraps from column 0 to last col
cb	str	(P)	clear to beginning of line, inclusive
cd	str	(P*)	clear to end of display
ce	str	(P)	clear to end of line
ch	str	(NP)	set cursor column (horizontal position)
cl	str	(P*)	clear screen and home cursor
cm	str	(NP)	screen-relative cursor motion to row m, col n
co	num		number of columns in a line
cr	str	(P*)	RETURN
cs	str	(NP)	change scrolling region to lines m thru n
ct	str	(P)	clear all tab stops
cv	str	(NP)	set cursor row (vertical position)
dB	num	(o)	milliseconds of bs delay needed (default 0)
dC	num	(o)	milliseconds of cr delay needed (default 0)
dF	num	(o)	milliseconds of ff delay needed (default 0)
dN	num	(o)	milliseconds of nl delay needed (default 0)
dT	num	(o)	ms of horizontal tab delay needed (default 0)
dV	num	(o)	ms of vertical tab delay needed (default 0)
da	bool		display may be retained above the screen
db	bool		display may be retained below the screen

dc	str	(P*)	delete character
dl	str	(P*)	delete line
dm	str		enter delete mode
do	str		down one line
ds	str		disable status line
eA	str	(P)	enable graphic character set
ec	str	(NP)	erase n characters
ed	str		end delete mode
ei	str		end insert mode
eo	bool		can erase overstrikes with a blank
es	bool		escape can be used on the status line
ff	str	(P*)	hardcopy terminal page eject
fs	str		return from status line
gn	bool		generic line type (for example dialup, switch)
hc	bool		hardcopy terminal
hd	str		half-line down (forward 1/2 linefeed)
ho	str	(P)	home cursor
hs	bool		has extra status line
hu	str		half-line up (reverse 1/2 linefeed)
hz	bool		cannot print ~s (Hazeltine)
i1	str		terminal initialization string (terminfo only)
i3	str		terminal initialization string (terminfo only)
iP	str		pathname of program for init (terminfo only)
ic	str	(P*)	insert character
if	str		name of file containing initialization string
im	str		enter insert mode
in	bool		insert mode distinguishes nulls
ip	str	(P*)	insert pad after character inserted
is	str		terminal initialization string
it	num		tab stops initially every n positions
k0-k9	str		sent by function keys 0-9
k;	str		sent by function key 10
kA	str		sent by insert-line key
kB	str		sent by back-tab key
kC	str		sent by clear-screen or erase key
kD	str		sent by delete-character key
kE	str		sent by clear-to-end-of-line key
kF	str		sent by scroll-forward/down key
kH	str		sent by home-down key
kI	str		sent by insert-char or enter-insert-mode key
kL	str		sent by delete-line key
kM	str		sent by insert key while in insert mode
kN	str		sent by next-page key
kP	str		sent by previous-page key
kR	str		sent by scroll-backward/up key
kS	str		sent by clear-to-end-of-screen key
kT	str		sent by set-tab key
ka	str		sent by clear-all-tabs key
kb	str		sent by backspace key
kd	str		sent by down-arrow key
ke	str		out of keypad transmit mode
kh	str		sent by home key
kl	str		sent by left-arrow key
km	bool		has a meta key (shift, sets parity bit)
kn	num	(o)	number of function (k0-k9) keys (default 0)
ko	str	(o)	termcap entries for other non-function keys

kr	str		sent by right-arrow key
ks	str		put terminal in keypad transmit mode
kt	str		sent by clear-tab key
ku	str		sent by up-arrow key
l0-l9	str		labels on function keys 0-9 if not f0-f9
la	str		label on function key 10 if not f10
le	str	(P)	move cursor left one position
lh	num		number of rows in each label
li	num		number of lines on screen or page
ll	str		last line, first column
lm	num		lines of memory if > li (0 means varies)
lw	num		number of columns in each label
ma	str	(o)	arrow key map (used by vi version 2 only)
mb	str		turn on blinking attribute
md	str		turn on bold (extra bright) attribute
me	str		turn off all attributes
mh	str		turn on half-bright attribute
mi	bool		safe to move while in insert mode
mk	str		turn on blank attribute (characters invisible)
ml	str	(o)	memory lock on above cursor
mm	str		turn on meta mode (8th bit)
mo	str		turn off meta mode
mp	str		turn on protected attribute
mr	str		turn on reverse-video attribute
ms	bool		safe to move in standout modes
mu	str	(o)	memory unlock (turn off memory lock)
nc	bool	(o)	no correctly-working cr (Datamedia 2500)
nd	str		non-destructive space (cursor right)
nl	str	(o)	NEWLINE character if not
ns	bool	(o)	terminal is a CRT but does not scroll
nw	str	(P)	NEWLINE (behaves like cr followed by do)
nx	bool		padding will not work, xoff/xon required
os	bool		terminal overstrikes
pO	str	(N)	turn on the printer for n bytes
pb	num		lowest baud where delays are required
pc	str		pad character (default NUL)
pf	str		turn off the printer
pk	str		function key n to type string s (terminfo only)
pl	str		function key n to exec string s (terminfo only)
pn	str	(NP)	label n to show string s (terminfo only)
po	str		turn on the printer
ps	str		print contents of the screen
pt	bool	(o)	has hardware tabs (may need to be set with is)
px	str		function key n to transmit string s (terminfo)
r1	str		reset terminal to sane modes (terminfo only)
r2	str		reset terminal to sane modes (terminfo only)
r3	str		reset terminal to sane modes (terminfo only)
rP	str	(P)	like ip but when in replace mode
rc	str	(P)	restore cursor to position of last sc
rf	str		name of file containing reset string
ri	?		unkown at present
rp	str	(NP*)	repeat character c n times
rs	str		reset terminal completely to sane modes
sa	str	(NP)	define the video attributes (9 parameters)
sc	str	(P)	save cursor position
se	str		end standout mode

sf	str	(P)	scroll text up
sg	num		# of garbage chars left by so or se (default 0)
so	str		begin standout mode
sr	str	(P)	scroll text down
st	str		set a tab stop in all rows, current column
ta	str	(P)	move cursor to next 8-position hardware tab
tc	str		entry of similar terminal - must be last
te	str		string to end programs that use termcap
ti	str		string to begin programs that use termcap
ts	str	(N)	go to status line, column n
uc	str		underscore one character and move past it
ue	str		end underscore mode
ug	num		# of garbage chars left by us or ue (default 0)
ul	bool		underline character overstrikes
up	str		upline (cursor up)
us	str		start underscore mode
vb	str		visible bell (must not move cursor)
ve	str		make cursor appear normal (undo vs/vi)
vi	str		make cursor invisible
vs	str		make cursor very visible
vt	num		virtual terminal number
wi	str	(N)	set curr window to line i thru j, col m thru n
ws	num		number of columns in status line
xb	bool		Beehive (f1=ESC, f2=^C)
xn	bool		NEWLINE ignored after 80 cols (Concept)
xo	bool		terminal uses xoff/xon handshaking
xr	bool	(o)	RETURN acts like ce cr nl (Delta Data)
xs	bool		standout not erased by overwriting (HP)
xt	bool		TAB chars destructive, magic so char (Teleray)
xx	bool	(o)	Tektronix 4025 insert-line

APPENDIX E

GLOSSARY

ADC: Analog to Digital Converter
AGC: Automatic Gain Control
AHOOK: Analog Direct-Hookup
BER: Bit-Error Rate
BERC: Bit-Error Rate Counter
BOT: Beginning Of Tape
BYP: Bypass
CRC: Cyclical Redundancy Check
DAC: Digital to Analog Converter
DEF: Deformatter
DHOOK: Digital Direct-Hookup
DR: Data Recovery
DRAM: Dynamic Random Access Memory
DRD: Data Recovery/Deformatter
DSCM: Data, Signal, and Control Module
DV: Data Validity
EOD: End Of Data
EOT: End Of Tape
EP: Extended Play (same as SLP)
ERRMES: Error Measurement
FF: Fast-Forward
FIFO: First-In-First-Out memory
FORM: Formatter
LAN: Local-Area Network
LP: Long Play
PB: Playback
PCFS: Personal Computer Field System (NASA Goddard Space Flight Center)
PLL: Phase-Lock Loop
PT: Playback Terminal
RCC: Recorder Control Computer
RCL: Radioastronomy Control Link (formerly Recorder Control Link)
RCLCO: Radioastronomy Control Link Communications program
REFCLK: Reference Clock
ROS: Recorder Operating System
ROSCO: Recorder Operating System Communications program
RPMON: Record-Playback Monitor
RT: Record Terminal
SCPLL: System Clock Phase-Lock Loop
SLP: Super-Long Play
SP: Short Play
SRAM: Static Random Access Memory
SRSYR: Servo-Sync (Record)
SVHS: Super VHS
TAM: Transport Array Module
TCL: Transport Control Link
TCB: Transport Control Board
TCP: Transport Control Processor
TRAN: Transport
TVG: Test Vector Generator
TVR: Test Vector Receiver
UI: User Interface
UIC: User Interface Card
UTC: Co-ordinated Universal Time
VCO: Voltage-Controlled Oscillator
VLBI: Very Long Baseline Interferometry
VSOP: VLBI Space Observatory Programme

Bibliography

[1] Brent Carlson, "S2 VLBI 'Tape ID' and 'User Info Field' Format Requirements for Canadian (S2) Correlator Operational Compatibility", DRAO-RPCS3001 Rev. 1.1, Dominion Radio Astrophysical Observatory, July 30, 1996
http://www.drao.nrc.ca/web/vlbi/docs/doc3001.ps.gz

[2] Georg Feil, "Timing and Synchronization in S2 Record and Playback Terminals", ISTS-SGL-TR94-030, Institute for Space and Terrestrial Science, Space Geodynamics Laboratory, June 1994
ftp://ftp.sgl.crestech.ca/pub/s2/doc/timing/timing.ps.Z

[3] ISTS/SGL, "VLBI System Interface Adapter (VIA) User's Manual", ver 1.3, Institute for Space and Terrestrial Science, Space Geodynamics Laboratory, October 1996
ftp://ftp.sgl.crestech.ca/pub/s2/doc/svlbi/via_1.3.ps.Z

[4] Rick Wietfeldt, "Compatibility of S2 and VSOPT Recordings at S2 and VSOP Correlators", ver 1.1, Institute for Space and Terrestrial Science, Space Geodynamics Laboratory, June 18, 1996
ftp://ftp.sgl.crestech.ca/pub/s2/svlbi/s2vsop_compat_memo_v1.1.ps.Z

[5] NASA/GSFC, "Mark IV Field System", volume 1 & 2 plus supplements, ver 9.3, Space Geodesy Program NASA/Goddard Space Flight Center, January, 1997
ftp://gemini.gsfc.nasa.gov/pub/fs/

[6] Motorola Inc., "MVME133XT VMEmodule 32 Bit Monoboard Microcomputer User's Manual", MVME133XT/D1, April 1988

[7] Georg Feil, "S2 GRT Operations Overview for VSOP", Version 1.5, CRESTech-SGL-TR98-040, Centre for Research in Earth and Space Technology, Space Geodynamics Laboratory, March 22, 1998
ftp://ftp.sgl.crestech.ca/pub/s2/svlbi/grtop.ps.Z

Index