

950RTS Remote Test System Operating Manual

Part Number 9100-0950-01 V.2.3 0498

CONTENTS

<i>Contents</i>	<i>Page</i>
SECTION I GENERAL INFORMATION	1—1
1-1 INTRODUCTION	1—1
1-1.1 Contacting Sage Instruments	1—1
1-1.2 User Access	1—2
1-1.3 UNIX Documentation	1—2
1-1.4 Typographical Conventions	1—2
SECTION II DEMAND TESTING	2—1
2-1 INTRODUCTION	2—1
2-1.1 Help Information	2—1
2-1.2 Screen Information	2—1
2-1.3 Status Boxes	2—2
2-2 NAVIGATION	2—4
2-2.1 Highlighting and Selecting	2—4
2-2.2 Shortcut Selection	2—5
2-3 GETTING STARTED	2—5
2-3.1 Starting Vftest	2—5
2-3.1.1 Selecting a Circuit in Direct Mode	2—6
2-3.1.2 Selecting a Circuit in DCS Mode	2—7
2-4 TEST SETUP	2—9
2-4.1 Tone Sender Control	2—9
2-4.1.1 Turning Tone Sender On and Off	2—10
2-4.1.2 Setting the Frequency	2—10
2-4.1.3 Setting the Level	2—11
2-4.2 Remote Audio	2—11
2-4.2.1 Port	2—11
2-4.2.2 Setup	2—12
2-4.3 VF Test Setup	2—12
2-4.3.1 Direct Mode	2—13
2-4.3.2 DCS Mode	2—13
2-4.3.2.1 Channel States	2—14
2-4.3.2.2 Test Direction	2—16
2-5 SELECTING AND PERFORMING TESTS	2—16
2-5.1 Measure Tone	2—16

Contents	Page
2-5.2 Measure Noise	2—17
2-5.3 Return Loss	2—18
2-5.4 Dial Menu	2—19
2-5.4.1 Supervision	2—20
2-5.4.2 Format	2—20
2-5.4.3 Digits	2—20
2-5.4.4 Extended Dialing	2—21
2-5.4.4.1 Digit Sequences	2—22
2-5.4.4.2 Digit Format	2—22
2-5.4.4.3 Pre-Dial	2—22
2-5.4.4.4 Send Digit Sequences	2—23
2-5.4.5 Quit	2—23
2-6 QUIT VFTEST	2—24
 SECTION III DEMAND TESTING OPTIONS	 3—1
3-1 OPTIONS	3—1
3-1.1 Demand-105	3—1
3-1.1.1 Predial	3—2
3-1.1.2 Selecting a Dial Format	3—3
3-1.1.3 Entering Numbers to Dial	3—3
3-1.1.4 Selecting Tests to Perform	3—4
3-1.1.5 Starting the Test	3—5
3-1.1.6 Quitting the Test	3—5
3-1.2 Impairment Tests	3—6
3-1.2.1 Intermodulation Distortion	3—6
3-1.2.2 Impulse Noise & Hits	3—7
3-1.2.3 Phase and Amplitude Jitter	3—9
3-1.2.4 Envelope Delay Distortion/Frequency Sweep	3—10
3-1.2.5 Absolute Delay	3—12
3-1.2.6 Peak to Average Ratio (PAR)	3—13
3-1.3 23 Tones	3—14
 SECTION IV DSO BERT TESTING	 4—1
4-1 INTRODUCTION	4—1
4-2 NAVIGATION	4—1
4-2.1 Screen Information	4—1
4-3 GETTING STARTED	4—2
4-3.1 Selecting a Circuit in Direct Mode	4—3
4-3.2 Selecting a Circuit in DCS Mode	4—4
4-4 PREPARING TO PERFORM A DSO BERT TEST	4—6

Contents	Page
4-4.1 Status Boxes	4—6
4-4.2 DSO BERT Results Menu	4—8
4-4.3 BERT Test Setup Menu	4—9
4-4.3.1 Choosing a Primary Bit Pattern	4—9
4-4.3.1.1 Choosing a User Pattern	4—10
4-4.3.2 Choosing a Results File	4—10
4-4.3.3 Selecting a Test Duration	4—11
4-4.4 Data Format	4—11
4-4.4.1 Selecting a Format	4—12
4-4.4.2 Selecting a Subrate	4—12
4-4.4.3 Turning the Echo Cancellor On / Off	4—12
4-4.5 Circuit Select	4—13
4-4.5.1 Mode (DCS only)	4—13
4-4.5.2 Direction (DCS only)	4—15
4-5 DIAL (SWITCHED 56 KB CIRCUITS ONLY)	4—15
4-5.1 Origin Supervision	4—16
4-5.2 Format	4—16
4-5.3 Digits	4—16
4-5.4 Extended Dialing	4—17
4-5.4.1 Digit Sequences	4—17
4-5.4.2 Dial Format	4—17
4-5.4.3 Pre-Dial	4—18
4-5.4.4 Send Digit Sequences	4—19
4-5.4.5 Quit	4—19
4-6 PERFORMING A DSO BERT TEST	4—20
4-6.1 Error Inject	4—20
4-6.2 Loopback/Loopup	4—21
4-6.2.1 Selecting a Loopback/Loopup Type	4—22
4-6.2.2 Selecting a Loopback Code	4—22
4-6.2.3 Selecting a Repeater	4—23
4-6.2.4 Performing a Loopback/Loopup	4—23
4-6.3 Network Codes	4—24
4-6.3.1 Selecting a Network Code	4—24
4-6.3.2 Sending a Network Code	4—25
4-7 VIEWING RESULTS HISTORY	4—25
 SECTION V DS1 BERT TESTING	 5—1
5-1 INTRODUCTION	5—1
5-2 NAVIGATION	5—1
5-2.1 Screen Information	5—1
5-3 GETTING STARTED	5—2

Contents	Page
5-3.1 Selecting a Circuit in Direct Mode	5—3
5-3.2 Selecting a Circuit in DCS Mode	5—4
5-4 PREPARING TO PERFORM A DS1 BERT TEST	5—6
5-4.1 Status Boxes	5—6
5-4.2 DS1 BERT Results Menu	5—7
5-4.3 Test Setup	5—7
5-4.3.1 Choosing a Primary Bit Pattern	5—8
5-4.3.1.1 Choosing a User Pattern	5—8
5-4.3.2 Choosing a Results File	5—9
5-4.3.3 Selecting a Test Duration	5—9
5-4.4 T1 Interface Configuration Menu	5—10
5-4.4.1 Framing	5—10
5-4.4.2 Coding Type	5—11
5-4.4.3 Performance Monitor	5—11
5-4.4.4 Length	5—11
5-4.5 CIRCUIT SELECT	5—11
5-4.5.1 Mode (DCS only)	5—12
5-4.5.2 Direction (DCS only)	5—13
5-5 PERFORMING A DS1 BERT TEST	5—14
5-5.1 Error Inject	5—15
5-5.2 Loopback/Loopup	5—16
5-5.2.1 Selecting a Loopback Code	5—16
5-5.2.1.1 Choosing a User Defined Code	5—17
5-5.2.2 Sending a Loop Down Code	5—17
5-5.2.3 Performing a Loopback/Loopup	5—18
5-6 VIEWING RESULTS HISTORY	5—18
 SECTION VI TRUNK ROUTINING	 6—1
6-1 AUTOMATIC TRUNK ROUTINING (ATR)	6—1
6-1.1 Software Configuration	6—1
6-2 CONFIGURING THE 950RTS	6—3
6-2.1 Downloading The Near-End Switch Trunks Database	6—3
6-2.2 Adding Site and Filter Files	6—3
6-3 MENU-BASED SCHEDULE MANAGER	6—3
6-3.1 Navigation	6—3
6-3.1.1 Screen Information	6—3
6-3.1.2 Quick Keys	6—4
6-3.2 Planning ATR Test Schedules	6—4
6-3.2.1 Required Information	6—4
6-3.2.2 Filling in the Schedule Worksheet	6—5
6-3.2.3 Adding the Schedule to the Database	6—6

Contents	Page
6-3.2.4 Enabling the New Schedule	6—6
6-3.2.5 Summary of Commands	6—6
6-3.3 Scheduling a Test	6—7
6-3.3.1 Initial Vsched Screen	6—7
6-3.3.2 Sorting	6—8
6-3.3.3 Preventing Schedule Overlap	6—8
6-3.3.4 Scrolling	6—8
6-3.3.5 Creating a New Schedule	6—8
6-3.4 Schedule Edit Screens	6—9
6-3.4.1 Text Entry Fields	6—9
6-3.4.2 Selection Fields	6—13
6-3.5 Schedule Save Screen	6—14
6-4 COMMAND-LINE SCHEDULE MANAGER	6—14
6-5 USER CHANGEABLE SCHEDULE FIELDS	6—15
6-5.1 Time Fields	6—17
6-5.1.1 Time To Run	6—17
6-5.1.2 Repeat Field	6—18
6-5.2 Test Description Fields	6—18
6-5.2.1 Site Field	6—18
6-5.2.2 Trunk Groups	6—18
6-5.2.3 Test Line Type	6—19
6-5.2.4 Test Line Access	6—19
6-5.2.5 Test Port	6—19
6-5.2.6 Test Type	6—20
6-5.3 Reporting Fields	6—20
6-5.3.1 Reporting and Test Limits	6—20
6-5.3.2 Results File	6—21
6-6 STATUS FIELDS	6—22
6-6.1 Last Run	6—22
6-6.2 Duration	6—23
6-6.3 Status	6—23
6-6.4 User	6—23
6-6.5 Last Results	6—23
6-7 MANAGING TESTS AND TEST RESULTS	6—23
6-7.1 Cancelling a Scheduled Test	6—23
6-7.2 Aborting a Running Test	6—24
6-7.3 Viewing Test Results	6—24
6-7.4 Running the Test Driver	6—24
6-7.5 Related Documentation	6—25

Contents	Page
SECTION VII ATME	7—1
7-1 INTRODUCTION	7—1
7-2 OPERATION PRINCIPLES OF ATME	7—1
7-2.1 ATME measurments	7—1
7-3 CMS LOCKING TONE AND THE ECHO CANCELER	7—2
SECTION VIII TERMINAL REMOTE CONTROL	8—1
8-1 INTRODUCTION	8—1
8-2 FACETTERM	8—1
8-2.1 Startup	8—1
7-2.2 Interacting With FacetTerm	8—2
8-2.2.1 Menu-Based Navigation	8—3
8-2.2.2 Command Line Navigation	8—4
8-3 TINY TERM	8—4
SECTION IX SAGE COMMAND LANGUAGE (SCL)	9—1
9-1 SUMMARY	9—1
9-2 BACKGROUND	9—1
9-3 TCL BASICS	9—2
9-4 SAGE EXTENSIONS TO TCL	9—2
9-5 TEST ACCESS COMMANDS	9—3
9-6 PARAMETER COMMANDS	9—3
9-7 TEST COMMANDS	9—4
9-8 GENERIC TEST CONTROL COMMANDS	9—5
9-9 BERT TEST CONTROL COMMANDS	9—5
9-10 MISCELLANEOUS CONTROL COMMANDS	9—6
9-11 TEST RESOURCES	9—7
9-12 INTERACTIVE USAGE	9—7
9-13 SCRIPTING USAGE	9—8
9-14 REFERENCES	9—9
9-15 SCL TEST PARAMETERS AND RESULT TYPES	9—10
9-15.1 ds0bert	9—10
9-15.2 ds1bert	9—11
9-15.3 mtone	9—11
9-15.4 noise	9—12
9-15.5 rloss	9—12

<i>Contents</i>	<i>Page</i>
-----------------	-------------

APPENDIX I AUTOMATIC TRUNK ROUTINING

ON-LINE MANUAL PAGES A1—1

A1-1 ATR (SAGE)	A1—1
A1-2 ATRD (SAGE)	A1—4
A1-3 SCHED (SAGE)	A1—5
A1-4 DB (SAGE)	A1—10
A1-5 DBRECORD (SAGE)	A1—14
A1-6 MDP105 (SAGE)	A1—17
A1-7 NEAREND (SAGE)	A1—20
A1-8 CU (LOCAL)	A1—22

APPENDIX II FAR-END ON-LINE MANUAL PAGES A2—1

A2-1 FARDEM (SAGE)	A2—1
A2-2 FAREND (SAGE)	A2—4

APPENDIX III SCL ON-LINE MANUAL PAGES A3—1

A3-1 CALL PROGRESS (SCL)	A3—1
A3-2 CLRTEST (SCL)	A3—3
A3-3 CELLCMPNDR (SCL)	A3—4
A3-4 CELLCONNECT (SCL)	A3—5
A3-5 CELLDEVL (SCL)	A3—6
A3-6 CELLFILTER (SCL)	A3—7
A3-7 CELLGET (SCL)	A3—8
A3-8 CELLRSSI (SCL)	A3—9
A3-9 CELLSECTOR (SCL)	A3—10
A3-10 CELLSET (SCL)	A3—11
A3-11 CONNECT (SCL)	A3—12
A3-12 DCS (SCL)	A3—14
A3-13 DISCONNECT (SCL)	A3—17
A3-14 DSOBERT (SCL)	A3—18
A3-15 DSOINJECT (SCL)	A3—21
A3-16 DSOLOOP (SCL)	A3—23
A3-17 DS1BERT (SCL)	A3—25
A3-18 DS1INJECT (SCL)	A3—28
A3-19 DS1LOOP (SCL)	A3—30
A3-20 DS1MONITOR (SCL)	A3—32
A3-21 EXIT (SCL)	A3—34
A3-22 HOOK (SCL)	A3—35

Contents	Page
A3-23 IMD (SCL)	A3—36
A3-24 INTFC (SCL)	A3—39
A3-25 JITTER (SCL)	A3—41
A3-26 MTONE (SCL)	A3—43
A3-27 NAP (SCL)	A3—45
A3-28 NOISE (SCL)	A3—46
A3-29 PAR (SCL)	A3—48
A3-30 REPORT (SCL)	A3—50
A3-31 RLOSS (SCL)	A3—51
A3-32 ROTLATD (SCL)	A3—52
A3-33 ROTLGAIN (SCL)	A3—53
A3-34 ROTLIMD (SCL)	A3—54
A3-35 ROTLIMP (SCL)	A3—55
A3-36 ROTLJIT (SCL)	A3—56
A3-37 ROTLLOSS (SCL)	A3—57
A3-38 ROTLNOISE (SCL)	A3—58
A3-39 ROTLPAR (SCL)	A3—59
A3-40 ROTLRLOSS (SCL)	A3—60
A3-41 SDIGITS (SCL)	A3—61
A3-42 SELECTTEST (SCL)	A3—62
A3-43 SETPARAM (SCL)	A3—63
A3-44 SETVAR (SCL)	A3—65
A3-45 STARTTEST (SCL)	A3—67
A3-46 STONE (SCL)	A3—69
A3-47 STOPTEST (SCL)	A3—71
A3-48 TONE23 (SCL)	A3—73
A3-49 TRACE (SCL)	A3—75
A3-50 UPDATE (SCL)	A3—76
 APPENDIX IV SAGE 950RTS COMMANDS	 A4—1
A4-1 TEST TOOL COMMANDS	A4—1
A4-2 AUTOMATED DAEMONS	A4—1
A4-3 ADMINISTRATION COMMANDS AND UTILITIES	A4—2
A4-4 SCL COMMANDS	A4—3
A4-4 .1 Test Access	A4—3
A4-4 .2 VF and Data Tests	A4—3
A4-4 .3 BERT and DS1 Tests	A4—3
A4-4 .4 Test Controls	A4—3
A4-4 .5 General Controls	A4—4
A4-4 .6 ROTL Tests	A4—4
A4-4 .7 Cellular Tests	A4—4

Contents	Page
A4-4 .8 Display Functions	A4—5
A4-4 .9 Scripting Functions	A4—5
APPENDIX V MAKEDB COMMAND SCRIPTS	A5—1
A5-1 MAKEDB	A5—1

SECTION I

GENERAL INFORMATION

1-1 INTRODUCTION

The Sage **950RTS** is a remotely operated telecommunications test system designed to mount directly in the telephone equipment rack. The system contains a wide variety of diagnostic tools for conducting both on-demand testing and automatic trunk routining operations on analog and digital telephone network structures.

This manual is divided into six sections: **Section I** provides general reference information on the **950RTS**; **Section II** describes the operation of the Voice Frequency (VF) Demand Testing feature of the **950RTS**; **Section III** describes the purchase options for the VF Demand Testing feature; **Section IV** describes the BERT feature; **Section V** describes the Automatic Trunk Routining (ATR) feature; and **Section VI** describes the remote control feature.

1-1.1 Contacting Sage Instruments

To contact Sage Instruments in writing, send correspondence to:

Sage Instruments, Inc.
240 Airport Blvd.
Freedom, CA 95019-2614

or send Email to:

sales@sageinst.com or support@sageinst.com

To reach our Technical Support and Customer Service Departments by phone, call:

(831) 761-1000, M-F, 9 a.m. to 5 p.m., Pacific Time.

or fax our Technical Support and Customer Service Departments any time at:

(831) 761-2452

To receive company and product information via the World Wide Web, visit our home page at:

<http://www.sageinst.com>

Be sure to specify *Model 950RTS* when asking for technical support. Customers located outside of the U.S. may also contact their nearest Sage distributor for assistance.

Section I General Information

1-1.2 User Access

You must obtain an ID and password from the system administrator to gain access to the **950RTS** multi-user system. You should also be familiar with the facility to be tested, and should have read this manual prior to operating the system.

You can access the **950RTS** through a remote terminal connected to the **950RTS** via serial port, modem, VGA console, or Ethernet connection. Direct connect terminals may be of almost any type.

Any workstation capable of emulating a **UNIX** compatible terminal may be setup as an interface terminal for the **950RTS** when using the text-based interface.

Refer to the **950RTS Installation and Administration Manual** for a complete explanation of user access.

1-1.3 UNIX Documentation

SCO OEM UNIX system documentation is provided with the **950RTS**. Review this information before operating the system.

1-1.4 Typographical Conventions

In this manual, text appearing in courier typeface denotes screen information and commands typed at the **UNIX** prompt. Three typographical conventions differentiate this screen text:

- Text appearing in normal courier typeface indicates screen information and system responses.

Example:

Upon logging on to the switch the system responds:

```
swlink rev 1.0  
Copyright (C) 1995 Sage Instruments, Inc.  
Connecting to <Omaha>. Use "<Enter>~." to quit.
```

- Text appearing in **bold courier** typeface indicates commands that the user must enter at the **UNIX** prompt exactly as they appear in the manual.

Example:

At the **UNIX** prompt type:

```
% vsched
```

- Text appearing in *italicised courier* typeface indicates variable text that the user must enter at the **UNIX** prompt.

Example:

At the **UNIX** prompt type:

```
logfile filename
```

SECTION II

DEMAND TESTING

2-1 INTRODUCTION

The **vf**test screens are menu-based and ordered hierarchically. The main **vf**test control screen displays the **Vf**test **Measurement Menu**, which provides access to several **vf**test functions:

- Measure Tone
- Measure Noise
- Return Loss
- Dial Digits
- 105-Demand Tests (purchase option)
- Impairment Tests (purchase option)
- 23 Tones (purchase option)
- Quit **Vf**test

This section of the manual discusses navigating, setting-up, and executing each of the **vf**test functions.

Options, such as **105-Demand**, **Impairment**, and **23 Tones** testing appear in your **Vf**test **Measurement Menu** even if you have not purchased them. When you try to select them, however, you will see the message: **"This option is not installed on this system. Press <Enter> to continue."**

2-1.1 Help Information

At the bottom of each screen, two help information lines are available for reference. These lines display the procedures for moving the highlight cursor, selecting items, exiting and returning to the previous display, as well as performing any functions that are particular to the selected screen.

2-1.2 Screen Information

The **950RTS** screen always gives some basic information about the **950RTS** and the facility under test. That information is located at the top of the screen. In the **vf**test function this information includes:

- **950RTS** system name
- Current date and time

2-1.3 Status Boxes

The **950RTS** screen displays two status boxes in **vf test** that are not user-selectable. Either the **vf test** program updates them automatically during a test, or the user updates them indirectly by making changes to the information from within the **Vf test Measurement** menu:

- Connection Status
- Line Status

Connection Status Box

The **connection status** box displays the type of test connection made. In **Direct Mode**, this includes the name of the circuit under test, and the state of the tone sender. (See **Figure 2-1**.)



```
Connection Status
rb101
Tone snd: OFF
```

Figure 2-1
Connection Status Box in Direct Mode

In **DCS mode**, the **connection status** box also displays the **Circuit ID information** and the channel states: **mon** (monitor) mode, **split** (test) mode, or **rel** (release) mode. (See **Figure 2-2**.)

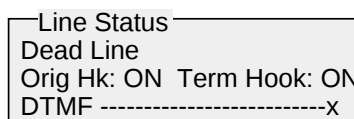


```
Connection Status
DCS TAD 1-1-1 Tap 1-16-1 rel eqp
Tone snd: OFF
```

Figure 2-2
Connection Status Box in DCS Mode

Line Status Box

In **direct mode**, the **Line Status** box displays the line status and the outpulsing mode (DTMF or MF). (See **Figure 2-3**.)



```
Line Status
Dead Line
Orig Hk: ON Term Hook: ON
DTMF -----X
```

Figure 2-3
Line Status Box in Direct and DCS Mode

The appearance of the **Line Status** box depends on the circuit selected. If the circuit is robbed bit, the **Line Status** box will display the on and off hook status. If the circuit is clear channel, the **Line Status** box will not display the on and off hook status.

The following is a list of possible line status messages:

Message Displayed	Description
Unknown	Unknown signal
Dead Line	More than 4 seconds of silence
Dial tone (350/440)	Steady 350 Hz/440 Hz dual tone
Dial tone (400)	Steady 400 Hz tone
Busy	.5 seconds on/off 480 Hz/620 Hz dual tone
Reorder	2.5 seconds on/off 480 Hz/620 Hz dual tone
Ringing	440 Hz/480 Hz dual tone; 2 seconds on, 4 seconds off
TPT	2225 Hz Test Progress Tone
Chime	MCI Chime Tone
Bong	AT&T Bong Tone
Beep Beep	2 bursts of 600 Hz Tone
Single tone	Steady single tone
Dual tone	Steady dual tone
Speech?	Suspected speech (unrecognized 0-, 3-, or 4-tone)
Listening	Line status not yet determined
Test in progress	Conducting test

2-2 NAVIGATION

2-2.1 Highlighting and Selecting

To choose a **menu item** or **value**:

1. Use the **Arrow** keys [**↑**, **↓**, **←**, **→**] to move the highlight cursor to the desired menu or menu item. Menus and subordinate menu items then become active when highlighted.
2. Accept a menu selection or argument by pressing **Enter**.

For Example:

1. Use the **Arrow** keys to place the highlight cursor over the word **Tone** in the **Vftest Measurement** menu. (See **Figure 2-4**.) The **Tone** function is now active and ready to be accepted.

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM
System name: s950p3 Wed Aug 21 1996 16:25:59

Connection Status	Line Status
DSC TAD 1-1-1 Tap 1-16-1 rel equip	Dead Line
Tone snd: OFF	Orig Hk: ON Term Hook: ON
	DTMF -----X

Vftest Measurement
Circuit Select

- Tone**
- Noise
- Return Loss
- Dial Digits
- 105-Demand Test
- Impairment Tests
- 23 Tone
- Quit

Press <Enter> to run test
<ctrl-T>: tone send, <ctrl-F>: off-hook, <ctrl-N>: on-hook, <ctrl-R>: rmt audio

Figure 2-4
Highlighting

2. Choose the **Tone** function in the **Vftest Measurement** menu by pressing **Enter** while **Tone** is highlighted. The **Measure Tone** menu appears. (See **Figure 2-5**.)

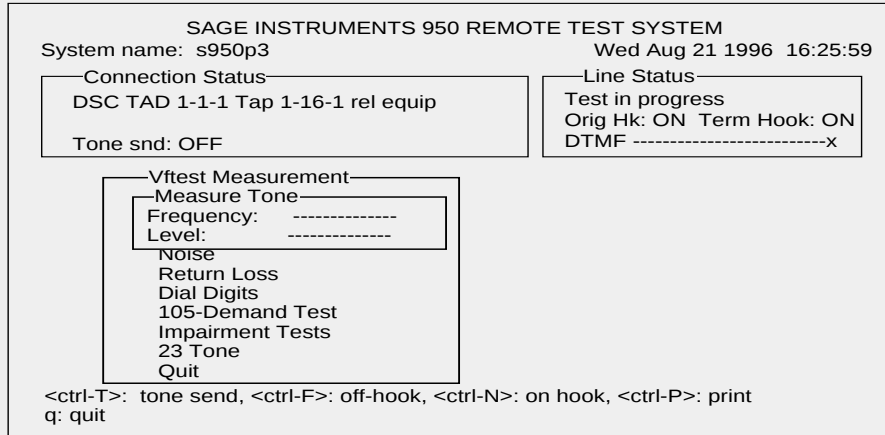


Figure 2-5
Selecting

2-2.2 Shortcut Selection

You can highlight and accept a menu selection with a single action by pressing the key on the keyboard corresponding to the first letter of the item in the active menu. For example, you can refresh the screen by typing **Control-L**.

The following shortcut commands are available for most menus:

- **<ctl-L>** refreshes the screen.
- **<ctl-t>** brings up the Tone Sender Menu.
- **<ctl-c>** clears a field.
- **q** exits a menu and returns to the previous menu.
- **Esc** exits to the previous menu level.

2-3 GETTING STARTED

2-3.1 Starting Vftest

After logging into the system, activate the **VF Demand Testing feature** by entering the **vftest** command at the **UNIX** prompt:

\$ vftest

The first menu to appear upon starting **vftest** is the **Circuit Select** menu.

Section II Demand Testing

The following Test Ports shown are examples only. The actual names may be different in your system. These are defined in the Interface Database file. Refer to the *950RTS Installation and Administration Manual* for a description of the Interface Database.

2-3.1.1 Selecting a Circuit in Direct Mode

Direct Mode enables you to perform VF testing of DSO circuits through a direct T1 connection. (See **Figure 2-6**.)

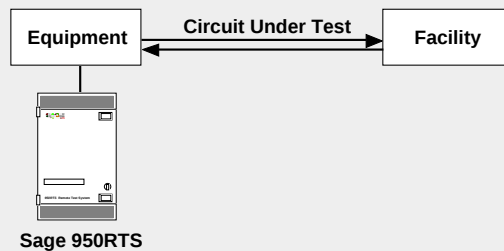


Figure 2-6
Direct Testing

To select a **test port**:

1. Use the **Arrow** keys to highlight the **Test Port** field in the **DS0 Circuit Select** menu. (See **Figure 2-7**.)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM
System name: s950p3 Wed Aug 21 1996 16:25:59
Connection Status Line Status
DS0 Circuit Select
Port Type: Robbed-bit DS0
Port: **DirectT1**
Quit Select
Enter name of test port, or <Space bar> to list. Press <Enter> to confirm.
Press <Tab> for next field, <Esc> to abort circuit selection

Figure 2-7
Selecting a Circuit (Direct Mode)

2. Press the **Space bar**. The **Test Port** menu appears. (See **Figure 2-8**.)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM

System name: s950p3 Wed Aug 21 1996 16:25:59

Connection Status _____ Line Status _____

DS0 Circuit Select _____

Port Type: Robbed-bit DS0

Port: _____

Test Port: **DirectT1**
DCS TAD

Quit Select _____

Enter name of test port, or <Space bar> to list. Press <Enter> to confirm.
Press <Tab> for next field, <Esc> to abort circuit selection

Figure 2-8
Selecting a Test Port

3. Use the **Arrow** keys to highlight a test port to undergo testing.
4. Press **Enter** to accept the selection. You will advance to the **Vftest Measurement** menu.

2-3.1.2 Selecting a Circuit in DCS Mode

DCS mode enables you to perform VF and switched 56kbps testing on DS0 circuits through a **DCS**. (See **Figure 2-9**.)

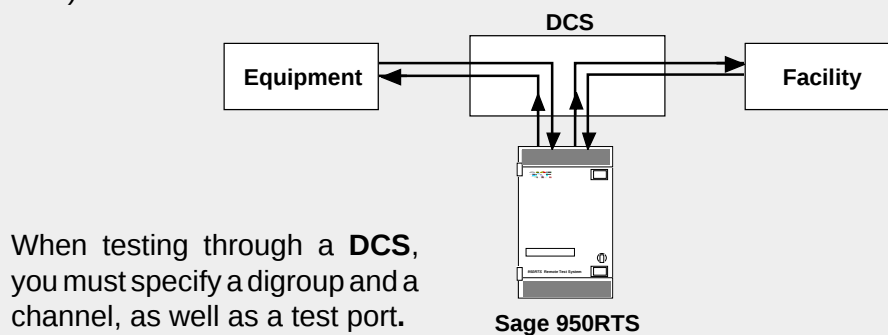


Figure 2-9
DCS Testing

To select a **circuit** if you are using **DCS** control:

1. Use the **Arrow** keys to highlight the **Test Port** field in the circuit select menu. (See **Figure 2-10**.)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM

System name: s950p3 Wed Aug 21 1996 16:25:59

Connection Status

Line Status

DS0 Circuit Select

Port Type: DCS Clear TAP

DCS Type: DSC

Port: **DSC TAD**

Digroup: 1

Chan: 1

Quit Select

Enter name of test port, or <Space bar> to list. Press <Enter> to confirm.
Press <Tab> for next field, <Esc> to cancel circuit select

Figure 2-10
Selecting a Circuit (DCS)

Note: The **Digroup** and **Chan** fields above are **ID fields** for a **DSC** brand **DCS**. These fields will vary, depending on the type of **DCS** selected.

2. Press the **Space bar**. The **Test Port** menu appears. (See **Figure 2-11**.)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM

System name: s950p3 Wed Aug 21 1996 16:25:59

Connection Status

Line Status

DS0 Circuit Select

Port Type: DCS Clear TAP

DCS Type: DSC

Port: **Test Port**

Digroup: 1

Chan: 1

Quit Select

Enter name of test port, or <Space bar> to list. Press <Enter> to confirm.
Press <Tab> for next field, <Esc> to cancel circuit select

Figure 2-11
Selecting a Test Port

3. Use the **Arrow** keys to highlight a **test port**.
4. The circuit ID fields under the **Port** field (i.e. **Digroup**, **Channel**) vary depending on the DCS type selected. Change these by highlighting them with the **Arrow** keys, then entering a new value.
5. Use the **Arrow** keys to highlight the **Quit Select** field. Press **Enter** to accept the selection. You will advance to the **DS1 BERT Control** menu.

2-4 TEST SETUP

Upon selecting a circuit to test you will advance to the **vfctest** screen. (See **Figure 2-12.**)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM	
System name: s950p3	Wed Aug 21 1996 16:25:59
Connection Status	Line Status
DSC TAD 1-1-1 Tap 1-16-1 rel equip	Dead Line
Tone snd: OFF	Orig Hk: ON Term Hook: ON
	DTMF -----x
Vfctest Measurement	
Circuit Select	
Tone	
Noise	
Return Loss	
Dial Digits	
105-Demand Test	
Impairment Tests	
23 Tone	
Quit	
Press <Enter> to change circuit under test	
<ctrl-T>: tone send, <ctrl-F>: off-hook, <ctrl-N>: on-hook, <ctrl-R>: rmt audio	

Figure 2-12
Main Test Screen

2-4.1 Tone Sender Control

The **Tone Sender Control** menu enables the user to:

- turn the tone on and off
- set the frequency (**20 Hz to 3904 Hz**)
- set the signal level (**-60 dBm to +3.0 dBm**)

Type **<ctl-t>** to enter the **Tone Sender Control** menu from any menu. (See **Figure 2-13.**)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM

System name: s950p3 Wed Aug 21 1996 16:25:59

Connection Status

DSC TAD 1-1-1 Tap 1-16-1 rel equip

Tone snd: OFF

Line Status

Test in progress

Orig Hk: ON Term Hook: ON

DTMF -----x

Vftest Measurement

Circuit Select

Tone

Noise

Return Loss

Dial Digits

105-Demand Test

Impairment Tests

23 Tone

Quit

Tone Sender Control

Sender: **OFF**

Frequency: 1004 Hz

Level: -16.0dBm

Send echo canceller disable tone

<Space bar> toggles sender on/off
Arrow keys to highlight, q to leave tone sender menu

Figure 2-13
Toggling Tone Sender Control ON/OFF
(Direct Mode)

2-4.1.1 Turning Tone Sender On and Off

The **ON/OFF Field** is automatically highlighted when the **Tone Sender Control** menu appears. (See **Figure 2-9.**)

Toggle between **ON** and **OFF** by using the **Space bar**.

If you set Tone to **OFF**, the message in the **Connection Status Box** will read: **Tone Send: OFF** (See **Figure 2-15.**)

If you set **Tone** to **ON**, the **Connection Status Box** will display the selected send frequency and tone levels. (See **Figure 2-14.**)

Connection Status

DSC TAD 1-1-1 Tap 1-16-1 rel equip

Tone snd: Tone 1004 Hz @ -16.0 dBm

Figure 2-14
Connection Status Box in DCS Mode
Tone Sender On

2-4.1.2 Setting the Frequency

To set the **frequency** of the tone you are sending:

1. Move to the **Frequency Field** in the **Tone Sender Control Menu** using the **Arrow** keys. (See **Figure 2-15.**)

2. Use the number keys to enter a new **frequency** value.

- A valid **frequency** value is between **20** and **3904 Hz**.

3. Press **Enter** to accept the value.

2-4.1.3 Setting the Level

To set the **level** of the tone you are sending:

1. Highlight the **Level** field using the **Arrow** keys, unless it has been automatically highlighted as described in step three of **Section 2-4.2.2**.

2. Use the number keys to enter a new **level** value.

- A valid **level** value is between **+3.0** and **-60.0 dBm**.

3. Enter the **"-"** sign to specify a negative level. When entering a positive level, a sign is unnecessary.

4. Press **Enter** to accept the signal level value.

2-4.2 Remote Audio

The **950RTS** remote audio feature is a combination of **remote audio monitor** and an **order-wire circuit**. The **remote audio monitor** allows you to monitor the audio path of a circuit under test. The **order-wire mode** allows you to talk over the circuit under test to a person located at the far end while tests are not being performed.

When the remote audio feature is enabled, the **950RTS** automatically switches from the order-wire mode to the remote audio mode when you start a test. When you stop the test, the **950RTS** returns from the remote audio mode to the order wire mode.

The remote audio is only available from vftest applications when a test is not running. You can select the remote audio feature by pressing **ctrl-r** from the main **Vftest Measurement** menu.

To use the **remote audio** feature:

Press **ctrl-r** from the **Vftest Measurement** menu. The **Remote Audio** menu appears.
(See **Figure 2-15**.)

2-4.2.1 Port

To select a **port**:

1. Use the **Arrow** keys to highlight the **Port** field.

Section II Demand Testing

2. Type the name of the desired port, **OR** press the **Space bar** to display the **Test Port** menu, use the **Arrow** keys to highlight the desired port corresponding to the remote audio dial out line, and press **Enter**. (See **Figure 2-15**.)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM

System name: s950p3 Wed Aug 21 1996 16:25:59

Connection Status
DSC TAD 1-1-1 Tap 1-16-1 rel equip
Tone snd: OFF

Line Status
Test in progress
Orig Hk: ON Term Hook: ON
DTMF -----X

Vftest Measurement
Remote Audio
Port: **r202**
Setup
Quit

Dial Digits
105-Demand Test
Impairment Tests
23 Tone
Quit

Enter name of audio port, or <Space bar> to list. Press <Enter> to confirm.
Press <Tab> for next field>, <Esc> to cancel remote audio setup

Figure 2-15
Remote Audio

If you have a port named **RemoteAudio**, the **950RTS** will search for that port first and display it in the Port field automatically when you enter the **Remote Audio** menu.

2-4.2.2 Setup

By selecting the **Setup** field in the **Remote Audio** menu, you will advance to the Remote Audio **Dial Digits** menu. This menu enables you to enter a remote audio phone number and place a call to set up a line for remote audio use.

The Remote Audio **Dial Digits** menu is identical in format to the main Dial Digits menu accessed from the **Vftest Measurement** menu. See **Section 2-5.4** for a complete description on how to use the DialDigits feature.

2-4.3 VF Test Setup

The **vftest measurement** function enables you to set up several **vftest** measurements:

- **Tone**
- **Noise**
- **Return Loss**
- **Dial Digits**
- **105-Demand Tests** (purchased option)
- **Impairment Tests** (purchased option)
- **23 Tones** (purchased option)

In addition, you can change circuits to test from the vftest Measurement menu without exiting and reentering the vftest function.

2-4.3.1 Direct Mode

In **Direct Mode**, you can change circuits to test from the **Vftest Measurement** menu without exiting and reentering the vftest function. (See **Figure 2-16**.)

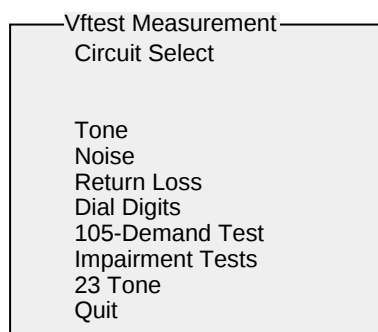


Figure 2-16
Vftest Measurement Menu
(Direct Mode)

To change circuits from within the **Vftest Measurement** menu:

1. Use the **Up and Down Arrow** keys to select the **Circuit Select** field.
2. Press **ENT** and then the **Space bar**. You will return to the **Circuit Select** menu. Follow the circuit selection procedure described in **Section 2-3.1.1, Selecting a Circuit in Direct Mode**.

**Demand
Testing**

2-4.3.2 DCS Mode

The **Vftest Measurement** menu includes fields for switching modes from release to monitor to split, and for changing the test direction (toward the equipment or facility). (See **Figure 2-17**.)

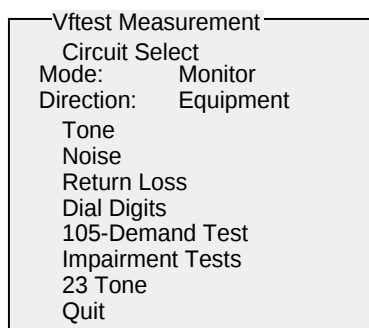


Figure 2-17
Vftest Measurement Menu
(DCS Mode)

To change circuits from within the **Vftest Measurement Menu**:

1. Use the **Up and Down Arrow** keys to select the **Circuit Select** field.
2. Press **ENT** and then the **Space bar**. You will return to the **Circuit Select** menu. Follow the circuit selection procedure described in **Section 2-3.1.2, Selecting a Circuit in DCS Mode**.

2-4.3.2.1 Channel States

When a channel is selected in **DCS** mode, you must place it in one of 3 states:

Monitor Mode

In this mode, you can check the status of the circuit to undergo testing without interrupting the circuit or transmitting test data. The circuit is monitored in both directions (towards both the equipment and facility sides) to check for In Band Signalling and busy conditions. This must be done before starting an intrusive test. (See **Figure 2-18**.)

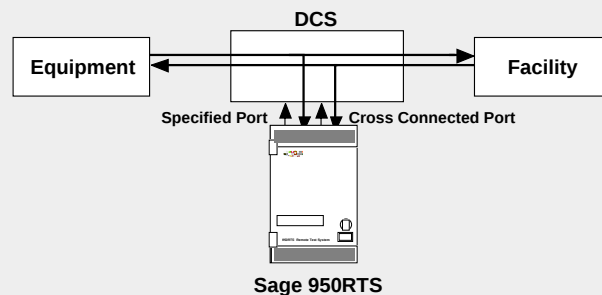


Figure 2-18
Monitor Mode

Split Mode

In this mode, you can perform intrusive tests on the designated channel. The channel under test is split and the **950RTS** tests either towards the equipment side or the facility side. (See **Figure 2-19**.)

Important! You should only enter this mode after monitoring the channel, otherwise, you may disrupt activity on the line if it is busy.

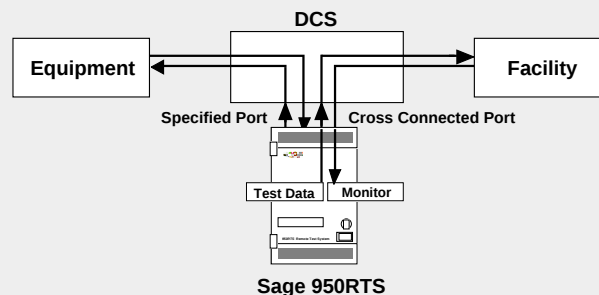


Figure 2-19
Split Mode

Release Mode

Enter this mode after testing to release the channel for use. You cannot enter back into **Split Mode** directly without entering **Monitor Mode** first. (See **Figure 2-20**.)

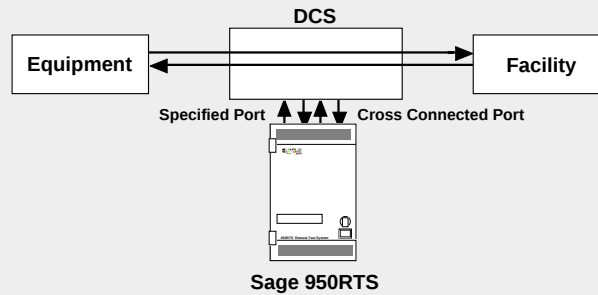


Figure 2-20
Split Mode

To select a **mode**:

1. Use the **Up and Down Arrow** keys to select the **Mode** field from the **Vftest Measurement** menu.
2. Press the **Space bar**. The **DCS Circuit** menu appears. (See **Figure 2-21**.)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM

System name: s950p3 Wed Aug 21 1996 16:25:59

Connection Status

DSC TAD 1-1-1 Tap 1-16-1 rel equip

Tone snd: OFF

Line Status

Vftest Measurement

Circuit Sel DCS Circuit

Mode: Release

Direction: **Monitor**

Split

Tone

Noise

Return Loss

Dial Digits

105-Demand Test

Impairment Tests

23 Tone

Quit

Type 'M' to monitor, 'S' to split, 'R' to release DCS circuit
<ctrl-T>: tone send, <ctrl-F>: off-hook, <ctrl-N>: on-hook

Figure 2-21
DCS Circuit Menu

3. Use the **Up and Down Arrow** keys to select a **channel state** and press **ENT**. You will be returned to the **Vftest Measurement** menu.

2-4.3.2.2 Test Direction

The test access mechanism of the **DCS** allows the **950RTS** to non-intrusively monitor a DS-0 cross connect, or intrusively split a cross connect. In both case, the **DCS** provides access to both directions of the cross connected circuit. The **950RTS** uses the terms **facility** and **equipment** to designate these two directions of test. The port that is used to identify the cross connect is designated the equipment direction, while the port on the other end of the **DCS** cross connect is designated the facility direction. In the switching environment, therefore, all DS0 circuits should be referred to by the port that connects to the switch so that the equipment direction of test will mean testing towards the switch, while the facility direction of test will refer to testing towards the transmission facilities.

When you are in **Split Mode**, you will need to select a direction in which to test. You can test either towards the equipment side or facility side of the **DCS**.

To select a **test direction**:

1. Use the **Up and Down Arrow** keys to select the **Direction** field.
2. Press the **Space bar** to toggle between **equipment** and **facility**.

2-5 SELECTING AND PERFORMING TESTS

Selection and performance of tests in the **Vftest Measurement** menu operates the same whether you are in **Direct** or **DCS Mode**. The following sections describe the seven different test types performed by the **950RTS**.

2-5.1 Measure Tone

The **Measure Tone** function displays the frequency and level of a tone on the circuit under test. The tone may be generated by far end equipment or it may be a loop-back tone generated by the **950RTS**.

To display the results of a **Measure Tone Test**:

1. Use the **Arrow** keys to select the tone measurement test by highlighting **Tone** in the **Vftest Measurement** menu.
2. Press **Enter** to begin the test. The measurement results are displayed in the **Measure Tone** result box. (See **Figure 2-22**.)

```

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM
System name: s950p3                      Wed Aug 21 1996 16:25:59
Connection Status                        Line Status
DSC TAD 1-1-1 Tap 1-16-1 rel equip      Test in progress
Tone snd: OFF                           Orig Hk: ON Term Hook: ON
                                         DTMF -----X
Vftest Measurement
Measure Tone
Frequency: -----
Level: -----
Noise
Return Loss
Dial Digits
105-Demand Test
Impairment Tests
23 Tone
Quit
<ctrl-T>: tone send, <ctrl-F>: off-hook, <ctrl-N>: on hook, <ctrl-P>: print
q: quit
    
```

Figure 2-22
Measure Tone Test Result Box

3. Press **Esc** to terminate the measurement and return to the **Vftest Measurement** menu.

2-5.2 Measure Noise

The **Measure Noise** function measures the noise level received through one of three filters:

- **C-notch** (measures all voiceband frequencies except 1010 Hz)
- **C-message** (limits measurement to the middle of the voiceband range)
- **3K Flat** (measures all voiceband frequencies below 3KHz). This function also measures **Signal-to-Noise Ratio (SNR)**.

(**SNR**=Noise through a C-Notch filter/-16.0dBrn.)

C-Notch's SNR is normally done while a **1004-1010Hz** tone is receiving. **SNR** is not valid without a holding tone.

To display the results of a **Measure Noise Test**:

1. Use the **Arrow** keys to select the **Measure Noise** function by highlighting the **Noise** field in the **Vftest Measurement** menu.
2. Press **Enter**. The **Measure Noise Test** results box will appear. (See **Figure 2-23**.)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM

System name: s950p3 Wed Aug 21 1996 16:25:59

Connection Status

DSC TAD 1-1-1 Tap 1-16-1 rel equip

Tone snd: OFF

Line Status

Test in progress

Orig Hk: ON Term Hook: ON

DTMF -----x

Vftest Measurement

Measure Noise

Type	Noise Level
C-Notch	0.0 dBmC
C-Msg	-----
SNR	-----
3K Flat	-----

105-Demand Test

Impairment Tests

23 Tone

Quit

<ctrl-T>: tone send, <ctrl-F>: off-hook, <ctrl-N>: on-hook, <ctrl-P>: print
 Arrow keys: select measurement, q: quit

Figure 2-23
Measure Noise Test Result Box

3. Use the **Up and Down Arrow** keys to move down the list of tests. The measured noise values are displayed next to the name of each test as it is selected.
4. After the test is finished, type **q** to return to the **Vftest Measurement** menu.

The value for a noise measurement will only be updated while its field is highlighted. The value fields are updated approximately once every second. The last measurement continues to be displayed after the highlight cursor is moved to another field. To see updated values, simply move the cursor back to the appropriate field.

2-5.3 Return Loss

Return Loss is the ratio of the power of an outgoing signal to the power of the signal reflected back from an impedance discontinuity. The **Return Loss** feature performs three different functions with a user selected tone:

- **Echo Return Loss** (measures the echo on a circuit, normally caused by crosstalk)
- **Singing Return Loss High (SRH)** (measures singing loss at the high end of the voiceband)
- **Singing Return Loss Low (SRL)** (measures singing loss at the low end of the voiceband)

To display the results of a **Return Loss Test**:

1. Use the **Arrow** keys to highlight the **Return Loss** field in the **Vftest Measurement** menu.
2. Press **Enter**. The **Return Loss Test** results box appears. (See **Figure 2-24**.)

```

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM
System name: s950p3                               Wed Aug 21 1996 16:25:59

Connection Status: DSC TAD 1-1-1 Tap 1-16-1 rel equip
Tone snd: Rtn Loss

Line Status:
Test in progress
Orig Hk: ON Term Hook: ON
DTMF -----x

Vftest Measurement
Return Loss
Type Loss
ERL 99.9 dB
SRL -----
SRH -----

Dial Digits
105-Demand Test
Impairment Tests
23 Tone
Quit

<ctrl-F>: off-hook, <ctrl-N>: on-hook, <ctrl-P>: print
Arrow keys: select measurement, q: quit
    
```

Figure 2-24
Select ERL/SRL/SRH Measurements

3. Use the **Arrow** keys to move down the list of tests. The measured loss values appear next to the name of each test as it is selected.

The highlight cursor must remain on the measurement until the test is complete or no value will be displayed. Values in the highlighted fields are updated at approximately one second intervals. This works like the **Measure Noise** menu.

4. Type **q** to terminate the function and return to the **Vftest Measurement** menu.

2-5.4 Dial Menu

The **Dial** function displays choices for **Format**, **Digits**, **Extended Dialing** and **Quit**.

- Use the **Format** field to select the dial format.
- Use the **Digits** field to set the digits to be dialed.
- Use the **Extended Dialing** function to transmit multiple digit sequences.
- Select **Quit** to return to the **Vftest Measurement** menu.

To access the **Dial Menu**:

1. Use the **Arrow** keys to highlight the **Dial Field** in the **Vftest Measurement** menu.
2. Press **Enter**. The **Dial** menu appears. (See **Figure 2-25**.)

The screenshot displays the SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM interface. At the top, it shows the system name 's950p3' and the date/time 'Wed Aug 21 1996 16:25:59'. Below this, there are two main sections: 'Connection Status' and 'Line Status'. The 'Connection Status' section includes 'DSC TAD 1-1-1 Tap 1-16-1 rel equip' and 'Tone snd: OFF'. The 'Line Status' section includes 'Dead Line', 'Orig Hk: ON', 'Term Hook: ON', and 'DTMF -----x'. In the center, there is a 'Vf test Measurement' menu with options: 'Dial', 'Orig Supervision: ON HOOK', 'Format: DTMF', 'Digits: -----', 'Extended Dialing', and 'Quit'. Below this menu, there is another section with options: '105-Demand Test', 'Impairment Tests', '23 Tone', and 'Quit'. At the bottom, there are instructions: '<Space bar>: cycle between DTMF and MF' and 'q: quit'.

Figure 2-25
Dial Menu

2-5.4.1 Supervision

To select a **origin supervision**:

1. Use the **Arrow** keys to scroll down to the **Orig Supervision** field and press **Enter** to select it. (See **Figure 2-25**.)
2. Press the **Space bar** to toggle between **ON HOOK** and **OFF HOOK**.

2-5.4.2 Format

To select a **dial format**:

1. Use the **Arrow** keys to scroll down to the **Format** field and press **Enter** to select it. (See **Figure 2-25**.)
2. Press the **Space bar** to toggle between **DTMF** and **MF**.

2-5.4.3 Digits

To enter the digits to be dialed in the **Enter Digits** field:

1. Use the **Arrow** keys to highlight the **Digits** field. (See **Figure 2-26**.)
2. Enter up to **18 digits** to be dialed by the **950RTS**.
3. Press **Enter** to accept the number. If not off-hook, the **950RTS** will go off-hook automatically, then send the digits.

```

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM
System name: s950p3                      Wed Aug 21 1996 16:25:59
Connection Status: DSC TAD 1-1-1 Tap 1-16-1 rel equip
Tone snd: OFF
Line Status: Dead Line
Orig Hk: ON Term Hook: ON
DTMF -----X

Vftest Measurement
Dial
  Orig Supervision: ON HOOK
  Format: DTMF
  Digits: 
  Extended Dialing
  Quit
105-Demand test
Impairment Tests
23 Tone
Quit

Enter digits: <ctrl-C>: clear, <Enter>=accept valid digits 0-9,A-D,*,#
q: quit
  
```

Figure 2-26
Entering Dial Digits

The arguments to follow the **Enter Digits** field must be appropriate to the signaling method selected in the **Format** field. The dialing format appears in the **Dial** menu. The appropriate arguments appear on a comment line at the bottom of the screen.

2-5.4.4 Extended Dialing

To send up to four digit sequences simultaneously:

1. Use the **Arrow** keys to highlight the **Extended Dialing** field.
2. Press **Enter**. The **Extended Dialing** menu appears. (See **Figure 2-27**.)

```

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM
System name: SCO_SV                      Mon Jan 30 1996 16:25:59
Connection Status: DSC TAD 1-1-1 Tap 1-16-1 rel equip
Tone snd: OFF
Line Status: Dead Line
Orig Hk: ON Term Hk: ON
DTMF -----X

Vftest Measurement
Extended Dialing
  Seq  Pre-Dial  Format  Digits
  1    Immediate  DTMF
  2     off      DTMF
  3     off      DTMF
  4     off      DTMF
  Send Digit Sequence
  Return to Basic Dialing
  Quit

Select digit format: MF or DTMF
<space> for selection menu, then <enter> to select
  
```

Figure 2-27
Extended Dialing

2-5.4.4.1 Digit Sequences

To enter a digit sequence:

1. Use the **Arrow keys** to highlight the appropriate line in the **Digits** field.
2. Use the **number keys** to enter a number to be dialed.

2-5.4.4.2 Digit Format

To change the dial digit format:

1. Use the **Arrow keys** to highlight the appropriate line in the **Format** field.
2. Press the **Space bar**. The **Digit Format** menu appears. (See **Figure 2-28**.)

The screenshot displays the SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM interface. At the top, it shows the system name 's950p3' and the date/time 'Wed Aug 21 1996 16:25:59'. Below this, there are two main sections: 'Connection Status' and 'Line Status'. The 'Connection Status' section shows 'DSC TAD 1-1-1 Tap 1-16-1 rel equip' and 'Tone snd: OFF'. The 'Line Status' section shows 'Dead Line', 'Orig Hk: ON', 'Term Hook: ON', and 'DTMF -----X'. In the center, there is a 'Vf test Measurement' section with a 'Send dial sequences' menu. This menu has four options: '1 Immediate', '2 off', '3 off', and '4 off'. To the right of this menu is a 'Digit Format' menu with two options: 'MF' and 'DTMF'. The 'DTMF' option is highlighted. Below the 'Send dial sequences' menu, there are options for 'Send Digit Sequence', 'Quit', and 'return to basic dial menu'. At the bottom, there is a prompt: 'Select digit format: MF or DTMF <space> for selection menu, then <enter> to select'.

Figure 2-28
Digit Format Menu

3. Use the **Arrow keys** to highlight either the **DTMF** or **MF** format.
4. Press **Enter**. You will be returned to the **Extended Dialing** menu.

2-5.4.4.3 Pre-Dial

The **pre dial** feature allows you to add signaling events, including winks and pauses, to the beginning of your dial sequence. Adding a wink tells the **950RTS** to wait for a wink before sending the digit sequence. Adding a pause injects a two second delay before the **950RTS** sends the digits.

To add a predial signaling event:

1. Use the **Arrow keys** to highlight the appropriate line in the **Pre-Dial** field.
2. Press the **Space bar**. The **Pre-Dial** menu appears. (See **Figure 2-29**.)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM
 System name: s950p3 Wed Aug 21 1996 16:25:59
 Connection Status: DSC TAD 1-1-1 Tap 1-16-1 rel equip Line Status: Dead Line
 Tone snd: OFF Orig Hk: ON Term Hook: ON
 DTMF -----X

Vftest Measurement
 Send dial sequences

Seq	Pre-Dial	Digits
1	Immediate	
2	Pause	
3	Wink	
4		

Send Digit Sequence
 Quit
 return to basic dial menu

Select digit format: MF or DTMF
 <space> for selection menu, then <enter> to select

Figure 2-29
Pre Dial Menu

3. Use the **Arrow keys** to highlight **immediate**, **pause**, or **wink**. **Immediate** will send the sequence immediately, **pause** will tell the **950RTS** to add a 2 second delay before sending, and **wink** will tell the **950RTS** to wait for a wink from the terminating end before sending.
4. Press **Enter**. You will be returned to the **Extended Dialing** menu.

2-5.4.4.4 Send Digit Sequences

To send the digit sequences:

1. Use the **Arrow keys** to highlight the **Send Digit Sequence** field.
2. Press **Enter**. The digit sequences entered will be sent.

2-5.4.5 Quit

To quit the **Extended Dialing** menu:

1. Use the **Arrow keys** to highlight the **Quit** field to return to the **Vftest Measurement** menu, or highlight the return to **Basic Dialing** field to return to the **Dial** menu.
2. Press **Enter** to activate your selection.

2-6 QUIT VFTEST

To exit **vftest**:

1. Use the **Arrow** keys to select **Quit** from the **Vftest Measurement Menu**.
2. Press **Enter**. A dialog box appears at the center of the screen asking you to confirm the quit instruction.
3. Use the **Arrow** keys to select yes or no from the confirmation dialog box.
4. Press **Enter**. The **vftest** is terminated and the screen returns to the network or server prompt. All tests will be terminated and DCS will be released.

SECTION III
DEMAND TESTING OPTIONS

3-1 OPTIONS

You may choose to purchase options that expand the testing capabilities of the **950RTS**. This section describes the available demand testing options and their operating procedures.

3-1.1 Demand-105

The **Demand-105** function allows you to perform various on-demand tests on Type-105 test lines. These include standard type-105 tests: **Loss, Noise, Noise/Tone, Gain Slope, ERL, SRL, and SRH**. During **Demand-105** testing, the **950RTS** acts as a near-end and assumes that a **Type-105 responder** is connected at the far-end.

To perform **demand-105** tests:

- 1. Use the **Arrow keys** to select **Demand-105** from the **Vftest Measurement** menu. You will advance to the **Demand 105 Testing** menu. (See **Figure 3-1**)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM

System name: SCO_SV

Mon Jan 30 `1996 16:25:59

Connection Status

Line Status

Brd#1 Iface#2 Chan#1

Dead Line

Sw 56 Kb 511

Orig Hk: ON Term Hk: ON

DTMF -----x

Demand 105 Testing

Seq Pre-dial Format Digits

1 Immediate DTMF

2 off DTMF

3 off DTMF

4 off DTMF

Start test

Quit

Results file: <none>

[X]-Loss [X]-ERL

[X]-Noise [X]-SRL

[X]-Noise/Tone [X]-SRH

[X]-Gain/Slope

Select action: wait for Wink, 2 second Pause, send Immediately, or sequence Off

<space> for selection menu, then <enter> to select

Figure 3-1
Demand 105 Testing Menu

Demand
Test Options

3-1.1.1 Predial

The **pre dial** feature allows you to add signaling events, including winks and pauses, to the beginning of your dial sequence. Adding a wink tells the **950RTS** to wait for a wink before sending the digit sequence. Adding a pause injects a two second delay before the **950RTS** sends the digits.

To add a **predial signaling event**:

1. Use the **Arrow keys** to highlight the appropriate line in the **Pre-Dial** field.
2. Press the **Space bar**. The **Pre-Dial** menu appears. (See **Figure 3-2**)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM
System name: SCO_SV Mon Jan 30 1996 16:25:59

Connection Status		Line Status	
Brd#1	Iface#2 Chan#1	Dead Line	
Sw	56 Kb 511	Orig Hk: ON	Term Hk: ON
		DTMF	-----x

Demand 105 Testing		
Seq	Pr	Pre-Dial
1	Imi	Immediate
2		Pause
3		Wink
4	off	DTMF

Start test
Quit
Results file: <none>

[X]-Loss	[X]-ERL
[X]-Noise	[X]-SRL
[X]-Noise/Tone	[X]-SRH
[X]-Gain/Slope	

Select action: wait for Wink, 2 second Pause, send Immediately, or sequence Off
<space> for selection menu, then <enter> to select

Figure 3-2
Pre Dial Menu

3. Use the **Arrow keys** to highlight **immediate**, **pause**, or **wink**. **Immediate** will send the sequence immediately, **pause** will tell the **950RTS** to add a 2 second delay before sending, and **wink** will tell the **950RTS** to wait for a wink from the terminating end before sending.
4. Press **Enter**. You will be returned to the **Digit Dial Sequences** menu.

3-1.1.2 Selecting a Dial Format

To set up each dial sequence in **DTMF** or **MF** format:

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM

System name: SCO_SV

Mon Jan 30 1996 16:25:59

Connection Status

Line Status

Brd#1 Iface#2 Chan#1

Dead Line

Sw 56 Kb 511

Orig Hk: ON Term Hk: ON

DTMF -----x

Demand 105 Testing

Seq	Pre-dial	Format	Digit Format
1	Immediate	D	MF
2	off	D	DTMF
3	off	D	
4	off	DTMF	

Start test

Quit

Results file: <none>

[X]-Loss

[X]-ERL

[X]-Noise

[X]-SRL

[X]-Noise/Tone

[X]-SRH

[X]-Gain/Slope

Select digit format: MF or DTMF

<space> for selection menu, then <enter> to select

Figure 3-3
Selecting a Digit Format

1. Use the **Arrow** keys to highlight each **Format** field. (See **Figure 3-1**)
2. Use the **Space bar** to cycle between **DTMF** and **MF**.

Digit formats for each of the digit strings may be mixed between **MF** and **DTMF**.

3-1.1.3 Entering Numbers to Dial

Extended Dialing is especially useful if you want to send a combination of DTMF and MF access numbers. While entering MF digit strings, a **K** (for KP) and an **S** (for ST) may be needed to complete a call.

To enter a **number-to dial**:

1. Use the **Arrow** keys to highlight each **Digits** field. (See **Figure 3-4**)

Figure 3-4 Entering Dialing Digits

- ### 3-1.1.4 Selecting Tests to Perform

1. Use the **Arrow** keys to highlight a **test checkbox**. (See **Figure 3-5**)

Figure 3-5 Selecting Tests

- Use the **Space bar** to toggle a test **ON/OFF**. (an x in a box indicates that you have selected it.)

3-1.1.5 Starting the Test

To begin a **Demand-105 Test**:

- Use the **Arrow** keys to move the highlight cursor to the **Start Test** field.
- Press **Enter** to begin the test. A results box appears on the screen displaying the results of each enabled test as it is completed. (See **Figure 3-6**)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM

System name: SCO_SV Mon Jan 30 `1996 16:25:59

Connection Status:

Brd#1 Iface#2 Chan#1

Tone snd: OFF

Line Status:

Test in progress

Orig Hk: ON Term Hk: ON

DTMF -----x

Demand 105 Testing

Trunk #: 12345	Post-Trunk Digits: 12345	
Call complete:	Connect time 14.2	
Loss	Near: 0.0 dB	Far: 0.1 dB
Noise	Near: 0.0 dBBrnC	Far: 15.5 dBBrnC
Noise/Tone	Near: 35.1 dBBrnC	Far: 36.5 dBBrnC
Slope 404	Near: 0.1 dB	Far: 0.1 dB
Slope 1004	Near: 0.0 dB	Far: 0.1 dB
Slope 2804	Near: 0.0 dB	Far: 0.1 dB
ERL	Near: 99.0 dB	Far: 39.5 dB
SRL	Near: 99.0 dB	Far: 39.5 dB
SRH	Near: 99.0 dB	Far: 39.5 dB

Type 'q' to abort test

Figure 3-6
Demand-105 Test Results Box

- When the test is completed, type **q** to quit the results box and then press any key to return to the **Demand-105** setup screen.

3-1.1.6 Quitting the Test

To quit a **Demand-105 Test**:

- Use the **Arrow** keys to move the highlight cursor to the **Quit** field.
- Press **Enter** to abort the **Demand-105** function without testing. You will return to the **Vftest Measurement** menu.
- Use the **Arrow** keys to highlight **Quit** in the **Vftest Measurement** menu.
- Press **Enter**.

Shortcut: The **Esc** key will abort the **Demand-105** function and return to the **Vftest Measurement** menu.

3-1.2 Impairment Tests

The Impairment test option enables the user to perform the six on-demand impairment tests that most affect data transmissions: **Intermodulation distortion**, **Impulse Noise & Hits**, **Phase and Amplitude Jitter**, **Envelope Delay Distortion**, **Absolute Delay**, and **Peak-to-Average ratio**.

3-1.2.1 Intermodulation Distortion

Intermodulation distortion is a form of nonlinear distortion caused by two or more waves passing through a nonlinear device such as an amplifier, modulator, demodulator, or switch. The frequencies produced are the sum and difference products of the input frequencies and their harmonics. In **PCM**, this distortion is caused by the quantizing process, which uses the logarithmic compression law to provide more steps per volt for small speech signals than for large samples.

Measuring nonlinear distortion involves measuring a complex signal consisting of two pairs of tones (**857 Hz, 863 Hz** and **1372 Hz, 1388 Hz**). The second order distortion products occur around **520 Hz** and **2240 Hz** while the third order products occur around **1900 Hz**. Measurements are expressed in terms of dB below the received value of the desired signal.

To measure **intermodulation distortion**:

1. Use the **Up/Down Arrow** keys to select impairment tests from the **Vf test Measurement** menu.
2. Press **Enter**. The **Impairment Tests** menu appears. (See **Figure 3-7**)

The screenshot displays the SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM interface. At the top, it shows the system name 'SCO_SV' and the date/time 'Mon Jan 30 `1996 16:25:59'. Below this, there are two main sections: 'Connection Status' and 'Line Status'. The 'Connection Status' section shows 'Brd#1 Iface#2 Chan#1' and 'Tone snd: Tone 1004 Hz @ -16.0 dBm'. The 'Line Status' section shows 'Test in progress', 'Orig Hk: ON', 'Term Hk: ON', and 'DTMF -----x'. In the center, the 'Vf test Measurement' menu is open, showing a list of impairment tests: 'Distortion - Intermod', 'Impulse Noise & Hits', 'Jitter - Phase & Amplitude', 'Envelope Delay Distortion', 'Absolute Delay', 'Peak/avg ratio (PAR)', and 'Quit Impairments Tests'. Below this list, there are options for '23 Tone' and 'Quit'. At the bottom, there are instructions: '<ctrl-T>: tone sender, <ctrl-F>: off-hook, <ctrl-N>: on-hook' and 'Arrow keys: select measurement'.

Figure 3-7
Impairment Tests Menu

3. Use the **Up/Down Arrow** keys to select **Distortion - Intermodulation** from the **Impairment tests** menu.

- Press **Enter**. The 4-Tone Intermodulation Distortion test results appear. (See **Figure 3-8**)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM

System name: SCO_SV

Connection Status

Brd#1 Iface#2 Chan#1

Tone snd: Tone 1004 Hz @ -16.0 dBm

Mon Jan 30 `1996 16:25:59

Line Status

Test in progress

Orig Hk: ON Term Hk: ON

DTMF -----x

Vftest Measurement

Impairment Tests

4-Tone Intermodulation Distortion

Sending: OFF

Rcv Status: No Tone

2nd Order IMD -----

3rd Order IMD -----

Quit Impairments Tests

23 Tone

Quit

<Space bar> to cycle through 2-tone: 4-tone: off
 <ctrl-F>: off-hook, <ctrl-N>: on-hook, <ctrl-P>: print, 'q' : quit

Figure 3-8
4-Tone Intermodulation Distortion Menu

- Press the **Space** key to cycle through the **Sending** options:

- **OFF**
- **2-Tone SNR** (Signal-to-Noise Ratio)
- **4-Tone IMD Measurement**

- Type **q** to terminate the measurement and return to the **Impairment Tests** menu.

3-1.2.2 Impulse Noise & Hits

Impulse Noise & Hits is measured by counting the number of times noise spikes exceed preset threshold levels. The **950RTS** uses three threshold levels (Low, medium, and high) which are spaced 4 dB apart.

This function also measures **phase hits**, **gain hits**, and **dropouts**. **Phase hits** are sudden changes in the frequency of a received signal lasting longer than **4 ms**. **Gain hits** are transient impairments signified by an increase or decrease in the received signal amplitude. They are less than **12 dB** change in level and last longer than **12 ms**. **Dropouts** are decreases in the received signal level greater than **12 db** and lasting longer than **4 ms**. Each of these types of distortion can cause data errors to occur.

To measure **impulse noise & hits**:

- Use the **Up/Down Arrow** keys to select **Impulse Noise & Hits** from the **Impairment Tests** menu.
- Press **Enter**. The **Impulse Noise & Hits** menu appears. (See **Figure 3-9**)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM

System name: SCO_SV

Connection Status

Brd#1 Iface#2 Chan#1

Tone snd: Tone 1004 Hz @ -16.0 dBm

Mon Jan 30 `1996 16:25:59

Line Status

Test in progress

Orig Hk: ON Term Hk: ON

DTMF -----x

Vftest Measurement

Impairment Tests

Impulse Noise and Hits Setup

Phase threshold: **20**deg

Gain threshold: 3 dB

Phase/gain meas rate 8/sec

Impulse thresh (low): 67 dBmC

Impulse thresh spread: 4 dB

Text time: Continuous

Start test

Quit

Impulse Noise and Hits Results

Test status: -----

Elapsed time: -----

Receive status: -----

Phase hits: -----

Gain hits: -----

Dropouts: -----

Impulse low: -----

Impulse med: -----

Impulse high: -----

Enter phase threshold, 1 to 45 degrees
s: start test, q: quit

Figure 3-9
Impulse Noise & Hits Setup and Test Results Menus

3. Use the **Up/Down Arrow** keys to scroll through the setup parameters. The help line at the bottom of the screen changes to display the acceptable values that can be entered for the highlighted parameter.

The following is a list of acceptable values:

Phase Threshold:	enter a value between 1 and 45 degrees
Gain Threshold:	enter a value between 1 and 10 dB
Phase/gain Measurement Rate:	enter a value between 7 and 99 meas/second
Impulse Thresh (low):	enter a value between 30 and 102 dBmC
Spread between Thresholds:	enter a value between 2 and 8 db
(Low, Med, Hi)	

4. Type the desired value for the selected parameters.
5. Type **s** to start testing. Results will appear in the **Impulse Noise and Hits Results** menu on the right side of the screen.

Note: **Control-z** will reset the values during test.

3-1.2.3 Phase and Amplitude Jitter

Phase/Amplitude jitter is a dithering of the phase which appears as phase or frequency modulation. It is measured by sending a test tone of **1004 Hz**. At the receive end, a phase locked loop establishes a phase reference and jitter is measured relative to this reference. Phase jitter measurements are made over a **4 Hz to 300 Hz** range in three bands: **4 Hz to 20 Hz**, **20 Hz to 300 Hz** and the entire **4 Hz to 300 Hz** band. Jitter standards for data transmission are no more than **10°** between **20 Hz** and **300 Hz** and **15°** between **4 Hz** and **300 Hz**.

To measure **phase and amplitude jitter**:

- 1. Use the **Up/Down Arrow** keys to select **Jitter- phase and amplitude** from the **Impairment Test** menu.
- 2. Press **Enter**. The **Phase and Amplitude Jitter** menu appears. (See **Figure 3-10**)

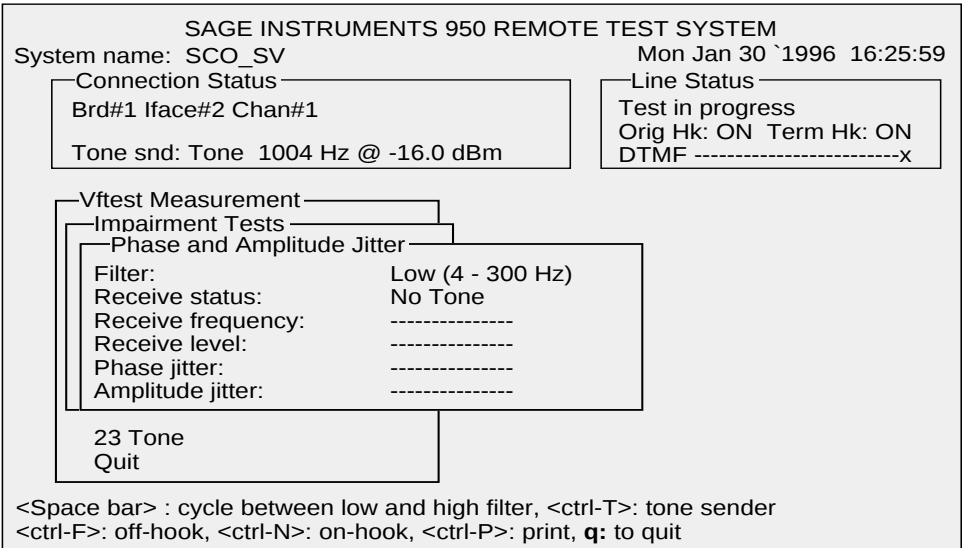


Figure 3-10
Phase and Amplitude Jitter Menu

Note: The **Phase and Amplitude Jitter** measurement results appear automatically when you call up this menu.

- 3. Press the **Space bar** to toggle between the low and high filters.
- 4. Type **q** to quit test.

3-1.2.4 Envelope Delay Distortion/Frequency Sweep

Envelope Delay Distortion results when different frequencies travelling along a circuit arrive at different times, even though the frequencies were transmitted together. This occurs when the rate of change of phase shift with frequency over the bandwidth of interest is not constant. EDD measurements are usually stated as one-half the difference between the delays of the two frequency extremes of the band of interest.

To measure **envelope delay distortion**:

1. Use the **Up/Down Arrow** keys to select **Envelope Delay Distortion** from the **Impairment Tests** menu.
2. Press **Enter**. The **Envelope Delay Distortion Setup** menu appears. (See **Figure 3-11**)

The screenshot displays the SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM interface. At the top, it shows the system name 'SCO_SV', the date and time 'Mon Jan 30 `1996 16:25:59', and the connection status 'Brd#1 Iface#2 Chan#1'. The tone is set to 'Tone snd: Tone 1004 Hz @ -16.0 dBm'. The line status is 'Test in progress' with 'Orig Hk: ON' and 'Term Hk: ON'. The DTMF status is '-----x'. The main menu is 'VfTest Measurement', with 'Impairment Tests' selected. The 'Envelope Delay Distortion Setup' menu is open, showing the 'Mode' as 'Measure'. The 'Send' is 'Sweep (measure forward)'. The 'Send level' is '16.0 dBm'. The 'Start freq.' is '304 Hz', 'Freq step' is '100 Hz', and 'End freq.' is '3204 Hz'. The 'Time/step' is '5.0 sec' and 'Skip 2600 Hz?' is 'No'. The 'Start EDD Test' option is visible. At the bottom, a help line states '<Space bar>: toggle between measure and repeat modes' and 's: start test, q: quit'.

Figure 3-11
Envelope Delay Distortion Setup Menu
(Sweep Measure forward.)

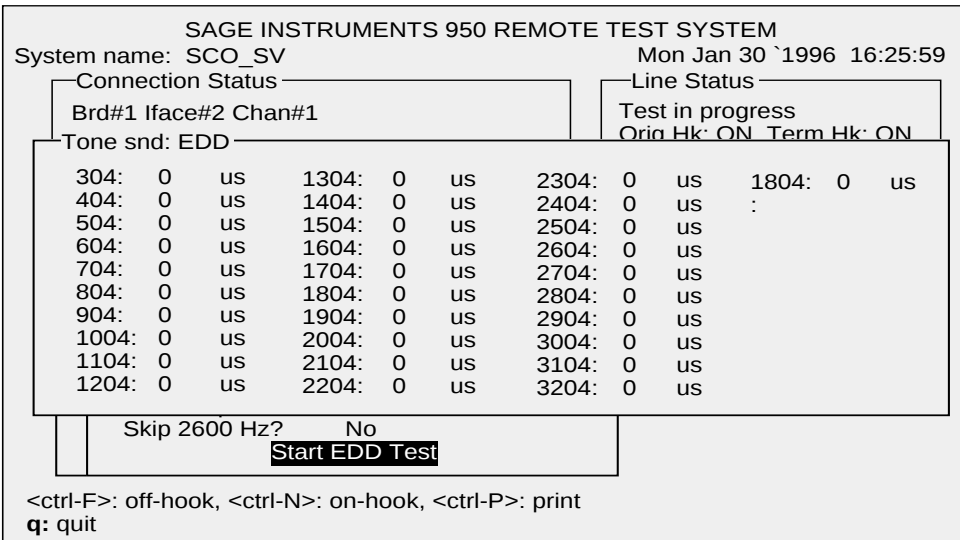
In **fixed (measure reverse)** mode, only the Mode, Send, Send Level, and Send Freq. fields are visible.

3. Use the **Up/Down Arrow** keys to scroll through the setup parameters. The help line at the bottom of the screen is different for each parameter and displays the acceptable values for that parameter.

The following is a list of the parameters:

Mode: use the **Space bar** to toggle between Measure and Repeat
Send: use the **Space bar** to toggle between **sweep** (meas forward) and **fixed** (measure reverse)
Send level: enter a value between **-60** to **+3 dBm**
Start frequency: enter a value between **20** to **3904 Hz**
Frequency step: enter a value between **10** to **1000 Hz**
End frequency: enter a value higher than the start frequency
Time/step: enter a value between **1** to **30 sec**
Skip 2600 Hz? use the **Space bar** to toggle between **Yes** and **No**

4. Type the desired value for the selected parameters **OR**, for **Mode** and **Send**, use the **Space bar** to toggle between the two selections.
5. Type **s** to start testing. Results appear in the **Envelope Delay Distortion** results box. (See **Figure 3-12**)



Demand
Test Options

Figure 3-12
Envelope Delay Distortion Results Menu

The **Envelope Delay Distortion** measurement results appear automatically.

6. Type **q** to quit the test and return to the setup menu.

3-1.2.5 Absolute Delay

Absolute delay is the time interval between transmission and reception of a signal.

To measure absolute delay:

1. Use the **Up/Down Arrow** keys to select **Absolute Delay** from the **Impairment Tests** menu.
2. Press **Enter**. The **Absolute Delay** menus appear. (See **Figure 3-13**)

The screenshot displays the SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM interface. At the top, it shows the system name 'SCO_SV' and the date/time 'Mon Jan 30 `1996 16:25:59'. Below this, there are two main sections: 'Connection Status' and 'Line Status'. The 'Connection Status' section shows 'Brd#1 Iface#2 Chan#1' and 'Tone snd: Tone 1004 Hz @ -16.0 dBm'. The 'Line Status' section shows 'Test in progress', 'Orig Hk: ON', 'Term Hk: ON', and 'DTMF -----x'. In the center, there is a menu structure with 'Vftest Measurement' at the top, followed by 'Impairment Tests', 'Absolute Delay Setup', and 'Absolute Delay Results'. The 'Absolute Delay Setup' menu is currently selected and shows 'Mode: Measure', 'Send level: 16.0 dBm', and 'Start Test'. Below this, there are options for 'Peak/avg ratio (PAR)', 'Quit Impairments Tests', '23 Tone', and 'Quit'. The 'Absolute Delay Results' menu on the right shows 'Status: OK', 'Delay: -211 ms', and 'Receive Level: -60.0 dBm'. At the bottom, there is a help line that reads '<Space bar>: toggles between forward/ repeat modes' and 's: start test, q: quit'.

Figure 3-13
Absolute Delay Menus

3. Use the **Up/Down Arrow** keys to scroll through the setup parameters. The help line at the bottom of the screen changes to display the acceptable values for the highlighted parameters.
4. Highlight the **Mode** field.
5. Press the **Space bar** to toggle between **Measure** and **Repeat**.
5. Highlight the **Send Level** field.
6. Enter a new **send level** between **+3** and **-40 dBm**
7. Type **s** to start testing. Results appear in the **Absolute Delay Results** menu on the right side of the screen.

3.1.2.6 Peak to Average Ratio (PAR)

Peak to Average Ratio (PAR) provides a measure of the channel dispersion (amplitude spreads over time) due to transmission impairments. By comparing P/AR readings over time, deterioration of a channel can be determined.

If a P/AR signal were received over a perfect channel entirely undistorted, the reading would be 100. A reading of 85 would indicate a 15 percent reduction in the P/AR. Acceptable P/AR ranges are:

<u>Connection Type</u>	<u>Min/Max P /AR Value</u>
End-to -end connections	50-100
Individual link, end link, or mid link	80-100
T1 carrier trunks	93-102
Non-repeated cable less than 18,000 feet long	97-100 (non-loaded), 94-100 (loaded)
Repeated cable less than 18,000 feet long	90-100 (non-loaded)
Repeated cable less than 36,000 feet long	90-100 (loaded)
Repeated cable more than 36,000 feet long	80-100 (loaded)

To measure **P/AR**:

1. Use the **Up/Down Arrow** keys to select **Peak/avg ratio (P/AR)** from the **Impairment Tests** menu.
2. Press **Enter**. The **Peak/Average Ratio (P/AR)** menus appear. (See **Figure 3-14**)

```

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM
System name: SCO_SV                               Mon Jan 30 `1996 16:25:59
Connection Status
Brd#1 Iface#2 Chan#1
Tone snd: Tone 1004 Hz @ -16.0 dBm

Line Status
Test in progress
Orig Hk: ON Term Hk: ON
DTMF -----x

Vftest Measurement
Impairment Tests
Peak/Average Ratio (P/AR)
Rcv status: OK
Rcv level: -60.0 dBm
Result: 80 P/AR units
Peak/avg ratio (PAR)
Quit Impairments Tests
23 Tone
Quit

<ctrl-F>: off-hook, <ctrl-N>: om-hook, <ctrl-P>: print
q: quit
    
```

Figure 3-14
Peak to Average Ratio (P/AR) Menus

The **P/AR** measurement results appear automatically upon entering the menu.

3. Type **q** to quit the test and return to the **Impairment Tests** menu.

3-1.3 23 Tones

23 Tones enables you to make a variety of transmission impairment measurements across the full channel spectrum in a short period of time.

The test signal consists of 23 equally spaced, phase coherent tones ranging from 203 Hz to 3228 Hz. The phase relationships of the tones create a signal that simulates the probability density of high speed modems such as V.29, V.32, and V.34. A single burst of 23 tones lasting only three seconds can yield multiple measurements:

- **Attenuation Distortion** at 23 frequencies
- **Envelope Delay Distortion (EDD)** at 22 frequencies
- **Signal to Total Distortion Ratio (S/TD)**
- Second and Third Order **Intermodulation Distortion (IMD)**

In addition, some of the measurements that correspond to conventional impairments tests have capabilities not available with traditional techniques:

- **Two wire EDD** is measured immediately without a repeater set or a return path.
- The **23-Tone S/TD** measurement uses a complex signal that stresses the channel much better than a single tone and intermodulation products are included in the reading.
- **ADPCM detection:** the complexity of the **23-Tone** signal causes channels that use signal compression to exhibit a characteristically high **S/TD** ratio.

To measure **23 tones**:

1. Use the **Up/Down Arrow** keys to select **23 Tones** from the **Vf test Measurement** menu.
2. Press **Enter**. The **23 Tone Results** menu appears. (See **Figure 3-15**)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM

System name: SCO_SV

Mon Jan 30 `1996 16:25:59

<23 Tones Results (level in dBm, delay in msec)>

Rcv sig pwr: -60.0 dBm

Sig/noise: 0.0 dB

Sig /tot dist: 0.0 dB

2nd order IMD: 99.9 dB

3rd order IMD: 99.9 dB

Freq	Level	Delay	Freq	Level	Delay	Freq	Level	Delay
283	-60.0		1375		-1	2547	-60.0	
281		6	1453	-60.0		2625		5
359	-60.0		1531		0	2703	-60.0	
438		4	1609	-60.0		2701		2
516	-60.0		1688		3	2059	-60.0	
594		3	1766	-60.0		2938		5
672	-60.0		1844		5	3016	-60.0	
750		1	1922	-60.0		3094		4
828	-60.0		2000		6	3172	-60.0	
986		1	2078	-60.0		3250		3
984	-60.0		2156		5	3320	-60.0	
1063		0	2234	-60.0		3406		2
1141	-60.0		2313		5	3484	-60.0	
1219		-1	2391	-60.0		3563		4
1297	-60.0		2469		5	3641	-60.0	

<ctrl-F> off-hook, <ctrl-N> on-hook, <ctrl-P> print

q: quit

Figure 3-15
23 Tones Menu

The **23 Tone** measurement results appear automatically upon entering the menu.

- 3. Type **q** to quit test and return to the **Vftest Measurement** menu.

SECTION IV

DSO BERT TESTING

4-1 INTRODUCTION

The **DSO BERT** feature of the **950RTS** allows you to collect several types of data transmission measurements: Bit Errors (**BitErr**), Bit Error Rate (**BER**), Synch Loss Seconds (**SLS**), Error Free Seconds (**EFS**), **%EFS**, Errored Seconds (**ES**), **%ES**, Severely Errored Seconds (**SES**), **%SES**, Degraded Seconds (**Deg**), **%Deg**, Failed Seconds (**Fail**), **%Fail**, Unavailable Seconds (**Unavail**), **%Unavail**, Available Seconds (**Avail**), and **%Avail**. These measurements enable the user to verify the digital integrity of one or several **DSOs** in a digital facility.

The **DSO BERT** function allows you to access the **DSO** via different types of network elements including **DCS**, direct **T-1** connections and switched connections. The user can configure the test pattern, duration, and data format for a test, and control the test via a menu interface.

When the test is started, the test results are displayed on the screen in real-time. The user can customize the result views to display all results or a subset of them, and capture the results in a user defined file for later reference. The **950RTS** allows the user to print and view the results file, including user defined custom views of the results file.

The **BERT** screens are menu-based and ordered hierarchically. The main **BERT** screen displays the **Test Controls** menu, in which you can set the parameters for a **BERT** test, and the **BERT Results** window, which displays test results when you begin testing.

This section of the manual will discuss the procedures for navigating, setting-up, and executing each of the **950RTS BERT** functions.

4-2 NAVIGATION

Navigation in the screen-oriented **BERT** function is the same as in the **vfctest** feature discussed in the beginning of **Section II** of this manual. You choose a menu or value by first highlighting the desired selection, then accepting it by pressing **Enter** or **Return**.

The **BERT** function also features pop-up menus which are activated by pressing the **Space bar** when a selection is highlighted in the main menu. In addition, items in the main display can be selected by typing the first letter in the item name.

4-2.1 Screen Information

The **950RTS** screen always gives some basic information about the **950RTS** and the facility under test. That information is located at the top of the screen.

Section IV DS0 BERT Testing

In the **BERT** function this information includes:

- **950RTS** system name
- Current date and time

In addition, the **950RTS** screen displays two status boxes. The **BERT** program updates them automatically during a test.

- Connection Status
- Line Status

At the bottom of the screen are two instruction lines which display help information for the displayed function.

4-3 GETTING STARTED

After logging into the system, activate the **BERT Feature** by entering the **berttest** command at the **UNIX** prompt:

```
$ berttest
```

The first menu to appear upon starting **berttest** is the **Test Type** menu. (See **Figure 4-1**)

```
SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM
System name: s950p3                      Wed Aug 21 1996 16:25:59
Connection Status                        Line Status
Test Type
DS0 BERT
DS1 BERT
FT1 BERT
Quit
Press <Space bar> to select DS0 level BERT testing
Press <Tab> for next field, <Esc> to cancel circuit select
```

Figure 4-1
Test Type Menu

To select **DS0** testing:

1. Use the **Arrow** keys to highlight the **DS0 BERT** field in the **Test Type** menu.
2. Press the **Space Bar**. The **DS0 Circuit Select** menu will appear. (See **Figure 4-2**)

The following Test Ports shown are examples only. The actual names may be different in your system. These are defined in the **Interface Database** file. Refer to the **950RTS Installation and Administration Manual** for a description of the **Interface Database**.

4-3.1 Selecting a Circuit in Direct Mode

Direct Mode enables you to perform testing of DS0 circuits through a direct T1 connection. (See **Figure 4-2**.)

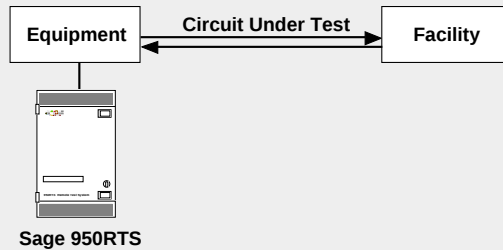


Figure 4-2
Direct Testing

To select a **test port**:

1. Use the **Arrow** keys to highlight the **Test Port** field in the **DS0 Circuit Select** menu. (See **Figure 4-3**)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM

System name: s950p3 Wed Aug 21 1996 16:25:59

Connection Status

Line Status

DS0 Circuit Select

Port Type: Robbed-bit DS0

Port: DirectT1

Quit Select

Enter name of test port, or <Space bar> to list. Press <Enter> to confirm.
Press <Tab> for next field, <Esc> to abort circuit selection

Figure 4-3
Selecting a Circuit (Direct Mode)

2. Press the **Space bar**. The **Test Port** menu appears. (See **Figure 4-4**)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM
System name: s950p3 Wed Aug 21 1996 16:25:59
Connection Status Line Status
DS0 Circuit Select
Port Type: Robbed-bit DS0
Port: Test Port
DirectT1
DCS TAD
Quit Select
Enter name of test port, or <Space bar> to list. Press <Enter> to confirm.
Press <Tab> for next field, <Esc> to abort circuit selection

Figure 4-4
Selecting a Test Port

3. Use the **Arrow** keys to highlight a test port to undergo testing.
4. Press **Enter** to accept the selection. You will advance to the **Vftest Measurement** menu.

4-3.2 Selecting a Circuit in DCS Mode

DCS mode enables you to perform DS0 and switched 56kbps testing on DS0 circuits through a DCS. (See **Figure 4-5**)

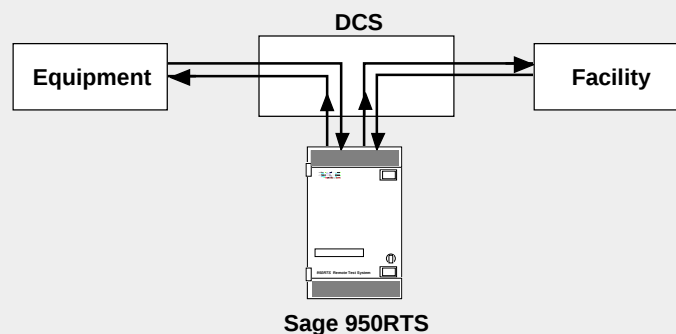


Figure 4-5
DCS Testing

To select a **circuit** if you are using **DCS** control:

1. Use the **Arrow** keys to highlight the **Test Port** field in the **DS0 Circuit Select** menu. (See **Figure 4-6**)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM

System name: s950p3 Wed Aug 21 1996 16:25:59

Connection Status

Line Status

DS0 Circuit Select

Port Type: DCS Clear TAP

DCS Type: DSC

Port: **DSC TAD**

Digroup: 1

Chan: 1

Quit Select

Enter name of test port, or <Space bar> to list. Press <Enter> to confirm.
Press <Tab> for next field, <Esc> to cancel circuit select

Figure 4-6
Selecting a Circuit (DCS)

Note: The **Digroup** and **Chan** fields above are **ID fields** for a **DSC** brand digital cross-connect. These fields will vary, depending on the type of **DCS** selected.

2. Press the **Space bar**. The **Test Port** menu appears. (See **Figure 4-7**)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM

System name: s950p3 Wed Aug 21 1996 16:25:59

Connection Status

Line Status

DS0 Circuit Select

Port Type: DCS Clear TAP

DCS Type: DSC

Port:

Test Port

DirectT1

DSC TAD

Digroup:

Chan:

Quit Select

Enter name of test port, or <Space bar> to list. Press <Enter> to confirm.
Press <Tab> for next field, <Esc> to cancel circuit select

Figure 4-7
Selecting a Test Port

3. Use the **Arrow** keys to highlight a **test port**.
4. The circuit ID fields under the **Port** field (i.e. **Digroup**, **Channel**) vary depending on the **DCS** type selected. You can change these by using the **Arrow** keys to highlight them, then enter a new value.
5. Use the **Arrow** keys to highlight the **Quit Select** field. Press **Enter** to accept the selection. You will advance to the **DS0 BERT Control** menu.

4-4 PREPARING TO PERFORM A DS0 BERT TEST

After selecting a circuit, you will advance to the **BERT Test Control** menu. (See **Figure 4-8**)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM

System name: SCO_SV

Mon Jan 30 1996 16:25:59

Connection Status

Direct T1

Sw 56 Kb 511

Line Status

Dead Line

Orig Hk: ON Term Hk: ON

DTMF: -----x

DS0 BERT Test Control

Start Test

Mode: Monitor

Direction: Equipment

Results View:

Results History

Test Setup

Data Format

Circuit Select

Dial

Quit

BERT Results

Status: Stopped Time: 0:01

BitEr: BER:

SLS: Data:

EFS: %EFS:

ES: %ES:

SES: %SES:

Deg: %Deg:

Fail: %Fail:

Avl: %Avl:

Uvl: %Uvl:

Press <Enter> to change circuit under test

<Tab>: next field, <ctrl-Z>: zero counts, <Ctrl-N>: on-hook, <ctrl-F>: off hook

Figure 4-8
Test Control Screen (DCS Mode)

The **Mode** and **Direction** fields do not appear in **Direct Mode**. When a **switched 56Kb Channel** has been selected, a **Dial** field appears below the **Circuit Select** field.

4-4.1 Status Boxes

The **950RTS** screen displays two status boxes in **berttest** that are not user-selectable. Either the **berttest** program updates them automatically during a test, or the user updates them indirectly by making changes to the information from within the **DS0 BERT Test Control** menu:

- Connection Status
- Line Status

The **connection status** box displays the type of test connection made. In **Direct Mode**, this includes the name of the circuit under test, the pattern type, framing, and coding type. (See **Figure 4-9**)

Connection Status

DirectT1

DS0A 56 Kb 511

Figure 4-9
Connection Status Box in Direct Mode

In **DCS mode**, the **connection status** box also displays the **Circuit ID information** and the channel states: **mon** (monitor) mode, **split** (test) mode, or **rel** (release) mode. (See **Figure 4-10**)

Connection Status

DCS TAD 1-1-1 Tap 1-16-1 rel eqp

DS0A 56 Kb 511

Figure 4-10
Connection Status Box in DCS Mode

In **switched 56Kb** testing, the **Line Status** box displays the line status as well as the on and off hook status of the originating and terminating equipment, and the outpulsing mode (DTMF or MF). (See **Figure 4-11**) On **non-switched** channels, the **Line Status** box shows whether the test is stopped and any network codes that have been received.

Line Status

Dead Line

Orig Hk: ON Term Hook: ON

DTMF -----x

Figure 4-11
Line Status Box

The following is a list of possible switched line status messages:

Message Displayed	Description
Unknown	Unknown signal
Dead Line	More than 4 seconds of silence
Dial tone (350/440)	Steady 350 Hz/440 Hz dual tone
Dial tone (400)	Steady 400 Hz tone
Busy	.5 seconds on/off 480 Hz/620 Hz dual tone
Reorder	.25 seconds on/off 480 Hz/620 Hz dual tone
Ringing	440 Hz/480 Hz dual tone; 2 seconds on, 4 seconds off
TPT	2225 Hz Test Progress Tone
Chime	MCI Chime Tone
Bong	AT&T Bong Tone
Beep Beep	2 bursts of 600 Hz Tone
Single tone	Steady single tone
Dual tone	Steady dual tone
Speech?	Suspected speech (unrecognized 0-, 3-, or 4-tone)
Listening	Line status not yet determined
Test in progress	Conducting test

4-4.2 DSO BERT Results Menu

Although the **950RTS** stores all test results, you can configure the **BERT** result screen to display a partial results listing by selecting one of the standard views, or by defining a new view.

To select the desired results set:

- 1. Use the **Arrow** keys to highlight the **Results View** field from the **Test Control** menu.
- 2. Press the **Space bar** to display a pop-up menu which lists the following standard views:

Default, Avail Time, Brief

- 3. Use the **Arrow** keys to highlight a pattern from the pop-up menu. The **BERT Results** menu will change, depending on the view selected.

If you selected **Default**, the **BERT Results** menu is the one appearing in **Figure 4-8**.

If you selected **Avail Time**, the **BERT Results** menu reads:

BERT Results		
Status: Stopped	Time:	0:01
Deg: 0	%Deg:	0.0
Fail: 0:00	%Fail:	0.0
Avl: 0	%Avl:	0.0
Uvl: 0:00	%Uvl:	0.0

Figure 4-12
Results Menu (Available Time)

If you selected **Brief**, the **BERT Results** menu reads:

BERT Results		
Status: Stopped	Time:	0:01
BitEr: _____	BER:	_____
SLS: _____	Data:	_____
EFS: _____	%EFS:	_____
ES: _____	%ES:	_____
SES:: _____	%SES:	_____

Figure 4-13
Results Menu (Brief)

4. Press **Enter** to accept the value and advance to the next field.

The system administrator can set up additional system-wide results views. You can create your own custom views by creating the file **.bertviews** in your home directory and following the format in the system-wide view file **/u/950rts/lib/bertviews**.

4-4.3 BERT Test Setup Menu

To change the **test setup**:

1. Use the **Arrow** keys or type **t** to highlight **Test Setup** in the **Bert Test Control** menu.
2. Press **Enter**. The **BERT Test Setup** menu appears. (See **Figure 4-14**)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM			
System name: SCO_SV		Mon Jan 30 1996 16:25:59	
Connection Status		Line Status	
Brd#1 Iface#2 Chan#1		Dead Line	
DS0A 56 Kb 511		Orig Hk: ON Term Hk: ON	
		DTMF -----X	
BERT Test Control		BERT Results	
BERT Test Setup			
Pattern: 511			
Result File: bertresults			
Duration: Continuous			
Quit Setup			
Circuit Select	Fail: _____	%Fail: _____	
Dial	Avl: _____	%Avl: _____	
Quit	Uvl: _____	%Uvl: _____	
Enter bit pattern (3-8 bits), or press <Space bar> to list standard patterns			
Press <Tab> for next field, <ESC> to cancel test setup			

Figure 4-14
Test Setup Screen

4-4.3.1 Choosing a Primary Bit Pattern

To choose a **pattern**:

1. Use the **Arrow** keys to highlight the **Pattern** field in the **BERT Test Setup** menu.
2. Press the **Space bar** to display a pop-up menu which lists the following standard patterns:

511, 2047, 2E15, 2E20, 2E23, DDS1, DDS2, DDS3, DDS4, DDS5, DDS3R, DDS5A, USER, 1:7

3. Use the **Arrow keys** to highlight a pattern from the pop-up menu.
4. Press **Enter** to accept the new value.

4-4.3.1.1 Choosing a User Pattern

To choose a **user pattern**:

1. Use the **Arrow** keys to highlight the **Pattern** field in the **BERT Test Setup** menu.
2. Press the **Space bar** to display the **Pattern** pop-up menu, which lists the following standard
3. Use the **Arrow keys** to highlight **USER** from the pop-up menu and press **Enter**. A **User Pattern** field will appear in the **BERT Test Setup** menu. (See **Figure 4-15**)
4. Use the **Arrow** keys to highlight the **User Pattern** field.

The screenshot displays the SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM interface. At the top, it shows the system name 'SCO_SV' and the date/time 'Mon Jan 30 1996 16:25:59'. Below this, there are sections for 'Connection Status' (Direct T1, DS0A 56 Kb 511) and 'Line Status' (Dead Line, Orig Hk: ON, Term Hk: ON, DTMF-----X). The main area is divided into 'DS0 BERT Test Control' and 'BERT Results'. The 'BERT Test Setup' menu is active, showing 'Pattern: User', 'User Pattern: 101011' (highlighted), 'Result File: bertresults', and 'Duration:'. A 'Quit Setup' button is at the bottom. Instructions at the bottom of the screen state: 'Enter bit pattern (3-8 bits), or press <Space bar> to list standard patterns. Press <Tab> for next field, <ESC> to cancel test setup'.

Figure 4-15
User Pattern

5. Use the **keyboard** to enter a user pattern (**0**'s and **1**'s). It should be between **3** and **8** bits in length.
6. Press **Enter** to accept the new value.

4-4.3.2 Choosing a Results File

To select a file in which to save your test results:

1. Use the **Arrow** keys to highlight the **Results File** field.
2. Use the **keypad** to enter "**none**" or a file name.
3. Press **Enter** to accept the new value or press **Tab** to accept the value and advance to the next field.

4-4.3.3 Selecting a Test Duration

To select a **test duration**:

1. Use the **Arrow** keys to highlight the **Duration** field.
2. Enter **c** for continuous testing, **d** to test for 24 hours, **h** to test for 1 hour, or **b** to test for 15 minutes.

(You can alternately use the keypad to enter a duration directly in the format **hrs:mins.**)

3. Press **Tab** to accept the value and advance to the next field.

After you have completed the **BERT** test setup, use the **Arrow Keys** to select **Quit Setup** and press **Enter** to exit to the **Test Control** menu.

4-4.4 Data Format

To change the **data format**:

1. Use the **Arrow** keys or type **f** to highlight the **Format** field from the **Test Control** screen.
2. Press the **Space bar** to display the **Data Format** menu. (See **Figure 4-16**).

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM			
System name: SCO_SV		Mon Jan 30 `1996 16:25:59	
Sender Status INTFC1 StandardPort.1 DS0A 56Kb 511		Line Status Listening...	
BERT Test Control Data Format Format: DS0A Subrate: 56 Kb Echo Cancel: OFF Quit Format		BERT Results Status: Stopped Time: 0:01 BitEr: 0 BER: <1.00E+00 EFS: 0 %EFS: 00 ES: 0:00 %ES: 100.0 SES 0:00 %SES: 100.0	
Data Format Circuit Select Quit			
Enter bit pattern (3-8 bits), or press <Space bar> to list standard patterns Press <Tab > for next field, <ESC> to cancel test setup			

Figure 4-16
Data Format Menu

If you selected a **switched 56Kb channel**, the **Format** and **Subrate** selection fields do not appear in the **Data Format** menu.

4-4.4.1 Selecting a Format

To select a **format type** (not available for **switched 56Kb** channels):

1. Use the **Arrow** keys to highlight the **Format** field.
2. Press the **Space bar** to display a pop-up menu which lists the following standard formats:

DS0A, 64 Kb

(You can alternately use the **keypad** to enter the first letter of a data format.)

3. Use the **Arrow** keys to highlight a format from the pop-up menu.
4. Press **Tab** to accept the value and advance to the next field.

4-4.4.2 Selecting a Subrate

To select a **subrate** (not available for **switched 56Kb** or **64Kb clear** channels):

1. Use the **Arrow** keys to highlight the **Subrate** field.
2. Press the **Space bar** to display a pop-up menu which lists the following standard subrates:

56 Kb, 19.2 Kb, 9.6 Kb, 4.8 Kb, 2.4 Kb

3. Use the **Arrow keys** to highlight a subrate from the pop-up menu and
4. Press **Enter** to accept the new value or press **Tab** to accept the value and advance to the next field.

4-4.4.3 Turning the Echo Canceller On/Off

To turn the **echo canceller on or off**:

1. Use the **Arrow** keys to highlight the **Echo Cancel** field.
2. Press the **Space bar** to toggle between **ON** and **OFF**.
3. Press **Tab** to accept the value and advance to the next field.

After you have completed the data format setup, use the **Arrow Keys** to select **Quit Format** and press **Enter** to exit to the **Test Control** menu.

4-4.5 Circuit Select

To select a circuit:

1. Use the **Arrow** keys or type **c** to highlight the **Circuit Select** field from the **Test Control** screen.
2. Press the **Space bar** to display the **Test Port list**.

If you are in **Direct Mode**, follow the circuit selection procedure described in **Section 4-3.1 Selecting a Circuit in Direct Mode**.

If you are in **DCS Mode**, Follow the circuit selection procedure described in **Section 4-3.2 Selecting a Circuit in DCS Mode**.

4-4.5.1 Mode (DCS only)

When a channel is selected in **DCS Mode**, you must place it in one of 3 states:

Monitor Mode

In this mode, you can check the status of the circuit to undergo testing without interrupting the circuit or transmitting test data. The circuit should be monitored in both directions (equipment or facility sides) to check for In Band Signalling and busy conditions. This must be done before starting an intrusive test. (See **Figure 4-17**)

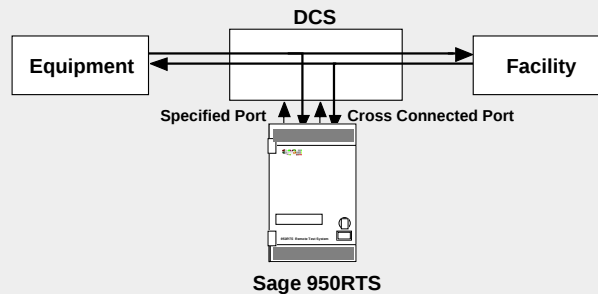


Figure 4-17
Monitor Mode

Split Mode

In this mode, you can perform intrusive tests on the designated channel. The channel under test is split and the **950RTS** tests either towards the equipment side or the facility side. (See **Figure 4-18**)

Important! You should only enter this mode after monitoring the channel, otherwise, you may disrupt activity on the line if it is busy.

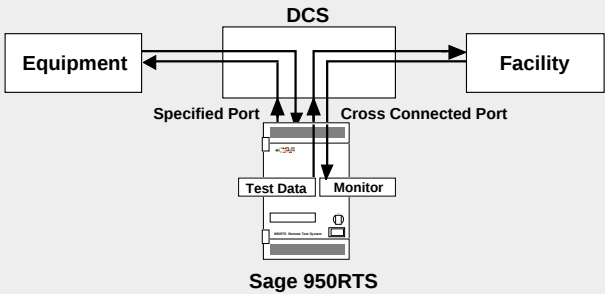


Figure 4-18
Split Mode

Release Mode You can manually enter this mode after testing to release the channel for use. (See **Figure 4-19**) Otherwise, the **950RTS** automatically releases the channel when you change circuits.

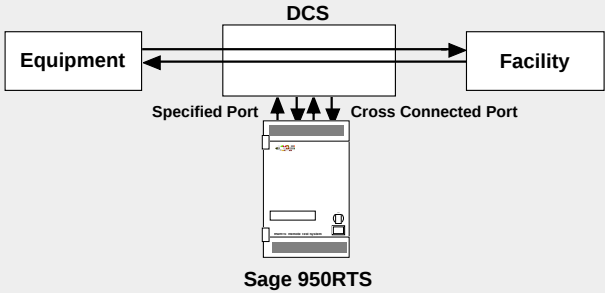


Figure 4-19
Release Mode

- To select a **mode**:
1. Use the **Up/Down Arrow keys** to select the **Mode** field from the **BERT Test Control** menu.
 2. Press the **Space bar**. The **DCS Circuit** menu appears. (See **Figure 4-20**)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM
System name: SCO_SV
Connection Status
dcs1 1-1-1 Tap 1-16-1 rel equip
DS0A 56 Kb 511

Mon Jan 30 `1996 16:25:59
Line Status
Dead Line
Orig Hk: ON Term Hk: ON
DTMF -----x

DS0 BERT Test Control
Start Test
Mode:
Direction:
Results View:
Results History
Test Setup
Format
Circuit Select
Quit

DCS Circuit
Release
Monitor
Split

BERT Results
Status: Stopped Time: 0:01
BitEr: BER:
SLS: Data:
EFS: %EFS:
ES: %ES:
SES: %SES:
Deg: %Deg:
Fail: %Fail:
Avl: %Avl:
Uvl: %Uvl:

Type 'm' to monitor, 's' to split, 'r' to release DCS circuit
<ctrl-T>: tone sender, <ctrl-F>: off-hook, <ctrl-N>: on-hook

Figure 4-20
DCS Circuit Menu

3. Use the **Up/Down Arrow** keys to select a **channel mode** and press **Enter**. You will be returned to the **BERT Test Control** menu.

4-4.5.2 Direction (DCS only)

The test access mechanism of the **DCS** allows the **950RTS** to non-intrusively monitor a DS0 cross connect, or intrusively split a cross connect. In both case, the **DCS** provides access to both directions of the cross connected circuit. The **950RTS** uses the terms **facility** and **equipment** to designate these two directions of test. The port that is used to identify the cross connect is designated the equipment direction, while the port on the other end of the **DCS** cross connect is designated the facility direction. In the switching environment, therefore, all DS0 circuits should be referred to by the port that connects to the switch so that the equipment direction of test will mean testing towards the switch, while the facility direction of test will refer to testing towards the transmission facilities.

To select a **direction**:

1. Use the **Up/Down Arrow** keys to select the **Direction** field from the **BERT Test Control** menu.
2. Press the **Space bar** to toggle between **equipment** and **facility**. Selecting **equipment** allows you to test towards the equipment side of the **DCS**, and selecting facility allows you to test towards the facility side.

4-5 DIAL (Switched 56 Kb Circuits Only)

The **Dial** function displays choices for **Origination Supervision**, **Format**, **Digits**, **Extended Dialing** and **Quit**.

- Use **Origin Supervision** to set the **950RTS** to an **ON HOOK** or an **OFF HOOK** status.
- Use the **Format** field to select the dial format.
- Use the **Digits** field to set the digits to be dialed.
- Use the **Extended Dialing** field to setup up to four digit sequences simultaneously, which can include both **MF** and **DTMF** phone numbers.
- Select **Quit** to return to the **BERT Test Control** menu.

To access the **Dial Menu**:

1. Use the **Arrow** keys to highlight the **Dial** field in the **BERT Test Control** menu.
2. Press **Enter**. The **Dial Menu** appears. (See **Figure 4-21**)

The screenshot displays the SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM interface. At the top, it shows the system name 'SCO_SV', connection status 'dcs1 1-1-1 Tap 1-16-1 rel equip Sw 56 Kb 511', and line status 'Dead Line Orig Hk: ON Term Hk: ON DTMF -----x'. The main menu is divided into 'DSO BERT Test Control' and 'BERT Results'. The 'DSO BERT Test Control' menu is currently selected, showing options: 'Dial', 'Res', 'Format', 'Circuit Select', and 'Quit'. The 'Dial' option is highlighted, and a sub-menu is displayed with 'Orig Supervision: ON HOOK', 'Format: DTMF', 'Digits: -----', 'Extended Dialing', and 'Quit'. The 'BERT Results' section shows a timer at '0:01' and various status fields: 'Deg: -----', 'Fail: -----', 'Avl: -----', 'Uvl: -----', '%Fail: -----', '%Avl: -----', and '%Uvl: -----'. At the bottom, instructions state: 'Enter digits: <ctrl-C>; clear, <Enter>=accept Valid digits 0-9, A-D, *, # q: quit'.

Figure 4-21
Dial Menu

4-5.1 Origin Supervision

The **Origin Supervision** function controls the **ON/OFF** status of the sender. To change the **Origin Supervision** status:

1. Use the **Arrow** keys to select the **Origin Supervision** field. (See **Figure 4-21**)
2. Use the **Space bar** to toggle between **ON HOOK** and **OFF HOOK**.

4-5.2 Format

To select a **dial format**:

1. Use the **Arrow** keys to scroll down to the **Format** field and press **Enter** to select it.
2. Press the **Space bar** to toggle between **DTMF** and **MF**.

4-5.3 Digits

To enter the digits to be dialed in the **Enter Digits** field:

1. Use the **Arrow** keys to highlight the **Enter Digits** field.
2. Enter up to **18 digits** to be dialed by the **950RTS**.
3. Press **Enter** to accept the number. If not off-hook, the **950RTS** will go off-hook automatically, then send the digits.

The arguments to follow the **Enter Digits** field must be appropriate to the signaling method selected in the **Format** field. The dialing format appears in the **Dial** menu. The appropriate arguments appear on a comment line at the bottom of the screen.

4-5.4 Extended Dialing

To send up to four digit sequences simultaneously:

1. Use the **Arrow** keys to highlight the **Extended Dialing** field.
2. Press **Enter**. The **Extended Dialing** menu appears. (See **Figure 4-22**)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM
 System name: SCO_SV Mon Jan 30 1996 16:25:59
 Connection Status Line Status
 DSC TAD 1-1-1 Tap 1-16-1 rel equip
 Dead Line
 Orig Hk: ON Term Hk: ON
 Tone snd: OFF DTMF -----x

Vf test Measurement
 Extended Dialing

Seq	Pre-Dial	Format	Digits
1	Immediate	DTMF	
2	off	DTMF	
3	off	DTMF	
4	off	DTMF	

Send Digit Sequence
 Return to Basic Dialing
 Quit

Select digit format: MF or DTMF
 <space> for selection menu, then <enter> to select

Figure 4-22
Extended Dialing Menu

4-5.4.1 Digit Sequences

To enter a **digit sequence**:

1. Use the **Arrow keys** to highlight the appropriate line in the **Digits** field.
2. Use the **number keys** to enter a number to be dialed.

4-5.4.2 Dial Format

To change the **dial format**:

1. Use the **Arrow keys** to highlight the appropriate line in the **Format** field.
2. Press the **Space bar**. The **Digit Format** menu appears. (See **Figure 4-23**)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM

System name: SCO_SV

Mon Jan 30 `1996 16:25:59

Connection Status

dcsl 1-1-1 Tap 1-16-1 rel equip

Sw 56 Kb 511

Line Status

Dead Line

Orig Hk: ON Term Hk: ON

DTMF -----x

DS0 BERT Test Control

BERT Results

Extended Dialing

0:01

Seq

Pre-Dial

Form

Digit Format

1

Immediate

DT

MF

2

off

DT

DTMF

3

off

DT

4

off

DTMF

Send Digit Sequence

Return to Basic Dialing

Quit

Select digit format: MF or DTMF

<space> for selection menu, then <enter> to select

Figure 4-23
Digit Format Menu

3. Use the **Arrow keys** to highlight either the **DTMF** or **MF** format.
4. Press **ENT**. You will be returned to the **Extended Dialing** menu.

4-5.4.3 Pre-Dial

The **pre dial** feature allows you to add signaling events, including winks and pauses, to the beginning of your dial sequence. Adding a wink tells the **950RTS** to wait for a wink before sending the digit sequence. Adding a pause injects a two second delay before the **950RTS** sends the digits.

To add a **pre-dial signaling event**:

1. Use the **Arrow keys** to highlight the appropriate line in the **Pre-Dial** field.
2. Press the **Space bar**. The **Pre-Dial** menu appears. (See **Figure 4-24**)

The screenshot displays the SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM interface. At the top, it shows the system name 'SCO_SV', the date and time 'Mon Jan 30 1996 16:25:59', and connection status 'dcs1 1-1-1 Tap 1-16-1 rel equip Sw 56 Kb 511'. Below this, there are two main sections: 'DSO BERT Test Control' and 'BERT Results'. The 'DSO BERT Test Control' section is further divided into 'Extended Dialing' and 'Pre-Dial'. The 'Pre-Dial' section contains a table with columns 'Seq', 'Pre-Dial', 'Format', and 'Digits'. The table has four rows: 1 (Im), 2 (Pre-Dial), 3 (Immediate), and 4 (Pause). The 'Pause' option is highlighted. Below the table, there are options to 'Send Digit Sequence', 'Return to Basic Dial Menu', and 'Quit'. The 'BERT Results' section shows a timer at '0:01' and a status 'Dead Line'. At the bottom, there is a instruction: 'Select action: wait for Wink, 2 second Pause, send Immediately, or sequence Off <space> for selection menu, then <enter> to select'.

Figure 4-24
Pre Dial Menu

- Use the **Arrow keys** to highlight **immediate**, **pause**, or **wink**. **Immediate** will send the sequence immediately, **pause** will tell the **950RTS** to add a 2 second delay before sending, and **wink** will tell the **950RTS** to wait for a wink from the terminating end before sending.
- Press **ENT**. You will be returned to the **Extended Dialing** menu.

4-5.4.4 Send Digit Sequences

To send the digit sequences:

- Use the **Arrow keys** to highlight the **Send Digit Sequence** field.
- Press the **Enter**. The digit sequences entered will be sent.

4-5.4.5 Quit

To quit the **Extended Dialing** menu:

- Use the **Arrow keys** to highlight the **Quit** field to return to the **BERT Test Control** menu, OR highlight the **Return to Basic Dialing** menu field to return to the **Dial** menu.
- Press **ENT** to activate your selection.

4-6 PERFORMING A DS0 BERT TEST

To perform a **BERT** test:

- 1. Use the **Arrow** keys or type **s** to select **Start Test** from the **BERT Test Control** menu. The **950RTS** begins testing and the **BERT Test Control** menu changes. (See **Figure 4-25**)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM

System name: SCO_SV

Mon Jan 30 `1996 16:25:59

Sender Status

INTFC1 StandardPort.1

DS0A 56Kb 511

Line Status

Listening...

BERT Test Control

Stop Test

Results View: Default

Results History

Error Inject

Loopback

Net Codes

Quit

BERT Results

Status: StoppedTime: 0:01

BitEr: 0BER: <1.00E+00

EFS: 0%EFS: 00

ES: 0:00%ES: 100.0

SES 0:00%SES: 100.0

Press <Enter> to start test

Press <Tab > for next field, <ctrl-Z> to xero counts, <ESC> to quit BERT testing

Figure 4-25
BERT Test Control Menu (During Testing)

The **Net Code** selection field only appears in the **BERT Test Control** menu during testing if you are using a DS0A data format.

- 2. Use the **Arrow** keys to select **Stop Test** or type **s** to stop a test.

4-6.1 Error Inject

To **inject an error** while testing:

- 1. Use the **Arrow** keys or type **e** to highlight the **Error Inject** field and press **Enter** to display the **Error Injection** menu. (See **Figure 4-26**).

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM

System name: SCO_SV

Mon Jan 30 `1996 16:25:59

Sender Status
INTFC1 StandardPort.1
DS0A 56Kb 511

Line Status
Listening...

BERT Test Control

Error Injection

Inject
Type: Single
Quit Inject Ctrl
Net Codes
Quit

BERT Results
Status: Stopped Time: 0:01
BitEr: 0 BER: <1.00E+00
EFS: 0 %EFS: 00
ES: 0:00 %ES: 100.0
SES 0:00 %SES: 100.0

Press <Enter> to inject errors
Press <Tab > for next field, <Esc> to quit inject control

Figure 4-26
Error Injection

2. Use the **Arrow** keys to highlight the **Type** field.
3. Press the **Space bar** to display a pop-up menu which lists the following standard error types:

Single, 6-Bit, 2:5, 3:5

4. Use the **Arrow** keys to highlight an error type from the pop-up menu.
5. Press **Enter** to accept the value and return to the **Error Injection** menu.
6. Use the **Arrow** keys to highlight **Inject**.
7. Press **Enter** to inject an error.
8. Use the **Arrow** keys to highlight **Quit Inject Ctrl** and exit to the **BERT Test Control** menu.

4-6.2 Loopback/Loopup

If you are testing a **switched 56 Kb circuit**, the following sections on setting up a loopback type and code do not apply. You should skip to **Section 4-6.1.4**.

To begin a loopback/loopup when testing on a **non-switched circuit**:

Use the **Arrow** keys or type **L** to highlight the **Loopback** field and press **Enter** to display the **Loopback control** menu. (See **Figure 4-27**)

If you are testing a switched 56Kb circuit, the Type, Code and Repeater Fields do not appear. The label Start **Loopback** field is replaced by **V.54 Loopup** and an additional field labeled **Loopdown** is added

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM

System name: SCO_SV Mon Jan 30 `1996 16:25:59

Sender Status

INTFC1 StandardPort.1

DS0A 56Kb 511

Line Status

Listening...

BERT Test Control

Loopback Control

Start Loopback

Type: Alternating

Code: CSU Alt

Repeater: 1

Configure MJU

QuitLoop Ctrl

Net Codes

Quit

BERT Results

Status: Stopped	Time: 0:01
BitEr: 0	BER: <1.00E+00
EFS: 0	%EFS: 00
ES: 0:00	%ES: 100.0
SES: 0:00	%SES: 100.0

Press <Enter> to start loopback
Press <Tab > for next field, <Esc> to quit loopback control

Figure 4-27
Loopback Control Menu

4-6.2.1 Selecting a Loopback/Loopup Type

To select a loopback type when testing on a **non-switched circuit**:

1. Use the **Arrow** keys to highlight the **Type** field.
2. Press the **Space bar** to toggle between **Alternating** and **Latched**. If you choose **Latched**, the **Start Loopback** field changes to read: **Start Loopup**.
3. Press **Tab** to accept the value and advance to the next field.

4-6.2.2 Selecting a Loopback Code

To select a loopback/loopup code when testing on a **non-switched circuit**:

1. Use the **Arrow** keys to highlight the **Code** field.
2. Press the **Space bar** to display a pop-up menu which lists standard loopback/loopup codes.

If you selected an **Alternating** loopback type, the available codes are:

CSU Alt, DSU Alt, HL96, OCU Alt, OCU N96, 56Kb #1, 56Kb #2

If you selected a **Latched** loopup type, the available codes are:

CSU Lat, DS0DP, HL222, OCU Lat

3. Use the **Arrow** keys to highlight a format from the pop-up menu.

- Press **Enter** to accept the value and return to the **Loopback Control** menu.

4-6.2.3 Selecting a Repeater

The repeater feature is only available only if **CSU Alt** was selected as a loopback type:

To select a repeater when testing on a **non-switched circuit**:

- Use the **Arrow** keys to highlight the **repeater** field.
- Use the **keyboard** to enter the number of the unit to loop.
- Press **Tab** to accept the value and advance to the **Quit Loop Control** field.

4-6.2.4 Performing a Loopback/Loopup

To perform a loopback/loopup when testing on a **non-switched circuit**:

- Use the **Arrow** keys to highlight **Start Loopback** (if an **Alternating** type was selected) or **Start Loopup** (if a **Latched** type was selected). The **950RTS** will perform the loopback/loopup.
- After you have completed the loopback/loopup, use the **Arrow Keys** to select **Quit Loop Ctrl** and press **Enter** to exit to the **BERT Test Control** menu.

To perform a loopback/loopup when testing on a **switched 56Kb circuit**:

- Use the **Arrow** keys or type 'l' to highlight the **Loopback** field from the Bert Test Control menu and press **Enter** to display the **Loopback control** menu. (See **Figure 4-28**)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM	
System name: SCO_SV	Mon Jan 30 `1996 16:25:59
Connection Status	Line Status
dcsl 1-1-1 Tap 1-16-1 rel equip	Dead Line
DSOA 56 Kb 511	Orig Hk: ON Term Hk: ON
	DTMF -----x
DSO BERT Test Control	BERT Results
Loopback Control	Status: Stopped Time: 0:01
V.54 Loopup	BitEr: BER:
	SLS: Data:
	EFS: %EFS:
	ES: %ES:
	SES: %SES:
	Deg: %Deg:
	Fail: %Fail:
	Avl: %Avg:
	Uvl: %Uvl:
Loopdown	
QuitLoop Ctrl	
Loopback	
Quit	
Press <Enter> to send loop up code	
Press <Tab> for next field, <Esc> to quit loopback control	

Figure 4-28
Loopback on a Switched 56 Kb Circuit

2. Use the **Arrow** keys to highlight **V.54 Loopup** to send a loopup code , (or **Latchup** if a **Latched** type was selected). The **950RTS** will perform the loopup.
3. Use the **Arrow** keys to highlight **loopdown** to send a loopdown code.
4. After you have completed the loopup/loopdown use the **Arrow Keys** to select **Quit Loop Ctrl** and press **Enter** to exit to the **BERT Test Control** menu.

4-6.3 Network Codes

Network Codes are only applicable if you selected a **DS0A** format.

To send network codes when testing on a non-switched circuit:

Use the **Arrow** keys to highlight the **Net Codes** field and press **Enter** to display the **Network Codes** menu. (See **Figure 4-29**)

DS0 BERT
Testing

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM

System name: SCO_SV

Sender Status

INTFC1 StandardPort.1
DS0A 56Kb 511

Mon Jan 30 `1996 16:25:59

Line Status

Listening...

BERT Test Control

Network Codes

Start Sending

Code: CMI

Quit Net Codes

Results History

Loopback

Net Codes

Quit

BERT Results

Status: StoppedTime: 0:01

BitEr: 0BER: <1.00E+00

EFS: 0%EFS: 00

ES: 0:00%ES: 100.0

SES 0:00%SES: 100.0

Enter first letter of network code, or <Space bar> to list
Press <Tab > for next field, <Esc> to quit Inet code control

Figure 4-29
Network Codes Menu

4-6.3.1 Selecting a Network Code

To select a network code type when testing on a non-switched circuit:

1. Use the **Arrow** keys to highlight the **Code** field.
2. Press the **Space bar** to display a pop-up menu which lists the following standard network codes:

CMI, Test, OOS, UMC, ASC

(You can alternately use the **keypad** to enter the first letter of a code)

3. Use the **Arrow** keys to highlight a code from the pop-up menu.
4. Press **Enter** to accept the value and return to the **Network Codes** menu.

4-6.3.2 Sending a Network Code

To send a network code when testing on a non-switched circuit:

Use the **Arrow** keys to highlight **Start Sending**. The **950RTS** will send the code.

After you have sent the network code, use the **Arrow** Keys to select **Quit Net Codes** and press **Enter** to exit to the **BERT Test Control** menu.

4-7 VIEWING RESULTS HISTORY

To view the results history after conducting a **BERT** test:

1. Use the **Arrow** keys to highlight **Result History** from the **BERT Test Control** menu.

The **Result History** menu will appear differently, depending on whether you chose, **Default**, **AvailTime**, or **Full** under **Results View**.

2. Press **Enter** to advance through the results screens, or type **q** to exit to the **BERT Test Control** display.

SECTION V

DS1 BERT TESTING

5-1 INTRODUCTION

The **DS1 BERT** function of the **950RTS** allows you to collect several types of data transmission measurements: Bipolar Violations (**BPV**), BPV Rate (**BPVrt**), Cyclic Redundancy Check Errors (**CRC**), CRC Error Rate (**CRCrt**), Frame Errors (**FrErr**), Frame Slips (**FrSlp**), Out-of-Frame-Count (**00F**), Bit Slips (**BtSlp**) and all the logic errors found in the **DS0 BERT** function.

The **DS1 BERT** function allows you to access the **DS1** via different types of network elements including direct **T-1** and **DCS FAD** connections. The user can configure the test pattern, duration, and data format for a test, and control the test via a menu interface.

When the test is started, the test results are displayed on the screen in real-time. The user can customize the result views to display all results or a subset of them, and capture the results in a user defined file for later reference. The **950RTS** allows the user to print and view the results file, including user defined custom views of the results file.

The **DS1 BERT** screens are menu-based and ordered hierarchically. The main **DS1 BERT** screen displays the **Test Controls** menu, in which you can set the parameters for a **DS1 BERT** test, and the **BERT Results** window, which displays test results when you begin testing.

This section of the manual will discuss the procedures for navigating, setting-up, and executing each of the **DS1 BERT** functions.

5-2 NAVIGATION

Navigation in the screen-oriented **DS1 BERT** function is the same as in the **vfctest** feature discussed in the beginning of **Section II** of this manual. You choose a menu or value by first highlighting the desired selection, then accepting it by pressing **Enter** or **Return**.

The **BERT** function also features pop-up menus which are activated by pressing the **Space bar** when a selection is highlighted in the main menu. In addition, items in the main display can be selected by typing the first letter in the item name.

5-2.1 Screen Information

The **950RTS** screen always gives some basic information about the **950RTS** and the facility under test. That information is located at the top of the screen.

Section V DS1 BERT Testing

In the **BERT** function this information includes:

- **950RTS** system name
- Current date and time

In addition, the **950RTS** screen displays two status boxes. The **BERT** program updates them automatically during a test.

- Connection Status
- Line Status

At the bottom of the screen are two instruction lines which display help information for the displayed function.

5-3 GETTING STARTED

After logging into the system, activate the **BERT Feature** by entering the **berttest** command at the **UNIX** prompt:

```
$ berttest
```

The first menu to appear upon starting **berttest** is the **Test Type** menu. (See **Figure 5-1**)

```
SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM
System name: s950p3                      Wed Aug 21 1996 16:25:59
Connection Status: [ ]                  Line Status: [ ]
Test Type:
  DS0 BERT
  DS1 BERT
  FT1 BERT
  Quit
Press <Space bar> to select DS1 level BERT testing
Press <Tab> for next field, <Esc> to cancel circuit select
```

Figure 5-1
Test Type Menu

To select **DS1** testing:

1. Use the **Arrow** keys to highlight **DS1 BERT** in the **Test Type** menu.
2. Press the **Space Bar**. The **DS1 Circuit Select** menu will appear. (See **Figure 5-2**)

The following test ports shown are examples only. The actual names may be different in your system. These are defined in the **Interface Database** file. Refer to the *950RTS Installation and Administration Manual* for a description of the **Interface Database**.

5-3.1 Selecting a Circuit in Direct Mode

Direct Mode enables you to perform testing of DS1 circuits through a direct T1 connection. (See **Figure 5-2**.)

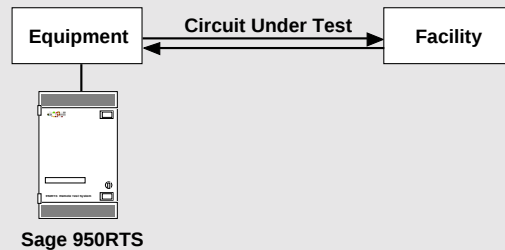


Figure 5-2
Direct Testing

To select a **test port**:

1. Use the **Arrow** keys to highlight the **Port** field in the **DS1 Circuit Select** menu. (See **Figure 5-3**)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM

System name: s950p3 Wed Aug 21 1996 16:25:59

Connection Status

Line Status

DS1 Circuit Select

Port Type: Full T1

Port: DirectT1

Quit Select

Enter name of test port, or <Space bar> to list. Press <Enter> to confirm.
Press <Tab> for next field, <Esc> to abort circuit selection

Figure 5-3
Selecting a Circuit (Direct Mode)

2. Press the **Space bar**. The **Test Port** menu appears. (See **Figure 5-4**)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM
System name: s950p3 Wed Aug 21 1996 16:25:59
Connection Status Line Status
DS1 Circuit Select
Port Type: Full T1
Port: Test Port
DirectT1
DCS FAD
Quit Select
Enter name of test port, or <Space bar> to list. Press <Enter> to confirm.
Press <Tab> for next field, <Esc> to cancel circuit select

Figure 5-4
Selecting a Test Port

3. Use the **Arrow** keys to highlight a test port to undergo testing.
4. Press **Enter** to accept the selection. You will advance to the **Vftest Measurement** menu.

5-3.2 Selecting a Circuit in DCS Mode

DCS mode enables you to test **DS1** circuits through a **DCS**. (See **Figure 5-5**)

Figure 5-5
DCS Testing

To select a **circuit** if you are using **DCS** control:

1. Use the **Arrow** keys to highlight the **Port** field in the **DS1 Circuit Select** menu.
(See **Figure 5-6**)

**Figure 5-6
Selecting a Circuit (DCS)**

Note: The **Digroup** and **Chan** fields above are **ID fields** for a **DSC** brand **DCS**. These fields will vary, depending on the type of **DCS** selected.

2. Press the **Space bar**. The **Test Port** menu appears. (See **Figure 5-7**)

**Figure 5-7
Selecting a Test Port**

3. Use the **Arrow** keys to highlight a **test port**.
4. The circuit ID fields under the **Port** field (i.e. **Digroup**, **Channel**) vary depending on the **DCS** type selected. You can change these by using the **Arrow** keys to highlight them, then enter a new value.
5. Use the **Arrow** keys to highlight the **Quit Select** field. Press **Enter** to accept the selection. You will advance to the **DS1 BERT Control** menu.

5-4 PREPARING TO PERFORM A DS1 BERT TEST

After selecting a circuit, you will advance to the **DS1 BERT Test Control** menu. (See **Figure 5-8**)

Figure 5-8
DS1 BERT Test Control Screen (DCS Mode)

Note: The **Mode** and **Direction** fields do not appear in **Direct Mode**.

5-4.1 Status Boxes

The **950RTS** screen displays two status boxes in **berttest** that are not user-selectable. Either the **berttest** program updates them automatically during a test, or the user updates them indirectly by making changes to the information from within the **DS1 BERT Test Control** menu:

- Connection Status
- Line Status

The **connection status** box displays the type of test connection made. In **Direct Mode**, this includes the name of the circuit under test, the pattern type, framing, and coding type. (See **Figure 5-9**)

Figure 5-9
Connection Status Box (Direct)

In **DCS mode**, the **connection status box** also displays the **Circuit ID information** and the channel states: **mon** (monitor) mode, **split** (test) mode, or **rel** (release) mode. (See **Figure 5-10**)

Figure 5-10
Connection Status Box (DCS)

The **Line Status box** in both **Direct** and **DCS modes** displays any T1 errors detected. (See **Figure 5-11**)

Figure 5-11
Line Status Box

5-4.2 DS1 BERT Results Menu

Although the **950RTS** stores all test results, you can configure the **BERT result** screen to display a partial results listing by selecting one of the standard views, or by defining a new view.

To select the desired results set:

1. Use the **Arrow keys** to highlight the **Results View** field from the **Test Control menu**.
2. Press the **Space bar** to display a pop-up menu which lists the predefined and user-defined views obtained from the **.bertviews** file.
3. Press **Enter** to accept the value and advance to the next field.

5-4.3 Test Setup

To change the test setup:

1. Use the **Arrow keys** or type **t** to highlight **Test Setup** in the **Bert Test Control menu**.
2. Press **Enter**. The **BERT Test Setup** menu appears. (See **Figure 5-12**)

**Figure 5-12
Test Setup Screen**

5-4.3.1 Choosing a Primary Bit Pattern

To choose a pattern:

1. Use the **Arrow** keys to highlight the **Pattern** field in the **BERT Test Setup** menu.
2. Press the **Space bar** to display a pop-up menu which lists the following standard patterns:

511, 2047, 2E15, 2E20, 2E23, QRSS, USER, 1:1, 1:7, 3in24, Ones, Hex55, Long54, 55Daly, Long72, Long96

(You can alternately use the keypad to enter a user pattern directly (three to eight zeroes or ones).

3. Use the **Arrow** keys to highlight a pattern from the pop-up menu.

5-4.3.1.1 Choosing a User Pattern

To choose a user pattern:

1. Use the **Arrow** keys to highlight the **Pattern** field in the **BERT Test Setup** menu.
2. Press the **Space bar** to display the **Pattern** pop-up menu. which lists the following standard
3. Use the **Arrow** keys to highlight **USER** from the pop-up menu and press **Enter**. A **User Pattern** field will appear in the **BERT Test Setup** menu. (See **Figure 5-13**)
4. Use the **Arrow** keys to highlight the **User Pattern** field.

**Figure 5-13
User Pattern**

5. Use the **keyboard** to enter a user pattern (0's and 1's). It should be between 3 and 8 bits in length.
6. Press **Enter** to accept the new value.

5-4.3.2 Choosing a Results File

To select a file in which to save your test results:

1. Use the **Arrow** keys to highlight the **Results File** field.
2. Use the **keypad** to enter "none" or a file name.
3. Press **Enter** to accept the new value or press **Tab** to accept the value and advance to the next field.

5-4.3.3 Selecting a Test Duration

To select a test duration:

1. Use the **Arrow** keys to highlight the **Duration** field.
2. Enter **c** for continuous testing, **d** to test for 24 hours, **h** to test for 1 hour, or **b** to test for 15 minutes.

(You can alternately use the keypad to enter a duration directly in the format **hrs:mins.**)

3. Press **Tab** to accept the value and advance to the next field.

Note: After you have completed the BERT test setup, use the **Arrow Keys** to select **Quit Setup** and press **Enter** to exit to the **Test Control** menu.

5-4.4 T1 Interface Configuration Menu

To change the T1 **Interface Configuration**:

1. Use the **Arrow** keys or type **f** to highlight the **Format** field from the **Test Control** screen.
2. Press the **Space bar** to display the **T1 Interface Configuration** menu. (See **Figure 5-14**).

**Figure 5-14
Line Format Menu**

5-4.4.1 Framing

To select a **framing type**:

1. Use the **Arrow** keys to highlight the **Framing** field.
2. Press the **Space bar** to display a pop-up menu which lists the following standard formats:

None, D4-SF, ESF

(You can alternately use the **keypad** to enter the first letter of a line format.)

3. Use the **Arrow** keys to highlight a format from the pop-up menu.
4. Press **Tab** to accept the value and advance to the next field.

5-4.4.2 Coding Type

To select a coding type:

1. Use the **Arrow** keys to highlight the **Coding** field.
2. Press the **Space bar** to display a pop-up menu which lists the following standard coding types:

AMI, B8ZS, B7Stuff

3. Use the **Arrow keys** to highlight a substrate from the pop-up menu.
4. Press **Enter** to accept the new value.

5-4.4.3 Performance Monitor

To select a performance monitor:

1. Use the **Arrow** keys to highlight the **Perf Mon** field.
2. Press the **Space bar** to display a pop-up menu which lists the following standard monitor types:

None, AT&T, ANSI

3. Press **Enter** to accept the value.

5-4.4.4 Length

To select a line build out:

1. Use the **Arrow** keys to highlight the **Length** field.
2. Press the **Space bar** to list the following lengths

0-133ft, 133-266ft, 266-399ft, 399-533ft, 533-655ft

3. Press **Enter** to accept the value.

5-4.5 CIRcuit Select

If you are in **Direct Mode**, follow the circuit selection procedure described in **Section 5-3.1 Selecting a Circuit in Direct Mode**.

If you are in **DCS Mode**, Follow the circuit selection procedure described in **Section 5-3.2 Selecting a Circuit in DCS Mode**.

Section V DS1 BERT Testing

After you have completed the data format setup, use the **Arrow Keys** to select **Quit Format** and press **Enter** to exit to the **Test Control** menu.

5-4.5.1 Mode (DCS only)

When a channel is selected in **DCS Mode**, you must place it in one of 3 states:

Monitor Mode In this mode, you can check the status of the circuit to undergo testing without interrupting the circuit or transmitting test data. The circuit should be monitored in both directions (equipment or facility sides) to check for In Band Signalling and busy conditions. This must be done before starting an intrusive test. (Refer to **Figure 5-15**)

Figure 5-15
Monitor Mode

Split Mode In this mode, you can perform intrusive tests on the designated channel. The channel under test is split and the **950RTS** tests either towards the equipment side or the facility side. (Refer to **Figure 5-16**)

Important! You should only enter this mode after monitoring the channel, otherwise, you may disrupt activity on the line if it is busy.

Figure 5-16
Split Mode

Release Mode You can manually enter this mode after testing to release the channel for use. (refer to **Figure 5-17**) Otherwise, the **950RTS** automatically releases the channel when you change circuits.



**Figure 5-17
Split Mode**

To select a mode:

1. Use the **Up/Down Arrow keys** to select the **Mode** field from the **BERT Test Control** menu.
2. Press the **Space bar**. The **DCS Circuit** menu appears. (See **Figure 5-18**)

**Figure 5-18
DCS Circuit Menu**

3. Use the **Up/Down Arrow keys** to select a **channel mode** and press **Enter**. You will be returned to the **BERT Test Control** menu.

5-4.5.2 Direction (DCS only)

The test access mechanism of the **DCS** allows the **950RTS** to non-intrusively monitor a DS0 cross connect, or intrusively split a cross connect. In both case, the **DCS** provides access to both directions of the cross connected circuit. The **950RTS** uses the terms **facility** and **equipment** to designate these two directions of test. The port that is used to identify the cross connect is designated the equipment direction, while the port on the other end of the **DCS** cross connect is designated the facility direction. In the switching environment, therefore, all DS1 circuits should be referred to by the port that connects to the switch so that the equipment direction of test will mean testing towards the switch, while the facility direction of test will refer to testing towards the transmission facilities.

To select a direction:

1. Use the Up/**Down Arrow** keys to select the **Direction** field from the **BERT Test Control** menu.
2. Press the **Space bar** to toggle between **equipment** and **facility**. Selecting **equipment** allows you to test towards the equipment side of the **DCS**, and selecting **facility** allows you to test towards the facility side.

5-5 PERFORMING A DS1 BERT TEST

TO perform a **BERT** test:

1. Use the **Arrow** keys or type **s** to select **Start Test** from the **BERT Test Control** menu. The **950RTS** begins testing and the **BERT Test Control** menu changes. (See **Figure 5-19**)

Figure 5-19
BERT Test Control Menu (During Testing)

2. Use the **Arrow** keys to select **Stop Test** or type **s** to stop a test.

5-5.1 Error Inject

To inject an error while testing:

1. Use the **Arrow** keys or type **e** to highlight the **Error Inject** field and press **Enter** to display the **Error Injection** menu. (See **Figure 5-20**).

Figure 5-20
DS1 Error Injection

2. Use the **Arrow** keys to highlight the **Error** field.
3. Press the **Space bar** to display a pop-up menu which lists the following standard error types:

Bit, BPV, CRC (ESF only), Frame (D4 only)

4. Use the **Arrow** keys to highlight an error type from the pop-up menu.
5. Press **Enter** to accept the value and return to the **Error Injection** menu.
6. Use the **Arrow** keys to highlight **Inject**.
7. Press **Enter** to inject an error.
8. Use the **Arrow** keys to highlight **Quit Inject Ctrl** and exit to the **DS1 BERT Test Control** menu.

5-5.2 Loopback/Loopup

To begin a **loopback/loopup**:

Use the **Arrow** keys or type **L** to highlight the **Loopback** field and press **Enter** to display the **Loopback control** menu. (See **Figure 5-21**)

The **Pattern** field only appears if **ESF** framing or **User** defined code is selected.

**Figure 5-21
Loopback Control Menu**

5-5.2.1 Selecting a Loopback Code

To select a **loopback/loopup** code:

1. Use the **Arrow** keys to highlight the **Code** field.
2. Press the **Space bar** to display a pop-up menu which lists standard loopback/loopup codes.

CSU, 4-Bit, 5-Bit, User

3. Use the **Arrow** keys to highlight a format from the pop-up menu.
4. Press **Enter** to accept the value and return to the **Loopback Control** menu.

5-5.2.1.1 Choosing a User Defined Code

To select a **user-defined loopback/loopup code**:

1. Use the **Arrow** keys to highlight the **Code** field.
2. Press the **Space bar** to display a pop-up menu which lists standard loopback/loopup codes.

CSU, 4-Bit, 5-Bit, User

3. Use the **Arrow** keys to highlight **User** from the pop-up menu and press **Enter**. A **User Code** field will appear in the **DS1 Loopback Control** menu. (See Figure 5-22)

**Figure 5-22
User Code**

4. Enter a loopback code (0's and 1's). It should be 3 to 8 bits in length.
5. Press **Enter** to accept the value and return to the **Loopback Control** menu.

5-5.2.2 Sending a Loop Down Code

To loop down:

1. Use the **Arrow** keys to highlight the **Loop Down** field.
2. Press **Enter**. A loopdown code will be sent.

5-5.2.3 Performing a Loopback/Loopup

To perform a loopback/loopup:

1. Use the **Arrow** keys to highlight the **Send Loopup** field. The **Connection Status** menu will read: **Sending Loop-up**.

5-6 VIEWING RESULTS HISTORY

To view the results history after conducting a BERT test:

1. Use the **Arrow** keys to highlight **Result History** from the **BERT Test Control** menu.

The **Result History** menu will appear differently, depending on whether you chose, **Default**, **AvailTime**, or **Full** under **Results View**.

2. Press **Enter** to advance through the results screens, or type **q** to exit to the **BERT Test Control** display.

SECTION VI

TRUNK ROUTINING

6-1 AUTOMATIC TRUNK ROUTINING (ATR)

This section of the manual describes the **950RTS** automatic trunk routing (**ATR**) feature. This feature enables you to schedule trunk testing for unattended execution at a later time using either a screen-based or command-line user interface.

Three main topics will be addressed in this section:

- an overview of the **ATR** function
- instructions for configuring **ATR**
- setup and management of **ATR** schedules

Refer to the following **950RTS** documents for complete explanations of the testing methods and key **950RTS** features mentioned in this section:

- *Basic Installation and Administration Instructions*
- *AT&T Compatibility Bulletin #106 (Dec. 1981)*

6-1.1 Software Configuration

The **ATR** feature of the **950RTS** consists of several programs and a collection of data files. The **ATR** programs are described below:

<u>ATR Programs</u>	<u>Description</u>
• sched	Schedule Manager. Provides a command-line method of creating, updating, and reviewing ATR schedules in the database.
• vsched	Visual Schedule Manager. Provides a screen-oriented user interface for creating, updating, and reviewing ATR schedules in the database.
• atrd	Scheduling Daemon. Periodically reviews the Schedules Database, executes schedules which are ready to run, and reschedules those which have completed.
• atr	Test Driver. Runs scheduled trunk tests when invoked by the ATR daemon (ATRD).
• mdp105, nearend	Test Engine. Performs automatic routing to a type 100 (silent term), type 102 (milliwatt), or type 105 (responder) test line.

During operation, the **ATR** programs use several data files that reside in the **950RTS** library directory (**/u/950rts/lib**) - **configuration files**, which describe the **950RTS** and test switch test resources. It also uses the **Schedules Database**, which contains user test specifications, and **Test Results Files**, which contain results generated by the test engine.

Section VI Trunk Routing

The **ATR** program data files are described below:

<u>Configuration Files</u>	<u>Description</u>
• 950intfc • 950dintfc • 950chan	Interface Database. Contains test port information. (T1 board interfaces and channels)
• default.trks	Trunk Database files. Contain information on near-end switch trunk groups and members.
• sites/*	Site Files. Contain far-end switch and test path information.
• filters/*	Filter Files. Contain test results failure limits.
• schedules.*	Schedules Database.
• schedule.MMDDHH	Test Results file. default test results filename.

Figure 6-1 shows the relationship between the **ATR** programs and data files.

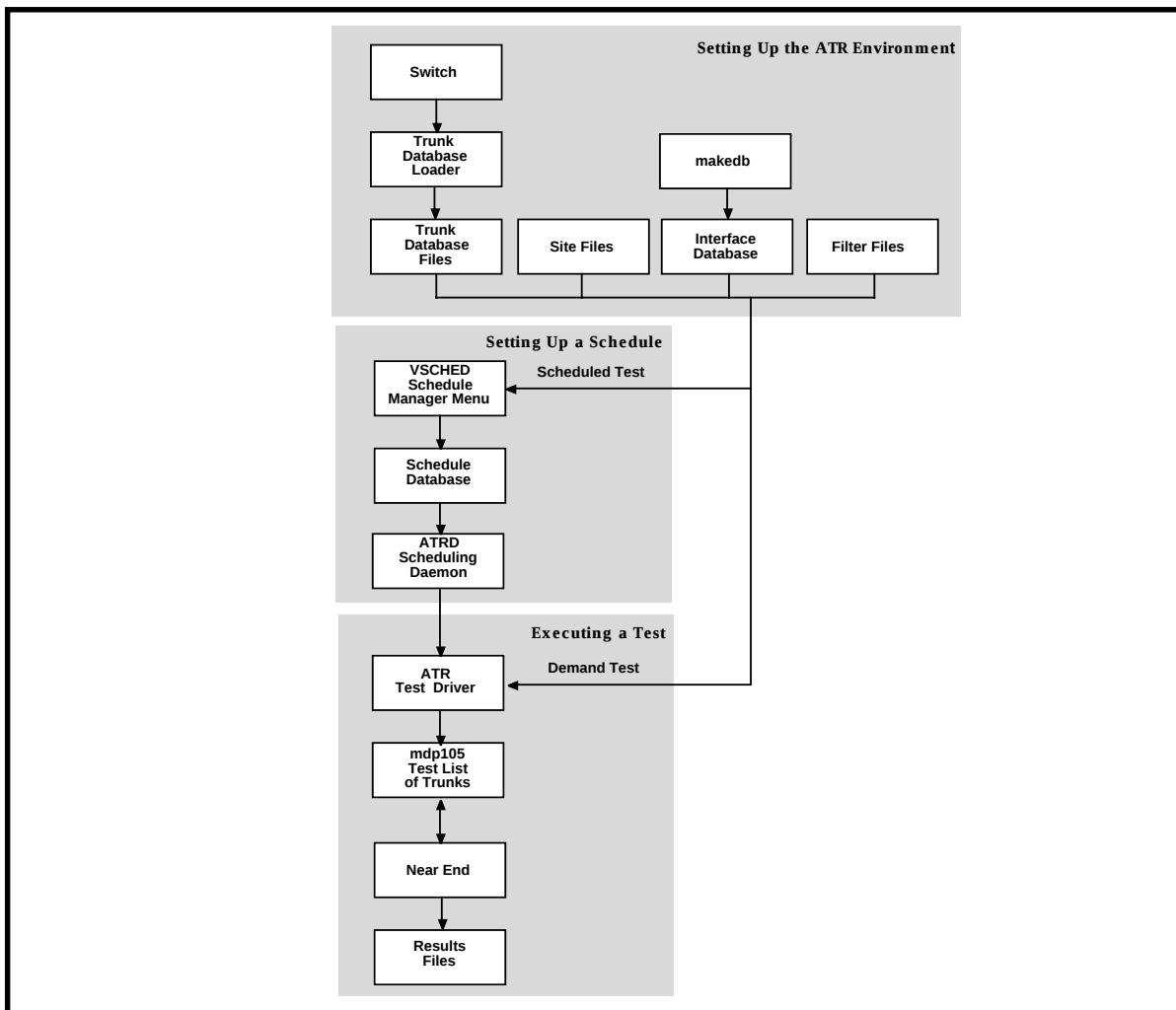


Figure 6-1
ATR Programs and Data File

6-2 CONFIGURING THE 950RTS

Before setting-up automated testing, the **950RTS administrator** must install and configure the **950RTS** as described in the *950RTS Remote Test System Installation and Administration Guide*. Confirm that testing is working correctly by running the demand testing utility, **vftest**. This utility is described in **Section II** of this manual.

6-2.1 Downloading The Near-End Switch Trunks Database

To further configure the **950RTS** for automatic trunk routing, the **950RTS administrator** downloads the near-end switch trunks database into the **ATR** trunks file, **default.trks** using the download programs. The download programs are described in the *950RTS Installation and Administration Guide*.

6-2.2 Adding Site and Filter Files

The **950RTS** administrator can optionally add site and filter files to the **950RTS** configuration. Site files contain test line and responder access codes for far-end switches, and trunk groups to the far-end. Adding a site file for each far-end switch makes it easier to enter test schedules. Custom filter files can also be added if the installation uses different test results limits than those in the standard filter file.

See **Section III** of the *950RTS Remote Test System Installation and Administration Guide* for a description of how to create these files.

6-3 MENU-BASED SCHEDULE MANAGER

6-3.1 Navigation

Navigation in the screen-oriented **ATR** program, **vsched**, is the same as in the **vftest** feature discussed in the beginning of **Section II** of this manual. You choose a menu or value by first highlighting the desired selection, then accepting it by pressing **Enter** or **Return**.

6-3.1.1 Screen Information

The **950RTS** screen always gives some basic information about the **950RTS** and the facility under test. That information is located at the top of the screen. This information includes:

- **950RTS** system name
- Current date and time

6-3.1.2 Quick Keys

Vsched has an information message at the bottom of each screen like those in the **vfctest** feature. In addition, it also has a set of **Quick Keys** listed at the bottom of each menu box that enable you to execute functions more quickly. (See **Figure 6-2**)

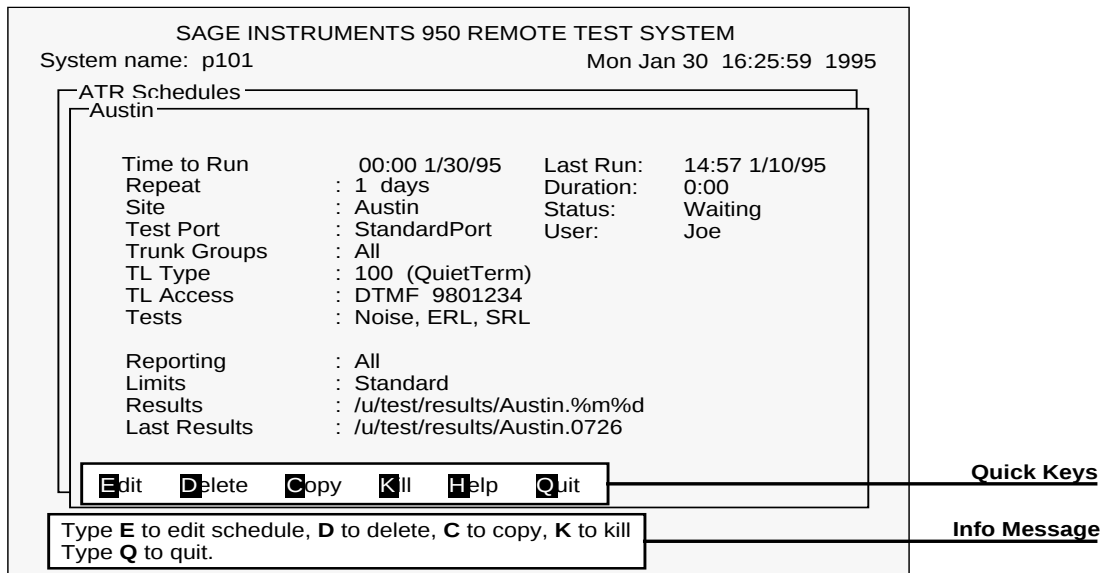


Figure 6-2
Quick Keys and Help Menu

Typing the first letter of a **Quick Key** name selects the corresponding function. For example, typing an **h** displays a help pop-up message. The **Quick Key** list on each menu includes **Help** and **Quit** functions. Pressing **Esc** will also leave any screen.

In addition, you can **refresh** the screen by typing **<ctl-L>**.

6-3.2 Planning ATR Test Schedules

6-3.2.1 Required Information

Before setting up a test schedule for automatic trunk routining, the following information must be obtained or selected:

- name for the schedule
- trunk groups to test
- test line or responder to use
- tests to run
- **950RTS** test port to use

The available **950RTS** test ports can be listed using the command **listports**. The **950RTS** administrator assigns names to the **950RTS** ports based on their connectivity.

The **950RTS** administrator may have set up site files describing trunk group connections and far-end switch test lines. The command **listsites** will show the configured sites. To view an individual site, type:

listsites site

Additional items you can set (change from defaults) in the schedule are:

- how often to repeat the test
- reporting style for the test results
- filename for the test results

6-3.2.2 Filling in the Schedule Worksheet

Use the **ATR Schedule Worksheet** in **Appendix II** of this manual to plan a schedule before entering it. An expanded tutorial version of the worksheet explains all schedule fields in detail.

When determining which trunk groups to test in a particular schedule, and when to run it, you should estimate how long it will run, and consider usage of test resources during this time.

For switches with maintenance dial plans that have a negligible call setup time, use the following figures to estimate the length of time the schedule will run:

- 9 seconds per individual test
- 50 seconds for a full set of 105 tests on a single trunk
- 8 minutes for full 105 tests on 10 trunks

For switches with a call setup time, this time that the schedule will be much longer.

To determine the number of trunks in a set of trunk groups, use the command **counttrunks**. To see all trunk groups in the default trunk file, with a count of members in each trunk, use **counttrunks all**.

The following test resources are required during the entire test period:

- **950RTS** digital signal processors (**DSP's**)
- **950RTS** test ports
- far-end responders
- **TCON** port (Ericsson switch)

Each test requires a single **DSP**, unless tests are looped back to the **950RTS** as a far-end, so that two **DSP's** are required. To determine the number of **DSP's** on the **950RTS**, use the command **countdsps**.

Each test requires a single test port (T1 interface and channel).

Section VI Trunk Routining

A test schedule will fail if there is no **DSP** available, no test port of the proper type is free, or if the far-end responder is busy.

6-3.2.3 Adding the Schedule to the Database

To enter the schedule, use the visual schedule manager **vsched**, or the command line version, **sched**. It is recommended that setting a time to run be delayed until the schedule has been checked by running a demand test.

To do this, exit from **vsched**, and run the command:

```
atr schedule-name
```

This will run the schedule immediately, with results shown on the display.

6-3.2.4 Enabling the New Schedule

Once the schedule has been verified, use **vsched** to set a time-to-run. This enables the schedule for automated testing.

6.3.2.5 Summary of Commands

The following commands are provided to assist in setup of **ATR** configuration files and schedules:

- **countdps**
- **counttrunks**
- **listgroups**
- **listlimits**
- **listports**
- **listschedules**
- **listsites**
- **printlimits**
- **printschedules**
- **printsites**

Any command which requires a command line argument will print a usage message if invoked without one.

6-3.3 Scheduling a Test

To schedule a test, create an **ATR schedule** using one of the schedule managers. For a full screen-based menu interface, use **vsched**. To run **vsched**, invoke it from the **UNIX** prompt:

```
% vsched
```

6-3.3.1 Initial Vsched Screen

After logging on to the system, the **ATR Schedules** screen displays a summary list of all schedules in the database. (See **Figure 6-3**)

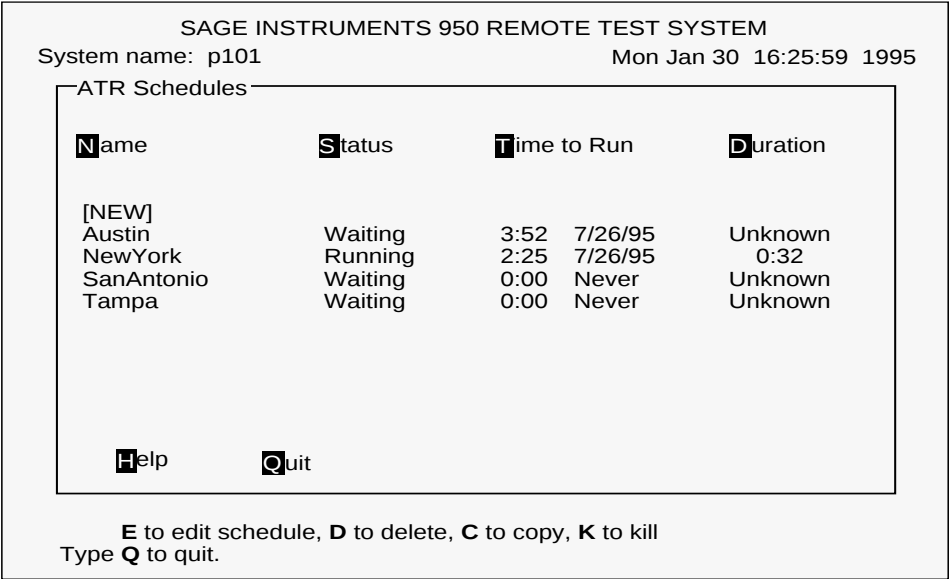


Figure 6-3
ATR Schedules Screen

If you selected a trunk that is connected to an **Ericsson** switch, the **Trunk Groups** field is replaced by the **Routes** field.

To select a schedule:

- 1. Scroll through the list in the **ATR Schedules** screen.
- 2. Use the **Arrow** keys to highlight a schedule.
- 3. Press **Enter** to accept the selection.

Section VI Trunk Routining

The system lists the **ATR schedules** alphabetically by schedule name on the initial **vsched screen**. The summary list also contains three other columns of information:

- Status
- Time to Run
- Duration

The **Status** field displays the current status of the schedule.

The **Time to Run** field displays the time and date at which the next test will take place or reads '**Never**' if the schedule is not scheduled to run.

The **Duration** field displays the length of time that the schedule last ran. You cannot change the information in the columns from within the **ATR Schedules** screen. It is updated automatically when you change the information in the **Schedule Edit** screen.

6-3.3.2 Sorting

You can sort the schedule list by **Time to-Run** or by any of the status fields by typing the first letter of the column header. For example, typing **T** will sort the list by **Time to Run**.

6-3.3.3 Preventing Schedule Overlap

The **Duration** field listed for each schedule in the summary list shows the elapsed time (**HH:MM**) of the most recent execution of the schedule. Use this information when selecting a time to run to minimize the chance of two schedules overlapping. This should be avoided if both schedules require a test resource that is not shareable.

6-3.3.4 Scrolling

If the list contains more entries than will fit on the screen, a **Down Arrow** will appear on the lower right portion of the screen. Pressing the **Down Arrow** key will scroll down through the entries.

6-3.3.5 Creating a New Schedule

To create a new test schedule:

1. Select **NEW** from the schedule list. The screen will change to a fill-in form showing the **Schedule Name** field.

Alternately, you can create a new schedule with a preexisting format by selecting an existing schedule, then typing **C** (Copy.) This will insure a consistent format from schedule-to-schedule.

2. Enter a name for the new schedule.

3. Press **Enter** to create your new schedule.

6-3.4 Schedule Edit Screens

After selecting an existing schedule or creating a new schedule, the system will display a **Schedule Edit** screen. (See **Figure 6-4**)

```

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM
System name: p101                               Mon Jan 30 16:25:59 1995

ATR Schedules
Austin

Time to Run      : 00:00 7/27/95      Last Run: 14:57 7/17/95
Repeat          : 1 days              Duration : 0:00
Site            : Austin              Status  : Waiting
Test Port       : StandardPort        User    : Joe
Trunk Groups    : All
TL Type         : 100 (QuietTerm)
TL Access       : DTMF 9801234
Tests           : Noise, ERL, SRL

Reporting       : All
Limits          : Standard
Results         : /u/test/results/Austin.%m%d
Last Results    : /u/test/results/Austin.0726

[Edit] [Delete] [Copy] [Kill] [Help] [Quit]

E to edit schedule, D to delete, C to copy, K to kill
Type Q to quit.
    
```

Figure 6-4
Schedule Screen

You can move from field-to-field using the **Arrow** keys, **Tab**, or **Enter**. Instructions for filling in the selected field appear in the information message at the bottom of the screen.

The **Schedule Edit** screen has two types of fill-in fields: **text entry**, and **selection**. These field types are described in the following sections.

6-3.4.1 Text Entry Fields

Text entry fields have instruction lines beginning with "**Enter...**". You can enter information in these fields from three different sources, depending on the text field:

- key entry
- pop-up menus
- checkboxes

In some fields, you must type in the desired value. (See **Figure 6-5**)

Section VI Trunk Routining

Fields that only support **key-entry** include:

- Time to Run
- Trunk Groups (with a DSC switch) **OR** Routes (with an Ericsson switch)
- Repeat (interval subfield)
- Results
- TL Access Number (number subfield)

```
SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM
System name: p101                               Mon Jan 30 16:25:59 1995
ATR Schedules
Austin
Time to Run      : 12:15 7/27/95      Last Run: 14:57 7/17/95
Repeat          : 1 days              Duration : 0:00
Site           : Austin              Status  : Waiting
Test Port      : StandardPort        User    : Joe
Trunk Groups   : All
TL Type        : 100 (QuietTerm)
TL Access      : DTMF 9801234
Tests          : Noise, ERL, SRL

Reporting      : All
Limits         : Standard
Results        : /u/test/results/Austin.%m%d
Last Results   : /u/test/results/Austin.0726

[Edit] [Delete] [Copy] [Kill] [Help] [Quit]

Enter date and time as HH:MM mm/dd/yy
Press <Tab> for next field, <Esc> to continue
```

Figure 6-5
Key Entry: Time to Run Field

If an **Ericsson** switch is recognized from the switch information line in the trunks file, the **Trunk Group** field header is replaced by **Routes**.

To key enter a value into a field:

1. Use the **Arrow** keys to highlight the field.
2. Enter the desired value.
3. Press **Enter** to accept the value.

Some text entry fields have **pop-up lists** or **checkboxboxes** to make function selection easier. You can activate these functions by pressing the **Space bar**. The instruction lines for these fields will end “...press **Space bar** to list.”

Pop-up lists contain selections that come standard with the **950RTS**, as well as any selections made when the **ATR** is configured. (See **Figure 6-6**)

Fields that have **pop-up** lists include:

- Site
- Test Port
- Limits

The screenshot displays the SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM interface. At the top, it shows 'System name: p101' and the date/time 'Mon Jan 30 16:25:59 1995'. Below this, the 'ATR Schedules' section is active, showing a list of schedules with 'Austin' selected. A pop-up list for the 'Site' field is displayed, showing 'Austin' as the selected option, with 'Tampa' and 'None' as other available options. The interface also shows various configuration details for the selected schedule, including 'Time to Run', 'Repeat', 'Last Run', 'Duration', 'Status', and 'User'. At the bottom, there are buttons for 'Edit', 'Delete', 'Copy', 'Kill', 'Help', and 'Quit', along with instructions on how to use these buttons.

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM

System name: p101 Mon Jan 30 16:25:59 1995

ATR Schedules

Austin

Time to Run : 00:00 7/27/95 Last Run: 14:57 7/17/95

Repeat : 1 days Duration : 0:00

Site Status : Waiting

Austin User : Joe

Tampa

None

Reporting : All

Limits : Standard

Results : /u/test/results/Austin.%m%d

Last Results : /u/test/results/Austin.0726

Edit Delete Copy Kill Help Quit

Type E to edit schedule, D to delete, C to copy, K to kill

Type Q to quit.

Figure 6-6
Pop-up Lists: Site List

To make selections from a **pop-up** list:

1. Press the **Space bar**. The pop-up will appear on the screen.
2. Use the **Arrow** keys to highlight the desired selection.
3. Press **Enter** to select your choice.

Section VI Trunk Routining

Checkboxes differ from the pop-up lists in that they contain a set number of functions that are not user-changeable. (See **Figure 6-7**)

Fields that have **checkboxes** include:

- Tests

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM

System name: p101 Mon Jan 30 16:25:59 1995

ATR Schedules

Austin

Time to Run	: 00:00 7/27/95	Last Run:	14:57 7/17/95
Repeat	: 1 days	Duration	: 0:00
Site	: Austin	Status	: Waiting
Test Port	: StandardPort	User	: Joe
Trunk Groups	: All		
TL Type	: 100 (QuietTerm)		
TL Access	: DTMF 9801234		
Tests	: Noise, ERL, SRL		

Type 105 Tests

☒ Loss	☒ Noise	☒ -Slope
☒ ERL	☒ SRL	☒ SRH

Results : /u/test/results/Austin.%m%d

Last Results : /u/test/results/Austin.0726

Edit **D**elete **C**opy **K**ill **H**elp **Q**uit

Press <Space bar> to select or unselect, <Tab> for next test, <Esc> to end

Figure 6-7
Checkboxes: Type 100 Tests

To select functions in a **checkbox**:

1. Press the **Space bar**. The **checkbox menu** will appear on the screen.
2. Use **Tab** to move between boxes.
3. Press the **Space bar** to check or un-check a box.
4. Press **Enter** after making all desired selections to accept your choices.

Pressing **Esc** will leave any **key entry box**, **pop-up** or **checkbox**.

6-3.4.2 Selection Fields

Selection fields have instruction lines beginning with “**Press Space bar to cycle through...**”. These fields have a preset list of selections. (See **Figure 6-8**)

Selection fields include:

- Repeat (units subfield)
- Reporting (digit format subfield)
- TL Access (DTMF/MF subfield)
- TL Type

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM

System name: p101 Mon Jan 30 16:25:59 1995

ATR Schedules —

Austin —

Time to Run	: 00:00 7/27/95	Last Run: 14:57 7/17/95
Repeat	: 1 days	Duration : 0:00
Site	: Austin	Status : Waiting
Test Port	: StandardPort	User : Joe
Trunk Groups	: All	
TL Type	: 100 (QuietTerm)	
TL Access	: DTMF 9801234	
Tests	: Noise, ERL, SRL	
Reporting	: All	
Limits	: Standard	
Results	: /u/test/results/Austin.%m%d	
Last Results	: /u/test/results/Austin.0726	

Edit **D**elete **C**opy **K**ill **H**elp **Q**uit

Press <Space bar> to cycle through selections
Press <Tab> for next field, <Esc> to continue

**Trunk
Routing**

Figure 6-8
Selection Fields: Reporting

To select functions in a **Selection** field:

1. Use the **Arrow** keys to highlight the **Selection** field.
2. Press the **Space bar** to scroll through the selections until the desired value appears in the field.
3. Press **Enter**.

Pressing **Esc** will leave any **Selection** field.

6-3.5 Schedule Save Screen

Pressing the **Esc** key leaves the **Schedule Edit** screen and displays the **Schedule Save** screen. This dialog box is identical to the **Schedule Edit** screen, except that the **Quick Keys** include **Save** and **Cancel** functions, and you can no longer edit the **Schedule** field. Saving the schedule will cause the changes you made to the fields to be saved in the database.

6-4 COMMAND-LINE SCHEDULE MANAGER

If you are familiar with the **ATR schedules**, you may want to use the command-line schedule manager, **sched**. Command-line operation also provides a means for programming **ATR** using **UNIX** scripting languages.

The following is a list of command-line operations:

To print a usage message, invoke sched with no arguments:

% sched

usage : sched key flags schedule

C	create, copy
M	modify
D	delete
K	kill
L	list

To create a new schedule, invoke sched with the **'C'** key.

For a new schedule with defaults for all fields, use:

% sched C new-schedule

or use an existing schedule as a template for your new schedule by typing:

% sched C old-schedule new-schedule

To change the field values in the new schedule, run **sched** with the **M** key.

To view a list of valid field option flags, run **sched** with the '**M**' key, but with no option flags:

```
% sched M
options usage:
    -d time [date]
    -c circuit-type
    -l line-type [DTMF | MF: access-code]
    -g group1 [... groupN]
    -s site
    -t test1 [... testN]
    -i interval [units]
    -f [q] filter
    -r resultfile
```

To display the schedule you just created, type:

```
% sched P schedule
```

To show a summary list of all schedules in the database, type:

```
% sched L
```

(For a complete description and more examples, See the **UNIX manual page** for **sched**. Type **man sched**)

6-5 USER CHANGEABLE SCHEDULE FIELDS

The user-changeable **schedule** fields are the same regardless of the manager program used. (See **Figure 6-9**)

SAGE INSTRUMENTS 950 REMOTE TEST SYSTEM

System name: p101

Mon Jan 30 16:25:59 1995

ATR Schedules

Austin

Time to Run : 00:00 7/27/95

Repeat : 1 days

Site : Austin

Test Port : StandardPort

Trunk Groups : All

TL Type : 100 (QuietTerm)

TL Access : DTMF 9801234

Tests : Noise, ERL, SRL

Reporting : All

Limits : Standard

Results : /u/test/results/Austin.%m%d

Last Results : /u/test/results/Austin.0726

Last Run: 14:57 7/17/95

Duration : 0:00

Status : Waiting

User : Joe

User-Changeable

Schedule Fields

E

D

C

K

H

Q

Edit

Delete

Copy

Kill

Help

Quit

E to edit schedule, D to delete, C to copy, K to kill

Type Q to quit.

Figure 6-9
User-changeable Schedule Fields

The following quick reference list describes the user-changeable **schedule** fields and their defaults:

(**Field Label** indicates the label displayed by **vsched**, and **Option Flag** is the field option flag used with **sched**.)

Table 6-1 User Changeable Schedule Fields

Field label	Option flag	Description	Default
Time to Run	-d	Date and time to run	Never
Repeat	-i	Rescheduling interval and units	0 days
Site	-s	Remote site trunks and responders	None
Trunk Groups	-g	Trunk groups at near-end switch	All
TL Type	-l	Remote test line or responder type	type 100
TL Access	-l	Access code for remote test line	DTMF 0
Test Port	-c	950RTS T1 interface & channel	first port in spanid
Tests	-t	950RTS voice frequency tests	Noise, ERL, SRL, SRH
Reporting	-f	Results reporting type	All
Limits	-f	Test failure limits	Standard
Results	-r	File pathname for test results	<i>schedule</i> .MMDDHH

User-changeable **schedule** fields can be divided into three types: **time** fields, **test description** fields, and **reporting** fields.

The following sections describe each of these types of user-changeable **schedule** fields in detail.

6-5.1 Time Fields ---

Time fields include the **Time to Run** field and the **Repeat** field. These fields determine when a given schedule will run.

6-5.1.1 Time To Run ---

The **Time to Run** field contains the **time of day** and **date** when the schedule will run. The **ATR daemon** uses this field to determine if a schedule is ready to run.

Time to run signifies the hour and minute at which the routine is scheduled to run. The time can be entered in a variety of formats:

hour (24-hour clock)
hour followed by **am** or **pm**
hour:minute (24-hour clock)
hour:minute followed by **am** or **pm**
noon or **midnight**

You can also enter the word '**now**' to indicate that a schedule will be executed immediately, or Enter the word '**never**' to indicate that a schedule is not ready-to-run.

Date to run signifies the day that the schedule will run. By default, the schedule will run when the time you indicated occurs next (today, or tomorrow). If you want to select a day other than the default day, specify the date. Dates are specified in one of the following ways:

MM/DD/YY (month/day/year)
day of the week (3 letters or full)
month name (3 letters, or full) followed by the **day**
today or **tomorrow**

Some examples of valid date/time specifications are:

22:15 6/12/95
11:45 pm oct23
midnight fri midnight next friday
3 3 a.m.the next day

After entering the desired time and date in the **Time to Run** field, press **Enter** to accept the values.

When running **sched**, use the **-d option** to display the **Time to Run** field.

6-5.1.2 Repeat Field

The **Repeat** function controls automatic rescheduling of tests. The fields have two subfields - a **numeric** field, and a **selection** field defining the rescheduling interval. When an **ATR schedule** completes, the scheduling daemon uses the values in the **Repeat** field to calculate the new time to run.

To request that a test be rescheduled after it runs, enter a number in the **Repeat** field. This number must be between **1** and **999**, (or **'0'** for no repeat). Press **Enter** to accept the value.

You can change the default repeat interval, **days**, by pressing **Tab** to reach the interval field, then pressing the **Space bar** until the desired interval appears. You may then select between **minutes**, **hours**, **days**, or **months**. Press **Enter** to accept the interval.

When running **sched**, use the **-i option** to change the **Repeat** field.

6-5.2 Test Description Fields

Test description field specify the **950RTS** ports to be used, test line access, trunks to be tested, and tests to be performed. These fields include the **Site** field, **Trunk Groups** field, and **Test Type**.

The **TL Type** field, **TL Access** field, and **Test Port** field are secondary fields that you must set if you did not specify a site in the **Site** field. If you have specified a site, you can set these secondary fields to override the settings in the **sites directory**.

6-5.2.1 Site Field

The **Site** field contains the name of a file in the **sites directory**. This file contains trunk groups and test equipment for the far-end switch. If you entered a site in the **Site** field, the test engine will use the contents of the **Site File**.

To use a **Site File**, enter (or select from the **pop-up menu**) the site name when running **vsched**, or use the **-s option** of **sched**.

6-5.2.2 Trunk Groups

The **Trunk Groups** field contains a list of trunk groups to be tested, or the default value, **All**. If the default is set, the test engine will test all trunks listed in the trunks file, **/u/950rts/lib/default.trks**, or all trunks in the **Site File**.

To restrict the trunks to be tested, enter a list of trunk groups into this field using **vsched** (or use the **-g option** of **sched**). This list can include hyphenated ranges of trunk groups. Some examples are:

12	test all trunks in group 12
9-12	test all trunks in groups 9 through groups 12
4,9-12,15	test all trunks in groups 4, 9 through 12, and 15

Note: Hyphenated ranges are not supported for the Ericsson switch, since the hyphen is valid in route names.

6-5.2.3 Test Line Type

The **TL** field indicates the type of far-end test equipment used during the test. Currently supported test line types include:

100	quiet termination
102	milliwatt
105	responder

To change the **test-line type** from the default, scroll through the **pop-up menu** until the desired line type is highlighted, and then press **Enter**.

6-5.2.4 Test Line Access

The **TL Access** field contains the dial digits for accessing the far-end test equipment. This field contains two subfields - a **format** field, set to **DTMF** or **MF**, and a **digits** field.

To enter the line access digits, press **Tab** while in the **TL** field to reach the subfield containing the digits. If the digits field is the default value, **0**, the test engine will look for the digits and format in the site file.

You may need to update the **Tests** field and **TL** field when the test-line type changes. Only a subset of the tests are available on type **100** and **102** test lines. The **Tests** field pop-up shows the tests valid for the currently set test line type.

6-5.2.5 Test Port

The **Test Port** function indicates which of the **950RTS T1 channels** are suitable for testing the trunks listed in the schedule. The test engine will select an available interface and channel of the specified type from the **Interface Database**.

Configured channels are shown in the **vftest circuits** screen and in the **vsched Test Port** field pop-up. To change the test port type, enter the new type in the **Test Port** field (or use the **-p option** of **sched**).

6-5.2.6 Test Type

The **Test** field is used by the test driver to determine which tests to run. The field defaults to all tests supported by the default type test-line (type 100).

To change the default setting, enter or select the desired tests in the **Tests** field (or use the **-t option** of **sched**). This release of **ATR** supports the following tests:

- Noise C-msg and/or C-notch noise
- Slope gain/slope
- Loss return loss
- ERL echo return loss
- SRL singing return loss low
- SRH singing return loss high

The *AT&T Compatibility Bulletin #106 (Dec. 1981)* gives detailed descriptions of these tests.

Any tests that are not supported by the far-end test line are ignored.

6-5.3 Reporting Fields

Reporting fields include the **Reporting** field, **Limits** field and **Results** field. These fields designate how the test results will be reported.

6-5.3.1 Reporting and Test Limits

The **Reporting and Test Limits** fields determine the style of reporting in the **Test Results File**. By default, the test engine saves results from all tests. For more concise results files, the you can change the reporting style to one of the following:

Table 6-2 Reporting and Test Limits Fields

Field value	Option flag	Description
FailedTrunks	-f	show tests for only trunks with test failures
FailedTests	-fq	show only tests that failed

Tests pass or fail based on the testing parameters specified in the **Test Limits File**. By default, the test engine uses the standard limits files, `/u/950rts/lib/filters/Standard.*`. You may select custom limits files from the **pop-up list** in the **Test Limits** field (or specify the file following the **-f flag** of **sched**).

6-5.3.2 Results File

The **Results File** field specifies the **UNIX** pathname of the file where the test engine saves the test results. By default, this file is created in your current directory, with a filename containing the schedule name and the time the test executed.

If the results file already exists, the test engine will append the new results to the end of the file. If you cannot write to the test file, the results will be saved in the **950RTS** default results directory, **/u/950rts/lib/results**. In either case, the name of the results file will appear in the **Last Results** Field of the schedule upon test completion.

To specify the results file name, fill in the **Results File** field while creating or editing a schedule in **vsched**, or use the **-r option**, followed by the filename, when using **sched**.

You can enter the results filename as a relative or absolute (starting with **/**) **UNIX** pathname. Include the schedule name in the results file, to assure uniqueness. (If schedules run concurrently with output to the same results file, the results will be interleaved in the file). You can include a time-stamp in the filename by embedding any special character sequences recognized by the **UNIX** date command. Some useful sequences are:

%m	month
%d	day of the month
%H	hour
%M	minute

(For a complete description, See the **UNIX manual page** for **date** or See **Appendix I** of this manual)

6-6 STATUS FIELDS

The **Schedule Edit** dialog box includes five status fields that are not user-changeable, but are changed by the **ATR daemon**. (See **Figure 6-10**)

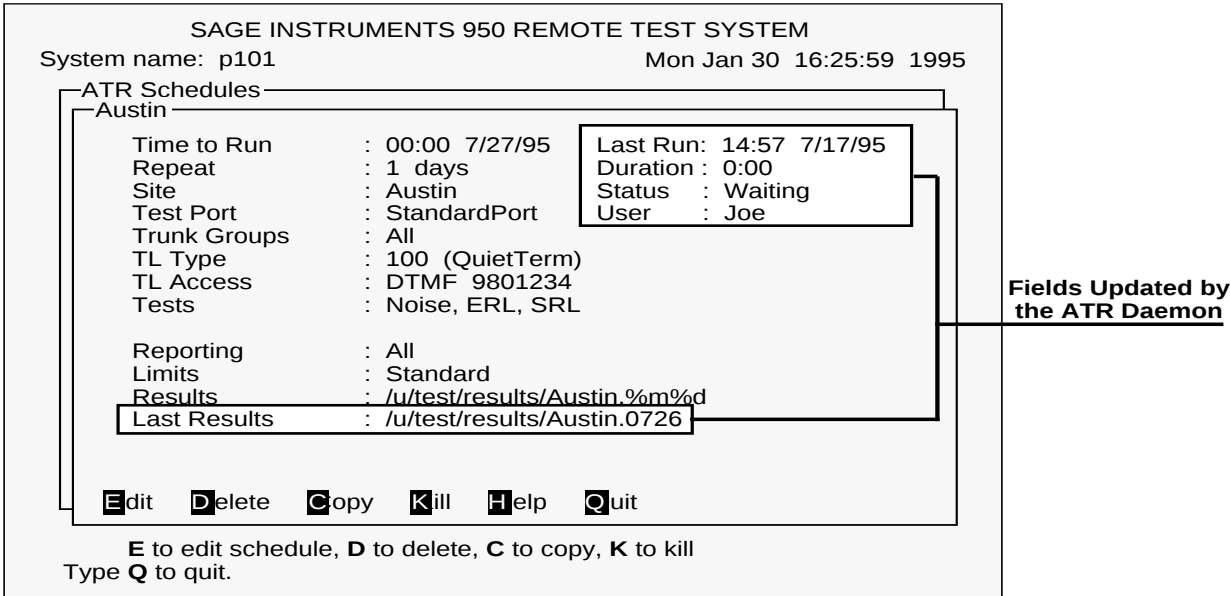


Figure 6-10
Status Fields

The following quick reference list describes the status fields changed by the **ATR daemon** and their defaults:

Table 6-3 Status Fields

Field label	Description	Initial value
Last Run	Date and time last execution began	Never
Duration	Elapsed hours:minutes of last run	Unknown (0:0)
Status	Current state	Waiting
User	User name of schedule creator	User-name
Last Results	Pathname of results file used last run	

6-6.1 Last Run

When execution of a schedule completes, the **ATR daemon** copies the information in the **Time to Run** field to this field.

6-6.2 Duration

When a schedule completes, the **scheduling daemon** saves the elapsed time since the schedule began in this field. This field is displayed in the format, **HH:MM** (hours:minutes).

6-6.3 Status

This field indicates the current status of a schedule. Currently, there are two status values:

- **Running** schedule is currently executing
- **Waiting** schedule is waiting to execute

6-6.4 User

This field contains the user name. The schedule creator owns the test results created while the schedule runs. This field is set only by the **sched** and **vsched** programs when a new schedule is created.

6-6.5 Last Results

This field contains the **UNIX** pathname of the test results file written to during the last run of the schedule.

6-7 MANAGING TESTS AND TEST RESULTS

The following sections discuss how to cancel tests, view test results, and bypass the **scheduling daemon**.

6-7.1 Cancelling a Scheduled Test

To **cancel a test** before it executes:

Change the schedule's **Time to Run** field to **Never**. You can do this by running **vsched**, selecting the schedule and typing **E** (Edit).

From the command line, use:

```
% sched M -d Never schedule
```

To **remove a schedule** from the database:

Run **vsched**, select the schedule from the summary list, then type **D'**(Delete). You will be asked to confirm by selecting or typing **Y** (Yes).

Section VI Trunk Routining

To use the **command-line interface** to delete a schedule, type:

```
% sched D schedule
```

6-7.2 Aborting a Running Test

To **cancel a test that is already running**:

1. Run **vsched**.
2. Select the schedule.
3. Type **K** (Kill). You will be asked to confirm by selecting or typing **'Y'** (Yes).

Alternately, from the command line, type:

```
% sched K schedule
```

6-7.3 Viewing Test Results

Test results are saved in a file with a name based on the **Results File** field of the **ATR schedule**. When the test completes, the actual results filename used is shown in the schedule's **Last Results** field. Test results are stored in plain **ASCII** files, and can be reviewed using any **UNIX** text editor or viewing utility; for example, **vi**. See the **UNIX** on-line documentation for more information on these utilities.

6-7.4 Running the Test Driver

You can run the test driver directly, without waiting for it to be executed by the **ATR daemon**. This allows you to test out schedules before setting a time to run, or even to run trunk routining without setting up a schedule.

To **test a schedule**, type:

```
% atr schedule
```

This command will execute the schedule immediately, with test results displayed on the terminal. To run an automated test without using a schedule, invoke **atr** with option flags specifying the test. These flags are analogous to the field option flags used with **sched**, excluding the scheduling options, **-d** (date) and **-i** (interval). An example of a test that runs all tests on all trunks listed in a site file titled **Austin**, with results displayed to the terminal, is:

```
% atr -s Austin
```

(For a complete description, See the **UNIX manual page** for **atr** or See **Appendix I** of this manual)

6-7.5 Related Documentation

The **UNIX** on-line manual pages for the **950RTS** programs contain further information on various topics. The following pages are currently available:

- atr
- atrd
- sched
- db
- dbrecord
- mdp105
- nearend
- CU

These manual pages can be accessed by typing:

man *topic*

at the **UNIX** prompt. They are also reprinted in ***Appendix I*** of this manual.

SECTION VII

ATME

7-1 INTRODUCTION

ATME stands for the CCITT automatic measuring and signaling testing equipment. It is intended to make transmission measurements, echo canceller tests and signaling system functional tests on all categories of international circuits terminating in exchanges with 4-wire switching.

ATME test line consists of both near end equipment (director) and far end equipment (responder). All commands are issued from the director to the responder in forms of MF digits. Although the ATME measurements are performed in both directions, the results measured at the far end are encoded as MF digits and sent back to the near end. ATME is similar to type 105 responder system in that both perform automatic measurements of the test circuit lines in both directions. The director initiates the test by calling the responder and waiting for an answer signal. The test sequence then begins when the answer supervision is detected. Unlike the type 105 responder, however, there is no test progress tone.

7-2 Operation Principles of ATME

7-2.1 ATME measurments

ATME essentially perform the following three basic types of tests:

1. **Transmission tests:**

These include power level measurements at three frequencies (400Hz, 1020Hz, and 2800Hz) and at two different levels (0dBm and -10dBm), Psophometric noise measurement (with or without the locking tone), S/TD (signal to total distortion) tests at -10 or -25 dBm levels.

2. **Echo-canceller tests:**

These include far-end and near-end echo canceller tests at two echo gain levels (+2dB and -10 dB) and at various echo delays ranging from 0 ms to 75 ms, and near-end and far-end echo canceller disabling test (to see if the echo canceller can be successfully disabled).

3. **Digital loop-back:**

This allows round trip BERT testing on digital circuits.

Transmission tests are performed first, then the echo canceller tests and lastly the BERT test.

7-3 CMS locking Tone and the Echo Canceler

The 2800Hz CMS locking is meant to lock and release the circuit multiplication system (CMS) present in some international circuits. In ATME, this tone is also used as filling tone during the transmission tests.

The reason for this is that the echo canceller, once disabled, will rearm itself when the circuit is quiet for more than 100 ms. During the transmission tests, this filling tone is inserted into any timing gaps that may be longer than 100 ms. Per O.22 recommendation, users are given to option the turn this locking tone on or off at the near end (director end). At the far end there is a special tone detector to make sure the far end is consistent with the near end (that is, when the near end is chosen to send CMS locking tone, so does the far end, and vice versa).

The echo canceller disabling tone is a steady 2100Hz tone with 180 deg phase reversals every 450 ms. Most echo cancellers require to see at least two phase reversals, which implies that the disabling tone must be longer than 900 ms. The O.22 recommendation also specifies an echo suppressor disabling tone, which is a pure 2100Hz tone with no phase reversals. The 'modern' echo cancellers use nonlinear processors to suppress the residual echos left over by the echo cancellers. Based on the O.22 recommendation, the nonlinear processor effect should be discounted, that is, it should be disabled during the echo canceller tests. However, the nonlinear processor in some echo cancellers cannot be disabled by this 2100Hz tone. Instead of disabling the nonlinear processor, the 2100Hz tone will put the echo canceller into a special data mode (with only half of its full cancellation power).

SECTION VIII

TERMINAL REMOTE CONTROL

8-1 INTRODUCTION

The **950RTS** can be controlled from a remote terminal using either **FacetTerm** or **Tiny Term**. The following sections provide brief descriptions of the basic user commands for setting up and running the two terminal emulation programs. Refer to their respective user's guides for more complete descriptions.

Note: **FacetTerm** is only available with the Automated Cellular Test System (ACTS).

8-2 FACETTERM

FacetTerm is a **UNIX**-based terminal emulation program that enables you to run multiple test sessions on the **950RTS** simultaneously. The **950RTS** currently supports up to **36** simultaneous users. With **FacetTerm**, you can log in as up to **10** users from a single terminal. In addition, you can run multiple programs on **FacetTerm**, which is particularly helpful when editing and importing database files from a word processor or editing program into the **ATR** program.

FacetTerm also provides windowing functions that enable you to switch quickly between windows. These windows can be resized and tiled on the desktop so that all test sessions can be viewed at the same time. (Resizing is only available if you are running an Alpha Terminal.)

Sage Instruments ships **FacetTerm** preinstalled on the **950RTS**. The following sections describe the basic setup and navigation commands for **FacetTerm**.

8-2.1 Startup

To start **FacetTerm**, at the **UNIX** prompt type:

```
facetterm <Return>
```

The initial startup screen will appear. (See **Figure 8.1**)

```
*****
*                               FacetTerm                               *
*                               Version 3.0.0                           *
*       Copyright (c) Structured Software Solutions, Inc. 1986-1993.   *
*****

FacetTerm user #1 of 8-Facet process: Facetline #0 is available.

Terminal type is vt220.  FacetTerm Terminal type is: vt220.

# FacetTerm vt220  default  description file 10/25/90
#   See also:          vt200  7-Bit mode    vt200 8-Bit mode    vt100 mode
# -----
#   80 column only    ==> vt220-7          vt220-8
#   80/132 column     Vt220-7          Vt220-8
# -----
# FacetTerm vt220-7 (80 column only)  description file 07/29/89
#   vt220
#   VT200 7-Bit mode switchable to VT200 8-Bit mode and VT100 mode
#   !!!IMPORTANT NOTE!!!
#   Set Terminal to: XON/XOFF

facetTerm 'menu hotkey' is control-F
FacetTerm 'window command hotkey' is Control-W - Press Return to start:
```

Figure 8.1
Initial Display

Verify that the setup information is correct. **FacetTerm** will then launch the startup configuration file **default.facet** and start shells in windows 1 and 2 and the pull down menu in window 10. You may copy this file to your home file to customize it so that **FacetTerm** automatically launches the windows running your selected programs or **950RTS** sessions.

You will notice that you are logged in four times: there is a window for the **TTY** device, windows for each of two shells, and a window for the menu.

From this point, all keystrokes are directed to the program running in the current window unless you press either of the hot keys (**Control-W**, **Control-F**) which enable you to interact with **FacetTerm**.

To verify that you are running a shell on a window, at the **UNIX** prompt type:

tty <Return>

To verify how many user windows are running, at the **UNIX** prompt type:

who <Return>

7-2.2 Interacting With FacetTerm

When you want to interact with **FacetTerm** to switch or resize menus, print, quit, or run another program, you can interact with the program either through the graphical menu interface, or the command line interface.

8-2.2.1 Menu-Based Navigation

To interact with FacetTerm through the graphical menu interface:

- 1. Press **<Ctrl> F** to access the **Menu Bar**. The Menu Bar is superimposed over the current window. (See **Figure 8.2**)

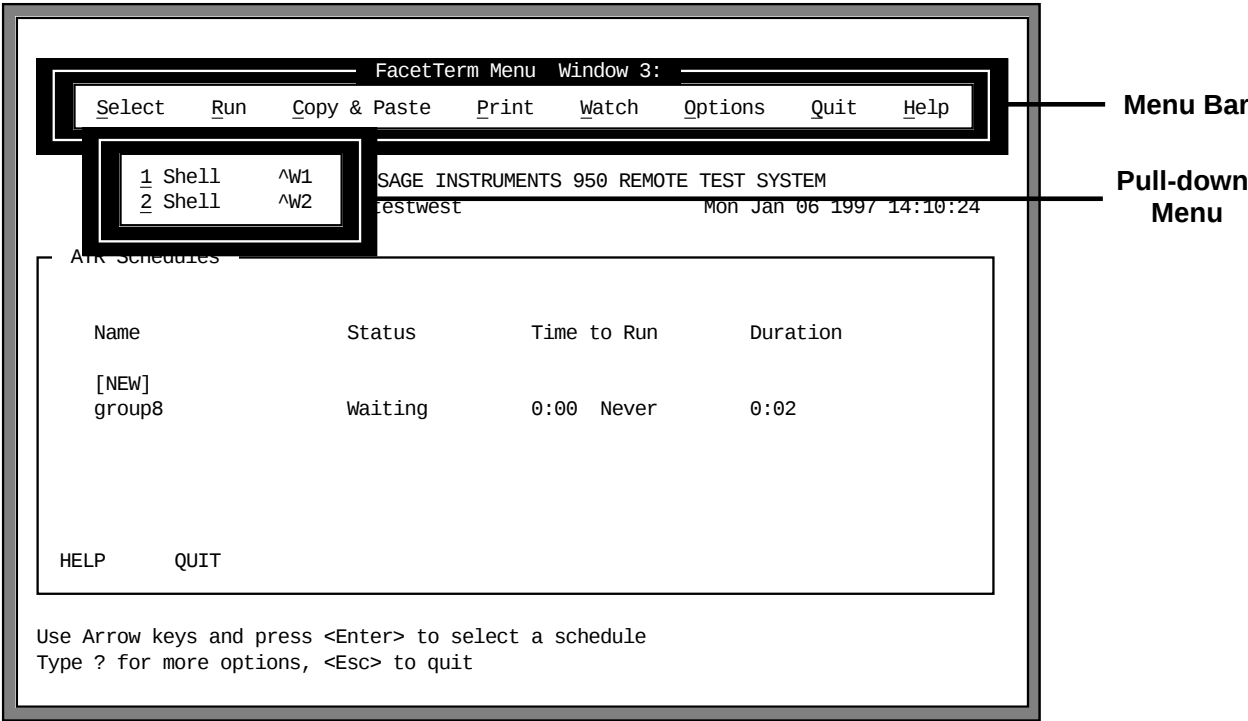


Figure 8.2
Menu Bar and Pull Down Menu

- 2. Use the **Left/Right Arrow** keys to move left or right through the selections in the menu bar.
- 3. Press **<Return>** to select an item in the menu bar. The associated pull down menu appears.
- 4. Use the Up/Down Arrow keys to move up or down through the selections in a pull-down menus
- 5. Press **<Return>** to select an item in the pull down menu.

You may use **HotKey** commands from within the menus by typing the first (underlined> letter of your selection. These **HotKey** commands are:

S	<u>S</u> elect and go to a shell	O	Select setup <u>O</u> ptions
R	<u>R</u> un a program or 950RTS session	Q	<u>Q</u> uit FacetTerm
C	<u>C</u> opy and Paste	H	Get <u>H</u> elp
P	<u>P</u> rint	^W1 through ^W0	Select windows 1 through 10.
W	A ctivate a <u>W</u> atch file		

8-2.2.2 Command Line Navigation

To use the command line interface

1. Press **<Ctrl> W**. Your keyboard input is directed to Facet**Term** rather than the program running in the current window.
2. Use the command line commands listed in **Section 7-2.2.1**.

Note: When selecting a shell, you do not need to enter **^W** before the shell number since you are already in command line mode. You only need to type the number of the window.

3. Press the **Spacebar** or **ESC** to cancel a command.

8-3 TINY TERM

The recommended emulator for DOS and Windows platforms is TinyTerm, a product of Century Software.

SECTION IX

SAGE COMMAND LANGUAGE (SCL)

9-1 SUMMARY

This document describes the *Sage Command Language (SCL)*, a command interpreter for the **Sage 950RTS** communications test system. **SCL** provides two capabilities:

- interactive command-line control of the **950RTS** testing functions
- a scripting language for automating testing applications

The design objectives for **SCL** were to provide a command-line tool that was flexible and concise enough for programming use, but simple and “friendly” enough for interactive use. An additional objective was to base **SCL** on a non-proprietary language with an existing user community. It is expected that **SCL** will grow, as Sage-developed commands and 3rd party extensions are added.

9-2 BACKGROUND

SCL is based on the general-purpose *Tool Control Language (TCL)* developed at UC Berkeley in the late 1980's. **TCL** was designed as a programmable control language for embedding in applications, and is currently in use in a variety of applications areas. The **TCL** language has been ported to Windows and Macintosh, as well as most **UNIX** platforms, and is under active development at Sun Microsystems.

SCL is a **TCL** interpreter which has been extended to include commands that control the **Sage 950RTS**. **TCL** provides commands that support standard programming tasks:

- setting and evaluating variables
- controlling execution flow
- defining procedures
- accessing files
- handling errors and exceptions.

SCL also contains the widely used **expect** extension to **TCL**. **Expect** provides commands for automating interaction with programs and devices, and is well suited for controlling remote telephony equipment over serial lines.

At Sage Instruments, **expect** has been used for automating call setup on Ericsson switches during automatic trunk routing, and for implementing remote control of the Sage **930** test system, from the **950RTS**.

9-3 TCL BASICS

A **TCL** command consists of the command name, followed by command arguments. Commands and arguments are separated by white-space, and a newline or semicolon terminates the command. A command can be a built-in command implemented within the interpreter, an application-specific command implemented in the interpreter, or a **TCL** procedure implemented externally as a script. A line beginning with **#** is a comment line.

Special syntax constructs used in **TCL** are:

" "	double quotes	group words together
{ }	curly braces	group words together, preventing substitution
[]	square brackets	group words, evaluate as a command, and substitute the return value
\$	dollar	substitute value of variable

All **TCL** commands, arguments, and variables are strings. Arithmetic expressions are not evaluated in the interpreter, but by the built-in **expr** command. **TCL** expressions have the same operators and precedence as in **C**.

All **TCL** commands return a result string and a result code. A non-zero result code indicates an error, and will cause the interpreter to exit, unless the **catch** command is used to handle the error. If an error occurs, the result string contains the error message. Otherwise, the results string contains the results of the command, if any. Some of the more basic **TCL** built-in commands are:

Control:	if, for, case, break, continue
Variable:	set, global, array
Procedure:	proc, return
Exceptions:	catch, error
Strings:	format, scan, string
Files:	source, open, close, puts, gets
Processes:	exec

9-4 SAGE EXTENSIONS TO TCL

SCL extends **TCL/Expect** by adding commands specific to communications testing:

- establishing a test connection
- setting test parameters
- running and controlling a test
- reporting test results
- configuring and obtaining status from test resources

All **950RTS** tests can be run from within **SCL**. To facilitate adding new Sage commands to **SCL**, an **SCL C API** has been written. This **API** hides some of the complexity of the **TCL C** interface, and allows future implementation of non-**TCL 950RTS** command languages. The **SCL C** programming interface is documented separately.

9-5 TEST ACCESS COMMANDS

Before a test can be initiated, a test connection must be established. The **SCL connect** command is used to define the circuit to test and reserve a test resource. Most commonly, the user would specify this information by supplying a valid test port from the interface database. (The **950RTS listports** commands shows all defined ports). With no arguments, **connect** returns the current connection information.

When testing is complete, the **disconnect** command is used to release the test resource. The current test connection is automatically disconnected when **SCL** exits. The **disconnect** command is only needed if the user wants to switch test connections during a single **SCL** session.

The initial release of **SCL** supports a single test connection.

Command summary:

connect	Get current connection info
connect circuit	Allocate circuit and reserve an appropriate test resource
disconnect	Release circuit and test resource

where **circuit** can be one of:

port	Use circuit info listed for this port in the interface database (idb)
span chan	Use the named channel, and the named DS1 from the idb , for DS0-level test
span	Use the named DS1 from the idb for DS1-level testing
dcs digrp chan	Use named circuit on a digital cross-connect for TAD test
dcs ds3 ds1	Use named circuit on a digital cross-connect for FAD test

9-6 PARAMETER COMMANDS

Most **SCL** tests have test parameters which are set up before running the test. In **SCL**, these are generally installed as variables which can be retrieved or set using the **setparam** command. These variables are set to default values when **SCL** begins. The default values can be viewed using **defparam**. This command can be used in conjunction with **setparam** to restore changed parameters to their defaults.

Parameter values can also be specified on the command-line as flags, using the format:

command -param1 val1 -param2 val2 ...

Some **SCL** commands also recognize a limited number of positional parameters, so that the parameter flag need not be used. The **help** message for each command indicates how to provide parameters.

Section IX SCL Overview

To display command parameters, use the following:

<code>showparam</code>	Show current settings of all test parameters
<code>showparam test</code>	Show current settings of all parameters for test
<code><TD>showparam -def</code>	Show default settings of all test parameters
<code><TD>showparam -def test</code>	Show default settings of all parameters for test

To retrieve and set parameter values, use:

<code>setparam test param</code>	Return current value of named parameter
<code>setparam test param val</code>	Set named parameter to val
<code>defparam test param</code>	Get default value of named parameter

Parameter values are always checked before setting, and an error message is returned, describing valid values.

9-7 TEST COMMANDS

A test is started by issuing one of the test commands or by selecting the test and issuing the **starttest** command. A test command will load the current test parameter settings, override with any command-line parameters, set the current test selection, and begin testing. A test command only returns a result if the test can not be started. If the test start succeeds, results can be obtained by using the **reporttest** command, described below. In **950RTS** Release 4.1, the following tests are provided in **SCL**:

absdelay	Absolute delay
callprogress	Call progress
ds0bert	DS0 BERT
ds1bert	DS1 BERT
ds1monitor	DS1 monitor
impnoise	Impulse noise
jitter	Phase and amplitude jitter
mtone	Measure tone
par	Peak/average ratio
rloss	Return loss
noise	Noise
tone23	23 Tones

A single test connection can support only one test at a time. A running test must be stopped before a new test can be started.

For convenience during interactive use, an installed flag variable, **testrunning**, indicates whether a test is currently running. The command **showvars**, will display the current setting of **SCL** installed variables.

The parameters and result types for each test are listed in **Section AV-15 SCL Test Parameters and Result Types**.

9-8 GENERIC TEST CONTROL COMMANDS

A set of generic test control commands is available for all **SCL** tests. Test results are returned from a running test using the **reporttest** command, and cumulative counters are cleared using the **clrtest** command.

The **reporttest** command has command-line arguments to specify the type and number of test results to return. Results are returned as a string consisting of result-type/result value pairs. Supplying a list of desired result types will restrict the results to only those requested.

The **stoptest** command stops a running test. The **starttest** command restarts a stopped test, or starts a test that has been chosen with the **selecttest** command. With no arguments, the **selecttest** command shows the currently chosen test.

Command Summary:

clrtest	Clear counters in a running test
reporttest #	Report # results from test
reporttest all	Report continuously from test
reporttest all res1...resN	Report only named result types
selecttest	Get currently selected test
selecttest test	Select the named test
starttest	Start the selected test
stoptest	Stop a running test

9-9 BERT TEST CONTROL COMMANDS

The following commands are available only while **ds0bert** and **ds1bert** tests are running.

Command Summary:

ds0inject type	Inject errors during DS0 BERT
ds0code type	Send net code during DS0 BERT
ds0loop	Show DS0 loopback status
ds0loop on type	Send loopup during DS0 BERT
ds0loop off	Send loopdown during DS0 BERT
ds1inject type	Inject errors during DS1 BERT
ds1loop type	Show DS1 loopback status
ds1loop on type	Send loopup during DS1 BERT
ds1loop off	Send loopdown during DS1 BERT

The loop, netcode, and error inject types are installed parameters, which may be overridden by either a positional or flag parameter.

Section IX SCL Overview

The parameters used with the BERT test control commands are:

<u>Command</u>	<u>Parameter</u>	<u>Default</u>	<u>Description</u>
ds0inject	injecttype	1bit	Type of error injection
ds0loop	loopcode	csualt	Loopback code
ds0code	netcode	cmi	Transmitted network code
ds1inject	injecttype	1bit	Type of error injection
ds1loop	loopcode	csu	Loopback code

The BERT test parameter names used by **SCL** are the same set used in **berttest**, both for command-line options, and in the **.bertconfigs** file.

9-10 MISCELLANEOUS CONTROL COMMANDS

Control commands are available to manage switched circuit dialing, control digital-cross connects, configure T1 interfaces, and handle other telephony functions.

Command Summary:

dcx	Identify connected digital cross-connect circuit and TAP status
dcx split	Split DCS circuit
dcx monitor	Monitor DCS circuit
dcx release	Release DCS circuit
echocan	Send echo canceller disable tone
nap msec	Delay
hook	Get hook state
hook on	Set hook state on
hook off	Set hook state off
hook term	Get termination hook state off
intfc	Get T1 interface config
intfc framing coding	Set T1 interface config (DTI only)
sdigits digits	Send digits
stone	Get current tone sender state
stone on freq level	Start tone sender
stone off	Stop tone sender

9-11 TEST RESOURCES

SCL provides commands to obtain status from and provide some independent control of test resources. There are currently two test resource commands: **dsp**, and **dti**.

Command Summary:

dsp	Identify currently allocated DSP
dsp version	Show DSP software version
dsp history brd dsp	Show DSP command/results log
dti	Identify currently allocated DTI
dti version	Show DTI software version
dti history brd vdp	Show DTI command/results log
dti debug	Dump the DTI board putc buffer

Other **dti** commands for development use only, are described in the help message.

9-12 INTERACTIVE USAGE

SCL appears similar in function to the standard **UNIX** command interpreters - the “shells”. When **SCL** is invoked interactively, it displays a prompt, reads a line of input and executes the command. Typing the **quit** (or **exit**) command will terminate **SCL** and return the user to the **UNIX** prompt.

SCL will interpret any **TCL**, **Expect**, or **SCL** command. **SCL** also can interpret **UNIX** commands — **TCL** provides a mechanism for executing a **UNIX** program in a subprocess, using the **exec** command, and this is performed automatically during interactive use. Additionally, during interactive use, commands can be abbreviated — the shortest unique string matching the command may be used.

Like the **UNIX** shells, a “command history” feature is available. The **history** command will show previous commands. Typing **!!** will repeat the previous command, and typing **!5** will repeat the 5th command in the history. This history feature is standard in **TCL** and is similar in style to the **UNIX** shell.

The **help** command lists available commands with a brief description. Individual help messages for each command show the options that are supported and how to invoke them.

Command summary:

help	Show summary help message
help command	Show help message for named command
quit	Shutdown test session and exit SCL
exit	Shutdown test session and exit SCL

Section IX SCL Overview

A sample session of using the **950RTS** tone sender interactively using **SCL**:

```
% scl
scl<1>      help
             (full help text here – see attached command summary)
scl<2>      help stone
             stone                      Return sender status
             stone on <freq> <level>   Turn on tone sender
             stone off                  Turn off tone sender

scl<3>      connect rb101
scl<4>      stone on 400 -16
             (other activity here)
scl<5>      stone off
scl<6>      stone on 200 -16
scl<7>      !5
scl<8>      quit
```

9-13 SCRIPTING USAGE

SCL commands can be collected in a file for execution at a later time. One difference from interactive use is that all Sage commands must be prefixed with **scl_**, to distinguish them from the general-purpose commands.

A sample **SCL** script file, which prints out the version numbers of the **950RTS** test resources is:

```
# resversion.scl - example of using SCL to print out
#      software versions of test resource embedded code
scl_dsp version
scl_dti version
```

After this script is saved in the file, **resversion.scl**, it is executed by typing:

```
scl resversion.scl
```

As with other **UNIX** scripts, **SCL** will be invoked implicitly by the **UNIX** shell, if the pathname of the **SCL** interpreter is included as the first line of the script file (following the characters **#!**).

Alternatively, an **SCL** file can be executed from an interactive **SCL** session using the **TCL** *source* command:

```
% scl
scl<1> source resversion.scl
```

A more elaborate version of this script could format the output, and perhaps check the versions for compatibility:

```
#!/u/950rts/bin/scl
#####
# resversion.scl - example of using SCL to print out
#           software versions of test resource embedded code
#####
# obtain versions from the test resources
set dspver [scl_dsp version]
set dtiver [scl_dti version]
# test for correct version of DTI embedded code
if {$dtiver < 4} {
    puts "DTI version mismatch: $dtiver"
    exit
}
# format and print
puts "950RTS Test Resources:"
puts "      DSP version $dspver"
puts "      DTI version $dtiver"
```

9-14 REFERENCES

Books

Tcl and the Tk Toolkit, John Ousterhout, Addison-Wesley, 1994

Definitive book describing TCL and the Tk graphical extension, written by the developer.

Exploring Expect, Don Libes, O'Reilly & Associates, 1995

Definitive book that fully describes the expect extension to TCL, written by the developer.

Introductory Papers

TCL: An Embeddable Command Language, John Ousterhout
Proceedings of the 1990 Winter USENIX Conference

expect: Scripts for Controlling Interactive Programs, Don Libes
Computing Systems, Vol. 4, No. 2
University of California Press Journals, November 1991.

Section IX SCL Overview

World-Wide Web and Internet Resources

TCL World-Wide Web Info

<http://www.sco.com/Technology/tcl/>

The World-Wide Web Virtual Library: Tcl and Tk

<http://cuiwww.unige.ch/eao/www/TclTk.html>

TCL 7.4 Manual Pages

<http://www.sco.com/technology/tcl/man2/tcl-man.html>

The Expect Home Page

<http://expect.nist.gov/>

The SunScript Home Page

<http://www.sunscript.sun.com>

comp.lang.tcl Usenet group

Sage Documentation

Sage 950RTS User's Guide

9-15 SCL TEST PARAMETERS AND RESULT TYPES

9-15.1 ds0bert

<u>Parameter</u>	<u>Default</u>	<u>Description</u>
pattern	511	BERT pattern
format	ds0a	data format
subrate	56kb	DS0A data subrate
blkcorrect	off	DS0A block correction enabled
echo	off	echo canceller state
<u>Result</u>	<u>Description</u>	
elapsed	duration of test, in seconds	
status	test status	
errs	bit error count	
ber	bit error rate	
sls	sync loss seconds	
efs	error free seconds	
pctefs	percent error free seconds	
es	errored seconds	
pctes	percent errored seconds	
ses	severely errored seconds	
pctses	percent ses	
fail	failed seconds	
pctfail	percent failed seconds	

avail	available seconds
pctavl	percent available seconds
unavl	unavailable seconds
pctunvl	percent unavailable seconds
data	PCM sample
netcode	received network code, or PCM sample

9-15.2 ds1bert

<u>Parameter</u>	<u>Default</u>	<u>Description</u>
pattern	511	BERT pattern
Result	Description	
elapsed	duration of test, in seconds	
status	test status	
errs	bit error count	
ber	bit error rate	
sls	sync loss seconds	
efs	error free seconds	
pctefs	percent error free seconds	
es	errored seconds	
pctes	percent errored seconds	
ses	severely errored seconds	
pctses	percent ses	
fail	failed seconds	
pctfail	percent failed seconds	
avail	available seconds	
pctavl	percent available seconds	
unavl	unavailable seconds	
pctunvl	percent unavailable seconds	
bpv	bipolar violations	
frm	frame errors	
crc	CRC errors	
oof	out-of-frame	
bslip	bit slips	
fslip	frame slips	

9-15.3 mtone

<u>Result</u>	<u>Description</u>
freq	frequency, in Hz
level	level, in dBm

Section IX SCL Overview

9-15.4 noise

<u>Parameter</u> type	<u>Default</u> cnotch	<u>Description</u> Noise measurement type
Returns noise measurement, in dB.		

9-15.5 rloss

<u>Parameter</u> type	<u>Default</u> erl	<u>Description</u> Return loss measurement type
<u>Type</u> erl	<u>Description</u> Echo return loss, in dB	erlC

APPENDIX I

AUTOMATIC TRUNK ROUTINING

ON-LINE MANUAL PAGES

A1-1 ATR (SAGE)

Name

atr - Sage 950RTS automatic trunk routing utility

Syntax

```
atr <schedule>

atr <dest> <options>
[-p port]
[-t @testfile | test1 [test2...testN] ]
[-g @<groupfile> | <group1> [... <groupN>] |
    <group1>-<groupN>]
    [-f[q] [filterfile]]
    [ -p <port> ]
```

where <dest> is:

```
-l <line-type> DTMF|MF:<access#>
-s <site> [-l <line-type>]
```

Description

The “atr” command is invoked by the ATR daemon, atrd (SAGE), to run scheduled trunk testing. “atr” can also be used from the command line to run immediate trunk testing. Test results are printed to the standard output, and are optionally processed by a test limits filter so that only error results are saved.

Before automated testing is set up, the 950RTS must be installed and configured as described in the “Basic Installation and Administration Instructions”. Confirm that testing is working correctly by running the demand testing utility, vftest(SAGE).

Test specifications are defined by an ATR schedule (see sched (SAGE)) or with command line options, described below.

Options

The following options are supported:

- f Specify that results be passed through a test limits filter so that only test failures are recorded in the results file. This option can be followed by the 'q' modifier, to request "quiet mode" results (only the test that failed will appear in the results file). The optional filter name argument specifies a custom filter (currently, a file in the directory /u/950rts/lib/filters) to use. The default filters are /u/950rts/lib/limit.*.
- g Specify the trunk groups to be tested. Trunk groups may be specified as a list containing trunk numbers and ranges of trunk numbers. Alternatively, the name of a file containing trunk numbers may be supplied, using the syntax @trunkfile.

If this option is not selected, all trunk groups in the site file, if specified, or else all trunks in the configuration file, /u/950rts/lib/trunkfile.cap, will be tested.

- l Specify the test line type and access code. Currently supported line types are: 100 (quiet termination), 102 (milliwatt), and 105 (responder).

If a site is specified with the '-s' option, then the access code need not be specified — it will default to the access code for that line type listed in the site file. If this option is not selected, but a site is, then this field Defaults to the "most capable" test line in the site file.

- p Specify the type of 950RTS port to be used during testing. Valid ports and their types are listed in the configuration file /u/950rts/lib/spanid. When testing begins, the ATR daemon selects an available port of the desired type.

If this option is not selected, the port field in the schedule defaults to the first port type listed in the configuration file.

- s Specify a test site. The test line and group specifications in the file /u/950rts/lib/testsites/<site> will be used unless overridden by the '-l' and/or '-g' options. The format of a site file is illustrated in the sample site file /u/950rts/lib/site.

- t Specify the tests to run. If this option is not selected, all available tests are run. Tests that are currently available include: erl, loss, noise, slope, srl, srh.

Examples

* Run tests

```
% atr -p cct2 -g 2 17-19 -l 100 DTMF:777 -t erl slope
```

This command runs all available tests on all trunks in groups 2, 17, 18, and 19. The tests will use an available 950RTS channel of type "cct2", and will use a quiet termination (type 100) test line accessed with the DTMF digits "777".

This example runs the "erl" and "slope" tests using a type type 105 responder, whose access code is listed in the austin"site file". The groups to be tested will also be obtained from the site file.

```
% atr -s austin -l 105 -f
```

This example runs the "erl" and "slope" tests using a type 105 responder, whose access code is listed in the austin "site file". The groups to be tested will also be obtained from the site file. Test results will be filtered using the default limits filter.

* Print usage message:

```
% atr
usage: atr <schedule>
      atr <dest>
      ...
```

Environment Variables

SAGEDIR	If set, overrides the default 950 directory (/u/950rts)
---------	---

Files

\$DBDIR/*.dat	SCO ISAM data files
\$DBDIR/*.idx	SCO ISAM index files
\$SAGEDIR/lib/schedule.dat	ISAM data file
\$SAGEDIR/lib/schedule.idx	ISAM index file
\$SAGEDIR/lib/spanid	Circuit configuration file
\$SAGEDIR/lib/trunkfile.cap	Trunk group configuration file
\$SAGEDIR/lib/limits.*	Default result filters
\$SAGEDIR/lib/filters/*	Custom result filters
\$SAGEDIR/lib/results/*	Result files
\$SAGEDIR/lib/testsites/*	Site description files

See also

sched (SAGE), atrd (SAGE), mdp105 (SAGE), nearend (SAGE) vftest (SAGE)

Sage 950RTS Basic Installation and Administration
Sage 950RTS Automatic Trunk Routining
Sage 950RTS Manual Testing User's Guide

Appendix I Sched

A1-2 ATRD (SAGE)

Name

atrd - Sage 950RTS automatic trunk routing daemon

Syntax

atrd [<interval> | -k] &

Description

The “atrd” daemon periodically reviews the Sage 950RTS “schedules” database, executing schedules which are ready to run, and rescheduling those which have completed.

When “atrd” examines the schedules, it selects the oldest runnable schedule and forks an atr(SAGE) process to manage it. When the ATR process completes, “atrd” updates the schedule status and history fields.

“atrd” is spawned (or awakened, if it already exists) whenever an ATR schedule is created or modified using sched (SAGE). “atrd” can also be started from the command line.

The “interval” argument specifies the number of minutes that “atrd” will sleep after completing one pass through the schedules. When the interval expires, or when “atrd” receives a HUP signal, the daemon awakens and reexamines the schedules. The default interval is 15 minutes.

Invoking “atrd” with the ‘-k’ flag causes “atrd” to shutdown gracefully. This option is restricted to the “admin” user and super-user.

Environment Variables

SAGEDIR	If set, overrides the default 950 directory (/u/950rts)
---------	---

Files

\$SAGEDIR//atrd.pid	daemon process id
\$SAGEDIR/log/atr.log	Default ATR log file
\$SAGEDIR/lib/schedule.*	Schedule database files
\$SAGEDIR/lib/results/*	Result files

See also

sched (SAGE), atr (SAGE)

Sage 950RTS Basic Installation and Administration
Sage 950RTS Operating Manual

A1-3 SCHED (SAGE)

Name

sched - schedule management for 950RTS automatic trunk routing

Syntax

```

sched C      <options>] <schedule>
              -p      <port>
              -d      <time> [<date...>]
              -l      <linetype> [DTMF|MF:<access#>]
              -g      <grp1> [...<grpN>] | @<grpfile> | <grp1>-<grpN>
              -t      <test1> [...<testN>] | @<testfile>
              -s      <site>
              -i      <interval> [<unit>]
              -f      [q] [<filter>]
              -r      <resultfile>

sched C      <schedule> <new-schedule>
sched M      <options> <schedule>
sched P | D | K <schedule>
sched H | V
sched L [L | T]
    
```

Description

The “sched” command provides a command-line method of creating, updating, and reviewing automatic trunk routing (ATR) schedules in the 950RTS database.

An ATR schedule consists of fields containing specifications for an automated test run, including which trunks to test, the date and time for testing to begin, and which tests to run. The “sched” option flags (listed below) are used to set or change these test specifications. The complete set of fields in a schedule database record is described in dbrecord(SAGE).

The ATR schedule database is examined periodically by the ATR daemon, atrd(SAGE), which spawns a testing process for each schedule that is ready to run. Testing proceeds, using the test resources indicated in the schedule. Upon completion of testing, results are saved in a results file and the status and history fields of the schedule are updated. If a rescheduling interval has been specified, a new date-time to run is set in the schedule.

Before automated testing is set up, the 950RTS must be installed and configured as described in the “Basic Installation and Administration Instructions”. Confirm that testing is working correctly by running the demand testing utility, vftest(SAGE).

The action of “sched” is controlled by a “function” argument, described below.

Functions

The following functions are provided by “sched”:

- C Create or copy a schedule. Option flags are used to set the test specifications. Any fields which are not explicitly set will use the default value.
- D Delete a schedule.
- H Help (print a usage message).
- K Kill a schedule (abort a running test).
- L List all schedules in the database.
- M Modify a schedule. The option flags are used to select the fields to modify.
- P Print a schedule.
- V Show version number of the “sched” program.

Options

Option flags are used with the ‘C’ and ‘M’ functions to indicate schedule fields to set or update. The following options are supported:

- d Specify the time and date to run the tests. The time can be specified as <hour>, or as <hour>:<minute>, optionally followed by “am” or “pm” (default is 24-hour clock). The special times “noon”, “midnight”, and “now” may be used.

The optional date can be specified as <day>/<month>/<year>, or as <month-name><day> or as <weekday>. Both the month-name and the weekday can be abbreviated to 3 letters. The special days “today” and “tomorrow” are recognized.

If no date is given, today is used if the current hour precedes the specified hour; otherwise, tomorrow is used. Similarly, if no year is given, the current year is used if the current day precedes the specified day; otherwise, next year is used.

If this option is not selected, the date/time field in the schedule is set to the default 0:0 1/1/99.

- f Specify that results be passed through a test limits filter, so that only test failures are recorded in the results file. This option can be followed by the ‘q’ modifier, to request “quiet mode” results (only the test that failed will appear in the results file). The optional filter name argument specifies a custom filter (currently, a file in the directory /u/950rts/lib/filters) to use. The default filters are /u/950rts/lib/limit.*

- g Specify the trunk groups to be tested. Trunk groups may be specified as a list containing trunk numbers and ranges of trunk numbers. Alternatively, the name of a file containing trunk numbers may be supplied, using the syntax @trunkfile.

If this option is not selected, all trunk groups in the site file, if specified, or else all trunks in the configuration file, /u/950rts/lib/trunkfile.cap, will be tested.

- i Specify a rescheduling interval. The numeric <interval> argument indicates that after a test completes, the schedule date/time will be reset to the current time plus <interval> units. The optional <unit> argument may be "days" (default), "hrs", "mins", or "months".

- l Specify the test line type and access code. Currently supported line types are: 100 (quiet termination), 102 (milliwatt), and 105 (responder).

If a site is specified with the '-s' option, then the access code need not be specified — it will default to the access code for that line type listed in the site file. If this option is not selected, but a site is, then this field defaults to the "most capable" test line in the site file.

- p Specify the type of 950RTS test port to be used during testing. Valid ports and their types are listed in the configuration file /u/950rts/lib/spanid. When testing begins, the ATR daemon selects an available port of the desired type.

If this option is not selected, the port field in the schedule defaults to the first port type listed in the configuration file.

- r Specify the pathname of a file where test results will be recorded. If the results file already exists, results will be appended. The current date can be included in the results filename by embedding one or more date specifications of the form %<spec>, where spec can be: d (day), m (month), y (year), M (minute), H (hour). The default results file path is /u/950rts/results <schedule>. <day>. <month>.

- s Specify a test site. The test line and group specification in the file /u/950rts/lib/testsites/<site> will be used unless overridden by the '-l' and/or '-g' options. The format of a site file is illustrated in the sample site file /u/950rts/lib/site.

- t Specify the tests to run. If this option is not selected, all available tests are run. Tests that are currently available include: erl, loss, noise, slope, srl, srh.

Examples

* Print usage message

% sched H

usage: sched <func> [<args>]

C <args> <schedule>	create schedule
C <schedule> <new-schedule>	copy schedule
M <args> <schedule>	modify schedule

Appendix I Sched

P <schedule>	print schedule
D <schedule>	delete schedule
K <schedule>	kill running schedule
L[T L]	list schedules
H	help
V	print program version

* List all schedules in the database

```
% sched L
name      time      status      lasttime      elapsed
-----
aus       10:13,04/12/95  waiting    10:13,04/11/95  1:01
sat       00:00,01/01/99  waiting    16:00,12/31/69   0:0
weekly    04:00,03/31/95  running    16:00,02/31/95   4:26
```

In this example, the “sat” test has not yet been scheduled, the “aus” test is waiting to run at 10:13 on 4/12, and the “weekly” test is currently in progress. The “weekly” test began at 4 am on 3/31, and the last time this schedule ran, it required 266 minutes to complete.

* Create a schedule

```
% sched C -d 4am fri -i 7 -p cct2 -g 2 17-19 -l 100 DTMF:777 weekly 'main'
Creating record with name 'main' in database 'schedules'
```

This command creates a schedule to run all available tests at 4 a.m. every Friday. All trunks in groups 2, 17, 18, and 19 will be tested. The tests will use an available 950RTS channel of type “cct2”, and will use a quiet termination (type 100) test line.

The next example creates a schedule to run the “erl” and “slope” tests at 11:30 pm every 3 days, beginning on April 1. The tests will use a type 105 responder, whose access code is listed in the austin “site file”. The groups to be tested will also be obtained from the site file.

```
% sched C -d 23:30 apr 1 -i 3 -p cct2 -s -T erl slope austin -l 105
aus.main
Creating record with name 'aus.main' in database 'schedules'
```

* Modify a schedule

```
% sched M -f -r aus.out aus.main
Updating record with name 'aus.main' in database 'schedules'
```

This example changes the schedule “aus.main” so that results are filtered using the default limits filter, and are written to the file “aus.out” in the current directory, instead of to the default results file (/u/950rts/results/aus.main.1.4).

* Kill a running schedule

```
% sched K weekly
sched: killing running schedule 'weekly'
```

* Print a schedule

% sched P weekly

Name : weekly
Time to Run : 04:00 3/31/95
Test Port : cct2
Trunk Groups : 2,17-19
Tests : Noise, Loss, Slope, ERL, SRL, SRH
Status : Waiting
TL Type : 100 (Quiet Term)
TL Access : DTMF:777
Site : None
Repeat : 7
Unit : days
Process ID : 203
User : test
Last run : 16:00 2/31/95
Duration : 1:20
Limits : Standard
Reporting : All
Results : /u/test/weekly.%m%d
Last Results : /u/test/weekly.0231

Environment Variables

SAGEDIR If set, overrides the default 950 directory (/u/950rts)

Files

\$SAGEDIR/lib/schedule.dat	ISAM data file
\$SAGEDIR/lib/schedule.idx	ISAM index file
\$SAGEDIR/lib/spanid	Circuit configuration file
\$SAGEDIR/lib/trunkfile.cap	Trunk group configuration file
\$SAGEDIR/lib/limits.*	Default result filters
\$SAGEDIR/lib/filters/*	Custom result filters
\$SAGEDIR/lib/results/*	Result files
\$SAGEDIR/lib/testsites/*	Site description files

See also

db (SAGE), atr (SAGE), atrd (SAGE), dbrecord (SAGE), vftest (SAGE)

Sage 950RTS Basic Installation and Administration
Sage 950RTS Automatic Trunk Routining
Sage 950RTS Manual Testing User's Guide

Appendix I DB

A1-4 DB (SAGE)

Name

db - Sage 950RTS database management utility

Syntax

h		help (usage)
p		print databases
p	[<db> [<name>]	print record
t	<db>	list fields
c	<db> <field>=<val>...	create record
m	<db> <name> <field>=<val>...	modify record
d	<db> <name>	delete record
i	<db>	initialize database
C	<db><name><newname>	copy record
X	<db>	remove database
L	<db><name><field>=<val>...	lock and modify record
T	<db>	list field labels
/*Not yet implemented*/		
db	u	unlock record
db	B	backup database
db	R	restore database
db	V	verify database

Description

The “db” command provides a command-line method of accessing the SAGE 950RTS database.

“db” provides functions to read, update, and maintain the database. The action of “db” is controlled by the “function” argument. The available functions are described below.

Most “db” functions are followed by a “database” argument, which indicates which database file to use. The command “db p” will print a list of valid databases. The database argument consists of the full name of the database or a unique leading substring. Most functions require that the database file be initialized — use “db i <database>” to do this.

Many “db” functions require a “name” argument. The name of a record in a database serves as the unique identifier of that record.

Note: only users in the same UNIX group as the user who initializes the database files will have permission to change the database.

The following functions are provided by “db”:

- c Create a database record. Fields in the record are set using <field>=<value> arguments. The valid fields in a database record can be listed using “db t <database>”. The “name” field must be set uniquely within the database file. All other “name” type fields must be set, and must refer to existing database records (unless the DB_NOCHECK environment variable is TRUE —see the “Environment Variables” section below).
- d Delete a database record. The “key” argument indicates which record to delete.
- h Help. Prints a usage message.
- i Initialize a database. The database file is created, but no records are created.
- m Modify a database record. The “name” argument indicates which record to modify. One or more fields may be modified, using “<field>=<value>” arguments.
- p Print database information. With no arguments, the ‘p’ function prints out a list of valid databases. With a database argument but no “name” argument, the function prints out all records in the database. If a “name” argument is given, only that record is printed.
- t List the names of fields in the database record.
- L Lock a record in the database. The record remains locked until the user types the interrupt key (DEL or <Ctrl>C). This function is provided in order to test record locking.
- X Delete a database. The database file is removed. Note that if a record in the database is locked, the first invocation of the ‘X’ function may fail. Repeating the command will remove the database.
- C Copy a record.
- T List field labels.

Examples

The following example shows how to create an “interfaces” database add a record for interface 0, and then change the record.

* Print usage message:

```
% db h
usage: db <func> <db> [args]
```

Appendix I DB

h		help (usage)
p		print databses
p	[<db> [<name>]	print record
t	<db>	list fields
c	<db> <field>=<val>...	create record
m	<db> <name> <field>=<val>...	modify record
d	<db> <name>	delete record
i	<db>	initialize database
C	<db><name><newname>	copy record
X	<db>	remove database
L	<db><name><field>=<val>...	lock and modify record
T	<db>	list field labels

* List valid databases:

```
% db p
Valid databases are:
950intfc
mqmtable
950chan
950dintfc
950dacs
bertconfigs
groups
trunks
equipment
schedules
```

* Create the "950intfc" database:

```
% db i 950intfc
Creating database '950intfc'
```

* List valid fields for database records:

```
% db t 950intfc
```

Valid fields in '950intfc' records are:

```
name
num
```

* Create a record

```
% db c950intfc name=main num=0
Creating record with name 'main' in database '950intfc'
```

* Print all records in the database:

```
% db p 950intfc
name      num
-----
main      0
```

* Update the record (change to interface #1)

```
% db m 950intfc main num=1
```

Updating record with name 'main' in database '950intfc'

* Print the changed record:

```
% db p 950intfc main
name      num
-----
main      1
```

Environment Variables

DBDIR	Database directory (defaults to /usr/950rts/lib)
DB_NOCHECK	Boolean indicating whether to allow creation of a record having a field that references a non-existent record in another database (defaults to FALSE)

Files

\$DBDIR/*.dat	SCO ISAM data files
\$DBDIR/*.idx	SCO ISAM index files

See also

dbrecord (SAGE)

Appendix I DBRECORD

A1-5 DBRECORD (SAGE)

Name

dbrecord - SAGE 950RTS database record formats

Description

The SAGE 950RTS database files contain information used by the 950 automatic test routing scheduler. The database can be setup and updated by screen-based editors, vsched (SAGE), or command-line oriented utilities, sched (SAGE), db (SAGE).

The database currently consists of 6 types of database files:

- interfaces
- channels
- groups
- trunks
- equipment
- schedules

The current version of the database uses the SCO Indexed Sequential Access Method (ISAM) for database storage. The ISAM record format for each database file is described below.

Formats

interfaces

0	20	128
+	+	+
name	num	
+	+	+
20	2	

channels

0	20	40	42	62	82	84	128
+	+	+	+	+	+	+	+
name	type	num	iface name	dialdigits	dialmode	status	
+	+	+	+	+	+	+	+
20	20	2	20	20	2	2	

groups

0	20	128
+	+	+
name	id	
+	+	+
20	20	

trunks

0	20	40	42	128
+	+	+	+	+
name	dialdigits	dialmode	group name	
+	+	+	+	+
20	20	2	20	

equipment

0	20	40	42	44
+	+	+	+	+
name	type	dialdigits	dialmode	status
+	+	+	+	+
20	20	2	2	2

schedules

0	20	24	44	64	94
+	+	+	+	+	+
name	date-time	channel type	group spec	tests	status
+	+	+	+	+	+
20	4	20	20	30	2
96	98	118	120		
equip type	equip dialdigits	equip dialmode	site		
+	+	+	+	+	+
2	20	2	20		
140	142	144	146	148	152
interval	interval unit	pid	uid	lasttime	elapsed
+	+	+	+	+	+
2	2	2	2	4	4

156	176	178	228	278	511
filter	filter mode	result file	last result file		+
20	2	50	50		

Files

\$DBDIR/	database directory
\$DBDIR/*.dat	SCO ISAM data files
\$DBDIR/*.idx	SCO ISAM index files
/usr/950rts/lib	default database directory (if DBDIR not set)

See also

dbrecord (SAGE), db (SAGE), dbedit (SAGE)

A1-6 MDP105 (SAGE)

Name

mdp105 - perform auto-routining with dialing via DSC maintenance dial plan

Synopsis

```
mdp105 <ckt> <trunkfile> [flag] <dest> [-t <testfile>]
```

Description

Performs automatic routining to a type 100 (silent term), type 102 (milliwatt), or type 105 (responder) test line. The test results are printed to stdout.

Parameters

<ckt>	A circuit name, defined in /u/950rts/lib/spanid, which identifies an interface and channel.
<trunkfile>	File containing trunk numbers and (optional) trunk groups.
[flag]	An optional flag identifying the type of test line: -s Silent term (Type 100) -m Milliwatt (Type 102) -r Responder (Type 105)
<dest>	Digits to dial to specify the destination test line. Default is DTMF unless the <dest> string starts with an 'm' character, or it contains a 'k' or 's' character.
-t <testfile>	Optional flag indicating that <testfile> is a file containing a list of tests to perform. Default is all applicable tests for the type of test line.

Maintenance dial plan dialing

The call is made by the following sequence:

1. Go off hook
2. Pause
3. Dial the trunk number, in DTMF, expanded to five digits.
4. If the destination number is MF, follow the trunk number with a DTMF pound sign (#) and pause.
5. Dial the destination number
6. Perform the test.
7. Go on hook.

The trunk number is assumed to be five digits or less. If less, it is left-padded with zeros. For example, trunk number 123 is signaled as 00123. The destination number can be MF or DTMF. The default is DTMF unless the destination number is prefixed with an 'm' character, or unless it contains a 'k' or 's' digit (corresponding to Kp or St). The following table gives the special characters to use in the <dest> string for MF digits:

To send	Use
KP	k
ST	s
STP	a
ST2P	b
ST3P	c

Trunk file format

The trunk file contains trunk and trunk group information. It is normally created from the trunks database by the automatic routing software. Only lines which start with a number (0-9) are used; the others are assumed to be headers or comments. The first number on each line is used as the trunk number, and the second number on each line is used as the trunk group number. Any following text on the line is ignored.

The trunk number is actually sent. The trunk group number is only used in the display of the results.

Test file format

The test file specifies which tests are to be performed. The file contains one line only, with any combination of the following words to specify the tests to perform:

- loss
- noise
- slope
- erl
- srl
- srh

For example, the following line in testfiles will case loss, noise, and gain/slope measurements to be performed:

loss noise slope

Output Format

Output is sent to stdout. The test results are prefaced with the following header:

```
mdp105 rev 1.3.950322
(c) 1995 Sage Instruments
Using ckt <test2> Far end equip # 9500505
```

Each trunk's results are formatted as follows (the Results and Test Start information are actually on the same line, but are split here for formatting purposes:

```
*** Results for trunk 337 group 4 Test start Fri Mar 24 09:51:20
Call complete time 2.2
Loss      Near:  1.0 dB      Far:   0.1 dB
Noise     Near: 18.0 dBrnC   Far: 15.5 dBrnC
Noise/Tone Near: 35.3 dBrnC   Far: 36.5 dBrnC
Slope 404 Near:  1.0 dB      Far:   0.7 dB
Slope 1004 Near:  0.6 dB      Far:   0.4 dB
Slope 2804 Near:  2.4 dB      Far:   1.1 dB
ERL       Near: 39.0 dB      Far: 39.5 dB
SRL       Near: 39.0 dB      Far: 39.5 dB
SRH       Near: 39.0 dB      Far: 39.5 dB
- Trunk 337 test complete
```

There is a blank line inserted between each trunk's results.

Other Files

mdp105 uses nearend to perform the actual testing, so nearend must be in the user's search path. This is normally the case as both utilities are kept in the same directory.

Examples

Assuming that the trunk and trunk group information is stored in a file called trunks, and that testing is to be performed via circuit test1 on an SS7 IMT to a milliwatt tone at 9500502, the command line would be

```
mdp105 test1 trunks -m 9500502
```

If the trunk under test was not SS7 (requiring MF signaling) and the destination was a 105 responder:

```
mdp105 test1 trunks -r m9200105
```

See Also

nearend(Sage), atr(Sage)

Standards Conformance

mdp105 is based on DSC Communication Corporation document FSD-010-0411-X08 Rev 1.0, SS7 Trunk Testing.

A1-7 NEAREND (SAGE)

Name

nearend - do a near-end 100/102/105 test line test, including signaling and dialing.

Synopsis

```
nearend <int> <chan><port name> <-0|-2|-5> [-d|-m <digits>]. [-t <t1>
[t2 t3...]>
```

Parameters

<int> DS1 interface number, starting at 1, that will be used to connect with the local switch

<chan> Channel on that DS1 to perform the test

<-0|-2|-5> Flag indicating silent term (-0 = TL100), milliwatt (-2 = TL102), or responder (-5=TL105)

[[-d | -m] <digits>] Optional mf (-m) or dtmf (-d) digits to <digits> send to connect to the destination. Multiple sets of digits can be given. Use 'w' to indicate wink, '~' to indicate a predial pause. Digits to send are contained in the file [[-d | -m] digits, one digit string per line

<@digits>]

[-t <t1> [t2...]] Optional flag proscribing which tests to perform (default is all applicable tests). The tests are any combination of

loss	noise	slope
erl	srl	srh

[-t @tests] The tests to be performed are in the file tests (format described below)

Dialing notes

The channel goes off-hook, and digits are sent in the order they are given on the command line. Multiple dial strings can be specified by concatenating them on the command line. They can be different formats; for example, a string of DTMF digits followed by a string of MF digits.

If a wink must be received before the digits are sent, precede the digits with a 'w'; if a pause is necessary before the digits are sent, precede the digits with a '~'. Multiple calls can be made with one invocation of nearend by putting the digits into a file, one digit string per line. The file is specified by preceding the file name with an '@' symbol and placing the '@filename' combination after the -d or -m flag. A separate call is made for each line in the file.

Specifying tests

The default is to perform all tests that are applicable to the type of test line dialed, as follows:

Silent term (TL100)	Noise (c-message), erl, srl, srh
Milliwatt (TL102)	Level, noise (c-notch)
Responder (TL105)	Level, noise (c-message and c-notch), gain/slope, erl, srl, srh

The -t flag is used to specify individual tests to perform. The tests can either follow the -t flag on the command line, or they can be listed in a file. Specify all of the tests to be performed in one call on one line, separated by spaces or tabs. If more than one line contains tests to perform, the tests on the first line will be done in one call, the tests on the second line will be performed in a subsequent call, and so on.

Output Format:

The test results are prefaced with the following header:
nearend rel 1.1 (c) 1995 Sage Instruments

Each set of test results is formatted as follows:

```
Test start Fri Mar 24 09:51:20 1995
Call complete time 2.2
Loss          Near:    1.0 dB          Far:    0.1 dB
Noise         Near:   18.0 dBrnC       Far:   15.5 dBrnC
Noise/Tone    Near:   35.3 dBrnC       Far:   36.5 dBrnC
Slope 404     Near:    1.0 dB          Far:    0.7 dB
Slope 1004    Near:    0.6 dB          Far:    0.4 dB
Slope 2804    Near:    2.4 dB          Far:    1.1 dB
ERL           Near:   39.0 dB          Far:   39.5 dB
SRL           Near:   39.0 dB          Far:   39.5 dB
SRH           Near:   39.0 dB          Far:   39.5 dB
Test complete
```

Examples

The following line will call a milliwatt test line (TL102) on interface 1, channel 7, dialing 9500022 DTMF after waiting for a wink:, and perform a loss test.

```
nearend 1 7 -2 -d w9500022 -t loss
```

The following line calls a responder (TL105) on interface 2, channel 22, and performs all tests. Subsequent trunks to test are in the file trunks. The dialing sequence is

1. Go off hook
2. Wait for a wink
3. Dial DTMF the trunk number, followed by a pound sign.
4. Pause.
5. Dial MF the sequence Kp105St.

```
nearend 2 22 -5 -d w -d @trunks -d # -m ~k105s
```

See also

mdp105(Sage)

A1-8 CU (LOCAL)

Name

cu - call another UNIX/XENIX system

Syntax

```
cu [-s speed] [-l line] [-h] [-t] [-xn] [-o | -e | -oe] [-n] telno
cu [-s speed] [-h] [-xn] [-o | -e | -oe] -l line [dir]
cu [-h] [-xn] [-o | -e | -oe] systemname
```

Description

The cu command calls up another UNIX system, a terminal, or possibly a non-UNIX system. It manages an interactive conversation with possible transfers of ASCII files.

The cu command accepts the following options and arguments:

- | | |
|-----------|---|
| : -sspeed | Specifies the transmission speed (150, 300, 600, 1200, 2400, 4800, 9600, 19200,38400). The default value is "Any" speed which will depend on the order of the lines in the /usr/lib/uucp/Devices file. A speed range can also be specified (for example, -s1200-4800). |
| -lline | Specifies a device name to use as the communication line. This can be used to override the search that would otherwise take place for the first available line having the right speed. When the -l option is used without the -s option, the and -s options are both used together, cu will search the Devices file to check if the requested speed for the requested line is available. If so, the connection will be made at the requested speed; otherwise, an error message will be printed and the call will not be made. The specified device is generally a directly connected asynchronous line (for example, /dev/ttyab) in which case a telephone number (telno) is not required. The specified device need not be in the /dev directory. If the specified device is associated with an auto dialer, a telephone number must be provided. Use of this option with systemname rather than telno will not give the desired result (see systemname below). |
| -h | Emulates local echo, supporting calls to other computer systems which expect terminals to be set to half-duplex mode. |
| -t | Used to dial an ASCII terminal which has been set to auto answer. Appropriate mapping of carriage-return to carriage-return-line-feed pairs is set. |
| -xn | Causes diagnostic traces to be printed; it produces a detailed output of the program execution on stderr. The debugging level, n, is a single digit in the range 0 to 9; -x9 is the most useful value. |
| -n | For added security, -n will prompt the user to provide the telephone number to be dialed rather than taking it from the command line. |

telno When using an automatic dialer, the argument is the telephone number with equal signs for secondary dial tone or minus signs placed appropriately for delays of 4 seconds.

systemname A UUCP system name may be used rather than a telephone number. In this case, cu will obtain an appropriate direct line or telephone number from /usr/lib/uucp/Systems. Note: the systemname option should not be used in conjunction with the -l and -s options as cu will connect to the first available line for the system name specified, ignoring the requested line and speed.

dir The keyword dir can be used with cu -lline, in order to talk directly to a modem on that line, instead of talking to another system via that modem. This can be useful when debugging or checking modem operation. Note: only users with write access to the Devices file are permitted to use cu-lline dir.

In addition, cu uses the following options to determine communications settings:

-o If the remote system expects or sends 7-bits with odd parity.

-e If the remote system expects or sends 7-bits with even parity.

-oe If the remote system expects or sends 7-bits, ignoring parity and sends 7-bits with either parity.

By default, cu expects and sends 8-bit characters without parity. If the login prompt received appears to contain incorrect 8-bit characters, or a correct login is rejected, use the 7-bit options described above.

After making the connection, cu runs as two processes: the transmit process and the receive process. The transmit process reads data from standard input and, except for lines beginning with "~", passes the data to the remote system. The receive process accepts data from the remote system and, except for lines beginning with "~", passes the data to standard output.

Normally, an automatic XON/XOFF protocol is used to control input from the remote system so the buffer is not overrun.

Lines beginning with "~" have special meanings.

The transmit process interprets the following user-initiated commands:

~. terminate the conversation.

~! escape to an interactive shell on the local system.

~!cmd . . . run cmd on the local system (via sh -c).

~\$cmd . . . run cmd locally and send its output to the remote system.

~+cmd . . . run cmd on the local system but take standard input from the remote system.

~%cd	change the directory on the local system. Note: ~!cd will cause the command to be run by a sub-shell, probably not what was intended.
~% ~%	take from [to] copy file from (on the remote system) to file to on the local system. If to is omitted, the from argument is used in both places.
~% ~%	put from [to] copy file from (on the local system) to file to on the remote system. If to is omitted, the from argument is used in both places.
	For both ~%take and ~%put commands, as each block of the file is transferred, consecutive single digits are printed to the terminal.
~~line	send the line ~line to the remote system.
~%break	transmit a BREAK to the remote system (which can also be specified as ~%b).
~%debug	toggles the -x debugging level between 0 and 9 (which can also be specified as ~%d).
~t	prints the values of the termio structure variables for the user's terminal (useful for debugging).
~l	prints the values of the termio structure variables for the remote communication line (useful for debugging).
~%nostop	toggles between XON/XOFF input control protocol and no input control. This is useful in case the remote system is one which does not respond properly to the XON and XOFF characters.

The use of ~%put requires stty(C) and cat(C) on the remote side. It also requires that the current erase and kill characters on the remote system be identical to these current control characters on the local system. Backslashes are inserted at appropriate places.

The use of ~%take requires the existence of echo(C) and cat(C) on the remote system. Also, tabs mode (see stty(C)) should be set on the remote system if tabs are to be copied without expansion to spaces.

The receive process normally copies data from the remote system to its standard output. It may also direct output to local files.

You can construct take and put commands that work between UNIX and non-UNIX systems by sending the appropriate characters to cu. To do this, you will need to know the equivalent of echo(C) and cat(C) on the non-UNIX system. For example, to transfer a file named fred from a remote non-UNIX system to the file /tmp/fred on the local UNIX system, construct a command similar to the following:

```
~%  
echo '~>':/tmp/fred  
cat fred  
echo '~>'
```

This creates a file /tmp/fred on the local UNIX system, putting the characters “~>” into it, which tells cu to start receiving data into this file. The file fred is then sent to standard output on the remote machine, and cu therefore receives it. Finally, a “~>” is echoed into the file; this is a signal to cu to stop receiving input. (Remember to replace echo and cat with the equivalent commands for the non-UNIX system.)

You can also append the file from the remote machine to an existing file on the local system:

```
~%  
echo '~>>':/tmp/fred  
cat fred  
echo '~>'
```

This appends the remote file onto the end of the existing file /tmp/fred.

When cu is used on system1 to connect to system2 and subsequently used on system2 to connect to system3, commands on system2 can be executed by using “~~”. Executing a tilde command reminds the user of the local system uname. For example, uname can be executed on systems 1, 2, and 3 as follows:

```
uname  
system3  
~!uname  
system1  
~~!uname  
system2
```

In general, “~” causes the command to be executed on the original machine, and “~~” causes the command to be executed on the next machine in the chain.

Examples

To dial a system whose telephone number is 9 201 555 1212 using 1200 baud (where dialtone is expected after the 9):

```
cu -s1200 9=12015551212
```

If the speed is not specified, “Any” is the default value.

To login to a system connected by a direct line:

```
cu -l /dev/ttyXX or cu -l ttyXX
```

To dial a system with the specific line and a specific speed:

```
cu -s1200 -l ttyXX
```

To dial a system using a specific line associated with an auto dialer:

```
cu -l ttyXX 9=12015551212
```

To call up a system named huey:

```
cu huey
```

To talk directly to an ACU (connect directly with the modem and enter modem commands manually):

```
cu -l ttyXX dir
```

Files

```
/usr/lib/uucp/Systems  
/usr/lib/uucp/Devices  
/usr/lib/uucp/LCK..(tty-device)
```

See also

cat(C), ct(C), echo(C), stty(C), uucp(C), uname(C)

Diagnostics

Exit code is zero for normal exit, otherwise, one.

Warnings

The cu command does not do any integrity checking on data it transfers. Data fields with special cu characters may not be transmitted properly. Depending on the interconnection hardware, it may be necessary to use a “~.” to terminate the conversion even if stty0 has been used. Non-printing characters are not dependably transmitted using either the ~%put or ~%take commands. cu between an IMBR1 and a penril modem will not return a login prompt immediately upon connection. A carriage return will return the prompt.

Note

There is an artificial slowing of transmission by cu during the ~%put operation so that loss of data is unlikely.

Standards conformance

cu is conformant with:

AT&T SVID Issue 2; and X/Open Portability Guide, Issue 3, 1989.

APPENDIX II

FAR-END ON-LINE MANUAL PAGES

A2-1 FARDEM (SAGE)

fardem(C) 03 May 1998 fardem(C)

Name

fardem - daemon for farend responder (type 105 testline).

Syntax

fardem ckt-name [-l 0 | 1 | 2]

Description

fardem creates an application daemon for a type 105 testline on the specified DS0 as indicated by ckt-name (a circuit defined in the 950chan database). One daemon must be launched for each desired testline (DS0).

The fardem daemons may be launched automatically using the farend script ("farend start"), which will create a daemon for each circuit in the 950chan database with a port name of "tl105" or "tl105w". A "tl105" will be started as a normal 105 testline; a "tl105w" will be initialized in Wilcom Emulation (DTMF) mode.

NOTE: Startup Error Messages may show up ONLY in the log file; If the daemon doesn't start, and no obvious reason was displayed to standard out, examine the log file.

Messages are logged to fardem.log in the /u/950rts/log directory.

fardem accepts the following options:

- x Same as -l (Debug Flag)
- l logging level. Controls what is logged in the log file:
 - 0 log only errors. Exclude information and debugging messages. Error messages are generated when the far end application cannot run; If a needed test resource cannot be allocated, for example. Some errors are FATAL, that is, the fardem daemon can not recover, and is stopped. Other errors are non-fatal, such as the above resource allocation error; the daemon does something to recover (like wait a while and try again).

Appendix II Far-end Manual Pages

- 1 log errors and informational messages. Information messages provide useful information on the operation of the daemon. When a fardem daemon is launched, an information message is logged indicating the DSO id and process id of the daemon. This is the default logging level.
 - 2 Add debug messages. Debug messages provide detailed information on the daemons operation, detailing each step that is accomplished. This would normally only be used if a problem is suspected.
 - 3 Add far end measurement messages; This shows various details of what the farend DSP function thought it measured; This is a development only feature (not nec. implemented at any given time).
- t Sets TPT Timeout, in milliseconds.
- c Sets 105 Command Timeout value, in milliseconds.
- w Force fardem to initialize the test line in Wilcom Emulation mode, irrespective of port name.
- h Causes fardem to issue a "usage" message.

Errors

If a test resource cannot be allocated at the time fardem is launched, the daemon will NOT be started, and an error message issued to standard out AND the log file. If the specified circuit cannot be allocated because it is busy or in an alarm state, the daemon will be started, and will periodically retry the circuit. If the circuit cannot be allocated because the ckt-name parameter specifies a port that does not exist in the 950Chan database, the daemon will NOT be started, and an error message issued to standard out AND the log file.

If after the daemon has started the circuit or test resource become unavailable, an error message is logged to the log file. The daemon will simply wait for a time (nominally 30 seconds) and try again, until it receives the needed resource. Durring this process, no response is made to attempted seizure on the circuit, as the daemon is not connected to the DSO and thus cannot detecet seizure.

In general, error or info messages from the daemon go only to the log file.

Examples

To create a a far end application daemon for DSO described in spanid as "tl105" and "tl105Alt", the following could be used:

```
% fardem tl105
    % fardem tl105Alt -l 0
```

The daemon launched for DS0 tl105Alt would log only error messages to the log file. The daemon launched for DS0 tl105 would log both error and informational messages, as level 1 is the default.

Files

/u/950rts/log/fardem.log
/u/950rts/etc/fardem

See also

farend (C)

Author

Benjamin A. Rolfe, Sage Instruments Inc.

Standards Conformance

fardem is not part of any currently supported standard; it is part of 950RTS Test System Software (C) Sage Instruments, Inc, All Rights Reserved.

A2-2 FAREND (SAGE)

farend(C)

03 March 1998

farend(C)

Name

farend - script used to start, stop, and/or status fardem daemons

Syntax

farend [start|stop|restart|stat]

Description

start: creates application (fardem) daemons for a type 105 testlines; A type 105 testline is a circuit (channel) named in the "950chan" database by the name "tl105" or "tl105w". A daemon is launched for each DS0 so designated. If the circuit is named "tl105" it is initialized to start in "105 Mode", meaning it will expect the MF command sequences of a type 105 ROTL farend. If the circuit is named "tl105w" the farend will be initialized to start in "WILCOM Emulation Mode", meaning it will expect the DTMF command set (see "fardem" man page for details).

Will NOT stop any existing fardem daemons; see restart.

stop: searches for any "fardem" daemons running and stops them (via a kill signal).

restart: stop followed by start;

stat: displays the current status of any far end application daemons running on the system. The status display includes the process ID of the daemon, the time it was started, the CPU time it has consumed, the has consumed, the name of the application (fardem) and the DS0 circuit name for which it was started.

Examples

To start fardem daemons on all named as 105 testlines:

```
% farend start
```

To stop all fardem daemons:

```
% farend stop
```

To stop all fardem daemons and restart:

```
% farend restart
```

To determine how many farend daemon (fardem) processes are running:

% farend stat

UID	PID	Start Time	CPU Time	DS0-id
admin	20952	17:14:37	0:04	fardem tl105
admin	20955	17:14:37	0:02	fardem tl105w

Files

/u/950rts/log/fardem.log

See also

fardem (C)

Author

Benjamin A. Rolfe, Sage Instruments

Standards conformance

farend is part of 950RTS Test System Software (C) Sage Instruments, Inc, All Rights Reserved.

Type 105 testline functions are defined by: AT&T Compatibility Bulletin No. 106

APPENDIX III
SCL ON-LINE MANUAL PAGES

A3-1 CALL PROGRESS (SCL)

Name

callprogress - monitor line status

Syntax

callprogress

Description

The "callprogress" command monitors a switched circuit. The current connection must be a DS0 -- either a direct connection, or a DCS TAD.

Once the test is running, it is controlled using the standard SCL test controls and reporting commands: "stoptest", "starttest", "clrtest", "report", and "showreport".

Installed Parameters

None.

Result Value

When the test is started successfully, an empty result string is returned, otherwise a description of the error is returned.

The test results returned by the reporting commands consists of a single value, which represents the current status of the line. The following values may be returned:

unknown	unknown signal
deadline	dead-line (> 4 s of silence)
dialtone	dial-tone (steady 350/440 2-tone)
dialtone400	dial-tone (steady 400 tone)
busy	busy-line (.5 s on/off 480/620 2-tone)
reorder	reorder (.25 s on/off 480/620 2-tone)
ringing	ringing (440/480 2-tone; 2s on/4 s off)
TPT	2225 Hz TPT
TPT-CDMA	1804 Hz TPT CDMA
TPT-TDMA	1404 Hz TPT TDMA
TPTend	end of TPT
digits	touch tone digit

chime	MCI Chime tone
bong	AT&T Bong tone
beepbeep	Beep Beep (2 bursts of 600 Hz tone)
singletone	steady single tone
dualtone	steady dual tone
speech	maybe speech (unrecognized 0-, 3-, or 4-tone)
SIT	Special Information Tone

Examples

The following example shows setting up a call and monitoring the call progress:

```
scl<1> connect r101
scl<2> hook off
scl<3> sdigits dtmf 12345
scl<4> callprogress
scl<5> showreport
deadline
deadline
<DEL>
interrupted
scl<6>
```

See Also

sdigits(SCL), hook(SCL), selecttest(SCL), starttest(SCL), stoptest(SCL), clrttest(SCL),
report(SCL)

A3-2 CLRTEST (SCL)

Name

clrtest - clear results for the currently running test

Syntax

clrtest

Description

The "clrtest" command clears any accumulated measurements in the currently running 950RTS test. Not all 950RTS tests support the "clrtest" command.

Measurements are cleared automatically when a test is stopped and restarted.

Result Value

None.

Examples

The following example shows clearing DS0 BERT test results in an interactive session.

```
scl<1> connect r101
scl<2> ds0bert
scl<3> showreport
elapsed      status      netcode      errs      ber
1            sync        89           0          <1.79E-05
2            sync        ac           0          <8.93E-06
3            sync        d9           1          <5.95E-06
<DEL>
interrupted
scl<4> clr
scl<5> showrep
elapsed      status      netcode      errs      ber
1            sync        89           0          <1.79E-05
2            sync        ac           0          <8.93E-06
<DEL>
interrupted
scl<6>
```

See Also

starttest(SCL), showreport(SCL), ds0bert(SCL), ds1bert(SCL), ds1monitor(SCL)

A3-3 CELLCMPNDR (SCL)

Name

cellcmpndr - perform a cellular compander test

Syntax

cellcmpndr

Description

This command performs a cellular cmpndr test, which is basically a set of three loss tests at 1004 Hz at three different power levels: -24, -18, and -12 dBm.

Result Value

The command will return tags and values for both near and far loss measurements at the three frequencies.

Examples

```
cellcmpndr near_24 1.7 far_24 -4.0 near_18 -1.1 far_18 -6.9 near_18 -  
3.4 far_18 5.1
```

See Also

A3-4 CELLCONNECT (SCL)

Name

cellconnect - connect to a cellular responder

Syntax

```
cellconnect -i [y|n] -f [mf|dtmf] -d [digit_string]
```

Description

This command will call a cellular responder and wait for its Test Progress Tone, TPT. The -i option controls looking for a dial-tone, yes or no. The -f option determines if MF or DTMF digits are send for the -d digit string. See sdigits(SCL) for valid digits. All the parameters are really optional. The dial-tone option, -i, defaults to yes, and the digit format option, -f, defaults to dtmf.

Result Value

The command will return the time it took to see TPT if successful; otherwise it will return the error for the call incompleation.

Examples

```
scl<1> cellconnect -i y -f dtmf -d 4823041 8.6
```

See Also

A3-5 CELLDEVL (SCL)

Name

celldevl - perform a cellular deviation limiting test

Syntax

celldevl

Description

This function will perform a deviation limiting test .

Result Value

If a test is successful, the deviation limiting will be reported in kHz.

Examples

scl<1> celldevl 10.7

See Also

A3-6 CELLFILTER (SCL)

Name

cellfilter - perform a cellular filter test

Syntax

cellfilter

Description

This command performs a cellular filter test, which is a set of three loss tests at three frequencies: 404, 1004, and 2396 Hz.

Result Value

The command will return tags and values for both near and far loss measurements at the three frequencies.

Examples

```
scl<1> cellfilter near_404 1.7 far_404 -4.0 near_1004 -1.1 far_1004 -  
6.9 near_2396 -3.4 far_2396 5.1
```

See Also

A3-7 CELLGET (SCL)

Name

cellget - get information from a cellular responder

Syntax

cellget [calltype|channel|phonenum|respver|truver|resptype|esn]

Description

The cellget command asks the cellular responder for various pieces of information. In particular one could for the type of call, the channel number, the phone number of the responder, the responders software version, the phone's software version, the responder type, or the esn, respectively.

Result Value

For the calltype, the command will return 0 for an amps call and 1 for an namps or tdma call depending on the responder. For the channel option, the function will return the channel number and slot, if appropriate, for the current call. For the resptype, the function will return the model number followed by the call type. The rest of the parameters listed should be obvious.

Examples

```
scl<1> cellget resptype 357 amps scl<2> cellget calltype 0 scl<3> cellget  
channel 724 scl<4> cellget phonenum 4086720805 scl<5> cellget respver  
2.04 scl<6> cellget truver11 scl<7> cellget esn 16400016775
```

See Also

A3-8 CELLRSSI (SCL)

Name

cellrssi - ask responder for current RSSI

Syntax

cellrssi

Description

This function will ask the responder for its receive signal strength.

Result Value

The RSSI will be returned in dbm.

Examples

```
scl<1> cellrssi -53.0
```

See Also

A3-9 CELLSECTOR (SCL)

Name

cellsector - control the sector scanning and sector locking of a cellular responder

Syntax

cellsector [add|clearerr|erasedb|listall|listerrs|lock|scan] value

Description

This command controls the sector scanning and sector locking features of a 366 responder. For sector scanning, one can add sectors to the sector database, erase the sector database, or list all of the sector database. In addition one can list the errored sectors as well as clear those errors. Obviously, one can turn sector scanning on or off. For sector locking, one can lock to a particular channel as well as tell the phone to lock to that channel and be amps only or dual mode.

Result Value

The command will return the appropriate values.

Examples

```
scl<1> cellsector clearerrs scl<2> cellsector erasedb scl<3> cellsector  
add 351 scl<4> cellsector add 352 scl<5> cellsector listall 0351 0352  
scl<6> cellsector listerrs 0352  
scl<7> cellsector lock 351 amps
```

See Also

A3-10 CELLSET (SCL)

Name

cellset - set a parameter related to cellular communication

Syntax

```
cellset [phone_mode|protocol|conn_type] value
```

Description

This function will set a parameter related to cellular communication. In the case of phone_mode, it is used to tell the responder to become amps only or dual mode with the values amps or dual. Both protocol and conn_type parameters affect the 950's communication to the responder. The protocol value is either bc for backwards compatibility or an for AckNack. The conn_type option requires a value of amps, tdma_is54, tdma_is36, namps, and unknown.

Result Value

None.

Examples

None.

See Also

A3-11 CONNECT (SCL)

Name

connect - create a 950RTS test connection

Syntax

connect [circuit]

Description

The "connect" command allocates the named circuit, reserves a test resource, and creates a test connection. With no arguments, "connect" returns information about the current test connection.

The "circuit" argument can be specified in the following ways:

port 950RTS test port listed in the interface database (IDB). Use "listports" to show all ports.

span numeric id number from the IDB (DS1 testing)

span chan span number and channel number(DS0 testing)

A test connection is torn down using the "disconnect" command, or automatically when SCL exits.

SCL supports a single test connection at a time. Therefore, a "disconnect" must occur before a new test connection can be made.

Result Value

When a connection is made with no errors, an empty result string is returned, otherwise a description of the error is returned.

Test connection information is returned as a string of type/value pairs. The following are returned:

port	connected 950RTS test port
type	type of circuit (ds0rbs, ds0clear, etc.)
board	board number
intfc	interface number
chan	channel number

If no test connection exists, "none" is returned in the result string.

Examples

The following example shows interactively setting up and tearing down a 950RTS test connection.

```
scl<1> connect r101
scl<2> connect
port r101 type ds0rbs board 0 intfc 0 chan 1
scl<3> disconnect
scl<4> connect
none
scl<5>
```

See Also

disconnect(SCL)

A3-12 DCS (SCL)

Name

dcx - control digital cross-connect

Syntax

```
dcx
dcx open [name]
dcx close
dcx monitor <ds3> <ds1> <ds0> [tap]
dcx [split|release|tapstat [tap]]
dcx side [equip|facility]
```

Description

The "dcx" command provides the ability to connect to a digital cross-connect system (DCS), to monitor, split, or release the cross-connected circuit, and to switch testing between equipment and facility directions. The DCS must be configured in the 950RTS DCS map file, and the DCS test ports must be represented as TAPs or FADs in the 950RTS interface database.

With no arguments, the "dcx" command returns the status of the current DCS connection, or "none" if no connection has been made.

The "open" option is used to establish a connection to the 950RTS DCS manager process for the desired DCS. The DCS name is determined from the interface database, based on the DCS TAP or FAD selected with the "connect" command. The "close" option will close the connection to the DCS manager.

When a "connect" is performed on a DCS test port, the 950RTS interface manager assigns a test access path (TAP). The TAP identifier is used internally for subsequent commands. It is also possible to override the TAP on the command line.

The "monitor", "split", and "release" options manage the currently allocated (or explicitly specified) TAP. The cross-connected circuit to test is specified to the "monitor" command as three circuit identifiers, representing the DS3, DS1, and DS0. Zero is substituted for any unused parameter; for example, the DS0 value would be 0 for DS1 testing, and the DS3 value would be zero on a 1/1 or 1/0 DCS. DCS's with unusual circuit representations may require different conventions for specifying the circuit to test.

The current status of the circuit under test is obtained using the "tapstat" option.

The "side" option allows setting the direction of testing -- to the forward ("equip"), or reverse ("facility") directions. With no arguments, this option returns the current test direction.

Result Value

The "dcs" option returns the "name" and "type" of the opened DCS.

The "tapstat" option returns the state of the circuit under test; "monitor", "split", or "release".

The "side" option returns the direction of testing: "equip" or "facility".

All other options return an empty result if successful, otherwise an error message.

Examples

The following example shows connecting to a DCS circuit, setting test direction, and running a DS0 BERT test.

```
scl<1> connect dcsClearTap
scl<2> dcs open
scl<3> dcs
name dcs1 type hadax
scl<4> dcs tapstat
name 1-16-1 status release
scl<5> dcs monitor 1 1 1
scl<6> ds0bert
scl<7> showreport
elapsed      status      netcode      errs      ber
1            sync        89           0          <1.79E-05
2            sync        ac           0          <8.93E-06
3            sync        d9           0          <5.95E-06
<DEL>
interrupted
scl<5> dcs split
scl<6> ds0bert
scl<7> showreport
elapsed      status      netcode      errs      ber
1            sync        89           0          <1.79E-05
2            sync        ac           0          <8.93E-06
<DEL>
interrupted
scl<8> dcs side
equip
scl<9> dcs side facility
scl<10> ds0bert
scl<11> showreport
elapsed      status      netcode      errs      ber
1            sync        89           0          <1.79E-05
2            sync        ac           0          <8.93E-06
<DEL>
interrupted
scl<12> stop
scl<13> dcs release
```

```
scl<14> dcs close  
scl<15>
```

See Also

connect(SCL), ds0bert(SCL), showreport(SCL), stoptest(SCL)

A3-13 DISCONNECT (SCL)

Name

disconnect - tear down a 950RTS test connection

Syntax

disconnect

Description

The "disconnect" command tears down the current test connection, releasing the allocated circuit and test resource.

Since SCL does not support multiple test connections, changing test connections requires a "disconnect" followed by a "connect" to the new circuit.

"disconnect" is invoked automatically when SCL exits.

NOTE: the "expect" disconnect command has been renamed "detach" so as not to conflict with the SCL "disconnect" command.

Result Value

None.

Examples

The following example shows interactively setting up and tearing down a 950RTS test connection.

```
scl<1> connect r101
scl<2> connect
port r101 type ds0rbs board 0 intfc 0 chan 1
scl<3> disconnect
scl<4> connect
none
scl<5>
```

See Also

connect(SCL)

A3-14 DS0BERT (SCL)

Name

ds0bert - run DS0 bit error rate (BERT) testing

Syntax

```
ds0bert [pattern]
        [-format <fmt>] [-subrate <rate>]
        [-upattern <bits>] [-echo on|off]
```

Description

The "ds0bert" command starts a DS0 BERT test on the current test connection. The current connection must be a DS0 -- either a direct connection, or a DCS TAP.

Test parameters are loaded from the "ds0bert" installed SCL parameters, which are overridden by any command-line parameters.

The "pattern" parameter is positional, all others are flag arguments.

Once the test is running, it is controlled using the BERT control commands: "ds0inject", and "ds0loop". The standard SCL test controls and reporting commands are also available: "stoptest", "starttest", "clrtest", "report", and "showreport".

A running test is stopped using the "stoptest" command, or automatically when SCL exits. SCL supports a single test at a time. Therefore, a "stop" must be issued before a new test can be run.

Installed Parameters

pattern	test pattern 511, 2047, 2e15, 2e20, 2e23 1:7, user, dds1, dds2, dds3, dds3r, dds4, dds5, dds5a
format	data format ds0a, 64kb, sw56kb
subrate	data subrate (DS0A format only) 56kb, 19.2kb, 9.6kb, 4.8kb, 2.4
upattern	user pattern 3 to 8 binary digits
echo	echo canceller on, off

Result Value

When the test is started successfully, an empty result string is returned, otherwise a description of the error is returned.

The test results returned by the reporting commands consist of the following measurement types:

elapsed	test time (seconds)
status	test status (nosync, sync)
errs	bit error counter
ber	bit error rate
sls	sync loss seconds
efs, pctefs	error-free seconds, percent EFS
es, pctes	errored seconds, percent ES
ses, pctses	severely errored seconds, percent SES
fail, pctfail	failed seconds, percent failed
avail, pctavl	available seconds, percent available
unavl, pctunvl	unavailable seconds, percent unavailable
data	PCM sample
netcode	received network code (or second PCM sample)

Examples

The following example shows starting a DS0 BERT test interactively, stopping, setting parameters, then restarting.

```
scl<1> connect r101
scl<2> ds0bert 2047
scl<3> showreport
elapsed      status      netcode      errs      ber
1            sync        89           0          <1.79E-05
2            sync        ac           0          <8.93E-06
3            sync        d9           0          <5.95E-06
<DEL>
interrupted
scl<4> stop
scl<5> showparams ds0bert
echo          off
format        ds0a
pattern       2047
subrate       56kb
scl<8> ds0bert 511 -echo on
```

This example shows scripting use of the "ds0bert" command.

```
#!/u/950rts/bin/scl

# make DS0 test connection
```

```
set port c124
puts "DS0 BERT testing on port $port"
scl_connect $port

# start BERT test
scl_ds0bert 2047

# retrieve results report and extract desired measurements
setarray res [scl_report]
if {$res(status) == "nosync"} {
puts "NOSYNC condition"
} else {
puts "BIT ERRS: $res(errs)"
}
```

See Also

```
ds0inject(SCL), ds0loop(SCL), dcs(SCL)
setparam(SCL), showparams(SCL),
selecttest(SCL), starttest(SCL), stoptest(SCL), clrtest(SCL),
report(SCL), showreport(SCL), ds1bert (SCL)
```

A3-15 DS0INJECT (SCL)

Name

ds0inject - inject errors during DS0 BERT testing

Syntax

ds0inject <type>

Description

The "ds0inject" command injects errors into a running DS0 BERT test. Use the "ds0bert" command to start the test.

The error type is configurable via the installed "injectttype" parameter, or by command-line.

Installed Parameters

injectttype	type of error injection bit, 6bit
-------------	--------------------------------------

Result Value

When the command succeeds, an empty result is returned, otherwise a description of the error is returned.

Examples

The following example shows starting a DS0 BERT test and injecting 2 bit errors, followed by 6 bit errors:

```
scl<1> connect r101
scl<2> ds0bert
scl<3> ds0inject; ds0inject
scl<4> showreport
elapsed      status      netcode      errs      ber
7            sync        2f           2          5.10E-06
8            sync        c8           2          4.46E-06
<DEL>
interrupted
scl<5> ds0inject 6bit
scl<6> report elapsed status errs
elapsed 12 status sync errs 8
```

Appendix III SCL Manual Pages

```
scl<5> showparams ds0inject  
injecttype      6bit  
scl<6>
```

See Also

ds0bert(SCL), ds0loop(SCL), report(SCL), showreport(SCL), showparams(SCL)

A3-16 DS0LOOP (SCL)

Name

ds0loop - loopback control during DS0 BERT testing

Syntax

ds0loop up|down [<type> [-unit #]]

Description

The "ds0loop" command sends loopup or loopdown codes during "ds0bert" test.

The loop type is configurable via the installed "loopcode" parameter, or by command-line argument. The unit or repeater number can be set using the "unit" installed parameter or command line option.

Installed Parameters

loopcode	type of loop csualt, dsualt, hl96alt, ocualt, ocun96alt, csulat, ds0dplat, hl222lat, oculat, first56alt, secnd56alt
unit	number of unit or repeater

Result Value

If the command succeeds, an empty result is returned, otherwise a description of the error is returned.

Examples

The following example shows starting a DS0 BERT test then initiating a CSU latching loopback, and then tearing it down:

```
scl<1> connect r101
scl<2> ds0bert
scl<3> showreport elapsed status
elapsed      status
2            nosync
3            nosync
<DEL>
interrupted
scl<4>      ds0loop up csulat
scl<5>      showreport
elapsed    status      netcode    errs      ber
10         sync        2f        0         5.10E-06
11         sync        c8        0         4.46E-06
```

```
<DEL>
interrupted
scl<6> ds0loop down
scl<7> showparams ds0loop
loopcode      csulat
unit          0
scl<8>
```

See Also

ds0bert(SCL), ds0loop(SCL), showreport(SCL), report(SCL), showparams(SCL)

A3-17 DS1BERT (SCL)

Name

ds1bert - run DS1 bit error rate (BERT) testing

Syntax

ds1bert [pattern]

Description

The "ds1bert" command starts a DS1 BERT test on the current test connection. The current connection must be a DS1 -- either a direct connection, or a DCS FAD.

The "pattern" test parameter is loaded from the installed parameters, or from the command-line positional parameter.

Once the test is running, it is controlled using the BERT control commands: "ds1inject", and "ds1loop". The standard SCL test controls and reporting commands are also available: "stoptest", "starttest", "clrtest", "report", and "showreport".

A running test is stopped using the "stoptest" command, or automatically when SCL exits. SCL supports a single test at a time. Therefore, a "stop" must be issued before a new test can be run.

Installed Parameters

pattern	test pattern
	QRSS, 511, 2047, 2e15, 2e20, 2e23, 1:7, 1:1, 3in24, ones, hex55, long54, 55daly, long72, long96, long120, user
upattern	user pattern
	3 to 24 binary digits

Result Value

When the test is started successfully, an empty result string is returned, otherwise a description of the error is returned.

The test results returned by the reporting commands consist of the following measurement types:

elapsed	test time (seconds)
status	test status (nosync, sync)
errs	bit error counter
ber	bit error rate
sls	sync loss seconds

efs, pctefs	error-free seconds, percent EFS
es, pctes	errored seconds, percent ES
ses, pctses	severely errored seconds, percent SES
fail, pctfail	failed seconds, percent failed
avail, pctavl	available seconds, percent available
unavl, pctunvl	unavailable seconds, percent unavailable
t1err	T1 error type
t1ebits	summary T1 error bits
nopcm	boolean no PCM indicator
frmloss	boolean frame loss indicator
frmcrc	boolean frame or CRC error indicator
bpvdetect	boolean BPV detection indicator
excess0s	boolean excess zeroes indicator
alarms	boolean alarm indicator
b8zs	boolean B8ZS detect indicator
fslipdetect	boolean frame slip indicator
bpv	BPV counter
frm	frame error counter
crc	CRC error counter
oof	out-of-frame counter
bslip	bit slip counter
fslip	frame slip counter

Examples

The following example shows setting up a DS1 circuit for testing, then starting an interactive DS1 BERT test.

```
scl<1> connect t1
scl<2> intfc setup sf ami
scl<3> ds1bert 2047
scl<4> showreport
elapsed      status      t1err      errs      ber
1            sync        noerrs     0          <1.79E-05

2            sync        noerrs     0          <8.93E-06
3            sync        noerrs     0          <5.95E-06
<DEL>
interrupted
scl<5>
```

This example shows scripting use of the "ds1bert" command.

```
#!/u/950rts/bin/scl

# make DS1 test connection
set port t1
puts "DS1 BERT testing on port $port"
scl_connect $port
```

```
# start BERT test
scl_ds1bert 2047

# retrieve results report and extract desired measurements
setarray res [scl_report]
if {$res(status) == "nosync"} {
  puts "NOSYNC condition"
} else {
  puts "BIT ERRS: $res(errs)"
}
```

See Also

```
ds1inject(SCL), ds1loop(SCL), ds1monitor(SCL), intfrc(SCL), dcs(SCL),
setparam(SCL), selecttest(SCL), starttest(SCL), stoptest(SCL),
clrtest(SCL), report(SCL)
```

A3-18 DS1INJECT (SCL)

Name

ds1inject - inject errors during DS1 BERT testing

Syntax

```
ds1inject [type [count]]
```

Description

The "ds1inject" command injects errors into a running DS1 BERT test. Use the "ds1bert" command to start the test.

The error type and count are configurable via the installed parameters or by command-line.

Installed Parameters

injecttype	type of error injection bit, bpv, frame, crc
injcount	number of errors to inject

Result Value

When the command succeeds, an empty result is returned, otherwise a description of the error is returned.

Examples

The following example shows starting a DS1 BERT test and injecting 2 bit errors, followed by 10 frame errors:

```
scl<1> connect r101
scl<2> intfc setup sf ami
scl<3> ds1bert
scl<4> ds1inject; ds1inject
scl<5> showreport
elapsed    status    netcode    errs    ber
7          sync      2f         2       5.10E-06
8          sync      c8         2       4.46E-06
<DEL>
interrupted
scl<6> ds1inject frame 5
scl<7> report elapsed status errs frm
elapsed    status    errs    frm
12         sync      2       5
scl<8> showparams ds1inject
injecttype    frm
```

```
injcount      5
scl<9> ds1inject
scl<10> !report
elapsed      status      errs      frm
12           sync        2          10
scl<11>
```

See Also

```
ds1bert(SCL), ds1loop(SCL), ds1monitor(SCL), showreport(SCL),
report(SCL), showparams(SCL)
```

A3-19 DS1LOOP (SCL)

Name

ds1loop - loopback control during DS1 BERT testing

Syntax

ds1loop up|down [<type> [<user-code>]]

Description

The "ds1loop" command sends loopup or loopdown codes during a "ds1bert" test.

The loop type is configurable via installed parameter, or by command-line argument. A user-defined loopback code can be used by setting the loop type to "user", and setting the code in the "uloop" installed parameter or on the command-line.

Installed Parameters

t1loop	type of loop csu, fac1, fac2, user
uloop	user loop code 3 to 7 binary digits

Result Value

If the command succeeds, an empty result is returned, otherwise a description of the error is returned.

Examples

The following example shows starting a DS1 BERT test then initiating a CSU loopback, and then tearing it down:

```
scl<1> connect t1
scl<2> ds1bert
scl<3> showreport elapsed status
elapsed      status
2            nosync
3            nosync
<DEL>
interrupted
scl<4> ds1loop up csu
scl<5> showreport
elapsed      status      netcode      errs      ber
10          sync        2f         0        5.10E-06
11          sync        c8         0        4.46E-06
```

```
<DEL>
interrupted
scl<6> ds1loop down
scl<7> showparams ds1loop
t1loop          csu
uloop           0
scl<8>
```

See Also

```
ds1bert(SCL), ds1inject(SCL), report(SCL), showreport(SCL),
showparams(SCL)
```

A3-20 DS1MONITOR (SCL)

Name

ds1monitor - run DS1 line monitor test

Syntax

ds1monitor

Description

The "ds1monitor" command monitors a DS1 circuit, collecting T1 line errors. The current connection must be a DS1 -- either a direct connection, or a DCS FAD.

Once the test is running, it is controlled using the standard SCL test controls and reporting commands: "stoptest", "starttest", "clrtest", "report", and "showreport".

A running test is stopped using the "stoptest" command, or automatically when SCL exits. SCL supports a single test at a time. Therefore, a "stop" must be issued before a new test can be run.

Installed Parameters

None.

Result Value

When the test is started successfully, an empty result string is returned, otherwise a description of the error is returned.

The test results returned by the reporting commands consist of the following measurement types:

elapsed	test time (seconds)
t1err	T1 error type
t1ebits	summary T1 error bits
nopcm	boolean no PCM indicator
frmloss	boolean frame loss indicator
frmcrc	boolean frame or CRC error indicator
bpvdetect	boolean BPV detection indicator
excess0s	boolean excess zeroes indicator
alarms	boolean alarm indicator
b8zs	boolean B8ZS detect indicator
fslipdetect	boolean frame slip indicator
bpv	BPV counter
frm	frame error counter
crc	CRC error counter
oof	out-of-frame counter

bslip	bit slip counter
fslip	frame slip counter

Examples

The following example shows setting up a DS1 circuit for testing, then starting a DS1 monitor test.

```
scl<1> connect t1
scl<2> intfc setup sf ami
scl<3> ds1monitor
scl<4> showreport
elapsed    t1err    frm      crc      bpv    excess0s    b8zs
1          frmloss  226      67      0      off         off
3          frmloss  2478     729     0      off         off
4          frmloss  3603     1060    0      off         off
5          frmloss  4729     1391    0      off         off
<DEL>
interrupted
scl<5>
```

See Also

ds1bert(SCL), intfc(SCL), dcs(SCL), selecttest(SCL), starttest(SCL),
stoptest(SCL), clrtest(SCL), report(SCL)

A3-21 EXIT (SCL)

Name

exit, quit - terminate SCL

Syntax

exit
quit

Description

The "exit" command stops any running test, tears down the current test connection, and exits SCL, returning the user to the UNIX shell.

The "quit" command is synonymous with "exit".

Result Value

None.

See Also

connect(SCL), disconnect(SCL)

A3-22 HOOK (SCL)

Name

hook - set or query origination hook switch state

Syntax

hook [on|off]

Description

The "hook" command sets the origination hook switch off or on. With no arguments, the command returns the current state.

Installed Parameters

None.

Result Value

When the hook state is successfully set, an empty result is returned, otherwise a description of the error is returned.

Examples

The following example shows setting off-hook, querying status, then going on-hook:

```
scl<1> connect r101
scl<2> hook
on
scl<3> hook off
scl<4> hook
off
scl<5> hook on
scl<6>
```

See Also

sdigits(SCL), callprogress(SCL), connect(SCL), dcs(SCL)

A3-23 IMD (SCL)

Name

imd - run Intermodulation Distortion testing

Syntax

```
imd [tone] [level]
    [-tone <t>] [-level <l>]
```

```
imtone [tone]
```

Description

The "imd" command starts a DS0 BERT test on the current test connection. The current connection must be a DS0 -- either a direct connection, or a DCS TAP.

Test parameters are loaded from the "imd" installed SCL parameters, which are overridden by any command-line parameters.

The "tone" and "level" parameters are positional, and can also be invoked as flag arguments.

Once the test is running, it is controlled using "imtone" command. The "imtone" command with no arguments returns the tone currently being sent. The "tone" argument will change the tone being sent.

The standard SCL test controls and reporting commands are also available: "stoptest", "starttest", and "report".

A running test is stopped using the "stoptest" command, or automatically when SCL exits. SCL supports a single test at a time. Therefore, a "stoptest" must be issued before a new test can be run.

Installed Parameters

tone	test tone (2tone, 4tone, off)
level	output tone level

Result Value

When the test is started successfully, an empty result string is returned, otherwise a description of the error is returned.

The test results returned by the "getreport" and "report" commands consist of the following measurement types:

status	test status (ok, notone, spurtone)
tone	received tone (2tone, 4tone, off)
type	test phase (snr, corr, uncorr)
imd2, imd3	IMD products
level	received tone level

Examples

The following example shows starting an IMD test interactively.

```
scl<1> connect r101
scl<2> imd
scl<3> showreport
status      tone      type      imd2      imd3      level
ok          2tone     snr       47.3      50.0      -14.0
ok          2tone     snr       47.4      50.1      -14.1
<DEL>
interrupted
scl<4> showparams imd
level       -16.0
tone        2tone
scl<5> imtone 4tone
scl<6> showreport
status      tone      type      imd2      imd3      level
ok          4tone     corr      47.7      51.2      -16.0
ok          4tone     corr      46.2      47.5      -16.1
ok          4tone     corr      48.3      50.9      -16.1
ok          4tone     corr      48.5      50.9      -16.0
<DEL>
interrupted
scl<7>
```

This example shows scripting use of the "imd" command.

```
#!/u/950rts/bin/scl

# make DS0 test connection
set port c124
puts "IMD testing on port $port"
scl_connect $port

# run 2-tone, then 4-tone test
scl_imd 2tone
setarray results [scl_report]
puts "2-Tone Test: IMD2 $results(imd2)   IMD3 $results(imd3)"
```

```
scl_imtone 4tone
setarray results [scl_report]
puts "4-Tone Test: IMD2 $results(imd2)    IMD3 $results(imd3)"
}
```

See Also

```
setparam(SCL),  selecttest(SCL),  starttest(SCL),  stopptest(SCL),
report(SCL)
```

A3-24 INTFC (SCL)

Name

intfc - configure and show status of DS1 interface

Syntax

```
intfc [setup <framing> <coding> <clock>]
intfc [setup [-framing <f> -coding <c> -clock <cl>]]
```

Description

The "intfc" command will setup the framing and coding on the currently connected circuit. The current connection must be a DS1 -- either a direct connection, or a DCS FAD.

The "framing" and "coding" parameters are loaded from the installed parameters, or from command-line positional parameters.

With no arguments, the "intfc" command will return the current T1 configuration.

Installed Parameters

framing	line framing none, esf, sf
coding	line coding ami, b8zs, b7stuff
clock	clock source internal, looptime

Result Value

If the setup is successful, an empty result is returned, otherwise a description of the error is returned.

The status is returned as type/value pairs, containing the following:

```
framing
coding
monitor
length
imped
clock
idlecode
```

Examples

The following example shows setting up a DS1 circuit for testing, then starting an interactive DS1 BERT test.

```
scl<1> connect t1
scl<2> intfc setup sf ami
scl<3> intfc
framing sf coding ami monitor none length 0-133 impd 100ohm
clock looptime
autob8 off idlecode 7f
scl<4>
```

See Also

ds1bert(SCL), dcs(SCL) setparam(SCL)

A3-25 JITTER (SCL)

Name

jitter - run Phase/Amplitude Jitter testing

Syntax

```
jitter [type] [filter]
        [-freq <f>] [-level <l>]
```

Description

The "jitter" command starts 23 Tones testing on the current test connection. The current connection must be a DSO -- either a direct connection, or a DCS TAP.

Test parameters are loaded from the "jitter" installed SCL parameters, which are overridden by any command-line parameters.

Both the "type" and "level" parameters are positional, and can also be invoked as flag arguments.

Once the test is running the standard SCL reporting commands are available.

A running test is stopped using the "stoptest" command, or automatically when SCL exits. SCL supports a single test at a time. Therefore, a "stoptest" must be issued before a new test can be run. The test can be restarted using "start".

Installed Parameters

type	"send" (send tone) or "meas" (measure only)
filter	measurement filter ("low" or "high")
freq	output tone frequency (20 to 3904 hz) if type is "send"
level	output tone level (+3.0 to -60 db)

Result Value

When the test is started successfully, an empty result string is returned, otherwise a description of the error is returned.

The full test results consist of the following measurement types:

status	test status ("ok", "notone", or "spurtone")
phase	phase jitter (degrees)
amp	amplitude jitter (%)
level	receive tone power (dbm)
freq	receive frequency (hz)

Examples

The following example shows starting a jitter test interactively.

```
scl<1> connect r101
scl<2> showparams jitter
filter          low
freq            1004
level           -16.0
type            send
scl<3> jitter -level -18
scl<4> showreport
status    phase    amp    freq    level
ok         0.8      1.6    1004    -16.1
ok         0.8      1.6    1004    -16.1
ok         0.9      1.6    1004    -16.1
ok         0.8      1.6    1004    -16.1
<DEL>
interrupted
scl<5> stop
```

This example shows scripting use of the "jitter" command.

```
#!/u/950rts/bin/scl

# make DS0 test connection
set port c124
puts "Phase/Amplitutde jitter testing on port $port"
scl_connect $port

# start BERT test
scl_jitter

# display jitter when tone comes on
while { 1 } {
    setarray results [scl_report]
    if {$results(status) == ok} {
        puts "Phase jitter: $results(phase) degrees"
        puts "Amplitude jitter: $results(amp) %"
        return
    }
}
```

See Also

```
setparam(SCL), selecttest(SCL), starttest(SCL), stoptest(SCL),
report(SCL)
```

A3-26 MTONE (SCL)

Name

mtone - run measure tone test

Syntax

mtone

Description

The "mtone" command starts tone measurement on the current test connection. The current connection must be a DS0 -- either a direct connection, or a DCS TAP.

Once the test is running, it is controlled using the standard SCL test controls and reporting commands: "stoptest", "starttest", "clrtest", "report", and "showreport".

A running test is stopped using the "stoptest" command, or automatically when SCL exits. SCL supports a single test at a time. Therefore, a "stoptest" must be issued before a new test can be run.

Installed Parameters

None.

Result Value

If the test is started successfully, an empty result string is returned, otherwise a description of the error is returned.

The test results returned by the reporting commands consist of the following measurement types:

freq	tone frequency (hz)
level	tone level (dbm)

Examples

The following example shows starting a measure tone test interactively, and then stopping the test.

```
scl<1> connect r101
scl<2> mtone
scl<3> showreport
freq      level
1333      -60.0
1333      -60.0
1333      -60.0
```

interrupted
scl<4> stop
scl<5>

A3-27 NAP (SCL)

Name

nap - suspend execution for short interval

Syntax

nap <msec>

Description

The "nap" command suspends the SCL process for approximately the number of milliseconds specified on the command line.

This command is intended for scripting usage.

Installed Parameters

None.

Result Value

None.

See Also

sleep command in expect(1), update(SCL)

A3-28 NOISE (SCL)

Name

noise - run noise test

Syntax

noise [type]

Description

The "noise" command starts return loss measurement on the current test connection. The current connection must be a DS0 -- either a direct connection, or a DCS TAP.

The "type" test parameter is loaded from the installed parameters, or from the command-line positional parameter.

Once the test is running, it is controlled using the standard SCL test controls and reporting commands: "stoptest", "starttest", "clrtest", "report", and "showreport".

A running test is stopped using the "stoptest" command, or automatically when SCL exits. SCL supports a single test at a time. Therefore, a "stoptest" must be issued before a new test can be run.

Installed Parameters

type	noise measurement type (cnotch, cmsg, snr, flat3k)
------	---

Result Value

If the test is started successfully, an empty result string is returned, otherwise a description of the error is returned.

The reporting commands return a single measurement; the noise measurement, in dBnC (cnotch and cmsg), dB (snr), or dBn (flat3k).

Examples

The following example shows starting a noise test interactively, and then stopping the test.

```
scl<1> connect r101
scl<2> noise cnotch
scl<3> showreport
13.0
13.0
13.0
```

```
interrupted
scl<4> stop
scl<5>
```

A3-29 PAR (SCL)

Name

par - run Peak/Average Ratio test

Syntax

```
par [type] [level]
par [-type <t>] [-level <l>]
```

Description

The "par" command starts PAR testing on the current test connection. The current connection must be a DS0 -- either a direct connection, or a DCS TAP.

Test parameters are loaded from the "par" installed SCL parameters, which are overridden by any command-line parameters.

Both the "type" and "level" parameters are positional, and can also be invoked as flag arguments.

Once the test is running the standard SCL reporting commands are available.

A running test is stopped using the "stoptest" command, or automatically when SCL exits. SCL supports a single test at a time. Therefore, a "stoptest" must be issued before a new test can be run. The test can be restarted using "starttest".

Installed Parameters

type	"send" (send 23 tones) or "meas" (measure only)
level	output tone level (+3 to -60 db), if type is "send"

Result Value

When the test is started successfully, an empty result string is returned, otherwise a description of the error is returned.

The full test results consist of the following measurement types:

status	test status ("ok", "notone", or "spurtone")
units	PAR units
level	signal power (dbm)

Examples

The following example shows starting a 23 tones test interactively.

```
scl<1> connect r101
scl<2> showparams par
level          -16.0
type           send
scl<3> par -level -18
scl<4> showreport
status         units         level
ok             99            -18.1
ok             99            -18.1
ok             99            -18.1
<DEL>
interrupted
scl<4> stop
```

This example shows scripting use of the "par" command.

```
#!/u/950rts/bin/scl

# make DS0 test connection
set port c124
puts "PAR testing on port $port"
scl_connect $port

# start BERT test
scl_par

# display PAR units when tone comes on
while { 1 } {
  setarray results [scl_report]
  if {$results(status) == ok} {
    puts "PAR Units: $results(units)"
    puts "Signal/Total Distortion: $results(stdr)"
    return
  }
}
```

See Also

```
selecttest(SCL), starttest(SCL), stoptest(SCL), report(SCL),
setparam(SSCL)
```

A3-30 REPORT (SCL)

Name

report - return results report from a running test

Syntax

report [type1 type2 ... typeN]

Description

The "report" command requests a results report from the test resource. If no "type" argument is given, all measurement types for the test are returned. See the manual page for each test to determine the measurements collected for that test.

For interactive use, the "showreport" command provides a continuous display of results

Result Value

The "report" command returns a list containing type/value pairs.

Examples

The following example shows use of the "report" command from a DS0 BERT test, while running interactively.

```
scl<1> connect r101
scl<2> ds0bert
scl<3> report elapsed status netcode errs ber
elapsed 1 status status netcode e3 errs 0 ber <1.79E-05
scl<4>
```

See Also

starttest(SCL), showreport(SCL), ds0bert(SCL), ds1bert(SCL), mtone(SCL), noise(SCL), rloss(SCL)

A3-31 RLOSS (SCL)

Name

rloss - run return loss test

Syntax

rloss [type]

Description

The "rloss" command starts return loss measurement on the current test connection. The current connection must be a DSO -- either a direct connection, or a DCS TAP.

The "type" test parameter is loaded from the installed parameters, or from the command-line positional parameter.

Once the test is running, it is controlled using the standard SCL test controls and reporting commands: "stoptest", "starttest", "clrtest", "report", and "showreport".

A running test is stopped using the "stoptest" command, or automatically when SCL exits. SCL supports a single test at a time. Therefore, a "stoptest" must be issued before a new test can be run.

Installed Parameters

type return loss type (erl, srl, srh)

Result Value

If the test is started successfully, an empty result string is returned, otherwise a description of the error is returned.

The reporting commands return a single measurement; the return loss value, in dB.

Examples

The following example shows starting a return loss test interactively, and then stopping the test.

```
scl<1> connect r101
scl<2> rloss erl
scl<3> showreport
.1
.1
.2
interrupted
scl<4> stop
scl<5>
```

A3-32 ROTLATD (SCL)

Name

rotlatd - perform an extended 105 attenuation-distortion test.

Syntax

rotlatd

Description

This command performs an extended 105 attenuation-distortion test (e.g. frequency sweep), which is a set of 16 tests at the following frequencies: 304, 404, 504, 604, 804, 1004, 1404, 1804, 2204, 2404, 2504, 2604, 2704, 2804, 3004, 3204.

Result Value

The command will return tags and values for both near and far loss measurements at the 16 frequencies.

Examples

```
scl<1> rotlatd near_304 14.6 far_304 7.3 near_404 12.3 far_404 6.4
near_504 11.1 far_504 5.7 near_604 10.5 far_604 5.6 near_804 10.0 far_804
5.4 near_1004 10.6 far_1004 4.3 near_1404 11.1 far_1404 6.1 near_1804
12.4 far_1804 6.7 near_2204 12.8 far_2204 8.2 near_2404 13.1 far_2404
8.8 near_2505 13.1 far_2504 9.4 near_2604 13.0 far_2604 10.1 near_2704
13.3 far_2704 10.8 near_2804 13.5 far_2804 11.4 near_3004 13.2 far_3004
12.5 near_3204 13.3 far_3204 14.2
```

See Also

A3-33 ROTLGAIN (SCL)

Name

rotlgain - perform an extended 105 gain slope test.

Syntax

rotlgain

Description

This command performs an extended 105 gain slope test, which is a set of three loss tests at three frequencies: 404, 1004, and 2804 Hz.

Result Value

The command will return tags and values for both near and far loss measurements at the three frequencies.

Examples

```
scl<1> rotlgain near_404 1.7 far_404 -4.0 near_1004 -1.1 far_1004 -6.9  
near_2804 -3.4 far_2804 5.1
```

See Also

A3-34 ROTLIMD (SCL)

Name

rotlimd - perform an extended 105 intermodulation distortion test

Syntax

rotlimd

Description

This command performs an extended 105 intermodulation distortion test.

Result Value

The command will return tags and values for both near and far measurements.

Examples

```
scl<1> rotlimd near_2nd 70 near_3rd 70 far_2nd 44 far_3rd 54
```

See Also

A3-35 ROTLIMP (SCL)

Name

rotlimp - perform an extended 105 impulse noise test

Syntax

rotlimp

Description

This command performs an extended 105 impulse noise test.

Result Value

The command will return tags and values for both near and far measurements.

Examples

```
scl<1> rotlimp near_low 0.0 near_mid 0.0 near_high 0.0 near_phs 0  
near_gain 0 near_drop 0 far_low 0.0 far_mid 0.0 far_high 0.0 far_phs 0  
far_gain 0 far_drop 0
```

See Also

A3-36 ROTLJIT (SCL)

Name

rotljit - perform an extended 105 jitter test

Syntax

rotljit [type]

Description

This command performs an extended 105 jitter test. The supported types are hi and lo.

Result Value

The command will return tags and values for both near and far measurements phase and amplitude jitter.

Examples

```
scl<1> rotljit hi near_phas 5.0 near_amp 7.3 far_phas 4.3 far_amp 4.9  
scl<1> rotljit lo near_phas 5.0 near_amp 7.3 far_phas 4.3 far_amp 4.9
```

See Also

A3-37 ROTLLOSS (SCL)

Name

rotlloss - perform a 105 loss test

Syntax

rotlloss

Description

This command performs a 105 loss test.

Result Value

The command will return tags and values for both near and far loss measurements.

Examples

```
scl<1> rotlloss near_1004 -1.1 far_1004 -6.9
```

See Also

A3-38 ROTLNOISE (SCL)

Name

rotlnoise - perform a 105 noise test.

Syntax

rotlnoise [type]

Description

This command performs a 105 noise test of the specified type. The supported types are cmsg, cnotch, s/n, and 3k.

Result Value

The command will return tags and values for both near and far noise measurements.

Examples

None.

A3-39 ROTLPAR (SCL)

Name

rotlpar - perform an extended 105 parametric test

Syntax

rotlpar

Description

This command performs an extended 105 parametric test.

Result Value

The command will return tags and values for both near and far measurements.

Examples

```
scl<1> rotlpar near 90 far 57
```

See Also

A3-40 ROTLRLOSS (SCL)

Name

rotlrloss - perform a 105 return loss test.

Syntax

rotlrloss [type]

Description

This command performs a 105 return loss test of the specified type. The supported types are erl, srl, and srh for echo return loss, singing return loss low, and singing returnloss high, respectively.

Result Value

The command will return tags and values for both near and far return loss measurements.

Examples

A3-41 SDIGITS (SCL)

Name

sdigits - send DTMF or MF digits

Syntax

```
sdigits <fmt> <digits>
```

Description

The "sdigits" command outpulses DTMF or MF digits over a test connection. The current connection must be a DS0 -- either a direct connection, or a DCS TAP.

The digit format and digits are set via command-line. The required arguments are:

fmt	dtmf, mf
digits	valid digit string (0-9a-cks*#)

Installed Parameters

None.

Result Value

When the digits are successfully sent, an empty result is returned, otherwise a description of the error is returned.

Examples

The following example shows sending a tone, turning it off, and querying the tone sender status:

```
scl<1> connect r101
scl<2> hook off
scl<3> sdigits dtmf 12354
scl<5>
```

See Also

hook(SCL), callprogress(SCL), connect(SCL), dcs(SCL)

A3-42 SELECTTEST (SCL)

Name

selecttest - set the current SCL test

Syntax

selecttest [test]

Description

The "selecttest" command sets the current SCL test selection. A subsequent "starttest" command will begin running the currently selected test. The test selection is set automatically when a test command is invoked. Use "help" to see the list of available test commands.

Result Value

If the test is valid, an empty result string is returned, otherwise an error message is returned.

With no arguments, the "selecttest" command returns the name of the currently selected test, or an empty result if there is no test selection.

Examples

The following example shows selecting a DS0 BERT test interactively and then starting the test, and then displaying the test selection.

```
scl<1> connect r101
scl<2> select
scl<3>
scl<4> select ds0bert
scl<5> start
scl<6> showreport
elapsed      status      netcode      errs          ber
1            sync        89           0             <1.79E-05
2            sync        ac           0             <8.93E-06
3            sync        d9           0             <5.95E-06
<DEL>
interrupted
scl<7> select
ds0bert
scl<8>
```

See Also

starttest(SCL), connect(SCL), ds0bert(SCL), ds1bert(SCL), mtone(SCL),
noise(SCL), rloss(SCL)

A3-43 SETPARAM (SCL)

Name

setparam - set or return current values of installed parameters

defparam - return default values of installed parameters

showparams - show current or default values of installed parameters

Syntax

setparam command param [value]

defparam command param

showparams [-def] [command [param]]

Description

Most SCL commands have configurable settings which are specified via command-line or through "installed parameters". The "setparam" command manages these installed parameters. Each installed parameter has an initial, default setting, which can be viewed or restored using the "defparam" command.

If a "value" argument is given to "setparam", the parameter value is changed; otherwise the value is returned. "defparam" returns the parameter value.

For interactive use, the "showparams" command produces a formatted display of current parameter values. With no arguments, "showparams" displays all installed SCL parameters. If a "command" argument is specified, "showparam" displays all installed parameters for that command. If the "-def" flag is used, the default parameter values are displayed instead of the current values.

NOTE: if a test parameter is changed while a test is running, it will not take effect until the test has been stopped and restarted.

The installed parameters for each command are listed on its manual page.

Result Value

If a parameter is set successfully, an empty result is returned, otherwise an error message is returned that indicated why the parameter setting failed.

If a single parameter is retrieved, its value is returned in the result string.

"showparams" send displays to standard output and returns an empty result string.

Examples

The following example shows setting DS0 BERT test parameters, then restoring them to defaults.

```
scl<3> showparams ds0bert
echo             off
format           ds0a
pattern          511
subrate          56kb
scl<2> setparam ds0bert pattern 2047
scl<2> setparam ds0bert echo on
scl<3> ds0bert
...
scl<4> stop
scl<5> defparam ds0bert pattern
511
scl<6> setparam ds0bert pattern 511
scl<7> start
```

See Also

ds0bert(SCL), ds1bert(SCL), mtone(SCL), noise(SCL), rloss(SCL),

A3-44 SETVAR (SCL)

Name

setvar - set or return current values of SCL installed variables

showvars - show current values of SCL installed variables

Syntax

```
setvar var [value]
```

```
showvars
```

Description

SCL provides a set of general configuration and status variables that are managed using the "setvar" command. Some of these variables are read-only, others can be modified by the user.

If a "value" argument is supplied to "setvar", the variable is changed. With no "value" argument, the variable value is returned.

For interactive use, the "showvars" command will display all available SCL variables.

The currently available variables are:

debugflag	boolean, turns on debug mode
testrunning	boolean, read-only, indicates if a test is currently running
dacsequip	boolean, used by "dcs" command

Result Value

If a variable is set successfully, an empty result is returned, otherwise an error message is returned describing why the setting failed.

If a single variable is retrieved, its value is returned in the result string.

"showvars" displays to standard output, and returns an empty result.

Examples

The following example shows displaying and setting SCL variables:

```
scl<1> showvars
debugflag      0
testrunning    1
dacsequip      0
scl<2>
```

```
setvar debugflag 1
scl<3> setvar debugflag
1
scl<4>
```

See Also

ds0bert(SCL), ds1bert(SCL), mtone(SCL), noise(SCL), rloss(SCL)

A3-45 STARTTEST (SCL)

Name

starttest - start the currently selected SCL test

Syntax

starttest

Description

The "starttest" command starts running the currently selected 950RTS test on the current test connection, using the current settings of the installed test parameters. The test connection is set up using the "connect" command.

Usually, a test is started by invoking it by name. Use the "help" command to list all available SCL tests. Running the test will make it the currently selected test. Once selected, the test can be restarted after a "stoptest" by using the "starttest" command.

Alternatively, a test can be selected using the "selecttest" command, the test parameters can be set using the "setparam" command, and the test can be started using the "starttest" command.

A running test is stopped using the "stoptest" command, or automatically when SCL exits. SCL supports a single test at a time. Therefore, a "stop" must be issued before a new test can be run.

The SCL boolean variable "testrunning" indicates whether a test is currently running. Use the "setvar" command to display this variable. Use the "selecttest" command to determine the currently running test.

Result Value

If the test is started successfully, an empty result string is returned, otherwise a description of the error is returned.

Examples

The following example shows starting a DS0 BERT test interactively, stopping, setting parameters, then restarting.

```
scl<1> connect r101
scl<2> select ds0bert
scl<3> start
scl<4> showreport
elapsed    status    netcode    errs    ber
1          sync      89         0       <1.79E-05
```

Appendix III SCL Manual Pages

```
2          sync      ac          0          <8.93E-06
3          sync      d9          0          <5.95E-06
<DEL>
interrupted
scl<4> stop
scl<5> showparams ds0bert
echo          off
format        ds0a
pattern       2047
subrate       56kb
scl<6> setparam ds0bert pattern 511
scl<7> setparam ds0bert echo on
scl<9> setvar testrunning
0
scl<8> start
scl<9> setvar testrunning
1
scl<10>
```

See Also

```
selecttest(SCL), stoptest(SCL), setparam(SCL), setvar(SCL), ds0bert(SCL),
ds1bert(SCL), mtone(SCL), noise(SCL), rloss(SCL)
```

A3-46 STONE (SCL)

Name

stone - control 950RTS tone sender

Syntax

```
stone [off | on [freq [level]]]
```

Description

The "stone" command controls and reports status of the 950RTS tone sender.

The "on" option of the "stone" command will turn the tone sender on, setting the frequency and level from installed parameters or from the command-line.

The "off" option will turn off the tone sender.

With no arguments, the "stone" command returns status of the tone sender.

Installed Parameters

freq	tone frequency (in Hz)
level	tone level (in dBm)

Result Value

When the tone sender is successfully turned on or off, an empty result is returned, otherwise a description of the error is returned.

Querying the tone sender status returns the frequency and level of the tone currently being sent, using the standard SCL reporting format (type-value pairs). If no tone is being sent, "off" is returned.

Examples

The following example shows sending a tone, turning it off, and querying the tone sender status:

```
scl<1> connect r101
scl<2> stone
off
scl<3> setparam stone
freq          1004
level         -16.0
scl<4> stone on
scl<5> stone
freq 1004 level -16.0
scl<4> stone on 55
freq 55 level -16.0
scl<6> stone off
scl<7>
```

See Also

mtone(SCL), connect(SCL), setparam(SCL)

A3-47 STOPTEST (SCL)

Name

stoptest - stop the currently running SCL test

Syntax

stoptest

Description

The "stoptest" command stops the currently running 950RTS test. A test is started using a test command, or using "selecttest" followed by the "starttest" command. Use "help" for a list of available tests.

A running test is stopped automatically when SCL exits. Since SCL supports a single test at a time, a "stoptest" must be issued before a new test can be run.

The SCL boolean variable "testrunning" indicates whether a test is currently running. Use the "setvar" command to display this variable. Use the "selecttest" command to determine the currently running test.

Result Value

None.

Examples

The following example shows starting and stopping a DS0 BERT test interactively.

```
scl<1> connect r101
scl<2> ds0bert
scl<3> showreport
elapsed      status      netcode      errs      ber
1            sync        89           0          <1.79E-05
2            sync        ac           0          <8.93E-06
3            sync        d9           0          <5.95E-06
<DEL>
interrupted
scl<4> setvar testrunning
1
scl<5> stop
scl<6> setvar testrunning
0
scl<7> start
scl<8> setvar testrunning
1
```

```
scl<9> select  
ds0bert  
scl<10>
```

See Also

```
starttest(SCL), setvar(SCL), connect(SCL)
```

A3-48 TONE23 (SCL)

Name

tone23 - run 23 Tones testing

Syntax

```
tone23 [type] [level]
tone23 [-type <t>] [-level <l>]
```

Description

The "tone23" command starts 23 Tones testing on the current test connection. The current connection must be a DSO -- either a direct connection, or a DCS TAP.

Test parameters are loaded from the "tone23" installed SCL parameters, which are overridden by any command-line parameters.

Both the "type" and "level" parameters are positional, and can also be invoked as flag arguments.

Once the test is running the standard SCL reporting commands are available.

A running test is stopped using the "stop" command, or automatically when SCL exits. SCL supports a single test at a time. Therefore, a "stop" must be issued before a new test can be run. The test can be restarted using "start".

Installed Parameters

```
type          "send" (send 23 tones) or "meas" (measure only)
level         output tone level (+3 to -60db), if type is "send"
```

Result Value

When the test is started successfully, an empty result string is returned, otherwise a description of the error is returned.

The full test results consist of the following measurement types:

stdr	signal/total distortion (ratio)
snr	signal/noise (ratio)
level	signal power (dbm)
imd2, imd3	IMD products
loss_203 ... loss3641	23 attenuations (dbm)
delay_281 ... delay 3563	22 envelope delays (ms)

The standard results subset for this command (shown by the "showreport" command) is:

```
stdr level loss_203 delay_281 loss_1766 delay_1844 loss_3641
```

Examples

The following example shows starting a 23 tones test interactively.

```
scl<1> connect r101
scl<2> showparams tone23
level          -16.0
type           send
scl<3> tone23 -level -18
scl<4> showreport
stdr    level    loss_203  delay_281  loss_1766  delay_1844  loss_3641
21.4     -18.3     -30.0      0         -30.0      0          -30.0
35.9     -18.1     -29.7      0         -29.7     -1          -29.7
35.9     -18.1     -29.7      0         -29.7     -1          -29.7
35.9     -18.1     -29.7      0         -29.7     -1          -29.7
<DEL>
interrupted
scl<4> stop
```

This example shows scripting use of the "tone23" command.

```
#!/u/950rts/bin/scl

# make DS0 test connection
set port c124
puts "23 Tones testing on port $port"
scl_connect $port

# start BERT test
scl_tone23

# retrieve results into an array and extract desired measurements
setarray results [scl_report]
puts "Signal/Noise: $results(snr)"
puts "Signal/Total Distortion: $results(stdr)"
}
```

See Also

```
setparam(SCL), selecttest(SCL), starttest(SCL), stoptest(SCL),
report(SCL)
```

A3-49 TRACE (SCL)

Name

trace - print SCL commands to standard output

Syntax

trace on | off

Description

The "trace" command provides the ability to turn on SCL command tracing. It is intended use in debugging scripts.

Result Value

None.

See Also

update(SCL)

A3-50 UPDATE (SCL)

Name

update - respond to 950RTS daemon processes

Syntax

update

Description

The "update" command responds to queries from the 950RTS manager processes. The update occurs automatically whenever a SCL command is invoked. Explicit invocation is only needed when a script spends more than 15 minutes executing non-SCL TCL commands (e.g. lengthy text processing).

If SCL process does not respond to the 950RTS resource managers within the timeout period (currently 15 minutes), the managers assume the process has died, and they will release the test resource and circuit used by the process.

Installed Parameters

None.

Examples

This example demonstrates use of "update" in a script:

```
#!/u/950rts/bin/scl

# make test connection
scl_connect r101

# long loop processing a text file
set file [open "trunkfile" r]
while {[eof $file] == 0} {
  set line [gets $file]
  puts $line

# notify the 950RTS managers of our continued existence
scl_update
}
```

Result Value

None.

APPENDIX IV

SAGE 950RTS COMMANDS

A4-1 TEST TOOL COMMANDS

- atrc [start | stop]** - Command used to start or stop atr daemon
- listlimits** - Displays all test limits filter files.
- listports** - Displays all ports defined in spanid file.
- listschedules** - Displays all schedules in schedule database.
- listsites** - Displays all sites listed in Sites directory.
- mdp105** - Perform auto-routining with dialing via DSC maintenance dial plan. (See man page.)
- nearend** - Do a near-end 100/102/105 test-line test, including signaling and dialing. (See man page.)
- printlimits <limits>** - Displays contents of limits file.
- printschedule <schedule>** - Displays contents of a schedule.
- printsites <site>** - Displays the content of a site file.
- runlimits <results> [<filter>]** - Runs test results through a test limits filter.
- sched** - Schedule management for **950RTS** automatic trunk routining. (See man page.)
- show950opts** - Displays enabled options on **950RTS**.
- timegroups [all | <grp1> [-grp2]] [<trunkfile>]** - Displays an estimate of test duration by trunk group size.
- vfctest** - Voice frequency test utility.
- vsched** - Visual ATR scheduling utility.

A4-2 AUTOMATED DAEMONS

- atrd** - Sage **950RTS** automatic trunk routining daemon. (See man page.)
- dacsd**-Demon for the Digital Cross Connect System.

fardem - Daemon for farend responder (type 105 test-line). (See man page.)

A4-3 ADMINISTRATION COMMANDS AND UTILITIES

atr - Sage **950RTS** automatic trunk routing utility. (See man page.)

backup_lib [floppy] - Archives **/u/950rts/lib** user files in cpio format.

countdspd - Displays name, base address, offset, interrupt vector, DMA, # of boards, and firmware release of all DSP boards.

counttrunks [all | <grp1>[-grp2]] [<trunkfile>] - Displays all specified trunk groups, the # of trunks in each group, and the total # of trunks

db - Sage **950RTS** database management utility. (See man page.)

dbrecord - Sage **950RTS** database record formats. (See man page.)

farend - Script used to start, stop, and/or status fardem daemons for all DS0's. (See man page.)

get950opts - Display all installed options in the **950RTS** options file.

listgroups [all | <grp1>[-grp2]] [<trunkfile>] - Displays trunks in specified trunk groups

loaddsp - Loads DSP driver.

logrts <module> <message> - Logs messages from shell scripts. Module = db, atrd, vs, test, trem, t1im, fardem, vftest, berttest, msg.

rz - Receive Zmodem.

swlink - Switch link trunk database extraction utility. (See **Appendix I** of the *950RTS Installation and Administration Guide* .)

sz - Send Zmodem.

t1config - T1 board status and configuration utility. (See **Sec. 3-4** of the *950RTS Installation and Administration Guide*.)

t1im [start | stop] - Start or stop the T1 interface manager.

trem [start | stop] - Start or stop the test resource manager.

A4-4 SCL COMMANDS

A4-4.1 Test Access

connect *port*- Connects test access.
disconnect- Disconnect test access.
help- Show help messages.
exit- Quit.
quit- Quit.
version *-all*- Returns the SCL version number.

A4-4.2 VF and Data Tests

absdelay *level*- Starts the Absolute Delay test.
callprogress- Starts Call Progress monitoring.
edd- Starts Envelope Delay test.
imd *tone level*- Starts the Intermodulation Distortion test.
imnoise- Starts the Impulse Noise test.
jitter *type filter*- Starts the Phase/Amplitude Jitter test.
mtone - Starts the Tone test.
noise *type*- Starts the Noise test.
par *type level*- Starts the Peak/Average Ratio test.
rloss *type*- Starts the Return Loss test.
23tones *type level*- Starts the 23 Tones test.

A4-4.3 BERT and DS1 Tests

ds0bert *rate pattern upattern*- Starts the DS0 Bit Error Rate test.
ds0inject *injecttype*- Injects errors during DS0 BERT test.
ds0loop up | down *loopcode*- Controls DS0 BERT loopback.
ds1bert *pattern upattern*- Starts the DS1 Bit Error Rate test.
ds1inject *type count*- Injects errors during DS1 BERT test.
ds1loop up | down *t1loop uloop*- Controls loopback during the DS1 BERT test.
ds1monitor- Starts the T1 line status monitor.

A4-4.4 Test Controls

clrtest- Clears counters in a running test.
selecttest *test*- Selects or returns currently selected test.
starttest- Starts the selected test.
stoptest- Stops a running test.

A4-4.5 General Controls

dcs option- Controls the Digital Cross Connect System.
echocan freq level mse- Sends the echo canceller disable tone.
farhook- Gets the termination hook switch state.
hook on / off- Sets or returns hook state.
intfc setup framing coding clock- Sets or gets the TI interface configuration.
radio option- Connects the remote audio monitor to a test circuit.
report result1 result2...resultN- Returns the current test results.
sdigits digits dtmf / mf- Sends digits.
setparam command param value- Sets or returns the value of installed command parameters.
setvar variable value- Sets or returns the current value of installed variable.
stone on / off freq level- Controls the tone sender.
vfloop on / off delay scale- Controls VF loopback.

A4-4.6 ATME Tests

atme- initiates an ATME test, controlling the ATME director function.
showatme- runs the ATME command, and displays the results in a tabular form.

A4-4.7 ROTL Tests

rotlconnect digits dtmf / mf- Calls a 105 responder and wait for TPT.
rotlatd- Performs an Extended 105 Attenuation Distortion test.
rotlgain- Performs an Extended 105 Gain Slope test.
rotlimd- Performs an Extended 105 Intermodulation Distortion test.
rotlimp- Performs an Extended 105 Impulse Noise test.
rotljit- Performs an Extended 105 Jitter test.
rotlloss- Performs a 105 Loss test.
rotlnoise type- Performs a 105 Noise test.
rotlpar- Performs an Extended 105 Peak to Average Ratio test.
rotlrloss type- Performs a 105 Return Loss test.

A4-4.8 Cellular Tests

cellconnect digits dtmf / mf- Connects to a cellular responder.
cellatd- Performs a cellular Attenuation Distortion test.
cellcmpndr- Performs a cellular Compander test.
celldevl- Performs a cellular Deviation Limiting test.
cellfar get | set option value- Gets/sets information in cellular responder.
cellfilter- Performs a Cellular Filter test.
cellgain- Performs a Cellular Gain Slope test.
cellimd- Performs a Cellular Intermodulation Distortion test.
cellimp- Performs a Cellular Impulse Noise test.
celljit- Performs a Cellular Jitter test.

celloss- Performs a Cellular Loss test.

cellnoise *type*- Performs a Cellular Noise test.

rotlpar- Performs a Cellular Peak to Average Ratio test.

cellrloss *type*- Performs a Cellular Return Loss test.

A4-4.9 Display Functions

showreport *count result1 result2...resultN*- Continuously displays standard results report for the currently running test.

showparams **-def** *command*- Displays the current or default values of installed command parameters.

showvars- Displays the current values of installed variables.

A4-4.10 Scripting Functions

nap *msec*- Delays for an interval.

trace on | off-Prints SCL commands to standard output. Used for debugging scripts.

update- Responds to the 950RTS manager processes.

APPENDIX V

MAKEDB COMMAND SCRIPTS

A5-1 MAKEDB

The following script, **makedb**, is provided to show how an initial **Interface Database** is created for a particular installation. The **Interface Database** describes the configuration of T1 spans/channels connected to the **950RTS** test ports

Subsequent software releases will include a visual tool for database maintenance.

```
#!/bin/sh
#^I@(#) SageRTS 0.0.0x $RCSfile: makedb,v $ $Revision: 1.9 $ $State: Exp $
$Date
: 1997/11/14 23:27:25 $ GMT
#
# T1 Interface and Channel database Setup Script
#
PATH=/u/950rts/bin; export PATH

# Fields in the 950intfc file:
# This database describes the cards, where they are located, and what type
# they are.
# Board and interface numbers start at zero and go up, so the first board is
# board 0 and has interfaces 0 and 1
# The id field is a unique number that is used to reference the interface in
# the other databases. It should start at 1 and go up; typically, board 0
# interface 0 is id 1, board 0 interface 1 is id 2, and so on.
# The name is arbitrary, but must start with a character (not a number). It
# can be up to 20 characters long, and typically identifies the equipment
# to which that interface is connected.
# The mgr and type fields are always fixed at "t1im" and "digital" respectively.

# The subtype field identifies the board type and whether it is T1 or E1:
# subtype^Idescription
# ds1-dti^IT1, Sage Digital Trunk Interface (DTI) board
# e1-dti^IE1, Sage Digital Trunk Interface (DTI) board
# ds1-xircom T1, Xircom ISA-48 board
# e1-xircom ^IE1, Xircom ISA-48 board
# The bandwidth field should be set to 1.544 or 2.048 for T1 or E1 respectively
# The cabinet field is not currently used, set to 1.
#

# delete the 950intfc database, then re-init it
```

makedb

Appendix V Sage 950RTS Commands

```
db X 950intfc
db i 950intfc

# set-up board 0 interface 0
db c 950intfc id=1
db m 950intfc 1 name=B0I0
db m 950intfc 1 board=0 intfcNum=0
db m 950intfc 1 mgr=t1im type=digital
db m 950intfc 1 subtype=ds1-dti
db m 950intfc 1 bandwidth=1.544000
db m 950intfc 1 cabinet=1

# set-up board 0 interface 1
db c 950intfc id=2
db m 950intfc 2 name=B0I1
db m 950intfc 2 board=0 intfcNum=1
db m 950intfc 2 mgr=t1im type=digital
db m 950intfc 2 subtype=ds1-dti
db m 950intfc 2 bandwidth=1.544000
db m 950intfc 2 cabinet=1

# Fields in the 950dintfc file:
# This database contains framing, line code, and similar information on each
# interface. It can also be modified by the t1config program; note, however,
# that t1config does ***NOT*** update this initialization script.
# The id field matches the id field in 950intfc.
# The framing field must be one of the following:
# ^IT1^I^IE1
# ^Inone^I^Inone
#     esf^I^Ipcm31-crc
# ^Isf^I^Ipcm31
# ^I^I^Ipcm30-crc
# ^I^I^Ipcm30
# The coding field must be one of the following:
# ^IT1^I^IE1
# ^Iami^I^Iami
# ^Ib8zs^I^Ihdb3
# ^Ib7stuff
# The length field indicates the approximate distance between the 950RTS and
# the DSX:
#         0-133
#         133-266
#         266-399
#         399-533
#         533-655
# The monitor field describes the type of T1-ESF performance monitoring
# on the datalink. Set it to "none" in all other framing types:
# ^Inone
```

makedb

```
#^Iansi$
#^Iatt$
#
# The clock field describes the clock source for the board, and whether$
# it provides MVIP master clock or is slaved to it.  $
# If the board is used for channel access, the board must be synchronized to
# the MVIP, and it may provide MVIP master clock. The valid values for clock
# for channel access boards are:
# ^Iprimary^I- this interface is the primary MVIP master clock
# ^Isecondary^I- this interface is the secondary MVIP master clock
# ^Itertiary^I- this interface is the tertiary MVIP master clock
# ^Islave^I- this interface is synchronized to the MVIP clock
# There can be only one primary, secondary, or tertiary clock. If only slave
# clocks are defined, the system uses internal clock.
# If the board is used for DS1 bit error rate testing (full T1 or FAD), the
# valid values for clock are
# ^Iinternal^I- this interface sources clock
# ^Ilooptime^I- this interface is looped time to its incoming signal
# These values are only used when the interface is not in use; otherwise clockin
# g
# for full DS1 testing is determine in the test application.

#delete the t1config database then re-init it
db X 950dintfc
db i 950dintfc

# board 0 interface 0
db c 950dintfc id=1 framing=sf coding=b7stuff length=0-133 monitor=none
clock=primary
db c 950dintfc id=2 framing=sf coding=b7stuff length=0-133 monitor=none
clock=secondary

# the 950chan database describes the circuits available for testing.
# The name field must be unique for each circuit. Applications which
# list circuits will sort on this field, so the presentation order is
# dependent on the name.
# The id field specifies the interface on which this circuit exists
# (reference the 950intfc file above).
# The num field specifies the channel number on that interface:
# T1 TAP^I1-12
# E1 TAP^I1-12
# T1 DS0^I1-24
# E1 DS0^I1-31
# others^I1

# The port field is an arbitrary (but not necessarily unique) name
# which is used to identify the circuit to the user in the application.
# Ports with the same name may be used interchangeably; the system will
# automatically pick the first non-busy port.
# The type field specifies what the circuit type is:
```

madeb

Appendix V Sage 950RTS Commands

```
# Type^IDescription
# e1-fad^IE1, facility access digroup to DCS
# t1-fad^IT1, facility access digroup to DCS
# e1^I^IFull E1 pipe
# e1-tad^IE1, test access digroup to DCS
# t1^I^IFull T1 pipe
# t1-tad^IT1, test access digroup to DCS
# ds0clear^IDS0, clear channel
# ds0rbs^IDS0, with robbed bit signaling
# dtapclear^ITest access path to DCS, clear channel
# dtaprbs^ITest access path to DCS, with robbed-bit signaling
#
# The sysname field is used to identify the TAP to the DCS. It
# not used and can be left blank for non-TAP circuits.
#
# Examples:
#
# db c 950chan name=dacs4 id=4 port=dacs4 num=1 type=t1-tad
# db c 950chan name=01 id=4 port=dacs4rbs num=1 sysname=1 type=dtaprbs

# delete the ds0 and re-init
db X 950chan
db i 950chan

db c 950chan name=z101 id=1 port=rb101 num=1 type=ds0rbs sysname=
db c 950chan name=z102 id=1 port=rb102 num=2 type=ds0rbs sysname=
db c 950chan name=z103 id=1 port=rb103 num=3 type=ds0rbs sysname=
db c 950chan name=z104 id=1 port=rb104 num=4 type=ds0rbs sysname=
db c 950chan name=z105 id=1 port=rb105 num=5 type=ds0rbs sysname=
db c 950chan name=z106 id=1 port=rb106 num=6 type=ds0rbs sysname=
db c 950chan name=z107 id=1 port=rb107 num=7 type=ds0rbs sysname=
db c 950chan name=z108 id=1 port=rb108 num=8 type=ds0rbs sysname=
db c 950chan name=z109 id=1 port=rb109 num=9 type=ds0rbs sysname=
db c 950chan name=z110 id=1 port=rb110 num=10 type=ds0rbs sysname=
db c 950chan name=z111 id=1 port=rb111 num=11 type=ds0rbs sysname=
db c 950chan name=z112 id=1 port=rb112 num=12 type=ds0rbs sysname=
db c 950chan name=z113 id=1 port=rb113 num=13 type=ds0rbs sysname=
db c 950chan name=z114 id=1 port=rb114 num=14 type=ds0rbs sysname=
db c 950chan name=z115 id=1 port=rb115 num=15 type=ds0rbs sysname=
db c 950chan name=z116 id=1 port=rb116 num=16 type=ds0rbs sysname=
db c 950chan name=z117 id=1 port=rb117 num=17 type=ds0rbs sysname=
db c 950chan name=z118 id=1 port=rb118 num=18 type=ds0rbs sysname=
db c 950chan name=z119 id=1 port=rb119 num=19 type=ds0rbs sysname=
db c 950chan name=z120 id=1 port=rb120 num=20 type=ds0rbs sysname=

db c 950chan name=z121 id=1 port=rb121 num=21 type=ds0rbs sysname=
db c 950chan name=z122 id=1 port=rb122 num=22 type=ds0rbs sysname=
db c 950chan name=z123 id=1 port=rb123 num=23 type=ds0rbs sysname=
db c 950chan name=z124 id=1 port=rb124 num=24 type=ds0rbs sysname=
```

makedb

```
db c 950chan name=z201 id=2 port=rb201 num=1 type=ds0rbs sysname=$
db c 950chan name=z202 id=2 port=rb202 num=2 type=ds0rbs sysname=$
db c 950chan name=z203 id=2 port=rb203 num=3 type=ds0rbs sysname=$
db c 950chan name=z204 id=2 port=rb204 num=4 type=ds0rbs sysname=$
db c 950chan name=z205 id=2 port=rb205 num=5 type=ds0rbs sysname=$
db c 950chan name=z206 id=2 port=rb206 num=6 type=ds0rbs sysname=$
db c 950chan name=z207 id=2 port=rb207 num=7 type=ds0rbs sysname=$
db c 950chan name=z208 id=2 port=rb208 num=8 type=ds0rbs sysname=$
db c 950chan name=z209 id=2 port=rb209 num=9 type=ds0rbs sysname=$
db c 950chan name=z210 id=2 port=rb210 num=10 type=ds0rbs sysname=$
db c 950chan name=z211 id=2 port=rb211 num=11 type=ds0rbs sysname=$
db c 950chan name=z212 id=2 port=rb212 num=12 type=ds0rbs sysname=$
db c 950chan name=z213 id=2 port=rb213 num=13 type=ds0rbs sysname=$
db c 950chan name=z214 id=2 port=rb214 num=14 type=ds0rbs sysname=$
db c 950chan name=z215 id=2 port=rb215 num=15 type=ds0rbs sysname=$
db c 950chan name=z216 id=2 port=rb216 num=16 type=ds0rbs sysname=$
db c 950chan name=z217 id=2 port=rb217 num=17 type=ds0rbs sysname=$
db c 950chan name=z218 id=2 port=rb218 num=18 type=ds0rbs sysname=$
db c 950chan name=z219 id=2 port=rb219 num=19 type=ds0rbs sysname=$
db c 950chan name=z220 id=2 port=rb220 num=20 type=ds0rbs sysname=$
db c 950chan name=z221 id=2 port=rb221 num=21 type=ds0rbs sysname=$
db c 950chan name=z222 id=2 port=rb222 num=22 type=ds0rbs sysname=$
db c 950chan name=z223 id=2 port=rb223 num=23 type=ds0rbs sysname=$
db c 950chan name=z224 id=2 port=rb224 num=24 type=ds0rbs sysname=$
```

makedb

