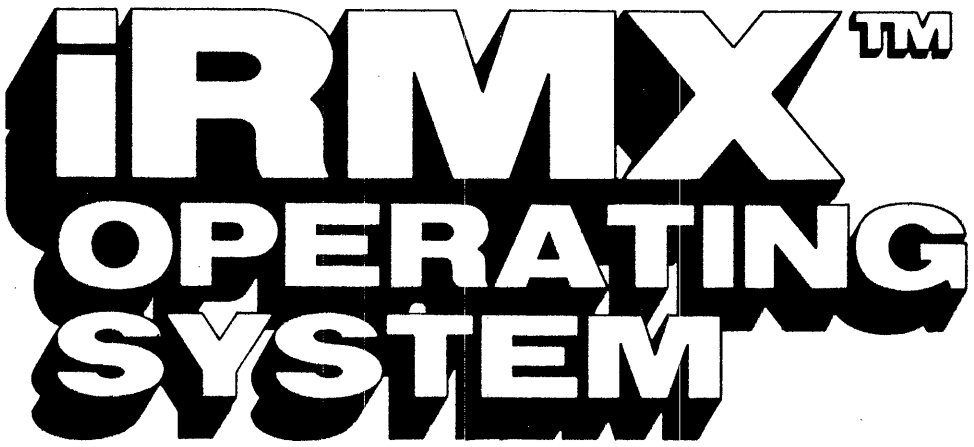
intel®

iRMX™ SYSTEM ANALYSIS TEST USER'S GUIDE



Intel Corporation, 3065 Bowers Avenue, Santa Clara, California 95051

Order Number: 173774-001

iRMX™ SYSTEM ANALYSIS TEST USER'S GUIDE

Order Number: 173774-001

Intel Corporation, 3065 Bowers Avenue, Santa Clara, California 95051

Additional copies of this manual or other Intel literature may be obtained from:

Literature Department
Intel Corporation
3065 Bowers Avenue
Santa Clara, CA 95051

The information in this document is subject to change without notice.

Intel Corporation makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Intel Corporation assumes no responsibility for any errors that may appear in this document. Intel Corporation makes no commitment to update or to keep current the information contained in this document.

Intel Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in an Intel product. No other circuit patent licenses are implied.

Intel Corporation makes no warranty for the use of its products and assumes no responsibility for any errors which may appear in this document nor does it make a commitment to update the information contained herein.

Intel retains the right to make changes to these specifications at any time, without notice.

Contact your local sales office to obtain the latest specifications before placing your order.

The following are trademarks of Intel Corporation and its affiliates and may be used only to identify Intel products:

BITBUS	iLBX	iPDS	Plug-A-Bubble
COMMputer	i_m	iRMX	PROMPT
CREDIT	iMMX	iSBC	Promware
Data Pipeline	Insite	iSBX	QUEX
Genius	int_el	iSDM	QUEST
Δ	Int_elBOS	iSXM	Ripplemode
1	Intelevision	Library Manager	RMX/80
i	$int_eligent$ Identifier	Megachassis	RUPI
I^2ICE	$int_eligent$ Programming	MICROMAINFRAME	Seamless
ICE	Intellec	MULTIBUS	SOLO
iCS	Intellink	MULTICHANNEL	SYSTEM 2000
iDBP	iOSP	MULTIMODULE	UPI
iDIS			

MDS is an ordering code only and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Sciences Corporation.

MULTIBUS is a patented Intel bus.

Printed in USA/IS-044/3.5K/1183/AP
SOFTWARE

REV.	REVISION HISTORY	DATE
-001	Original issue	12/83

This manual describes the iRMX™ System Analysis Test (RSAT) and how to use it. RSAT allows you to exercise the system hardware and the iRMX operating system for extended periods of time, enabling you to detect intermittent, device driver, and controller errors. RSAT also lets you determine the system software's ability to run on the chosen hardware.

NOTATIONAL CONVENTIONS

This manual uses to following notational conventions to illustrate command syntax.

BOLDED	The bolded portion of any commands must be entered verbatim. However, since RSAT is not case sensitive, this information may entered in either upper- or lower-case.
NOT BOLDED	The portions of a command line that are not bolded indicate information that must be supplied by the user. This information may also be entered in either upper- or lower-case.
[ ]	Brackets indicate optional parameters.

RELATED PUBLICATIONS

When you are using your RSAT, you may find it useful to consult the following publications. These publications, and other Intel publications, are available from:

Literature Department
Intel Corporation
3065 Bowers Avenue
Santa Clara, CA 95051

Introduction to the System 310 Microcomputer, Order Number 173202

Introduction to the System 86/380 and System 86/330A Microcomputer Systems, Order Number 172758

iRMX-86 Release 5 Basic I/O System Reference Manual for System 86/300 Series Microcomputer Systems, Order Number 172766

iRMX-86 Release 5 Extended I/O System Reference Manual for System 86/300 Series Microcomputer Systems, Order Number 172767

iSBC® 86/14 and iSBC 86/30 Single Board Computer Hardware Reference Manual, Order Number 144044

iSBC 337 MULTIMODULE™ Numeric Data Processor Hardware Reference Manual, Order Number 142887

System 86/300 Series Diagnostic Maintenance Manual, Order Number 144813

United States customers may obtain service and repair assistance by contacting the Intel Product Service Center in Phoenix, Arizona. Customers outside the United States should contact their sales source (Intel Sales Office or Authorized Distributor) for service information.

Before calling the Product Service Center you should have the following information:

- The date you received the product.
- The complete part number (including the dash number) of the product. This number is usually silk-screened onto printed circuit boards and stamped on the label of other products.
- The serial number of the product. This is usually silk-screened onto printed circuit boards and stamped on the label of other products.
- Your shipping and billing addresses.
- A purchase order number for billing purposes if your Intel product warranty has expired.
- Extended warranty agreement information, if applicable.

SERVICE AND REPAIR ASSISTANCE

Use the following telephone numbers to contact the Intel Product Service Marketing Administration group:

	Regional Telephone Numbers	TWX Numbers
Western Region	(602) 869-4951	910-951-1330
Midwestern Region	(602) 869-4392	910-951-0687
Eastern Region	(602) 869-4045	
International	(602) 869-4391	

Always contact the Intel Product Service Marketing Administration group before returning a product to Intel for repair. When you make the request you will be given a repair authorization number, shipping instructions, and other information that will help Intel provide you with fast, efficient service.

If you are returning a product because of damage sustained during shipment or if the product is out of warranty, a purchase order is required before Intel can initiate repair.

Use the original factory packaging material in preparing a product for shipment to the repair center. If that material is not available, ensure the product is adequately protected by wrapping it in cushioning material before enclosing it in a heavy-duty corrugated shipping carton. All cartons should be labeled "FRAGILE" to ensure careful handling. If a printed circuit board is being returned, a material such as Air Cap TH-240, manufactured by the Sealed Air Corporation of Hawthorne, New Jersey, should be used to give adequate cushioning.

Address and ship only to the address specified by Intel Product Service Marketing Administration group personnel.

CONTENTS

iRMX™ SYSTEM ANALYSIS TEST USER'S GUIDE

The primary purpose of the iRMX™ System Analysis Test (RSAT) is to detect hardware and software test errors. RSAT allows you to exercise the system hardware and iRMX operating system for extended periods enabling you to detect intermittent errors, boundary conditions, device driver errors, and controller board errors. RSAT performs a variety of asynchronous functions simultaneously and consists of the following two categories of tests for the System 300 Series microcomputers:

- I/O routines for disk devices
- Numeric Data Processor Extension (NDPX) routines

When you invoke RSAT, you may specify a maximum of 16 test routines from the two categories combined. If you are using test routines from only one of the two categories, you may specify a maximum of 18 routines.

HARDWARE AND SOFTWARE REQUIREMENTS

You must have the following hardware before you can run RSAT:

- An 8086- or 80286-based System 300 Series Microcomputer capable of supporting the iRMX operating system
- Sufficient memory to load the operating system and RSAT (the amount of memory will vary depending on the configuration of your iRMX operating system and the buffer sizes defined when you invoke RSAT)
- A terminal with cursor movement capability activated by the receipt of an ESCAPE character and an 8-bit code

You will also need the following software:

- iRMX 86 Operating System (Release 5 or later) or iRMX 286R (Release 1 or later) installed on the system
- iRMX 86 Release 5 terminal support software and Human Interface layer configured into the system
- RSAT
- TERMDEFS file (unless an ANSI-standard terminal is used)

FIGURES

INSTALLATION

RSAT may be invoked and run from any iRMX storage device. If you wish to invoke RSAT from the Winchester disk, copy all the RSAT directory files from the release diskette to a directory on the Winchester disk. Be sure that the flexible diskette drive is properly attached before attempting to install RSAT. To install the RSAT directory on your Winchester disk, enter the following commands:

submit :dev:**install.csd**(:dev:)

where :dev: is the logical name for the flexible diskette drive. The preceding command creates the RSAT directory under the :HOME: directory.

The System Analysis Test contains the following files under the directory RSAT.

RSAT -- The iRMX System Analysis Test

READ_ME -- A help file containing standard invocation information

TERMDEFS -- The iRMX terminal definition data base file

RESET -- A file that when copied to :co: cancels the terminal support translation in case RSAT finishes abnormally (for example: terminating RSAT with a CONTROL-C). More information about the RESET command is found in Appendix B of this manual.

RSAT INVOCATION

RSAT is invoked like any other iRMX Human Interface command. The command for invoking RSAT from a Winchester disk is:

rsat/rsat terminal=$\begin{matrix}\text{termspec}\\ \text{ANSI}\end{matrix}$

The command for invoking RSAT from a flexible diskette is:

:dev:**rsat/rsat terminal**=$\begin{matrix}\text{termspec}\\ \text{ANSI}\end{matrix}$

INPUT PARAMETERS

The input parameters for the preceding invocation commands are as follows:

terminal= The parameter that allows you to define your terminal if it does not support ANSI (X3.64) cursor control.

termspec	The name of the terminal in the TERMDEFS file that defines your terminal. See Appendix A.
ANSI	The response which tells RSAT that your terminal supports ANSI (X3.64) cursor control.

All terminal specifications are defined in the TERMDEFS terminal definition data base file. Appendix A of this manual includes a list of all the terminals and their terminal specification names defined on the TERMDEFS file. If your terminal is not included in this list, refer to Appendix B of this manual for instructions to add a new terminal specification.

If you do not include the **terminal=** parameter to invoke RSAT, RSAT scans the TERMDEFS file for the terminal specification labeled DEFAULT:>. The default terminal, as defined by Intel, is the Hazeltine 1510. (Refer to Appendix B for instructions on how to change the default terminal specification.) If RSAT cannot find the default specification, it displays a message indicating a terminal specification is required.

Instead of using the full parameter name **terminal=** in the command line, you can use the abbreviation **term=**.

After you invoke RSAT, it searches the TERMDEFS file for the Terminal Support Code setup for the terminal type you specified. It then displays the characters :><: indicating that your terminal is now ready to use cursor controls to display RSAT output. The first output you see after RSAT clears the screen are the headings "File name" and "Buffer size" at the top of the screen. RSAT then issues the prompt character > to ask you for a file name to be used in the I/O test. RSAT accepts any valid iRMX 86 pathname of 1 to 18 characters.

After you enter the pathname and a carriage return, RSAT issues another prompt character (>) under the heading "Buffer size". Enter the buffer size that is to be used with this I/O test. The buffer size (in bytes) must be between 1 and 65,534 decimal, although you can place a "k" after the buffer size to denote that the number should be multiplied by 1024 to obtain the actual buffer size. If you enter a value of zero, an invalid value, or a carriage return for the buffer size, the test picks a random number between 100 and 10,000 for the buffer size.

After you set up the first test, RSAT prompts you for another file name (on the line immediately under the previous file name). Type in the pathname and the buffer size for the second test. RSAT can query you for 18 file names and 18 buffer sizes. You can stop the query at any time in the process by typing a carriage return when RSAT asks you for a file name.

After you enter the test information, RSAT performs a quick check to determine if your system contains a Numeric Data Processor Extension (NDPX). If you do have an NDPX and you haven't specified more than 15 I/O tests, RSAT queries you for the number of

NDPX tests to run. The following message appears on the screen:

How many NDPX tests: <xx> possible?

where <xx> is the number of NDPX tests that can be run. Respond to the question with the number of NDPX tests you wish to run. A carriage return signifies a response of zero. If you respond with a number greater than xx, RSAT runs xx number of NDPX tests. The number xx is dependent on the number of file names entered previously.

If you do not have an NDPX in your system, RSAT will not query you for the NDPX tests.

STANDARD INVOCATION

If you intend to use RSAT as a system stress test for the System 300 Series Microcomputers, the following standard invocation will produce the worst case conditions and therefore perform the most comprehensive testing:

rsat t#(4,9k,15k) :fd0:t#(2,5k,7k) ndpx=2 terminal=<term spec>

In the preceding invocation command, the entries inside the parentheses indicate the number of files to be created, the minimum buffer size, and the maximum buffer size, respectively. The minimum buffer size must be specified but the maximum buffer size is optional. If a maximum buffer size is not specified, RSAT assumes the buffer size to be only the minimum value. If you want the test files to be created in a directory other than the default directory, precede the file names with the name of the directory. The diskette drive must be attached as :fd0: or the correct device name must be used in place of :fd0:. If your system does not have an NDPX math coprocessor, omit the NDPX parameter.

RSAT OPERATION

During RSAT execution, the output of the tests is displayed on the terminal screen. Each test uses one terminal line to report its data. Thus, test information is displayed across the screen and not scrolled.

The basic RSAT I/O test process has two phases: filling a file until the device is full and truncating files until the device is empty. The first phase is as follows:

1. Take a file name and a buffer size as input.
2. Delete the file to make sure it does not exist.
3. Create the file and open it for reading and writing.
4. Fill a buffer with a byte value.
5. Write the buffer to the file.
6. Search back the buffer size and read in the data that was just written to the file.

7. Compare the data to the byte value to make sure the I/O transfer was successful.
8. Increment the byte value and repeat the process until the device is full.

The second phase of the RSAT I/O test process is as follows:

1. Seek to the end of the file and then search back the buffer size.
2. Read in the data and compare the data to the expected byte value.
3. Seek back the buffer size again and truncate the file.
4. Repeat the compare process until the entire file is checked.

The test updates the screen at various times. Each test updates all its information at the beginning and halfway through a pass. A complete pass of the test is from one file creation to another. Half a pass occurs when the file is full and the test starts checking the file.

RSAT OUTPUT

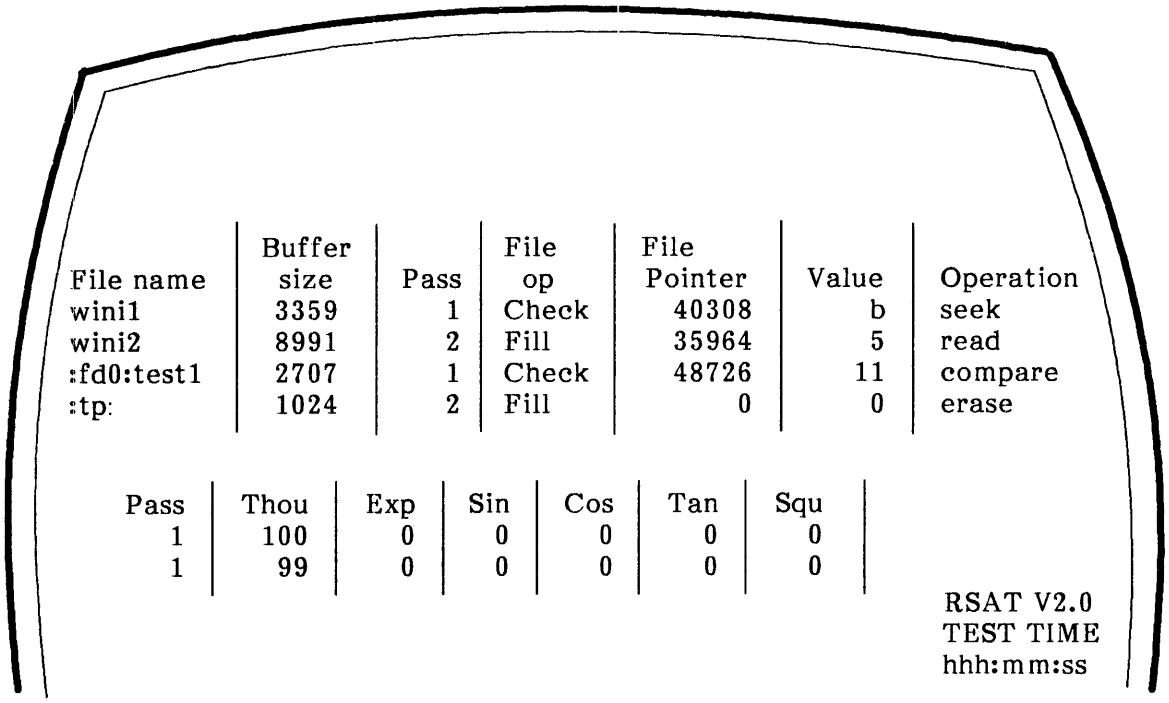
The example in Figure 1 shows RSAT output on a terminal screen. Explanations of the various fields of information shown in Figure 1 are as follows:

File name	This field contains the file name that the test is using.
Buffer size	This field contains the buffer size used by the test.
Pass	This field contains the number of the pass the test is executing.
File op	This field lists the file operation. The file operation is either "Fill" to signify the test is filling the file with data or "Trunc" to signify that the test is checking back through the file for valid data.
File Pointer	This field contains the eight-digit decimal value of the current file pointer.
Value	This field contains the hexadecimal value the test places in the I/O buffer at the byte specified by the position of the file pointer.
Operation	This field contains a short description of the current operation of the test.

When filling the file, the sequence of operations for the I/O disk test is Write, Seek, Read, and Compare. At the end of the sequence the test updates the current file pointer, increments the buffer value, and displays the new value under the heading "Value". The test repeats this sequence until the disk device is full.

For the second half of the I/O disk test the sequence under the "Operation" heading is Seek, Read, Compare, Seek, and Truncate. At the end of the sequence the test updates the current file pointer, decrements the buffer value, and displays the new value under the heading "Value".

At the begining of a pass and halfway through a pass the test updates all the I/O test screen information. If an error occurs, the test refreshes all of the test data except the rightmost field, titled "Operation".



File name	Buffer size	Pass	File op	File Pointer	Value	Operation
wini1	3359	1	Check	40308	b	seek
wini2	8991	2	Fill	35964	5	read
:fd0:test1	2707	1	Check	48726	11	compare
:tp:	1024	2	Fill	0	0	erase

Pass	Thou	Exp	Sin	Cos	Tan	Squ
1	100	0	0	0	0	0
1	99	0	0	0	0	0

RSAT V2.0
TEST TIME
hhh:mm:ss

Figure 1. RSAT Output F-0076

Like the I/O tests, each NDPX test displays its information across the terminal line. From left to right, the information displayed by each NDPX test is as follows:

Pass — The pass number for the test.

Thou — The count of how many thousand iterations of the test loop have been executed.

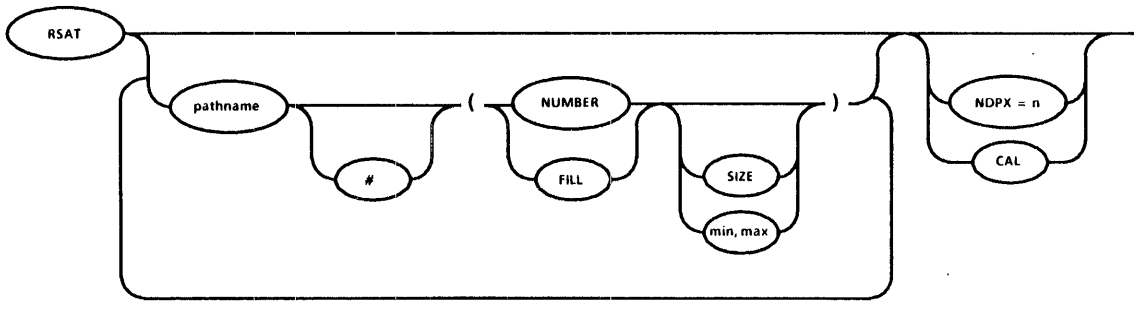
Exp	The total number of errors for the exponential test. Values other than zero imply that the NDPX is defective.
Sin	The total number of errors for the sine test. Values other than zero imply that the NDPX is defective.
Cos	The total number of errors for the cosine test. Values other than zero imply that the NDPX is defective.
Tan	The total number of errors for the tangent test. Values other than zero imply that the NDPX is defective.
Squ	The total number of errors for the square root test. Values other than zero imply that the NDPX is defective.

TEST TIME

RSAT shows the time that the test has been executing at the lower right corner of the screen in the form hhh:mm:ss, where "hhh" is the hours, "mm" is the minutes, and "ss" is the seconds of elapsed time since the test started. The time is updated every two seconds by a task that runs at a higher priority than any of the test tasks. Immediately above the test time is the version number of the RSAT being run.

OPTIONAL PARAMETERS

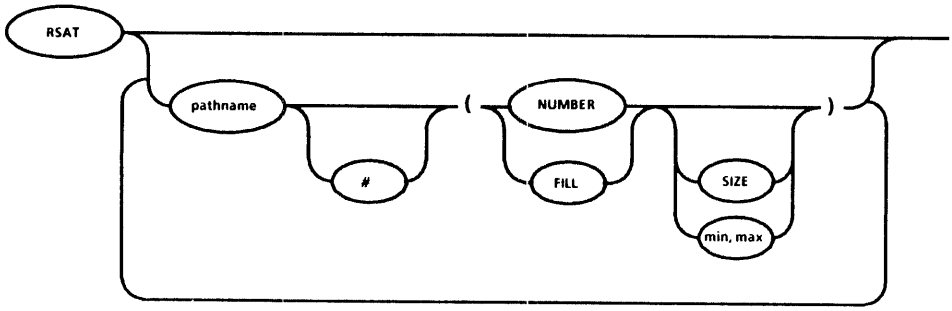
There are a number of optional command line parameters which will help you in setting up the RSAT tests. These optional parameters are shown below in the RSAT syntax statement and are described on the following pages.



F-0145-1

PATHNAME AND ASSOCIATED PARAMETERS

Several of the parameters allow you to set up a number of tests with one invocation. The syntax is as follows:



F:0146-1

Parameters

pathname Any valid iRMX pathname between 1 and 18 characters. You can enter up to eight sets of pathname parameters on one command line, which allows testing of up to 20 devices. Each device to be tested must be attached before RSAT is run. Each of the pathname parameters must be separated by a space.

Used by RSAT to generate unique file names (since each test process operates on a separate file). Wherever the # sign appears in the pathname, the RSAT substitutes a number. For the first test a "1" is substituted and each additional test uses a number that is one greater than the last.

number Indicates the number of tests that will use this file name. The number can be between 1 and 18 or can be the word "FILL". The parameter "FILL" signifies that the rest of the screen should be filled with tests using this pathname and parameters.

size An optional parameter allowing you to specify the buffer size (in decimal bytes) to be used with each test. The buffer size is system dependent. If you omit buffer size, RSAT prompts you for the buffer size. If the value is zero, RSAT generates random buffer sizes between 100 and 10,000.

min,max Optional parameters that allow you to specify the minimum and maximum values for the buffer size. If you choose this option, RSAT generates a random buffer size between these two values. The maximum value must be less than or equal to 65,534 decimal. A "k" can be placed after the minimum and maximum values to denote that the value should be multiplied by 1024 decimal to obtain the actual value.

The following examples use the file name parameter.

RSAT t#(5)
Causes RSAT to generate five tests with the file names t1, t2, t3, t4, and t5. As each test process is generated, RSAT asks you for the buffer size to use.

RSAT t#(5,1000)
Causes RSAT to generate five tests with the file names t1, t2, t3, t4, and t5, and gives each test a buffer size of 1000 bytes.

RSAT t#(5,1000,10000)
Causes RSAT to generate five tests with the file names t1, t2, t3, t4, and t5, and gives each of the tests a random buffer size between 1000 and 10,000 bytes.

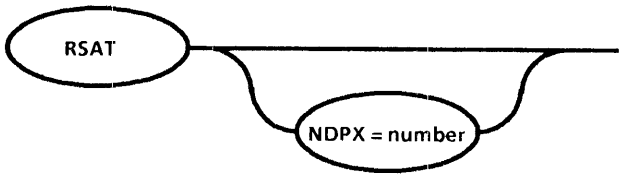
RSAT t#(5,1000,10000) :fd0:t#(f,100,2k)
Causes RSAT to generate five tests as described in the previous paragraph and fill the rest of the screen with tests using the file name :fd0:t#. RSAT randomly generates a buffer size between 100 and 2048 bytes for each of the tests using the file name fd0:t#.

RSAT ignores tests if you specify more than 18 tests in the same command line. Starting from the left, RSAT attempts to generate each test. If RSAT reaches an eighteenth test, it ignores the remaining tests. The following command line illustrates this situation:

RSAT t#(10,1000,10000) :fd0:t#(10,100,2000)
Here RSAT generates the first ten tests using the first pathname and parameters, executes the first eight tests in the second pathname and parameters, and ignores the remaining two tests.

NDPX PARAMETER

The NDPX parameter provides a quick way of specifying the number of NDPX component tests. This parameter is position-dependent since it assumes that the rest of the screen should be used for NDPX component tests. Thus, this parameter should be used after the file name parameters. The syntax is as follows:



F-0147-1

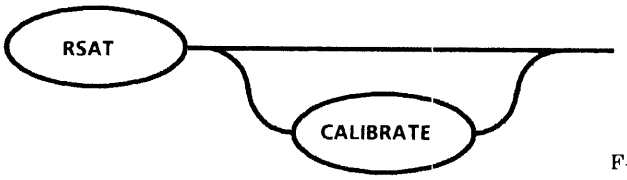
where "number" is a decimal number designating the number of NDPX component tests RSAT should generate.

For example:

RSAT t#(5,1000,20000) NDPX=5
The pathname and parameters generate five I/O tests and the NDPX parameter generates five NDPX component tests.

CALIBRATE PARAMETER

The CALIBRATE parameter is used to tell RSAT to switch to calibration mode. This allows you to calibrate either the transfer rates of your devices or the CPU idle time. In this mode the clock is updated every minute and the default speed for the I/O tests is FULL. RSAT also computes and displays the maximum device transfer rate and the idle time limit in this mode. These calibrated values can then later be entered when RSAT is reinvoked. (For more information, refer to the sections of this chapter on the FULL, RATE, and IDLE parameters.) The syntax for the CALIBRATE parameter is as follows:



F-0148-1

You can abbreviate the CALIBRATE parameter. (For example, you can use C, CA, or CAL.) Once you have performed the calibration you must exit RSAT and then reinvoke RSAT. An example of this parameter is shown below:

RSAT t#(1,63k) CAL
Here the CALIBRATE parameter is used with the pathname and parameters. If you choose the option to specify the buffer size, use only one test.

RUNTIME PARAMETERS

There are a number of runtime parameters that you can use to interactively change the RSAT test environment and obtain performance information about the system. These runtime parameters include HELP, FULL, IDLE, NORMAL, QUIT, RATE, REDRAW, and STOP.

To use the runtime parameters, enter the parameter name while RSAT is performing its output display. Complete your entry, which the test does not echo on the screen, with a carriage return. You can abbreviate the parameters to the first unique letters and use either upper- or lower-case.

Each of the RUNTIME parameters is discussed on the following pages.

HELP PARAMETER

The HELP parameter lists the runtime parameters followed by a short description. The HELP parameter locks the screen to prevent the tests from performing output to the screen. Eventually all of the test tasks are queued waiting to perform screen output.

The HELP parameter clears the screen and displays the information shown in Figure 2.

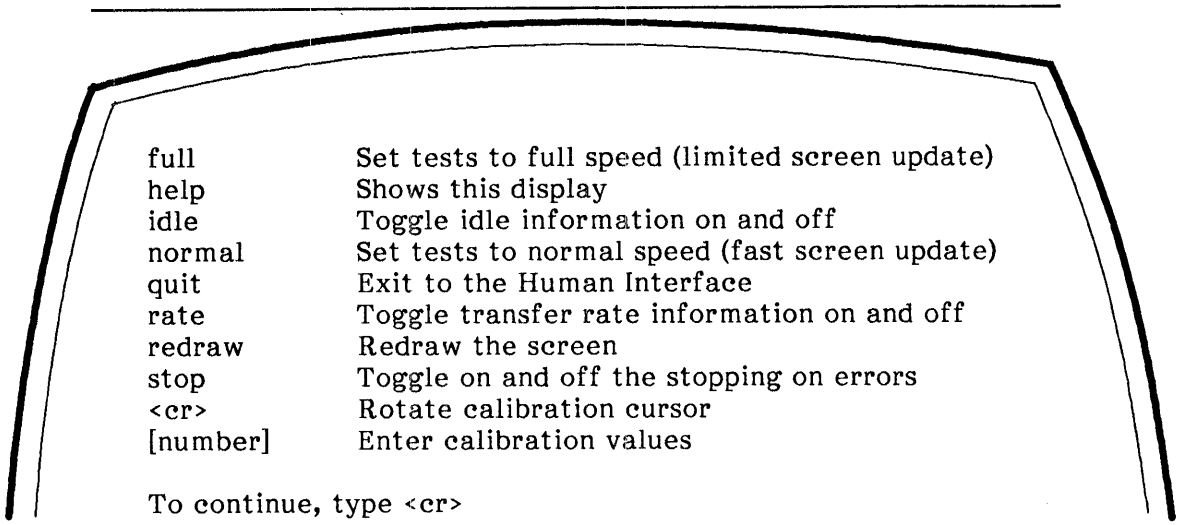


Figure 2. The HELP Screen F-0077

To continue with the RSAT test, type a carriage return. The HELP command then redraws the RSAT screen display and the tests update the screen.

NORMAL PARAMETER

The NORMAL runtime parameter resets the I/O tests to update the screen data and clock after every operation.

FULL PARAMETER

To run the RSAT I/O test at full speed the FULL parameter should be used. The FULL parameter causes the test to update I/O test data both at the beginning and halfway through a pass. The clock will be updated every 60 seconds. Under the "Operation" heading on the screen, the phrase "Full speed" appears for the current test, signifying that RSAT is running at full speed. The NDPX component tests are not affected by the FULL parameter because the tests run at a lower priority than the I/O tests.

IDLE PARAMETER

The IDLE parameter requests RSAT to display the processor IDLE information if RSAT is not displaying the information currently. If RSAT is already displaying the processor IDLE information, this parameter clears the IDLE information from the screen and requests RSAT to stop displaying the information. The IDLE information appears in the lower right-hand corner of the screen next to the time display. Figure 3 shows the form of the IDLE display.

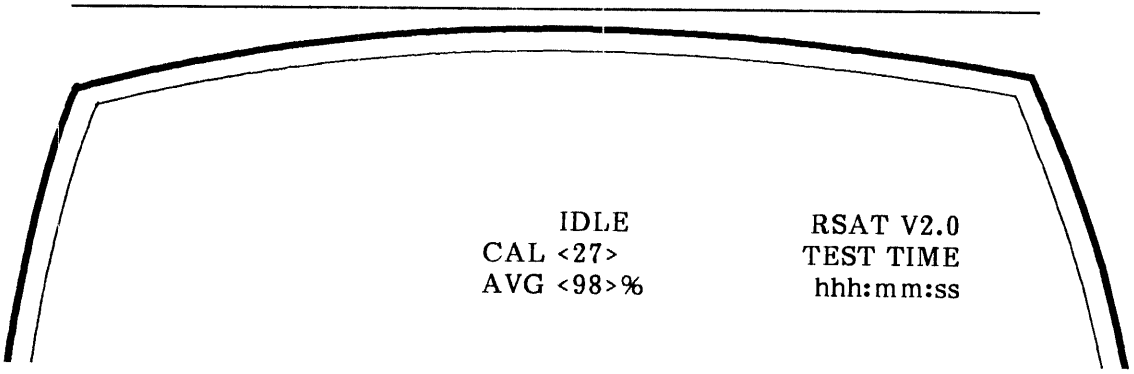


Figure 3. IDLE Information F-0078

The IDLE time displayed is the average idle time since the IDLE parameter was invoked and is displayed as a percentage. As shown in the example, the processor has been idle 98% of the time since the IDLE parameter was invoked.

Before IDLE calculations can be performed, RSAT requires the entry of a calibration number relative to the speed of your system. (See the section in this chapter titled "Entering Calibration Values" for more information on entering calibration numbers.) In the above example, the CAL number is 27.

The CAL number is obtained by invoking RSAT without any tests and with the CAL parameter (i.e., the command line would read simply "RSAT CAL"). When RSAT prompts for a file name, enter a carriage return and then select the IDLE option by typing "I" and a carriage return. After about 30 seconds RSAT will display a number in the CAL field of the IDLE display. This is the number to enter when RSAT is reinvoked. To obtain the most accurate calibration number, RSAT should be allowed to run for about two minutes in the calibration mode with no other tests and no operator input.

The IDLE value is a figure relative to RSAT when run with no tests and at full speed. This value should be used for comparison only within RSAT and should not be compared to IDLE values obtained by other means.

RATE PARAMETER

The RATE parameter causes the device transfer rate information to be displayed if RSAT is not displaying the information currently. If RSAT is already displaying the device transfer rate information, this parameter clears the information from the screen and requests RSAT to stop displaying the information.

The device rate information consists of three values: the maximum transfer rate, the running average of the number of bytes per second transferred, and the running average of the percentage of maximum device transfer rate. RSAT computes the number of bytes transferred to or from the device every minute and then divides the result by 60 to obtain the number of bytes per second transferred. The test averages the number into the running average that it has maintained since you invoked the RATE parameter. RSAT also computes the percent of maximum transfer rate for the last 60 seconds and averages the percent into the running average that it has maintained since you invoked the RATE parameter. The test displays the transfer rate information in the lower middle of the screen as shown in Figure 4.

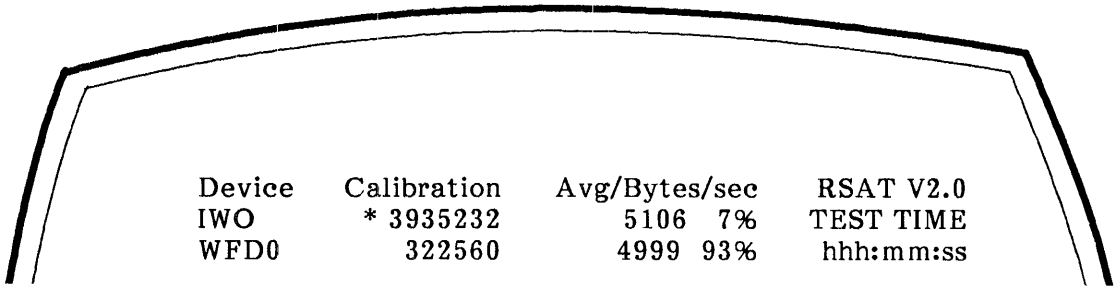


Figure 4. RATE Information F-0079

As shown in Figure 4, the running average transfer rate for the sample system is 4999 bytes per second or 93 percent of the maximum transfer rate for the device WFD0, (single-sided, double-density diskette drive). For the PRIAM Winchester in the sample system the average transfer rate is 5106 bytes per second or 7 percent of its maximum transfer rate.

Before RATE calculations can be performed, you must enter a calibration value (maximum transfer rate) for each device. More detailed information about entering calibration values is contained in the next section of this chapter, "Entering Calibration Values."

To determine the maximum transfer rate of a device, invoke RSAT with the CAL parameter and specify one file and a large buffer size. The buffer size should be a multiple of the device granularity to achieve the maximum throughput. For the example above, one test file was run on a PRIAM Winchester with a buffer size of 63K, the maximum buffer size which is a multiple of 1K. The maximum buffer size for the iRMX Basic I/O System is 64K - 1.

Note that in Figure 4 RSAT displays the number of bytes transferred in the last 60 seconds, along with the maximum number of bytes transferred (shown under the "Calibration" heading). When you reinvoke RSAT, you must enter the number that represents the maximum number of bytes transferred. The information is valid only for the first device specified under the heading "Device".

There are many factors that can affect the transfer rate values. Some of these factors are the BIOS buffer size, the interleave factor and granularity of the volume, the file granularity, the volume fragmentation, the processor clock rate, and the general speed of the system. The transfer rate is also decreased if the buffer size is not a multiple of the device granularity.

Typically, the RATE parameter would be used to measure the effect of changes to the factors mentioned above. For example, a systems programmer writing a device driver might use the RATE parameter to choose the optimum value for these factors. First RSAT would be invoked with the CALIBRATE parameter to measure the maximum transfer under the current conditions. Then the programmer would change one of the above factors and reinvoke RSAT to observe the change in the maximum transfer rate.

When using the RATE option, be aware that even as the test runs and fills the disk (thus increasing fragmentation) it affects the RATE values.

Entering Calibration Values

Both the IDLE and the RATE options require you to enter a calibration value in order for these options to perform calculations. If the values are not entered, RSAT prints question marks in place of the percentage values.

Calibration values can be entered for the IDLE option or for any of the devices in the RATE option. If only one of these options is active, the calibration value is entered by entering the number followed by a carriage return. If both the IDLE and RATE options are entered, or if there is more than one device in the RATE option, RSAT prints an asterisk (*) adjacent to the calibration value it is ready to accept. To move the asterisk from device to device or from option to option, enter a carriage return. Once the asterisk is adjacent to the value you wish to enter, enter the value followed by a carriage return.

REDRAW PARAMETER

The REDRAW parameter first clears the screen and then recreates the RSAT screen display.

STOP PARAMETER

The STOP parameter resets RSAT to stop on a test error if RSAT is not currently set to stop on errors. If RSAT is set to stop on a test error, this parameter resets RSAT to not stop on the error. When you set RSAT to stop, the message "stop on error" appears in the lower left corner of the screen. Otherwise the corner of the screen will be blank.

When you have set RSAT to stop on an error, any test errors occuring during the execution of the I/O tests halt the execution of RSAT and control passes to the iSDM™ monitor for debugging. When an error occurs, RSAT flags the I/O test that failed by placing a > in the leftmost character position (just to the left of the file name). The failing test refreshes all its test information except for the rightmost current "Operation" field. RSAT then displays the following error information in the lower left corner of the terminal screen:

```
Conn = Fcccc> IORS = <iiii> Seg = <ssss>
<eeee> error <wwww>     Unit status <uuuu>
break at <bbbb:bbbb>
.stop on error
```

where:

<cccc>	TOKEN for the connection to the file.
<iiii>	TOKEN for the IORS returned by the Basic I/O System.
<ssss>	Token for the segment containing the test's I/O buffer.
<eeee>	The last function or iRMX system call the test performed. An example of a function is a read/write mismatch. Typical system calls include read, write, and seek.
<wwww>	The error message and status code of the failing iRMX call.
<uuuu>	Unit status of the device if the status code was an E$IO (002BH).
<bbbb:bbbb>	The address of the iSDM breakpoint.

After examining all information that you wish via the iSDM monitor, enter an iSDM "G" command. RSAT then executes an RQ$EXIT$IO$JOB and deletes itself. The iRMX Human Interface prompt should then appear.

QUIT PARAMETER

The QUIT parameter stops execution of RSAT and returns control to the iRMX operating system.

CR PARAMETER

This parameter is used to rotate the calibrate cursor (an asterisk) between the IDLE option and the devices in the RATE option. This parameter enables you to select which calibration value you want to enter. The parameter is activated by typing a carriage return. Refer to the section in this chapter titled "Entering Calibration Values" for more information.

[number] PARAMETER

This parameter is used to enter a new calibration value into the selected calibration field and then cause RSAT to use the new value in its calculations. This parameter is valid only when the IDLE or RATE options are active. The parameter is activated by typing the new value and terminating it with a carriage return. Refer to the section in this chapter titled "Entering Calibration Values" for more information.

ERROR MESSAGES

RSAT displays error messages for four types of errors:

- Parameter syntax errors (can be caused by typing errors in the RSAT command line)
- Invocation errors (can be caused by improperly answering the RSAT set-up questions)
- Test errors (can be caused by hardware or software problems)
- Internal errors (can be caused by corruption of the iRMX operating system software)

PARAMETER SYNTAX ERRORS

Syntax errors can occur when you use the optional RSAT parameters. The syntax error messages appear in two formats: either a short error message or the parameter in error followed by a short error message. In the following explanations the term <parameter> is used to designate something you entered in error.

When RSAT cannot identify a parameter, it displays the following message:

<parameter>, unrecognized parameter

If a FILENAME parameter contains a file name which is greater than 18 characters, RSAT displays the file name and the following error message:

<parameter>, file name too long

When you use a FILENAME parameter and the number of tests parameter is in error, RSAT displays the following error message:

<parameter>, illegal number of tests parameter

When you use a FILENAME parameter and the minimum or maximum buffer sizes are greater than 65,535, RSAT displays one of the following error messages:

<parameter>, min value too large
<parameter>, max value too large

When you use a FILENAME parameter and the min or max values are terminated with an alphabetic character other than "k", RSAT displays one of the following error messages:

<parameter>, illegal min value
<parameter>, illegal max value

When you use a FILENAME parameter and the min value exceeds the max value, RSAT displays the following error message:

<parameter>, min value greater than max

When you use a FILENAME parameter and the number of parameters specified between the parantheses is greater than three, RSAT displays the following error message:

<parameter>, too many parameters

When you use the NDPX parameter and the number of NDPX tests is not within an acceptable range, RSAT displays the following error message:

<parameter>, illegal number of NDPX tests

INVOCATION ERRORS

A number of errors can appear during invocation of RSAT. The errors indicate that you have set up the RSAT environment incorrectly. The following errors can appear at the terminal.

Invalid logical names in a pathname cause RSAT to display the following error message:

8004: E$PARAM, illegal logical name

RSAT extracts the logical device name for each pathname and invokes the SGETFILE$STATUS system call. Any errors returned by this system call are displayed in the following message form:

<4 digit error code>: EXCEP, rq_S_get_file_status error

RSAT will display the following error message if it cannot find the the terminal specification that you entered:

<parameter>, terminal specification not found in termdefs file

where <parameter> is the terminal specification you specified.

If you did not designate a particular terminal specification when invoking RSAT and RSAT cannot find the DEFAULT specification in the TERMDEFS file, then RSAT displays the following error message:

Default not found in termdefs file, terminal spec required

If RSAT finds the terminal specification you specified but cannot find the terminator <: or if RSAT attempts to read more than 1024 characters between the initial delimiter and the trailing delimiter, then RSAT displays the following error message:

Error in termdefs file, string between :><: too long or <: missing

When RSAT finds the terminal specification you designated but then finds another :> before the terminating <:, then RSAT displays the following error message:

Error in termdefs file, :> encountered while searching for <:

A TERMDEFS file that is not in the same directory as RSAT causes RSAT to display the following error message:

0020: E$FNEXIST, termdefs file not found

When you use the NDPX parameter and the test does not receive the proper initial status from the NDPX processor, RSAT displays the following error message:

E$PARAM, NDPX not in system or not responding

TEST ERRORS

The primary purpose of RSAT is to detect test errors. Test errors are due to hardware failure or software problems. When a test error occurs, RSAT flags the failing test with an asterisk (*) just to the left of the file name. The failing test refreshes all its test information except for the rightmost "Operation" field. The "Operation" field will contain the operation that failed and an iRMX exception code. If the exception code is 002Bh (E$IO), the "Operation" field contains E$IO and the unit status for the device.

INTERNAL ERRORS

Internal errors signify that the iRMX Operating System is damaged. These errors are displayed at the terminal screen. RSAT uses the Human Interface system call C$FORMAT$EXCEPTION to display the resulting exception code's hexadecimal value and mnemonic. Following the mnemonic RSAT displays its own error description. The RSAT description consists of the name of the procedure and the name of the iRMX system call that failed. For example:

<0002>: <E$MEM>, <toggle_rate> <rq_create task>

In the example the iRMX system call CREATE$TASK in the procedure "toggle rate" returned an E$MEM exception code.

The following errors indicate catastrophic problems with the iRMX Operating System:

create_io rq_create_mailbox error
create_io rq_create_segment error
create_io rq_create_task error
create_io rq_receive_units error
create_87 rq_create_task error
create_87 rq_receive_units error
curses error
getdevtable rq_create_segment
getdevtable rq_delete_segment
getdevtable rq_C_set_parse_buffer
getdevtable rq_C_get_command_name
getdevtable rq_C_get_parameter
getdevtable rq_S_attach_file
getdevtable rq_S_delete_connection
getdevtable rq_S_get_file_status
getdevtable rq_S_open
getdevtable rq_S_read_move
idle_task rq_get_time error
io_test rq_receive_units error
io_test rq_send_units error
io_test rq_sleep error
main line rq_create_semaphore error

main line rq_create_task error
main line rq_get_priority error
main line rq_receive_units error
main line rq_send_units error
options rq_send_units error
parse rq_C_get_parameter
rate_task rq_sleep error
read_check rq_delete_segment error
read_check rq_sleep error
term_support_set rq_c_get_command name
term_support_set rq_s_attach_file
term_support_set rq_s_open
term_support_set rq_s_read_move
term_support_set rq_s_close
toggle_idle rq_create_task error
toggle_idle rq_delete_task error
toggle_rate rq_create_task error
toggle_rate rq_delete_task error
test_error rq_suspend_task error
tape_test rq_send_units error
test_error rq_sleep error
tape_test rq_receive_units error
tape_test rq_sleep error
test_NDPX rq_receive_units error
test_NDPX rq_send_units error
test_NDPX rq_sleep error
wait_io rq_delete_segment error
wait_resp rq_receive_message error

The following terminals are supported by the iRMX TERMDEFS data base. The list specifies the terminal manufacturer and model along with the name used by the TERMDEFS data to identify that particular terminal. The TERMDEFS name is used to specify terminal type during RSAT invocation.

TERMINAL NAME	TERMDEFS NAME
ADDS Regent and Viewpoint	adds
Beehive Mini Bee II	beehive
Cobar 3132 (VT52 mode)	cobar
DEC VT100 (VT52 mode)	vt100
DEC VT50 & 52	vt52
Hazeltine 1500, 1510, 1520, and Executive 80	hazeltine
Liberty Electronics Freedom 100	freedom100
Zentec Cobra (86/735)	cobra

The following TERMDEFS entries were derived from the XENIX* TERMCAP but have not been tested.

TERMINAL NAME	TERMDEFS NAME
ADDR Info	addrinfo
ADDS Consul 980	adds980
Ampex Dialogue 80	ampex
Ann Arbor	annarbor
Carlock	carlock
CDC 456	cdc456
Compucolor II	compucolor
Concept C100	c100
Concept C108	c108
Cybernex XL-83	xl83
Data General 6053	dg6053
Datamedia 2500	dm2500
Datamedia 3025	dm3025
Datamedia 5000	dm3045
Delta Data 5000	dd5000
Exidy Smart	exidy
General Terminal 100A	gt100
Heathkit H19	h19

*XENIX is a trademark of Microsoft Corporation.

TERMINAL NAME	TERMDEFS NAME
Hewlett-Packard 2626	hp2626
IBM Personal Computer	ibmpc
Microterm Act IV	microterm
Microterm Mime I	mime1
Omron 8025AG	omron
Perkin Elmer 550, 1100, 1200	perkin$elmer
Sol	sol
Soroc 120	soroc
Southwest Technical Products CT82	ct82
Teleray 1061	t1061
Teleray 3800	t3800
Terak (Datamedia 1520 emulation)	terak
Ubell Char	ubell
Volker-Craig 404	vc404
Visual 200	visual200
Zentec 30	zentec 30

APPENDIX B
CUSTOMIZING YOUR TERMINAL SUPPORT SOFTWARE

RSAT is designed to run on most terminals. To do this, RSAT uses the Terminal Support Code to translate escape sequences into equivalent terminal character sequences that the terminal understands and displays.

To perform this translation, the TERMDEFS file defines a number of terminals. This TERMDEFS file, which is based on the information found in the iRMX BIOS manual, contains terminal specifications separated by the :< and :> delimiters.

RSAT does not support all terminals; it is limited to the support available through the BIOS terminal support. For a terminal to be supported, its directional arrows must be definable using a character sequence from table B-1. As long as the directional arrows can be defined, BIOS can simulate the cursor addressing, the clear screen, and the clear to end of line functions.

If you have a terminal that is not listed in the TERMDEFS file, use the worksheet on the following pages to create a terminal specification similar to the one shown in Figure B-1. To create this specification, you must know the characteristics of your terminal. (These characteristics can be found in your terminal reference manual.)

```
HAZELTINE:>OSC: hazeltine 1500, executive 80,
T:T=1,      ; translation:  1 = enable, 0 = disable
F=0,        ; Coordinates specification:  0 to 7 as follows
            ; bit
            ; 0 ==> 0 = horiz first,     1 = vert first
            ; 1 ==> 0 = increase left to right, 1 = decrease
            ; 2 ==> 0 = increase top to bottom, 1 = decrease
U=96,       ; cursor addr offset (logical zero, zero)
V=52,       ; fall back after 127
X=80,       ; number of columns
Y=24,       ; number of lines
e2=16       ; cursor right
e3=8        ; cursor left
e4=44       ; cursor up
e5=43       ; cursor down
e6=49       ; cursor address (simulated by specifying e6=192)
e31=47      ; clear to end of line (simulate by e31=192)
ST<:
```

Figure B-1. Terminal Specification for a Hazeltine Terminal

Terminal Specification Worksheet

CODE	TERMINAL FUNCTION	EXPLANATION
_:>OSC	Terminal Specification	Enter the terminal specification name that you wish to use when invoking RSAT.
T:T=1		Do not make changes to line number 1.
F=_	Axis sequence and orientation control	Enter the decimal number that defines three terminal axes' characteristics. Calculate this number by adding the appropriate value together. Terminal axis sequence: 0 = List or enter the horizontal coordinate first. 1 = List or enter the vertical coordinate first. Horizontal axis orientation: 0 = Coordinates increase from left to right. 2 = Coordinates decrease from left to right. Vertical axis orientation: 0 = Coordinates increase from top to bottom. 4 = Coordinates decrease from top to bottom. If your calculation is the value zero, do not change this line.
U=_	Cursor addressing offset	Enter the value that starts the numbering sequence on both axes. If your terminal is like a Hazeltine, it supports the numbering convention that starts with the value 96.
V=_	Overflow offset	Enter the value to which the numbering of axes must "fall back" after reaching position 127. If your terminal is like a Hazeltine, it supports an overflow value of 32.
X=_	Screen width	Enter the number of character positions on each line of the terminal's screen. Most terminals support 80 positions per line.

Technical Specification Worksheet

CODE	TERMINAL FUNCTION	EXPLANATION
Y=_	Screen height	Enter the number of lines on your terminal's screen. Most terminals support 24 lines per screen.
e2=_	Control sequence that moves the cursor right	Enter the number of your terminal's corresponding terminal characteristic from Table B-1, column m. Hazeltine terminals use number 16 which corresponds to the Terminal Character Sequence 10H (Control-P).
e3=_	Control sequence that moves the cursor left	Enter the number of your terminal's corresponding terminal characteristic from Table B-1, column m. Hazeltine terminals use number 8 which corresponds to the Terminal Character Sequence 08H (Control-H).
e4=_	Control sequence that moves the cursor up	Enter the number of your terminal's corresponding terminal characteristic from Table B-1, column m. Hazeltine terminals use number 44 which corresponds to the Terminal Character Sequence Esc 0CH.
e5=_	Control sequence that moves the cursor down	Enter the number of your terminal's corresponding terminal characteristic from Table B-1, column m. Hazeltine terminals use number 43 which corresponds to the Terminal Character Sequence Esc 0BH.
e6=_	Control sequence that moves the cursor to a point specified by a coordinate pair	Enter the number of your terminal's corresponding terminal characteristic from Table B-1, column m. Hazeltine terminals use number 49 which corresponds to the Terminal Character Sequence Esc 11H.

Terminal Specification Worksheet

CODE	TERMINAL FUNCTION	EXPLANATION
e31_	Escape sequence that places blanks from cursor to end of line	Enter the number of your terminal's corresponding terminal characteristic from Table B-1, column m. The Hazeltine terminals use number 47 which corresponds to the Terminal Character Sequence Esc 0FH.
ST<:		Signifies the end of the terminal specification.

After the worksheet is completed, use a text editor to add the terminal specification to the TERMDEFS file. Once you have invoked the text editor, the easiest way to add a specification is to duplicate a copy of an existing specification and modify it to meet the new requirements.

If you want terminal initialization to be automatic, change your terminal specification to be the DEFAULT. Invoke the editor and move a copy of your specification to the front of the file where the existing DEFAULT is. Leave your existing specification unchanged. Change the name of the duplicate specification to "DEFAULT" (upper- or lower-case) and then delete the old DEFAULT specification. (A copy of the the old DEFAULT specification is still available under the name "hazeltine".) Although it is not necessary that the DEFAULT specification be at the front of the TERMDEFS file, it is more rapidly retrieved if it is.

After you invoke RSAT, it scans the TERMDEFS file for the specification that you specified. Having found the specification, RSAT copies it to the console. If RSAT displays the characters ":><:", your terminal is now ready to use terminal support translation to display RSAT output.

Normal exit from RSAT cancels the terminal support translation. For abnormal exit, such as the use of Control-C, you can cancel the terminal support translation by copying the file RESET to the terminal as illustrated in the following example:

```
COPY RESET TO :CO:
:><:
reset COPIED TO :CO:
```

As shown, the reset file displays the ":><:", signifying that the terminal support code interpreted the escape sequences.

Table B-1. Terminal Character Sequences

m	Terminal Character Sequence	m	Terminal Character Sequence
1	01H (Control-A)	47	Esc 0FH
2	02H (Control-B)	48	Esc 10H
3	03H (Control-C)	49	Esc 11H
4	04H (Control-D)	50	Esc 12H
5	05H (Control-E)	51	Esc 13H
6	06H (Control-F)	52	Esc 14H
7	07H (Control-G)	53	Esc 15H
8	08H (Control-H)	54	Esc 16H
9	09H (Control-I)	55	Esc 17H
10	0AH (Control-J)	56	Esc 18H
11	0BH (Control-K)	57	Esc 19H
12	0CH (Control-L)	58	Esc 1AH
13	0DH (Control-M)	59	Esc 1BH
14	0EH (Control-N)	60	Esc 1CH
15	0FH (Control-O)	61	Esc 1DH
16	10H (Control-P)	62	Esc 1EH
17	11H (Control-Q)	63	Esc 1FH
18	12H (Control-R)	64	Esc 20H
19	13H (Control-S)	65	Esc 21H
20	14H (Control-T)	66	Esc 22H
21	15H (Control-U)	67	Esc 23H
22	16H (Control-V)	68	Esc 24H
23	17H (Control-W)	69	Esc 25H
24	18H (Control-X)	70	Esc 26H
25	19H (Control-Y)	71	Esc 27H
26	1AH (Control-Z)	72	Esc 28H
27	1BH (Esc)	73	Esc 29H
28	1CH (FS)	74	Esc 2AH
29	1DH (GS)	75	Esc 2BH
30	1EH (RS)	76	Esc 2CH
31	1FH (US)	77	Esc 2DH
32	Esc 00H	78	Esc 2EH
33	Esc 01H	79	Esc 2FH
34	Esc 02H	80	Esc 30H
35	Esc 03H	81	Esc 31H
36	Esc 04H	82	Esc 32H
37	Esc 05H	83	Esc 33H
38	Esc 06H	84	Esc 34H
39	Esc 07H	85	Esc 35H
40	Esc 08H	86	Esc 36H
41	Esc 09H	87	Esc 37H
42	Esc 0AH	88	Esc 38H
43	Esc 0BH	89	Esc 39H
44	Esc 0CH	90	Esc 3AH
45	Esc 0DH	91	Esc 3BH
46	Esc 0EH	92	Esc 3CH

Table B-1. Terminal Character Sequences (Continued)

m	Terminal Character Sequence	m	Terminal Character Sequence
93	Esc 3DH	139	Esc 6BH
94	Esc 3EH	140	Esc 6CH
95	Esc 3FH	141	Esc 6DH
96	Esc 40H	142	Esc 6EH
97	Esc 41H	143	Esc 6FH
98	Esc 42H	144	Esc 70H
99	Esc 43H	145	Esc 71H
100	Esc 44H	146	Esc 72H
101	Esc 45H	147	Esc 73H
102	Esc 46H	148	Esc 74H
103	Esc 47H	149	Esc 75H
104	Esc 48H	150	Esc 76H
105	Esc 49H	151	Esc 77H
106	Esc 4AH	152	Esc 78H
107	Esc 4BH	153	Esc 79H
108	Esc 4CH	154	Esc 7AH
109	Esc 4DH	155	Esc 7BH
110	Esc 4EH	156	Esc 7CH
111	Esc 4FH	157	Esc 7DH
112	Esc 50H	158	Esc 7EH
113	Esc 51H	159	Esc 7FH
114	Esc 52H		
115	Esc 53H		
116	Esc 54H		
117	Esc 55H		
118	Esc 56H		
119	Esc 57H		
120	Esc 58H		
121	Esc 59H		
122	Esc 5AH		
123	Esc 5BH		
124	Esc 5CH		
125	Esc 5DH		
126	Esc 5EH		
127	Esc 5Fh		
128	Esc 60H		
129	Esc 61H		
130	Esc 62H		
131	Esc 63H		
132	Esc 64H		
133	Esc 65H		
134	Esc 66H		
135	Esc 67H		
136	Esc 68H		
137	Esc 69H		
138	Esc 6AH		

In addition to the values listed in Table B-1, the following values provide special functions:

- 192 Simulate the escape function
- 193 Discard the escape sequence

intel®

iRMX™ System Analysis Test
User's Guide
173774-001

REQUEST FOR READER'S COMMENTS

Intel's Technical Publications Departments attempt to provide publications that meet the needs of all Intel product users. This form lets you participate directly in the publication process. Your comments will help us correct and improve our publications. Please take a few minutes to respond.

Please restrict your comments to the usability, accuracy, organization, and completeness of this publication. If you have any comments on the product that this publication describes, please contact your Intel representative. If you wish to order publications, contact the Literature Department (see page ii of this manual).

1. Please describe any errors you found in this publication (include page number).

_____

_____

_____

_____

2. Does this publication cover the information you expected or required? Please make suggestions for improvement.

_____

_____

_____

_____

3. Is this the right type of publication for your needs? Is it at the right level? What other types of publications are needed?

_____

_____

_____

_____

4. Did you have any difficulty understanding descriptions or wording? Where?

_____

_____

_____

5. Please rate this publication on a scale of 1 to 5 (5 being the best rating). _____

NAME _____ DATE _____

TITLE _____

COMPANY NAME/DEPARTMENT _____

ADDRESS _____

CITY _____ STATE _____ ZIP CODE _____

(COUNTRY)

Please check here if you require a written reply. ☐

WE'D LIKE YOUR COMMENTS . . .

This document is one of a series describing Intel products. Your comments on the back of this form will help us produce better manuals. Each reply will be carefully reviewed by the responsible person. All comments and suggestions become the property of Intel Corporation.

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 79 BEAVERTON, OR

POSTAGE WILL BE PAID BY ADDRESSEE

Intel Corporation
5200 N.E. Elam Young Pkwy.
Hillsboro, Oregon 97123

ISO-N Technical Publications

INTEL CORPORATION, 3065 Bowers Avenue, Santa Clara, California 95051 (408) **987-8080**

Printed in U.S.A.