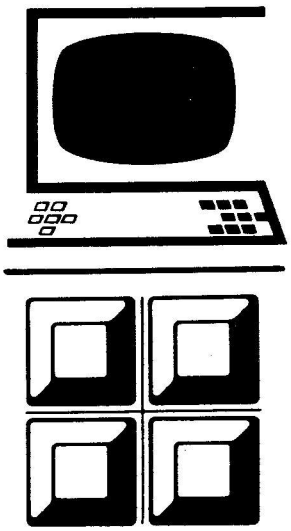
System 6300 Service Manual



87601194A

MOTOROLA
Four-Phase Systems, Inc.

List of Effective Pages

This publication contains 62 pages consisting of the following:

Page Number	Date	Page Number	Date	Page Number	Date
Title	15 Jun 84				
A	15 Jun 84				
i thru v	15 Jun 84				
vi (Blank)	15 Jun 84				
1-1 thru 1-6	15 Jun 84				
2-1 thru 2-5	15 Jun 84				
2-6 (Blank)	15 Jun 84				
3-1 thru 3-15	15 Jun 84				
3-16 (Blank)	15 Jun 84				
4-1 thru 4-18	15 Jun 84				
5-1 thru 5-4	15 Jun 84				
User's Comments	– – –				
Reply Card	– – –				
Inside Back Cover (Blank)	– – –				
Back Cover	– – –				

*Asterisks indicate pages changed, added or deleted by the current change. The issue date for the change appears in place of the previous issue date at the bottom of each changed page included in the change package. Where a change involves a technical correction or the addition of new material, a vertical line appears at the appropriate place in the margin of the affected page. Deletions and editorial corrections are not specifically indicated.

Stock Number: 87601194A

Issue A: 15 June 1984

PREFACE

This manual describes diagnostic tests for use in manufacturing and field servicing the Motorola System 6300 Computer, which contains one 68010 processor board.

You can select an entire test or an individual subtest, or add new tests and subtests. In addition, test sequences, which are combinations of specific tests and subtests, execute using a single command.

Section 1 of this manual describes how to load and run the diagnostics. In addition, Section 1 describes the interactive help feature, special hardware requirements, and the error reporting modes.

Section 2 describes each test and its subtests in detail. Section 2 also discusses error output and notes the parameter values that you specify to run each test. In addition, this section provides a complete list of errors and output for each test.

Section 3 describes how an experienced programmer adds a new test.

Section 4 describes the general error conditions that can arise when any of the tests are executing.

Appendix A provides a list of test sequences that include multiple tests.

Appendix B provides output for the default test sequence.

Appendix C recommends a sequence for testing the System 6300 Computer.

Appendix D describes how to use the RS-232-C emulation in the Fastrak 30 Workstation's read only memory (ROM) to run diagnostics when the Fastrak 30 Workstation cannot be downloaded from the Winchester disk drive (for example, prior to installation).

NOTE ON PARAMETER NAMES

Throughout this manual, optional parameter names are enclosed in square brackets ([]).

Motorola, Inc.
10700 North De Anza Boulevard
Cupertino, California 95014

CONTENTS

LIST OF TABLES

SECTION 1: RUNNING THE DIAGNOSTICS

This section describes the System 6300 diagnostics and how to run them.

TYPES OF DIAGNOSTICS

There are three types of System 6300 diagnostics--subtests, tests, and test sequences.

Subtests and Tests

Subtests are simple checks. For example, one subtest verifies that timer 0 is working correctly. A test is a group of related subtests. For example, the memory test comprises write 0's, write 1's, and write addresses subtests; and the parity test consists of read and execute parity error subtests.

The specifications for both tests and subtests are pairs of positive integers. For a subtest, the first number of the pair is the test number, and the second number is the subtest number. For a test, the pair consists of the test number and a zero. For example, in the following pairs:

```
8,5
8,0
```

the first pair specifies subtest 5 of test 8. The second pair specifies test 8 (i.e., all subtests available for test 8). (The default for the second value is zero.)

Test Sequences

Test sequences are predefined combinations of tests and subtests. Some test sequences (for example, the parity test sequence) consist of a single test. Other test sequences (for example, the default test sequence) consist of multiple tests.

Alphabetic characters denote test sequences. To run a test

sequence from a terminal, specify only the command letter that denotes the given test sequence. For example, to execute the d default test sequence, type only the letter d and press the RETURN key.

Appendix A lists the test sequences that include multiple tests.

SPECIAL HARDWARE REQUIREMENTS

There are two special hardware requirements for running System 6300 Computer diagnostics:

1. To use a terminal, attach the terminal via a cable containing a line for data set ready (DSR) to the RS-232-C port. The diagnostics use DSR to determine that a terminal is connected.

2. To run the RS-232-C tests, connect a terminator plug for each RS-232-C port that you test. On a basic system executing in attended mode, the only requirement is connector B with the following connections:

   pin 2 to pin 3
   pin 4 to pin 5
   pin 8 to pin 20.

LOADING THE DIAGNOSTICS

With the workstation downloaded and working, use the following steps to loads the diagnostics:

1. Power on.

2. Insert the floppy diskette containing the diagnostic program in the appropriate drive and close the door.

3. Wait for the disk to initialize.

4. Press the RESET button.

If a terminal is connected, the following header/prompt appears on the display:

```
MINIFRAME DIAGNOSTICS - V X.X.X
Parity interrupts disabled
Memory tests will begin at XXXXX, end at YYYYY

command>
```

where

X.X.X	is the version number of the System 6300 diagnostics.
XXXXX	is the memory start address in hexadecimal at which memory tests start.
YYYYY	is the address of the last valid 32-bit word in memory.

Make sure that the memory end address (in hexadecimal) and the amount of memory in your system correspond as follows:

```
  7FFFC=512 kbyte
  FFFFC=1 Mbyte
 17FFFC=1.5 Mbyte
 1FFFFC=2 Mbyte
```

ERROR REPORTING MODES

The error reporting modes for running the diagnostics are

- Unattended
- Attended
  - Interactive
  - Long
  - Continuous.

Unattended mode is the only output mode available, where using a terminal is infeasible (e.g., in an oven in a manufacturing environment). In unattended mode, you can run only the default set of diagnostics. If a error occurs, unattended mode lights the top red LED in the back panel and waits for an operator to connect a terminal to print more information about the error. The top yellow back panel LED indicates taking the interrupt.

If a terminal is connected, you can run the diagnostics in any of the attended modes. In the attended modes, you enter commands at the terminal to run the diagnostics and to specify the desired

attended mode--either interactive, long, or continuous. (See the command format subsection below for more information.) Like unattended mode, the attended modes display errors in the top red LED in the back panel and indicate taking the interrupt in the top yellow LED. The attended modes are described in more detail below.

Interactive mode is the default operation mode, if a terminal is connected to the processor. If an error is detected in interactive mode, the diagnostic stops executing, and the error is displayed on the terminal. The system then reinitializes and is ready for a new test. If the terminal is removed while a test is running, the test continues and errors are handled as if in unattended mode.

Long mode keeps track of errors by test and subtest using an internal table. At the completion of the specified test, long mode prints the total number of errors detected, the number of errors detected by each test and subtest, and as many of the applicable error descriptions as can fit in the buffer.

Continuous mode is the same as interactive mode except that after displaying an error, continuous mode returns to continue the test.

CAUTION

Be careful when using continuous mode, since it forces the diagnostic to continue after detecting error conditions.

RUNNING THE DIAGNOSTICS FROM A TERMINAL

If a terminal is connected to your system, you can run the diagnostics by entering commands. In addition, you can optionally run the diagnostics in four general modes and access an interactive HELP feature.

Inputting Diagnostic Commands

If a terminal is connected, a "command>" prompt indicates that the diagnostic program is ready to accept your commands. This prompt always appears after a diagnostic completes successfully or after an error occurs.

After you type a diagnostic command, press the RETURN key to enter it. To correct a typographical error, use the BACKSPACE key to move the cursor toward the left side of the screen, so you can type the correct data over the error.

General Output Modes

The four modes for running the diagnostics are parity, page, line echo, and disk ignore. The modes are turned on and off by typing commands. Since all modes function as switches, the same commands disable the modes that enable them. For example, if you typed PM (uppercase or lowercase) to enable a mode, type PM (uppercase or lowercase) again to disable it.

By default, parity interrupts are disabled when you initially load the diagnostics. To enable parity interrupts, type the letters PE (uppercase or lowercase) after the "command>" prompt and press the RETURN key. (To disable parity interrupts if they are on, type PE (uppercase or lowercase) after the "command>" prompt and press the RETURN key.) Parity interrupts are always disabled after an error message.

By default, the diagnostic output scrolls automatically (page mode is off). With page mode on, you can scroll output manually to allow more time for reading output. To turn page mode on, enter the letters PM (uppercase or lowercase) after the "command>" prompt. With page mode on, the following prompt appears at the bottom of each full page of output:

```
<1=next line, e=disable page mode, anything else=next
  page>
```

Press 1 to see the next line, press e to disable page mode, or press any other terminal key to go to the next page.

Line echo prints all display output on the printer, if one is present. To enable line printer echo, enter LE (uppercase or lowercase) after the "command>" prompt. If any of the diagnostics return an error, line echo automatically disables. (Line echo is not valid for the clock test.)

By default, disk ignore mode is off. With disk ignore mode off, if a disk drive is not present and you try to run a test sequence that includes a disk test, the diagnostics return an error and terminate. With disk ignore mode on, the diagnostics return a message and continue to run. To turn disk ignore mode on, enter DI (uppercase or lowercase) after the "command>" prompt.

Using the HELP Feature

If you are using a terminal, you can access the interactive HELP feature for online assistance in running the diagnostics. HELP provides the following information:

- Command format
- A list of test sequences
- Lists of tests and subtests.

Command Format and List of Test Sequences

To display the command format and a complete list of available test sequences, type a question mark (?) or back slash (/) after the "command>" prompt and press the RETURN key. For example,

```
command> ?
```

displays the command format and the list of test sequences. (Appendix A lists the test sequences that include multiple tests.)

List of Tests and Subtests

To request a list of tests and subtests in a given test sequence, type a question mark or back slash followed by the command letter denoting the test sequence and press the RETURN key. For example,

```
command> ?d
```

displays the tests that execute in the default test sequence. If the test sequence consists of only one test, the HELP message contains the test number and a list of subtests.

NOTE

The diagnostic program is case insensitive--typing uppercase D results in the same response as typing lowercase d.

The Diagnostic Command

The commands that run diagnostics specify

- The test sequences, tests, or subtests to execute.
- The number of times each test sequence, test, or subtest executes, and in which order.
- The error reporting mode.

The command syntax for running a simple diagnostic is as follows:

```
[RepeatCount][:]Test[,Subtest][L/C]
```

where

RepeatCount	is a positive integer that specifies the number of times the diagnostic named in the command executes.
:	is either a colon (:) or a period (.) if the next parameter is an integer. A period is required if the Fastrak 30 Workstation's RS-232-C ROM emulation is used.
Test	is either a letter that specifies the test sequence to run or is a positive integer that specifies the number of the test to run. If this parameter is an integer, a colon (:) always precedes.
,SubtestNumber	is a positive integer that specifies the individual subtest to run. This parameter is allowed only if the previous parameter is an integer not a letter). If SubtestNumber is 0 (the default), the entire test specified executes. If both test and subtest numbers are specified, the required delimiter between the integers is a comma (,).

L/C	is a letter that specifies the error reporting mode for the diagnostic and can be either L (or 1) for long mode or C (or c) for continuous mode. This parameter is optional and, if omitted, defaults to interactive.

A simple diagnostic executes RepeatCount times before the "command>" prompt reappears. To combine multiple diagnostics into a single command, separate the simple diagnostics with semicolons as follows:

```
SimpleDiag;SimpleDiag;SimpleDiag;...;SimpleDiag
```

You can specify as many simple diagnostics in a single command as can fit on one line. You can also add a repeat count that applies to the entire sequence of diagnostics in the command. To do this, start the command by entering the number of repetitions followed by a vertical bar (|) or back slash (/). For example

```
RepeatCount|SimpleDiag;SimpleDiag;SimpleDiag;...;SimpleDiag
```

executes the sequence of diagnostics specified in the command RepeatCount repetitions. Individual diagnostics can still have repeat counts that apply only to them.

An example of a simple diagnostic command could be

```
command> m
```

This command executes test sequence m one time in interactive mode. The following commands

```
command> :8
command> :8,Ø
```

both execute all subtests available for test 8. (Note the colon before the test number.) Another command

```
command> 1ØØ:8,6C
```

executes subtest 6 of test 8 in continuous mode 1ØØ times. The following more complicated command:

```
command> 5Ø|2:Ø,1L;3:1,1L
```

executes subtest 1 of test Ø twice in long mode, followed by three repetitions of subtest 1 of test 1 in long mode. This entire sequence repeats 5Ø times. The last example

```
command> 2Ø|2m;3:1,1L;p
```

executes a mix of test sequences and subtests.

SECTION 2: TESTS AND SUBTESTS

This section presents the tests in numeric order and describes the subtests available for each test. The tests are listed in Table 2-1. In addition, this section notes which tests display prompts that ask you to specify parameter values. (Each prompt specifies a default value.)

Table 2-1. Diagnostic Tests

Test Number	Test Name	Sequence*	Subtests in Sequence
Ø	Memory Read, Write, and Refresh Test	m	all subtests
1	Map Translation Test	p	"
2	Page Fault Test	f	"
3	Parity Test	r	"
4	Clock Test	c	"
5	User I/O Error Test	u	"
6	Winchester Disk Test	w	subtests 1,4,5
7	Reserved		
8	Floppy Disk Test	o	subtests 1,4,5
1Ø	Map RAM Test	a	all subtests
11	RS-232-C (8274) Test	h	"
12	Page Protection Test	e	**
13	Line Printer Test	l	all subtests
14	Fast Communication Port Test	k	subtest 1

*For a detailed explanation of test sequence, see the discussion in Section 1.

**This test does not include subtests.

This manual does not describe procedures for correcting hardware problems. For information regarding remedial procedures, refer to the discussion of the theory of operation in the Technical Manual.

The diagnostics produce two types of errors: operator and hardware. Operator errors occur when the diagnostic is unable to

recognize the test specification input by the operator. If an error is present in the command line, the name of the routine that gives the message appears in parentheses (). The second type of error reports hardware malfunctions detected by the diagnostics. These error messages list the diagnostic that that detected the error, and give some information about the error. For example, an error message could be

```
Test:10, Subtest:5: Memory overlap error: location and
  contents read back are  4003FE,  1F2E
```

In this example, subtest 5 of test 10 discovered a memory overlap error at 4003FE.

The following discussions of each test include a list of all possible errors, and, where practical, output for a sample test run. In the lists of errors, x is a hexadecimal value; d is a decimal value; and s is a string. Some of the tests ask questions that require a response before the test continues. In the sample outputs, user input is shown in boldface and is underlined. Remember that you must press the Return key to enter both default values and your typed responses.

TEST 0: MEMORY READ, WRITE, AND REFRESH TEST (m)

The memory test checks every location in memory. The test comprises the following subtests:

1. Write and read 0's. (32-bit)
2. Write and read -1's. (32-bit)
3. Write and read aaaaaaaa's. (32-bit)
4. Write and read 55555555's. (32-bit)
5. Write and read addresses. (32-bit)
6. Write and read complements of addresses. (32-bit)
7. Write and read 0's. (16-bit)
8. Write and read -1's. (16-bit)
9. Write and read aaaa's. (16-bit)
10. Write and read 5555's. (16-bit)
11. Write and read addresses. (16-bit)
12. Write and read complements of addresses. (16-bit)

Subtests 1-4 and 7-10 check to make sure that each memory cell is working properly. The remaining subtests, subtests 5, 6, 11, and 12, make sure that the address lines are functioning correctly. Each subtest has two versions: a 32-bit version (subtests 1-6) and a 16-bit version (subtests 7-12).

First, all subtests of this test write and read data to every location in memory. Next, every location in memory is read a second time to test the refresh cycle. If any of the read and writes do not match, an error returns.

Subtests 1 and 7 write and read a zero (Ø) to each location in memory. Subtests 2 and 8 are the same as subtests 1 and 7 except that they write and read a -1 instead of Ø. Subtests 3 and 9 write bit patterns of alternating Ø's and 1's to each address in memory. Subtests 4 and 1Ø write the one's complement of subtests 3 and 9. Subtests 5 and 11 write and read the addresses of each location in memory; subtests 6 and 12 write and read the complements of addresses to every location in memory.

Errors

Test Ø can return the following errors:

```
Memory error at Address x: Wrote Ø's; Read Back x.
Memory error at Address x: Wrote 1's; Read Back x.
Memory error at Address x: Wrote x's; Read Back x.
Memory overlap error: Location and contents read back are x x.
```

Test Output

Output for a run of the memory test is as follows:

```
MEMORY SUBTEST 1
MEMORY SUBTEST 2
MEMORY SUBTEST 3
MEMORY SUBTEST 4
MEMORY SUBTEST 5
MEMORY SUBTEST 6
MEMORY SUBTEST 7
MEMORY SUBTEST 8
MEMORY SUBTEST 9
MEMORY SUBTEST 1Ø
MEMORY SUBTEST 11
MEMORY SUBTEST 12
```

TEST 1: MAP TRANSLATION TEST (p)

The map translation test checks the status bits and page table entry swapping. The subtests are

1. Read and write memory test, check access and dirty bits.
2. Map tests (page table entry swapping).

Subtest 1 operates as follows:

1. Set register to status=valid (Ø1).
2. Read from the page addressed by register.
3. Read register and check to make sure that status=accessed (1Ø).
4. Write to the page addressed by the register.
5. Read register again and check to see if status=modified (11).

Subtest 2 checks mapping to the correct physical memory location and operates as follows:

1. Write to two locations on different pages.
2. Swap the page table entries for the two pages.
3. Read back the two locations and check to make sure that the values are swapped.

Errors

Test 1 can return the following errors:

```
Page dirty bit not set for page number.
Page mapping error: map number, logical address, and address
  read back are x x x.
Page access bit not set for page number.
Page access bits wrong:  page number and page bits are x x.
No interrupt on executing protected page.
```

Test Output

Output for a run of the map translation test is as follows:

```
MAP TRANSLATION TEST
MAP TRANSLATION SUBTEST 1
MAP TRANSLATION SUBTEST 2
```

TEST 2: PAGE FAULT TEST (f)

The page fault test verifies that (1) the 68Ø1Ø processor can successfully restart instructions and (2) pages having a "not valid" status code cannot be accessed. This test sets each map

register to "not valid" and then tries to write, read, and execute. Each of these attempts should generate a bus error, and the tests verify that a page fault returns. If not, the operation aborts, and an error message returns. The subtests are

1. Test page faults caused by reading and writing.
2. Test page faults caused by trying to execute.

Subtest 1 sets the page table entry to "not valid" and tries to read and write to the mapped memory. Subtest 2 sets the page table entry to "not valid" and tries to execute that page.

Errors

Test 2 can return the following errors:

```
Page fault does not occur when reads nonmapped page.
Page fault does not occur when executes nonmapped page.
No page fault received on write.
No page fault received on read.
Write/Read-back not continued properly.
No page fault received on execution test.
Page fault when none expected.
```

Test Output

Output for a run of the page fault test is as follows:

```
PAGE FAULT TEST
PAGE FAULT SUBTEST 1, Read/Write test
PAGE FAULT TEST
PAGE FAULT SUBTEST 2, Execution test
```

TEST 3: PARITY TEST (r)

The parity test checks the parity logic and makes sure that the processor, bus status registers (BSR's), and the general status register (GSR) handle exceptions correctly. The parity test subtests are

1. Reads a parity error through memory to check the BSR.
2. Execution of a parity error.

Subtest 1 forces a parity error at every memory address and takes the interrupt, thus forcing all addresses through the BSR. Subtest 2 writes an instruction containing a parity error to memory and tries to execute it.

Errors

Test 3 can return the following errors:

```
No Parity Interrupt at location x.
BSR incorrect after parity error at location x.
BSRØ=x.
BSR1=x.
No Parity Interrupt during execution test-User mode.
No Parity Interrupt during execution test-Supervisor mode.
BSR incorrect after parity error during execution.
BSRØ=x.
BSR1=x.
```

Test Output

Output for a run of the parity test for a 512 kbyte memory is as follows:

```
            PARITY TEST
             PARITY TEST - SUBTEST 1 WRITE/READ TEST
Reached Address 20000
Reached Address 30000
Reached Address 40000
Reached Address 50000
Reached Address 60000
Reached Address 70000
  PARITY TEST - SUBTEST 2 EXECUTION TEST
```

TEST 4: CLOCK TEST (c)

The clock test verifies that the three clocks contained in the Computer are working correctly. The subtests are

1. Test timer Ø.
2. Test timer 1 and Counter 2.
3. Test timer Ø and 8259A.

Subtest 1 verifies that timer Ø (programmable timer) causes interrupts. This subtest operates as follows:

1. Program the timer.
2. Take one interrupt from timer Ø.

Subtest 2 verifies timer 1 (60-Hz clock) and Counter 2 (tick

count). This subtest operates as follows:

1. Run timer 1 for 10 seconds.

2. Every second, print number of seconds remaining (countdown).

3. Check to see that Counter 2 matches the number of interrupts received.

Subtest 3 programs timer 0. Subtest 3 operates as follows:

1. Take 6,000 interrupts.
2. Reset timer between each interrupt.

Errors

Test 4 can return the following errors:

Stray interrupt from fast clock.
No interrupt generated by the fast clock.
Stray interrupt from slow clock.
N ticks missed.
N more clock interrupts processed than count ticks (N/2) recorded.

Test Output

Output for a run of the clock test is as follows:

```
                CLOCK TEST
                Subtest 1 - Timer 0 TEST

                Subtest 2 - Timer 1/Counter 2 TEST
    9 8 7 6 5 4 3 2 1

                Subtest 3 - Timer 0/8259 TEST
```

TEST 5: USER I/O ERROR TEST (u)

The user input/output error test makes sure that neither the external processor registers nor input/output registers, located in the input/output space (all virtual memory above 4 Mbytes), can be accessed in user mode. Any attempt to access these registers in user mode should generate an exception. This test also verifies that the general status register (GSR) and the bus status registers (BSR's) are set correctly. The subtests are

1. Accesses processor register address space.
2. Accesses input/output address space.

Subtest 1 tries to access the GSR and should cause an exception; subtest 2 tries to access an I/O address register and also should generate an exception.

Errors

Test 5 can return the following error:

No interrupt on User I/O test.

Test Output

Output for a run of the user I/O error test is as follows:

```
USER I/O ERROR TEST
      USER I/O ERROR SUBTEST 1
      USER I/O ERROR SUBTEST 2
```

TESTS 6 AND 8: DISK TESTS -- WINCHESTER AND FLOPPY (w AND o)

Systems contain a Winchester hard disk drive and a floppy disk drive. (For more information about disk drives, see the Technical Manual.) The System 6300 disk tests are

Test 6. Winchester Disk Test (w)
Test 8. Floppy Disk Test (o)

The same subtests are provided for both disks except that an additional subtest, read track N, is provided for the floppy disk. The subtests are

1. Recal.
2. Format (interactive).
3. Write over all the disk with sector address data.
4. Read all the disk.
5. Random seeks with read sector.
6. Write multiple sectors (interactive).
7. Read multiple sectors (interactive).
8. Spare a sector (interactive).
9. Force DMA faults.
10. Initialize VHB and BBT.
11. Write VHB and BBT after aborted test 3.
12. Print VHB and BBT.
13. Download to disk.
14. Toggle alien disk mode.

15. Read track N.
16. Receive from fast comm and write to disk.
17. Read from disk and send via fast comm.
18. Reserved.
19. Reserved.
20. Reserved.
21. Report on unreadable blocks.
22. Compare RS-422 input to local disk.
23. Surface test.
24. DMA load test.
25. Sort BBT.

Again, note that subtest 15 is used only to test the floppy disk.

Subtest 1: Subtest 1, Recal (recalibrate), initializes the disk, brings the head back to the beginning of the disk (sector 0), and tries to read and verify the first two blocks of the disk. The first block is the volume home block (VHB), which contains descriptive information (number of cylinders, number of tracks per cylinder, and number of sectors per track), and the second block contains the bad block table (BBT). Subtest 1 can result in a number of error conditions, but the two most common are Response=4 (Can't recal) and Response=1 (Can't read the volume home block).

Subtest 2: Subtest 2, Format, is more a utility than a diagnostic. First, subtest 2 performs hardware formatting of the disk. Next, it writes all the sector headers on the disk and the volume home block according to your specifications, and any bad blocks discovered. This subtest prompts for header information for the volume home block, and an operator should be present at the terminal to enter this information.

Subtest 3: Subtest 3 writes sector address data over the entire disk and then performs a check/read to make sure it read the same data it wrote. In interactive mode, subtest 3 allows you to specify the data pattern in hex, the start block, the number of blocks to transfer, the maximum transfer size, and the number of subtest repetitions. In unattended mode, the subtest runs using defaults. The default pattern is a generated address pattern based on where you are in the disk. By default, subtest 3 writes over the entire disk, reading the data after it has been written (in

increments of the selected transfer size). After completing the write phase, the same area of disk is read again to verify that no addressing problems exist. **NOTE:** Specifying the blocks that contain the diagnostics will cause the diagnostics to be overwritten.

Subtest 4: Subtest 4 performs only the read part of subtest 3 (see above). Subtest 4 reads a disk, checking for either the default pattern or a hex pattern that you specify. Subtest 4 asks all the same questions that subtest 3 asks (see above).

Subtest 5: Subtest 5 performs random seeks to different sectors in the disk and then reads the sectors, checking to make sure that seek works consistently and that the head goes to the correct place. Subtest 5 allows you to specify the maximum block number and the number of subtest repetitions (random seeks) to execute.

Subtests 6 and 7: Subtests 6 and 7 respectively write and read multiple sectors to the disk from memory. These subtests allow you to specify the area of memory to write to disk, or, conversely, to read from some place on the disk. Parameters are the start address in memory, the start block on disk, and the number of blocks to transfer. **NOTE:** For subtest 6, specifying the blocks that contain the diagnostics overwrite the diagnostics.

Subtest 8: Subtest 8, more a utility than a diagnostic, allows you to specify bad blocks in the bad block table, so that you can avoid using known bad blocks. Subtest 8 asks you to specify either the track number (i.e., cylinder number and head number) and byte number, physical sector, or logical block of the bad block.

Subtest 9: Subtest 9 forces parity errors and page faults during a disk transfer. This subtest verifies that the appropriate nonmaskable interrupt (NMI) occurs and that the GSR and BSR are functioning correctly.

Subtest 10: Subtest 10 allows you to create the volume home block and the bad block table. This subtest is the same as subtest 2 except that subtest 10 does not format the disk, saving

the data previously on the disk. Subtest 1Ø prompts for the same information requested by subtest 2.

Subtest 11: Subtest 11 restores the volume home block and bad block table, if they are lost as a result of running either subtest 3 or 23. Normally, subtest 3 automatically saves the volume home block and bad block table in an internal buffer. However, if subtest 3 or 23 inadvertently writes over the volume home block and bad block table (which is the case with the default parameters), you can run subtest 11 to rewrite the volume home block and the bad block table automatically when the test completes. If subtest 3 aborts for any reason, you should immediately run subtest 11 to rewrite the volume home block and bad block table.

Subtest 12: Subtest 12 prints the contents of the volume home block and bad block table.

Subtest 13: Subtest 13 is reserved.

Subtest 14: Subtest 14 toggles the alien mode flag for the disk type specified. When the alien mode flag is set, subtests 1, 6, and 7 interpret the disk as a nonstandard format disk (i.e., no volume home block and different bad block handling). No other subtests are valid on a disk that is in alien mode. The alien mode flags for the various disks are separate and can be set and cleared independently. Subtest 14 asks the same questions asked by subtest 2 so that the disk driver can drive the disk.

Subtest 15: Subtest 15, for floppy disks only, reads an arbitrary floppy track that you specify. (Subtest 15 does not display the data.)

Subtests 16 and 17: Subtests 16 and 17 receive and transmit, respectively, data stored on disk through the fast communication (RS-422) port. Like subtests 2 and 3 of the fast communication port test, subtests 16 and 17 require two Computers: one to send data and one to receive it. Subtests 16 and 17 prompt for the disk start block and the number of blocks to be transferred. Start the transmit subtest before the receive subtest.

Subtests 18-21: Tests 18-2Ø are reserved.

18 : Disk Test Disk upgrade

Subtest 21: Subtest 21 prompts for a start block number and a number of blocks. This portion of the disk is read and any blocks that cannot be read are displayed together with the failing status.

Subtests 22: Subtest 22, which verifies correct data transmission in conjunction with subtest 17, is identical to subtest 16 (see above), except that in addition to receiving data, subtest 22 compares the data with the data on disk and prints only the differences found. The input sequence for subtests 22 and 17 is the same as for subtests 16 and 17. Subtest 22 does not write to disk.

Subtest 23: Subtest 23, which surface tests a disk, is identical to subtest 3 (see above) except that instead of returning an error, subtest 23 adds any bad blocks found to the bad block table.

Subtest 24: Subtest 24 repeatedly loads the bus to see if any stray interrupts occur. Subtest 24 reads the first disk cylinder over and over while initializing RS-422 transmissions.

Subtest 25: Subtest 25 sorts the Bad Block Table.

Errors

Tests 6 and 8 can return the following errors:

```
Error during Disk Format:Response = x.
Not on Cylinder zero after Recal:Maybe on d,Response = x.
Can't Recal:Response = x.
Can't Write the new VHB:Response = x.
Can't Write the new Bad Block Table:Response = x.
Error on Write:response = x,Start Block = d.
Error on Re-Read;Response = x,Start Block = d.
Re-Read Data Fail:Start Block = d.
Error on Read:Response = x,Start Block = d.
Bad Block Table Overflow when adding Sector d.
Bad Block Table:Multiple use of alternate d.
Disk is not System 6300 Format.
No parity error on disk DMA.
Improper BSR after disk DMA parity error.
Improper BSR after disk DMA page fault.
No page fault on disk read.
Page status not = accessed after disk write, reg = x.
Page status not = modified after disk read, reg = x.
No Bad Block Recovery for Floppy Disk.
```

```
Disk Error on write: response=x.
Disk Error on Read Track: response=x.
Read Track time out.
Invalid mode switch.
```

Test Output

Output for the floppy disk drive subtests follows.

Output for Subtest 1

```
            DISK TEST Floppy

(Subtest 1) Recal Disk.
```

Output for Subtest 2

```
            DISK TEST floppy

(Subtest 2) Format Disk.
Do you want to format the Floppy disk (erasing contents)?y
Give # of Cylinders (RETURN = default of 80):
Give # of Tracks per Cylinder (RETURN = default of 2):
Give # of Sectors per Track (RETURN = default of 8):
Cylinders = 80, Tracks = 2, Sectors = 8
Give Pack Name (RETURN = default of Floppy):
Give Density (RETURN = default of 2):
Give Step Rate (RETURN = default of 0):
Give Size of Partition 0 (RETURN = default of 640):
Do you want a Loader (Answer [Y/y] or [N/n]:y
Give start Logical Block (RETURN = default of 2):
Give size in Blocks (Default = 12):
Do you want a Dump Area (Answer [Y/y] or [N/n]:n
Do you want a Down Load File (Answer [Y/y] or [N/n]:n
Do you want a Bootable Program (Answer [Y/y] or [N/n]:y
Do you want it in the Reserved Area (Answer [Y/y] or [N/n]:y
Give start Logical Block (RETURN = default of 14):
Give size in Blocks (Default = 200):
Give Interlace Factor (Default = 1):
The Bad Block Table contains 0 entries.
The Bad Block Table contains 0 entries.
```

Output for Subtest 3

```
                DISK TEST Floppy

(Subtest 3) Disk Write and Check Read test
Do you want to write to the Floppy disk?y
Volume Name: Floppy
Give Data Pattern in hex (RETURN = default of address):
Give start Logical Block (RETURN = default of 16):
Give # of Blocks to transfer (RETURN = default of 624):
Give Max transfer size in Blocks (RETURN = default of 4):
Give # of Times to Repeat  (RETURN = default of 1):

Data Pattern = ØXFFFFCFC7, Start Block = 16, # Blocks = 624,
Increment = 4, Repeats = 1
Floppy:Initiating Check Read for pass Ø
The Bad Block Table contains Ø entries.
```

Output for Subtest 4

```
                DISK TEST Floppy

(Subtest 4) Disk Read test.
Volume Name: DIAG24
Give Data Pattern in hex (RETURN = default of address):
Give start Logical Block (RETURN = default of 16):
Give # of Blocks to transfer (RETURN = default of 624):
Give Max transfer size in Blocks (RETURN = default of 4:
Give # of Times to Repeat  (RETURN = default of 1):

Data Pattern = ØXFFFFCFC7, Start Block = 16 # Blocks = 624
Increment = 4, Repeats = 1
Test:8, Subtest:4: Floppy=Check-Read Data Fail:
Start Block = 16, Byte=Ø, Received E2Ø6E, Expected FFFFEFFF
```

Output for Subtest 5

```
                DISK TEST Floppy

(Subtest 5) Random Seek test.
Volume Name: Floppy
Give Maximum Block # (RETURN = default of 64Ø):
Give # of Times to Repeat (RETURN = default of 1ØØ):

Maximum Block = 64Ø, Repeat count = 1ØØ
```

Output for Subtest 6

```
          DISK TEST Floppy

(Subtest 6) Disk Write function.
Do you want to write to the Floppy disk?y
Volume Name: Floppy
Give Buffer Address in hex (RETURN = default of ØX33ØØØ):
Give start Logical Block (RETURN = default of Ø):
Give # of Blocks to transfer (RETURN = default of 64Ø):

Buffer Address = ØX33ØØØ, Start Block = Ø, # Blocks = 64Ø
```

Output for Subtest 7

```
          DISK TEST Floppy

(Subtest 7) Disk Read function.
Volume Name: Floppy
Give Buffer Address in hex (RETURN = default of ØX33ØØØ):
Give start Logical Block (RETURN = default of Ø):
Give # of Blocks to transfer (RETURN = default of 64Ø):

Buffer Address = ØX33ØØØ, Start Block = Ø, # Blocks = 64Ø
```

Output for Subtest 8

```
          DISK TEST Floppy

(Subtest 8) Spare a sector function.
Volume Name: Floppy
Specify Spare: [1/RETURN]=(Cylinder, Head and Byte)
[2]=Physical Sector
[3]=Logical Block:
Give Cylinder Number:
Give Head Number:
Give Byte Number:
No Bad Block Recovery for Floppy Disk.

Give Cylinder Number    :7Ø
Give Head Number        :1
Give Byte Number        :3
Added Bad Block: Cylinder 7Ø, Track 1, Sector Ø.
        Used Track 588 as the Alternate.
Another (Default: RETURN = Yes) :n
The Bad Block Table contains 1 entries.
```

Output for Subtest 9

```
                DISK TEST Floppy

     (Subtest 9) Force DMA Faults.
```

Output for Subtest 10

```
                DISK TEST Floppy

     (Subtest 10)  Initialize VHB and BBT.
     Do you want to change the VHB on the Floppy drive:y
     Give # of Cylinders (RETURN = default of 80):
     Give # of Tracks per Cylinder (RETURN = default of 2):
     Give # of Sectors per Track (RETURN = default of 8):

     Cylinders = 80, Tracks = 2, Sectors = 8
     Give Pack Name (RETURN = default of Floppy):
     Give Step Rate (RETURN = default of 0):
     Give Size of Partition 0 (RETURN = default of 640):
     Do you want a Loader (Answer [Y/y] or [N/n]:y
     Give start Logical Block (RETURN = default of 2):
     Give size in Blocks (Default = 12):
     Do you want a Dump Area (Answer [Y/y] or [N/n]:n
     Do you want a Down Load File (Answer [Y/y] or [N/n]:n
     Do you want a Bootable Program (Answer [Y/y] or [N/n]:y
     Do you want it in the Reserved Area (Answer [Y/y] or [N/n]:
     Give start Logical Block (RETURN = default of 14):
     Give size in Blocks (Default = 200):
     Give Interlace Factor (Default =1):
     The Bad Block Table contains 0 entries.
```

Output for Subtest 11

```
                DISK TEST Floppy

(Subtest 11) Write VHB and BBT after aborted Test 3.
The Bad Block Table contains 0 entries.
```

Output for Subtest 12

```
                DISK TEST Floppy

     (Subtest 12) Print VHB.
     Volume Name:  Floppy
     80 Cylinders. 2 Heads per Cylinder.
     There are 8 Physical Sectors (of 512 bytes) per Track.
     16 Physical Sectors per Cylinder, 1280 Physical Sectors per Disk.
```

```
There are 4 Logical Blocks (of 1024 bytes) per Track,
  8 Logical Blocks per Cylinder, 640 Logical Blocks per Disk.
The Floppy is Double density
The Step Rate supplied to the Controller is 0.
Partition 0: start Track=0, size (in Blocks)=640
Loader starts at Block 2 (size=12 Blocks).
Bad Block Table starts at Block 1 (size=1 Blocks).
The Bad Block Table contains 0 entries.
```

Output for Subtest 14

```
          DISK TEST Floppy

(Subtest 14) Toggle Alien Disk mode.
Give # of Cylinders (RETURN = default of 80):
Give # of Tracks per Cylinder (RETURN = default of 2):
Give # of Sectors per Track (RETURN = default of 8):

Cylinders = 80, Tracks = 2, Sectors = 8
Give Pack Name (RETURN = default of Floppy):
Give Density (RETURN = default of 2):
Give Step Rate (RETURN = default of 0):
Give Interlace Factor (Default = 1)
Floppy:  Alien mode enabled
```

Output for Subtest 15

```
          DISK TEST Floppy

(Subtest 15) Floppy Read Track.
Give Required Track Number (Default = 0)
Give Required Density (Default = 2)
68640 words of Track 0 data is at address 33000
```

Output for Subtest 16

```
          DISK TEST Floppy

(Subtest 16) Write to disk from fast comm port.
Volume Name:  DIAG24
Give start Logical Block (RETURN = default of 0):
Give # of Blocks to transfer (RETURN = default of 640):
```

Output for Subtest 17

```
                DISK TEST Floppy

     (Subtest 17) Send from disk to fast comm port
     Volume Name: Floppy
     Give start Logical Block (RETURN = default of Ø):
     Give # of Blocks to transfer (RETURN = default of 56Ø):
```

TEST 9 (RESERVED)

Test 9 is reserved.

TEST 1Ø: MAP RAM TEST (a)

The map RAM test verifies the memory used for virtual memory translation tables. The subtests are:

1. Write and read Ø's. (16-bit)
2. Write and read -1's. (16-bit)
3. Write and read aaaa's. (16-bit)
4. Write and read 5555's. (16-bit)
5. Write and read addresses. (16-bit)
6. Write and read complements of addresses. (16-bit)

These subtests correspond exactly to subtests 7-12 of the memory tests (see above), except that the map RAM tests ignore the three unused bits in the page table entries.

Errors

Test 1Ø can return the following errors:

```
Memory error at Address x:  Wrote Ø's; Read Back x.
Memory error at Address x:  Wrote 1's; Read Back x.
Memory error at Address x:  Wrote x's; Read Back x.
Memory overlap error:  Location and contents read back are
  x x.
```

Test Output

Output for a map RAM test is as follows:

```
MAP RAM TEST
MAP RAM SUBTEST 1
MAP RAM SUBTEST 2
MAP RAM SUBTEST 3
MAP RAM SUBTEST 4
MAP RAM SUBTEST 5
MAP RAM SUBTEST 6
```

TEST 11: RS-232-C (8274) TESTS (h)

Test 11 verifies the RS-232-C port. Note that to run this test, you need to connect a loopback plug to Port B with the pin connections listed in the special hardware requirements subsection of Section 1. The subtests for test 11 are

1. Test 8274 control options.
2. Data transfer test.
3. Error condition test.

Subtest 1 programs the Intel 8274 chip and sends a small amount of data through the RS-232-C port in loopback mode, which means that what goes out Port B comes back in Port B. This subtest executes at all available speeds. Subtest 2, also performed in loopback mode, transfers large amounts of data at 19,200 baud. Both subtests take interrupts. Subtest 3 forces the various error conditions, such as framing errors, that can occur via loopback, and makes sure that the 8274 handles and detects errors properly.

Errors

Test 11 can return the following errors:

```
Carrier not on.
Didn't receive frame Error.
Corrupted data at specified byte (RX/TX data): x.
Data not transmitted after ~2 seconds for d.
Not all the data has been received, so far: x.
Unknown Port.
Bad RS232 port (d).
TX underrun on tty d.
```

Test Output

```
          8274 TEST
(Subtest 1) Test 8274 Control options.
          8274 TEST
(Subtest 2) Data Transfer test.
          8274 TEST
(Subtest 3) Error Condition test.
```

TEST 12: PAGE PROTECTION TEST (e)

The page protection test checks all combinations of the various factors used to control page access, ensuring that the combinations permit access and generate exceptions as required. This test also verifies instruction restart. The test sets the page protection variables to a combination and then tries to access the page. The variables tested are page status code (not valid/valid), write enable, (set/not set), address (< 512K/> 512K/virtual), and mode (supervisor/user). Some combinations should always generate an exception (for example, any combination that tries to write below 512K in user mode). The page protection test contains no subtests.

Errors

Test 12 can return the following errors:

No Page fault received on write.
No Page fault received on read.
Write/Read-back not coninued properly.
No page fault received on execution test.
Page fault when none expected.

Test Output

```
PAGE PROTECTION TEST
```

TEST 13: LINE PRINTER TEST (1)

The line printer test checks the status register of the line printer and verifies that a line printer can print data

correctly. The subtests are

1. Check the line printer status register and print the observed status.

2. Print a rotating pattern on the printer using interrupts. Check the status on each interrupt.

Subtest 1 checks the status of the data/status register, returns the status, and makes sure that the status is valid (either line printer present, line printer selected, or line printer out of paper). For each repetition, subtest 2 prints one full page of wallpaper patterns, consisting of 132-column rotations of the ASCII character set.

Errors

Test 13 can return the following errors:

Unrecognized LP status x.

Test Output

Output for a run of the line printer test is as follows:

```
            LINE PRINTER TEST
            LINE PRINTER SUBTEST 1, Status Test.
LP Selected.
            LINE PRINTER SUBTEST 2, Transfer Test.
```

At this point in the test, the line printer prints a full page of rotations of the ASCII character set.

TEST 14: FAST COMMUNICATION (RS-422) PORT TEST (k)

Test 14 verifies the fast communication (RS-422) port. Like subtests 16 and 17 of the disk tests, test 14 requires two computers: one to transmit data and the second to receive it. The computer receiving data should be ready to receive data before the transmission subtest starts. The subtests are

1. Auto send/receive.
2. Transmit increasing sized packets.
3. Receive the packets sent by subtest 2.
4. Transmit user specified block of memory.
5. Receive user specified block of memory.

Subtest 1 determines which computer will begin transmitting. Subtests 2 and 3, respectively, transmit and receive data packets through the fast communication port. The packet size for the first transmission is four words long. For each subsequent transmission, the packet size doubles. Transmissions continue until the packets reach a word count of 32,768 (2^{15}). Subtests 4 and 5, respectively, transmit and receive blocks of memory. For subtests 4 and 5, you must specify the start address in memory and the number of words to transfer. Start the receive subtests before the transmit subtests to avoid transmitting data before it can be accepted.

Errors

Test 14 can return the following errors:

Timeout on package receive complete. Packet size = d. d words left to transfer.
Received data invalid. Got d, expected d, transfering d words.
No read status changes received.
Invalid status while waiting for read complete, polling returned d.
Unknown synch down data received: x.
Fast comm line has uninterrupted carrier.
Carrier signal doesn't come on after clock disabled.
Timeout waiting for carrier to go after disabling clock.
Transmit underrun error on transfer.
Terminal count not received after read end of message.
Erroneous read status returned = x.
Loss of carrier before completion of RS-422 operation. Counter = d.
No carrier loss after terminal count on transmit.

Test Output

FAST COMMUNICATION PORT TEST

SECTION 3: ADDING A NEW TEST

This section describes how an experienced programmer adds a new test. The procedures for running new tests must conform to the procedures for running existing tests (see the discussion on how to run tests). The following three steps are necessary to add a new test:

1. Access the source. (You need a source license to access the source.)

2. Change the source. (Only an experienced programmer should alter the source in any way.)

3. Recompile the source using the **make** utility.

Each diagnostic must consist of two parts: the diagnostic itself, which calls and passes functions, and an initialization sequence, which disables any interrupts enabled by the diagnostic.

FUNCTIONS

The diagnostic function is called with a subtest number (int) as its single argument. If this integer equals Ø, then all subtests should run. The diagnostic itself must manage the meaning of this number and make sure that it is within range. The diagnostic must also update the global int CurSubtest to contain the current subtest number. A function error handles all error processing. The diagnostic must be prepared for this function to return when executing in long or continuous mode. The calling sequence is:

```
error(longstring,subtest,shortstring)
```

where

longstring	is a string that will be printed on a terminal, if a terminal is present.
subtest	is a positive integer that specifies the subtest.

shortstring	is a string saved for summary information in long mode.

Longstring and shortstring can be identical strings.

INTERRUPT HANDLING

For interrupt handling, you must insert in trap.c a case in the switch corresponding to the vector number on which interrupts for your test will occur. The procedure trap is primarily a large switch statement on the interrupt vector. At the top of the function trap(), the general status register (GSR) and bus status register (BSR) are saved with all the bits positive true (i.e., the sense of the negative true bits is inverted). Use a flag to indicate which test is running so that you can differentiate between interrupts which occur while you are running your test and other, spurious interrupts. Communication between the interrupt handler and the diagnostic function should be through global flags or data structures.

ADDING A TEST SEQUENCE HEADER

To add a test sequence header, add the following to the data structure TestList in the program main.c:

1. The test name (a char).
2. A short help description (a *char).
3. A long help description (also a *char).
4. The test sequence (a set of test, subtest pairs ended by (-1,Ø)).

In addition, you must add a pointer to the function in the array fn_array. The function's position in fn_array determines the test number.

Refer to the data structure TestList for examples.

SECTION 4: GENERAL ERRORS

General errors are errors that are not an expected part of a diagnostic, and indicate relatively serious system malfunctions. General errors can occur at any time during processing and while any diagnostic is executing. The three types of general errors are discussed in this section.

STRAY BUS ERRORS

A stray bus error is any bus error that is not an expected outcome of a diagnostic. This type of general error has the following format:

```
Unexpected bus error, GSR=x, PC=x, RPS=x
```

UNEXPECTED NONMASKABLE INTERRUPTS

Unexpected nonmaskable interrupts (NMIs) are high priority interrupts that are not an expected result (of any of the diagnostics) at the time they occur. An unexpected nonmaskable interrupt could originate at one of several locations, such as the parity board. Output for this type of general error indicates the origin of the error and has the following general format:

```
Unexpected NMI, PC=x, RPS=x
           GSR=x  BSRØ=x  BSR1=x
           interrupt from level d, PC=x

           RPS=x  GSR=x
           BSRØ=x  BSR1=x
```

MISCELLANEOUS UNEXPECTED INTERRUPTS

All unexpected interrupts that are not NMIs or bus errors represent the third category of general errors. **NOTE:** Subtest

24 of the disk test checks for stray interrupts. Output for this type of general error has the following format:

```
Unexpected interrupt from level d, PC=x, RPS=x,
          GSR=x  BSRØ=x  BSR1=x
```

APPENDIX A: LIST OF TEST SEQUENCES THAT INCLUDE MULTIPLE TESTS

System 6300 test sequences are predefined combinations of tests and subtests. The System 6300 diagnostic test sequences include three sequences that run multiple tests and subtests. This appendix lists the tests and subtests that execute in these sequences.

Default Test Sequence (d)

The following tests execute in the default test sequence (d):

1. Memory Read, Write, and Refresh Test
2. Map Translation Test
3. Page Fault Test
4. Parity Test
5. Map RAM Test
6. User I/O Error Test
7. Clock Test
8. RS-232-C (8274) Test
9. Fast Communication Port Test.

Functional Verification Test Sequence (v)

The functional verification test sequence (v) executes all tests included in the d test sequence (see above) except the fast communication port test and plus subtests 3 and 5 of the disk test. The v test sequence executes with disk ignore mode enabled and with default values so the operator does not have to be present to press the RETURN key to continue the tests.

Complete System Test Sequence (s)

The system verification test sequence (s) requires two systems to execute. In addition to all tests included in the d test sequence (see above), this test sequence executes the disk test. The s test sequence should be executed with disk ignore mode enabled and with default values so the operator does not have to be present to press the RETURN key to continue the tests.

APPENDIX B: OUTPUT FOR THE DEFAULT TEST SEQUENCE (d)

This appendix provides input and output for the default test sequence (d), which includes the following tests:

1. Memory Read, Write, and Refresh Test
2. Map Translation Test
3. Page Fault Test
4. Parity Test
5. Map RAM Test
6. User I/O Error Test
7. Clock Test
8. RS-232-C (8274) Test
9. Fast Communication Port Test.

```
command> d
           MEMORY TEST
           MEMORY SUBTEST 1
           MEMORY SUBTEST 2
           MEMORY SUBTEST 3
           MEMORY SUBTEST 4
           MEMORY SUBTEST 5
           MEMORY SUBTEST 6
           MEMORY SUBTEST 7
           MEMORY SUBTEST 8
           MEMORY SUBTEST 9
           MEMORY SUBTEST 10
           MEMORY SUBTEST 11
           MEMORY SUBTEST 12
           MAP RAM TEST
           MAP RAM SUBTEST 1
           MAP RAM SUBTEST 2
           MAP RAM SUBTEST 3
           MAP RAM SUBTEST 4
           MAP RAM SUBTEST 5
           MAP RAM SUBTEST 6
           PARITY TEST
            PARITY TEST - SUBTEST 1 WRITE/READ TEST
Reached Address 20000
Reached Address 30000
Reached Address 40000
Reached Address 50000
Reached Address 60000
Reached Address 70000
           PARITY TEST - SUBTEST 2 EXECUTION TEST
```

```
          MAP TRANSLATION TEST
          MAP TRANSLATION SUBTEST 1
          MAP TRANSLATION SUBTEST 2
          PAGE FAULT TEST
          PAGE FAULT SUBTEST 1, Read/Write test
          PAGE FAULT TEST
          PAGE FAULT SUBTEST 2, Execution test
          USER I/O ERROR TEST
USER I/O ERROR SUBTEST 1
USER I/O ERROR SUBTEST 2
          CLOCK TEST
          Subtest 1 - Timer Ø TEST

          Subtest 2 - Timer 1/Counter 2 TEST
9 8 7 6 5 4 3 2 1 Ø

          Subtest 3 - Timer Ø/8259 TEST
               8274 TEST
     (Subtest 1) Test 8274 Control options.
               8274 TEST
     (Subtest 2) Data Transfer test.
               8274 TEST
     (Subtest 3) Error Condition test.
          FAST COMMUNICATION PORT TEST
```

APPENDIX C: RECOMMENDED TEST SEQUENCE

This appendix recommends a test sequence for initial testing of the System 6300 computer. This sequence is as follows:

1. The default test sequence (d), which consists of the following tests:

    1. Memory Read, Write, and Refresh Test
    2. Map Translation Test
    3. Page Fault Test
    4. Parity Test
    5. Map RAM Test
    6. User I/O Error Test
    7. Clock Test
    8. RS-232-C (8274) Test (using the plug described in the special hardware requirements subsection in Section 1)
    9. Fast Communication Port Test (using two machines).

2. Subtests 1, 5, 9, 23, and 24 of the disk tests for both the hard and floppy disk drives (tests 6 and 8).

3. Line printer test (1), using a line printer connected to the parallel port.

APPENDIX D: USING THE FASTRAK 30 WORKSTATION'S RS-232-C EMULATION MODE TO RUN THE DIAGNOSTICS

This appendix explains how to use RS-232-C emulation in the Fastrak 30 Workstation's ROM to run the diagnostics and describes some input considerations regarding this minimal implementation.

HOW TO ENTER RS-232-C EMULATION MODE

To enter RS-232-C emulation mode:

1. Connect the Fastrak 30 Workstation to the System 6300 computer with an RS-232 cable.

2. Depress the space bar while at the same time powering on the Fastrak 30 Workstation. The following menu appears:

   B,C,E,F,M,R,S,T:

3. Type E. The Fastrak 30 Workstation enters emulation mode.

4. Boot the diagnostics. After loading, the diagnostics program identifies itself.

SOME INPUT CONSIDERATIONS

Some input considerations for using RS-232-C emulation mode are as follows:

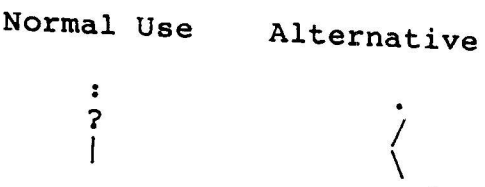
1. Do not use the SHIFT, LOCK, CODE, or BACKSPACE keys.

2. Do not use the function, control, or cursor control keys.

3. The numeric keypad generates only the shifted punctuation characters.

The diagnostic can be run with the restricted character set described above. The diagnostic program is case insensitive to alphabetic characters. Nonalphanumeric input that controls the diagnostic is available either in this character set or has the

following alternative:



INDEX

USER'S COMMENTS

System 6300 Diagnostics Manual

HELP!

Help us help you! Please take the time to complete this form and send it to us. If you do, you may see some of your own contributions in the next manual you obtain from us.

- Does this manual provide the information you need? ☐Yes ☐No
  – What is missing?

- Is the manual accurate? ☐Yes ☐No
  – What is incorrect? (Be specific.)

- Is the manual written clearly? ☐Yes ☐No
  – What is unclear?

- What other comments can you make about this manual?

- What do you like about this manual?

- On a scale of 1 to 10, how do you rate this manual? Low ⊢+-+-+-+-+-+-+-+-+-⊣ High
  1 2 3 4 5 6 7 8 9 10
- Was this manual difficult to obtain? ☐Yes ☐No

Please include your name and address if you would like a reply.

Name _____
Company _____
Address _____
_____

No postage required if mailed within the USA.

System 6300 Diagnostics Manual

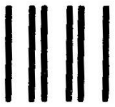
- What is your occupation?

☐ Programmer ☐ Operator ☐ Manager
☐ Systems Analyst ☐ Instructor ☐ Customer Engineer
☐ Engineer ☐ Student ☐ Other _____

- How do you use this manual?

☐ Reference Manual ☐ Introduction to the Subject
☐ In a Class ☐ Introduction to the System
☐ Self Study ☐ Other _____

fold fold



FIRST CLASS
Permit No. 194
Cupertino,
California

BUSINESS REPLY MAIL
No Postage Necessary if Mailed in the United States

Postage will be Paid by . . .

MOTOROLA, INC.
10700 North De Anza Blvd.
Cupertino, Calif. 95014

Attention: Technical Services, MS 42-1C8

fold fold

Cut Along Here

Staple Here

B1847A

System 6300 Diag stics M ual

MOTOROLA

Four-Phase Systems, Inc.

10700 North De Anza Blvd., Cupertino, California 95014 • (408) 255-0900