LISA 1.75 BOOT ROM
Program Reference Specification
Version 1.0

August 16, 1983
George Cossey

Apple Computer Confidential

Table of Contents

1.0 Introduction.
1.1 General 4
1.2 System Diagnostics 4
1.3 Standard User 5
1.4 Advanced User 5
1.5 Hardware Drivers 5

2.0 Reference documents 5

3.0 Symbols and abreviations 6

4.0 Support Enviroment
4.1 Support Software 6
4.1.1 System 6
4.1.2 Utilities 6
4.1.3 Generation and installation procedures 6
4.1.4 Required hardware 6
4.1.5 Processor Type and memory size 7
4.1.6 Optional hardware and peripherals 7

5.0 Performance Criteria
5.1 Program timing parameters 8

6.0 Quality Assurance Test Cases. 8

7.0 Detailed Product Description
7.1 Primary Operation 9
7.2 Diagnostic Testing 9
7.3 Comparison of Lisa 1.0 vs 1.75 diagnostics 10

8.0 System Initialization
8.1 MMU Mapping 11
8.2 Memory 11
8.3 Contrast 11
8.4 Speaker volume 11
8.5 Keyboard/Mouse interface 12
8.6 Configuration check 12

9.0 Boot Devices 12
9.1 Boot ROM protocols 13
9.2 Hard disk booting 13
9.3 Floppy disk booting 13
9.4 I/O expansion slot booting 14
9.5 Internal slot booting. 14

10.0 Monitor modes
10.1 Customer mode 15
10.2 Service mode 16

11.0 External interfaces 18

12.0 Minimal system configuration 18

Appendix A: Memory map 19

Appendix B: ROM routines for external use 22

Appendix C: Error codes 23

1.0 Introduction

1.1 General

The Boot ROM in the Lisa 1.75 system must provide a number of features. It does a basic diagnostic check of the system to assure that the hardware is working properly. It provides a special "service" mode for debug use in case of a failure in any of the diagnostics and that failure prevents the loading of more extensive diagnostics. It initializes the hardware to a known state. The Boot ROM loads a boot program from the user selected boot device and starts it up.

The Boot ROM provides a very simple interface for the standard user and provides a more extensive debug interface for the service tech. Additional manually run diagnostics are provided to allow the user interfaces to be tested directly. These interface tests are necessary if keyboard or Mouse failures prevent the user from performing a normal startup and prevent more extensive diagnostics from being loaded and run.

1.2 System Diagnostics

The system diagnostics are designed to test as much of the system as possible in a very short time. These tests verify the proper operation of the hardware using the standard hardware drivers, whenever possible and if the drivers are in ROM. Some tests are at a much "lower" level than the standard drivers, and these tests access individual portions of the hardware in a manner that they are not normally used by an applications program.

The system diagnostics are ordered in such a way as to use as little untested hardware as possible for the first tests. From then on, each test will test out one small part of untested hardware at a time, using previously tested hardware. This approach makes it easier to locate hardware errors by having only one small part of the hardware added to a test that is not yet checked out.

Each test is allowed to use any part of the hardware, which was tested already, to do it's own test. When a test fails, the only new hardware used in the test is that being tested. There are some cases where more than one new piece of hardware is added, this is done where two or more pieces of hardware are sharing a common path to the next test node. A test node is a hardware point that the testing enviroment has access to. There are usually cases where the testing enviroment has access only to the end result of a hardware path and not to intermediate results. When this is the case, the particular failure mode will help to pinpoint the failure.

1.3 Standard User

The standard user, someone who just wants a check of the system and will not be doing any repair work, will see a minimal amount of test information. This will usually be limited to icons showing which part of the system is being tested. Everyone is a standard user in the normal boot process.

1.4 Advanced User

The more advanced user, field service or the repair depot, has available more detailed information about a test failure. This additional information is on the "service" screen. The standard screen will show what the standard user will see, while by using a special keystroke combination (or possibly some other method) the advanced user may switch to the service screen. This screen displays information that applies to the failing test. This additional information is only displayed on a failure.

Also better error recovery is provided, no endless restart loops are in the program flow. If a fatal error occurs, every effort will be made to display that error and then either go to the monitor routine, for service mode, or enter a special debug loop. The major difference from the Lisa 1.0 ROM is that even on a fatal error, an effort is made to display the error message on the screen before entering a debug loop. Every effort will be made to allow trouble-shooting of a fatal error, the only problem here may be if the video memory is bad or a major failure prevents the CPU from operating properly. More debugging aids are added, these include the new tests added along with a few more memory debugging commands.

1.5 Hardware drivers.

If system drivers are also available in ROM, and they have sufficient error recovery, then they are to be used by the diagnostics. These drivers must permit the type of hardware control necessary for diagnostics. If not, then the Boot ROM will have its own set of drivers.

2.0 Reference documents

LISA 1.75 ERS, July 14, 1983.
LISA 1.75 Hardware Reference manual, (*not written yet*).
LISA 1.75 Boot ROM Test Cases, (*not written yet*).
LISA 1.75 schematics, (*not written yet*).

3.0 Symbols and addreviations

ROM, Read Only Memory.
MB, Mega byte.
Built-in hard disk, Widget.

4.0 Support Environment

4.1 Support Software

4.1.1 System

Designed to work on any software system, standard jump vectors to some of the ROM's internal routines are available for use by anyone. This is a standard boot process which can be used by any system software.

4.1.2 Utilities

NOTE: If there is a Hardware Interface ROM designed then the following paragraph applies.

This program uses the standard drivers in the other system ROM. That ROM must be installed for the Boot ROM to do a boot and if the ROM is not installed then the Boot ROM will place the error on the screen and be available for service mode use.

4.1.3 Generation and installation procedures.

The program can be generated by executing the ROM.GEN.TEXT file under the latest Monitor release. The program must then be programmed into a 32K ROM. The ROM is then installed into the first ROM socket on the CPU board of a LISA 1.75 system. Refer to documentation on the other ROMs for possible interdependant relationships.

4.1.4 Required Hardware

LISA 1.75 or equivalent hardware. All other hardware is optional, this includes:

Memory boards.
Floppy drive.
Built-in hard disk.
Expansion cards.

4.1.5 Processor Type and Memory Size

Motorola 68010 microprocessor. This program will run with only the video memory, used as a scratch pad for program tests. No memory boards are required.

4.1.6 Optional hardware and peripherals

This program uses the Keyboard and the Mouse as primary input devices.
The Sony floppy drive is supported, as is the built-in hard disk.
Any legal memory configuration.
Any expansion cards.

5.0 Performance Criteria (*timing to be determined*)

Program Timing Parameters	*Times*	*Sub-times*
System setup.	x	
CPU-I/O board diagnostics.	x	
Boot ROM checksum test.		x
System #2 ROM checksum test.		x
MMU register check.		x
Parity Generator/Checker test.		x
Video circuit tests.		x
Video Memory diagnostics.	x	
Read/Write pattern tests.		x
Address tests.		x
Pattern tests.		x
Main Memory diagnostics.	x	
Read/Write pattern tests.		x
Address tests.		x
Pattern tests.		x
RS232 Port tests.	x	
Clock/Calendar test.	x	
Timer chip test.	x	
Expansion card diagnostics.	Time depends on card.	
Built-in hard disk diagnostics.	x	
Sony disk diagnostics.	x	
Service mode.	No time limit.	
Keyboard test.	No time limit.	
COPS/Mouse test.	No time limit.	
Manual Contrast test.	No time limit.	
Manual Speaker volume/frequence test.	No time limit.	

6.0 Quality Assurance Test Cases

Refer to the "1.75 Boot Rom Test Cases" documentation.

7.0 Detailed Product Description

7.1 Primary Operation

The ROM will execute its full range of internal diagnostics and boot from the selected device. If a 'cold' start, normal powerup procedure, is in process then all standard diagnostics will be run. If the boot device is the built-in hard disk, then while the disk is coming ready more extensive memory tests will be run. If a 'Warm' start, the reset button on the rear of the system, then limited diagnostics will be done (a subset of the full diagnostics) to bring the system up fast.

7.2 Diagnostic Testing.

The ROM will execute a variety of diagnostic tests, and attempt to narrow failures to the replaceable module. For most failures, the user will be required to get the system repaired. However, certain types of failures will still enable the system to be used, and the ROM will allow the user the option of continuing on a non-fatal error. *Note that the hardware drivers MUST make use of this failure information and prevent the user from using the hardware that has failed the diagnostics.* A mask of failed ports is in the Power-up status section in memory.

Possible non-fatal errors are:
1. Serial port A.
2. Serial port B.
3. IWM port to the floppy drive.
4. Built-in hard disk.
5. Two-port card.
6. Any expansion card.

Note that a failure on any of the above ports automatically makes the failing port an illegal boot device.

For example, if the Serial port A fails the diagnostics then the operating system should inform the user that the port is bad if the user tries to use that port. The user should then be prevented from using that port.

7.3 Comparasion of "old" verses "new" diagnostics.

LISA 1.0 diagnostics.	LISA 1.75 diagnostics.
1. Checksum of Boot ROM.	1. Checksum of Boot ROM and the other three ROMs.
2. Read/Write of MMU.	2. Extensive Read/Write of MMU.
3. Address test of MMU.	3. Extensive Address test of MMU.
4. Memory test, write/read, 1st 2K.	
5. Memory test, write/read, screen area.	4. New Memory tests of video memory.
6. Memory test, write/read, rest of memory.	5. New Memory tests of main memory.
7. Parallel Port VIA, Timer 1 register test. 8. Keyboard Port VIA, Timer 1 register test.	6. Timer chip test.
9. Basic COPS test.	7. Extended COPS test.
10. Video VTIR signal test. 11. Read serial number.	8. Video VTIR signal test. 9. Read serial number.
12. Parity circuitry test, 1 address only.	10. Parity circuitry test, 1 address in every xxxxK segment.
13. RS232, Ch A interrupt vector register test.	11. RS232, Ch A register and internal loopback test.
14. RS232, Ch B internal loopback test.	12. RS232, Ch B register and internal loopback test.
15. Get 871 ROM status.	13. IWM (Sony) diagnostic test.
16. Write/read, 1 location, of shared memory.	
17. Disable command to 871.	
18. Read COPS for date and time.	14. Read COPS for date and time.
19. Status programs on expansion cards.	15. Status programs on expansion cards.
	16. Built-in hard disk diagnostic.

8.0 System Initialization.

The system will be left in the following state after the diagnostics:

8.1 MMU Mapping.

The MMU will be setup so that memory appears to start at address 0 and continue contiguously for the amount of memory contained in the system. All other MMU registers are set for invalid access except for the segments listed below. Context 0, supervisor mode, is set this way. All other contexts are set for invalid access. This setup may be changed if the O.S. group requests.

Segment	Function
94	Video Screen, sound/pwm table.
96	ROM 0
97	ROM 1
98	ROM 2
99	ROM 3
108	Status register and controls.

8.2 Memory.

All of the "main" memory will be left written with 0's. Parity will be left enabled.

Video memory will be left in the following state:

a) Video page - A Blank desktop displaying the wait icon for the boot device. A message saying something like "LISA 1.75, (C) APPLE, 1983" may also be displayed. It will be left in LISA mode.

b) Sound page - Cleared to all zeros.

c) PWM page - Initially cleared to all zeros, if booted from the floppy drive then it will contain the last setting from the driver.

d) The rest - Cleared to all zeros, except for the memory locations listed in appendix A.

8.3 Contrast.

The contrast will be set to the standard value of $xxxx. The shadow location in memory contains the value written.

8.4 Speaker volume.

Speaker volume will be set to minimum, the value of $xxxx. The shadow location in memory contains the value written.

8.5 Keyboard/Mouse Interface.

The port for this interface will be turned on and the initial reset codes will be read and saved. If none are received (e.g. a "warm-start), the ROM will reset the keyboard so that the initial reset codes are again sent. The mouse will be enabled with the timing interval set to 16ms.

8.6 Configuration check.

Each I/O slot will be scanned for the presence of an installed card and, if found, the card ID will be read and saved in memory. If the card ID indicates a status initialization routine is present, the ROM will load and execute the card's status program with the returned status also saved in low memory.

The interface to the I/O cards will remain the same as in LISA 1.0, this allows expansion cards to be used in both if the hardware permits.

9.0 Boot Devices

The built-in hard disk is the new default boot device. This replaces the Profile on the parallel port in Lisa 1.0 as the default device. As a way of telling apart Lisa 1.0 and Lisa 1.75 boot devices, the boot I.D. code of $AAAA for Lisa 1.0 is a standard so Lisa 1.75 boot devices will have an I.D. code of $5555. If a Lisa 1.0 boot device is the boot device, an appropiate message will be displayed with a warning and the user will not be able to continue. The process may be continued from service mode only. The SONY disk is to be added to the possible boot devices along with the serial port running in Apple Bus mode.

Boot device determination:

a) Top priority is the keyboard/mouse input selection of a boot device.

b) Next is the boot device that was saved previously in the COPS.

c) Next is the check of expansion card combinations:

If an APPLENET and a Test card are installed then the APPLENET card will be booted, this is for manufacturing requirements.

Next, if an intelligent test card is installed, IDs $xxxx to $xxxx, then that test card will be booted. This is for LMO stations.

d) Last is a default of the built-in hard disk.

If the boot device fails to boot then the user will be presented with the option of retrying or selecting a new boot device. If the hard disk is the boot device, the Boot ROM will execute extensive memory tests while waiting for the drive to come ready. If the timeout expires, or if the user selects to abort the boot process (note that the keyboard and mouse inputs will be monitored) then the user is returned to the boot device selection menu. Before the hard disk is booted a diagnostic is run on it. If the diagnostic fails then the hard disk will become an invalid boot device and control will return to the boot device selection menu.

9.1 Boot ROM protocols.

If a diagnostic for a specific port fails then that port will be made an invalid boot device. If that failing port was the selected boot device then the alternate boot selection menu will be displayed.

After executing the power-up tests and saving the configuration data, the ROM will boot a startup program from the selected device, which will be a maximum of 512 bytes loaded into video memory. The default boot device is the built-in hard disk. The boot blocks loaded in from the boot device will now have the added check of having to pass a checksum test. The checksum method is described in one of the appendixes, this is the same method as used by Lisa 1.0 Boot ROM. The checksum requirement may be overridden in the service mode.

9.2 Hard disk booting.

Block 0 is assumed to be the boot area. With 532 bytes read, 20 for the header, and 512 for the startup program. The header field must be $5555, note that the LISA 1.0 system expects a header of $AAAA, the fields are different to keep a LISA 1.0 disk from trying to boot on the LISA 1.75. There are enough hardware differences that booting a LISA 1.0 program would 'crash'. If the header verifies OK and the checksum test of the boot block passes, and no other errors occur, the ROM will jump to the first program location, also set up to be at address xxxK. There will be a timeout used for the wait on the disk ready. This allows a way to get out if the disk never comes ready. The I.D. check can be overridden by the service mode user.

9.3 Floppy Disk booting.

For floppy (Sony disk) booting, track 0, side 0, sector 0 (with header) for the startup program for a total of 524 bytes. This will be loaded with header starting at location xxxK minus header length, the actual program will start at location xxxK. The startup program header will be checked for a file ID of $5555 to ensure that the data read was a boot program. If the initial load is successful, the ROM will then jump to the initial location of the program code at address xxxK. Any failures will then be saved in low video memory. The I.D. check can be overridden by the service mode user. A checksum test is also done on the block loaded in. If the checksum of the loaded data is correct, the ROM will jump to the initial program location.

9.4 I/O Slot boot.

This will be done by reading a "boot ROM" on the selected I/O card. If the checksum of the loaded data is correct and initial status checks are OK, the ROM will jump to the initial program location.

9.5 Internal slot booting.

This is to allow booting from an alternate processor card installed in one of the two "new" slots.
Method is to be determined.

10.0 Monitor modes.

If errors are detected during power-up testing, appropriate icons and tones will be used to notify the operator. After the icons are displayed, and if the errors prevent automatic booting, a menu of options will appear at the top of the LISA screen requesting user input. At this point the ROM will be in a special "monitor" program that contains two levels. One is called the "Customer mode" and the other is called the "Service mode". The standard Lisa user interface will be used to the extent that a menu bar and dialog box will be used at each level, the menu selections are available via mouse selection or keyboard input.

10.1 Customer mode.

In this preliminary mode, the error icon, if any, will be displayed on the screen. Options available are:

Restart, Continue, Startup from..., Turn Off

Restart.
This option provides a means of resetting the system from the keyboard or mouse to retry the power-up tests. It will cause a "double bus fault" to occur, which in turn will reset the system and cause the startup diagnostics and procedures to be reexecuted.

Continue.
This option gives the capability of continuing after a non-fatal error, So the system can be used in a "degraded" mode or extended diagnostics running under LisaTest can be run. For example, if one of the RS232 ports fails the diagnostics, and if that port is not used by the user, then the system is still usable as far as the user is concerned.

Startup from...
In the case of a boot failure, this option provides the means of easily retrying the boot from the same or another device.

Turn Off.
If this option is selected then the system will be turned off.

10.2 Service mode.

This mode may be entered by specifing APPLE S during the boot process.

This mode is provided primarily for engineering, manufacturing, and field service use. It provides basic hardware analysis and manipulation instructions.
The options available are:

Lisa debug
Call program
Execute a test
Loop on a test
Adjust L video
Adjust S video
Grey pattern
Power cycle
Help
Talk, Serial B
Exit

The following special functions are provided in the 'Lisa debug' mode:

SB addr xx	Write a byte value, odd or even address allowed.
SW addr xxxx	Write a word value, default for S addr xxxx, must be even address.
SL addr xxxxxxxx	Write a long word value, must be even address.
SCB addr xx	Write a byte continuously.
SCW addr xxxx	Write a word continuously, must be even address.
SCL addr xxxxxxxx	Write a long word continuously, must be even address.
DW addr	Read a word value, default for D addr, must be even address.
DL addr	Read a long word value, must be even address.
DS addr	Read a section of memory, must be even address.
DCB addr	Read a byte continuously, with display when the value read changes.
DCW addr	Read a word continuously, with display when the value read changes, must be even address.
RC addr	Read a word continuously, no display, must be even address.
X	Exit continuous loop, interrupt driven.
TS addr	Test write a location, writes pattern shown to addr. $0000, $FFFF, $AAAA, $5555, $AAAA, $5555
TSR addr	Test write a location, writes pattern shown to address then does a read of that location. $0000, $FFFF, $AAAA, $5555, $AAAA, $5555

TLOC addr xxxx xxxx	Test a location, writes two (2) users patterns to the address and then does a read of that location.

COPS xx	Send COPS a byte command or data.
MONCOPS	Monitor all data coming from the COPS, displayed in a separate box.
XMONCOPS	Exits monitor COPS mode.
HELP	Displays special addresses of interest and service commands available.
TMODE x y	Sets timer x (0 to 2) to mode y (0 to 5).
TIMER0 xxxx	Writes timer #0 to a specific word value.
TIMER1 xxxx	Writes timer #1 to a specific word value.
TIMER2 xxxx	Writes timer #2 to a specific word value.
RB	Re-boot.
QUIT	Returns to main service menu.

All exceptions are trapped and all values displayed.

Display MMU contents.

Change MMU contents.

Extended memory tests. These tests may run from less than one minute to as much as several hours long.

NOTE: *Depending on size constraints, some of the above may not be available or even more options may be available.*

The 'Call a program' option allows you to call any location in memory as a separate program.

The 'Execute a test' option allows access to the standard diagnostics and a list of extended diagnostics. This mode will probably be used the most to run the extended memory tests.

The 'Loop on a test' option is similar to the 'Execute a test' option except that the test will be looped on continuously until a failure.

There are three (3) video adjustment modes. The first two show a crosshatch pattern in the selected video mode (square dots or Lisa mode). The L is for Lisa 1.0 mode and the S is for square dot mode. The third is a grey pattern.

The 'Help' option lists the Lisa debug mode commands, some addresses in memory of areas of interest, and short descriptions of other options.

The 'Talk, Serial B' option is for external diagnostic use and executes a special TALK driver thru the B serial port.

11.0 External Interfaces.

A main jump table will support outside programs with the Boot ROM's diagnostics and other routines. The interfaces to these diagnostics will be consistant with LMO and LisaTest needs. The error codes established by LISA 1.0 remain the same, with the exceptions described in the appendix, and additional error codes have been added.

12.0 Minimal system configuration.

The Boot ROM is able to run on a system without any memory boards, only the video memory will be available. This means that all data, exception vectors, stack, and the boot block of the boot device will be placed in the video memory for use. This mode of operation is necessary for the system operating as a CPU board and memory board checkout station.

Appendicies

Appendix A.

Memory Map.

The ROM uses part of low video memory from address $000 to $400 (relative to location xxxxxxxx) as a save area for error and configuration data. Differences with Lisa 1.0 are specified. The layout and contents are as follows:

Address	Contents
$000-$003	Power-up status #1, $00000000 = OK, any bit = '1' indicates an error.

Bit meanings are:

0 - ROM 0 checksum.
1 - ROM 1 checksum.
2 - Video memory.
3 - Video memory parity.
4 - Video circuitry.
5 - Serial number.
6 - Timer #0.
7 - Timer #1.
8 - Timer #2.
9 - COPS test.
10 - Main memory.
11 - Main/memory parity.
12 - MMU read/write.
13 - MMU address.
14 - MMU functional.
15 -
16 -
17 -
18 -
19 - Keyboard COPS.
20 - CPU board COPS.
21 - Line 1010 or 1111.
22 - Illegal instruction.
23 - Misc exception (e.g divide by zero).
24 - Address error.
25 - Bus error.
26 - Unexpected level 2 (slot 4) interrupt.
27 - Unexpected level 3 (slot 3) interrupt.
28 - Unexpected level 4 (slot 2) interrupt.
29 - Unexpected level 5 (slot 1) interrupt.
30 - Unexpected level 6 (SCC, timer 1) interrupt.
31 - Unexpected level 7 (NMI, parity) interrupt.

$004-$007 Power-up status #2, $00000000 = OK, any bit = '1' indicates an error.

Bit meanings are:

0 - RS232 port A.
1 - RS232 port B.
2 - IWM, floppy disk.
3 - Built-in hard disk.
4 - Slot 4 status.
5 - Slot 3 status.
6 - Slot 2 status.
7 - Slot 1 status.
8 -
9 -
10 -
11 -
12 -
13 -
14 -
15 -
16 -
17 -
18 -
19 -
20 -
21 -
22 -
23 -
24 -
25 -
26 -
27 - Fatal error occured.
28 - Non-fatal error occured.
29 - No memory installed.
30 - Mouse disconnected.
31 - Keyboard disconnected.

$008-00B Maximum physical memory address+1.
$00C-00F Minimum physical memory address.
$010-013 Total amount of memory.
$014-01F Reserved

```
$020-021  I/O slot 1 device ID.
$022-023    "    "  2     "    " .
$024-025    "    "  3     "    " .
$026-027    "    "  4     "    " .
$028-029  Reserved.
$02A        "    "  1     "    status.
$02B        "    "  2     "      "   .
$02C        "    "  3     "      "   .
$02D        "    "  4     "      "   .
$02E-02F  Reserved.

$030    Boot device ID, same as code from COPS.
            Bit meanings are:
               0 to 4 - Boot device or power cycle.
                 Value of $00 is Built-in hard disk.
                         $01 is built-in floppy disk.
                         $02 is Expansion slot 1, port 1.
                         $03 is Expansion slot 1, port 2.
                         $04 is Expansion slot 2, port 1.
                         $05 is Expansion slot 2, port 2.
                         $06 is Expansion slot 3, port 1.
                         $07 is Expansion slot 3, port 2.
                         $08 is Slot 4.
                         $09 is Reserved.
                         $0A is Serial port B using AppleBus.
                         $0B is Serial port B using TALK.
                         $0C is abort to ROM monitor.
                         $0D is Reserved.
                         $0E is CPU board burn-in.
                         $0F is Power cycling.
                  5    - Reserved.
                  6    - Extended memory tests.
                  7    - Odd Parity for code in COPS.

$031    Keyboard I.D.
```

The ROM version number is located at $xxxxxxxx, it is one word long and in the same format as the LISA 1.0 boot rom version number.

Appendix B: ROM Routines.

The following pages give details on the interfaces for various user accessable ROM routines. Unless specified otherwise, all registers are preserved as they would be from PASCAL, this means that D0-D3 and A0-A3 may be destroyed.

NAME:
FUNCTION:
INPUTS REQUIRED:
OUTPUTS:
CALLING SEQUENCE: JSR to $xxxxxxxx

Execute diagnostics.
$xxxxxxxx (To be defined)
$xxxxxxxx
$xxxxxxxx

Graphics routines.
$xxxxxxxx (To be defined)
$xxxxxxxx
$xxxxxxxx

General utilites.
$xxxxxxxx (To be defined)
$xxxxxxxx
$xxxxxxxx
$xxxxxxxx

LISA 1.0 interfaces to be removed if a Hardware interface ROM is available:

1. Hard-disk read a block routine.
2. 871 disk read a sector routine.
3. Hard-disk handshake routine.
4. Hard-disk get response routine.
5. Send COPS command.
6. Read clock/calendar setting.
7. Read mouse position.
8. Set contrast value.
9. Beep speaker.

Appendix C.

Error code differences, as the diagnostics are written this list will be added to.

Error messages.
CPU - I/O board failure.
Memory board failure.
Expansion Card failure.
Keyboard failure.
Startup failure, on boot device.
Keyboard disconnected.
Mouse disconnected.

CPU Logic Failure. Indicating a bad CPU card.

Error Code	Meaning
41	MMU failure.
42	Video failure, VTR, CS, or VID signal.
43	Parity circuit failure.
44	Unexpected NMI.
45	Bus error.
46	Address error.
47	Illegal instruction error.
48	Line 1111 or 1010 trap.
49	Other unexpected exception.
70	Main memory failure.
72	Video memory failure.

Floppy disk failure.

Error Code	Meaning
xx	Standard floppy failure code.
38	Bad boot file ID code, expect $5555 for Lisa 1.75.
39	Timeout error, drive did not respond.

I/O Logic Failure. Indicating a bad I/O portion of the CPU-I/O board.

Error Code	Meaning
52	COPS #1 (I/O board) failure (bad or no reset code).
53	COPS #2 (keyboard) failure.
54	Clock failure (incorrect operation).
55	RS232 port A failure.
56	RS232 port B failure.
57	Disk controller (IWM) failure.

Memory Failure. If two boards installed, ROM will attempt to pinpoint which board.

WIDGET disk failure.

Error Code	Meaning
80	Hard disk does not appear to be attached.
81	Hard disk is not ready.
82	Unexpected response.
83	Nonzero status bytes.
84	Bad boot file ID.
85	Timeout error, drive did not respond.

I/O Slot Failure. Bad status returned from installed card.

Error Code	Meaning
90	No card installed in expansion slot requested.
91	Invalid card ID code.
92	Bad checksum on cards boot rom.
93	Bad status returned from card.

Keyboard Failure. Keyboard internal diagnostics error.