

# Sun™ Ultra™ 5/10 Product Note

---



THE NETWORK IS THE COMPUTER™

**Sun Microsystems Computer Company**

A Sun Microsystems, Inc. Business  
901 San Antonio Road  
Palo Alto, CA 94303-4900 USA  
650 960-1300 fax 650 969-9131

Part No.: 805-3647-10  
Revision A, November 1997

Copyright 1997 Sun Microsystems, Inc., 901 San Antonio Road, Palo Alto, CA 94303-4900 USA. All rights reserved.

This product or document is protected by copyright and distributed under licenses restricting its use, copying, distribution, and decompilation. No part of this product or document may be reproduced in any form by any means without prior written authorization of Sun and its licensors, if any. Third-party software, including font technology, is copyrighted and licensed from Sun suppliers.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, AnswerBook, SunDocs, Ultra, ShowMe How, OpenBoot, and Solaris are trademarks, registered trademarks, or service marks of Sun Microsystems, Inc. in the U.S. and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK and Sun™ Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

**RESTRICTED RIGHTS:** Use, duplication, or disclosure by the U.S. Government is subject to restrictions of FAR 52.227-14(g)(2)(6/87) and FAR 52.227-19(6/87), or DFAR 252.227-7015(b)(6/95) and DFAR 227.7202-3(a).

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

Copyright 1997 Sun Microsystems, Inc., 901 San Antonio Road, Palo Alto, CA 94303-4900 Etats-Unis. Tous droits réservés.

Ce produit ou document est protégé par un copyright et distribué avec des licences qui en restreignent l'utilisation, la copie, la distribution, et la décompilation. Aucune partie de ce produit ou document ne peut être reproduite sous aucune forme, par quelque moyen que ce soit, sans l'autorisation préalable et écrite de Sun et de ses bailleurs de licence, s'il y en a. Le logiciel détenu par des tiers, et qui comprend la technologie relative aux polices de caractères, est protégé par un copyright et licencié par des fournisseurs de Sun.

Des parties de ce produit pourront être dérivées des systèmes Berkeley BSD licenciés par l'Université de Californie. UNIX est une marque déposée aux Etats-Unis et dans d'autres pays et licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, AnswerBook, SunDocs, Ultra, ShowMe How, OpenBoot, et Solaris sont des marques de fabrique ou des marques déposées, ou marques de service, de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays. Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

L'interface d'utilisation graphique OPEN LOOK et Sun™ a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciés de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui en outre se conforment aux licences écrites de Sun.

CETTE PUBLICATION EST FOURNIE "EN L'ETAT" ET AUCUNE GARANTIE, EXPRESSE OU IMPLICITE, N'EST ACCORDEE, Y COMPRIS DES GARANTIES CONCERNANT LA VALEUR MARCHANDE, L'APTITUDE DE LA PUBLICATION A REPENDRE A UNE UTILISATION PARTICULIERE, OU LE FAIT QU'ELLE NE SOIT PAS CONTREFAISANTE DE PRODUIT DE TIERS. CE DENI DE GARANTIE NE S'APPLIQUERAIT PAS, DANS LA MESURE OU IL SERAIT TENU JURIDIQUEMENT NUL ET NON AVENU.



# Sun Ultra 5/10 Product Note

---

This product note contains changes to Sun Ultra 5/10 documentation that became known after release of the documentation. It contains the following sections:

- ShowMe How Audio Button Problem — page 4
- POST Screen Changes — page 5
- OBDiag Screen Changes — page 13
- German Acoustic Compliance Statement — page 27

---

# ShowMe How Audio Button Problem

This section of the product note describes a problem you may experience when using the audio buttons in the ShowMe™ How™ multimedia documentation on some Ultra™ 5/10 systems. The problem is caused by alias settings in the system's `.cshrc` file.

## Problem Description

The audio buttons in the ShowMe How multimedia documentation might not work correctly if the shell from which you invoked ShowMe How has an alias that contains file name range-substitution functions.

For example, adding the following line

```
alias lh ls -a [a-z] [a-z]*
```

to your `.cshrc` file could cause the audio buttons in ShowMe How not to work correctly.

## Problem Resolution

To fix this problem:

**1. Check which alias statements are included in your shell's `.cshrc` file:**

At a command line prompt within the shell from which you are trying to run ShowMe How, type `alias` and press Return.

All alias statements in the `.cshrc` file are listed.

**2. Use an editor to remove any alias statements that contain file name range substitutions from the `.cshrc` file (and from any additional files it sources).**

**3. Save your changes and close the `.cshrc` file.**

**4. Type `source .cshrc` at the command prompt.**

**5. Restart ShowMe How.**

---

## POST Screen Changes

This section of the product note contains changes to the power-on self-test (POST) screens that were made after release of the *Sun Ultra 5/10 Service Manual (805-0423-10)*.

This section of the product note contains updated Sections 3.4.1 and 3.4.2 from the *Sun Ultra 5/10 Service Manual*.

### 3.4.1 diag-level Variable Set to max

When the `diag-level` variable is set to `max`, POST enables an extended set of diagnostic-level tests. This mode requires approximately 1 minute to complete (with 64 Mbytes of DIMM installed). CODE EXAMPLE 3-1 identifies a typical serial port A POST output with the `diag-level` variable set to `max`.

---

**Note** – There will be no video output while POST is initialized.

---

#### CODE EXAMPLE 3-1 `diag-level` Variable Set to max

```
ok Hardware Power ON

@(#) Sun Ultra 5/10 UPA/PCI x.xx Version x created xxxx/xx/xx xx:xx
Probing keyboard Done
%o0 = 0000.0000.0000.4001

Executing Power On SelfTest

@(#) Sun Ultra 5/10 (Darwin) POST x.x.x (Build No. xxx) xx/xx/xx:
xx:xx

CPU: UltraSPARC-LC (MHz: 270 Ecache Size: 256KB)

Init POST BSS
  Init System BSS
NVRAM
  NVRAM Battery Detect Test
  NVRAM Scratch Addr Test
  NVRAM Scratch Data Test
DMMU TLB Tags
  DMMU TLB Tag Access Test
```

**CODE EXAMPLE 3-1** diag-level Variable Set to max (Continued)

```
DMMU TLB RAM
  DMMU TLB RAM Access Test
Probe Ecache
  Probe Ecache
Ecache Tests
  Ecache RAM Addr Test
  Ecache Tag Addr Test
  Ecache RAM Test
  Ecache Tag Test
All CPU Basic Tests
  V9 Instruction Test
  CPU Tick and Tick Compare Reg Test
  CPU Soft Trap Test
  CPU Softint Reg and Int Test
All Basic MMU Tests
  DMMU Primary Context Reg Test
  DMMU Secondary Context Reg Test
  DMMU TSB Reg Test
  DMMU Tag Access Reg Test
  DMMU VA Watchpoint Reg Test
  DMMU PA Watchpoint Reg Test
  IMMU TSB Reg Test
  IMMU Tag Access Reg Test
All Basic Cache Tests
  Dcache RAM Test
  Dcache Tag Test
  Icache RAM Test
  Icache Tag Test
  Icache Next Test
  Icache Predecode Test
Sabre MCU Control & Status Regs Init and Tests
  Init Sabre MCU Control & Status Regs
  Initializing SC registers in SabreIO
Memory Probe and Init
  Probe Memory
  INFO: All the memory banks in 10 bit column mode
  INFO:      64MB Bank 0
  bank 2:  OMB
frequency = 270, refvalue = 131, no_of_banks = 1
INFO: MC0 = 0x00000000.80000183, MC1 = 0x00000000.0626168a
  Malloc Post Memory
  Memory Addr with Ecache
  Load Post In Memory
  Run POST from MEM
  .....
loaded POST in memory
  Map PROM/STACK/NVRAM in DMMU
```

**CODE EXAMPLE 3-1** diag-level Variable Set to max (Continued)

```
Update Master Stack/Frame Pointers
All FPU Basic Tests
  FPU Regs Test
  FPU Move Regs Test
  FPU State Reg Test
  FPU Functional Test
  FPU Trap Test
UPA Data Bus Line Test
Memory Tests
  Init Memory
  INFO:      64MB at bank 0 stack 0 (2 dimms per bank)
  .....
  INFO:      0MB at bank 0 stack 1
  INFO:      0MB at bank 2 stack 0
  INFO:      0MB at bank 2 stack 1
  Memory Addr with Ecache Test
  INFO:      64MB at bank 0 stack 0 (2 dimms per bank)
  INFO:      0MB at bank 0 stack 1
  INFO:      0MB at bank 2 stack 0
  INFO:      0MB at bank 2 stack 1
  ECC Memory Addr Test
  INFO:      64MB at bank 0 stack 0 (2 dimms per bank)
  INFO:      0MB at bank 0 stack 1
  INFO:      0MB at bank 2 stack 0
  INFO:      0MB at bank 2 stack 1
  Block Memory Addr Test
  INFO:      64MB at bank 0 stack 0 (2 dimms per bank)
  INFO:      0MB at bank 0 stack 1
  INFO:      0MB at bank 2 stack 0
  INFO:      0MB at bank 2 stack 1
  Block Memory Test
  INFO:      64MB at bank 0 stack 0 (2 dimms per bank)
Write 0x33333333.33333333
.....
Read .....
Write 0x55555555.55555555
.....
Read .....
Write 0xcccccccc.cccccccc
.....
Read .....
Write 0xaaaaaaaa.aaaaaaaa
.....
Read .....

INFO:      0MB at bank 0 stack 1
```

**CODE EXAMPLE 3-1** diag-level Variable Set to max (Continued)

```
INFO:      OMB at bank 2 stack 0
INFO:      OMB at bank 2 stack 1
ECC Blk Memory Test
INFO:      64MB at bank 0 stack 0 (2 dimms per bank)
Write 0xa5a5a5a5.a5a5a5a5
.....
Read .....
Write 0x96969696.96969696
.....
Read .....
Write 0xbbbbbbbb.bbbbbbbb
.....
Read .....
Write 0xdddddddd.dddddddd
.....
Read .....

INFO:      OMB at bank 0 stack 1
INFO:      OMB at bank 2 stack 0
INFO:      OMB at bank 2 stack 1
All Basic Sabre MMU Tests
  Init Sabre
  PIO Decoder and BCT Test
  PCI Byte Enable Test
  Interrupt Map (short) Reg Test
  Interrupt Set/Clr Reg Test
  Sabre IOMMU Regs Test
  Sabre IOMMU RAM Address Test
  Sabre IOMMU CAM Address Test
  IOMMU TLB Compare Test
  IOMMU TLB Flush Test
  PBMA PCI Config Space Regs Test
  PBMA Control/Status Reg Test
  PBMA Diag Reg Test
  Sabre IO Regs Test
All Advanced CPU Tests
  DMMU Hit/Miss Test
  IMMU Hit/Miss Test
  DMMU Little Endian Test
  IU ASI Access Test
  FPU ASI Access Test
  Ecache Thrash Test
All CPU Error Reporting Tests
  CPU Data Access Trap Test
  CPU Addr Align Trap Test
  DMMU Access Priv Page Test
```



**CODE EXAMPLE 3-1** diag-level Variable Set to max (Continued)

```
DMMU Write Protected Page Test
All Advanced Sabre IOMMU Tests
  Init Sabre
  Consist DMA Rd, IOMMU miss Ebus Test
  Consist DMA Rd, IOMMU hit Ebus Test
  Consist DMA Wr, IOMMU miss Ebus Test
  Consist DMA Wr, IOMMU hit Ebus Test
  Pass-Thru DMA Rd, Ebus device Test
  Pass-Thru DMA Wr, Ebus device Test
  Consist DMA Rd, IOMMU LRU Lock Ebus Test
  Consist DMA Wr, IOMMU LRU Locked Ebus Test
All Basic Cheerio Tests
  Cheerio Ebus PCI Config Space Test
  Cheerio Ethernet PCI Config Space Test
  Cheerio Init
All Sabre IOMMU Error Reporting Tests
  Init Sabre
  PIO Read, Master Abort Test
  PIO Read, Target Abort Test

Status of this POST run:PASS
manufacturing mode=OFF
Time Stamp [hour:min:sec] xx:xx:xx [month/date year] xx/xx/xxxx

Power On Selftest Completed
```

## 3.4.2 diag-level Variable Set to min

When the `diag-level` variable is set to `min`, POST enables an abbreviated set of diagnostic-level tests. This mode requires approximately 30 seconds to complete (with 64 Mbytes of DIMM installed). CODE EXAMPLE 3-2 identifies a serial port A POST output with the `diag-level` NVRAM variable set to `min`.

---

**Note** – Video output is disabled while POST is initialized.

---

### CODE EXAMPLE 3-2 `diag-level` Variable Set to min

```
Executing Power On SelfTest

@(#) Sun Ultra 5/10 (Darwin) POST x.x.x (Build No. xxx) xx/xx/xx:
xx:xx

CPU: UltraSPARC-LC (MHz: 270 Ecache Size: 256KB)

Init POST BSS
  Init System BSS
NVRAM
  NVRAM Battery Detect Test
  NVRAM Scratch Addr Test
  NVRAM Scratch Data Test
DMMU TLB Tags
  DMMU TLB Tag Access Test
DMMU TLB RAM
  DMMU TLB RAM Access Test
Probe Ecache
  Probe Ecache
Ecache Tests
  Ecache RAM Addr Test
  Ecache Tag Addr Test
All CPU Basic Tests
  V9 Instruction Test
  CPU Soft Trap Test
  CPU Softint Reg and Int Test
All Basic MMU Tests
  DMMU Primary Context Reg Test
  DMMU Secondary Context Reg Test
  DMMU TSB Reg Test
  DMMU Tag Access Reg Test
  IMMU TSB Reg Test
  IMMU Tag Access Reg Test
All Basic Cache Tests
```

**CODE EXAMPLE 3-2** diag-level Variable Set to min (Continued)

```
Dcache RAM Test
Icache RAM Test
Sabre MCU Control & Status Regs Init and Tests
  Init Sabre MCU Control & Status Regs
  Initializing SC registers in SabreIO
Memory Probe and Init
  Probe Memory
  INFO: All the memory banks in 10 bit column mode
  INFO:      64MB Bank 0
  bank 2:  OMB
frequency = 270, refvalue = 131, no_of_banks = 1
INFO: MCO = 0x00000000.80000183, MC1 = 0x00000000.0626168a
  Malloc Post Memory
  Memory Addr with Ecache
  Load Post In Memory
  Run POST from MEM
  .....
loaded POST in memory
  Map PROM/STACK/NVRAM in DMMU
  Update Master Stack/Frame Pointers
All FPU Basic Tests
  FPU Regs Test
  FPU Move Regs Test
UPA Data Bus Line Test
Memory Tests
  Init Memory
  INFO:      64MB at bank 0 stack 0 (2 dimms per bank)
  .....
  INFO:      OMB at bank 0 stack 1
  INFO:      OMB at bank 2 stack 0
  INFO:      OMB at bank 2 stack 1
  ECC Memory Addr Test
  INFO:      64MB at bank 0 stack 0 (2 dimms per bank)
  INFO:      OMB at bank 0 stack 1
  INFO:      OMB at bank 2 stack 0
  INFO:      OMB at bank 2 stack 1
All Basic Sabre MMU Tests
  Init Sabre
  Interrupt Map (short) Reg Test
  Interrupt Set/Clr Reg Test
  Sabre IOMMU Regs Test
  Sabre IOMMU RAM Address Test
  Sabre IOMMU CAM Address Test
  PBMA PCI Config Space Regs Test
  PBMA Control/Status Reg Test
  PBMA Diag Reg Test
  Sabre IO Regs Test
```

**CODE EXAMPLE 3-2** diag-level Variable Set to min (*Continued*)

```
All Advanced CPU Tests
  IU ASI Access Test
  FPU ASI Access Test
All CPU Error Reporting Tests
  CPU Data Access Trap Test
  CPU Addr Align Trap Test
  DMMU Access Priv Page Test
  DMMU Write Protected Page Test
All Advanced Sabre IOMMU Tests
  Init Sabre
  Consist DMA Rd, IOMMU miss Ebus Test
All Basic Cheerio Tests
  Cheerio Ebus PCI Config Space Test
  Cheerio Ethernet PCI Config Space Test
  Cheerio Init
All Sabre IOMMU Error Reporting Tests
  Init Sabre
  PIO Read, Master Abort Test
  PIO Read, Target Abort Test

Status of this POST run:PASS
manufacturing mode=OFF
Time Stamp [hour:min:sec] xx:xx:xx [month/date year] xx/xx xxxx

Power On Selftest Completed
```

---

## OBDiag Screen Changes

This section of the product note contains changes to the OpenBoot™ diagnostic (OBDiag) screens that were made after release of the *Sun Ultra 5/10 Service Manual* (805-0423-10).

This section of the product note contains updated Section 4.7 (and subsections) from the *Sun Ultra 5/10 Service Manual*.

---

### 4.7 OpenBoot Diagnostics

The OpenBoot diagnostic (OBDiag) is a menu-driven diagnostic tool that verifies:

- Internal I/O system
- Ethernet
- EIDE
- Keyboard
- Mouse
- Serial port
- Parallel port
- Audio
- Diskette
- NVRAM
- PCIO ASIC

OBDiag performs root-cause failure analysis on the referenced devices by testing internal registers, confirming subsystem integrity, and verifying device functionality.

To initialize the OBDiag menu:

1. At the `ok` prompt, type `obdiag`.
2. Verify that the OBDiag screen is displayed (CODE EXAMPLE 4-6).

#### CODE EXAMPLE 4-6 OBDiag Screen

```
ok obdiag
stdin: fffeld10
stdout: fffeld18
loading code into: /pci@1f,0/pci@1,1/ebus@1
loading code into: /pci@1f,0/pci@1,1/ebus@1/eprom@14,0
loading code into: /pci@1f,0/pci@1,1/ebus@1/su@14,3062f8
loading code into: /pci@1f,0/pci@1,1/ebus@1/se@14,400000
```

**CODE EXAMPLE 4-6** OBDiag Screen (Continued)

```
loading code into: /pci@1f,0/pci@1,1/network@1,1
loading code into: /pci@1f,0/pci@1,1/ebus@1/fdthree@14,3023f0
loading code into: /pci@1f,0/pci@1,1/ide@3
loading code into: /pci@1f,0/pci@1,1/ide@3/disk
loading code into: /pci@1f,0/pci@1,1/ide@3/cdrom
Debugging enabled
ok
```

3. At the `ok` prompt, type `obtest`.
4. Verify that the OBDiag menu is displayed (CODE EXAMPLE 4-7).
5. At the OBDiag menu prompt, type `16` to enable toggle script-debug messages.

**CODE EXAMPLE 4-7** OBDiag Menu

```
ok obtest

          OBDiag Menu

0 ..... PCI/Cheerio
1 ..... EBUS DMA/TCR Registers
2 ..... Ethernet
3 ..... Keyboard
4 ..... Mouse
5 ..... Floppy
6 ..... Parallel Port
7 ..... Serial Port A
8 ..... Serial Port B
9 ..... NVRAM
10 ..... Audio
11 ..... EIDE
12 ..... Video
13 ..... All Above
14 ..... Quit
15 ..... Display this Menu
16 ..... Toggle script-debug
17 ..... Enable External Loopback Tests
18 ..... Disable External Loopback Tests

Enter (0-13 tests, 14 -Quit, 15 -Menu) ==>
```

## 4.7.1 PCI/PCIO

The PCI/PCIO diagnostic performs the following:

1. `vendor_id_test` - Verifies the PCIO ASIC vendor ID is 108e.
2. `device_id_test` - Verifies the PCIO ASIC device ID is 1000.
3. `mixmode_read` - Verifies the PCI configuration space is accessible as half-word bytes by reading the EBus2 vendor ID address.
4. `e2_class_test` - Verifies the address class code. Address class codes include bridge device (0 x B, 0 x 6), other bridge device (0 x A and 0 x 80), and programmable interface (0 x 9 and 0 x 0).
5. `status_reg_walk1` - Performs walk one test on status register with mask 0 x 280 (PCIO ASIC is accepting fast back-to-back transactions, DEVSEL timing is 0 x 1).
6. `line_size_walk1` - Performs tests 1 through 5.
7. `latency_walk1` - Performs walk one test on latency timer.
8. `line_walk1` - Performs walk one test on interrupt line.
9. `pin_test` - Verifies interrupt pin is logic-level high (1) after reset.

CODE EXAMPLE 4-8 identifies the PCI/PCIO output message.

### CODE EXAMPLE 4-8 PCI/PCIO Output Message

```
Enter (0-13 tests, 14 -Quit, 15 -Menu) ==> 0

TEST='all_pci/cheerio_test'
SUBTEST='vendor_id_test'
SUBTEST='device_id_test'
SUBTEST='mixmode_read'
SUBTEST='e2_class_test'
SUBTEST='status_reg_walk1'
SUBTEST='line_size_walk1'
SUBTEST='latency_walk1'
SUBTEST='line_walk1'
SUBTEST='pin_test'
Enter (0-13 tests, 14 -Quit, 15 -Menu) ==>
```

## 4.7.2 EBus DMA/TCR Registers

The EBUS DMA/TCR registers diagnostic performs the following:

1. `dma_reg_test` - Performs a walking ones bit test for control status register, address register, and byte count register of each channel. Verifies that the control status register is set properly.
2. `dma_func-test` - Validates the DMA capabilities and FIFOs. Test is executed in a DMA diagnostic loopback mode. Initializes the data of transmitting memory with its address, performs a DMA read and write, and verifies that the data received is correct. Repeats for four channels.

CODE EXAMPLE 4-9 identifies the EBus DMA/TCR registers output message.

### CODE EXAMPLE 4-9 EBus DMA/TCR Registers Output Message

```
Enter (0-13 tests, 14 -Quit, 15 -Menu) ==> 1

TEST='all_dma/ebus_test'
SUBTEST='dma_reg_test'
SUBTEST='dma_func_test'
Enter (0-13 tests, 14 -Quit, 15 -Menu) ==>
```

## 4.7.3 Ethernet

The Ethernet diagnostic performs the following:

1. `my_channel_reset` - Resets the Ethernet channel.
2. `hme_reg_test` - Performs walk1 on the following registers set: global register 1, global register 2, bmac xif register, bmac tx register, and the mif register.
3. `MAC_internal_loopback_test` - Performs Ethernet channel engine internal loopback.
4. `10_mb_xcvr_loopback_test` - Enables the 10Base-T data present at the transmit MII data inputs to be routed back to the receive MII data outputs.
5. `100_mb_phy_loopback_test` - Enables MII transmit data to be routed to the MII receive data path.
6. `100_mb_twister_loopback_test` - Forces the twisted-pair transceiver into loopback mode.

CODE EXAMPLE 4-10 identifies the Ethernet output message.



**CODE EXAMPLE 4-10 Ethernet Output Message**

```
Enter (0-13 tests, 14 -Quit, 15 -Menu) ===> 2

TEST='ethernet_test'
Using Onboard Transceiver - Link Up.
SUBTEST='my_channel_reset'
SUBTEST='hme_reg_test'
SUBTEST='global_reg1_test'
SUBTEST='global_reg2_test'
SUBTEST='bmac_xif_reg_test'
SUBTEST='bmac_tx_reg_test'
SUBTEST='mif_reg_test'
SUBTEST='mac_internal_loopback_test'
SUBTEST='10mb_xcvr_loopback_test'
SUBTEST='100mb_phy_loopback_test'
SUBTEST='100mb_twister_loopback_test'
Enter (0-13 tests, 14 -Quit, 15 -Menu) ===>
```

## 4.7.4 Keyboard

The keyboard diagnostic consists of an external and internal loopback. The external loopback requires a passive loopback connector. The internal loopback verifies the keyboard port by transmitting and receiving 128 characters.

CODE EXAMPLE 4-11 identifies the keyboard output message.

**CODE EXAMPLE 4-11 Keyboard Output Message**

```
Enter (0-13 tests, 14 -Quit, 15 -Menu) ===> 3

TEST='keyboard_test'
SUBTEST='internal_loopback'
Enter (0-13 tests, 14 -Quit, 15 -Menu) ===>
```

## 4.7.5 Mouse

The mouse diagnostic performs a keyboard-to-mouse loopback.

CODE EXAMPLE 4-12 identifies the mouse output message.

### CODE EXAMPLE 4-12 Mouse Output Message

```
Enter (0-13 tests, 14 -Quit, 15 -Menu) ==> 4

TEST='mouse_test'
Enter (0-13 tests, 14 -Quit, 15 -Menu) ==>
```

## 4.7.6 Floppy

The floppy diagnostic verifies the diskette drive controller initialization. It also validates the status of a selected disk drive and reads the diskette drive header.

CODE EXAMPLE 4-13 identifies the floppy output message.

---

**Note** – A diskette must be inserted into the diskette drive.

---

### CODE EXAMPLE 4-13 Floppy Output Message

```
Enter (0-13 tests, 14 -Quit, 15 -Menu) ==> 5

TEST='floppy_test'
SUBTEST='floppy_id0_read_test'
Enter (0-13 tests, 14 -Quit, 15 -Menu) ==>
```

## 4.7.7 Parallel Port

The parallel port diagnostic performs the following:

1. `sio-passive-lb` – Sets up the SuperIO configuration register to enable extended/compatible parallel port select, then does a write 0, walk one, write 0 x ff to the data register. It verifies the results by reading the status register.

2. `dma_read` – Enables ECP mode and ECP DMA configuration, and FIFO test mode. Transfers 16 bytes of data from memory to the parallel port device and then verifies the data is in TFIFO.

CODE EXAMPLE 4-14 identifies the parallel port output message.

**CODE EXAMPLE 4-14** Parallel Port Output Message

```
Enter (0-13 tests, 14 -Quit, 15 -Menu) ==> 6

TEST='parallel_port_test'
SUBTEST='dma_read'
Enter (0-13 tests, 14 -Quit, 15 -Menu) ==>
```

## 4.7.8 Serial Port A

The serial port A diagnostic invokes the `uart_loopback` test. This test transmits and receives 128 characters and checks the transaction validity. The following baud rates are tested in asynchronous mode: 460800, 307200, 230400, 153600, 76800, 57600, 38400, 19200, 9600, 4800, 2400, and 800.

CODE EXAMPLE 4-15 identifies the serial port A output message.

**CODE EXAMPLE 4-15** Serial Port A Output Message

```
Enter (0-13 tests, 14 -Quit, 15 -Menu) ==> 7

TEST='uarta_test'
BAUDRATE='1200'
SUBTEST='internal_loopback'
BAUDRATE='1800'
SUBTEST='internal_loopback'
BAUDRATE='2400'
SUBTEST='internal_loopback'
BAUDRATE='4800'
SUBTEST='internal_loopback'
BAUDRATE='9600'
SUBTEST='internal_loopback'
BAUDRATE='19200'
SUBTEST='internal_loopback'
BAUDRATE='38400'
SUBTEST='internal_loopback'
BAUDRATE='57600'
SUBTEST='internal_loopback'
BAUDRATE='76800'
```

**CODE EXAMPLE 4-15** Serial Port A Output Message (*Continued*)

```
SUBTEST='internal_loopback'  
BAUDRATE='115200'  
SUBTEST='internal_loopback'  
BAUDRATE='153600'  
SUBTEST='internal_loopback'  
BAUDRATE='230400'  
SUBTEST='internal_loopback'  
BAUDRATE='307200'  
SUBTEST='internal_loopback'  
BAUDRATE='460800'  
SUBTEST='internal_loopback'  
Enter (0-13 tests, 14 -Quit, 15 -Menu) ===>
```

---

**Note** – The serial port A diagnostic will stall if the TIP line is installed on serial port A. CODE EXAMPLE 4-16 identifies the serial port A output message when the TIP line is installed on serial port A.

---

**CODE EXAMPLE 4-16** Serial Port A Output Message With TIP Line Installed

```
Enter (0-12 tests, 13 -Quit, 14 -Menu) ===> 7  
  
TEST='uarta_test'  
'UART A in use as console - Test not run.'  
Enter (0-12 tests, 13 -Quit, 14 -Menu) ===>
```

## 4.7.9 Serial Port B

The serial port B diagnostic is identical to the serial port A diagnostic.

CODE EXAMPLE 4-17 identifies the serial port B output message.

---

**Note** – The serial port B diagnostic will stall if the TIP line is installed on serial port B.

---

### CODE EXAMPLE 4-17 Serial Port B Output Message

```
Enter (0-13 tests, 14 -Quit, 15 -Menu) ===> 8

TEST='uartb_test'
BAUDRATE='1200'
SUBTEST='internal_loopback'
BAUDRATE='1800'
SUBTEST='internal_loopback'
BAUDRATE='2400'
SUBTEST='internal_loopback'
BAUDRATE='4800'
SUBTEST='internal_loopback'
BAUDRATE='9600'
SUBTEST='internal_loopback'
BAUDRATE='19200'
SUBTEST='internal_loopback'
BAUDRATE='38400'
SUBTEST='internal_loopback'
BAUDRATE='57600'
SUBTEST='internal_loopback'
BAUDRATE='76800'
SUBTEST='internal_loopback'
BAUDRATE='115200'
SUBTEST='internal_loopback'
BAUDRATE='153600'
SUBTEST='internal_loopback'
BAUDRATE='230400'
SUBTEST='internal_loopback'
BAUDRATE='307200'
SUBTEST='internal_loopback'
BAUDRATE='460800'
SUBTEST='internal_loopback'
Enter (0-13 tests, 14 -Quit, 15 -Menu) ===>
```

## 4.7.10 NVRAM

The NVRAM diagnostic verifies the NVRAM operation by performing a write and read to the NVRAM.

CODE EXAMPLE 4-18 identifies the NVRAM output message.

### CODE EXAMPLE 4-18 NVRAM Output Message

```
Enter (0-13 tests, 14 -Quit, 15 -Menu) ====> 9

TEST='nvram_test'
SUBTEST='write/read_patterns'
SUBTEST='write/read_inverted_patterns'
Enter (0-13 tests, 14 -Quit, 15 -Menu) ====>
```

## 4.7.11 Audio

The audio diagnostic performs the following:

1. cs4231\_test - Verifies the cs4231 internal registers.
2. Line-in to line-out external loopback.
3. Microphone to headphone external loopback.

CODE EXAMPLE 4-19 identifies the audio output message.

### CODE EXAMPLE 4-19 Audio Output Message

```
Enter (0-13 tests, 14 -Quit, 15 -Menu) ====> 10

TEST='audio_test'
SUBTEST='cs4231_test'
Codec_ID='8a'
Version_ID='a0'
SUBTEST='ind_reg_test'
Enter (0-13 tests, 14 -Quit, 15 -Menu) ====>
```

## 4.7.12 EIDE

The EIDE diagnostic validates both the EIDE chip and the EIDE bus subsystem.

CODE EXAMPLE 4-20 identifies the EIDE output message.

### CODE EXAMPLE 4-20 EIDE Output Message

```
Enter (0-13 tests, 14 -Quit, 15 -Menu) ==> 11

TEST='ide_test'
SUBTEST='probe-cmd-device'
SUBTEST='hd-and-cd-check'
```

## 4.7.13 Video

The video diagnostic validates the video subsystem.

CODE EXAMPLE 4-21 identifies the video output message.

### CODE EXAMPLE 4-21 Video Output Message

```
Enter (0-13 tests, 14 -Quit, 15 -Menu) ==> 12

TEST='video_test'
    Connect the monitor when running this test
    Please use ttya for display when running this test
    if you are using the screen it may be become unreadable
Enter (0-13 tests, 14 -Quit, 15 -Menu) ==>
```

## 4.7.14 All Above

The all above diagnostic validates the system unit.

CODE EXAMPLE 4-22 identifies the all above output message.

---

**Note** – The all above diagnostic will stall if the TIP line is installed on serial port A or serial port B.

---

### CODE EXAMPLE 4-22 All Above Output Message

```
Enter (0-13 tests, 14 -Quit, 15 -Menu) ==> 13

TEST='all_pci/cheerio_test'
SUBTEST='vendor_id_test'
SUBTEST='device_id_test'
SUBTEST='mixmode_read'
SUBTEST='e2_class_test'
SUBTEST='status_reg_walk1'
SUBTEST='line_size_walk1'
SUBTEST='latency_walk1'
SUBTEST='line_walk1'
SUBTEST='pin_test'

TEST='all_dma/ebus_test'
SUBTEST='dma_reg_test'
SUBTEST='dma_func_test'

TEST='ethernet_test'
Using Onboard Transceiver - Link Up.
SUBTEST='my_channel_reset'
SUBTEST='hme_reg_test'
SUBTEST='global_reg1_test'
SUBTEST='global_reg2_test'
SUBTEST='bmac_xif_reg_test'
SUBTEST='bmac_tx_reg_test'
SUBTEST='mif_reg_test'
SUBTEST='mac_internal_loopback_test'
SUBTEST='10mb_xcvr_loopback_test'
SUBTEST='100mb_phy_loopback_test'
SUBTEST='100mb_twister_loopback_test'

TEST='keyboard_test'
SUBTEST='internal_loopback'
SUBTEST='external_loopback'
```



**CODE EXAMPLE 4-22** All Above Output Message (*Continued*)

```
TEST='mouse_test'  
SUBTEST='mouse_loopback'  
  
TEST='floppy_test'  
SUBTEST='floppy_id0_read_test'  
  
TEST='parallel_port_test'  
SUBTEST='sio_passive_lb'  
SUBTEST='dma_read'  
  
TEST='uarta_test'  
'UART A in use as console - Test not run.'  
  
TEST='uartb_test'  
BAUDRATE='1200'  
SUBTEST='internal_loopback'  
SUBTEST='external_loopback'  
BAUDRATE='1800'  
SUBTEST='internal_loopback'  
SUBTEST='external_loopback'  
BAUDRATE='2400'  
SUBTEST='internal_loopback'  
SUBTEST='external_loopback'  
BAUDRATE='4800'  
SUBTEST='internal_loopback'  
SUBTEST='external_loopback'  
BAUDRATE='9600'  
SUBTEST='internal_loopback'  
SUBTEST='external_loopback'  
BAUDRATE='19200'  
SUBTEST='internal_loopback'  
SUBTEST='external_loopback'  
BAUDRATE='38400'  
SUBTEST='internal_loopback'  
SUBTEST='external_loopback'  
BAUDRATE='57600'  
SUBTEST='internal_loopback'  
SUBTEST='external_loopback'  
BAUDRATE='76800'  
SUBTEST='internal_loopback'  
SUBTEST='external_loopback'  
BAUDRATE='115200'  
SUBTEST='internal_loopback'  
SUBTEST='external_loopback'  
BAUDRATE='153600'  
SUBTEST='internal_loopback'  
SUBTEST='external_loopback'
```

**CODE EXAMPLE 4-22 All Above Output Message (Continued)**

```
BAUDRATE='230400'  
SUBTEST='internal_loopback'  
SUBTEST='external_loopback'  
BAUDRATE='307200'  
SUBTEST='internal_loopback'  
SUBTEST='external_loopback'  
BAUDRATE='460800'  
SUBTEST='internal_loopback'  
SUBTEST='external_loopback'  
  
TEST='nvram_test'  
SUBTEST='write/read_patterns'  
SUBTEST='write/read_inverted_patterns'  
  
TEST='audio_test'  
SUBTEST='cs4231_test'  
Codec_ID='8a'  
Version_ID='a0'  
SUBTEST='external_lpbk'  
Starting External Audio Line-out/Line-in Loopback Test  
Starting External Audio Headphone/Microphone Loopback Test  
SUBTEST='ind_reg_test'  
  
TEST='ide_test'  
SUBTEST='probe-cmd-device'  
SUBTEST='hd-and-cd-check'  
  
TEST='video_test'  
    Please connect the monitor and use ttya/ttyb when running this  
test  
    if you are using the screen it may be become unreadable  
Enter (0-13 tests, 14 -Quit, 15 -Menu) ===>
```

---

## German Acoustic Compliance Statement

This section of the product note contains a compliance statement that is required for customers in Germany:

**ACHTUNG:** Der arbeitsplatzbezogenr Schalldruckpegel nach DIN 45 635 Teil 1000 beträgt 70 Db(A) oder weniger.

