![Sun microsystems logo]

Sun™ Dual Gigabit Ethernet and Dual SCSI/P Adapter Installation and User's Guide

Please
Recycle

Adobe PostScript™

Regulatory Compliance Statements

Your Sun product is marked to indicate its compliance class:

- Federal Communications Commission (FCC) — USA
- Industry Canada Equipment Standard for Digital Equipment (ICES-003) — Canada
- Voluntary Control Council for Interference (VCCI) — Japan
- Bureau of Standards Metrology and Inspection (BSMI) — Taiwan

Please read the appropriate section that corresponds to the marking on your Sun product before attempting to install the product.

FCC Class A Notice

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions:

1. This device may not cause harmful interference.
2. This device must accept any interference received, including interference that may cause undesired operation.

Note: This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy, and if it is not installed and used in accordance with the instruction manual, it may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case the user will be required to correct the interference at his own expense.

Modifications: Any modifications made to this device that are not approved by Sun Microsystems, Inc. may void the authority granted to the user by the FCC to operate this equipment.

ICES-003 Class A Notice - Avis NMB-003, Classe A

This Class A digital apparatus complies with Canadian ICES-003.

Cet appareil numérique de la classe A est conforme à la norme NMB-003 du Canada.

VCCI 基準について

クラス A VCCI 基準について

クラス A VCCI の表示があるワークステーションおよびオプション製品は、クラスA情報技術装置です。これらの製品には、下記の項目が該当します。

> この装置は、情報処理装置等電波障害自主規制協議会 (VCCI) の基準に基づくクラス A情報技術装置です。この装置を家庭環境で使用すると電波妨害を引き起こすことがあります。この場合には使用者が適切な対策を講ずるよう要求されることがあります。

Graphic showing the Japanese VCCI-A regulatory statement.

BSMI Class A Notice

The following statement is applicable to products shipped to Taiwan and marked as Class A on the product compliance label.

警告使用者：
這是甲類的資訊產品，在居住的環境中使用
時，可能會造成射頻 干擾，在這種情況下，
使用者 會被要求採取某些適當的對策。

Graphic showing the BSMI Class A Notice for products shipped to Taiwan.

Declaration of Conformity

Compliance Model Number: Caulfron
Product Family Name: Sun Dual Gigabit Ethernet and Dual SCSI/P adapter

EMC

USA—FCC Class A

This equipment complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions:

1. This equipment may not cause harmful interference.
2. This equipment must accept any interference that may cause undesired operation.

European Union

This equipment complies with the following requirements of the EMC Directive 89/336/EEC:

As Telecommunication Network Equipment (TNE) in Both Telecom Centers and Other Than Telecom Centers per (as applicable):
EN300-386 V.1.3.1 (09-2001) Required Limits:

EN55022/CISPR22	Class A
EN61000-3-2	Pass
EN61000-3-3	Pass
EN61000-4-2	6 kV (Direct), 8 kV (Air)
EN61000-4-3	3 V/m 80-1000MHz, 10 V/m 800-960 MHz, and 1400-2000 MHz
EN61000-4-4	1 kV AC and DC Power Lines, 0.5 kV Signal Lines
EN61000-4-5	2 kV AC Line-Gnd, 1 kV AC Line-Line and Outdoor Signal Lines, 0.5 kV Indoor signal Lines > 10m
EN61000-4-6	3 V
EN61000-4-11	Pass

As Information Technology Equipment (ITE) Class A per (as applicable):
EN55022:1998/CISPR22:1997   Class A
EN55024:1998 Required Limits:

EN61000-4-2	4 kV (Direct), 8 kV (Air)
EN61000-4-3	3 V/m
EN61000-4-4	1 kV AC Power Lines, 0.5 kV Signal and DC Power Lines
EN61000-4-5	1 kV AC Line-Line and Outdoor Signal Lines, 2 kV AC Line-Gnd, 0.5 kV DC Power Lines
EN61000-4-6	3 V
EN61000-4-8	1 A/m
EN61000-4-11	Pass
EN61000-3-2:1995 + A1, A2, A14	Pass
EN61000-3-3:1995	Pass

Safety: This equipment complies with the following requirements of the Low Voltage Directive 73/23/EEC:

EC Type Examination Certificates:

EN60950:2000, 3rd Edition	TÜV Rheinland Certificate No. xxxxxxxxxxxx		
IEC 60950:2000, 3rd Edition	CB Scheme Certificate No. xxxxxxxxxxxx		
Evaluated to all CB Countries			
UL 60950, 3rd Edition, CSA C22.2 No. 60950-00	File:	Vol.	Sec.
UL 60950, 3rd Edition, CSA C22.2 No. 950-00	File:	Vol.	Sec.
FDA DHHS Accession Number (Monitors Only)			

Supplementary Information: This product was tested and complies with all the requirements for the CE Mark.

/S/   /S/

Dennis P. Symanski   DATE   Pamela J. Dullaghan   DATE
Manager, Compliance Engineering   Quality Program Manager
Sun Microsystems, Inc.   Sun Microsystems Scotland, Limited
4150 Network Circle, MPK15-102   Springfield, Linlithgow
Santa Clara, CA 95054 U.S.A.   West Lothian, EH49 7LR
Tel: 650-786-3255   Scotland, United Kingdom
Fax: 650-786-3723   Tel: +44 1 506 672 395   Fax: +44 1 506 670 011

Contents

Figures

Tables

Preface

The *Sun Dual Gigabit Ethernet and Dual SCSI/P Adapter Installation and User's Guide* provides installation instructions for the Sun™ Dual Gigabit Ethernet and Dual SCSI/P adapter. This manual also describes how to configure the driver software.

These instructions are designed for enterprise system administrators with experience installing network hardware and software.

How This Book is Organized

- **Chapter 1 "Product Overview,"** provides a description of the adapter, including hardware and software.
- **Chapter 2, "Installing the Adapter,"** describes how to install the adapter in your system and verify that it has been installed correctly. It then describes how to install the GigaSwift Ethernet driver software using the automated script. Finally, it describes how to edit the network host files after installing the adapter on your system.
- **Chapter 3, "Configuring Driver Parameters,"** describes how to configure the driver parameters used by the Sun Dual Gigabit Ethernet and Dual SCSI/P adapter.
- **Appendix A, "Specifications,"** lists the specifications for the Sun Dual Gigabit Ethernet and Dual SCSI/P adapter.
- **Appendix B, "Installing the Software Manually,"** describes how to install the GigaSwift Ethernet driver software manually.
- ,**Appendix C "Interface Signals, "** lists the interface signals for both the SCSI-2 and RJ-45 connectors.

- **Appendix D, "Diagnostic Software and Troubleshooting Issues,"** provides an overview of the SunVTS diagnostic application and instructions for testing the adapter using the onboard FCode selftest. There is also a section outlining some common troubleshooting issues.

Using UNIX Commands

This document may not contain information on basic UNIX® commands and procedures such as shutting down the system, booting the system, and configuring devices.

See one or more of the following for this information:

- AnswerBook2™ online documentation for the Solaris™ operating environment
- Other software documentation that you received with your system

Typographic Conventions

TABLE P-1   Typographic Conventions

Typeface	Meaning	Examples
AaBbCc123	The names of commands, files, and directories; on-screen computer output	Edit your `.login` file. Use `ls -a` to list all files. `% You have mail.`
AaBbCc123	What you type, when contrasted with on-screen computer output	`% `**`su`** `Password:`
AaBbCc123	Book titles, new words or terms, words to be emphasized	Read Chapter 6 in the *User's Guide*. These are called *class* options. You *must* be superuser to do this.
	Command-line variable; replace with a real name or value	To delete a file, type `rm` *filename*.

Shell Prompts

TABLE P-2   Shell Prompts

Shell	Prompt
C shell	*machine_name%*
C shell superuser	*machine_name#*
Bourne shell and Korn shell	$
Bourne shell and Korn shell superuser	#

Related Documentation

TABLE P-3   Related Documentation

Application	Title
Dynamic Reconfiguration Installation	*Sun Enterprise 6x00, 5x00, 4x00, and 3x00 Systems Dynamic Reconfiguration User's Guide*
Diagnostic Software	*SunVTS User's Guide* *SunVTS Test Reference Manual*
OpenBoot™ Commands	*OpenBoot 3.x Command Reference Manual*

Accessing Sun Documentation Online

The `docs.sun.com`^{sm} web site enables you to access Sun technical documentation on the Web. You can browse the `docs.sun.com` archive or search for a specific book title or subject at:

```
http://docs.sun.com
```

Sun Welcomes Your Comments

Sun is interested in improving its documentation and welcomes your comments and suggestions. You can email your comments to Sun at:

`docfeedback@sun.com`

Please include the part number (817-4339-10) of your document in the subject line of your email.

Product Overview

This chapter provides a description of both the Sun Dual Gigabit Ethernet and Dual SCSI/P adapter hardware and software.

This chapter includes the following sections:

- "Hardware Overview" on page 1
- "Hardware and Software Requirements" on page 4
- "Product Features" on page 5

Hardware Overview

The Sun Dual Gigabit Ethernet and Dual SCSI/P adapter provides two Ultra2 SCSI and two gigabit Ethernet interfaces. The half-size card has a high performance PCI host interface with two UTP RJ-45 connectors for 10/100/1000 Mbps Ethernet and a dual SCSI connector for 80 MBytes/sec SCSI interfaces.

The adapter provides attachment to gigabit Ethernet LANs as well as connectivity to peripheral devices that require the Ultra2 SCSI interface.

Each of the two `Ethernet` channels provides 10BASE-T, 100BASE-TX, or 1000BASE-T networking interfaces. The device driver automatically sets the link speed to 10, 100, or 1000 Mbps and conforms to the IEEE 802.3 Ethernet standard. The single MAC/PHY chip provides the PCI interface and medium access control (MAC) functions. The local transceiver, which connects to an RJ-45 connector, provides the physical-layer functions.

The local transceiver is capable of all the operating speeds and modes listed below:

- 1000 Mbps, full-duplex
- 100 Mbps, full-duplex
- 100 Mbps, half-duplex
- 10 Mbps, full-duplex

■ 10 Mbps, half-duplex

The Sun Dual Gigabit Ethernet and Dual SCSI/P adapter performs autonegotiation with the remote end of the link (the link partner) to select a common mode of operation.

Sun Dual Gigabit Ethernet and Dual SCSI/P adapter also supports a non-autonegotiated mode of operation. You can set the speed and mode using the ndd utility, or, to make sure the changes persist after a system reboot, you can create a ce.conf file. See "To Set Driver Parameters Using a ce.conf File" on page 44 for more information about setting driver parameters.

Ethernet connectors
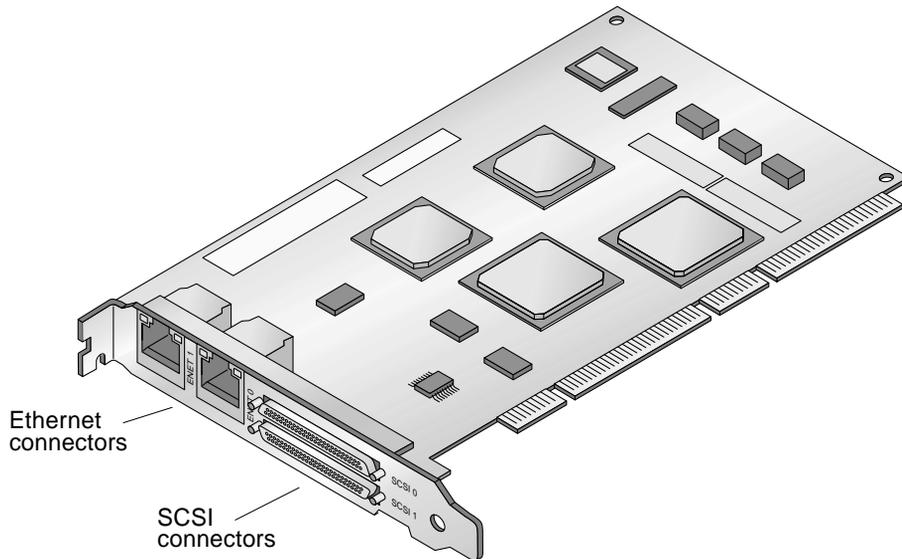
SCSI connectors

FIGURE 1-1 Sun Dual Gigabit Ethernet and Dual SCSI/P Adapter

LED Displays

Two Gigabit Ethernet port connectors are on the front panel of the Sun Dual Gigabit Ethernet and Dual SCSI/P adapter. Each port has two LED's. The explanation of the function of each LED is given in TABLE 1-1.

TABLE 1-1   Front Panel Display LEDs

Color	Meaning
Green	Physical connection to the network
Amber	Activity on the port

Hardware and Software Requirements

Before using the Sun Dual Gigabit Ethernet and Dual SCSI/P adapter, make sure your system meets the following hardware and software requirements:

Hardware and Software	Requirements
Hardware	Sun Ultra™ 30, 60, 80 Netra™ 200, 1125, 1405, 20, 240 Sun Enterprise™ 210, 240, 220R, 420R, 250, 450 Sun Blade™ 100, 150, 1000, 1500, 2000, 2500 Sun Fire™ 280R, 440, 480R, 880R, 1280, 4800, 6800, F12K/15K
Operating Environment	Solaris 8 2/02 and subsequent compatible releases
Firmware	OpenBoot™ PROM version 3.0 or greater
Peripherals Attached to PCI Adapter	All Ultra 2 SCSI devices
SCSI Cables	Only standard, Sun-supported cabling to ensure reliable SCSI interface connections
SunVTS	Version 4.4 and subsequent compatible releases

Supported Cables

The following cables, which you can order from Sun Microsystems, Inc., are required for this adapter:

- X1132A   530-2452-02 (CBL,ULTRA SCSI-3/VHDCI,68P.8M)
- X3832A   530-2453-02 (CBL,ULTRA SCSI-3/VHDCI,68P 2M)
- X3830A   530-2454-02 (CBL,ULTRA SCSI-3/VHDCI,68P 4M)
- X3831A   530-2455-02 (CBL,ULTRA SCSI-3/VHDCI,68P 10M)

Product Features

Following is a list of features for the Sun Dual Gigabit Ethernet and Dual SCSI/P adapter:

PCI Bus Interface

- PCI Local Bus Rev 2.2 compliant (6.8-inch x 4.2-inch short card)
- 33/66-MHz, 32- or 64-bit bus master
- Universal PCI slot (3.3V and 5V)
- Hot-plug capable
- Infinite Burst capable on Ultra III machines

Gigabit Ethernet Interfaces

- UTP RJ-45 connectors for 10/100/1000 Mbps Ethernet
- Two ports of IEEE 10/100/1000BASE-T compliant network interfaces
- Low CPU utilization—Frees up server system resource and bandwidth
- Dynamic Reconfiguration (DR) and Redundancy/Failover support
- Load balancing for RX packets among multiple CPUs
- RAS support

Ultra2 SCSI Interfaces

- 68-pin dual 0.8mm high-density SCSI interfaces
- 8-, 16-, 32-, and 64-byte PCI bursts
- Two independent 16-bit Ultra 2 SCSI channels
- Each SCSI interface supports single-ended (SE) or low volted differential (LVD) SCSI connector
    - Maximum LVD transfer rate of 80 MBytes/sec
    - Maximum SE transfer rate of 40 MBytes/sec
- 64-bit PCI transfers at 33 MHz
- Dynamic Reconfiguration (DR) and Redundancy/Failover support
- RAS support

Diagnostic Support

- User executable selftest using OpenBoot PROM
- SunVTS™ diagnostic tool

Installing the Adapter

This chapter describes how to install the adapter in your system and verify that it has been installed correctly. It then describes how to install the driver software using the automated script. Finally, it describes how to edit the network host files after installing the adapter on your system.

This chapter contains the following sections:

- "Installing the Adapter Without Dynamic Reconfiguration" on page 7
- "Installing the Adapter With Dynamic Reconfiguration" on page 19
- "Using the Installation Script" on page 19
- "Configuring the Network Host Files" on page 26

Installing the Adapter Without Dynamic Reconfiguration

▼ To Install the Adapter

> **Note –** The following instructions describe the basic tasks required to install the adapter. Refer to your system installation or service manual for detailed PCI adapter installation instructions.

1. **Halt and power off your system.**

2. **Power off all of the peripherals connected to your system.**

3. **Open the system unit.**

4. **Attach the adhesive copper strip of the antistatic wrist strap to the metal casing of the power supply. Wrap the other end twice around your wrist, with the adhesive side against your skin.**

5. **Holding the PCI adapter by the edges, unpack it and place it on an antistatic surface.**

6. **Using a No. 1 Phillips screwdriver, remove the PCI filler panel from the slot in which you want to insert the PCI adapter.**

   Save the filler panel screw for Step 9.

7. **Holding the PCI adapter by the edges, align the adapter edge connector with the PCI slot. Slide the adapter face plate into the small slot at the end of the PCI opening.**

8. **Applying even pressure at both corners of the adapter, push the PCI adapter until it is firmly seated in the slot.**

Caution – Do not use excessive force when installing the adapter into the PCI slot. You may damage the adapter's PCI connector. If the adapter does not seat properly when you apply even pressure, remove the adapter and carefully reinstall it again.

9. **Secure the adapter to the PCI slot using the screw you removed in Step 6.**

10. **Detach the wrist strap and close the system unit.**

11. **Connect the SCSI cables.**

12. **Connect the Ethernet cables.**

    a. **Connect one end of the Ethernet cable to the port on the card.**

    b. **Connect the other end of the Ethernet cable to an active Ethernet network.**

▼ To Verify the Installation

After you have installed the Sun Dual Gigabit Ethernet and Dual SCSI/P adapter, but *before* you boot your system, perform the following tasks to verify the installation. Refer to the your Solaris documentation for the detailed instructions.

Note – Verification is not required if your system supports dynamic reconfiguration (DR).

1. **Power on the system, and when the banner appears, press the Stop-A key sequence to interrupt the boot process and display the OpenBoot (ok) prompt.**

2. **Use the** `show-devs` **command to list the system devices.**

You should see PCI adapter output similar to the example below.

```
ok show-devs
/pci@2,2000/pci@2              -----> Bridge Chip
/pci@2,2000/pci@2/scsi@2,1  -----> SCSI port #2
/pci@2,2000/pci@2/scsi@2     -----> SCSI port #1
/pci@1f,2000/pci@1/network@1 -----> Network #1
/pci@1f,2000/pci@1/network@0 -----> Network #2
/pci@2,2000/pci@2/scsi@2,1/tape -> SCSI Devices that can be Connected
/pci@2,2000/pci@2/scsi@2,1/disk
/pci@2,2000/pci@2/scsi@2/tape
/pci@2,2000/pci@2/scsi@2/disk
```

If these devices are not listed, check that the adapter is properly seated and reinstall the adapter, if necessary.

3. **List the SCSI interfaces on your system.**

```
ok probe-scsi-all:
/pci@1e,600000/pci@4/scsi@2,1
Target d
    Unit 0   Disk     HITACHI DK32EJ72NSUN72G PQ08
Target e
    Unit 0   Disk     HITACHI DK32EJ72NSUN72G PQ08
Target f
    Unit 0   Disk     HITACHI DK32EJ72NSUN72G PQ08

/pci@1e,600000/pci@4/scsi@2
Target 8
    Unit 0   Disk     SEAGATE ST336607LSUN36G 0507
Target 9
    Unit 0   Disk     SEAGATE ST336607LSUN36G 0507
Target a
    Unit 0   Disk     SEAGATE ST336607LSUN36G 0507
```

- `scsi@2,1` identifies the first SCSI interface on the Dual SCSI/P adapter.
- `scsi@2` identifies the second SCSI interface on the Dual SCSI/P adapter.

If these devices are not listed, check that the adapter is properly seated and reinstall the adapter, if necessary.

4. **View the** `.properties` **file for a list of device properties.**

   The `.properties` command displays the specific information about the installed adapter.

```
ok cd /pci@1e,600000/pci@4/scsi@2,1
ok .properties
assigned-addresses      81021110 00000000 00001100 00000000 00000100
                         83021114 00000000 03704000 00000000 00002000
                         8302111c 00000000 03706000 00000000 00002000
                         82021130 00000000 03720000 00000000 00010000
device_type             scsi-2
clock-frequency         02625a00
reg                     00021100 00000000 00000000 00000000 00000000
                         01021110 00000000 00000000 00000000 00000100
                         03021114 00000000 00000000 00000000 00002000
                         0302111c 00000000 00000000 00000000 00002000
                         02021130 00000000 00000000 00000000 00010000
name                    scsi
compatible              pci1000,b.7
                         pci1000,b
                         pciclass,010000
                         pciclass,0100
fcode-rom-offset        00000000
devsel-speed            00000001
class-code              00010000
interrupts              00000002
latency-timer           00000040
cache-line-size         00000010
max-latency             00000040
min-grant               00000011
revision-id             00000007
device-id               0000000b
vendor-id               00001000
```

Note – Your `.properties` list may differ slightly from the previous examples. The properties list will be affected by the system the SCSI is connected to.

5. **When you finish looking at the** `.properties` **values, be sure to enter the following command:**

```
ok device-end
```

6. **Type the following to verify the second SCSI device:**

```
ok cd /pci@1e,600000/pci@4/scsi@2
ok .properties
assigned-addresses      81021010 00000000 00001000 00000000 00000100
                         83021014 00000000 03700000 00000000 00002000
                         8302101c 00000000 03702000 00000000 00002000
                         82021030 00000000 03710000 00000000 00010000
device_type             scsi-2
clock-frequency         02625a00
reg                     00021000 00000000 00000000 00000000 00000000
                         01021010 00000000 00000000 00000000 00000100
                         03021014 00000000 00000000 00000000 00002000
                         0302101c 00000000 00000000 00000000 00002000
                         02021030 00000000 00000000 00000000 00010000
name                    scsi
compatible              pci1000,b.7
                         pci1000,b
                         pciclass,010000
                         pciclass,0100
fcode-rom-offset        00000000
devsel-speed            00000001
class-code              00010000
interrupts              00000001
latency-timer           00000040
cache-line-size         00000010
max-latency             00000040
min-grant               00000011
revision-id             00000007
device-id               0000000b
vendor-id               00001000
```

Note – Your .properties list may differ slightly from the previous example. The properties list will be affected by the system the SCSI is connected to.

7. **When you finish looking at the** .properties **values, be sure to enter the following command:**

```
ok device-end
```

8. List the network devices on your system.

```
ok show-nets
```

Use the `show-nets` command to list the system devices. You should see the full path name of the `network` devices, similar to the example below. In this example, the `network@0` and the `network@1` devices are the Sun Dual Gigabit Ethernet and Dual SCSI/P adapter and the `network@2` is the onboard Ethernet device.

```
ok show-nets
a) /pci@1f,700000/network@2
b) /pci@1e,600000/pci@4/network@1
c) /pci@1e,600000/pci@4/network@0
q) NO SELECTION
Enter Selection, q to quit: b
```

- network@0 identifies the first Ethernet port
- network@1 identifies the second Ethernet port.

Note – If you do not see the device listed, check that the adapter is properly seated and, if necessary, reinstall the adapter.

9. View the device that you installed.

Using the previous example, type:

```
cd /pci@1e,600000/pci@4/network@1
```

10. **View the** `.properties` **file for a list of device properties.**

The `.properties` command displays the specific information about the installed adapter.

```
ok cd /pci@1e,600000/pci@4/network@1
ok .properties
assigned-addresses      82020810 00000000 03400000 00000000 00200000
                         82020830 00000000 03600000 00000000 00100000
d-fru-len               00000000
d-fru-off               0000e800
d-fru-dev               eeprom
s-fru-len               00000800
s-fru-off               0000e000
s-fru-dev               eeprom
compatible              pci100b,35.30
                         pci100b,35
                         pciclass,020000
                         pciclass,0200
                         pci108e,abba
reg                     00020800 00000000 00000000 00000000 00000000
                         02020810 00000000 00000000 00000000 00200000
                         02020830 00000000 00000000 00000000 00100000
address-bits            00000030
max-frame-size          00004000
network-interface-type  ethernet
device_type             network
name                    network
local-mac-address       00 03 ba 0e 95 46
version                 Sun PCI Dual Gigabit Ethernet 10/100/1000Base-T
FCode 2.12 03/11/21
phy-type                mif
board-model             501-6635
model                   SUNW,pci-ce
fcode-rom-offset        00000000
66mhz-capable
fast-back-to-back
devsel-speed            00000002
class-code              00020000
interrupts              00000001
latency-timer           00000040
cache-line-size         00000010
max-latency             00000040
min-grant               00000040
revision-id             00000030
device-id               00000035
vendor-id               0000100b
```

> **Note –** If you are going to set the `local-mac-address?` variable, note the local MAC address of your device at this time. See "Setting the `local-mac-address?` Variable" on page 17 for more information.

11. **Type the following when you finish looking at the** `.properties` **values:**

```
ok device-end
```

12. Type the following to verify the second Ethernet device:

```
ok cd /pci@1e,600000/pci@4/network@0
ok .properties
assigned-addresses      82020010 00000000 03200000 00000000 00200000
                         82020030 00000000 03100000 00000000 00100000
d-fru-len               00000000
d-fru-off               0000e800
d-fru-dev               eeprom
s-fru-len               00000800
s-fru-off               0000e000
s-fru-dev               eeprom
compatible              pci100b,35.30
                         pci100b,35
                         pciclass,020000
                         pciclass,0200
                         pci108e,abba
reg                     00020000 00000000 00000000 00000000 00000000
                         02020010 00000000 00000000 00000000 00200000
                         02020030 00000000 00000000 00000000 00100000
address-bits            00000030
max-frame-size          00004000
network-interface-type  ethernet
device_type             network
name                    network
local-mac-address       00 03 ba 0e 95 45
version                 Sun PCI Dual Gigabit Ethernet 10/100/1000Base-T
FCode 2.12 03/11/21
phy-type                mif
board-model             501-6635
model                   SUNW,pci-ce
fcode-rom-offset        00000000
66mhz-capable
fast-back-to-back
devsel-speed            00000002
class-code              00020000
interrupts              00000001
latency-timer           00000040
cache-line-size         00000010
max-latency             00000040
min-grant               00000040
revision-id             00000030
device-id               00000035
vendor-id               0000100b
```

13. **Type the following when you finish looking at the** `.properties` **values:**

```
ok device-end
```

Setting the `local-mac-address?` Variable

Note – Enabling the `local-mac-address?` variable is only required if you boot from the network.

Every Sun Dual Gigabit Ethernet and Dual SCSI/P adapter comes with two unique media access control (MAC) addresses that represent the 48-bit Ethernet address for that interface.

A system is not obligated to use this assigned MAC address if it has a system-wide MAC address. In such cases, the system-wide MAC address applies to all network interfaces on the system.

The `local-mac-address?` variable of the network device specifies the network address (system-wide or `local-mac-address?`) used for booting the system. If the `local-mac-address?` NVRAM variable is set to `true` the system will set the MAC address for the network interfaces of the Sun Dual Gigabit Ethernet and Dual SCSI/P adapter to be the addresses provided by the adapter. If this variable is set to `false`, the system will set the MAC address for both of the adapter's network interfaces to be the same as the *system* MAC address.

To start using the MAC address assigned to the network interface of the Sun Dual Gigabit Ethernet and Dual SCSI/P adapter, set the NVRAM configuration variable `local-mac-address?` to `true`.

```
ok setenv local-mac-address? true
```

▼ To Set the Gigabit Ethernet Device as the Primary Boot Device

Use this procedure *only* if you want the Gigabit Ethernet device to be your primary boot device.

1. **List the network devices on your system.**

```
ok show-nets
a) /pci@1f,2000/pci@1/network@1
b) /pci@1f,2000/pci@1/network@0
c) /pci@1f,4000/network@1,1
q) NO SELECTION
Enter Selection, q to quit:q
```

2. **Set the Sun Dual Gigabit Ethernet and Dual SCSI/P adapter device to be your default boot device by typing:**

```
ok setenv boot-device /pci@9,600000/pci@1/pci@0/network@0
```

Note that the command shown in this example sets your boot device to be port 0 on the network portion of the card

Rebooting the System

After verifying the adapter installation, use the boot -r command to perform a reconfiguration boot on your system.

```
ok boot -r
```

Installing the Adapter With Dynamic Reconfiguration

If you have a Sun Enterprise system that supports dynamic reconfiguration (DR), you do not have to reboot your system after installing the adapter.

The process of adding and configuring an adapter with DR involves (1) connecting the attachment point and (2) configuring its occupant. In most cases, the cfgadm(1M) command can perform both steps at once.

Note – If you have a Sun Enterprise system that supports dynamic reconfiguration (DR), refer to the *Sun Enterprise Dynamic Reconfiguration User's Guide* and your system's documentation for further information about dynamic reconfiguration.

Using the Installation Script

The *Sun Dual Gigabit Ethernet and Dual SCSI/P* CD provides an automated installation for the GigaSwift Ethernet driver software. In most cases, you can use the following procedure and execute the installation script. If you have any problems, see Appendix B for manual installation procedures.

When you have executed the script, select the interface card that is installed on your system. For this product, select the "Dual Gigabit Ethernet and Dual SCSI/P Adapter."

The script first checks to ensure that the software driver package has already been installed. If the package is installed, the script verifies the patch level and updates the driver with the updated patch if necessary. If the packages are not installed on Solaris 8 or 9 software, the installation script installs the latest pre-patched packages. If the system does not require updating because the patch level is at or beyond the current install version, the script exits with an appropriate message.

▼ To Install the Software Using the Installation Script

1. **At the system console, become superuser (root).**

2. **Insert the** *Sun Dual Gigabit Ethernet Dual SCSI/P Adapter CD* **into a CD-ROM drive that is connected to your system.**

   If your system is running *Sun Enterprise Volume Manager*, it should automatically mount the CD-ROM to the /cdrom/cdrom0 directory.

   If your system is not running *Sun Enterprise Volume Manager*, mount the CD-ROM as follows:

   ```
   # mount -F hsfs -o ro /dev/dsk/c0t6d0s2 /cdrom/cdrom0
   ```

   Note – If you are installing the software from the download site, follow the online download instructions.

3. **Change to the directory were the installation script resides:**

   ```
   # cd /media_path/
   ```

   Where:

   - *media_path* = /cdrom/cdrom0, if you are installing from the CD-ROM

   Or

   - unzipped *GigaSwift_path*/Sun_GigaSwift_Ethernet_Driver

4. **Execute the install script:**

```
# ./install
```

The script displays a menu of products:

```
     Product Installation

  1 - Dual Gigabit Ethernet Dual SCSI/P Adapter
  2 - GigaSwift Ethernet Adapter (copper/fiber)
  3 - GigaSwift Ethernet Compact PCI Adapter
  4 - Quad GigaSwift Ethernet Adapter

       Please select the product to install: 1
```

5. **Select 1, the Sun Dual Gigabit Ethernet and Dual SCSI/P adapter, and answer the questions, if required, in the installation script.**

Note – After the installation script runs, you will see a summary of the software package or patch that was installed. You can also find a log of the installation in /var/tmp. The log file information is displayed when the installation completes unless the software meets or exceeds the required package or patch level.

Example 1: All Requirements Met

Following is an example of an installation script output if the driver and patch requirements are met:

CODE EXAMPLE 2-1   Script Output When Requirements are Met

```
          Product Installation

   1 - Dual Gigabit Ethernet Dual SCSI/P Adapter
   2 - GigaSwift Ethernet Adapter (copper/fiber)
   3 - GigaSwift Ethernet Compact PCI Adapter
   4 - Quad GigaSwift Ethernet Adapter

        Please select the product to install: 1

 Dual GigaSwift Ethernet Adapter Driver Installation.
 Copyright 2004 Sun Microsystems, Inc.  All rights reserved.
```

CODE EXAMPLE 2-1 Script Output When Requirements are Met

```
Use is subject to license terms.

*** Checking for correct version of GigaSwift Ethernet Driver
Packages...
*** Checking for correct version of GigaSwift Ethernet Driver
Patches...

*** Checking for correct version of VLAN Utility Driver Packages..
*** Checking for correct version of VLAN Utility Driver Patches...

The CE/VLAN driver(s) installed on this system for:

        Dual GigaSwift Ethernet Adapter Driver

meets or exceeds the required package or patch version. No changes
to the system are required - exiting.
```

Example 2: Updates Required

The following is an example of an installation script output, if the packages are
found and need updating by patching:

CODE EXAMPLE 2-2 Script Output When Packages Require Patches

```
            Product Installation

   1 - Dual Gigabit Ethernet Dual SCSI/P Adapter
   2 - GigaSwift Ethernet Adapter (copper/fiber)
   3 - GigaSwift Ethernet Compact PCI Adapter
   4 - Quad GigaSwift Ethernet Adapter

        Please select the product to install: 1

Dual GigaSwift Ethernet Adapter Driver Installation.

Copyright 2004 Sun Microsystems, Inc.  All rights reserved.
Use is subject to license terms.

*** Checking for correct version of GigaSwift Ethernet Driver
Packages...
*** Checking for correct version of GigaSwift Ethernet Driver
Patches...

*** Checking for correct version of VLAN Utility Driver Packages..
*** Checking for correct version of VLAN Utility Driver Patches...
```

CODE EXAMPLE 2-2   Script Output When Packages Require Patches *(Continued)*

```
This script is about to take the following actions:
- Install GigaSwift Ethernet Patch: 112817-16.
- Install VLAN Ethernet Utility Patch: 114600-02.

Press return to continue, or 'Q' followed by a return to quit:

*** Installing patch 112817-16 for Solaris 9...

Checking installed patches...
Verifying sufficient filesystem capacity (dry run method)...
Installing patch packages...

Patch number 112817-16 has been successfully installed.
See /var/sadm/patch/112817-16/log for details

Patch packages installed:
  SUNWcea
  SUNWceax
  SUNWced
  SUNWcedu
  SUNWcedu

*** Installing patch 114600-02 for Solaris 9...

Checking installed patches...
Verifying sufficient filesystem capacity (dry run method)...
Installing patch packages...

Patch number 114600-02 has been successfully installed.
See /var/sadm/patch/114600-02/log for details

Patch packages installed:
  SUNWvld
  SUNWvldu
  SUNWvldx

Installation completed. Summary:

 Dual GigaSwift Ethernet Adapter Driver has been updated using
patch
 112817-16. It will be necessary to reboot the system to reload
 the updated driver.

 Optional VLAN Utility has been updated using  114600-02. It
 may be necessary to reboot the system to reload the updated
 driver.
```

```
A log of this Install can be found at:
    /var/tmp/GigaSwift.install.2003.11.17.1213
```

Example 3: Packages Missing

Following is an example of an installation script output if the package(s) are not found:

CODE EXAMPLE 2-3 Script Output When Packages Are Not Found

```
           Product Installation

   1 - Dual Gigabit Ethernet Dual SCSI/P Adapter
   2 - GigaSwift Ethernet Adapter (copper/fiber)
   3 - GigaSwift Ethernet Compact PCI Adapter
   4 - Quad GigaSwift Ethernet Adapter

        Please select the product to install: 1

Dual GigaSwift Ethernet Adapter Driver Installation.

Copyright 2003 Sun Microsystems, Inc.  All rights reserved.
Use is subject to license terms.

*** Checking for correct version of GigaSwift Ethernet Driver
Packages...

*** Checking for correct version of VLAN Utility Driver Packages..

The following supporting driver(s) can be used with Dual GigaSwift
Ethernet Adapter
Driver:

    VLAN Utility Driver software
Install the Optional VLAN Utility Driver Software [y,n,?,q] y

This script is about to take the following actions:
- Install Dual GigaSwift Ethernet Adapter Driver packages.
- Optionally Install VLAN Utility Driver Software.

Press return to continue, or 'Q' followed by a return to quit:

*** Installing Sun GigaSwift Ethernet  packages...
Copyright 2002 Sun Microsystems, Inc.  All rights reserved.
Use is subject to license terms.
```

```
System configuration files modified but ce driver not loaded or
attached.

Installation of <SUNWced> was successful.
Copyright 2002 Sun Microsystems, Inc.  All rights reserved.
Use is subject to license terms.

Installation of <SUNWcedx> was successful.
Copyright 2002 Sun Microsystems, Inc.  All rights reserved.
Use is subject to license terms.

Installation of <SUNWcedu> was successful.
Copyright 2002 Sun Microsystems, Inc.  All rights reserved.
Use is subject to license terms.

Installation of <SUNWcea> was successful.
Copyright 2002 Sun Microsystems, Inc.  All rights reserved.
Use is subject to license terms.

Installation of <SUNWceax> was successful.

*** Installing Sun VLAN Utility  packages...
Copyright 2001 Sun Microsystems, Inc. All rights reserved.

Installation of <SUNWvld> was successful.
Copyright 2001 Sun Microsystems, Inc. All rights reserved.

Installation of <SUNWvldx> was successful.
Copyright 2001 Sun Microsystems, Inc. All rights reserved.

Installation of <SUNWvldu> was successful.

Installation completed. Summary:

 GigaSwift Ethernet Driver packages have been installed.

 Optional VLAN Utility packages have been installed.

A log of this Install can be found at:
    /var/tmp/GigaSwift.install.2003.11.17.1205
```

Configuring the Network Host Files

After installing the driver software, you must create a `hostname.ce#` file (Where `#` = the instance number of the `ce` interface.) for the adapter's Ethernet interface. You must also create both an IP address and a host name for its Ethernet interface in the `/etc/hosts` file.

1. **At the command line, use the `grep` command to search the `/etc/path_to_inst` file for `ce` interfaces.**

```
# grep ce /etc/path_to_inst
"/pci@1e,600000/pci@4/network@0" 0 "ce"
"/pci@1e,600000/pci@4/network@1" 1 "ce"
```

In the example above, the device instance is from a Sun Dual Gigabit Ethernet and Dual SCSI/P adapter. For clarity, the instance number is in bold italics.

2. **Use the `ifconfig` command to setup the adapter's `ce` interface.**

Use the `ifconfig` command to assign an IP address to the network interface. Type the following at the command line, replacing *ip_address* with the adapter's IP address:

```
# ifconfig ce0 plumb ip_address up
```

Refer to the `ifconfig`(1M) man page and the Solaris documentation for more information.

- If you want a set-up that will remain the same after you reboot, create an `/etc/hostname.ce`*number* file, where *number* corresponds to the instance number of the `ce` interface you plan to use.

  To use the adapter's `ce` interface in the Step 1 example, create an `/etc/hostname.ce0` file, where `0` is the number of the `ce` interface. If the instance number were *1*, the filename would be `/etc/hostname.ce1`.

- Do not create an `/etc/hostname.ce#` file for a Sun Dual Gigabit Ethernet and Dual SCSI/P adapter interface you plan to leave unused.

- The `/etc/hostname.ce#` file must contain the hostname for the appropriate `ce` interface.

- The host name should have an IP address and should be listed in the `/etc/hosts` file.

- The host name should be different from any other host name of any other interface, for example: /etc/hostname.ce0 and /etc/hostname.ce1 cannot share the same host name.

The following example shows the /etc/hostname.ce# file required for a system called zardoz that has a Sun Dual Gigabit Ethernet and Dual SCSI/P adapter (zardoz-11).

```
# cat /etc/hostname.hme0
zardoz
# cat /etc/hostname.ce0
zardoz-11
```

3. **Create an appropriate entry in the** /etc/hosts **file for each active** ce **interface.**

For example:

```
# cat /etc/hosts
#
# Internet host table
#
127.0.0.1      localhost
129.144.10.57 zardoz     loghost
129.144.11.83 zardoz-11
```

Configuring Driver Parameters

This chapter describes how to configure the driver parameters used by the Sun Dual Gigabit Ethernet and Dual SCSI/P adapter. This chapter contains the following sections:

- "GigaSwift Ethernet Driver Parameter Values and Definitions" on page 29
- "Setting `ce` Driver Parameters" on page 39
- "Usability Enhancements to the Driver" on page 46
- "GigaSwift Ethernet Driver Operating Statistics" on page 47

GigaSwift Ethernet Driver Parameter Values and Definitions

TABLE 3-1 describes the parameters and settings for the `ce` device driver.

TABLE 3-1   `ce` Driver Parameter, Status, and Descriptions

Parameter	Status	Description
`instance`	Read and write	Device instance
`adv-autoneg-cap`	Read and write	Operational mode parameter
`adv-1000fdx-cap`	Read and write	Operational mode parameter
`adv-1000hdx-cap`	Read and write	Operational mode parameter
`adv-100T4-cap`	Read and write	Operational mode parameter
`adv-100fdx-cap`	Read and write	Operational mode parameter
`adv-100hdx-cap`	Read and write	Operational mode parameter
`adv-10fdx-cap`	Read and write	Operational mode parameter

TABLE 3-1 ce Driver Parameter, Status, and Descriptions *(Continued)*

Parameter	Status	Description
adv-10hdx-cap	Read and write	Operational mode parameter
adv-asmpause-cap	Read and write	Flow control parameter
adv-pause-cap	Read and write	Flow control parameter
master-cfg-enable	Read and write	Gigabit link clock mastership controls
master-cfg-value	Read and write	Gigabit link clock mastership controls
use-int-xcvr	Read and write	
enable-ipg0	Read and write	Enable additional delay before transmitting a packet
ipg0	Read and write	Additional delay before transmitting a packet
ipg1	Read and write	Interpacket gap parameter
ipg2	Read and write	Interpacket gap parameter
rx-intr-pkts	Read and write	Receive interrupt blanking values
rx-intr-time	Read and write	Receive interrupt blanking values
red-dv4to6k	Read and write	Random early detection and packet drop vectors
red-dv6to8k	Read and write	Random early detection and packet drop vectors
red-dv8to10k	Read and write	Random early detection and packet drop vectors
red-dv10to12k	Read and write	Random early detection and packet drop vectors
tx-dma-weight	Read and write	PCI interface parameter
rx-dma-weight	Read and write	PCI interface parameter
infinite-burst	Read and write	PCI interface parameter
disable-64bit	Read and write	PCI interface parameter
accept-jumbo	Read and write	Enable jumbo frames

Operational Mode Parameters

The following parameters determine the transmit and receive speed and duplex. TABLE 3-2 describes the operational mode parameters and their default values

TABLE 3-2   Operational Mode Parameters

Parameter	Description
adv-autoneg-cap	Local interface capability advertised by the hardware 0 = Forced mode 1 = Autonegotiation (default)
adv-1000fdx-cap	Local interface capability advertised by the hardware 0 = Not 1000 Mbit/sec full-duplex capable 1 = 1000 Mbit/sec full-duplex capable (default)
adv-1000hdx-cap	Local interface capability advertised by the hardware 0 = Not 1000 Mbit/sec half-duplex capable 1 = 1000 Mbit/sec half-duplex capable (default)
adv-100T4-cap	Local interface capability advertised by the hardware 0 = Not 100T4 capable (default) 1 = 100T4 capable
adv-100fdx-cap	Local interface capability advertised by the hardware 0 = Not 100 Mbit/sec full-duplex capable 1 = 100 Mbit/sec full-duplex capable (default)
adv-100hdx-cap	Local interface capability advertised by the hardware 0 = Not 100 Mbit/sec half-duplex capable 1 = 100 Mbit/sec half-duplex capable (default)
adv-10fdx-cap	Local interface capability advertised by the hardware 0 = Not 10 Mbit/sec full-duplex capable 1 = 10 Mbit/sec full-duplex capable (default)
adv-10hdx-cap	Local interface capability advertised by the hardware 0 = Not 10 Mbit/sec half-duplex capable 1 = 10 Mbit/sec half-duplex capable (default)

Note – If a parameter's initial setting is 0, it cannot be changed. If you try to change it, it will revert back to 0.

If all these parameters are set to 1, autonegotiation will use the highest speed possible. If all these parameters are set to 0, you will receive the following error message:

```
NOTICE: Last setting will leave ce0 with no link capabilities.
WARNING: ce0: Restoring previous setting.
```

Flow Control Parameters

The ce device is capable of sourcing (transmitting) and terminating (receiving) pause frames conforming to the IEEE 802.3x Frame Based Link Level Flow Control Protocol. In response to received flow control frames, the ce device can slow down its transmit rate. On the other hand, the ce device is capable of sourcing flow control frames, requesting the link partner to slow down, provided that the link partner supports this feature. By default, the driver advertises both transmit and receive pause capability during autonegotiation.

TABLE 3-3 provides flow control keywords and describes their function.

TABLE 3-3   Read-Write Flow Control Keyword Descriptions

Keyword	Description
adv_asmpause_cap	The adapter supports asymmetric pause, which means it can pause only in one direction. 0=Off (default) 1=On
adv_pause_cap	This parameter has two meanings depending on the value of adv_asmpause_cap. (Default=0) If adv_asmpause_cap = 1 while adv_pause_cap = 1 pauses are received. If adv_asmpause_cap = 1 while adv_pause_cap = 0 pauses are transmitted. If adv_asmpause_cap = 0 while adv_pause_cap = 1 pauses are sent and received. If adv_asmpause_cap = 0 then adv_pause_cap determines whether Pause capability is on or off.

Gigabit Link Clock Mastership Controls

The concept of link clock mastership was introduced with gigabit twisted-pair technology. One side of the link is the master that provides the link clock and the other is the slave that uses the link clock. Once this relationship is established, the link is up, and data can be communicated. Two physical layer parameters control whether your side is the master or the slave or whether mastership is negotiated with the link partner.

TABLE 3-4   Forced Mode Parameter

Parameter	Description
master-cfg-enable	Determines whether or not during the auto-negotiation process the link clock mastership is setup automatically.
master-cfg-value	• If the master-cfg-enable parameter is set then the mastership is not set up automatically but depends on the value of master-cfg-value. If the master-cfg-value is set then the physical layer expects the local device to be the link master. If it is not set then it expects the link partner to be the master. • If Auto-negotiation is not enabled, then the value of master-cfg-enable is ignored and the value of master-cfg-value is key to the link clock mastership. If the master-cfg-value is set, then the physical layer expects the local device to be the link master. If the value is not set then it expects the link partner to the master.

Caution – Do not adjust the link clock mastership parameters unless you clearly understand the settings of the link partner. Incorrect link clock mastership configuration results in link up failure.

Interpacket Gap Parameters

The ce device supports a programmable mode called enable_ipg0.

When an instance receives a packet with enable_ipg0 set (the default), it adds an additional time delay before transmitting the packet. This delay, set by the ipg0 parameter, is in addition to the delay set by the ipg1 and ipg2 parameters. The additional ipg0 delay helps to reduce collisions.

If enable_ipg0 is disabled, the value of ipg0 is ignored and no additional delay is set. Only the delays set by ipg1 and ipg2 will be used. Disable enable_ipg0 if other systems keep sending a large number of back-to-back packets. Systems that have enable_ipg0 set might not have enough time on the network.

You can add the additional delay by setting the ipg0 parameter from 0 to 255, which is the media byte time delay.

TABLE 3-5 defines the enable_ipg0 and ipg0 parameters.

TABLE 3-5   Parameters Defining enable_ipg0 and ipg0

Parameter	Values	Description
enable_ipg0	0 1	enable_ipg0 reset enable_ipg0 set (Default=8)
ipg0	0 to 255	The additional time delay (or gap) before transmitting a packet (after receiving the packet) (Default=8)

The ce device supports the programmable Interpacket Gap (IPG) parameters ipg1 and ipg2. The total IPG is the sum of ipg1 and ipg2. The total IPG is 0.096 microseconds for the link speed of 1000 Mbps.

TABLE 3-6 lists the default values and allowable values for the IPG parameters.

TABLE 3-6   Read-Write Interpacket Gap Parameter Values and Descriptions

Parameter	Values (Byte-time)	Description
ipg1	0 to 255	Interpacket gap 1 (Default = 8)
ipg2	0 to 255	Interpacket gap 2 (Default = 4)

By default, the driver sets ipg1 to 8-byte time and ipg2 to 4-byte time, which are the standard values. (Byte time is the time it takes to transmit one byte on the link, with a link speed of 1000 Mbps.)

If your network has systems that use longer IPG (the sum of ipg1 and ipg2), and if those machines seem to be slow in accessing the network, increase the values of ipg1 and ipg2 to match the longer IPGs of other machines.

Interrupt Parameters

TABLE 3-7 describes the receive interrupt blanking values.

TABLE 3-7   RX Blanking Register for Alias Read

Field Name	Values	Description
rx_intr_pkts	0 to 511	Interrupt after this number of packets have arrived since the last packet was serviced. A value of zero indicates no packet blanking. (Default=3)
rx_intr_time	0 to 524287	Interrupt after 4.5 US ticks have elapsed since the last packet was serviced. A value of zero indicates no time blanking. (Default=1250)

Random Early Drop Parameters

TABLE 3-8 describes the RX random early detection 8-bit vectors, which allows you to enable random early drop (RED) thresholds. When received packets reach the RED range packets are dropped according to the preset probability. The probability should increase when the fifo level increases. Control packets are never dropped and are not counted in the statistics.

TABLE 3-8   RX Random Early Detecting 8-Bit Vectors

Field Name	Values	Description
red_dv4to6k	0 to 255	Random early detection and packet drop vectors for when fifo threshold is greater than 4096 bytes and less than 6,144 bytes. Probability of drop can be programmed on a 12.5 percent granularity. For example, if bit 0 is set the first packet out of every eight will be dropped in this region. (Default=0)

TABLE 3-8   RX Random Early Detecting 8-Bit Vectors

Field Name	Values	Description
red_dv6to8k	0 to 255	Random early detection and packet drop vectors for when fifo threshold is greater than 6,144 bytes and less than 8,192 bytes. Probability of drop can be programmed on a 12.5 percent granularity. For example, if bit 8 is set the first packet out of every eight will be dropped in this region. (Default=0)
red_dv8to10k	0 to 255	Random early detection and packet drop vectors for when fifo threshold is greater than 8,192 bytes and less than 10,240 bytes. Probability of drop can be programmed on a 12.5 percent granularity. For example, if bit 16 is set the first packet out of every eight will be dropped in this region. (Default=0)
red_dv10to12k	0 to 255	Random early detection and packet drop vectors for when fifo threshold is greater than 10,240 bytes and less than 12,288 bytes. Probability of drop can be programmed on a 12.5 percent granularity. For example, if bit 24 is set the first packet out of every eight will be dropped in this region. (Default=0)

PCI Bus Interface Parameters

These parameters allow you to modify PCI interface features to gain better PCI interperformance for a given application.

TABLE 3-9   PCI Bus Interface Parameters

Parameter	Description
tx_dma_weight	Determine the multiplication factor for granting credit to the TX side during a weighted round robin arbitration. Values are 0 to 3. (Default=0) Zero means no extra weighting. The other values are power of 2 extra weighting, on that traffic. For example of tx_dma_weight = 0 and rx_dma_weight = 3 then as long as RX traffic is continuously arriving its priority will be 8 times greater than TX to access the PCI

TABLE 3-9   PCI Bus Interface Parameters

Parameter	Description
rx_dma_weight	Determine the multiplication factor for granting credit to the RX side during a weighted round robin arbitration. Values are 0 to 3. (Default=0)
infinite_burst	allows the infinite burst capability to be utilized. When this is in effect and the system supports infinite burst. The adapter will not free the bus until complete packets are transferred across the bus. Values are 0 or 1. (Default=0)
disable_64bit	Switches off 64 bit capability of the adapter. In some cases, it is useful to switch off this feature. Values are 0 or 1. (Default=0, which enables 64 bit capability)

Jumbo Frames

The variable `accept-jumbo` allows the driver to instruct the upper layers that the driver is operating in jumbo frames mode.

Note – The Solaris 8 software allows you to set the MTU to greater than 1500, but it does not automatically change the default.

TABLE 3-10   `accept-jumbo` Parameters

Parameter	Values	Description
accept-jumbo	1	Enables jumbo frames mode and sets MTU to greater than 1500.
	0	Jumbo frames is not enabled. MTU is at 1500.

Although Jumbo Frames is configurable in 10/100 mode, Jumbo Frames is only supported in the Gigabit (1000Mbps) mode. Configuring Jumbo Frames allows the Ethernet interfaces to send/receive packets of up to 9216 bytes. However, the actual transfer size depends on the switch capability.

Refer to the documentation that came with your switch for exact commands to configure Jumbo Frames support.

Jumbo Frames with Trunking 1.3

When using Jumbo Frames with Trunking 1.3, you must explicitly enable Jumbo Frames on each link participating in the Trunk.

▼ To Configure Jumbo Frames Using ndd

1. **Bring down and unplumb the interface before using** ndd **to set Jumbo Frame.**

2. **Set the instance number:**

```
% ndd -set /dev/ce instance 0
```

In this example, the instance number set is 0.

3. **Set the instance to accept jumbo frames:**

```
% ndd -set /dev/ce accept-jumbo 1
```

4. **Plumb the interface up:**

```
% ifconfig ce0 plumb xx.xx.xx.xx up
```

Where *xx.xx.xx.xx* = the IP address of the interface.

5. **Repeat steps 1 throug 4 to set both the Ethernet interfaces to use jumbo frames.**

Using the infinit_burst Parameter

Because only UltraSPARC™ III machines support the infinit_burst parameter it is turned off by default. If you are using an UltraSPARC III machine, you can enable this parameter. To discover if your machine is an UltraSPARC III, run the following command:

```
% modinfo | grep pcisch
```

If you get the following response:

```
22 101e1663   d5c1 109   1  pcisch (PCI Bus nexus driver 1.194)
```

Then the machine is an UltraSPARC III, and you can safely enable the `infinit_burst` parameter.

Setting `ce` Driver Parameters

You can set the `ce` device driver parameters in two ways:

- Using the `ndd` utility
- Using the `ce.conf` file

If you use the `ndd` utility, the parameters are valid only until you reboot the system. This method is good for testing parameter settings.

To set parameters so they remain in effect after you reboot the system, create a `/platform/sun4u/kernel/drv/ce.conf` file and add parameter values to this file when you need to set a particular parameter for a device in the system.

Setting Parameters Using the `ndd` Utility

Use the `ndd` utility to configure parameters that are valid until you reboot the system. The `ndd` utility supports any networking driver, which implements the Data Link Provider Interface (DLPI).

The following sections describe how you can use the `ce` driver and the `ndd` utility to modify (with the `-set` option) or display (without the `-set` option) the parameters for each `ce` device.

▼ To Specify Device Instances for the `ndd` Utility

Before you use the `ndd` utility to get or set a parameter for a `ce` device, you must specify the device instance for the utility.

1. **Check the** `/etc/path_to_inst` **file to identify the instance associated with a particular device.**

```
# grep ce /etc/path_to_inst
"/pci@1e,600000/pci@4/network@0" 0 "ce"
"/pci@1e,600000/pci@4/network@1" 1 "ce"
```

In the example above, the two Gigabit Ethernet instances are from the installed adapter. The instance numbers are in bold italics for clarity.

2. **Use the instance number to select the device.**

```
# ndd -set /dev/ce instance instance
```

The device remains selected until you change the selection.

Noninteractive and Interactive Modes

You can use the `ndd` utility in two modes:

- Noninteractive
- Interactive

In noninteractive mode, you invoke the utility to execute a specific command. Once the command is executed, you exit the utility. In interactive mode, you can use the utility to get or set more than one parameter value. (Refer to the `ndd`(1M) man page for more information.)

▼ To Use the `ndd` Utility in Noninteractive Mode

This section describes how to modify and display parameter values.

● **To modify a parameter value, use the** `-set` **option.**

If you invoke the `ndd` utility with the `-set` option, the utility passes *value*, which must be specified, down to the named `/dev/ce` driver instance, and assigns it to the parameter:

```
# ndd -set /dev/ce parameter value
```

When you change any adv parameter, a message similar to the following appears:

```
ce0: xcvr addr:0x01 - link up 1000 Mbps full duplex
```

● **To display the value of a parameter, specify the parameter name and omit the value.**

When you omit the `-set` option, a query operation is assumed and the utility queries the named driver instance, retrieves the value associated with the specified parameter, and prints it:

```
# ndd /dev/ce parameter
```

▼ To Use the ndd Utility in Interactive Mode

● **To modify a parameter value in interactive mode, specify** ndd /dev/ce**, as shown below.**

The ndd utility then prompts you for the name of the parameter:

```
# ndd /dev/ce
name to get/set? (Enter the parameter name or ? to view all
parameters)
```

After typing the parameter name, the ndd utility prompts you for the parameter value (see TABLE 3-1 through TABLE 3-12).

- **To list all the parameters supported by the** ce **driver, type** ndd /dev/ce.

  (See TABLE 3-1 through TABLE 3-12 for parameter descriptions.)

```
# ndd /dev/ce
name to get/set ? ?
?                              (read only)
instance                       (read and write)
adv_autoneg_cap                (read and write)
adv_1000fdx_cap                (read and write)
adv_1000hdx_cap                (read and write)
adv_100T4_cap                  (read and write)
adv_100fdx_cap                 (read and write)
adv_100hdx_cap                 (read and write)
adv_10fdx_cap                  (read and write)
adv_10hdx_cap                  (read and write)
adv_asmpause_cap               (read and write)
adv_pause_cap                  (read and write)
master_cfg_enable              (read and write)
master_cfg_value               (read and write)
use_int_xcvr                   (read and write)
enable_ipg0                    (read and write)
ipg0                           (read and write)
ipg1                           (read and write)
ipg2                           (read and write)
rx_intr_pkts                   (read and write)
rx_intr_time                   (read and write)
red_dv4to6k                    (read and write)
red_dv6to8k                    (read and write)
red_dv8to10k                   (read and write)
red_dv10to12k                  (read and write)
tx_dma_weight                  (read and write)
rx_dma_weight                  (read and write)
infinite_burst                 (read and write)
disable_64bit                  (read and write)
accept_jumbo                   (read and write)
name to get/set ?
#
```

Setting the Autonegotiation Mode

By default, autonegotiation is set to on. This means that the adapter communicates with its link partner to determine a compatible network speed, duplex mode, and flow control capability.

▼ To Disable Auto-negotiation Mode

If your network equipment does not support autonegotiation, or if you want to specify your network speed, you can set autonegotiation to `off` on the `ce` device.

Note – Disabling auto-negotiation mode can cause collisions.

---

1. **Set the following driver parameters to the values that are described in the documentation that shipped with your link partner (for example, a switch):**
   - `adv-1000fdx-cap`
   - `adv-100fdx-cap`
   - `adv-100hdx-cap`
   - `adv-10fdx-cap`
   - `adv-10hdx-cap`
   - `adv-asmpause-cap`
   - `adv-pause-cap`

   See TABLE 3-2 for the descriptions and possible values of these parameters.

Note – According to IEEE 802.3 specifications, when using 1000 mbps speed over copper media, auto-negotiation is not optional. In the absense of auto-negotiation, (for example, while using forced mode), link syncing between link partners may not happen and the link may not come up.

---

2. **Set the** `adv-autoneg-cap` **parameter to** `0`.

```
# ndd -set /dev/ce adv-autoneg-cap 0
```

When you change any `ndd` link parameter, a message similar to the following appears:

```
xcvr addr:0x00 - link up 100 Mbps full duplex
```

Setting Parameters Using the `ce.conf` File

Specify the driver parameter properties on a per-device basis by creating a `ce.conf` file in the `/platform/sun4u/kernel/drv` directory. Use a `ce.conf` file when you need to set a particular parameter for a device in the system. The parameters you set are read and write parameters that are listed in "GigaSwift Ethernet Driver Parameter Values and Definitions" on page 29.

The man pages for `prtconf`(1M) and `driver.conf`(4) include additional details. The next procedure shows an example of setting parameters in a `ce.conf` file.

● **To access any man page, type the** `man` **command plus the name of the man page. For example, to access man pages for** `prtconf`(1M), **type:**

```
% man prtconf
```

▼ To Set Driver Parameters Using a `ce.conf` File

1. **Obtain the hardware path names for the** `ce` **devices in the device tree.**

   a. **Check the** `/etc/driver_aliases` **file to identify the name associated with a particular device:**

   ```
   # grep ce /etc/driver_aliases
   ce "pci108e,abba"
   ce "pci100b,35"
   ```

   b. **Locate the path names and the associated instance numbers are in the** `/etc/path_to_inst` **file.**

   ```
   # grep ce /etc/path_to_inst
   "pci@1e,600000/pci@4/network@0" 0 "ce"
   "pci@1e,600000/pci@4/network@1" 1 "ce"
   ```

   - In the previous example:
     - The first part within the double quotes specifies the hardware node name in the device tree.
     - The number not enclosed in quotes is the instance number (shown in bold italics).
     - The last part in double quotes is the driver name.

- In the hardware node name, the last component after the last / character and before the @ character is the device name.
- The path name before the last component is the parent name.
- The number after the final @ character within quotes is referred to as unit-address.

To identify a PCI device unambiguously in the `ce.conf` file, use the name, parent name, and the unit-address for the device. Refer to the `pci(4)` man page for more information about the PCI device specification.

In the first line in the previous example:

- `parent` = "pci@1e,600000/pci@4"
- `unit-address` = "0"

In the second line in the previous example:

- `parent` = "pci@1e,600000/pci@4"
- `unit-address` = "1"

2. **Set the parameters for the above devices in the** `/platform/sun4u/kernel/drv/ce.conf` **file.**

   In the following example, the `adv_autoneg_cap` and `adv_1000fdx_cap` parameters are set for all Sun GigaSwift Ethernet devices. See the `driver.conf(4)` man page for more information.

   ```
   adv-autoneg-cap=0 adv-1000fdx-cap=0;
   ```

   In the following example, the `adv-autoneg-cap` and `adv-1000fdx-cap` parameters are set for a single instance of the Sun GigaSwift Ethernet device.

   ```
   name="pci100b,35" parent="pci@1e,600000/pci@4" unit-address="0"
   adv-autoneg-cap=0 adv-100hdx-cap=0 adv-100fdx-cap=1 adv-1000fdx-cap=0 adv-
   10hdx-cap=0 adv-10fdx-cap=0 adv-1000hdx-cap=0 adv-100T4-cap=0;
   ```

   Note – The difference between setting parameters for all Sun GigaSwift Ethernet devices and setting parameters for a single instance of the device depends on whether you include the `name=`, `parent=`, and `unit-address=`. If you omit these definitions, the settings become global to all Sun GigaSwift Ethernet instances.

3. **Save the** `ce.conf` **file.**

Usability Enhancements to the Driver

In older Ethernet device drivers, determining the link status for a driver required a two-step process:

First, you had to set the instance:

```
# ndd -set /dev/ce instance 2
```

Then you had to get the link status:

```
# ndd -get /dev/ce link_status
1
```

Starting in the Solaris 9 operating system, this method of determining the link status is discouraged, and in some cases, it is removed completely as a driver feature.

The new improved approach moves all read-only parameters from the ndd options into kstat. This simplifies getting link status information by allowing you to do it with one simple command.

● **To get the link status of a driver, type the following command:**

```
# kstat ce:n | grep link_up
link_up                         0
```

Where *n* = instance.

For more information about the kstat command, refer to the kstat man page.

GigaSwift Ethernet Driver Operating Statistics

These statistics are part of the statistics presented by the `netstat -k` command.

TABLE 3-11 describes the read-only Media Independent Interface (MII) capabilities. These parameters define the capabilities of the hardware. The Gigabit Media Independent Interface (GMII) supports all of the following capabilities.

TABLE 3-11   Read-Only `ce` Device Capabilities

Parameter	Description (Local interface Capabilities)
`cap_autoneg`	0 = Not capable of autonegotiation 1 = Autonegotiation capable
`cap_1000fdx`	Local interface full-duplex capability 0 = Not 1000 Mbit/sec full-duplex capable 1 = 1000 Mbit/sec full-duplex capable
`cap_1000hdx`	Local interface half-duplex capability 0 = Not 1000 Mbit/sec half-duplex capable 1 = 1000 Mbit/sec half-duplex capable
`cap_100fdx`	Local interface full-duplex capability 0 = Not 100 Mbit/sec full-duplex capable 1 = 100 Mbit/sec full-duplex capable
`cap_100hdx`	Local interface half-duplex capability 0 = Not 100 Mbit/sec half-duplex capable 1 = 100 Mbit/sec half-duplex capable
`cap_10fdx`	Local interface full-duplex capability 0 = Not 10 Mbit/sec full-duplex capable 1 = 10 Mbit/sec full-duplex capable
`cap_10hdx`	Local interface half-duplex capability 0 = Not 10 Mbit/sec half-duplex capable 1 = 10 Mbit/sec half-duplex capable
`cap_asm_pause`	Local interface flow control capability 0 = Not asymmetric pause capable 1 = Asymmetric pause (from the local device) capable
`cap_pause`	Local interface flow control capability 0 = Not Symmetric pause capable 1 = Symmetric pause capable

Reporting the Link Partner Capabilities

TABLE 3-12 describes the read-only link partner capabilities.

TABLE 3-12   Read-Only Link Partner Capabilities

Parameter	Description
lp_cap_autoneg	0 = No autonegotiation 1 = Autonegotiation
lp_cap_1000fdx	0 = No 1000 Mbit/sec full-duplex transmission 1 = 1000 Mbit/sec full-duplex
lp_cap_1000hdx	0 = No 1000 Mbit/sec half-duplex transmission 1 = 1000 Mbit/sec half-duplex
lp_cap_100fdx	0 = No 100 Mbit/sec full-duplex transmission 1 = 100 Mbit/sec full-duplex
lp_cap_100hdx	0 = No 100 Mbit/sec half-duplex transmission 1 = 1000 Mbit/sec half-duplex
lp_cap_10fdx	0 = No 10 Mbit/sec full-duplex transmission 1 = 10 Mbit/sec full-duplex
lp_cap_10hdx	0 = No 10 Mbit/sec half-duplex transmission 1 = 10 Mbit/sec half-duplex
lp_cap_asm_pause	0 = Not asymmetric pause capable 1 = Asymmetric pause towards link partner capability
lp_cap_pause	0 = Not symmetric pause capable 1 = Symmetric pause capable

If the link partner is *not* capable of autonegotiation (when lp_autoneg_cap is 0), the remaining information described in TABLE 3-12 is not relevant and the parameter value = 0.

If the link partner *is* capable of autonegotiation (when lp_autoneg_cap is 1), then the speed and mode information is displayed when you use autonegotiation and the link partner capabilities.

TABLE 3-13 describes the netstat -k transmit and receive parameters:

TABLE 3-13   Transmit and Receive Parameters

Parameter	Description
xcvr_inits	Number of Physical layer re-initializations every time you change link parameters using NDD this increments.
rev_id	Revision ID of the GigaSwift Ethernet device useful for recognition of device being used in the field.
xcvr_addr	GMII/MII Physical layer device address for management interface.
xcvr_id	GMII/MII Physical layer device Identification Decimal copy of MII registers 2 and 3.
lb_mode	Copy of the Loopback mode the device is in, if any.
qos_mode	When zero, the TX queues operate in a simple round robin queueing scheme, based on TCP/UDP destination port number. If set the TX queues operate in a scheme designed to provide VLAN priorities.
tx_starts	Number of times that the driver attempted to transmit a packet.
tx_dma_bind_fail	Number of times a page table entry was not available to allow the driver to map the kernel memory to device accessible memory for transmission.
tx_queue0	Number of packets queued for transmission on the first hardware transmit queue.
tx_queue1	Number of packets queued for transmission on the second hardware transmit queue.
tx_queue2	Number of packets queued for Transmission on the third hardware transmit queue.
tx_queue3	Number of packets queued for Transmission on the fourth hardware transmit queue.
tx_max_pend	Maximum number of transmits pending on any of the four queues.
rx_hdr_pkts	Number of packets received that were less than 256 bytes.
rx_mtu_pkts	Number of packets received that were greater than 256 bytes and less than 1514 bytes.
rx_split_pkts	Number of packets that were split across two pages.
rx_no_comp_wb	Number of times the hardware cannot post completion entries for received data.
rx_no_buf	Number of times the hardware cannot receive data because there is no more receive buffer space.
rx_new_pages	Number of pages that got replaced during reception.

TABLE 3-13 Transmit and Receive Parameters

Parameter	Description
rx_new_hdr_pgs	Number of pages that were filled with packets less than 256 bytes that got replaced during reception.
rx_new_mtu_pgs	Number of pages that were filled with packets greater than 256 bytes and less than 1514 that got replaced during reception.
rx_new_nxt_pgs	Number of pages that contained packets that were split across pages that got replaced during reception.
rx_hdr_drops	Number of times a whole page of packets less than 256 bytes was dropped because the driver was unable to map a new one to replace it.
rx_mtu_drops	Number of times a whole page of packets greater than 256 bytes and less than 1514 was dropped because the driver was unable to map a new one to replace it.
rx_nxt_drops	Number of times a page with a split packet was dropped because the driver was unable to map a new one to replace it.
rx_rel_flow	Number of times the driver was told to release a flow.

▼ To Check Link Partner Settings

- **If you are running the Solaris 8 operating evironment, type the** `netstat -k` **command as superuser:**

```
# netstat -k ce0
ce0:
ipackets 0 ipackets64 0 ierrors 0 opackets 0 opackets64 0
oerrors 0 collisions 0 rbytes 0 rbytes64 0 obytes 0 obytes64 0
multircv 0 multixmt 0 brdcstrcv 0 brdcstxmt 0 norcvbuf 0
noxmtbuf 0 first_collision 0 excessive_collisions 0 late_collisions 0
peak_attempts 0 length_err 0 alignment_err 0 crc_err 0 code_violations 0
ifspeed 1000000000 promisc off rev_id 48 xcvr_inits 1 xcvr_inuse 1
xcvr_addr 1 xcvr_id 536894584 cap_autoneg 1 cap_1000fdx 1 cap_1000hdx 1
cap_100T4 0 cap_100fdx 1 cap_100hdx 1 cap_10fdx 1 cap_10hdx 1
cap_asmpause 0 cap_pause 0 lp_cap_autoneg 1 lp_cap_1000fdx 1
lp_cap_1000hdx 1 lp_cap_100T4 0 lp_cap_100fdx 1 lp_cap_100hdx 1
lp_cap_10fdx 1 lp_cap_10hdx 1 lp_cap_asmpause 0 lp_cap_pause 1
link_T4 0 link_speed 1000 link_duplex 2 link_asmpause 0 link_pause 0
link_up 1 mac_mtu 0 lb_mode 0 qos_mode 0 trunk_mode 0 tx_inits 0
tx_starts 0 tx_nocanput 0 tx_msgdup_fail 0 tx_allocb_fail 0
tx_no_desc 0 tx_dma_bind_fail 0 tx_uflo 0 tx_queue0 0 tx_queue1 0
tx_queue2 0 tx_queue3 0 tx_hdr_pkts 0 tx_ddi_pkts 0 tx_dvma_pkts 0
mdt_reqs 0 mdt_hdr_bufs 0 mdt_pld_bufs 0 mdt_pkts 0 mdt_hdrs 0
mdt_plds 0 mdt_hdr_bind_fail 0 mdt_pld_bind_fail 0 tx_jumbo_pkts 0
tx_max_pend 0 rx_inits 0 rx_jumbo_pkts 0 rx_hdr_pkts 0 rx_mtu_pkts 0
rx_split_pkts 0 rx_no_buf 0 rx_no_comp_wb 0 rx_ov_flow 0
rx_len_mm 0 rx_tag_err 0 rx_nocanput 0 rx_msgdup_fail 0
rx_allocb_fail 0 rx_new_pages 0 rx_new_hdr_pgs 0 rx_new_mtu_pgs 0
rx_new_nxt_pgs 0 rx_hdr_drops 0 rx_mtu_drops 0 rx_nxt_drops 0
rx_rel_flow 0 rx_rel_bit 0 rx_pkts_dropped 0 pci_bus_speed 66
pci_err 0 pci_rta_err 0 pci_rma_err 0 pci_parity_err 0 pci_bad_ack_err 0
pci_drto_err 0 pci_dmawz_err 0 pci_dmarz_err 0 rx_taskq_waits 0
ipackets_cpu00 0 ipackets_cpu01 0 ipackets_cpu02 0 ipackets_cpu03 0
```

- **If you are running the Solaris 9 operating environment, type the** `kstat` **command as superuser:**

CODE EXAMPLE 3-1   Output from the `kstat` Command

```
# kstat ce:0
module: ce                                    instance: 0
name:   ce0                        class:    net
         alignment_err                        0
         brdcstrcv                            0
         brdcstxmt                            0
         cap_1000fdx                          1
```

CODE EXAMPLE 3-1 Output from the kstat Command *(Continued)*

```
# kstat ce:0
        cap_1000hdx                 1
        cap_100T4                   0
        cap_100fdx                  1
        cap_100hdx                  1
        cap_10fdx                   1
        cap_10hdx                   1
        cap_asmpause                0
        cap_autoneg                 1
        cap_pause                   0
        code_violations             0
        collisions                  0
        crc_err                     0
        crtime                      2345.407585961
        excessive_collisions        0
        first_collision             0
        ierrors                     0
        ifspeed                     0
        ipackets                    0
        ipackets64                  0
        ipackets_cpu00              0
        ipackets_cpu01              0
        ipackets_cpu02              0
        ipackets_cpu03              0
        late_collisions             0
        lb_mode                     0
        length_err                  0
        link_T4                     0
        link_asmpause               0
        link_duplex                 0
        link_pause                  0
        link_speed                  0
        link_up                     0
        lp_cap_1000fdx              0
        lp_cap_1000hdx              0
        lp_cap_100T4                0
        lp_cap_100fdx               0
        lp_cap_100hdx               0
        lp_cap_10fdx                0
        lp_cap_10hdx                0
        lp_cap_asmpause             0
        lp_cap_autoneg              0
        lp_cap_pause                0
        multircv                    0
        multixmt                    0
        norcvbuf                    0
        noxmtbuf                    0
```

CODE EXAMPLE 3-1 Output from the kstat Command *(Continued)*

```
# kstat ce:0
        obytes                          0
        obytes64                        0
        oerrors                         0
        opackets                        0
        opackets64                      0
        pci_bad_ack_err                 0
        pci_dmarz_err                   0
        pci_dmawz_err                   0
        pci_drto_err                    0
        pci_err                         0
        pci_parity_err                  0
        pci_rma_err                     0
        pci_rta_err                     0
        peak_attempts                   0
        promisc                         off
        qos_mode                        0
        rbytes                          0
        rbytes64                        0
        rev_id                          1
        rx_allocb_fail                  0
        rx_hdr_drops                    0
        rx_hdr_pkts                     0
        rx_inits                        0
        rx_len_mm                       0
        rx_msgdup_fail                  0
        rx_mtu_drops                    0
        rx_mtu_pkts                     0
        rx_new_hdr_pgs                  0
        rx_new_mtu_pgs                  0
        rx_new_nxt_pgs                  0
        rx_new_pages                    0
        rx_no_buf                       0
        rx_no_comp_wb                   0
        rx_nocanput                     0
        rx_nxt_drops                    0
        rx_ov_flow                      0
        rx_pkts_dropped                 0
        rx_rel_bit                      0
        rx_rel_flow                     0
        rx_split_pkts                   0
        rx_tag_err                      0
        rx_taskq_waits                  0
        snaptime                        2352.506837979
        tx_allocb_fail                  0
        tx_ddi_pkts                     0
        tx_dma_bind_fail                0
```

```
# kstat ce:0
        tx_dma_hdr_bind_fail         0
        tx_dma_pld_bind_fail         0
        tx_dvma_pkts                 0
        tx_hdr_pkts                  0
        tx_inits                     0
        tx_max_desc                  0
        tx_max_pend                  0
        tx_msgdup_fail               0
        tx_no_desc                   0
        tx_nocanput                  0
        tx_queue0                    3
        tx_queue1                    0
        tx_queue2                    0
        tx_queue3                    0
        tx_starts                    0
        tx_uflo                      0
        xcvr_addr                    1
        xcvr_id                      2121809
        xcvr_inits                   1
        xcvr_inuse                   1
```

▼ To Discover Link Partner Capabilities

● **Use the** kstat **command to discover link partner capabilities.**

```
# kstat ce:0 | grep lp_
        lp_cap_1000fdx               1
        lp_cap_1000hdx               1
        lp_cap_100T4                 0
        lp_cap_100fdx                0
        lp_cap_100hdx                0
        lp_cap_10fdx                 0
        lp_cap_10hdx                 0
        lp_cap_asmpause              0
        lp_cap_autoneg               1
        lp_cap_pause                 0
```

▼ To Discover Link Settings

● **Use the** `kstat` **command to discover link settings.**

```
# kstat ce:0 | grep link
        link_T4                 0
        link_asmpause           0
        link_duplex             2
        link_pause              0
        link_speed              1000
        link_up                 1
```

Configuration Checking

The jumbo frame configuration checking occurs at Layer 2 and Layer 3, depending on the configuration methods.

Layer 2 Configuration Checking

You can view the MTU configuration of a `ce` instance at any time using the `kstat` command. The `kstat mac_mtu` variable represents the complete size of the Ethernet frame, which includes the Ethernet header, maximum payload, and `crc`. This value is consistant with the definition of MTU presented by switch vendors.

Use the following command to get the information:.

```
# kstat ce:0 | grep mac_mtu
```

Layer 3 Configuration Checking

You can do this by invoking the `ifconfig` command on a preconfigured interface as shown in the above examples:.

```
ifconfig ce123000
ce123000: flags=1000843<.....> mtu 9194
inet ip_address-vlan-1 .......
ether 8:0:20:c4:51:df
```

Specifications

This appendix lists the specifications for the Sun Dual Gigabit Ethernet and Dual SCSI/P adapter. It contains the following sections:

- "Physical Characteristics" on page 57
- "Performance Specifications" on page 58
- "Power Requirements" on page 58

Physical Characteristics

Dimension	Measurement
Length	175 mm (6.8 inches)
Width	107 mm (4.2 inches)

Performance Specifications

Feature	Specification
PCI clock	33/66 MHz max.
PCI data burst transfer rate	1up to 64-byte bursts
SCSI synchronous transfer rate	20 x 80 MBytes/sec.
SCSI asynchronous transfer rate	Max. 12 MBytes/sec 16-bit Max. 6 MBytes/sec 8-bit
Data burst size	4 GByte max.
PCI Data/Address Lines	AD63-0
PCI modes	Master/slave
SCSI interface	Single-ended (SE) /Low voltage differential (LVD)
SCSI Bus parity	Yes
SCSI 8-Bit Bus devices	Yes
SCSI 16-Bit Bus devices	Yes
Maximum number of devices per port	15
100BASE-TX transfer rate	<= 100 Mbps (in each direction for full duplex)
10BASE-T transfer rate	<= 10 Mbps (in each direction for full duplex)

Power Requirements

Specification	Measurement
Maximum power consumption	16.65 watts
Voltage	3.3V and 5V

Installing the Software Manually

This appendix describes how to install the driver software manually. It contains the following sections:

- "Installing the Driver Software Manually" on page 59
- "Verifying Patches" on page 62

Installing the Driver Software Manually

The *Sun Dual Gigabit Ethernet and Dual SCSI/P Adapter* CD contains the driver software required to operate the adapter. When you update your Solaris operating system, refer to the documentation that shipped with the *Solaris Supplement* CD-ROM for a listing of the available network drivers. Install the Sun GigaSwift Ethernet driver for your updated version of the Solaris operating system.

1. **Become superuser.**

2. **Insert the** *Sun Dual Gigabit Ethernet and Dual SCSI/P Adapter* **CD into a CD-ROM drive that is connected to your system.**

   - If your system is running Sun Enterprise Volume Manager™, it should automatically mount the CD-ROM to the `/cdrom/cdrom0` directory.

- If your system is not running Volume Manager, mount the CD-ROM as follows:

```
# mkdir /cdrom
# mkdir /cdrom/cdrom0
# mount -F hsfs -o ro /dev/dsk/c0t6d0s2 /cdrom/cdrom0
```

You will see the following files and directories in the
/cdrom/sun_gigaswift_ethernet_driver directory or the
/cdrom/cdrom0 directory, depending on how you mounted the CD-ROM.

TABLE 3-14 Files and Directories on the CD-ROM

File or Directory	Contents
Copyright	U.S. copyright file
FR_Copyright	French copyright file
DualGigabitEthernet/ Solaris_*OS-VER*/Packages/	Contains the Sun GigaSwift Ethernet software packages applicable to your version (*OS-VER)* of the Solaris software: • SUNWcea—32-bit adb macros • SUNWceax—64-bit adb macros • SUNWced.u—32-bit adapter driver • SUNWcedx.u—64-bit adapter driver • SUNWcedu—adapter driver headers • SUNWcem—man pages (optional, Solaris 8 only) The optional VLAN packages for Solaris 8 only: • SUNWvld—VLAN utility routines • SUNWvldx—VLAN utility routines (64-bit) • SUNWvldu—VLAN utility headers
DualGigabitEthernet/Docs/en	Contains PDF copy of the user manual

Note – If you intend to use VLAN, you must install VLAN packages when you
install the software packages.

3. . **Determine if the** ce **and** vlan **drivers are installed on the system:**

```
#/usr/bin/pkginfo SUNWced SUNWcedx SUNWvld SUNWvldx
system      SUNWced      Sun GigaSwift Ethernet Adapter (32-bit Driver)
system      SUNWcedx     Sun GigaSwift Ethernet Adapter (64-bit Driver)
system      SUNWvld      Sun Ethernet Vlan Utility Routines
system      SUNWvldx     Sun Ethernet Vlan Utility Routines (64-bit)
```

- If the output of the command is similar to the example in Step 3, skip to Step 6.
- If nothing is reported back on SUNWced or SUNWcedx, then follow Step 4 and select SUNWcea, SUNWceax, SUNWced.u, SUNWcedx.u, SUNWcedu, and SUNWcem (if present).
- If nothing is reported back for SUNWvld or SUNWvldx, then follow Step 4 and select SUNWvld, SUNWvldx, SUNWvldu packages.
- If nothing is reported back, the follow Step 4 and select "all" packages.

4. **Install the software packages by typing the following at the command line, replacing the** *OS_VER* **with your version (8 or 9) of the Solaris Operating System:**

```
# /usr/sbin/pkgadd -d /cdrom/cdrom0/DualGigabitEthernet/Solaris_OS_VER/
Packages/
```

For example, if your system is running the Solaris 8 Operating Environment, you would type the following:

```
# /usr/sbin/pkgadd -d /cdrom/cdrom0/DualGigabitEthernet/Solaris_8/Packages/
```

A menu similar to the following displays:

```
The following packages are available:
  1  SUNWcea        Sun GigaSwift Ethernet Adapter Driver 32 bit adb Macros
                    (sparc) 1.0,REV=2001.05.04
  2  SUNWceax       Sun GigaSwift Ethernet Adapter Driver 64 bit adb Macros
                    (sparc) 1.0,REV=2001.05.04
  3  SUNWced.u      Sun GigaSwift Ethernet Adapter (32-bit Driver)
                    (sparc.sun4u) 1.0,REV=2001.05.04
  4  SUNWcedu       Sun GigaSwift Ethernet Adapter Driver Headers
                    (sparc) 1.0,REV=2001.05.04
  5  SUNWcedx.u     Sun GigaSwift Ethernet Adapter (64-bit Driver)
                    (sparc.sun4u) 1.0,REV=2001.05.04
  6  SUNWcem        Sun GigaSwift Ethernet Adapter Driver Man Pages
                    (sparc) 1.0,REV=2001.05.04
  7  SUNWvld        Sun Ethernet Vlan Utility Routines
                    (sparc) 1.0,REV=2001.05.04
  8  SUNWvldu       Sun Ethernet Vlan Utility Headers
                    (sparc) 1.0,REV=2001.05.04
  9  SUNWvldx       Sun Ethernet Vlan Utility Routines (64-bit)
                    (sparc) 1.0,REV=2001.05.04

Select package(s) you wish to process (or 'all' to process
all packages). (default: all) [?,??,q]:
```

5. **Select the packages you want to install:**

   ■ Press Return or type **all** to accept the default and install all packages (if neither the GigaSwfit Ethernet Driver nor VLAN Utility packages are installed).
   ■ Type the numbers of all but the optional packages if you prefer not to install them.

   See TABLE 3-14 to identify the optional packages.

6. **If the Packages are already installed, verify the patch version.**

   See the following section.

Verifying Patches

Verify whether your system has the needed patches for your version of the Solaris Operating System to ensure that you do not need to add patches.

Note – If patches meet the requirements, nothing needs to be installed on the install system.

TABLE B-1 lists and describes the patch versions:

TABLE B-1   Patch Versions

Solaris	Patch Version	Description
8	111883-23	SunOS 5.8: Sun GigaSwift Ethernet 1.0 driver patch
	112119-04	SunOS 5.8: vlan driver patch*
9	112817-16	SunOS 5.9: Sun GigaSwift Ethernet 1.0 driver patch
	114600-02	SunOS 5.9: vlan driver patch*

* Only required if packages SUNWvld or SUNWvldx is installed.

▼ To Install Patches

1. **Determine the version of the Solaris operating system your system is running:**

```
% cat /etc/release
Solaris 9 8/03 s9s_u4wos_08a SPARC
Copyright 2003 Sun Microsystems, Inc.  All Rights Reserved.
                    Use is subject to license terms.
                           Assembled 13 June 2003
```

2. **Verify patch versions for the GigaSwift Etnernet driver:**
   - To verify patches for Solaris 8 Operating Systems, type the following:

```
# showrev -p | grep 111883
```

   The patch version should be -23 or greater.
   - To verify patches for Solaris 9 Operating Systems, type the following:

```
# showrev -p | grep 112817
```

   The patch version should be -16 or greater.

3. **If the patch version is not correct, install the correct patch:**

```
# cd /cdrom/cdrom0
# patchadd DualGigabitEthernet/OS/Patches/patch_id
```

Where:

OS = Solaris_8, or Solaris_9

patch_id = 111883-23 for Solaris 8, or 112817-16 for Solaris 9.

For example, using the information shown in Step 1, you would type the following:

```
# /usr/sbin/patchadd DualGigabitEthernet/Solaris_9/Patches/112817-16
Checking installed patches...
Executing prepatch script...


Verifying sufficient filesystem capacity (dry run method)...
Installing patch packages...

Patch number 112817-16 has been successfully installed.
See /var/sadm/patch/112817-16/log for details

Patch packages installed:
SUNWcea
SUNWceax
SUNWced
SUNWcedu
SUNWcedx
```

▼ To Install a Sun VLAN Utility Patch

Note – The VLAN Utility is only supported in Solaris 8 and Solaris 9 software.

1. **Determine the version of the Solaris operating system your system is running:**

```
% cat /etc/release
Solaris 9 8/03 s9s_u4wos_08a SPARC
Copyright 2003 Sun Microsystems, Inc.  All Rights Reserved.
                        Use is subject to license terms.
                            Assembled 13 June 2003
```

2. **Verify patch versions for the VLAN utility software:**

■ To verify patches for Solaris 8 Operating Systems, type the following:

```
# showrev -p | grep 112119
```

The patch version should be -04 or greater.

■ To verify patches for Solaris 9 Operating Systems, type the following:

```
# showrev -p | grep 114600
```

Verify that the patch version is -02 or higher.

3. **If the patch versions are not correct, install the correct patch:**

```
# cd /cdrom/cdrom0
# patchadd DualGigabitEthernet/OS/Patches/patch_id
```

Where:

OS = Solaris_8 or Solaris_9

patch_id = 112119-04 for Solaris 8, or 114600-02 for Solaris 9.

For example, using the information shown in Step 1, you would type the following:

```
# /usr/sbin/patchadd DualGigabitEthernet/Solaris_9/Patches/114600-02
Checking installed patches...
Executing prepatch script...


Verifying sufficient filesystem capacity (dry run method)...
Installing patch packages...

Patch number 114600-02 has been successfully installed.
See /var/sadm/patch/114600-02/log for details

Patch packages installed:
SUNWvld
SUNWvldu
SUNWvldx
```

Interface Signals

This appendix provides information for both the Ethernet and SCSI interface signals. It includes the following sections:

- "PCI Adapter Connectors" on page 67
- "Single-End SCSI-2 Connector Signals" on page 68
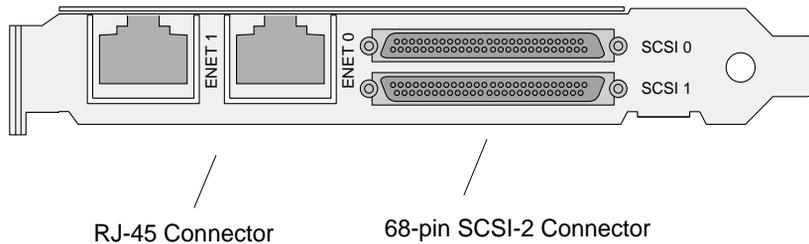- "RJ-45 Connector Signals" on page 69

PCI Adapter Connectors

FIGURE C-1   Sun Dual Gigabit Ethernet and Dual SCSI/P Adapter Backplate

Single-End SCSI-2 Connector Signals

TABLE C-1   Single-End SCSI-2 Connector Signals

Pin	Signal	Pin	Signal
1	SCSI_GND	35	SCSI_D<12>_
2	SCSI_GND	36	SCSI_D<13>_
3	SCSI_GND	37	SCSI_D<14>_
4	SCSI_GND	38	SCSI_D<15>_
5	SCSI_GND	39	SCSI_DATA_PARITY1_
6	SCSI_GND	40	SCSI_D<0>_
7	SCSI_GND	41	SCSI_D<1>_
8	SCSI_GND	42	SCSI_D<2>_
9	SCSI_GND	43	SCSI_D<3>_
10	SCSI_GND	44	SCSI_D<4>_
11	SCSI_GND	45	SCSI_D<5>_
12	SCSI_GND	46	SCSI_D<6>_
13	SCSI_GND	47	SCSI_D<7>_
14	SCSI_GND	48	SCSI_DATA_PARITY0_
15	SCSI_GND	49	SCSI_GND
16	SCSI_GND	50	SCSI_GND
17	SCSI_TERMPOWER	51	SCSI_TERMPOWER
18	SCSI_TERMPOWER	52	SCSI_TERMPOWER
19	OPEN	53	OPEN
20	SCSI_GND	54	SCSI_GND
21	SCSI_GND	55	SCSI_ATN_
22	SCSI_GND	56	SCSI_GND
23	SCSI_GND	57	SCSI_BSY_
24	SCSI_GND	58	SCSI_ACK_
25	SCSI_GND	59	SCSI_RST_
26	SCSI_GND	60	SCSI_MSG_

TABLE C-1   Single-End SCSI-2 Connector Signals *(Continued)*

Pin	Signal	Pin	Signal
27	SCSI_GND	61	SCSI_SEL_
28	SCSI_GND	62	SCSI_CD_
29	SCSI_GND	63	SCSI_REQ_
30	SCSI_GND	64	SCSI_IO_
31	SCSI_GND	65	SCSI_D<8>_
32	SCSI_GND	66	SCSI_D<9>_
33	SCSI_GND	67	SCSI_D<10>_
34	SCSI_GND	68	SCSI_D<11>_

RJ-45 Connector Signals

TABLE C-2   RJ-45 Connector Signals

Pin	Signal
1	Transmit+
2	Transmit-
3	Receive+
4	No Connection
5	No Connection
6	Receive-
7	No Connection
8	No Connection

APPENDIX **D**

Diagnostic Software and Troubleshooting Issues

This appendix provides an overview of the SunVTS diagnostic application and instructions for testing the adapter using the onboard FCode self-test. There is also a section outlining some common troubleshooting issues. This appendix contains the following sections:

- "SunVTS Diagnostic Software" on page 71
- "Using the OpenBoot PROM FCode Self-Test" on page 73

SunVTS Diagnostic Software

The SunVTS software executes multiple diagnostic hardware tests from a single user interface and is used to verify the configuration and functionality of most hardware controllers and devices. The SunVTS software operates primarily from a graphical user interface, enabling test parameters to be set quickly and easily while a diagnostic test operation is being performed.

The SunVTS `nettest` diagnostic can be used to test all of the networking interfaces on the system, including the interfaces on the adapter.

To use the `nettest` diagnostic, you must have the SunVTS software installed on your system. Refer to the *Solaris on Sun Hardware Platform Guide* for SunVTS installation instructions.

71

Refer to the SunVTS documentation (listed in TABLE D-1) for instructions on how to run and monitor the nettest diagnostic. These SunVTS documents are available on the *Solaris on Sun Hardware Collection*, which is provided on the Solaris Supplement CD for the Solaris release on your system.

TABLE D-1   SunVTS Documentation

Title	Description
SunVTS User's Guide	Describes the SunVTS diagnostic environment
SunVTS Test Reference Manual	Describes each SunVTS test (including the nettest) and describes the various test options and command-line arguments
SunVTS Quick Reference	Provides an overview of the user interface

Using the SunVTS netlbtest

You must have the Ethernet card and the device driver installed, a loopback connector in place, and Intervention mode enabled before running netlbtest. netlbtest cannot run if the network interface is connected to a live network, and requires that the Ethernet device be configured offline before running the test. Use the ifconfig(1M) command to bring the Ethernet device down before running netlbtest.

▼ To Use the netlbtest

1. **Ensure that the SunVTS software and the** ce **driver are installed on your system.**

2. **Plug in a loopback cable.**

3. **Unplumb the interface from the system, using the** ifconfig **command**:

```
# ifconfig ceinstance down
# ifconfig ceinstance unplumb
```

Where *instance* is the instance numbr of the interface.

Refer to SunVTS documentation for instructions on how to run netlbtest.

Using the OpenBoot PROM FCode Self-Test

The Sun Dual Gigabit Ethernet and Dual SCSI/P adapter's self-test verifies the correct operation of the adapter and identifies any hardware problems with the adapter. The self-test is a suite of tests that reside in the adapter's FCode PROM.

You can only invoke the FCode self-test diagnostics using the OpenBoot PROM `test` or `test-all` commands. In order to access the OpenBoot PROM, you must shut down and halt your system to display the `ok` prompt.

You can use the adapter's self-test any time you want to determine the status of the hardware. If you encounter an error while running the diagnostic test, the self-test displays appropriate error messages.

Note – Although the self-test does not require the adapter's interface to be connected to an active network, you will see failing error messages if it is not connected.

▼ To Run the FCode Self-Test Diagnostic

For more information about the OpenBoot commands (including `test`) in this section, refer to the *OpenBoot Command Reference* manual.

1. **Before shutting down the operating environment and halting the system, make sure that all significant application and network activity on the server has stopped.**

2. **Follow the appropriate procedures, as documented in the system's service manual, to shut down and halt the system.**

   Refer to the system's documentation for the complete shutdown procedure.

   After shutting down the system, you should see the OpenBoot PROMs `ok` prompt on the system console.

3. **At the `ok` prompt, set the `auto-boot?` configuration variable to false.**

   ```
   ok setenv auto-boot? false
   ```

4. **Set the** `diag-switch?` **parameter to true in order to see the diagnostic power-on self-test (POST) messages.**

```
ok setenv diag-switch? true
```

5. **Reset the system to activate these parameter changes.**

```
ok reset-all
```

6. **Use the** `show-nets` **command to display the list of network devices on the system.**

Among the list of devices, you should see the two interfaces specific to the Sun Dual Gigabit Ethernet and Dual SCSI/P adapter you want to test, as shown in the following example.

```
ok show-nets
a) /pci@1f,700000/network@2
b) /pci@1e,600000/pci@4/network@1
c) /pci@1e,600000/pci@4/network@0
q) NO SELECTION
Enter Selection, q to quit:
```

Note – The preceding device paths are examples. The actual device paths on your system may be different from the examples shown here.

7. **Use the** `test` **command, and the device paths found in Step 6, to test the adapter when you use the FCode self-test.**

```
ok test device_path
```

Refer to the appropriate *OpenBoot Command Reference Manual* for more information on the `test` command.

The following tests are run when you execute the `test` command:

- MAC internal loopback
- 1000 Mbps PHY loopback
- 100 Mbps PHY loopback
- 10 Mbps PHY loopback

Note – To test all of the interfaces of the adapter, you will need to execute the `test` command separately for each `SUNW,ce` interface.

If all of the tests pass, you will see these messages:

```
ok test pci@1e,600000/pci@4/network@1
Testing pci@1e,600000/pci@4/network@1
Register tests: passed
Internal loopback test: passed
pci@1e,600000/pci@4/network@1: 1000 Mbps full duplex link up
ok
```

If the adapter's interface is not connected, you will see the following error message:

```
ok test pci@1e,600000/pci@4/network@1
Testing /pci@1e,600000/pci@4/network@1
Register tests: passed
Internal loopback test: passed
/pci@8,600000/pci@1/network@0: Timed out waiting for Auto-
Negotation to co
mplete
pci@1e,600000/pci@4/network@1: Cannot establish link via Auto-
Negotation
Please check cable and/or connection
pci@1e,600000/pci@4/network@1: link down
ok
```

Note – If the `test` command fails, verify that the adapter hardware is installed correctly and the Ethernet device is connected to an active network. If necessary, replace the adapter or contact your service provider for support.

8. **After testing the adapter, type the following commands to return the OpenBoot PROM to the standard operating mode.**

```
ok setenv diag-switch? false
```

9. **At the** ok **prompt, reset the** auto-boot? **configuration variable to true.**

```
ok setenv auto-boot? true
```

10. Reset and reboot the system.

```
ok reset-all
```

Refer to the system's documentation for the correct procedures for bringing up the system.

Index