



C120-E199-04EN

PCI Hot Plug User's Guide

2.5.1



FUJITSU



Preface

Purpose

This manual describes the PCI Hot Plug feature, which enables you to logically configure and disconnect PCI cards without shutting down the operating system.

Intended Readers

This manual is intended for the following readers:

- System software developers and test engineers
- System support engineers
- System administrators who introduce and operate this software
- Technicians who maintain system hardware

Organization

This manual is organized as follows:

- “Chapter 1 Overview of PCI Hot Plug” is an introduction to the features and applications of PCI Hot Plug.
- “Chapter 2 PCI Hot Plug Administration” provides the technical background of PCI Hot Plug. The administration issues are discussed
- “Chapter 3 Operation flow of PCI Hot Plug” describes the operation flow of PCI Hot Plug.
- “Chapter 4 PCI Hot Plug command line interface” describes the command line interface of PCI Hot Plug.
- “Chapter 5 Messages and PCI Hot Plug Error Conditions” provides the meaning of messages displayed by this software. It also describes what to do when you get error messages.

Related manuals

- PCI Hot Plug User’s Guide I/O device edition
- Dynamic Reconfiguration User’s Guide (C120-E115)
- Partition Operation Guide (C120-E087)
- Dynamic Reconfiguration Architecture Guide (C120-E114)
- System Administration Guide: Solaris Containers -- Resource Management and Solaris Zones

Notation Used

The following shows the notation conventions used in this manual.

- The titles of chapters are enclosed in parentheses ("").
Example: See "Chapter1 Overview of PCI Hot Plug"
- Commands and other input use the following prompts:
C shell prompt:
prompt%
Bourne and korn shell prompt:
prompt\$
Super user prompt:
#
- Commands entered by the user is shown in bold:
cfgadm -c configure pcipsy2:R0B00-PCI#slot01
- Key combinations are represented, for example, by CTL-C, which means to simultaneously press the

Control and C key.

Trademarks

Sun, Sun Microsystems, the Sun Logo, Solaris and all Solaris based marks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries, and are used under license. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products with the SPARC trademark are based on architecture developed by Sun Microsystems.

All other products or company names mentioned in this document are claimed as trademark and trade names by their respective companies.

FUJITSU LIMITED

July 2005

Notes

<p>This manual may not be copied by any means without the express written permission of FUJITSU LIMITED. FUJITSU LIMITED reserves the right to make changes to this manual without prior notice.</p>
--

All Rights Reserved, Copyright (C) FUJITSU LIMITED 2005

Chapter 1 Overview of PCI Hot Plug

This chapter describes the basic functions of PCI Hot Plug. The PCI Hot Plug command interface **cfgadm (cfgadm_pci(1M))**, pushbutton interface and other system modules like I/O Multipathing will be introduced.

1.1 Feature Overview

PCI Hot Plug allows the user to physically remove or insert PCI cards while the operating system is still running. The user can modify the configuration of the system without shutting down the operating system.

The typical applications are:

- Removal of PCI cards with faulty I/O device/controllers while the system is running.
- Addition of new PCI cards to expand the I/O capacity of the system while the system is running.

PCI Hot Plug can be used with hardware that offers Extended Partitioning (XPAR). Refer to “Partition Operation Guide” or “Dynamic Reconfiguration Architecture Guide”.

The following models offer XPAR.

- PRIMEPOWER900/1500/2500

1.2 PCI Hot Plug Requirements

This section provides an overview of the PCI Hot Plug requirements.

1.2.1 Hardware

PCI Hot Plug is supported only on PRIMEPOWER450(Rackmount Type(7U))/900/1500 (with SPARC64 V)/2500/ HPC2500 platforms.

In this manual, descriptions about PRIMEPOWER1500 are only for the SPARC64 V models.

There are the following features on each PCI slot.

PCI Hot Plug pushbutton

A pushbutton for PCI Hot Plug operation is located on each PCI slot.

LED

The following three LEDs show the status of each PCI slot.

List of LEDs' status

LED	READY	ONLINE	ALARM	
Color	Green	Green	Orange	
Meaning	Status of	Power of slot	PCI card	Hardware
	On	On	Configured	Alarm
	Blink		In transition	Used to check the location
	Off	Off	Disconnected	Normal

1.2.2 Software

Either of the following software and patches must be installed.

- 64bit Solaris™ 8 Operating System 2/02 or later version, ESF2.3 or later version and the following Solaris 8 OS Patches
 - 108528-19 or later

- 110460-26 or later
 - 110842-11 or later
 - 110900-07 or later
 - 109885-09 or later
 - 64bit Solaris™ 9 Operating System 4/03 or later version, ESF 2.3 or later version, the following Solaris 9 OS Patch
 - 113068-04 or later
 - 64bit Solaris™ 10 Operating System and ESF 2.5 or later version
- Also the following device driver must be installed.
- Device driver supporting PCI Hot Plug
- Please refer to “2.1.1 Requirements and notes of PCI Hot Plug operations” for detail.

1.3 Command line Interface

This section describes basic PCI Hot Plug operations using the command line interface **cfgadm** (**cfgadm_pci(1M)**). The detailed reference of **cfgadm** (**cfgadm_pci(1M)**) can be found in “Chapter 4 PCI Hot Plug command line interface”.

1.3.1 Display status of PCI cards

Use the following command to display the status of PCI slots.

```
cfgadm [ap_id...|-s “select=class(pci)”]
```

There are three formats of *ap_id* according to Solaris™ Operating System version.

- a. **Solaris 8 OS and Solaris 9 OS**
 - **pcipsyX:CYM0Z-PCI#slotWW (slots in cabinet)**
WW represents physical slot number, *X* represents PCI Bus nexus driver instance number, *Y* represents cabinet number(0,1) and *Z* represents internal number of system boards(0,1,2,3).
 - **pcipsyX:RYB0Z-PCI#slotWW (slots in PCI/ disk box)**
WW represents physical slot number, *X* represents PCI Bus nexus driver instance number, *Y* represents I/O cabinet number (0,1,2,3) and *Z* represents PCI/ disk box number (0,1,2,3,4,5,6,7).
 - **pcipsyX:PCI#slotWW**
WW represents physical slot number, *X* represents PCI Bus nexus driver instance number.
- b. **Solaris 10 OS**
 - **CYM0Z-PCI#WW (slots in cabinet)**
WW represents physical slot number, *Y* represents cabinet number(0,1) and *Z* represents internal number of system boards(0,1,2,3).
 - **RYB0Z-PCI#WW (slots in PCI/ disk box)**
WW represents physical slot number, *Y* represents I/O cabinet number (0,1,2,3) and *Z* represents PCI/ disk box number (0,1,2,3,4,5,6,7).
 - **PCI#WW**
WW represents physical slot number.

To show status of PCI cards only, issue the following command:

```
cfgadm -s “select=class(pci)”
```

1.3.2 Configure PCI cards

Use the following command to perform the PCI Hot Plug configure operation.

```
cfgadm -c configure ap_id
```

The formats of *ap_id* are same as “1.3.1 Display status of PCI cards”.

This command configures a PCI card to the operating system in a partition. The I/O resources on the PCI slot are configured and made available to the Solaris OS.

1.3.3 Remove PCI cards

Use the following command to perform the PCI Hot Plug disconnect operation.

```
cfgadm -c disconnect ap_id
```

The formats of *ap_id* are same as “1.3.1 Display status of PCI cards”. This command disconnects a PCI card from a partition and removes all the I/O resources from the operating system.

1.3.4 Disable pushbutton interface

Use the following command to disable pushbutton interface for a PCI slot specified as *ap_id*.

```
cfgadm -x disable_autoconfig ap_id
```

Please refer to “1.4 Pushbutton Interface” for details of pushbutton interface.

Use the following command to enable pushbutton interface for a PCI slot. By default, pushbutton interface is enabled.

```
cfgadm -x enable_autoconfig ap_id
```

1.3.5 Display LED status

Use the following command to print the LED status of the PCI slot with *ap_id*.

```
cfgadm -x led ap_id
```

fault LED corresponds to ALERM LED and active LED corresponds to ONLINE LED.
power LED and attn LED are not supported and the statuses are shown as “unknown”.

1.4 Pushbutton Interface

Instead of “1.3 Command line Interface”, pushbuttons on slots are available for basic PCI Hot Plug operations (Configure and disconnect PCI cards).

1.5 I/O Multipathing

This section explains how I/O Multipathing can be used in conjunction with PCI Hot Plug to provide high availability I/O.

1.5.1 Hardware

I/O Multipathing requires a special device that supports multiple I/O channels. Please refer to each multipath device manual for details.

1.5.2 Software

I/O Multipathing is a framework for high reliability I/O. It provides Path Redundant configuration and Medium Redundant configuration. The Path Redundant configuration enhances reliability by providing two or more paths for a device, and increases throughput whenever a device can be accessed simultaneously with two or more paths. The Medium Redundant configuration enhances reliability by providing two or more units (media) with the same connection path. In the case of a disk, this function is generally called software mirroring.

1.5.3 I/O Multipath with PCI Hot Plug

The Path Redundant capability of I/O Multipath can be utilized to maintain uninterrupted access to the media in case of a PCI card disconnection. An I/O device can be configured such that it can be accessed from two different PCI cards. Even if one of the PCI cards is disconnected, the I/O device remains accessible. However, I/O performance may be degraded because the number of redundant paths is reduced. Without I/O Multipath, all I/O devices on the PCI card must be deactivated before the PCI card is disconnected.

For more information regarding PCI Hot Plug and I/O device, please refer to “PCI Hot Plug User’s Guide I/O Device edition”.

Chapter 2 PCI Hot Plug Administration

2.1 System Components

The system resources that are configured or disconnected during the PCI Hot Plug process are PCI cards. To perform PCI Hot Plug correctly, several conditions must be met. This section provides PCI Hot Plug requirements and notes.

2.1.1 Requirements and notes of PCI Hot Plug operations

2.1.1.1 Common requirements and notes of PCI Hot Plug operations

- System's run level should be multi-user to operate PCI Hot Plug.
- To insert or remove PCI cards, the READY LED must be off. If it is on, the operation may cause hardware failures. The Solaris OS device driver architecture implicitly requires that all drivers support dynamic addition of a new device instance. Therefore all drivers should support the PCI Hot Plug configure operation. Refer to "1.2.1 Hardware" for more details regarding LED.
- The OBP (Open Boot Prom) device tree is not updated when a PCI Hot Plug configure/disconnect operation is performed.
- When the pushbutton interface for a PCI slot is disabled, the conditions for each PCI Hot Plug operation may vary.

Please refer to the following table for combinations of settings and available operations.

Combinations of settings and available PCI Hot Plug operations

enable / disable pushbutton interface	PCI Hot Plug operations using pushbutton	operations using command line interface
<i>enable_autoconfig</i>	Y	Y
<i>disable_autoconfig</i>	N	Y

During PCI Hot Plug operations, Dynamic Reconfiguration (DR) operations must not be performed. DR operations might cause the system to go down. Please refer to "Dynamic Reconfiguration User's Guide" for details regarding DR.

- Please note the followings when using PCI Hot Plug in zones.
 - PCI Hot Plug operation is only possible in a global zone. PCI Hot Plug operation is not available in a non-global zone.
 - If physical devices are assigned to the non-global zone, PCI Hot Plug disconnect operation might be failed. When using PCI Hot Plug in zones, please consider about these limitations prior to the system deployment.

For more details about zones, please refer to "System Administration Guide: Solaris Containers --

2.1.1.2 PCI Hot Plug configure operation

- There are two types of PCI slots, one’s bus runs at 66MHz frequency and another runs at 33MHz. Usually the PCI card’s frequency should match with PCI slot’s frequency. When a 33MHz PCI card is inserted into a 66MHz bus, the following message is shown and the PCI card can’t be configured.

“66Mhz bus can't accept 33Mhz card”

However 66MHz PCI cards can be degraded to 33MHz. When a 66MHz PCI card is inserted into a 33MHz bus, the following message is shown and the PCI card runs at 33MHz.

“66Mhz capable card throttled back to 33Mhz”

The following table shows the available combinations of PCI cards and PCI slots.

Y: Yes/acceptable N: No/unacceptable		Frequency of slot	
		66MHz	33MHz
Frequency of PCI card	66MHz	Y	Y *1
	33MHz	N *2	Y

Notes

*1 When a 66MHz PCI card is inserted into a 33MHz bus, message is shown and the PCI card runs at 33MHz.

*2 The frequency of PCI slot is decided according to the frequency of PCI card in the slot at system boot time. The frequency will be kept until system reboot. Thus to configure a PCI card with different frequency from the current frequency of the slot, the following manual operation is needed. PCI Hot Plug is not available for this operation.

- 1) Shutdown the system
- 2) Insert the target PCI card
- 3) Boot the system

If a slot is empty at the system boot, the slot runs at 66MHz

- After the PCI Hot Plug configure operation, **devfsadm(1M)** automatically invokes the **devfsadm(1M)** command to reconfigure I/O device tree. Device path names newly added are written to the **/etc/path_to_inst** file. The same path names are also added to the **/devices** hierarchy, and links are created in the **/dev** directory. Disk controllers are numbered consecutively as the **devfsadm(1M)** command encounters them. On a newly inserted PCI card, disk controllers are assigned the next available lowest number by **devfsadm(1M)**. If a re-configured system board consists of the same I/O configuration as it was disconnected: the same I/O controllers and the same I/O devices installed at the same slots, the system keeps the same disk controller numbers as before. Otherwise, different numbers may be assigned. In such a case, re-configuring may require changes of **/etc/vfstab** and may affect some applications. The system administrator needs to manage I/O configuration with regard to this issue.
- During PCI Hot Plug configure operation with network devices, if **/etc/hostname.interface** or **/etc/hostname6.interface** exists, the Reconfiguration Coordination Manager (RCM) activates the network device automatically.

- Defective PCI cards can't be detected during PCI card configuration. Thus before adding PCI cards, be sure the PCI cards are working without failure.
- During PCI Hot Plug configure and disconnect operations, occasionally the following messages are printed. These messages are not an indication of any errors in PCI Hot Plug operations. Even though these messages are printed, the PCI Hot Plug operations should be done successfully. This bug is fixed with Solaris 8 OS patch 110837-05 and Solaris 9 OS patch 113571-03 as Sun BugID: 4722493.

```
interpreter[8306]: pci:map-out: XXXXXXXXX not mcookie!
interpreter[8306]: /usr/lib/efcode/sparcv9/interpreter: FC_RUN_PRIV: Priv
violation: map-out ( XXXX XXXXXXXXXXXXXXX )
interpreter[8306]: pci:map-out: failed
```

- During PCI Hot Plug configure operation without installing corresponding driver for the target device, the following message is printed. After configure operation, the device is configured for the system but the driver is not attached.
Please install the corresponding device driver to the system. If the problem still exists, please contact our customer service.

```
pcihp (DX): failed to attach one or more drivers for the card in the slot S
```

2.1.1.3 PCI Hot Plug Disconnect operation

- All device drivers running on the system must support PCI Hot Plug functions (PCI Hot Plug safe drivers). PCI Hot Plug safe drivers should support the following Solaris OS DDI/DKI entries.
 - DDI_DETACH:detach(9E)

If any *PCI Hot Plug unsafe* driver (not PCI Hot Plug safe) is loaded, the PCI Hot Plug disconnect procedure could fail. For detailed information about supported device drivers, refer to “PCI cards with support for PCI Hot Plug” in the “PCI Hot Plug User's Guide I/O device edition”.

- The device must not be in use or opened by any user or system process. The following description shows some examples of disk and network card. Please refer to “PCI Hot Plug User's Guide I/O device edition” for detail.

1. Disk

If any mounted disk exists on the outgoing PCI card, the PCI Hot Plug disconnect operation will fail. To avoid this problem, please unmount the disk as follows:

```
# umount pathname of the mount point
```

If any dump device or swap device exists on the outgoing PCI card, the PCI Hot Plug disconnect operation will fail.

To avoid this problem, please stop using the disk as follows:

```
# swap -d device name
# dumpadm -d device name
```

Disk space resources should be recalculated when the above configurations are changed.

2. Network card

If any activated network device exists on the outgoing PCI card, the PCI Hot Plug disconnect operation will fail.

Even though the RCM deactivates the network device automatically during PCI Hot Plug

disconnect operations, the network devices on the outgoing PCI card should be manually deactivated as follows:

```
# ifconfig interface down
# ifconfig interface unplumb
```

- Multi path I/O configurations are recommended to make I/O devices available for PCI Hot Plug operations. For example, network devices or disk devices can be set up in multi path I/O configurations. For details about I/O multipathing, refer to “PCI Hot Plug User’s Guide I/O device edition”.

The necessity for Multi path I/O configuration is explained below using the cases of disk and network cards.

1. Disk

It is not possible to unmount the root file system or any other file system that is vital to the user. To solve this problem, the administrator can use the I/O Multipathing feature to set up alternate paths to the crucial devices, or use other disk mirroring software to mirror the crucial file systems.

2. Network card

If the deactivated interface is the primary network interface for the system (the IP address of defined in the file `/etc/nodename`), all basic network applications like **ftp(1)**, **rsh(1)**, **rcp(1)**, **rlogin(1)**, **telnet(1)** will not function and NFS client and server operations are also affected. The user must use **I/O Multipathing** software to solve this problem. In addition, when the System Management Console (SMC) is used, functions of the SMC become unavailable if a network interface to the SMC is disconnected. This issue should be avoided in the same manner.

- The Path Redundant capability of I/O Multipath can be utilized to maintain uninterrupted access to the media in case of a PCI card disconnection. An I/O device can be configured such that it can be accessed from two different PCI cards. Even if one of the PCI cards is disconnected, the I/O device remains accessible. However, I/O performance may be degraded because the number of redundant paths is reduced.
- The following error message may appear in Solaris 9 OS despite of successful PCI Hot Plug disconnect operation. When the READY LED, ONLINE LED and ALARM LED are turned off at the PCI slot of the outgoing PCI device, or **cfgadm(cfgadm_pci(1M))** shows the Receptacle field of outgoing PCI device as ‘disconnected’, the operation is successful. This bug is fixed with Solaris 9 OS patch 112958-03 as Sun BugID: 4798214.

Message	cfgadm: Hardware specific failure
---------	--

2.2 Administration Issues

2.2.1 I/O nullification option

OBP disables all LAN ports and PCI cards installed (onboard serial ports are excluded) on a system board by changing this environment variable. Internal SCSI boards on PRIMEPOWER900/1500/2500/HPC2500 also can be disabled. When you change an OBP environment variable using the **eeeprom(1M)** command, the system must be rebooted to make the change effective.

If partitioning is available on the system, OBP environment variables exist on each partition. The setting for

one partition does not influence other partitions. Refer to “Dynamic Reconfiguration User’s Guide” for more details.

- For GP7000Fmodel 1000/2000 and PRIMEPOWER800/1000/2000
eeprom no-io-sb-cX=Y
- For PRIMEPOWER450/900/1500/2500/HPC2500
eeprom no-io-sb="XY"
- For PRIMEPOWER900/1500/2500 and XPAR environment
eeprom no-io-sb="XY-N"

where X is the cabinet number and Y is the slot number within the cabinet and N is offset number in the system board.

Example:

- For GP7000Fmodel 1000/2000 and PRIMEPOWER800/1000/2000
eeprom no-io-sb-c0
Check the I/O nullification setting on cabinet #0.
eeprom no-io-sb-c0=134
The I/O devices on boards numbered 1, 3 and 4 in cabinet #0 will be hidden.
- For PRIMEPOWER450/900/1500/2500/HPC2500
eeprom no-io-sb
Check the I/O nullification setting for all boards.
eeprom no-io-sb="01 03 04 10"
The I/O devices on boards numbered 1, 3, 4 in cabinet #0 and the ones on board numbered 0 in cabinet #1 will be hidden.

When PCI card is configured in a PCI slot with no-io-sb-cX or no-io-sb setting, the slot is powered on. PCI Hot Plug operations are not available for PCI cards. Removing a PCI card may cause a system failure.

Chapter 3 Operation flow of PCI Hot Plug

This chapter describes the following PCI Hot Plug operation flows

- Operation flow of PCI Hot Plug using the command line interface **cfgadm (cfgadm_pci(1M))**
- Operation flow of PCI Hot Plug using the pushbutton interface

3.1 Overview of PCI Hot Plug operations

The following operations are available for each interface.

3.1.1 Operation flow of PCI Hot Plug using command line interface

The following operations are available using **cfgadm (cfgadm_pci(1M))** command.

- To add PCI cards, refer to “3.2.1 Check the status of PCI cards” and “3.2.2 Add PCI cards”.
- To configure PCI cards, refer to “3.2.1 Check the status of PCI cards”, “3.2.2 Add PCI cards” and “3.2.3 Configure PCI cards”.
- To disconnect PCI cards, refer to “3.2.1 Check the status of PCI cards” and “3.2.4 Disconnect PCI cards”.
- To remove PCI cards, refer to “3.2.1 Check the status of PCI cards”, “3.2.4 Disconnect PCI cards” and “3.2.5 Remove PCI cards”.

3.1.2 Operation flow of PCI Hot Plug using pushbutton interface

The following operations are available using the pushbutton on each slot.

- Configure PCI cards (Refer to “3.2.3 Configure PCI cards”).
- Disconnect PCI cards (Refer to “3.2.4 Disconnect PCI cards”).
- For details of checking the status, adding and removing PCI cards, refer to “3.2.1 Check the status of PCI cards”, “3.2.2 Add PCI cards” and “3.2.5 Remove PCI cards”.

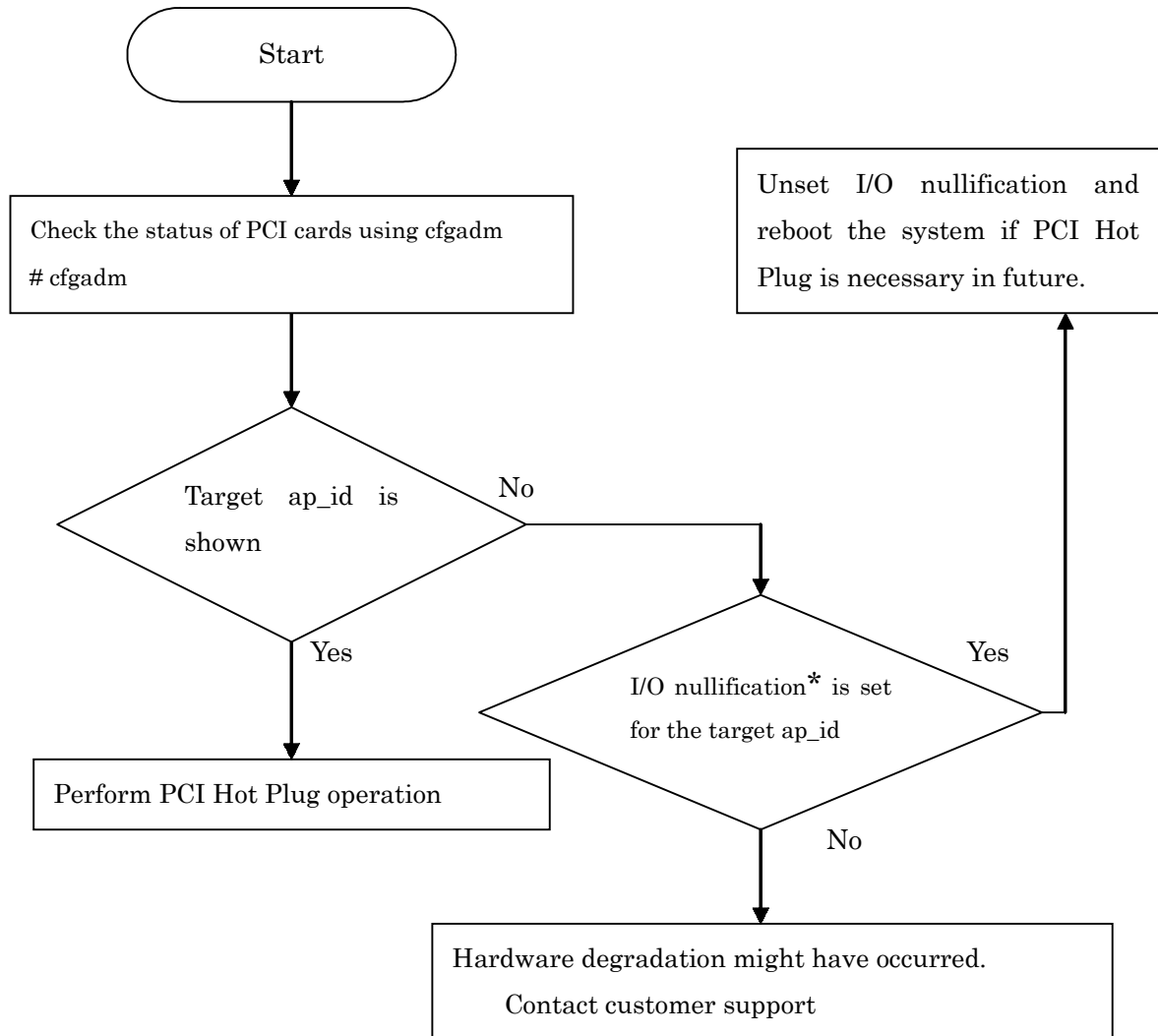
3.2 Detail operation flow of PCI Hot Plug

Please refer to "PCI Hot Plug User's Guide I/O device edition" for details of the operations.

3.2.1 Check the status of PCI cards

Follow this flow chart to check the status of PCI cards before add/remove operations.

Check the status of PCI cards

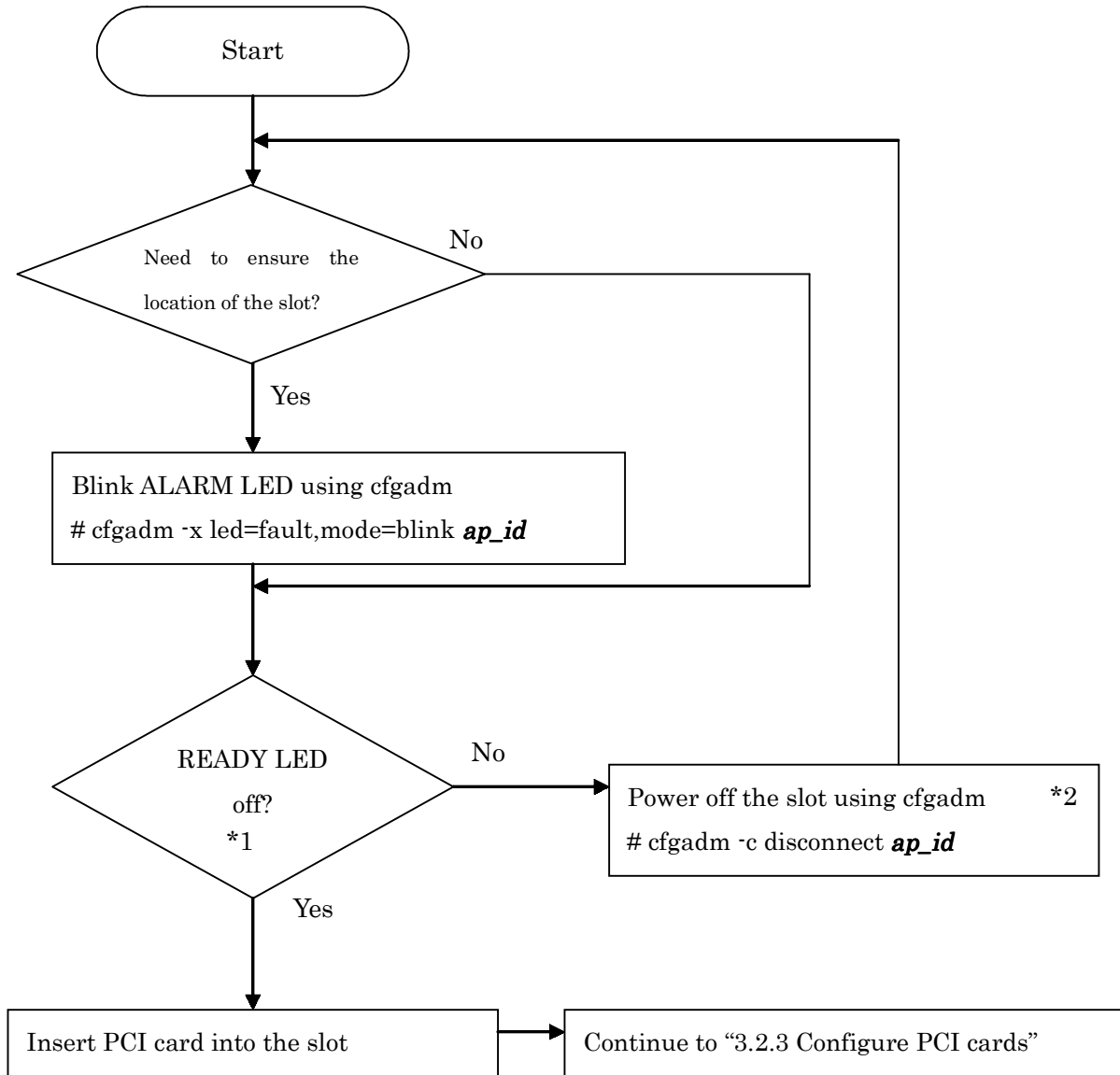


* For details of I/O nullification option, refer to “2.2.1 I/O nullification option”.

3.2.2 Add PCI cards

Follow this flow chart to add PCI cards.

Add PCI cards



*1

If the ALARM LED is on, there was an error during the PCI Hot Plug operation. Stop the operation on the slot and contact customer service.

*2

If the READY LED is still on after the power off operation, the hardware might not work correctly. Do not insert any cards into or remove any cards from the slot. The operation might cause a hardware failure and make the system unstable. In that case, please contact customer service.

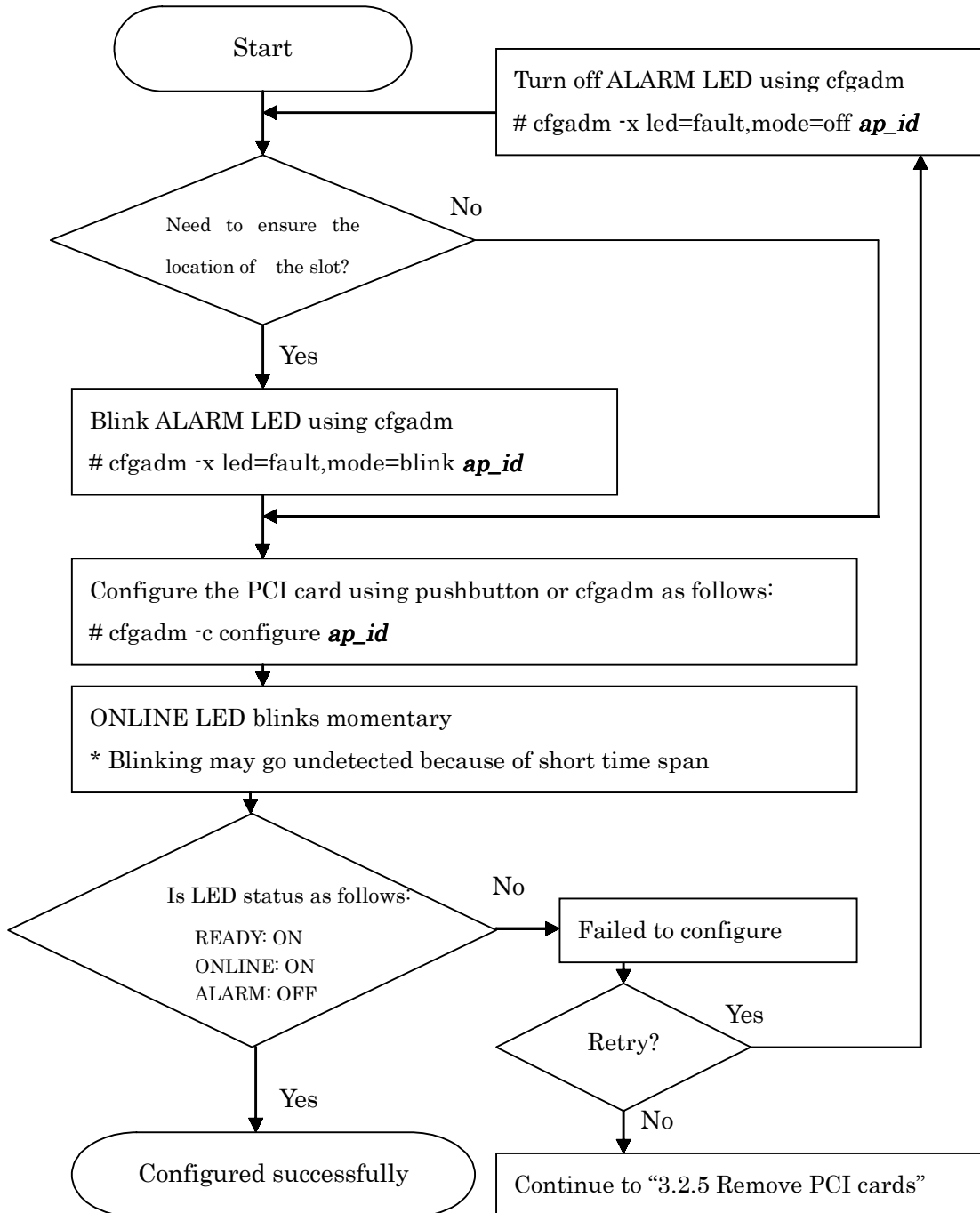
Other notes

If PCI card runs at a different frequency from PCI slot, refer to “2.1.1 Requirements and notes of PCI Hot Plug operations” and confirm if the PCI card is usable.

3.2.3 Configure PCI cards

Follow this flow chart to configure PCI cards.

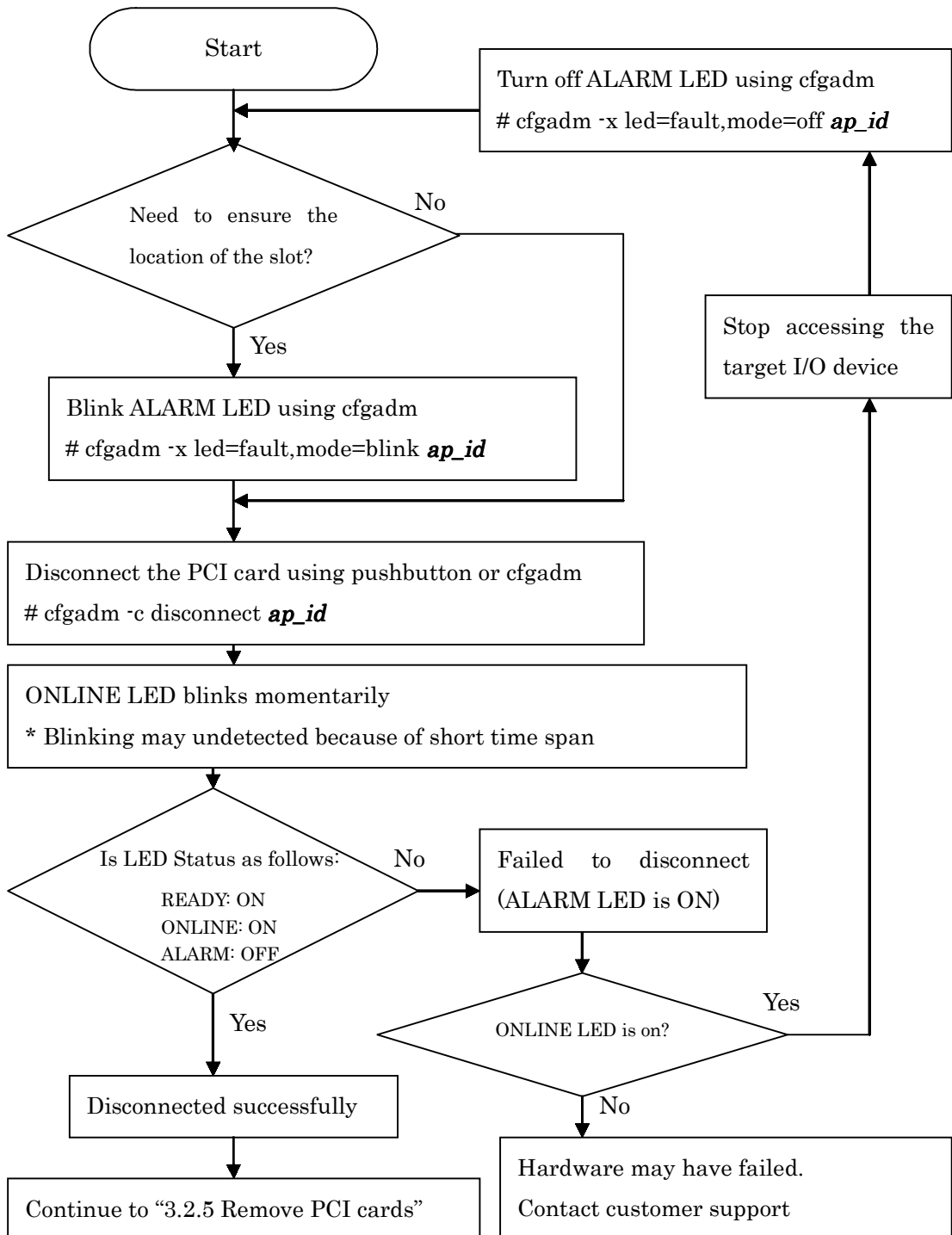
Configure PCI cards



3.2.4 Disconnect PCI cards

Follow this flow chart to disconnect PCI cards.

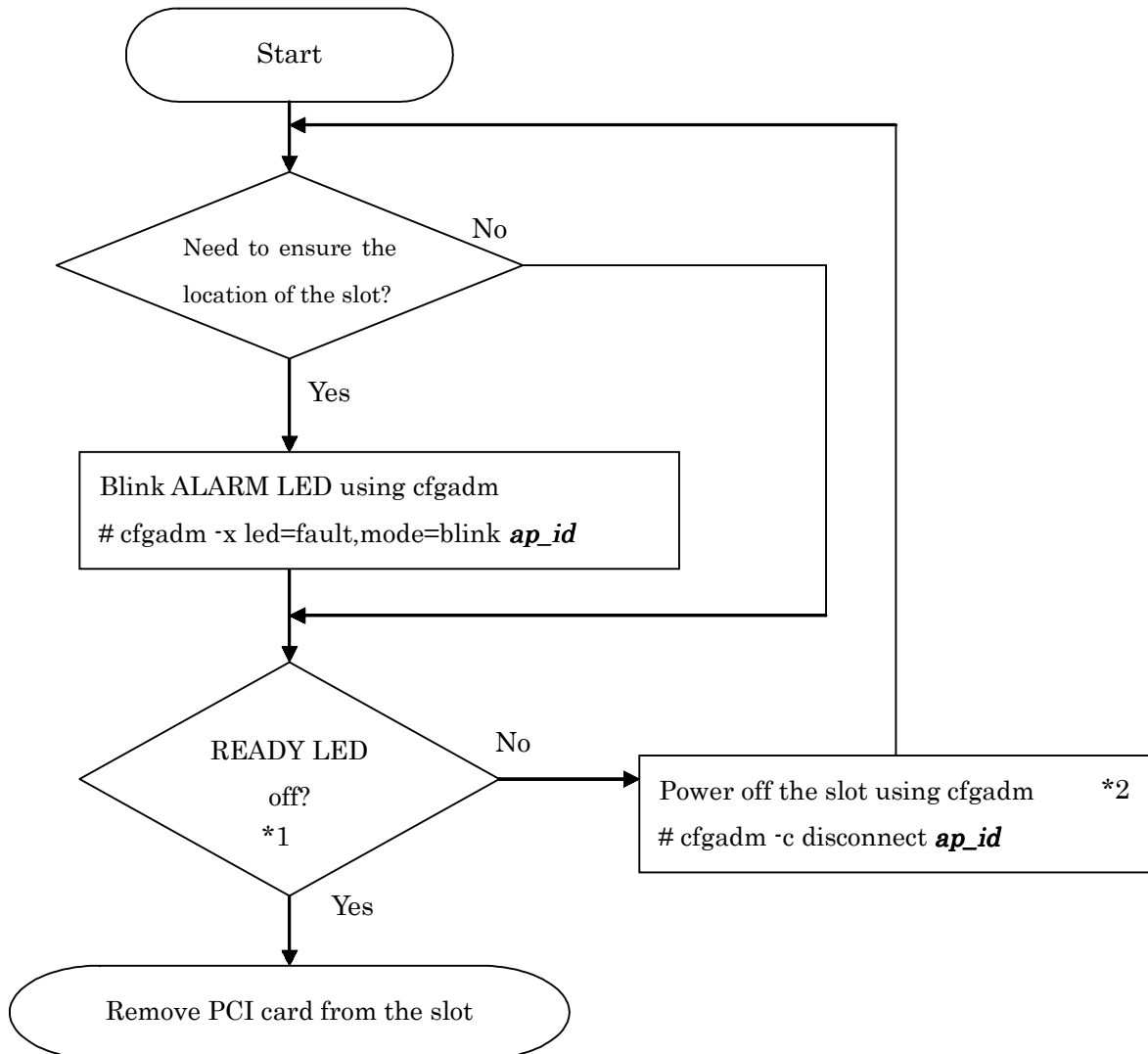
Disconnect PCI cards



3.2.5 Remove PCI cards

Follow this flow chart to remove PCI cards.

Remove PCI cards



*1

If the ALARM LED is on, there was an error during the PCI Hot Plug operation. No PCI Hot Plug operation can be continued with the slot. The ALARM LED is not usable for confirming the slot's location. In this case, please contact customer service. To remove PCI cards safely, the READY LED must be off.

*2

If the READY LED is still on after a power off operation, the hardware might not work correctly. Do not insert any cards into or remove any cards from the slot. The operation might cause hardware failure and make the system unstable. In this case, please contact customer service.

Chapter 4 PCI Hot Plug command line interface

This chapter describes the following PCI Hot Plug interface

- Command line interface **cfgadm (cfgadm_pci(1M))**, **prephp(1M)**, **postphp(1M)**, **inst2comp(1M)**
- PCI Hot Plug daemon **fjsvdrd (1M)**

4.1 cfgadm (cfgadm_pci(1M))

NAME

cfgadm – PCI Hot Plug operation

SYNOPSIS

```
/usr/sbin/cfgadm [-v] [ap_id | -s "select=class(pci)"]
/usr/sbin/cfgadm -c [disconnect | configure] ap_id [ap_id...]
/usr/sbin/cfgadm -x led[=fault,mode=[on | off | blink]] ap_id [ap_id...]
/usr/sbin/cfgadm -x [enable_autoconfig | disable_autoconfig] ap_id [ap_id...]
```

AVAILABILITY

- The user must be a super user to execute this command.
For requirements of this command, please refer to “1.2 PCI Hot Plug Requirements”.

DESCRIPTION

This command provides the PCI Hot Plug feature. The features include power on/off the PCI slots and configure/disconnect the PCI devices.

The command must be executed from the system partition where the PCI slots are located.

OPTIONS

The `cfgadm` command has the following options.

“`ap_id`” represents the PCI card’s logical ID displayed by `cfgadm`.

- Options for display slot status information
 - No option**
Lists information for all PCI slots.
Type, Receptacle, Occupant and Condition are displayed for each PCI slot.
 - Type**
Shows types of PCI slots. PCI slots displayed as type “*/hp” are PCI Hot Plug supported PCI slots.
 - Receptacle**
Shows connection status of PCI slots. There are three kinds of status, disconnected (power off), connected (power on) and empty (no card is inserted).
 - Occupant**
Shows occupancy status of PCI slots. There are two kinds of status, configured (PCI device is configured) and unconfigured (not yet configured).
 - Condition**
Shows failure status of PCI slots. There are four kinds of status, ok (normal), failed (had failure), unknown (not yet evaluated) and unusable (disabled).
- ap_id...**
Shows status of a PCI slot specified with `ap_id`.
There are three formats of *ap_id* according to the Solaris OS versions.

a. Solaris 8 OS and Solaris 9 OS

pcipsyX:CYM0Z-PCI#slotWW (slots in cabinet)

WW represents physical slot number, *X* represents PCI Bus nexus driver instance number, *Y* represents cabinet number(0,1) and *Z* represents internal number of system boards(0,1,2,3).

pcipsyX:RYB0Z-PCI#slotWW (slots in PCI/ disk box)

WW represents physical slot number, *X* represents PCI Bus nexus driver instance number, *Y* represents I/O cabinet number (0,1,2,3) and *Z* represents PCI/ disk box number (0,1,2,3,4,5,6,7).

pcipsyX:PCI#slotWW

WW represents physical slot number, *X* represents PCI Bus nexus driver instance number.

b. Solaris 10 OS

CYM0Z-PCI#WW (slots in cabinet)

WW represents physical slot number, *Y* represents cabinet number(0,1) and *Z* represents internal number of system boards(0,1,2,3).

RYB0Z-PCI#WW (slots in PCI/ disk box)

WW represents physical slot number, *Y* represents I/O cabinet number (0,1,2,3) and *Z* represents PCI/ disk box number (0,1,2,3,4,5,6,7).

PCI#WW

WW represents physical slot number.

-s “select=class(pci)”

Show status of PCI cards only.

-v

Execute in verbose mode.

● Options for PCI Hot Plug operations

-c function

The following functions are used for PCI Hot Plug operations.

disconnect

Unconfigure and power off a PCI slot specified with *ap_id*.

This option is specified to complete detaching the PCI slot.

configure

Power on and configure a PCI device specified with *ap_id*.

This option is specified to configure or hotswap the PCI slot.

-x led=[/*led*],mode=[*mode*]]

If “=[*led*],mode=[*mode*]” is not specified, prints the LED status of the PCI slot with *ap_id*. fault LED corresponds to ALARM LED and active LED corresponds to ONLINE LED. power LED and attn LED are not supported and the statuses are shown as “unknown”.

If “=[*led*],mode=[*mode*]” is specified, sets the mode of the specified LED on the PCI slot with the designated *ap_id*.

- The following arguments are used for *led*.

fault: ALARM LED

- The following arguments are used for *mode*.

on: Indicates LED light on.

blink: Indicates LED light blinking.

off: Indicates LED light off.

-x function

The following functions are used for PCI Hot Plug operations. PCI Hot Plug operation using each pushbutton can be enabled/disabled with this command. It is highly recommended to disable the PCI Hot Plug feature on the PCI cards to avoid miss operations when the PCI cards are used. When PCI cards need to be removed or exchanged, the functions can be enabled.

enable_autoconfig

Set autoconfig feature for a PCI device specified with *ap_id*. With this option, per-slot

pushbuttons can perform either of the following operations.

- Power on and configure the PCI device with `ap_id`.
- Unconfigure and power off the PCI device with `ap_id`.

This option is specified to prepare to use PCI Hot Plug feature with pushbutton on the PCI slot.

Default is enabled.

disable_autoconfig

Disables autoconfig feature for a PCI device specified with `ap_id`. The pushbutton of the PCI slot cannot perform power on/off or configure/unconfigure operations after this command is executed.

EXAMPLES

a. Solaris 8 OS and Solaris 9 OS

- Print status of all PCI slots.

```
# cfgadm
```

This command prints as follows:

Ap_Id	Type	Receptacle	Occupant	Condition
c0	scsi-bus	connected	configured	unknown
c1	scsi-bus	connected	configured	unknown
c10	scsi-bus	connected	configured	unknown
c11	scsi-bus	connected	unconfigured	unknown
c12	scsi-bus	connected	unconfigured	unknown
c13	scsi-bus	connected	unconfigured	unknown
c14	scsi-bus	connected	unconfigured	unknown
c15	scsi-bus	connected	unconfigured	unknown
c2	scsi-bus	connected	configured	unknown
c3	scsi-bus	connected	unconfigured	unknown
c4	scsi-bus	connected	unconfigured	unknown
c5	scsi-bus	connected	unconfigured	unknown
pcipsy24:ROB00-PCI#slot01	mult/hp	connected	configured	ok
pcipsy24:ROB00-PCI#slot02	unknown	empty	unconfigured	unknown
pcipsy25:ROB00-PCI#slot00	pci-pci/hp	connected	configured	ok
pcipsy26:ROB00-PCI#slot04	bridge/hp	connected	configured	ok
pcipsy26:ROB00-PCI#slot05	network/hp	connected	configured	unusable
pcipsy26:ROB00-PCI#slot06	unknown	empty	unconfigured	unknown
pcipsy27:ROB00-PCI#slot03	fibre/hp	connected	configured	ok
pcipsy28:ROB00-PCI#slot08	atm/hp	connected	configured	ok
pcipsy28:ROB00-PCI#slot09	unknown	empty	unconfigured	unusable
pcipsy29:ROB00-PCI#slot07	bridge/hp	connected	configured	ok

- View the detail information of PCI slot specified with `ap_id` `pcipsy2:ROB00-PCI#slot01`.

```
# cfgadm -v pcipsy2:ROB00-PCI#slot01
```

This command prints as follows:

Ap_Id	Type	Busy	Receptacle	Occupant	Condition	Information
When			Phys_Id			
pcipsy2:ROB00-PCI#slot01			connected	configured	ok	pci80b:2 ROB00-PCI#01
Jun 27 09:34	pci-pci/hp	n	/devices/pci@80,4000:ROB00-PCI#slot01			

- Power on and configure a PCI slot specified with `ap_id` `pcipsy2:ROB00-PCI#slot01`. The PCI slot is described as:

Physical slot number 01, PCI Bus nexus driver instance number 2, cabinet number is 0 and I/O box number is 00.

```
# cfgadm -c configure pcipsy2:R0B00-PCI#slot01
```

- Confirms the mode of the LEDs on a PCI device specified with ap_id pcipsy2:R0B00-PCI#slot01.
cfgadm -x led pcipsy2:R0B00-PCI#slot01
This command prints as follows:

Ap_Id	Led
pcipsy2:R0B00-PCI#slot01	fault=off, active=off

- Let the ALARM LED blink on the PCI slot with ap_id pcipsy2:R0B00-PCI#slot01. Then confirm the location of PCI slot to be disconnected.
cfgadm -x led=fault,mode=blink pcipsy2:R0B00-PCI#slot01
- Unconfigure and power off the PCI slot with ap_id pcipsy2:R0B00-PCI#slot01.
cfgadm -c disconnect pcipsy2:R0B00-PCI#slot01

b. Solaris 10 OS

- Print status of all PCI slots.

cfgadm

This command prints as follows:

Ap_Id	Type	Receptacle	Occupant	Condition
COM01-PCI#05	unknown	empty	unconfigured	unknown
COM01-PCI#06	unknown	empty	unconfigured	unknown
COM01-PCI#07	unknown	empty	unconfigured	unknown
ROB00-PCI#00	etherne/hp	connected	configured	ok
ROB00-PCI#01	unknown	empty	unconfigured	unknown
ROB00-PCI#02	unknown	empty	unconfigured	unknown
ROB00-PCI#03	unknown	empty	unconfigured	unknown
ROB00-PCI#04	unknown	empty	unconfigured	unknown
ROB00-PCI#05	unknown	empty	unconfigured	unknown
ROB00-PCI#06	unknown	empty	unconfigured	unknown
ROB00-PCI#07	unknown	empty	unconfigured	unknown
ROB00-PCI#08	unknown	empty	unconfigured	unknown
ROB00-PCI#09	unknown	empty	unconfigured	unknown
SB0		disconnected	unconfigured	unknown
SB1		disconnected	unconfigured	unknown
SB2		disconnected	unconfigured	unknown
SB3		disconnected	unconfigured	unknown
SB4		disconnected	unconfigured	unknown
SB5	System_Brd	connected	configured	ok
SB6		disconnected	unconfigured	unknown
SB7		disconnected	unconfigured	unknown
SB8		disconnected	unconfigured	unknown
SB9	System_Brd	connected	configured	ok
SB10		disconnected	unconfigured	unknown
SB11		disconnected	unconfigured	unknown
SB12		disconnected	unconfigured	unknown
SB13	System_Brd	connected	configured	ok
SB14		disconnected	unconfigured	unknown
SB15		disconnected	unconfigured	unknown
c0	scsi-bus	connected	configured	unknown
c1	scsi-bus	connected	configured	unknown
c2	scsi-bus	connected	configured	unknown
c3	scsi-bus	connected	unconfigured	unknown
c4	scsi-bus	connected	unconfigured	unknown
c5	scsi-bus	connected	unconfigured	unknown

- View the detail information of PCI slot specified with ap_id ROB00-PCI#01.

cfgadm -v ROB00-PCI#01

This command prints as follows:

Ap_Id	Type	Busy	Receptacle	Phys_Id	Occupant	Condition	Information
ROB00-PCI#01	etherne/hp	n	connected	/devices/pci@cc,2000:ROB00-PCI#00	configured	ok	pcicca:1 ROB00-PCI#01
Aug 25 14:30							

- Power on and configure a PCI slot specified with ap_id ROB00-PCI#01. The PCI slot is described as:
Physical slot number 01, cabinet number is 0 and I/O box number is 00.

cfgadm -c configure ROB00-PCI#01

- Confirms the mode of the LEDs on a PCI device specified with ap_id R0B00-PCI#01.
cfgadm -x led R0B00-PCI#01
This command prints as follows:

Ap_Id	Led
R0B00-PCI#01	power=unknown, fault=off, attn=unknown, active=on

- Let the ALARM LED blink on the PCI slot with ap_id R0B00-PCI#01. Then confirm the location of PCI slot to be disconnected.
cfgadm -x led=fault,mode=blink R0B00-PCI#01
- Unconfigure and power off the PCI slot with ap_id R0B00-PCI#01.
cfgadm -c disconnect R0B00-PCI#01

EXIT STATUS

This command returns the following values.

0: display information successfully

>0: An error occurred.

Please refer to console messages in “5.1 Command line Messages”

SEE ALSO

cfgadm(1M)

NOTES

- Please confirm the PCI card to be disconnected/connected is available for PCI Hot Plug operation using `cfgadm (cfgadm_pci(1M))` with no option.
- Before disconnecting a file related device, be sure that the I/O device is not busy. For more information, refer to “PCI Hot Plug User’s Guide I/O device edition”
- To insert or remove PCI cards, the READY LED must be off. If it is on, the operation may cause hardware failures.
- The connect and unconfigure functions for the `-c` option should not be used
- The `enable_slot` and `disable_slot` functions for the `-x` option are not used.
- If the status of a LED is changed using `-x led` option, the LED might not show the correct status. Blinking the ALARM LED to locate the target slot as follows should not have any problems.
cfgadm -x led=fault,mode=blink ap_id
- When a failure on a PCI card occurs and the condition of the slot is detected as ‘failed’, the slot is not usable. Do not attempt to operate PCI Hot Plug on a slot indicating a ‘failed’ condition.
- Printing status using `cfgadm (cfgadm_pci(1M))` might take long time when the device configuration process `devfsadm (1M)` is running. `devfsadm (1M)` is responsible for device reconfiguration at boot time or dynamic reconfiguration. While `devfsadm (1M)` updates the device information, the necessary information for `cfgadm` is locked by `devfsadm`. Therefore, `cfgadm (1M)` needs to wait until the device reconfiguration is finished.
If the system has many device modifications, or it is the first time to configure the device information, reconfiguration of device information takes long time. After the device configuration is finished, the `cfgadm(cfgadm_pci(1M))` commands will respond promptly.

EXAMPLE ERROR MESSAGES

This command sometimes fails because of syntax error or other reasons.

The followings are example error messages that are most likely to occur.

Error messages of usual PCI Hot Plug operations

- When an invalid ap_id is specified.

Message	cfgadm: Attachment point not found
Cause	Specified ap_id is invalid.
Remedy	Confirm the correct ap_id using cfgadm (cfgadm_pci(1M)) and try again.

- When specified state transition is not executable.

Message	cfgadm: Configuration operation invalid: invalid transition
Cause	Specified state transition is not executable.
Remedy	Confirm the available transition and try again.

- When specified arguments are invalid. (LED operations, etc.)

Message	cfgadm: Configuration operation invalid
Cause	Specified arguments are invalid.
Remedy	Confirm the valid arguments and try again.

- When configure operation is invoked for empty PCI slot.

Message	cfgadm: Hardware specific failure: configure failed
Cause	Configure operation is invoked for empty PCI slot. Note: This error occurs because the specified PCI Hot Plug operation is not executable. The error message “hardware specific failure” is different from the cause of the error.
Remedy	Confirm the correct ap_id and try again.

4.2 prephp(1M)

NAME

prephp(1M) – Pre PCI Hot Plug operation

SYNOPSIS

/usr/sbin/FJSVmadm/prephp

DESCRIPTION

Prephp stops the daemon that needs to be stopped before executing a PCI Hot Plug operation.

The function stopped by the prephp command is following.

- Hardware abnormality monitoring daemon

EXIT STATUS

This command returns the following values.

0: daemon was stopped successfully

>0: An error occurred.

If an error occurs, execute this command again.

NOTES

Please be sure to execute this command before disconnecting or configuring a PCI card.

The disconnect or configure command might fail when this command is not executed beforehand.

4.3 postphp(1M)

NAME

postphp(1M) – post PCI Hot Plug operation

SYNOPSIS

/usr/sbin/FJSMadm/postphp

DESCRIPTION

The postphp command does the following procedures which need to be executed after completing a PCI Hot Plug operation.

- Re-invoking the hardware abnormality monitoring daemon which is stopped by the prephp(1M) command.
- Reconfigure the hardware configuration database of the Machine Administration function.
- Re-invoking the picld(1M) daemon

EXIT STATUS

This command returns the following values.

0: procedure was executed successfully

>0: **An error occurred.**

If an error occurs, execute this command again.

NOTES

Please be sure to execute this command after disconnecting or configuring a PCI card.

The abnormality monitoring and notifying function does not work correctly when this command is not executed.

4.4 inst2comp(1M)

NAME

inst2comp – print physical slot name

SYNOPSIS

/usr/sbin/FJSMadm/inst2comp inst_name

DESCRIPTION

Inst2comp prints the ap_id of the PCI card specified by the instance name of the driver.

inst_name specifies the instance name of the adapter card driver.

EXAMPLE

Display the ap_id of the adapter that is controlled by the driver name 'glm2'.

```
example% /usr/sbin/FJSMadm/inst2comp glm2
pcipsy1:R0B01-PCI#slot01
```

EXIT STATUS

0

Information was displayed successfully

1

There is no device specified by inst_name.

<0

Another error occurred

4.5 fjsvdrd

NAME

fjsvdrd – PCI Hot Plug daemon

SYNOPSIS

/usr/platform/sun4us/lib/fjsvdrd

DESCRIPTION

This daemon runs at PCI Hot Plug environment. fjsvdrd wakes up at system boot. This daemon implements per-slot PCI Hot Plug pushbutton interface. PCI Hot Plug pushbutton interface is available when this daemon is running.

SEE ALSO

cfgadm(1M), **cfgadm_pci(1M)**

Chapter 5 Messages and PCI Hot Plug Error Conditions

5.1 Command line Messages

For command line messages printed by `cfgadm(cfgadm_pci(1M))`, please refer to “EXAMPLE ERROR MESSAGES” of “4.1 `cfgadm (cfgadm_pci(1M))`”.

Also for notes regarding command line messages, please refer to “2.1.1 Requirements and notes of PCI Hot Plug operations”.

5.2 Console Messages

For notes regarding command line messages, please refer to “2.1.1 Requirements and notes of PCI Hot Plug operations”.

5.2.1 Warning and Error Message List

The two kinds of messages are shown using the following Solaris OS.

- Solaris 8 OS: please refer to “5.2.1.1 Warning and Error Message List (Solaris 8 OS)”
- Solaris 9 OS: please refer to “5.2.1.2 Warning and Error Message List (Solaris 9 OS / Solaris 10 OS)”

5.2.1.1 Warning and Error Message List (Solaris 8 OS)

66Mhz bus can't accept 33Mhz card
66Mhz capable card throttled back to 33Mhz
Cannot Get Board Type..
Cannot change control register.
Cannot map ntbridge child #
Cannot set config space map for pci device <i>number Y</i>
Could not power-off AP_ID
Could not power-on AP_ID
DX: PCI device Y Busy
DX: PCI device Y Failed on Configure
DX: PCI device Y Failed on Unconfigure
DX: can't allocate pcihp structure
DX: failed to register the bus with HPS
DX: pcihp instance already initialized!
DX: pcihp instance is busy
Error configuring ntbridge child dev=Y
Failed to add conf reg for ntbridge child.
High level interrupt not supported

No Z bytes IO space window for S
No Z bytes memory window for S
Not supported this bus type
PCI Expansion ROM is not accessible
Unconfig Error ntbridge child dev=Y
Unconfigure: Failed to add conf reg prop for ntbridge child
bad format in slot-table
can't allocate memory for slot-table
couldn't find slot-table property
ddi_get_iblock_cookie FAILED
failed to add interrupt
fc_ioctl: fc_get_my_args: wrong state (#)
fc_ioctl: fc_get_parameters: wrong state (#)
fc_ioctl: fc_run_priv: results (from ops function) overflow
fc_ioctl: fc_run_priv: wrong state (#)
fc_ioctl: fc_validate: wrong state (#)
fc_mmap: Wrong state (#) for mmap
fc_read: Wrong state (#) for read
fc_rem_resource: Item not on list!
fc_rem_resource: NULL list head!
fc_remove_child: not found
fcode ops: fail service name #
fcode_timer: Timeout waiting for interpreter - Interpreter did not pick up request
fcode_timer: Timeout waiting for interpreter - Interpreter is executing request
fcpci: conf probe failed.l=#
fcpcii: conf probe failed.l=#
hot-plug disabled on AP_ID
hot-plug switch disabled AP_ID
hpc3130X: bus-status undefined
hpc3130X: can't allocate memory for bus-status
hpc3130X: ddi_create_minor_node failed for port_Y
hpc3130X: ddi_regs_map_setup failed.
hpcsvc: can't create event dispatch thread
invalid register offset value
ntbridge bus range invalid !(#,#)
ntbridge: IO resource information failure
ntbridge: Mem resource information failure
ntbridge: unconfigure failed
pci_fc_ops_free: unknown resource type #
pcicfg: Cannot map ntbridge child Y
pcicfg: Fail:cant load nontransparent bridgd driver..
pcicfg: cannot map config space, to get map type
pcicfg: cant unload ntbridge driver..children.
pcihp (DX): Card is still in configured state for pci dev Y
pcihp (DX): SLOT_CONFIGURE event ocurred for pci dev Y (slot S), Slot disabled for auto-configuration.
pcihp (DX): SLOT_CONFIGURE event re-ocurred for pci dev Y (slot S)
pcihp (DX): SLOT_UNCONFIGURE event for pci dev Y (slot S) ignored, Slot

disabled for auto-configuration.
pcihp (DX): SLOT_UNCONFIGURE event re-occured for pci dev Y (slot S)
pcihp (DX): card is CONFIGURED in the slot S (pci dev Y)
pcihp (DX): card is UNCONFIGURED in the slot S (pci dev Y)
pcihp (DX): card is inserted in the slot S (pci dev Y)
pcihp (DX): card is removed from the slot S
pcihp (DX): ddi_create_minor_node failed for pci dev Y
pcihp (DX): failed to attach one or more <i>drivers</i> for the card in the slot S
pcihp (DX): failed to configure the card in <i>the</i> slot S
pcihp (DX): install event handler failed for pci dev Y
pcihp (DX): latch is shut for the slot S
pcihp (DX): latch is open for the slot S
pcihp (DX): unknown event # for this slot S
pcihp_new_slot_state: unknown slot_state U
pfc_config_fetch: invalid config addr: #
pfc_config_store: invalid config addr: #
pfc_dma_cleanup: ddi_dma_free failed!
pfc_dma_map_out: ddi_dma_free failed!
pfc_fc_ops_free: DMA seen!
pfc_get_fcode: Unable to copy out fcode image
slot-table undefined
unitp not filled

5.2.1.2 Warning and Error Message List (Solaris 9 OS / Solaris 10 OS)

Cannot Get Board Type..
Cannot map ntbridge child #
Cannot set config space map for pci device <i>number Y</i>
DX: PCI device Y Busy
DX: PCI device Y Failed on Configure
DX: PCI device Y Failed on Unconfigure
DX: can't allocate pcihp structure
DX: failed to register the bus with HPS
DX: pcihp instance already initialized!
DX: pcihp instance is busy
Error configuring ntbridge child dev=Y
Failed to add conf reg for ntbridge child.
No Z bytes IO space window for S
No Z bytes memory window for S
PCI Expansion ROM is not accessible
Unable to allocate non relocatable IO Space #
Unconfig Error ntbridge child dev=Y
Unconfigure: Failed to add conf reg prop for ntbridge child
fc_ioctl: fc_get_my_args: wrong state (#)
fc_ioctl: fc_get_parameters: wrong state (#)
fc_ioctl: fc_run_priv: results (from ops function) overflow
fc_ioctl: fc_run_priv: wrong state (#)
fc_ioctl: fc_validate: wrong state (#)

fc_mmap: Wrong state (#) for mmap
fc_read: Wrong state (#) for read
fc_rem_resource: Item not on list!
fc_rem_resource: NULL list head!
fc_remove_child: not found
fcode ops: fail service name #
fcode_timer: Timeout waiting for interpreter - Interpreter did not pick up request
fcode_timer: Timeout waiting for interpreter - Interpreter is executing request
fcpci: conf probe failed.l=#
fcpcii: conf probe failed.l=#
fjsv: 66Mhz bus can't accept 33Mhz card
fjsv: 66Mhz capable card throttled back to 33Mhz
fjsv: Cannot change control register.
fjsv: Could not power-off AP_ID
fjsv: Could not power-on AP_ID
fjsv: High level interrupt not supported
fjsv: Not supported this bus type
fjsv: bad format in slot-table
fjsv: bus-status undefined
fjsv: can't allocate memory for bus-status
fjsv: can't allocate memory for slot-table
fjsv: couldn't find slot-table property
fjsv: ddi_get_iblock_cookie FAILED
fjsv: ddi_regs_map_setup failed.
fjsv: failed to add interrupt
fjsv: hot-plug disabled on AP_ID
fjsv: hot-plug switch disabled AP_ID
fjsv: invalid register offset value
fjsv: slot-table undefined
hpcsvc: can't create event dispatch thread
ntbridge bus range invalid !(#,#)
ntbridge: IO resource information failure
ntbridge: Mem resource information failure
ntbridge: unconfigure failed
pci_fc_ops_free: unknown resource type #
pcicfg: Cannot map ntbridge child Y
pcicfg: Fail:cant load nontransparent bridgd driver..
pcicfg: cannot map config space, to get map type
pcicfg: cant unload ntbridge driver..children.
pcihp (DX): Card is still in configured state for pci dev Y
pcihp (DX): SLOT_CONFIGURE event occurred for pci dev Y (slot S), Slot disabled for auto-configuration.
pcihp (DX): SLOT_CONFIGURE event re-occured for pci dev Y (slot S)
pcihp (DX): SLOT_UNCONFIGURE event for pci dev Y (slot S) ignored, Slot disabled for auto-configuration.
pcihp (DX): SLOT_UNCONFIGURE event re-occured for pci dev Y (slot S)
pcihp (DX): card is CONFIGURED in the slot S (pci dev Y)
pcihp (DX): card is UNCONFIGURED in the slot S (pci dev Y)

pcihp (DX): card is inserted in the slot <i>S</i> (pci dev <i>Y</i>)
pcihp (DX): card is removed from the slot <i>S</i>
pcihp (DX): ddi_create_minor_node failed for pci dev <i>Y</i>
pcihp (DX): failed to attach one or more <i>drivers</i> for the card in the slot <i>S</i>
pcihp (DX): failed to configure the card in <i>the</i> slot <i>S</i>
pcihp (DX): install event handler failed for pci dev <i>Y</i>
pcihp (DX): latch is shut for the slot <i>S</i>
pcihp (DX): latch is open for the slot <i>S</i>
pcihp (DX): unknown event # for this slot <i>S</i>
pcihp_new_slot_state: unknown slot_state <i>U</i>
pfc_config_fetch: invalid config addr: #
pfc_config_store: invalid config addr: #
pfc_dma_cleanup: ddi_dma_free failed!
pfc_dma_map_out: ddi_dma_free failed!
pfc_fc_ops_free: DMA seen!
pfc_get_fcode: Unable to copy out fcode image
unitp not filled

5.2.2 Message Explanation

This section explains the console messages printed by PCI Hot Plug driver.

5.2.2.1 Progress Messages

Message	pcihp (DX): card is inserted in the slot <i>S</i> (pci dev <i>Y</i>)
Explanation	PCI card is inserted. (<i>D</i> :Driver name, <i>X</i> :Instance number, <i>Y</i> :Device number, <i>S</i> :Slot name)

Message	pcihp (DX): SLOT_CONFIGURE event occurred for pci dev <i>Y</i> (slot <i>S</i>), Slot disabled for auto-configuration.
Explanation	Auto-configuration for PCI card is disabled. (<i>D</i> :Driver name, <i>X</i> :Instance number, <i>Y</i> :Device number, <i>S</i> :Slot name)

Message	pcihp (DX): card is CONFIGURED in the slot <i>S</i> (pci dev <i>Y</i>)
Explanation	PCI card is configured. (<i>D</i> :Driver name, <i>X</i> :Instance number, <i>Y</i> :Device number, <i>S</i> :Slot name)

Message	pcihp (DX): SLOT_UNCONFIGURE event for pci dev <i>Y</i> (slot <i>S</i>) ignored, Slot disabled for auto-configuration.
Explanation	Auto-configuration for PCI card is disabled. (<i>D</i> :Driver name, <i>X</i> :Instance number, <i>Y</i> :Device number, <i>S</i> :Slot name)

Message	pcihp (DX): card is UNCONFIGURED in the slot <i>S</i> (pci dev <i>Y</i>)
Explanation	PCI card is unconfigured. (<i>D</i> :Driver name, <i>X</i> :Instance number, <i>Y</i> :Device number, <i>S</i> :Slot name)

Message	pcihp (DX): card is removed from the slot S
Explanation	PCI card is removed. (<i>D</i> :Driver name, <i>X</i> :Instance number, <i>S</i> :Slot name)

Message	pcihp (DX): latch is shut for the slot S
Explanation	Latch is shut. (<i>D</i> :Driver name, <i>X</i> :Instance number, <i>S</i> :Slot name)

Message	pcihp (DX): latch is open for the slot S
Explanation	Latch is open. (<i>D</i> :Driver name, <i>X</i> :Instance number, <i>S</i> :Slot name)

Message	66Mhz capable card throttled back to 33Mhz
Explanation	66Mhz capable card is inserted into 33Mhz bus. The PCI card is throttled back to 33Mhz. To use the PCI card with 66Mhz, please insert the card into 66Mhz bus.

5.2.2.2 Error and Warning Messages

Message	fc_read: Wrong state (#) for read
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	fc_mmap: Wrong state (#) for mmap
Cause	There may be inconsistency in the system.
Remedy	Repeat the operation once again. If the problem still exists, please contact our customer service.

Message	fc_ioctl: fc_get_parameters: wrong state (#)
Cause	There may be inconsistency in the system.
Remedy	Repeat the operation once again. If the problem still exists, please contact our customer service.

Message	fc_ioctl: fc_get_my_args: wrong state (#)
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	fc_ioctl: fc_run_priv: wrong state (#)
Cause	There may be inconsistency in the system.
Remedy	Repeat the operation once again. If the problem still exists, please contact our customer service.

Message	fc_ioctl: fc_validate: wrong state (#)
Cause	There may be inconsistency in the system.

Remedy	Repeat the operation once again. If the problem still exists, please contact our customer service.
--------	--

Message	fc_ioctl: fc_run_priv: results (from ops function) overflow
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	pfc_fc_ops_free: DMA seen!
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	pci_fc_ops_free: unknown resource type #
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	pfc_dma_map_out: ddi_dma_free failed!
Cause	Failed to release memory resource.
Remedy	Please contact our customer service.

Message	pfc_dma_cleanup: ddi_dma_free failed!
Cause	Failed to release memory resource.
Remedy	Please contact our customer service.

Message	pfc_config_fetch: invalid config addr: #
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	fcpcii: conf probe failed.l=#
Cause	Failed to probe PCI Configuration Space.
Remedy	Repeat the operation once again. If the problem still exists, please contact our customer service.

Message	pfc_config_store: invalid config addr: #
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	fcpci: conf probe failed.l=#
Cause	Failed to probe PCI Configuration Space.
Remedy	Repeat the operation once again. If the problem still exists, please contact our customer service.

Message	pfc_get_fcode: Unable to copy out fcode image
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	Unable to allocate non relocatable IO Space #
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	fc_rem_resource: NULL list head!
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	fc_rem_resource: Item not on list!
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	fcode ops: fail service name #
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	fcode_timer: Timeout waiting for interpreter - Interpreter did not pick up request
Cause	Cannot run FCode interpreter.
Remedy	Check if efdaemon(1M) is running. If not, please start efdaemon(1M).

Message	fcode_timer: Timeout waiting for interpreter - Interpreter is executing request
Cause	Timeout waiting for FCode interpreter.
Remedy	Please wait for a while and check the status of the PCI card via cfgadm (cfgadm_pci(1M)). If the PCI card is configured, there is no problem with the system. If not, reduce the system load and repeat the operation once again.

Message	fc_remove_child: not found
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	pcicfg: Fail:cant load nontransparent bridgd driver..
Cause	Failed to attach PCI Bus Bridge driver.
Remedy	Please contact our customer service.

Message	Failed to add conf reg for ntbridge child.
Cause	Failed to update kernel data.
Remedy	Confirm if the system has enough free memory spaces, and repeat the operation once again. If the problem still exists, please contact our customer service.

Message	Cannot map ntbridge child #
Cause	Failed to map PCI Configuration Space.
Remedy	Please contact our customer service.

Message	Error configuring ntbridge child dev=Y
Cause	Failed to configure PCI card. (Y:Device number)
Remedy	Please contact our customer service.

Message	Unconfig Error ntbridge child dev=Y
Cause	Cannot remove the PCI card after the configuration failed. (Y:Device number)
Remedy	Please contact our customer service.

Message	pcicfg: cant unload ntbridge driver..children.
Cause	Cannot remove the PCI card after the configuration failed.
Remedy	Please contact our customer service.

Message	ntbridge: Mem resource information failure
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	ntbridge: IO resource information failure
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	ntbridge bus range invalid !(#,#)
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	Unconfigure: Failed to add conf reg prop for ntbridge child
Cause	Failed to update kernel data.
Remedy	Confirm if the system has enough free memory spaces, and repeat the operation once again. If the problem still exists, please contact our customer service.

Message	pcicfg: Cannot map ntbridge child Y
---------	--

Cause	Failed to map PCI Configuration Space. (<i>Y</i> :Device number)
Remedy	Please contact our customer service.

Message	pcicfg: cannot map config space, to get map type
Cause	Failed to map PCI Configuration.
Remedy	Please contact our customer service.

Message	ntbridge: unconfigure failed
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	No <i>Z</i> bytes memory window for <i>S</i>
Cause	PCI memory space is insufficient. (<i>Z</i> :bytes, <i>S</i> :device)
Remedy	Please contact our customer service.

Message	No <i>Z</i> bytes IO space window for <i>S</i>
Cause	PCI I/O space is insufficient. (<i>Z</i> :bytes, <i>S</i> :device)
Remedy	Please contact our customer service.

Message	PCI Expansion ROM is not accessible
Cause	PCI Expansion ROM is not accessible
Remedy	Please contact our customer service.

Message	pcihp (<i>DX</i>): unknown event # for this slot <i>S</i>
Cause	Unknown event occurred. (<i>D</i> :Driver name, <i>X</i> :Instance number, <i>S</i> :Slot name)
Remedy	Please contact our customer service.

Message	<i>DX</i>: PCI device <i>Y</i> Failed on Unconfigure
Cause	Failed to unconfigure PCI card. (<i>D</i> :Driver name, <i>X</i> :Instance number, <i>Y</i> :Device number)
Remedy	Repeat the operation once again. If the problem still exists, please contact our customer service.

Message	<i>DX</i>: PCI device <i>Y</i> Busy
Cause	PCI card is busy. (<i>D</i> :Driver name, <i>X</i> :Instance number, <i>Y</i> :Device number)
Remedy	Repeat the operation once again. If the problem still exists, please contact our customer service.

Message	<i>DX</i>: PCI device <i>Y</i> Failed on Configure
Cause	Failed to configure PCI card. (<i>D</i> :Driver name, <i>X</i> :Instance number, <i>Y</i> :Device number)
Remedy	Repeat the operation once again. If the problem still exists, please contact our customer

	service.
--	----------

Message	pcihp (DX): failed to attach one or more drivers for the card in the slot S
Cause	Failed to attach drivers. (D:Driver name, X:Instance number, S:Slot name)
Remedy	Please check if the corresponding device driver is installed. Please install the corresponding device driver to the system. If the problem still exists, please contact our customer service.

Message	DX: pcihp instance already initialized!
Cause	There may be inconsistency in the system. (D:Driver name, X:Instance number)
Remedy	Please contact our customer service.

Message	DX: can't allocate pcihp structure
Cause	There may be inconsistency in the system. (D:Driver name, X:Instance number)
Remedy	Please contact our customer service.

Message	DX: failed to register the bus with HPS
Cause	There may be inconsistency in the system. (D:Driver name, X:Instance number)
Remedy	Please contact our customer service.

Message	DX: pcihp instance is busy
Cause	PCI driver corresponding to the instance number is busy. (D:Driver name, X:Instance number)
Remedy	Ensure there is no access to PCI driver and repeat the operation once again.

Message	pcihp (DX): ddi_create_minor_node failed for pci dev Y
Cause	Failed to create minor node. (D:Driver name, X:Instance number, Y:Device number)
Remedy	Please contact our customer service.

Message	pcihp (DX): install event handler failed for pci dev Y
Cause	There may be inconsistency in the system. (D:Driver name, X:Instance number, Y:Device number)
Remedy	Please contact our customer service.

Message	pcihp (DX): Card is still in configured state for pci dev Y
Cause	PCI card is still in configured state. (D:Driver name, X:Instance number, Y:Device number)
Remedy	Repeat the operation once again. If the problem still exists, please contact our customer service.

Message	pcihp_new_slot_state: unknown slot state <i>U</i>
Cause	There may be inconsistency in the system. (<i>U</i> : slot state)
Remedy	Please contact our customer service.

Message	pcihp (<i>DX</i>): SLOT_CONFIGURE event re-occured for pci dev <i>Y</i> (slot <i>S</i>)
Cause	There may be inconsistency in the system. (<i>D</i> :Driver name, <i>X</i> :Instance number, <i>Y</i> :Device number, <i>S</i> :Slot name)
Remedy	Please contact our customer service.

Message	pcihp (<i>DX</i>): failed to configure the card in the slot <i>S</i>
Cause	There may be inconsistency in the system. (<i>D</i> :Driver name, <i>X</i> :Instance number, <i>S</i> :Slot name)
Remedy	Repeat the operation once again. If the problem still exists, please contact our customer service.

Message	pcihp (<i>DX</i>): SLOT_UNCONFIGURE event re-occured for pci dev <i>Y</i> (slot <i>S</i>)
Cause	There may be inconsistency in the system. (<i>D</i> :Driver name, <i>X</i> :Instance number, <i>Y</i> :Device number, <i>S</i> :Slot name)
Remedy	Please contact our customer service.

Message	Cannot set config space map for pci device number <i>Y</i>
Cause	Failed to map PCI Configuration Space. (<i>Y</i> :Device number)
Remedy	Please contact our customer service.

Message	Cannot Get Board Type..
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	hpcsvc: can't create event dispatch thread
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	hot-plug switch disabled <i>AP_ID</i>
Cause	PCI Hot Plug pushbutton is disabled.
Remedy	PCI Hot Plug pushbutton is disabled. After enabling PCI Hot Plug operation with -x option, repeat the operation once again. If the problem still exists, please contact our customer service.

Message	hot-plug disabled on <i>AP_ID</i>
Cause	PCI Hot Plug operation is disabled.
Remedy	PCI Hot Plug operation is disabled. After enabling PCI Hot Plug operation with -x option,

	repeat the operation once again. If the problem still exists, please contact our customer service.
--	--

Message	unitp not filled
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	Cannot change control register.
Cause	PCI Hot Plug operation is disabled.
Remedy	PCI Hot Plug pushbutton is disabled. After enabling PCI Hot Plug operation with -x option, repeat the operation once again. If the problem still exists, please contact our customer service.

Message	High level interrupt not supported
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	couldn't find slot-table property
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	slot-table undefined
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	can't allocate memory for slot-table
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	bus-status undefined
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	can't allocate memory for bus-status
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	bad format in slot-table
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	ddi_get_iblock_cookie FAILED
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	ddi_regs_map_setup failed.
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	invalid register offset value
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	ddi_create_minor_node failed for port_Y
Cause	Failed to create minor node. (Y:Slot number)
Remedy	Please contact our customer service.

Message	failed to add interrupt
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	66Mhz bus can't accept 33Mhz card
Cause	66Mhz bus can't accept 33Mhz card.
Remedy	Please move the 33Mhz card to 33Mhz bus.

Message	Not supported this bus type
Cause	There may be inconsistency in the system.
Remedy	Please contact our customer service.

Message	Could not power-on AP_ID
Cause	Could not power on the slot.
Remedy	Unable to operate PCI Hot Plug on the slot. Please contact our customer service.

Message	Could not power-off AP_ID
Cause	Could not power off the slot.
Remedy	Unable to operate PCI Hot Plug on the slot. Please contact our customer service.