CITCA

Installation and Operating Information

Order Number: EK-CITCA-OM. 001

Digital Equipment Corporation

August 1992

This document was prepared using VAX DOCUMENT, Version 2.0.

Contents

4 Diagnostics

5 Functional Description

A Registers

B Boot Errors

C Part Numbers

Index

Examples

Figures

Tables

About This Manual

This manual describes the CITCA option, which provides an interface between the Computer Interconnect (CI) bus and the TURBOchannel bus.

Intended Audience

This manual is intended for Digital Customer Service engineers, or licensed customers who install and/or maintain this option.

Manual Structure

This manual contains five chapters:

- Chapter 1, Introduction
- Chapter 2, Site Preparation and Installation
- Chapter 3, Verification and Acceptance
- Chapter 4, Diagnostics
- Chapter 5, Functional Description

An appendix section contains three appendixes:

- Appendix A, Registers
- Appendix B, Boot Errors
- Appendix C, Part Numbers

Related Documentation

The following documentation can provide additional information about subjects relating to the CITCA.

Table 1 Related Documents

Title	Order Number
DECsystem 5900 Owners Guide	EK-D590A-OG
Site Environmental Preparation Guide	EK-CSEPG-MA
DECsystem 5900 Pocket Service Guide	EK-D590A-PS
DECsystem 5900 Site Preparation Guide	EK-D590A-SP
DECstation 5000 Model 240 Maintenance Guide	EK-PM38C-MG
SC008 Star Coupler User Guide	EK-SC008-UG
TURBOchannel Maintenance Guide	EK-TRBOC-MG

1
Introduction

This chapter introduces the Computer Interconnect to TURBOchannel Adapter (CITCA). The CITCA option provides the hardware interface between the Computer Interconnect (CI) bus and the TURBOchannel bus in a TURBOchannel based system.

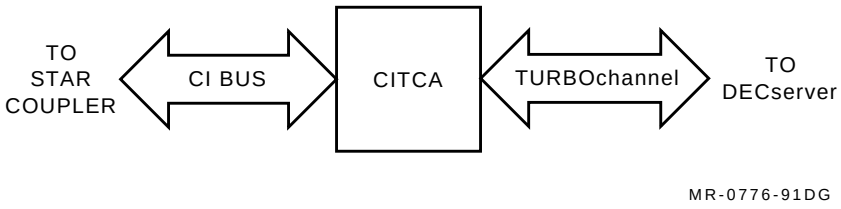
This chapter includes:

- General Description
- Physical Description
- Features
- Specifications

1.1 General Description

The CITCA is an intelligent controller, residing on a triple-slot module, that connects the TURBOchannel bus to the CI bus. Figure 1–1 is a simplified diagram of the bus connection.

Figure 1–1 Simplified CITCA Connection



The CITCA uses a microprocessor and a control store FLASH PROM/RAM to communicate with the operating system and control the CI interface. The CITCA processes commands found on the command queues and packets received from the CI bus. The CITCA supports independent dual CI paths.

1.2 Physical Description

1.2.1 Components

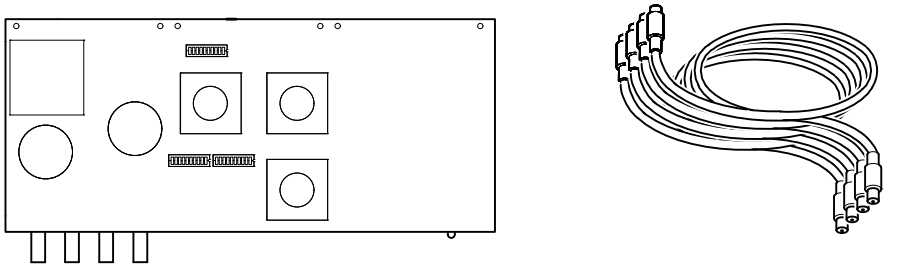
The CITCA-AA option consists of a single CITCA module. It is installed with four ThinWire CI cables that are ordered separately. Table 1–1 lists the part numbers for the hardware components shown in Figure 1–2. A list of related parts can be found in Appendix C.

Table 1–1 CITCA Hardware Components

Component	Part Number
CITCA module	54-20253-01
Set of four ThinWire cables	BNCIA-xx[1]

[1] xx = cable length in meters; either 10 or 20

Figure 1–2 Hardware Components



1.2.2 Logic Components

The CITCA module contains the following logical parts:

- TURBOchannel interface logic
- Packet Buffer (PB) RAMs
- Control Store RAMs
- Control Store FLASH PROMs
- Local Store RAMs
- Five gate arrays:
  - TMOV: TURBOchannel interface control and data movers
  - MCWI: PB memory control
  - MCDP: Microprocessor and sequencer
  - CIRT (2): CI receive/transmit logic, including Manchester encode/decode logic, byte framer and shift register

See Chapter 5 for details on the function of the module.

1.3 Features

The following list describes the features of the CITCA option.

- Resequencing dual path operation
- 62.5 Mbyte/second packet buffer
- 2 data movers
- 32-bit internal data paths
- 64-nanosecond microcycle
- Parity on all internal buses and control stores
- Field updateable control store
- Simultaneous writes to multiple HSCs capability
- Diagnostic loopback capability (both internal and external)
- Data integrity through Cyclic Redundancy Checking (CRC)
- Round-robin arbitration at heavy loading, for each path
- Contention arbitration at light loading, for each path
- Packet-oriented data transmission
- Immediate acknowledgment of packet reception
- Operational modes:
  - Uninitialized
  - Disabled
  - Enabled

1.4 Specifications

Table 1–2 Environmental Specifications

Temperature	
Operating	10°C to 40°C (50°F to 104°F) ambient temperature with a gradient of 10°C (18°F) per hour.
Storage/shipping	-40°C to 66°C (-40°F to 151°F) ambient temperature with a gradient of 20°C (36°F) per hour.
Relative humidity	
Operating	10% to 90% with a maximum wet bulb temperature of 28°C (82°F) and a minimum dew point of 2°C (36°F) with no condensation
Storage/shipping	10% to 95% with no condensation
Altitude	
Operating	Sea level to 2.4 km (8000 ft)
	Maximum operating temperatures decrease by a factor of 1°C/1000 ft (1.8°F/1000 ft) for operation above sea level
Shipping/storage	Up to 9.1 km (30,000 ft) above sea level (actual or effective by means of cabin pressurization)
Shock	10 Gs peak at 10 ms duration in three mutually perpendicular axes (maximum)

Table 1–3 Electrical Specifications

Power consumption
+5.0 V at 6.6 A nominal
+12.0 V at less than 1 A nominal

2

Site Preparation and Installation

Caution

Electrostatic discharge (ESD) can damage the module components. Always use an antistatic wrist strap and mat when handling the module. Place the wrist strap on your wrist, and attach the wrist strap and antistatic mat to the system chassis. Do not touch the gate arrays.

2.1 Site Preparation

Before you begin, take the following steps to ensure a smooth installation and minimize system downtime.

- The CITCA option requires three adjacent TURBOchannel option slots — make sure they are available in the system.
- Discuss the switch settings with your customer. See Section 2.2.
- Ask the system manager to configure the ULTRIX operating system to include the CITCA and CI devices. The *ULTRIX Guide to Configuration File Maintenance* (AA-ME90C-TE) describes how to include devices into the operating system configuration.
- Label and route the CI cables. See Section 2.2.4.

- Refer to the *Site Environmental Preparation Guide* to make sure that you are installing the CITCA in the proper environment.
- Know the current version of microcode so you will know if an update is necessary. This information may be found in the CITCA section of the Revision Matrix document.
- Be prepared — make sure that the following items are available on site.
  - An antistatic kit
  - CITCA-AA
  - 4 ThinWire cables
  - 12 1/4-inch screws (6 are shipped with the module, and 6 more are available after you remove the blank bulkhead panels)
  - This document

2.2 Module Setup

Before installing or replacing a CITCA module, set the switches to the proper position. Table 2–1 provides a summary of the switch settings, and Figure 2–1 shows their location on the module. It also shows the orientation of the switches on switchpack 3. The orientation of the switches on the other two switchpacks is the same as switchpack 3. Tables 2–2 through 2–9 list the individual switch settings.

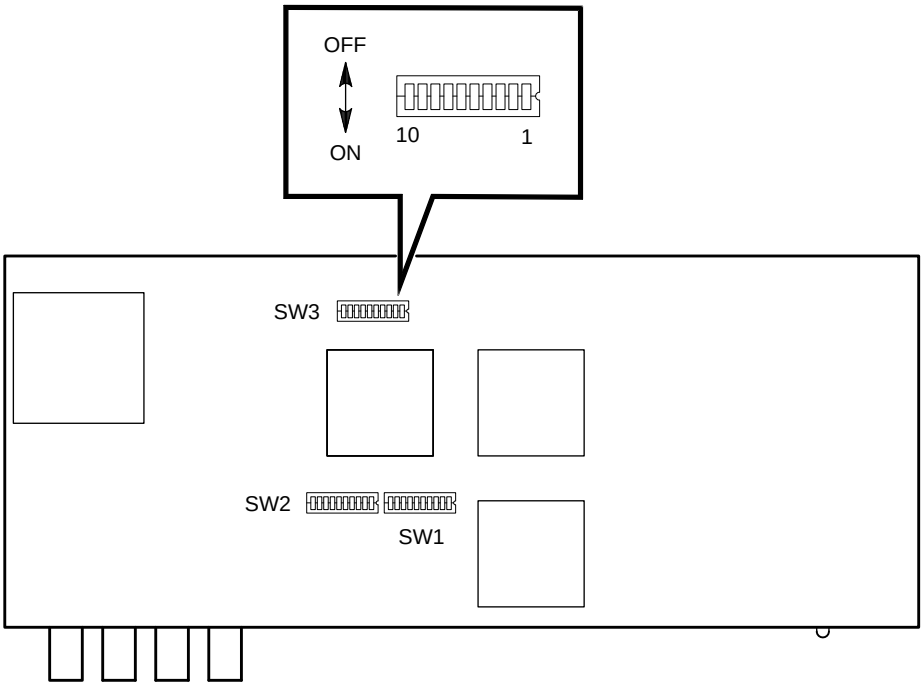
Table 2–1 Summary of Switch Settings

Switches	Description	Default	Refer to . . .
Switchpack 1			
SW1	FLASH PROM write control	Off	Table 2–2
SW2	Arbitration mode	Off	Table 2–3
SW3 - SW10	CI node address	—[1,2]	Table 2–4
Switchpack 2			
SW1	Extend header	Off	Table 2–5
SW2	Extend ACK timeout	Off	Table 2–6
SW3 - SW10	CI node address - duplicate	—[1,2]	Table 2–4
Switchpack 3			
SW1 - SW3	Quiet slot time	1 & 2 Off 3 On	Table 2–7
SW4 - SW6	Arbitration modulus (node count)	All Off	Table 2–8
SW7 - SW10	Boot time	All Off	Table 2–9

[1]The system manager should determine the value of the CI node address based on the system configuration.

[2]The CI node address (switchpack 1, SW3 - SW10) and its duplicate (switchpack 2, SW3 - SW10) must be configured exactly the same.

Figure 2–1 CITCA Module Switches and Connectors



2.2.1 Switchpack 1

Table 2–2 Switchpack 1 - Switch 1

SW1	Description
Off *	FLASH PROM writes are enabled
On	FLASH PROM writes are disabled

* = Default position
Off = switch open
On = switch closed

Table 2–3 Switchpack 1 - Switch 2

SW2	Description
Off *	Allow normal CI star coupler arbitration.
On	Disable normal CI star coupler arbitration — Do not use this position.

* = Default position
Off = switch open
On = switch closed

Setting the CI Node Address Switches

Your customer will determine what the CI node address should be. Each switch represents a bit weight 2^n as shown below. To interpret a CI node address, add the bit weight values of each switch that is set. For example, to set CI node address 100, set SW4, SW5 and SW8 (64 + 32 + 4 = 100).

Switch	Bit Weight
3	128
4	64
5	32
6	16
7	8
8	4
9	2
10	1

Table 2–4 Switchpack 1 - Switches 3 through 10 and Switchpack 2 - Switches 3 through 10

CI Node Address (Decimal)	SW3	SW4	SW5	SW6	SW7	SW8	SW9	SW10
0	Off	Off	Off	Off	Off	Off	Off	Off
1	Off	Off	Off	Off	Off	Off	Off	On
2	Off	Off	Off	Off	Off	Off	On	Off
.				.				
.				.				
.				.				
223	On	On	Off	On	On	On	On	On
224	On	On	On	Off	Off	Off	Off	Off
.				.				
.				.				
.				.				
255	On	On	On	On	On	On	On	On

See the previous page for an explanation on the switch settings.
Off = switch open
On = switch closed

Note

The CI node address (switchpack 1, SW3 - SW10) and its duplicate (switchpack 2, SW3 - SW10) must be configured exactly the same.

CI Node addresses 224 through 255 are reserved.

2.2.2 Switchpack 2

Table 2–5 Switchpack 2 - Switch 1

SW1	Description
Off *	Normal header
On	Extended header — Do not use this position.

* = Default position
Off = switch open
On = switch closed

Table 2–6 Switchpack 2 - Switch 2

SW2	Description
Off *	Normal ACK timeout
On	Extended ACK timeout — Do not use this position.

* = Default position
Off = switch open
On = switch closed

Switchpack 2 — Switches 3 through 10

The CI node address duplicate switches (Switchpack 2 — switches 3 through 10) are set exactly the same as the CI node address switches (Switchpack 1 — switches 3 through 10). Refer to Table 2–4 to see how these switches are set.

2.2.3 Switchpack 3

Table 2–7 Switchpack 3 - Switches 1 through 3, Quiet Slot Time Configuration Switches

Quiet Slot Count	SW1	SW2	SW3
7	Off	Off	Off
10[1] *	Off	Off	On
Reserved	Off	On	Off
Reserved	Off	On	On
Reserved	On	Off	Off
Reserved	On	Off	On
Reserved	On	On	Off
Programmable	On	On	On

[1] See note below
* = Default position
Off = switch open
On = switch closed

Note

The quiet slot time configuration switches must be set to 10.

Table 2–8 Switchpack 3 - Switches 4 through 6

Node Count (Decimal)	SW4	SW5	SW6
16 *	Off	Off	Off
32	Off	Off	On
64	Off	On	Off
128	Off	On	On
224	On	Off	Off
Reserved	On	Off	On
Reserved	On	On	Off
Reserved	On	On	On

* = Default position
Off = switch open
On = switch closed

Table 2–9 Switchpack 3 - Switches 7 through 10

Time (seconds)	SW7	SW8	SW9	SW10
1500 *	Off	Off	Off	Off
1400	Off	Off	Off	On
1300	Off	Off	On	Off
1200	Off	Off	On	On
1100	Off	On	Off	Off
1000	Off	On	Off	On
900	Off	On	On	Off
800	Off	On	On	On
700	On	Off	Off	Off
600	On	Off	Off	On
500	On	Off	On	Off
400	On	Off	On	On
300	On	On	Off	Off
200	On	On	Off	On
100	On	On	On	Off
0	On	On	On	On

* = Default position
Off = switch open
On = switch closed

2.2.4 CI Cable Routing

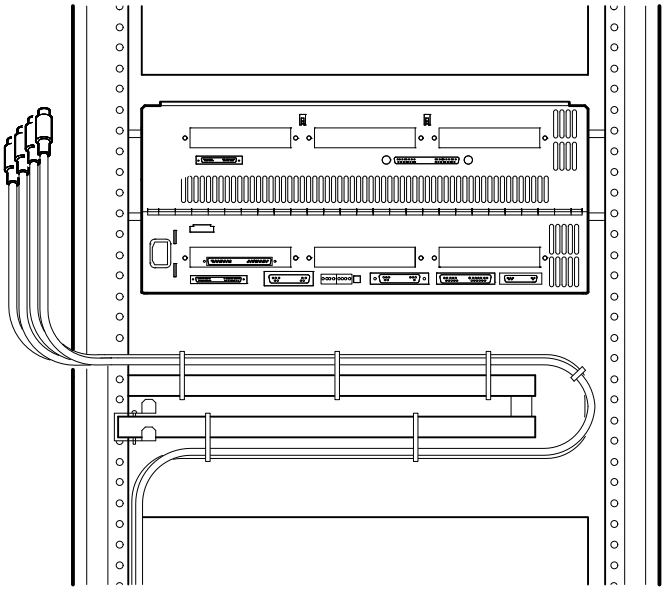
Take the following steps to label and route the ThinWire CI cables (BNCIA-10 /20).

1. Label both ends of the 4 ThinWire cables as follows:

   Cable 1 - TA (transmit path A)
   Cable 2 - TB (transmit path B)
   Cable 3 - RA (receive path A)
   Cable 4 - RB (receive path B)

2. Connect one end of the cables to the star coupler. Refer to the *SC008 Star Coupler User Guide* for more information.
3. Route the cables from the star coupler to the system where you are installing the CITCA module.
4. Route the cables up the back of the system cabinet as shown in Figure 2–2.
5. Use one or two tie wraps to loosely hold the cables in place for now. They will be tightened after the CITCA is installed.

Figure 2–2 Cable Routing



MR-0306-92DG

2.3 Installation

This section describes how to install a CITCA module into a TURBOchannel based system. To illustrate the procedure, we described the installation of a CITCA module into a TURBOchannel extender (TCE) on a DECsystem 5900 system. The installation procedure is similar in other systems. Refer to your system documentation for system-specific details.

Caution

Electrostatic discharge (ESD) can damage the module components. Always use an antistatic wrist strap and mat when handling the module. Place the wrist strap on your wrist, and attach the wrist strap and antistatic mat to the system chassis. Do not touch the gate arrays.

To install a CITCA option, perform the following steps.

1. Perform the tasks described earlier, which include:
   - Set the switches on the CITCA module. See Sections 2.2.1 through 2.2.3.
   - Label and route the CI cables. See Section 2.2.4.
   - Ask the system manager to configure the ULTRIX operating system to include the CITCA and CI devices.
2. Ask the system manager or operator to shut down the operating system.

   Note

   The console commands are case sensitive. Be sure to type commands *exactly* as shown in the examples that are provided. Refer to your system documentation for detailed information about the console commands.

3. Use the console command *printenv* to see if the system has autoboot enabled. If the halt action field shown on the console printout equals b (for boot), disable the autoboot with the following command.

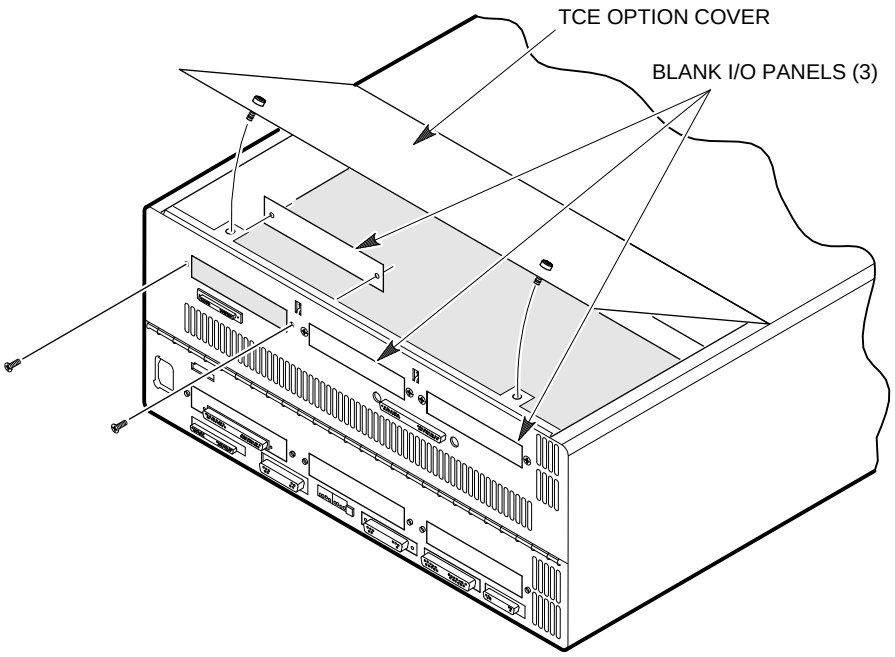
   ```
   setenv haltaction h
   ```

4. Open the front door of the system cabinet.
5. Power down the system by placing the system power switch on the CPU drawer in the O (off) position.
6. Pull out the stabilizer foot from the front of the cabinet.
7. Pull out the CPU drawer until you hear the safety locks click. This will allow you to install the CITCA into the system drawer from the front of the cabinet.

   OR

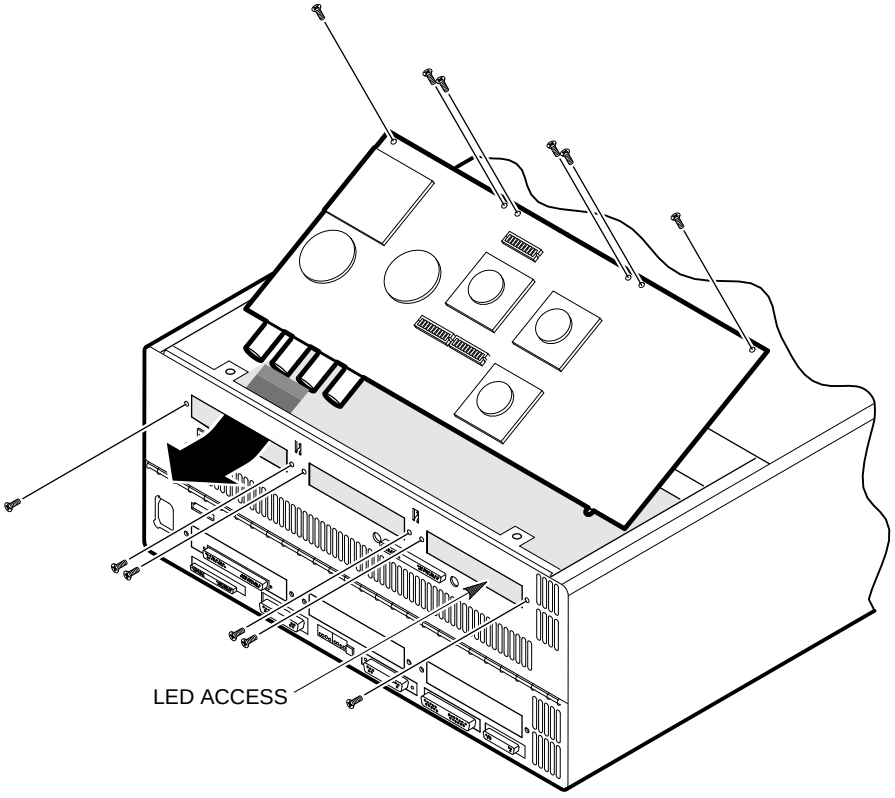
   Pull out the mass storage drawer that is located directly above the CPU drawer until you hear the safety locks click. This will allow you to install the CITCA into the CPU drawer from the back of the cabinet.
8. Open the rear door of the cabinet.

Figure 2–3 CPU Drawer — Rear View



9. Release the two TCE option cover screws by turning them to the left until they release, and remove the option cover. See Figure 2–3.
10. Make sure that the TCE is empty. The CITCA uses all three TCE slots. If a module is already installed in the TCE, move it to an unused TURBOchannel slot.
11. Remove the three blank I/O panels, and save the screws. See Figure 2–3.
12. Put the CITCA in place by first aligning the back of the module with the I/O panel cutouts, then aligning the front of the module with the three TURBOchannel connectors. Next apply backward and downward pressure to the front of the module using care to avoid bending the connector pins until the module pops into place. See Figure 2–4.

Figure 2–4 Module Placement



13. Secure the module in place with 12 screws as shown in Figure 2–4. If the CPU drawer was pulled out in step 7, you may want to push it in before securing the screws on the bulkhead.
14. Verify the operation of the CITCA by performing the acceptance procedures listed in Chapter 3.

2.3.1 Removing a CITCA

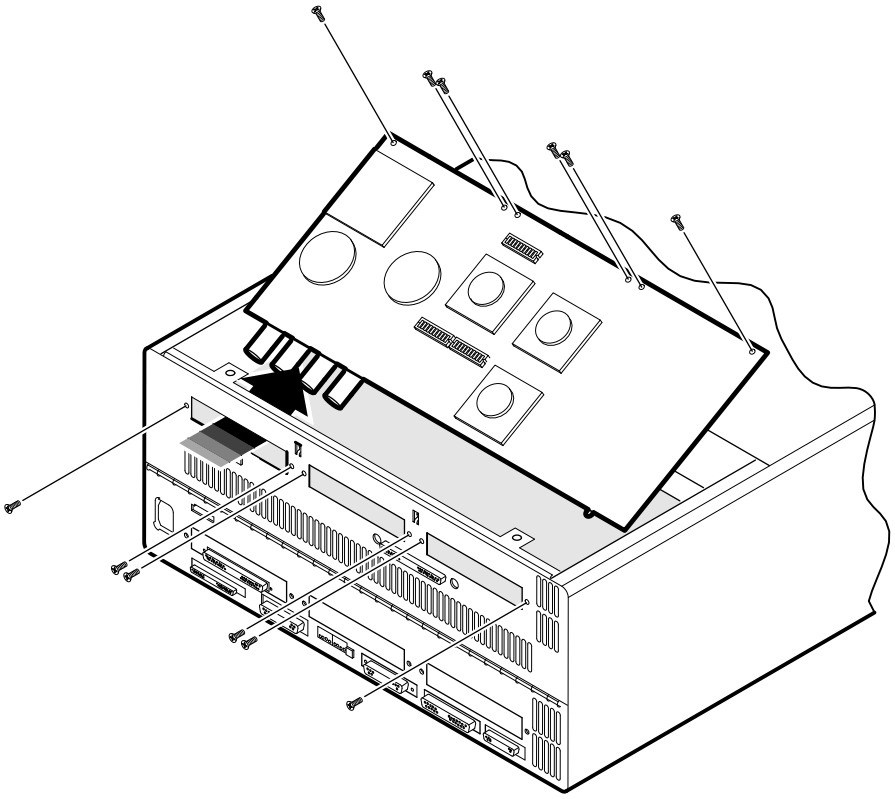
Caution

Electrostatic discharge (ESD) can damage the module components. Always use an antistatic wrist strap and mat when handling the module. Place the wrist strap on your wrist, and attach the wrist strap and antistatic mat to the system chassis. Do not touch the gate arrays.

The procedure to remove a CITCA module from a DECsystem 5900 system is described in the following steps. The removal procedure for other systems is similar.

1. Ask the system manager or operator to shut down the operating system.
2. Power down the system by placing the system power switch in the O (off) position.
3. Pull out the stabilizer foot from the front of the cabinet.
4. Verify that the CI cables are labeled correctly (Figure 3–2), and label them if necessary.
5. Disconnect the CI cables.
6. Go to the front of the cabinet and pull out the CPU drawer until you hear the safety locks click.
7. Release the two TCE option cover screws by turning them to the left until they release, and remove the option cover. See Figure 2–3.
8. Remove the 12 screws shown in Figure 2–5.
9. Gently lift the front of the module up until the TURBOchannel connectors release from the expander module, then pull the module forward and away. See Figure 2–5.
10. If you are not replacing the CITCA, replace the three I/O panel covers. See Figure 2–3.

Figure 2–5 Module Removal



MR-0217-92DG

3
Verification and Acceptance

3.1 Verification Procedure

The following list describes all the steps required to verify the installation and operation of the CITCA option. This procedure begins with:

- The operating system down
- The system power off
- The CI cables disconnected from the CITCA

1. Attach two CI attenuator/loopback connectors (PN 12-19907) as follows:

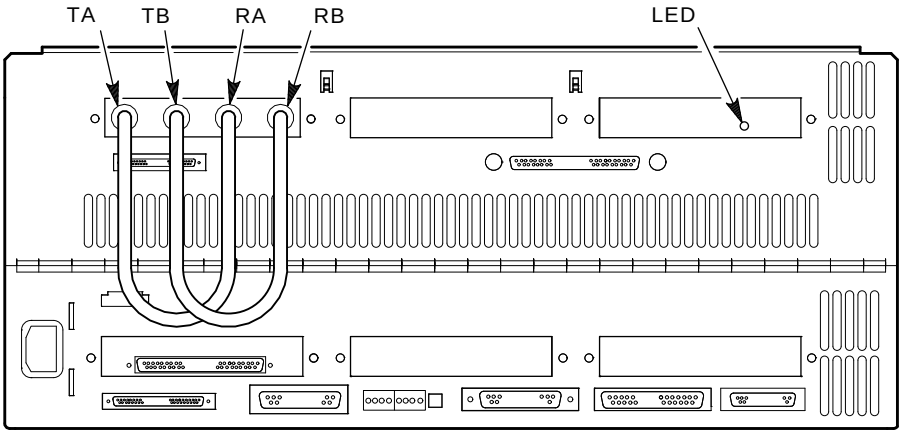
   At the back of the system, attach one end of the first CI attenuator cable to J1 (labeled TA) and the other end to J3 (labeled RA).

   Connect one end of the second CI attenuator cable to J2 (labeled TB) and the other end to J4 (labeled RB). See Figure 3–1.

2. Power up the system by placing the system power switch in the | (on) position.

3. Check to see if the self-test LED is on (Figure 3–1). TCAST lights the self-test LED on the CITCA module to show that it has completed successfully. TCAST verifies the operation of the CITCA option, and is run automatically when the system is powered up. See Chapter 4 for details about TCAST.

Figure 3–1 Connecting the Attenuator/Loopback Cables



MR-0779-91DG

4. Type *cnfg* at the console prompt to verify that the system sees the CITCA, and the microcode is up to date. Example 3–1 shows the console output that you can expect to see, although the output on your system console may differ slightly to show specific features of your system. The cnfg command provides the following information:

   - The first column lists the TURBOchannel slot into which the device is installed.
   - The next column lists the device type and manufacturer.
   - The next column lists the diagnostic microcode version.
   - The next column lists the device architecture.
   - The last column lists important information about the device. For CITCA it lists the CI node address and the contents of the failing test code (PDFLT) field of the port diagnostic control/status register (PDCSR).

Note

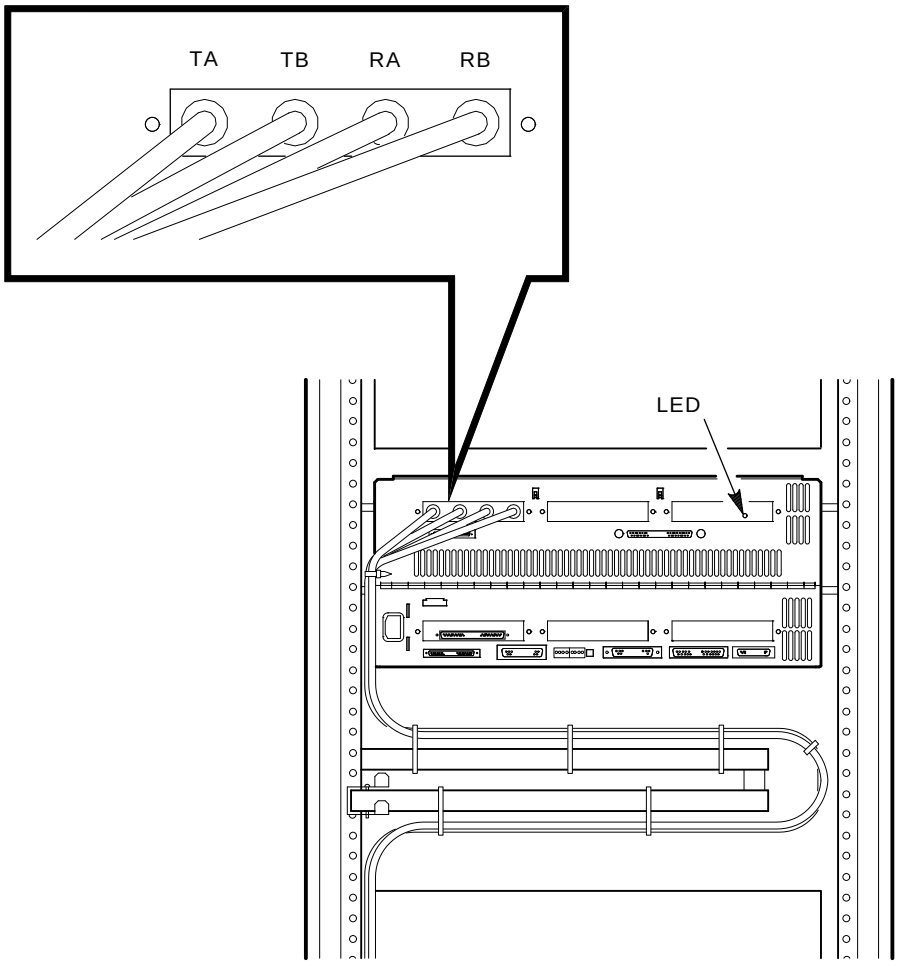
PDFLT code *e0* indicates that TCAST completed successfully.

Example 3–1 Using the cnfg Command

```
>>cnfg
  3: KN03-AA  DEC      V5.1b    TCF0  (128 MB,   1 MB NVRAM)
                                      (enet: 08-00-2b-1c-1d-07)
                                      (SCSI = 7)
  0: CITCA-AA DEC      0x.01    TCF0[OPTION ROM]
  (CI node address: 014{d}, PDFLT = e0)
```

5. Use the *boot slot/l* command to verify the ability of the CITCA to send and receive packets. See Section 4.3.1 for more details about this command.
   - If the loopback test fails, refer to Section 3.2, Troubleshooting. Appendix B describes the error codes.
   - If the loopback test passes, continue on to the next step.
6. Power down the system by placing the system power switch in the O (off) position.
7. Remove the CI attenuator/loopback cables.
8. Attach the CI cables as shown in Figure 3–2. If the cables have not been routed in the cabinet, see Section 2.2.4.
9. Power up the system by placing the system power switch in the | (on) position.
10. Verify that the self-test LED is on. See Figure 3–2.
11. Use the Revision Matrix Document and the *cnfg* console command to verify the revision level of the CITCA FLASH PROM. Example 3–1 shows the console output from the *cnfg* command. Refer to Section 4.4.8 if you need to update the FLASH PROM.
12. Use the *boot slot/c* console command to verify the CI path connections, and also to make sure there are no duplicate CI node numbers in the CI environment. See Section 4.3.2 for details about this command.
13. Use a tie wrap to hold the CI cables on the cable arm as shown in Figure 3–2.

Figure 3–2 Connecting the CI Cables



MR-0783-91DG

14. Replace the drawer cover, push in the drawer, and close the system doors.
15. Enable auto-boot if you disabled it earlier.
16. Ask the system manager or operator to boot the operating system.

3.2 Troubleshooting

This section contains a flowchart to be used to diagnose and repair problems that may occur during the installation of a CITCA module into a system. It also contains information to help you recognize error conditions described in the flowchart.

Table 3–1 lists the PDFLT codes that may be corrected in the field. PDFLT codes not listed in Table 3–1 require module replacement.

Table 3–1 PDFLT Codes

Code	Description	Required Action
D1	Node and duplicate node addresses differ	Set SP1, SW3 - 10 and SP2, SW3 - 10 to the same value.
E0	TCAST completed successfully	None
F1	Illegal arbitration modulus (cluster size)	Set SP3, SW4 - 6 to a value of 16 or 32.
F2	Invalid node number - Node number is higher than the node count	Set the node address value (SP1, SW3 - 10 and SP2, SW3 - 10) lower than the node count value (SP3, SW4 - 6); or set the node count value higher than the node address value.
F3	Quiet slot time is illegal	Verify SP3, SW1 - 3 is set to a value of 10.
F4	Node and duplicate node addresses differ	Set SP1, SW3 - 10 and SP2, SW3 - 10 to the same value.

Figure 3–3 Installation Verification/Troubleshooting Flowchart (Part 1 of 2)

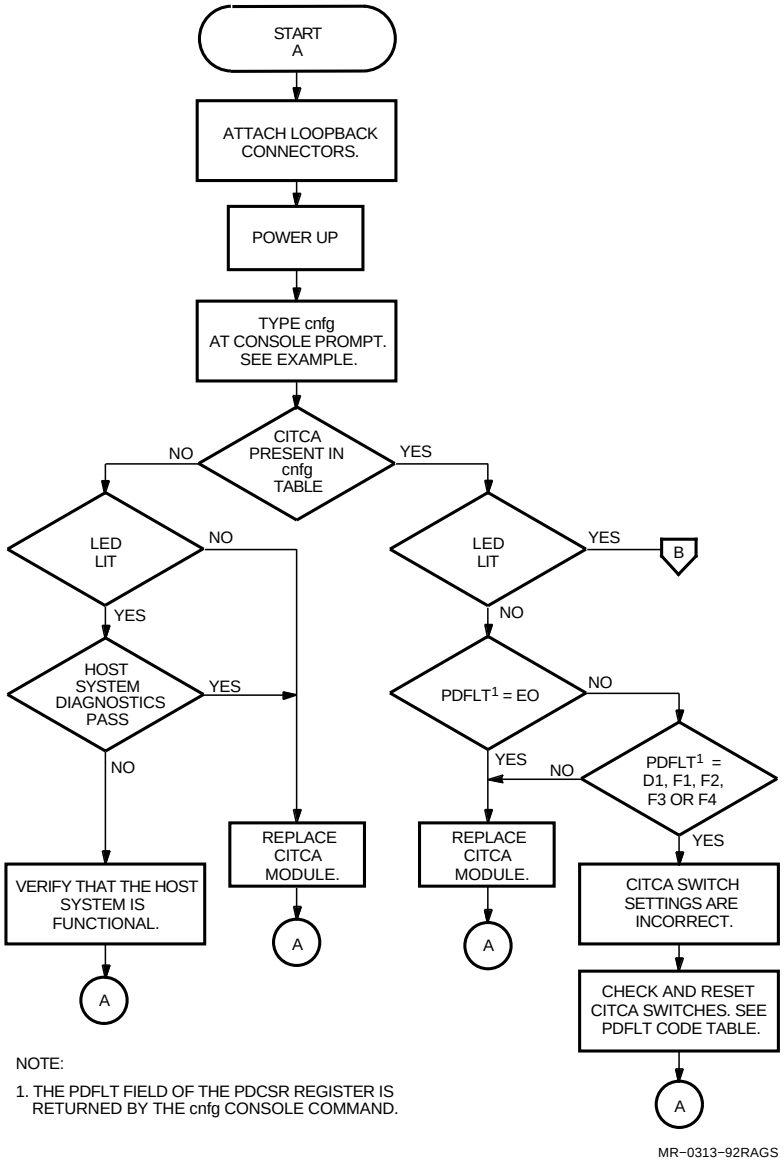
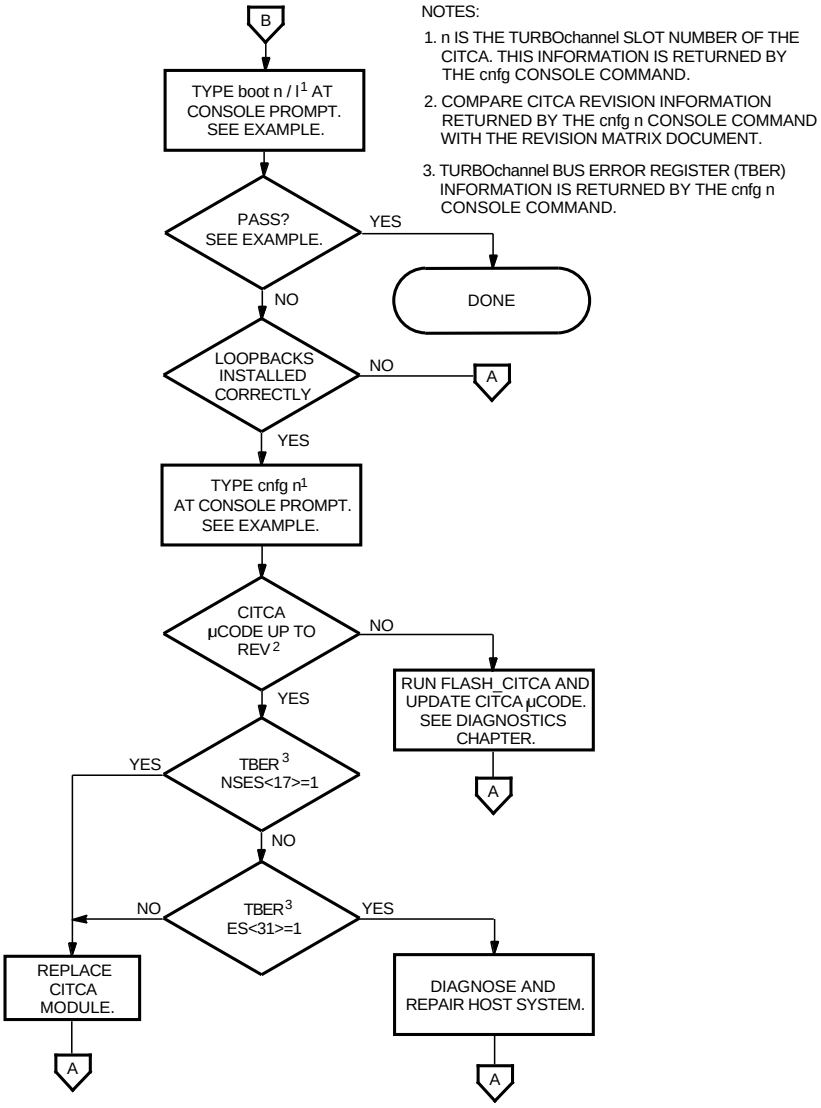


Figure 3–4 Installation Verification/Troubleshooting Flowchart (Part 2 of 2)



Example 3–2 shows a sample of the console output from the *boot n/l*[1] command. The output on your system console may differ slightly to reflect specific features of your system. In this example, a CITCA is installed in TURBOchannel slot 0.

Example 3–2 Boot n/l

```
>>boot 0/l
TCA Slot Number = 0, Mips Address = be000000
[TCA Boot Code Version: V1]
[Functional ucode version is :ae]

 *** Adapter Max PGRP before reading cnfg jumpers: 0.
 *** Adapter Max PGRP after the reading : 15.

         CITCA CI own Node number is: 5.

 *** CI path A loopback test:  OK
 *** CI path B loopback test:  OK

>>
```

Example 3–3 shows the console output from the *cnfg n*[1] console command. The output on your system console may differ slightly to reflect specific features of your system. In this example, a CITCA is installed in TURBOchannel slot 0.

Example 3–3 Using the *cnfg n* Command

```
>>cnfg 0
  0: CITCA-AA DEC      0x.01   TCF0[OPTION ROM]
  (CI node address: 001{d}, PDFLT = e0{x} )

The Slot Base Address = be800000 of TURBOchannel Slot 0
TDEV  = 80ff00ca
TBER  = 1c000000
PDCSR = 800000e0
ASNR  = 7fffffff
        ---------------------------------------
        CITCA Module Serial Number = AS268435455
        CITCA Module H/W revision- = S15
        CITCA Diagnostic Microcode revision   0x.01
        CITCA Functional Microcode revision   0x.01
        CITCA MIPS_resp  Microcode revision   0x.01
        CITCA External Switch Enable Register value:
                  SWCHENR:[ 00200101 ]
        ---------------------------------------
```

[1] *n* is the TURBOchannel slot number where the CITCA is installed.

3.3 Booting from CITCA

To boot the system software from an MSCP disk drive on the CI bus, use the format *boot slot/CI node_ra_drive/file* where:

- *boot* specifies the boot command.
- *slot* specifies the CITCA slot number.
- *CI node* specifies the node number of the HSC drive controller.
- *ra* specifies the type of disk drive that performs the boot operation.
- *drive* specifies the MSCP drive number.
- *file* specifies the name of the file that you want to boot.

See your system user guide for optional arguments for the boot command.

The command *boot 0/12ra3/vmunix* shows what the boot command would look like if you were trying to boot a system disk through the CITCA under the following conditions:

- The CITCA is installed in TURBOchannel slot 0
- The CI node number of the HSC disk controller is 12
- You are booting from disk number 3
- You wish to boot the file named vmunix

3.3.1 Boot Errors

If the system fails to boot the operating software from an MSCP disk drive on the CI bus, perform the following steps:

1. Use the troubleshooting flowchart shown in Figure 3–3.
2. Use the *boot slot/c* console command to verify the CI path connections, and also to make sure there are no duplicate CI node numbers in the CI environment. See Section 4.3.2 for details about this command.
3. Make sure that the ULTRIX configuration file includes the CITCA and CI devices. Refer to the *ULTRIX Guide to Configuration File Maintenance*.
4. Refer to Appendix B for information about troubleshooting boot errors.

4
Diagnostics

The following diagnostic tools may be used to verify correct operation of the CITCA.

- TCAST — CITCA power-up self-test
- PST CITCA — MIPS-based power-up self-test, CITCA subtest
- CI verification — CI functional testing
- FLASH_CITCA[1] — CITCA standalone diagnostic and FLASH PROM update utility

4.1 TCAST

TCAST verifies the operation of the CITCA. Located in FLASH PROM on the CITCA module, the TCAST uses a bottom-up approach in testing; each test executes only after all preceding tests have successfully completed.

[1] The CD ROM that contains the FLASH_CITCA diagnostic is not available to the field except in the case of a field-upgradable microcode update.

4.1.1 TCAST Tests

The tests that are performed by TCAST are as follows:

Test 1 - MCDP Processor ALU Status and Branch Test
Test 2 - ALU Arithmetic/Logical Function Test
Test 3 - General Purpose Register Test
Test 4 - Microsequencer Stack Test
Test 5 - Internal Bus Loopback Test
Test 6 - Interval Timer Test
Test 7 - Local Store Test
Test 8 - Memory Control and Wire Interface Test
Test 9 - Data Mover A Test
Test 10 - Data Mover B Test
Test 11 - TURBOchannel Commander Test
Test 12 - TURBOchannel Responder Test
Test 13 - Data Mover Loopback Test
Test 14 - MCWI Error Detection Logic Test
Test 15 - TMOV Error Detection Logic Test
Test 16 - Interrupt Control Registers Test
Test 17 - CI Maintenance Loopback Test

4.1.2 Running TCAST

TCAST runs automatically when the system is powered up or the CITCA is reset. Alternately, it may be run as test 4 from FLASH_CITCA. It takes less than 1 second to complete. Pass/fail information is provided as shown in Table 4–1. Figure 4–1 shows the location of the self-test passed LED on the CITCA module. It can be seen through an access hole in the rear bulkhead shield.

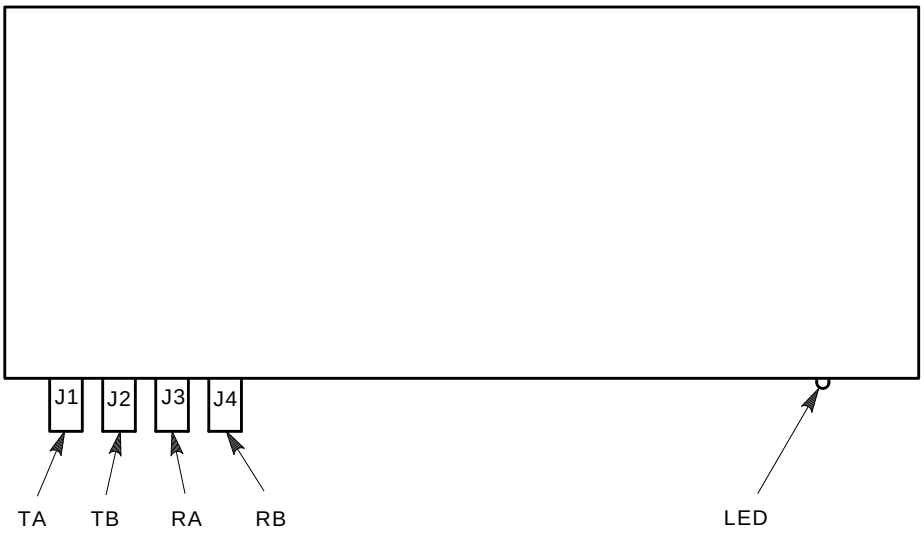
Table 4–1 TCAST Pass/Fail Information

Location of Error Indicator	If TCAST Passed . . .	If TCAST Failed . . .
Self-test Passed LED on the CITCA [1]	LED on	LED off
Self-test Failed (STF) bit in the TURBOchannel Bus Error Register (TBER) [2]	Bit cleared	Bit set

[1] See Figure 4–1

[2] TBER (BB + 08), bit 16

Figure 4–1 Location of CITCA Self-Test Passed LED



4.1.3 Interpreting Error Information

In most cases, a TCAST failure requires the replacement of the CITCA module, however an incorrect switch setting could cause a failure that may be corrected in the field. Use the *cnfg* console command to determine the port diagnostic failing test code (PDFLT). Table 4–2 describes the PDFLT codes that may be corrected in the field. PDFLT codes not listed in Table 4–2 require module replacement.

Table 4–2 PDFLT Codes

Code	Description	Required Action
D1	Node and duplicate node addresses differ	Set SP1, SW3 - 10 and SP2, SW3 - 10 to the same value.
E0	TCAST completed successfully	None.
F1	Illegal arbitration modulus (cluster size)	Set SP3, SW4 - 6 to a value of 16 or 32.
F2	Invalid node number - Node number is higher than the node count	Set the node address value (SP1, SW3 - 10 and SP2, SW3 - 10) lower than the node count value (SP3, SW4 - 6); or set the node count value higher than the node address value.
F3	Quiet slot time is illegal	Verify SP3, SW1 - 3 is set to a value of 10.
F4	Node and duplicate node addresses differ	Set SP1, SW3 - 10 and SP2, SW3 - 10 to the same value.

4.2 PST

The MIPS-based PST runs a comprehensive test of the system hardware. This section describes the subtest that verifies the ability of the host system to communicate successfully with the CITCA.

Refer to your system documentation for details about the PST.

4.2.1 PST CITCA Subtests

The PST subtest that tests the CITCA contains the following two tests:

- pdflt — Checks the contents of the PDFLT field in the port diagnostic control and status register (PDCSR) and reports error conditions if they exist.
- ROM — Checks the CITCA option ROM header information and reports error conditions if they exist.

4.2.2 Running the CITCA Subtest

PST runs automatically when the system is powered up or reset. Error information is reported through the system console terminal and self-test LED on the module.

To run an individual test on the CITCA, use the format *t slot/test_name* where:

- *t* specifies the test command.
- *slot* specifies the TURBOchannel slot where the CITCA is installed.
- *test_name* specifies the individual test that you wish to run.

Example 4–1 shows the command that you would type if you wished to run the pdflt test on a CITCA that is installed in slot 0. This example also shows how the console reports error information. User input is underlined.

A successful pass of the test does not return a status message.

Example 4–1 Running the pdflt Subtest

```
>>t 0/pdflt
?TFL: 0/pdflt (The CITCA in slot 0 has a self-test error = D1.) [CITCA]
```

4.2.3 Interpreting Error Information

Error information is reported on the system console terminal in the format *?TFL slot/test (error_code: description) [module]* where:

- *?TFL* indicates that an error occurred.
- *slot* specifies the TURBOchannel slot number where the module that reported the error is installed.
- *test* specifies the individual test that failed.
- *error_code: description* specifies which part of the test failed. and provides a brief description of the failure.
- *module* identifies the failing module number.

The error message shown in Example 4–1 states that the CITCA module in slot 0 failed the pdflt test, and the error code is D1. Table 4–2 describes the PDFLT codes that may be corrected in the field. PDFLT codes not listed in Table 4–2 require module replacement.

4.3 CI Verification

The CI verification tests ensure that CITCA is correctly connected to the CI bus and is able to communicate with other nodes in the CI environment. These tests are CITCA specific and use the boot command to execute. The following tests make up the CI verification tests:

CI External Loopback Test
CI Node Configuration Test

4.3.1 CI External Loopback Test

The CI external loopback test allows you to verify the operation of the CITCA module and the CI cables. During this test, the CITCA sends a datagram to its own CI node address — with or without CI attenuator/loopback connectors attached. A successful pass of this test verifies the CITCA send and receive logic and, optionally, the CI cables as well.

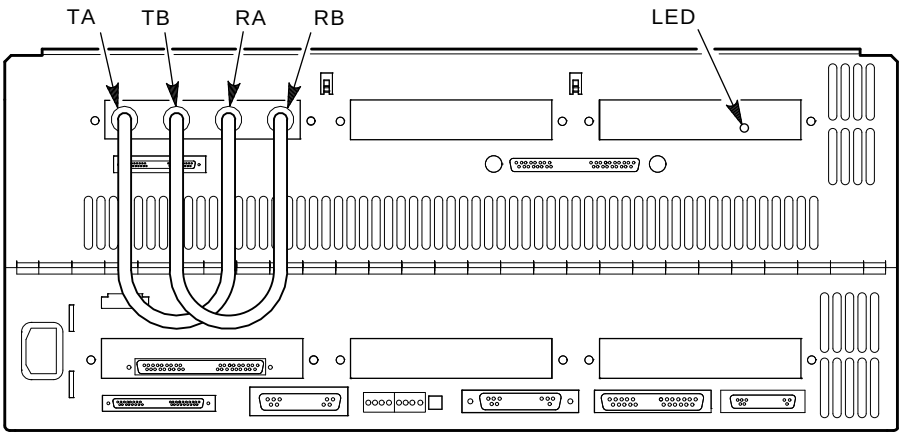
If you suspect a problem with the CI bus, first run the CI external loopback test with CI attenuator/loopback connectors attached to the CITCA. This will verify the operation of the CITCA module. Next, remove the CI attenuator /loopback connectors, attach the CI cables, and run the loopback test a second time. This will verify the operation of the CI cables.

4.3.1.1 Connecting the CI Attenuator/Loopback Connectors

Attach two CI attenuator/loopback connectors as follows:

1. Power down the system by placing the system power switch in the O (off) position.
2. At the back of the system, attach one end of the first CI attenuator/loopback connector to J1 (labeled TA) and the other end to J3 (labeled RA). Connect one end of the second CI attenuator/loopback connector to J2 (labeled TB) and the other end to J4 (labeled RB). See Figure 4–2.
3. Power up the system by placing the system power switch in the | (on) position.

Figure 4–2 Connecting the CI Attenuator/Loopback Connectors



MR-0779-91DG

4.3.1.2 Program Execution

To execute the loopback test, use the format *boot slot/l* where:

- *boot* specifies the boot command.
- *slot* specifies the CITCA slot number.
- *l* specifies that you wish to run the loopback test.

Example 4–2 shows the console output from a successful loopback test. In this example, the CITCA is installed in TURBOchannel slot 0, and the attenuator/loopback connectors are installed. User input is underlined.

Example 4–2 CI External Loopback Test — Passed

```
>>boot 0/l
TCA Slot Number = 0, Mips Address = be000000
[TCA Boot Code Version: V1]
[Functional ucode version is :ae]

 *** Adapter Max PGRP before reading cnfg jumpers: 0.
 *** Adapter Max PGRP after the reading : 15.

         CITCA CI own Node number is: 5.

 *** CI path A loopback test:  OK

 *** CI path B loopback test:  OK
>>
```

If the loopback test fails, error information is output to the console terminal. The last line of the console message contains an error code. Appendix B describes the error codes.

Example 4–3 shows the console output from a loopback test that failed because of a bad connection on CI path B. In this example, the CITCA is installed in TURBOchannel slot 0, and the CI cables are connected to the CITCA. User input is underlined.

Example 4–3 CI External Loopback Test — Failed

```
>>boot 0/l
TCA Slot Number = 0, Mips Address = be000000
[TCA Boot Code Version: V1]
[Functional ucode version is :ae]

 *** Adapter Max PGRP before reading cnfg jumpers: 0.
 *** Adapter Max PGRP after the reading : 15.

         CITCA CI own Node number is: 5.

 *** CI path A loopback test:  OK

 *** Time-out - No response
 *** Please, check CI cable connection
 *** CI path B test:  FAILED

error/status code is: 2f
>>
```

4.3.2 CI Node Configuration Test

The CI node configuration test allows you to verify that the CITCA is able to communicate with the other nodes in the CI environment. During this test, the CITCA issues a request for information to all nodes in the CI environment. The other nodes respond with their node number and hardware type.

If the CI node configuration test fails, error information including an error code is output to the console terminal. Appendix B describes the error codes.

To show the CI node configuration, use the format *boot slot/c* where:

- *boot* specifies the boot command.
- *slot* specifies the CITCA slot number.
- *c* specifies that you wish to check the CI node configuration.

Example 4–4 shows the console output from the CI node configuration test. In this example, the CITCA is installed in TURBOchannel slot 0. User input is underlined.

Example 4–4 CI Node Configuration Test

```
>>boot 0/c
[TCA Slot Number = 0, Mips Address = be000000]
[TCA Boot Code Version: V1]
[Functional ucode version is : ae]

 *** Adapter Max PGRP before reading cnfg jumpers: 0.
 *** Adapter Max PGRP after reading cnfg jumpers: 15.

              CITCA CI own Node number is: 5.

        Node: 2         HSCXX
        Node: 3         HSCXX
        Node: 5         CITCA
        Node: 7         CITCA
        Node: 9         HSCXX
        Node: 13        HSCXX
        Node: 14        CITCA
>>
```

4.4 FLASH_CITCA

FLASH_CITCA contains 7 tests and 8 utility programs for a total of 15 standalone images. It is used in the manufacturing environment to test the module logic and function of the CITCA. Although it extensively tests the CITCA, this diagnostic is not required to verify the operation of the CITCA at the customer site. Field use of FLASH_CITCA is intended to be limited to microcode updates.

FLASH_CITCA runs in standalone mode under the control of the system console executive (REX). It resides on CD ROM, but is not available to the field except as a microcode update. See Section 4.4.2 for examples of loading FLASH_CITCA.

FLASH_CITCA does not support power fail/restart. If a power failure occurs while it is running, FLASH_CITCA must be reloaded and restarted.

4.4.1 FLASH_CITCA Tests and Utilities

FLASH_CITCA contains the following tests:

Test 0: Execute tests 1 through 7
Test 1: TURBOchannel scan path
Test 2: FLASH PROM region checksums

Subtest 1: Self test microcode region checksum
Subtest 2: Functional microcode region checksum
Subtest 3: Responder microcode region checksum
Subtest 4: Responder microcode data 1 region checksum
Subtest 5: Responder microcode data 2 region checksum
Subtest 6: Responder microcode data 3 region checksum
Subtest 7: Responder microcode data 4 region checksum

Test 3: RAM memory
Test 4: TCAST self-test via node reset
Test 5: TURBOchannel contained registers
Test 6: TURBOchannel served registers
Test 7: TURBOchannel host memory DMA data exchange

In addition to the above tests, FLASH_CITCA contains the following utility programs:

Test 8: Clear FLASH PROM regions
Test 9: Clear and erase FLASH PROM regions
Test a: Update all FLASH PROM regions
Test b: Verify all FLASH PROM regions
Test c: INIT_CB - Initialize the FLASH PROM control block
Test d: EXAM_CB - Examine the FLASH PROM control block

Test e: EXAM_PG - Examine the program control block
Test f: Operator - Change flags and TURBOchannel slot

Typing *h* or *?* at the flash_citca> prompt lists all the FLASH_CITCA tests on the console terminal.

4.4.1.1 Test Description

Test 0: Execute tests 1 - 7 — Executes tests 1 - 7 without operator intervention.

Test 1: TURBOchannel scan path — Verifies the data integrity and addressability of the scan path from the TURBOchannel bus to the port scan data register (PSDR).

Test 2: FLASH PROM region checksums — Verifies that the seven regions in FLASH PROM contain valid data. A separate checksum subtest is run for each region.

- Subtest 1: Self-test microcode region checksum — A failure produces a self-test microcode checksum error.
- Subtest 2: Functional microcode region checksum — A failure produces a functional microcode checksum error.
- Subtest 3: Responder microcode region checksum — A failure produces a responder microcode checksum error.
- Subtest 4: Responder microcode data 1 region checksum — A failure produces a responder data 1 checksum error.
- Subtest 5: Responder microcode data 2 region checksum — A failure produces a responder data 2 checksum error.
- Subtest 6: Responder microcode data 3 region checksum — A failure produces a responder data 3 checksum error.
- Subtest 7: Responder microcode data 4 region checksum — A failure produces a responder data 4 checksum error.

Test 3: RAM Memory — Verifies the data integrity and addressability of the RAM.

Test 4: TCAST self-test via node reset — Initiates a complete node reset by writing a 1 to the node reset (NRST) bit in the TURBOchannel Bus Error Register (TBER). The reset causes the TCAST diagnostic to execute.

Test 5: TURBOchannel contained registers — Ensures that the TURBOchannel contained register addresses can be read.

Test 6: TURBOchannel served registers — Ensures that the TURBOchannel served register addresses can be read.

Test 7: TURBOchannel host memory DMA data exchange — Ensures that the CITCA is able to execute read and write DMA commands to the host system memory.

4.4.1.2 Utility Description

Test 8: Clear FLASH PROM regions — Writes 0s in all regions in the FLASH PROM to clear the data. This utility is normally not run; if it is, you must update the FLASH PROM to replace the microcode that was cleared.

Test 9: Clear and erase FLASH PROM regions — Writes 0s in all regions in the FLASH PROM to clear the data, then writes 1s in all regions to erase the data regions. This utility is normally not run; if it is, you must update the FLASH PROM to replace the microcode that was cleared.

Test a: Update all FLASH PROM regions — Allows the user to update the option ROM, functional, diagnostic, and responder firmware stored in the FLASH PROM. This utility includes a copy of the firmware, which it uses to update the FLASH PROM. First, this utility clears and erases the FLASH PROM, then it loads the FLASH PROM with the updated information. Next, the program verifies the success of the update by reading the code from FLASH PROM and comparing it to the copy that is included in FLASH_CITCA.

Test b: Verify all FLASH PROM regions — Allows the user to verify that the FLASH PROM contains the current version of the option ROM, functional, diagnostic, and responder microcode. This utility includes a copy of the firmware, which it compares with the firmware stored in FLASH PROM.

Test c: INIT_CB - Initialize the Control Block — Allows the user to use the operator console terminal to initialize the module serial number and module revision information stored in the control block section of the FLASH PROM. This utility is normally not run.

Test d: EXAM_CB - Examine the Control Block - Displays the following information from the control block section of the FLASH PROM.

- Module serial number
- Module revision level
- Functional microcode revision level
- Diagnostic microcode revision level
- Responder microcode revision level
- Option ROM firmware revision level

- Functional microcode region checksum
- Diagnostic microcode region checksum
- Responder microcode region checksum
- Responder data 1 region checksum
- Responder data 2 region checksum
- Responder data 3 region checksum
- Responder data 4 region checksum

Test e: EXAM_PG - Examine the Program control block — Displays the following information from the program control block section of the FLASH_ CITCA program image.

- Diagnostic microcode revision level
- Functional microcode revision level
- Responder microcode revision level
- Option ROM firmware revision level

Test f: Operator - Change flags and TURBOchannel slot — Allows the user to change program event flags and the TURBOchannel address pointer. This utility is menu driven. The following list shows the program event flags and their default values.

Table 4–3 FLASH_CITCA Event Flags

Event Flag	Description (When Set)	Default Value
Halt	Halt on error	Cleared
Loop	Loop on error	Cleared
Inhibit	Inhibit error message reporting	Cleared
Quick	Runs an abbreviated version of the diagnostic	Set
TURBOchannel Address	Where the diagnostic expects the CITCA to be installed	TURBOchannel slot 1

4.4.2 Loading FLASH_CITCA

The load procedure consists of the following four steps:

1. Booting the FLASH_CITCA program image (Section 4.4.2.1).
2. Resetting the system.
3. Booting the system console program image (Section 4.4.2.2).
4. Overlaying the existing ROM objects with a pointer to the FLASH_CITCA image (Section 4.4.2.3).

4.4.2.1 Booting the FLASH_CITCA Program Image

To boot the standalone diagnostic image, use the format >>> *boot slot/device /flash_citca* where:

- *boot* specifies the boot command.
- *slot* specifies the TURBOchannel slot number of the boot device controller.
- *device* specifies the type and number of the device from which you wish to boot.
- *flash_citca* indicates that you wish to boot the FLASH_CITCA program image.

Example 4–5 shows the command that you would type when:

- The controller for the drive you wish to boot is in option slot 3.
- You are booting an RZ-series drive, disk number 0.
- The file you wish to boot is called flash_citca.

Example 4–5 also shows an example of what you can expect to see as a system response. User input is underlined.

Example 4–5 Loading FLASH_CITCA Program Image

```
>>boot 3/rz0/flash_citca

Ultrixboot - V4.2  Tue Mar 19 06:15:25 EST 1991

Loading 3/rz0/flash_citca ...

Sizes:
text = 0
data = 736224
bss  = 0
Starting at 0xa0330000

Enter image name:
```

4.4.2.2 Booting the System Console Program Image

After the FLASH_CITCA program image has been loaded, **reset the system** by pressing the system reset button. (On the DECsystem 5900, this button is located on the back of the system box.) Then boot the system console program image using the format >>> *boot slot/device/file_name* where:

- *boot* specifies the boot command.
- *slot* specifies the slot number of the boot device controller.
- *device* specifies the type and number of the device from which you wish to boot.
- *file_name* specifies the name of the file you wish to boot.

Example 4–6 shows the command that you would type when:

- The controller for the drive you wish to boot is in option slot 3.
- You are booting an RZ-series drive, disk number 0.
- The file you wish to boot is called bigmax.

Example 4–6 also shows an example of what you can expect to see as a system response. User input is underlined.

Example 4–6 Loading the System Program Image

```
>>boot 3/rz0/bigmax

Ultrixboot - V4.2 Tue mar 19 06:15:25 EST 1991

Loading 3/rz0/bigmax ...

Sizes:
text = 64288
data = 107408
bss  = 22048
Starting at 0xa0240100

KN03-AA V5.0a    [11/14/91 10:15:36 ferguson]
>>
```

4.4.2.3 Overlaying the Existing ROM Objects

After both program images have been loaded, type the three commands shown in Example 4–7. This is a patch to overlay the existing ROM objects for the boot device with a pointer to the FLASH_CITCA image.

Example 4–7 Overlay ROM Objects

```
>>d 0xa0200b98 0xa0330120
>>d 0xa0200b9c 0
>>cnfg
  3: KN03-AA  DEC      V5.0a    TCF0  ( 64 MB,   1 MB NVRAM)
                                      (enet: 08-00-2b-1c-82-92)
                                      (SCSI = 7)
  0: CITCA-AA DEC      V1.00    TCF0  (CI node address: 001{d}, PDFLT = e0{x})
>>
```

4.4.3 Running FLASH_CITCA

The FLASH_CITCA program may be executed after the load procedure described in Section 4.4.2 is complete. To run FLASH_CITCA, use the command `t slot/flash` where:

- *t* specifies the test command.
- *slot* specifies the TURBOchannel slot where the CITCA is installed.
- *flash* specifies that you wish to run the FLASH_CITCA program.

Example 4–8 shows an example of the command and console output. User input is underlined.

Note

FLASH_CITCA uses the default value of 1 for the TURBOchannel slot number. If the CITCA is not installed in slot 1, use the operator utility (menu item f) to direct FLASH_CITCA to the correct value. See Section 4.4.4.

In the DECsystem 5900, the CITCA is typically installed in the TURBOchannel extender - slot 0.

Example 4–8 Running FLASH_CITCA

```
>>t 0/flash

         flash_citca 1.00 {CITCA Utility} Modified 03/20/92

Using CITCA in TURBOchannel address = be800000
TURBOchannel slot # = 01

 Select one for program execution:
          0, Execute Tests 1-7 without Operator input
          1, Test the TURBOchannel Scan Path
          2, Test the Flashprom region checksums
          3, Test the RAM Memory
          4, Test the TCAST Selftest via Node Reset
          5, Test the TURBOchannel Contained Registers
          6, Test the TURBOchannel Served Registers
          7, Test the TURBOchannel Host Memory DMA data exchange
          8, Utility - Clear Flashprom regions (write 0)
          9, Utility - Clear and Erase of Flashprom Memory
          a, Utility - UPDATE all Flashprom regions
          b, Utility - VERIFY all Flashprom regions
          c, Utility - INIT_CB - Init Flashprom Control Block
          d, Utility - EXAM_CB - Examine Flashprom Control Block
          e, Utility - EXAM_PG - Examine program Control Block
          f, Utility - OPERATOR - Change flags and TURBOchannel slot
          h, or ? - retype this header
```

Warning

Menu items 8 and 9 destroy the contents of the FLASH PROM. These utilities should never be run.

4.4.4 Selecting the TURBOchannel Slot Number

If the CITCA is not installed in TURBOchannel slot 1 (the default), use the Operator utility (FLASH_CITCA test f) to change the TURBOchannel slot value. This utility is menu driven, and the procedure consists of four steps.

1. Select *f* at the flash_citca > prompt.
2. Respond to the four event flag prompts.
3. Type y in response to the following line.

   ```
   Change to a different TURBOchannel address (N/y)
   ```

4. Select the correct slot number from the lists of choices.

Example 4–9 shows how the Operator utility is used to change the TURBOchannel value to 0. User input is underlined.

Example 4–9 Selecting the TURBOchannel Slot

```
flash_citca 1.00 >  f

Using CITCA in TURBOchannel address = be800000
TURBOchannel slot # = 01

 Event flag  HALT   is =  CLEARED  - Change flag state (N/y)?
 Event flag  LOOP   is =  CLEARED  - Change flag state (N/y)?
 Event flag INHIBIT is =  CLEARED  - Change flag state (N/y)?
 Event flag  QUICK  is =    SET    - Change flag state (N/y)?
 Change to a different TURBOchannel address (N/y)? y
 Enter 0: DS5900/DS5240, TURBOchannel slot 0, address be000000
 Enter 1: DS5900/DS5240, TURBOchannel slot 1, address be800000
 Enter 2: DS5900/DS5240, TURBOchannel slot 2, address bf000000
 Enter 3: [Reserved for MFG!] TURBOchannel slot 0, address be000000
 Enter 4: [Reserved for MFG!] TURBOchannel slot 1, address be400000
 Enter 5: [Reserved for MFG!] TURBOchannel slot 2, address be800000
 Enter TURBOchannel address menu selection: 0

Using CITCA in TURBOchannel address = be000000
TURBOchannel slot # = 00

 Event flag  HALT   is =  CLEARED
 Event flag  LOOP   is =  CLEARED
 Event flag INHIBIT is =  CLEARED
 Event flag  QUICK  is =    SET

 flash_citca  1.00 >
```

4.4.5 Running Individual FLASH_CITCA Tests

The FLASH_CITCA program is menu driven. At the flash_citca > prompt, select a single character to execute any one of the FLASH_CITCA tests. Example 4–10 shows how to run test 3, the RAM memory test.

Example 4–10 Running Test 3

```
flash_citca V1.00 >  3

Test 3 - RAM Memory (15 seconds)

flash_citca V1.00 >
```

4.4.6 Exiting from FLASH_CITCA

To exit from the FLASH_CITCA program, type q at the flash_citca > prompt. This will return control to the console executive.

When you have finished using the FLASH_CITCA program, power down the system to reset the ROM objects. Failure to do so will cause a failure when you try to boot the operating system because the ROM objects have been set to point to the FLASH_CITCA image. Refer to Section 4.4.2.3.

4.4.7 Fault Isolation

When FLASH_CITCA detects a failure, it isolates the source of the problem to the failing function and reports error information to the console. The error information and register contents are written in hexadecimal notation. Example 4–11 shows an error report. In this example, the failing RAM location is provided (address), along with the expected and received data.

Example 4–11 FLASH_CITCA Error Report

```
flash_citca V1.00 >  3

Test 3 - RAM Memory (15 seconds)

pmcs ram data (555555) error
Address - 0000
Expected       00155555        55555555        55555555
Received       0015d555        55555555        55555555

pmcs ram data (AAAAAA) error
Address - 0000
Expected       002aaaaa        aaaaaaaa        aaaaaaaa
Received       002aefaa        aaaaaaaa        aaaaaad5
```

4.4.8 Updating the FLASH PROM

To update the microcode in the FLASH PROM:

1. Set the FLASH PROM write enable switch on the CITCA module (hardware switchpack 1, switch 1) to off.
2. Load the FLASH_CITCA program (Section 4.4.2).
3. Run the FLASH_CITCA program (Section 4.4.3).
4. Select menu item a at the flash_citca > prompt.
5. Exit the FLASH_CITCA program by typing q at the flash_citca > prompt.
6. Reset the system power by placing the system power switch in the off (O) position, then the on (|) position. See the system documentation for the location of this switch.
7. Verify the success of the update by checking the version of the microcode with the *cnfg* command.
8. Boot the operating system.

Example 4–12 shows the console output from steps 4 through 7. User input is underlined.

Example 4–12 Updating the FLASH PROM

```
flash_citca V1.00 > a

Flashprom clear operation started

   Clearing region 0...
   Clearing region 1...
   Clearing region 2...
   Clearing region 3...
   Clearing region 4...
   Clearing region 5...
   Clearing region 6...
   Clearing region 7...

Flashprom erase operation started
```

(continued on next page)

Example 4–12 (Cont.) Updating the FLASH PROM

```
   Erasing region 0...
   Erasing region 1...
   Erasing region 2...
   Erasing region 3...
   Erasing region 4...
   Erasing region 5...
   Erasing region 6...
   Erasing region 7...

Starting to write the DIAGNOSTIC Microcode (Flashprom region 0)
Starting to write the FUNCTIONAL Microcode (Flashprom region 1)
Starting to write the RESPONDER  Microcode (Flashprom region 2)
Starting to write the RESPONDER    data1   (Flashprom region 4)
Starting to write the RESPONDER    data2   (Flashprom region 5)
Starting to write the RESPONDER    data3   (Flashprom region 6)
Starting to write the RESPONDER    data4   (Flashprom region 7)

Flashprom UPDATE operation completed

Starting to VERIFY the DIAGNOSTIC Microcode (Flashprom region 0)
Starting to VERIFY the FUNCTIONAL Microcode (Flashprom region 1)
Starting to VERIFY the RESPONDER  Microcode (Flashprom region 2)
Starting to VERIFY the RESPONDER    data1   (Flashprom region 4)
Starting to VERIFY the RESPONDER    data2   (Flashprom region 5)
Starting to VERIFY the RESPONDER    data3   (Flashprom region 6)
Starting to VERIFY the RESPONDER    data4   (Flashprom region 7)

Flashprom VERIFY operation completed


flash_citca V1.00> q
>>

! Reset the system power

>>cnfg
7:   KN02-AA   DEC       V5.3a     TCF0    ( 32 MB)
6:   PMAD_AA   DEC       V5.3a     TCF0    (enet: 08-00-2b-1c-1d-07)
5:   PMAZ_AA   DEC       V5.3b     TCF0    (SCSI = 7)
2:   CITCA-AA  DEC       V1.01     TCF0    (CI node address: 001{d}, PDFLT = e0{x})

>>
```

5
Functional Description

The CITCA is an intelligent controller that connects the TURBOchannel bus to the CI bus. It uses its own microprocessor and control store FLASH PROM/RAM to interface with the operating system and control the CI bus. The CITCA processes commands found on the command queues and packets received from the CI bus. The CITCA supports independent, dual CI paths.

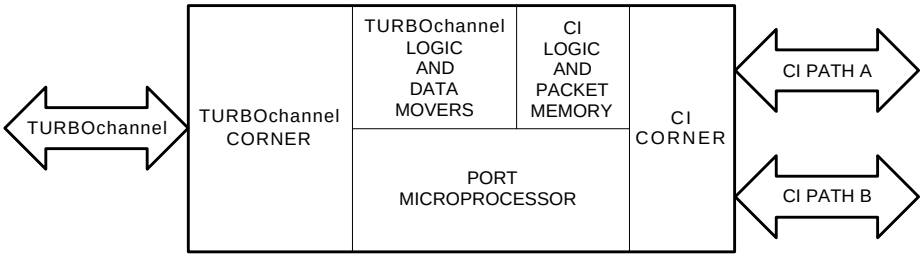
The CITCA can be logically divided into 5 parts as shown in Figure 5–1.

- TURBOchannel Corner - Receives and transmits signals on the TURBOchannel bus.
- TURBOchannel Logic and Data Movers - Controls and responds to commands on the TURBOchannel corner, and moves data packets to and from host memory.
- CI Logic and Packet Memory - Controls the CI corner and implements the CI datalink protocol.
- Port Microprocessor - Processes data, calculates addresses, and controls the rest of the CITCA.
- CI Corner - Interfaces to the CI bus.

5.1 Logic Description

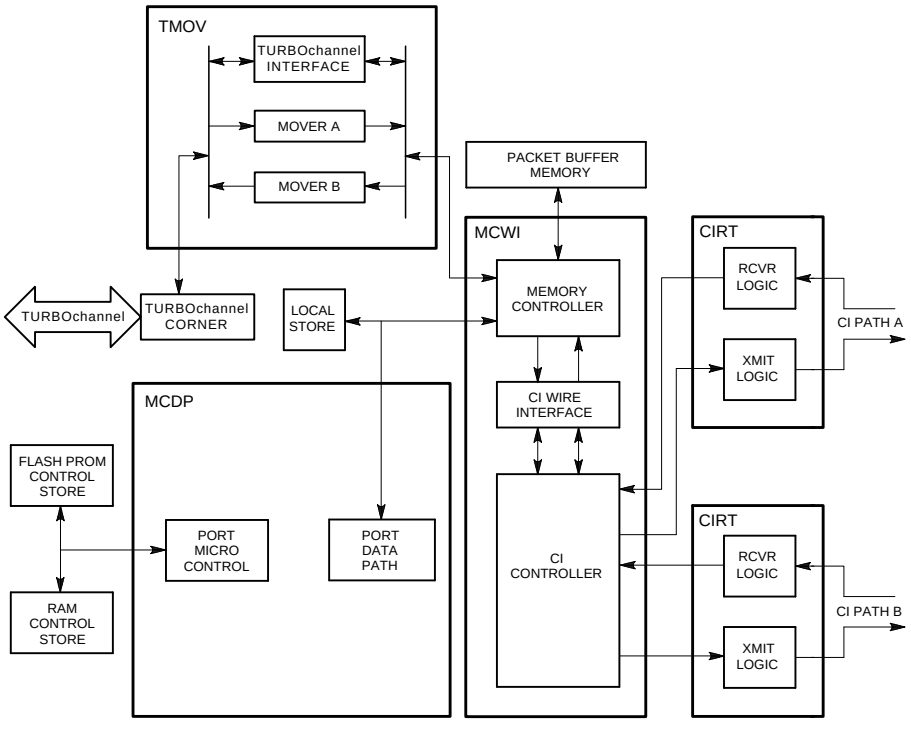
Figure 5–2 is a logical block diagram of the CITCA. This section describes each of the logical blocks.

Figure 5–1 CITCA Simplified Block Diagram



MR-0774-91DG

Figure 5–2 CITCA Block Diagram



MR-0775-91DG

The TURBOchannel corner consists of five transceivers and an 8-bit Programmable Array Logic (PAL) chip. These devices are used to receive and drive the address/data, interrupt, and control signals of the TURBOchannel bus.

The TURBOchannel logic and data mover (TMOV) gate array consists of the TURBOchannel interface and two 32-bit data movers (Mover A and Mover B). The TURBOchannel interface controls and responds to the TURBOchannel corner and contains the interrupt logic. Each data mover transfers data in a single direction; Mover A performs TURBOchannel DMA read transactions and Mover B performs TURBOchannel DMA write transactions. Once started by the MCDP microprocessor, the movers are free running.

The following parts make up the MCDP microprocessor subsystem.

- The micro control and data path (MCDP) gate array
- A control store random access memory (CSRAM)
- A control store flash erasable and programmable read only memory (FLASH PROM)
- An 8K local store random access memory (LSRAM)

The MCDP gate array contains a custom design microprocessor and microsequencer which allow execution of ALU operations every 64 nanoseconds, and next address calculations every 128 nanoseconds. The MCDP microprocessor implements a 32-bit-wide data path with internal parity. It contains 32 general purpose registers (GPRs) and a 16 location microaddress stack.

The memory controller wire interface (MCWI) gate array and its associated packet buffer RAMs make up the CI control logic. The MCWI implements the CI datalink protocol and controls the functions of the CI corner. It also provides the interface, management, and arbitration for the packet buffer memory.

Two independent CI receive/transmit (CIRT) gate arrays make up the CI corner logic. The CI corner interfaces with the CI bus and is capable of servicing a dual path CI. It allows simultaneous use of both paths to transmit and receive data at the same time. The CIRT logic performs Manchester encoding, clock from data separation, and byte framing and synchronization.

5.2 TURBOchannel Transactions

The CITCA responds to the following TURBOchannel transactions:

- Single-word I/O read cycle
- Single-word I/O write cycle (byte masking is not supported)

The CITCA is capable of generating the following TURBOchannel transactions:

- DMA single-word read cycle
- DMA single-word write cycle
- DMA multiple-word read cycle
- DMA multiple-word write cycle
- Interrupt

5.2.1 Sending a Packet to a CI Node

The following list describes the sequence of events that occur in the CITCA when the host system port driver sends a packet to a remote CI node. The CITCA must be in the enabled state before these steps occur.

1. The host system port driver places a command on one of three priority command queues, and instructs the CITCA to look at the command queue by writing the appropriate driver to channel command queue register (CCQ0IR, CCQ1IR, and CCQ2IR).
2. The MCDP microprocessor uses the TURBOchannel interface to initiate a DMA read operation to remove the command from the queue, and validates information in the command.
3. The packet's header information is updated by the MCDP microprocessor, and is placed into the packet buffer to be concatenated with the data portion of the packet.
4. The data portion of the packet is transferred from the host memory, through Data Mover A to the packet buffer.
5. The MCDP microprocessor uses the CI control logic to initiate the transmission of the packet.

6. Upon completion of the transmission, the MCDP microprocessor parses CI transmit done status to determine if the transmitted packet was successfully received by the remote port. CI transmit done status reports one of the following:

   ACK — if the packet was successfully buffered by the remote port.
   NACK — if the transmitted packet was received but not buffered by the remote port.
   NRSP — if the remote port did not send an ACK or NACK response.

   If the transmission was *not* successful (a NACK or NRSP response), the MCDP microprocessor initiates a retransmit. The MCDP microprocessor makes from 4 to 256 attempts to transmit a packet before reporting an error to the host system. The number of retransmit attempts depends on path selection mode, circuit state status, command type, type of response, and the ability of the receiving node to simultaneously receive packets on both CI paths.

   If transmit status is not returned within a specified time-out period, the transmission has failed, and the path is marked bad. There are two types of this failure:

   Arbitration timeout — which results in a retry on the other CI path if the other path is good.
   Transmit timeout — which causes the CITCA to log a fatal ACC error and shut down.

7. The MCDP microprocessor determines if the port driver requested a response.

   - If a response *was not* requested, the microprocessor reacts differently depending on the error status.
     - If there were no errors, the microprocessor ends the transaction by returning the command entry to a free queue.
     - If an error was encountered, the microprocessor places an error response on the response queue before ending the transaction.
   - If a response *was* requested, the microprocessor places a success or error response on the response queue before ending the transaction.

5.2.2 Receiving a Packet from a CI Node

The following list describes the sequence of events that occur in the CITCA when a packet is received from a remote CI node.

1. The CI control logic recognizes its address on an incoming CI packet, and checks for an available receive packet buffer.
   - If one *is* available, the CI control logic accepts the packet and places it in the receive packet buffer.
   - If one *is not* available, the CI control logic sends a negative acknowledge (NACK) signal to the transmitting CI node, and the transmitting node will retry the transaction at a later time.
2. The CI control logic checks the CRC and then notifies the MCDP microprocessor of the received CI packet.
   - If the CRC *is* valid, the CI control logic sends an ACK response signal to the transmitting CI node.
   - If the CRC *is not* valid, no response is sent to the transmitting CI node. The transmitting node eventually flags a no response error (NRSP).
3. The MCDP microprocessor does the following:
   - Validates packet information
   - Removes an entry from a free queue
   - Moves the data portion of the packet through Data Mover B into host system memory
   - Updates packet status and opcode
   - Places a response on the response queue through the TURBOchannel control logic, and interrupts the host system

A
Registers

Some of the CITCA registers are required for communicating with the TURBOchannel bus, or the CI bus, and can be examined by the user from the console or from software using the port driver and the TURBOchannel bus. Other registers are required for internal communications only, and therefore cannot be examined externally. The microcode has access to these internal registers through the port internal bus (port IB). Many of the CITCA registers are accessible from both the port IB and the TURBOchannel bus.

This appendix describes the TURBOchannel accessible registers.

A.1 Register Addressing

The CITCA module has a dedicated address space starting at hexadecimal offset 0000 of the CITCA base address. The base address of the CITCA depends on the set of TURBOchannel slots the CITCA is installed in, and is system implementation specific.

In this appendix, the base address is referred to as BB. The external address of each register is shown as an offset to the base address in the form BB + offset (hex). See Example A–1.

Example A–1 Register Addressing Example

```
TURBOchannel bus error register (TBER)    BB + 08
                                              |
The external TURBOchannel address is --------+
the CITCA base address plus 8 (hex)
```

Table A–1 lists the CITCA registers as seen from the TURBOchannel.

Table A–1 CITCA Register List

Register	Mnemonic	TURBOchannel Address Offset (BB + . . .)
TURBOchannel Device Register	TDEV	0000
TURBOchannel Bus Error Register	TBER	0008
Port Scan Control Register	PSCR	0010
Port Scan Data Register	PSDR	0018
Port Diagnostic Control/Status Register	PDCSR	0020
Channel/Adapter Status Register	CASR	0028
Channel/Adapter Failing Address Register	CAFAR	0030
Adapter Serial Number Register	ASNR	0038
Adapter Block Base Register	ABBR	0080
Channel Command Queue 2 Insertion Register	CCQ2IR	0088
Channel Command Queue 1 Insertion Register	CCQ1IR	0090
Channel Command Queue 0 Insertion Register	CCQ0IR	0098
Adapter Datagram Free Queue Insertion Register	ADFQIR	00A0
Adapter Message Free Queue Insertion Register	AMFQIR	00A8
Channel/Adapter Status Release Control Register	CASRCR	00B0
Channel Enable Control Register	CECR	00B8
Channel Initialize Control Register	CICR	00C0
Port Maintenance/Sanity Timer Control Register	AMTCR	00C8

(continued on next page)

Table A–1 (Cont.) CITCA Register List

Register	Mnemonic	TURBOchannel Address Offset (BB + . . .)
Port Maintenance/Sanity Timer Expiration Control Register	AMTECR	00D0
Adapter Interrupt Holdoff Timer Control Register	AITCR	00D8
Adapter Maintenance Control/Status Register	AMCSR	00E0

A.2 TURBOchannel Registers

The bit numbers within a register bitmap are expressed in decimal notation. A description of the contents of the register follows each bitmap.

TURBOchannel Device Register (TDEV) - BB + 00

Contains the adapter revision and device type.

Figure A–1 TURBOchannel Device Register (TDEV)

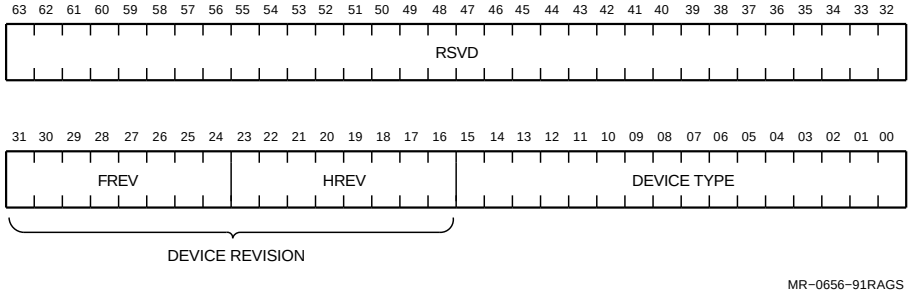


Table A–2 TDEV Bit Description

Bit	Name	Description
<63:32>	RSVD	Reserved (0)
<31:24>	FREV	Firmware revision (RO:R/W,RESETC) - This field indicates the firmware revision of the CITCA. It is loaded by the MCDP processor at the end of a self-test and is decoded as follows:

Bits <31:24>[1]	Description
00000000	V00[2]
00000001	V01
00000002	V02
.	.
.	.
.	.
00010000	V10
00010001	V11
.	.
.	.
.	.
01111111	V7F

[1]The high order bit (bit <31>) designates experimental firmware; it should never be set in the field.

[2]A 0 value indicates an uninitialized node.

(continued on next page)

Table A–2 (Cont.) TDEV Bit Description

Bit	Name	Description
<23:16>	HREV	Hardware revision of the CITCA, decoded as follows: • Bits <23:20> contain the alphabetic field • Bits <19:16> contain the numeric field

Bits <23:20>	Description	Bits <19:16>	Description
0000	Invalid[2]	0000	Invalid[2]
0001	A	0001	1
0010	B	0010	2
0011	C	0011	3
0100	D	0100	4
0101	E	0101	5
0110	F	0110	6
0111	H	0111	7
1000	J	1000	8
1001	K	1001	9
1010	L	1010	10
1011	M	1011	11
1100	N	1100	12
1101	P	1101	13
1110	R	1110	14
1111	S	1111	15

<15:00>	DTYPE	Device type of the option. Device type for CITCA is: 00CA.

[2] A 0 value indicates an uninitialized node.

TURBOchannel Bus Error Register (TBER) - BB + 08

Used to indicate TURBOchannel errors and to force a hard reset of the CITCA.

Figure A–2 TURBOchannel Bus Error Register (TBER)

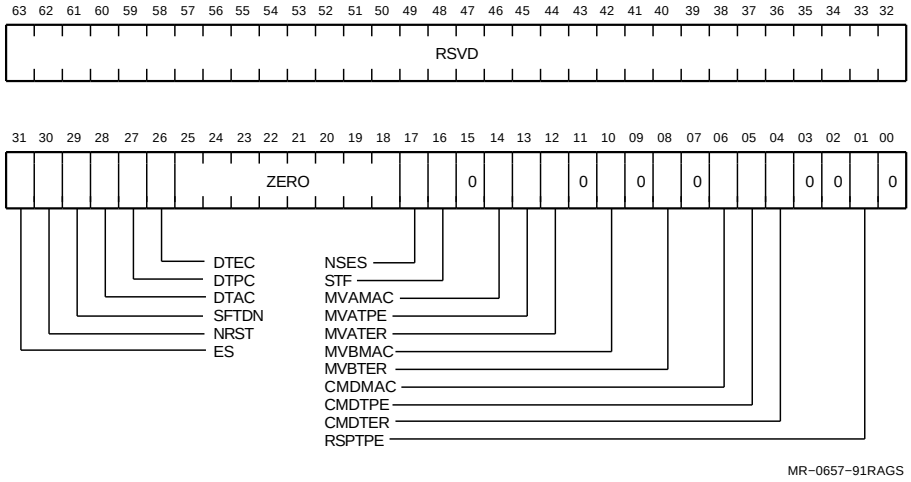


Table A–4 TBER Bit Description

Bit	Name	Description
<63:32>	RSVD	Reserved (0)
<31>	ES	Error Summary - Represents the logical OR of the error bits in this register (bits 29, 17, 14, 13, 12, 10, 8, 6, 5, 4 and 1).
<30>	NRST	Node reset - Writing a 1 to this location initiates a complete power-up reset including TCAST execution.
<29>	SFTDN	Shift done - When set, indicates that a scan shift operation has completed.
<28>	DTAC	Disable TURBOchannel acknowledge check - When set, disables reporting of all TURBOchannel ACK checks by the CITCA.

(continued on next page)

Table A–4 (Cont.) TBER Bit Description

Bit	Name	Description
<27>	DTPC	Disable TURBOchannel parity check - When set, disables reporting of all TURBOchannel parity checks by the CITCA.
<26>	DTEC	Disable TURBOchannel error check - When set, disables reporting of all TURBOchannel ERR signal receptions by the CITCA.
<25:18>	—	Zero (0)
<17>	NSES	Node-specific error summary - This bit is set when one or more error bits are set in AMCSR.
<16>	STF	Self-test fail - When set, indicates that the CITCA has not yet passed its self-test.
<15>	—	Zero (0)
<14>	MVAMAC	Mover A missing ACK - When set, indicates that a data mover A DMA read transaction terminated due to a missing TURBOchannel ACK.
<13>	MVATPE	Mover A TURBOchannel parity error - When set, indicates that a data mover A DMA read transaction terminated due to the detection of a parity error on the data returned from memory on the TURBOchannel.
<12>	MVATER	Mover A TURBOchannel error - When set, indicates that a data mover A DMA read transaction terminated due to the reception of the TURBOchannel ERR signal from the TURBOchannel memory control.
<11>	—	Zero (0)
<10>	MVBMAC	Mover B missing ACK - When set, indicates that a data mover B DMA write transaction terminated due to a missing TURBOchannel ACK.
<09>	—	Zero (0)
<08>	MVBTER	Mover B TURBOchannel error - When set, indicates that a data mover B DMA write transaction terminated due to the reception of the TURBOchannel ERR signal from the TURBOchannel memory control.
<07>	—	Zero (0)
<06>	CMDMAC	Commander missing ACK - When set, indicates that an expected ACK signal was not received in response to a commander port DMA read or write transaction.

(continued on next page)

Table A–4 (Cont.) TBER Bit Description

Bit	Name	Description
<05>	CMDTPE	Commander TURBOchannel parity error - When set, indicates that the commander port DMA read transaction terminated due to the detection of a parity error on the data returned from memory on the TURBOchannel.
<04>	CMDTER	Commander TURBOchannel error - When set, indicates that the commander port DMA transaction terminated due to the reception of the TURBOchannel ERR signal from the TURBOchannel memory control.
<03:02>	—	Zero (0)
<01>	RSPTPE	Responder TURBOchannel parity error - When set, indicates that the responder port detected a parity error on the data or address sent to the CITCA on an I/O write or read operation.
<00>	—	Zero (0)

Port Scan Control Register (PSCR) - BB + 10

Used to load and dump the TMOV-MCDP scan path.

Figure A–3 Port Scan Control Register (PSCR)

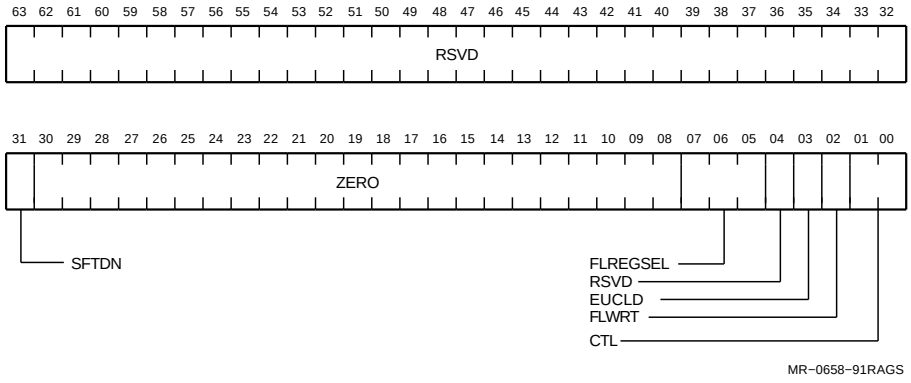


Table A–5 PCSR Bit Description

Bit	Name	Description
<63:32>	RSVD	Reserved (0)
<31>	SFTDN	Shift done - When set, indicates the completion of a scan shift operation.
<30:08>	—	Zero (0)
<07:05>	FLREGSEL	FLASH PROM region select - Selects one of the following eight regions in FLASH PROM to load data: 0 = TCAST Firmware Region 1 = Functional Firmware Region 2 = MIPS Responder Region 3 = Spare Region 4 = MIPS Responder Data 1 Region 5 = MIPS Responder Data 2 Region 6 = MIPS Responder Data 3 Region 7 = MIPS Responder Data 4 Region
<04>	RSVD	Reserved (0)
<03>	EUCLD	E_UCODE_LD - When set, enables the designated external pins to be used as the scan path.
<02>	FLWRT	Write FLASH PROM
<01:00>	CTL	Scan control - Control bits for the scan logic with values as follows:

Bit 01	Bit 00	Diagnostic Shift Register	Diagnostic Control Register
0	0	Hold	Hold
0	1	Hold	Load
1	0	Shift	Hold
1	1	Load	Hold

Port Scan Data Register (PSDR) - BB + 18

Used for loading and retrieving data during a scan operation.

Figure A–4 Port Scan Data Register (PSDR)

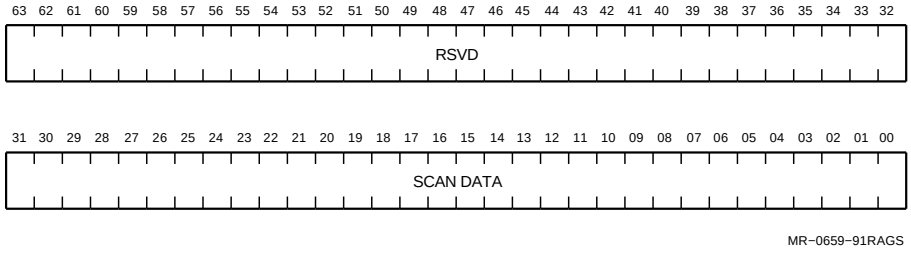


Table A–7 PSDR Bit Description

Bit	Name	Description
<63:32>	RSVD	Reserved (0)
<31:00>	DATA	Scan data

Port Diagnostic Control/Status Register (PDCSR) - BB + 20

Used by the TCAST to report failed test numbers.

Figure A–5 Port Diagnostic Control/Status Register (PDCSR)

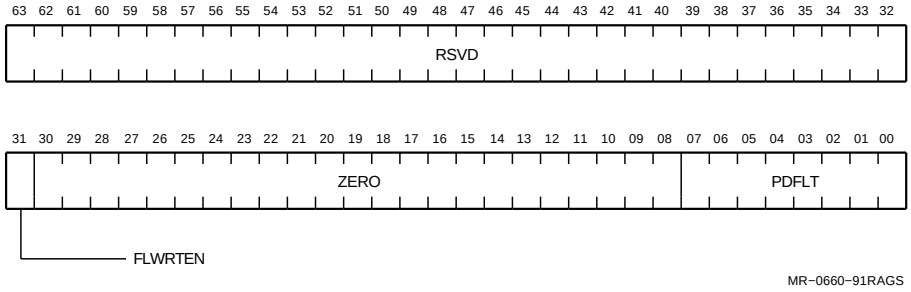


Table A–8 PDCSR Bit Description

Bit	Name	Description
<63:32>	RSVD	Reserved (0)
<31>	FLWRTEN	FLASH PROM write enable
<30:08>	—	Zero (0)

(continued on next page)

Table A–8 (Cont.) PDCSR Bit Description

Bit	Name	Description
<07:00>	PDFLT	Failing test code - Contains the TCAST test number about to be executed.

Hex Code	Description
AA	Self-test passed
01 - DF	TCAST test number that failed
E0	Responder Initialization Passed Code
E1	Initiated Region 1 Copy (Functional Region)
E2	Initiated Region 2 Copy (RBD Diagnostic Region)
E3 - E7	Not Used
E8	Unexpected Responder Interrupt
E9	Unexpected GPR Value
EA	Unexpected RWBMR Bits Set
EB	ROM Read Access Below Byte Address 3A0
EC	ROM Write Access Below Byte Address 3A0
ED	Processor Trap to 100
EE	RBD Parser Error
EF	Not Used
F1	Illegal arbitration modulus (cluster size) value in switches
F2	Invalid node number - Node number is higher than the node count
F3	Quiet slot time is illegal
F4	Node and duplicate node addresses differ

Channel/Adapter Status Register (CASRLO) & (CASRHI) - BB + 28 & BB + 2C

This 64-bit contained register returns status to the port driver after an interrupt.

The CASR is valid only after an interrupt and before writing the CASRCR.

Note

To read all 64 bits of CASR, the host must read addresses BB + 28 (CASRLO) and BB + 2C (CASRHI).

Figure A–6 Channel/Adapter Status Register (CASR)

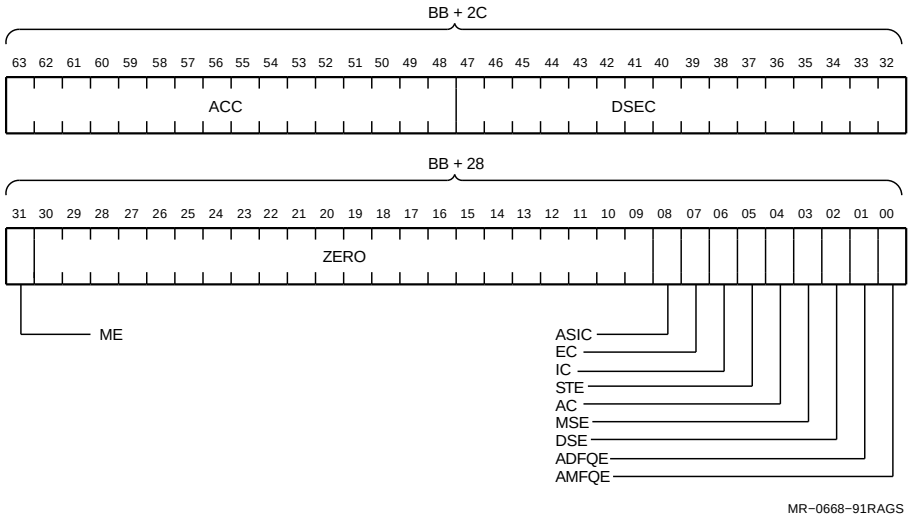


Table A–9 CASR Bit Description

Bit	Name	Description
<63:48>	ACC	Abnormal condition code - This code is valid only if CASR_AC is set. ACC consists of the following two fields: • Bits <63:56> define the failing module within the microcode. • Bits <55:48> contain the specific error that occurred.
<47:32>	DSEC	Data structure error code (RO:R/W) - This code is valid only if CASR_DSE is set. DSEC consists of the following two fields: • Bits <47:40> contain errors specific to the CITCA implementation. • Bits <39:32> contain architecturally defined errors.
<31>	ME	Maintenance error
<30:09>	—	Zero (0)
<08>	ASIC	Adapter single interrupt completion
<07>	EC	Enable complete
<06>	IC	Initialization complete
<05>	STE	Sanity timer expiration
<04>	AC	Abnormal condition - The actual error code is located in the ACC field error code.
<03>	MSE	Memory system error
<02>	DSE	Data structure error
<01>	ADFQE	Datagram free queue exhausted
<00>	AMFQE	Message free queue exhausted

Channel/Adapter Failing Address Register (CAFAR) - BB + 30

This 64-bit register contains the physical memory address where one of the following error conditions occurred:

- Memory System Error (MSE)
- Data Structure Error (DSE)
- The generation of a response queue entry with buffer memory system error status

Figure A–7 Channel/Adapter Failing Address Register (CAFAR)

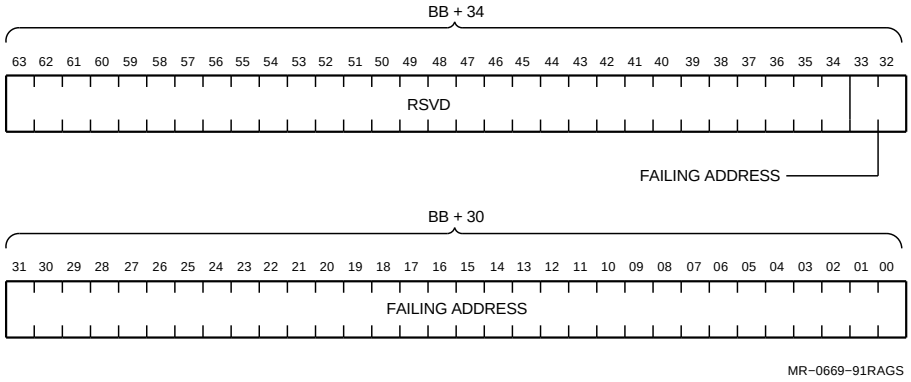


Table A–10 CAFAR Bit Description

Bit	Name	Description
<63:34>	RSVD	Reserved (0)
<33:00>	FAD	Failing address

Note

The failing address is 34 bits long. To read all 34 bits, the host must read addresses BB + 30 and BB + 34.

Adapter Serial Number Register (ASNR) - BB + 38

The serial number of the CITCA Adapter. This register is loaded by microcode with information from the FLASH PROM.

Figure A–8 Adapter Serial Number Register (ASNR)

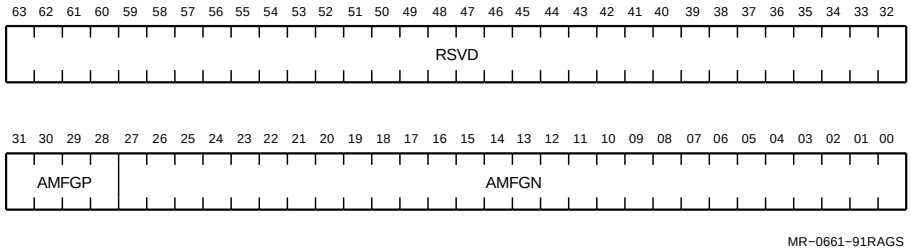


Table A–11 ASNR Bit Description

Bit	Name	Description
<63:32>	RSVD	Reserved (0)
<31:28>	AMFGP	Adapter manufacturing plant - Defines the manufacturing plant with one of the following codes: 0 = Illegal 1 = MRO 2 = ASO 3 = GAO
<27:00>	AMFGN	Adapter manufacturing number - Defines the adapter serial number.

Adapter Block Base Register (ABBR) - BB + 80

The uppermost bits of the physical address of the base of the adapter block. Also used by TCAST to select the test number.

Figure A–9 Adapter Block Base Register (ABBR)

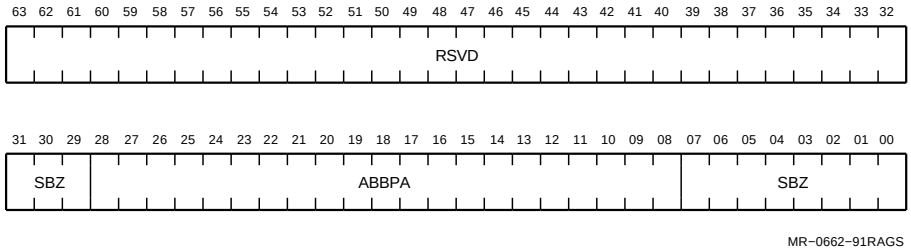


Table A–12 ABBR Bit Description

Bit	Name	Description
<63:32>	RSVD	Reserved (0)
<31:29>	SBZ	Should be zero (0)
<28:08>	ABBPA	Adapter block base physical address - Bits <33:13> of the physical address of the base of the adapter block.
<07:00>	SBZ	Should be zero (0)

Note

ABBR is readable and writeable by the port driver and writeable only when the adapter is in the uninitialized state. Its value before it is written is unpredictable.

Channel Command Queue 2 Insertion Register (CCQ2IR) - BB + 88

The physical address of the new stopper carrier when the port driver inserts entries into DCCQ2.

CCQ2IR is a write-only register. Reading this register returns undefined data.

Figure A–10 Channel Command Queue 2 Insertion Register (CCQ2IR)

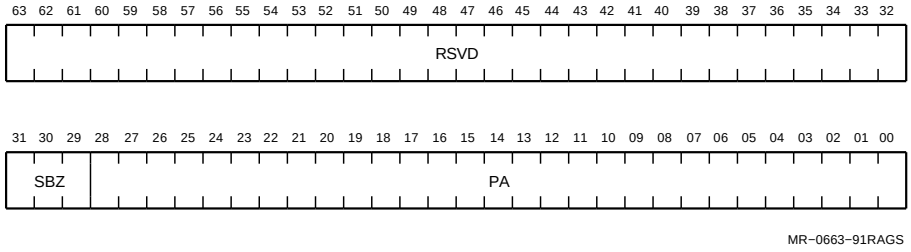


Table A–13 CCQ2IR Bit Description

Bit	Name	Description
<63:32>	RSVD	Reserved (0)
<31:29>	SBZ	Should be zero (0)
<28:00>	PA	Physical address - Bits <33:05> of the new stopper carrier.

Channel Command Queue 1 Insertion Register (CCQ1IR) - BB + 90

The physical address of the new stopper carrier when the port driver inserts entries into DCCQ1.

CCQ1IR is a write-only register. Reading this register returns undefined data.

Figure A–11 Channel Command Queue 1 Insertion Register (CCQ1IR)

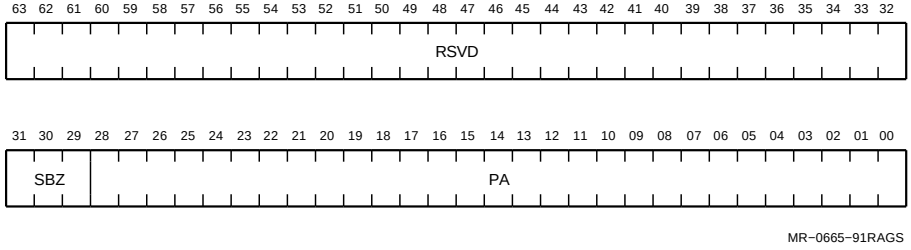


Table A–14 CCQ1IR Bit Description

Bit	Name	Description
<63:32>	RSVD	Reserved (0)
<31:29>	SBZ	Should be zero (0)
<28:00>	PA	Physical address - Bits <33:05> of the new stopper carrier.

Channel Command Queue 0 Insertion Register (CCQ0IR) - BB + 98

The physical address of the new stopper carrier when the port driver inserts entries into DCCQ0.

CCQ0IR is a write-only register. Reading this register returns undefined data.

Figure A–12 Channel Command Queue 0 Insertion Register (CCQ0IR)

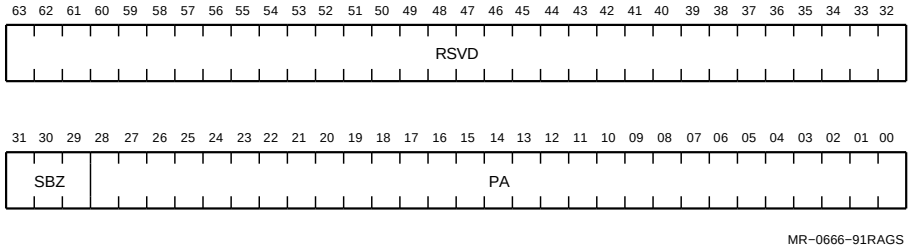


Table A–15 CCQ0IR Bit Description

Bit	Name	Description
<63:32>	RSVD	Reserved (0)
<31:29>	SBZ	Should be zero (0)
<28:00>	PA	Physical address - Bits <33:05> of the new stopper carrier.

Adapter Datagram Free Queue Insertion Register (ADFQIR) - BB + A0

The physical address of the new stopper carrier when the port driver inserts entries into DADFQ.

ADFQIR is a write-only register. Reading this register returns undefined data.

Figure A–13 Adapter Datagram Free Queue Insertion Register (ADFQIR)

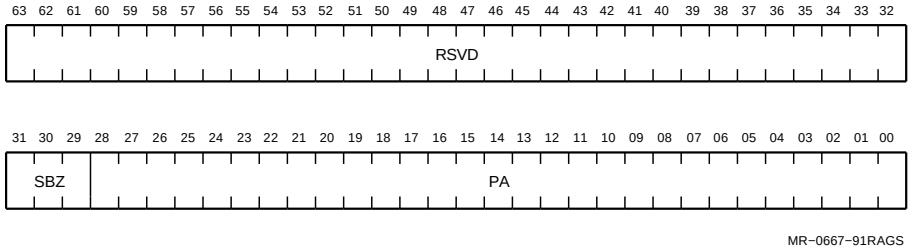


Table A–16 ADFQIR Bit Description

Bit	Name	Description
<63:32>	RSVD	Reserved (0)
<31:29>	SBZ	Should be zero (0)
<28:00>	PA	Physical address - Bits <33:05> of the new stopper carrier.

Adapter Message Free Queue Insertion Register (AMFQIR) - BB + A8

The physical address of the new stopper carrier when the port driver inserts entries into DAMFQ.

Figure A–14 Adapter Message Free Queue Insertion Register (AMFQIR)

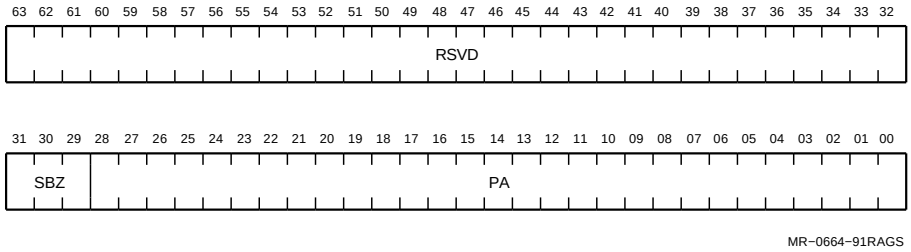


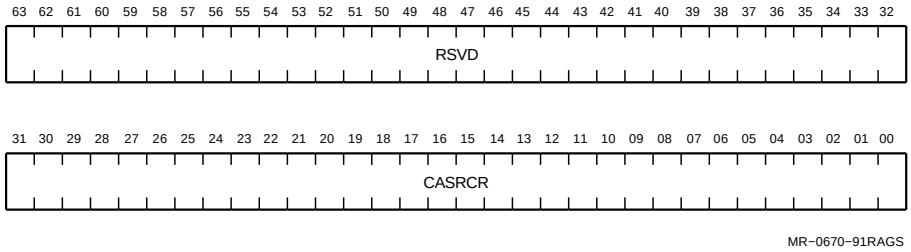
Table A–17 AMFQIR Bit Description

Bit	Name	Description
<63:32>	RSVD	Reserved (0)
<31:29>	SBZ	Should be zero (0)
<28:00>	PA	Physical address - Bits <33:05> of the new stopper carrier.

Channel/Adapter Status Release Control Register (CASRCR) - BB + B0

Writing to this register returns control of CASR to the adapter after the port driver has read CASR in response to an interrupt.

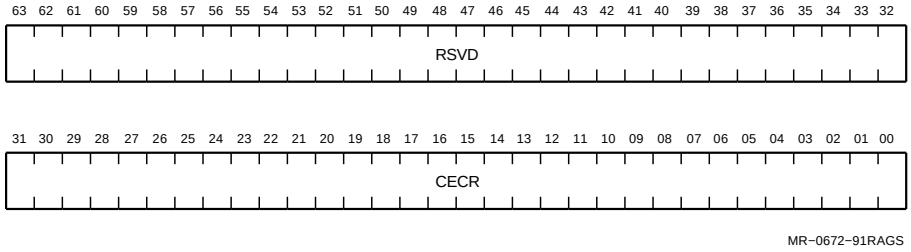
Figure A–15 Channel/Adapter Status Release Control Register (CASRCR)



Channel Enable Control Register (CECR) - BB + B8

When written to, this register enables the port driver to place the channel into the enabled state from the disabled state.

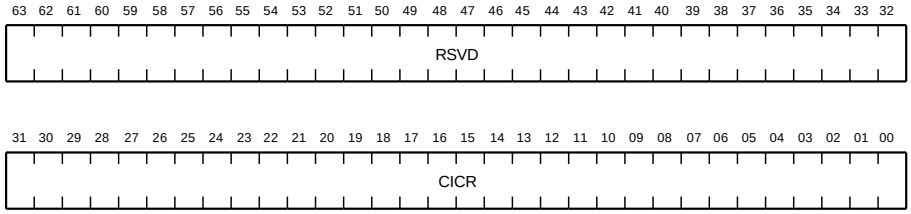
Figure A–16 Channel Enable Control Register (CECR)



Channel Initialize Control Register (CICR) - BB + C0

Writing to this register initializes the channel.

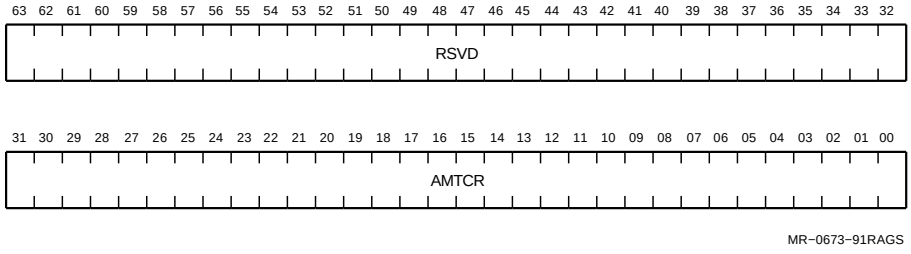
Figure A–17 Port Initialize Control Register (CICR)



Adapter Maintenance/Sanity Timer Control Register (AMTCR) - BB + C8

Writing to this register forces the maintenance/sanity time to reset its expiration time.

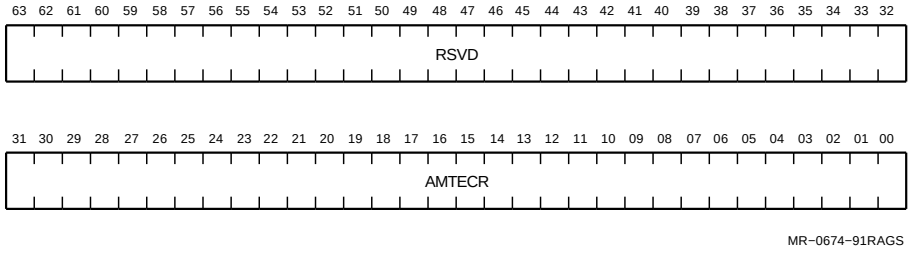
Figure A–18 Adapter Maintenance/Sanity Timer Control Register (AMTCR)



Adapter Maintenance/Sanity Timer Expiration Control Register (AMTECR) - BB + D0

Writing to this register forces a Maintenance/Sanity Timer Expiration (STE) interrupt.

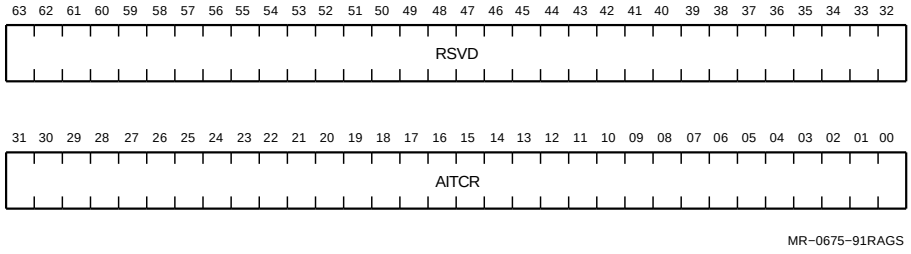
Figure A–19 Adapter Maintenance/Sanity Timer Expiration Control Register (AMTECR)



Adapter Interrupt Holdoff Timer Control Register (AITCR) - BB + D8

Writing to this register forces the interrupt holdoff timer to reset its expiration time.

Figure A–20 Adapter Interrupt Holdoff Timer Control Register (AITCR)



Adapter Maintenance Control/Status Register (AMCSR) - BB + E0

Contains interrupt and adapter initialization control bits.

Figure A–21 Adapter Maintenance Control/Status Register (AMCSR)

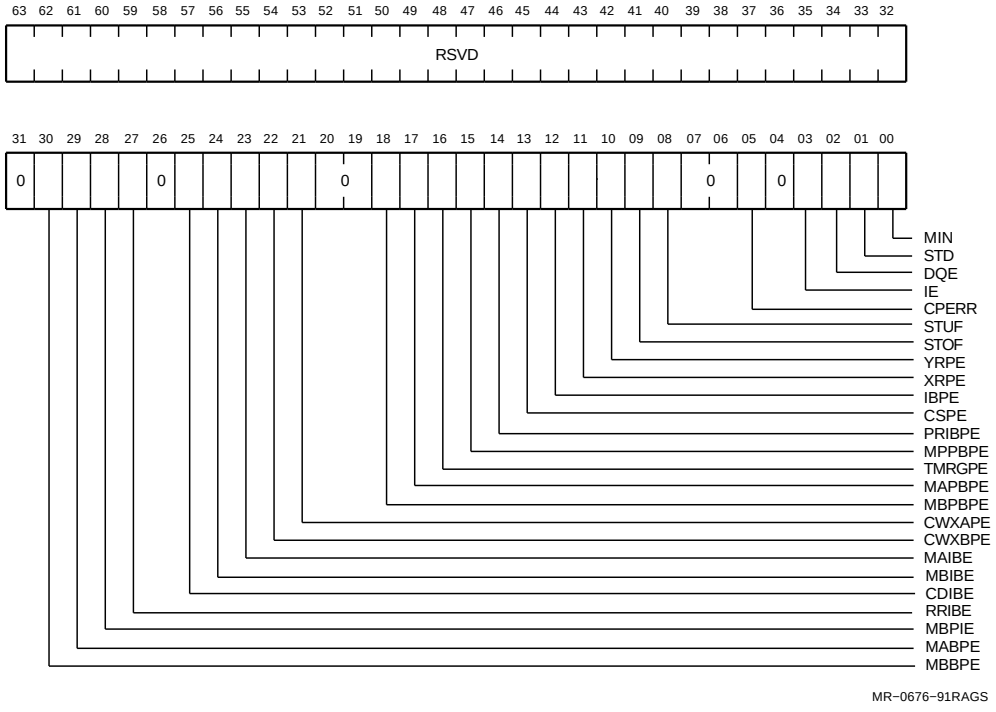


Table A–18 AMCSR Bit Description

Bit	Name	Description
<63:32>	RSVD	Reserved (0)
<31>	—	Zero (0)
<30>	MBBPE	Mover B byte parity error
<29>	MABPE	Mover A byte parity error
<28>	MBPIE	Mover B detected a PB_IB parity error during a packet buffer read operation.
<27>	RRIBE	Responder register PB_IB parity error during a register write operation.
<26>	—	Zero (0)
<25>	CDIBE	Commander detected a PB_IB parity error during a register write operation.
<24>	MBIBE	Mover B detected a PB_IB parity error during a register write operation.
<23>	MAIBE	Mover A detected a PB_IB parity error during a register write operation.
<22>	CWXBPE	CWIN transmit path B parity error
<21>	CWXAPE	CWIN transmit path A parity error
<20:19>	—	Zero (0)
<18>	MBPBPE	Mover B packet buffer read parity error
<17>	MAPBPE	Mover A packet buffer write parity error
<16>	TMRGPE	TMOV register read parity error
<15>	MPPBPE	MCDP packet buffer read parity error
<14>	PRIBPE	PORT_IB receive parity error
<13>	CSPE	Control store parity error
<12>	IBPE	Internal bus parity error
<11>	XRPE	X register parity error
<10>	YRPE	Y register parity error
<09>	STOF	Microstack overflow
<08>	STUF	Microstack underflow
<07:06>	—	Zero (0)

(continued on next page)

Table A–18 (Cont.) AMCSR Bit Description

Bit	Name	Description
<05>	CPERR	MCDP microprocessor error
<04>	—	Zero (0)
<03>	IE	Interrupt enable
<02>	DQE	Datagram free queue exhausted interrupt enable
<01>	STD	Sanity timer disable
<00>	MIN	Maintenance initialization - When set, clears all hardware states, including errors, and puts the port in the uninitialized state *without* copying microcode from FLASH PROM to RAM or executing self-test.

B
Boot Errors

This appendix provides information to help you troubleshoot boot errors. The information is valid for errors encountered while trying to:

- Boot ULTRIX through the CITCA
- Run the CITCA loopback test (boot *slot*/l)
- Run the CI node configuration test (boot *slot*/c)

B.1 What to Do

If the system won't boot, use the following list or use Table B–1 to determine the source of the problem.

1. Verify the syntax of the boot command. See Section 4.4.2.
2. Verify the system configuration. See Section 4.3.2.
3. Run TCAST to verify the CITCA operation. See Section 4.1.
4. Verify that the CITCA switches are set correctly and that the settings don't conflict with other nodes in the CI environment. See Section 2.2.
5. Verify the operation and connections of the CI cables. See Section 4.3.1.
6. Verify that the HSC controller is operational and on-line.
7. Verify that the boot device is operational, on-line and is port-enabled.
8. Verify that the host system passes self-test.
9. Verify the boot image on the boot device.

B.2 Error Codes

When the system encounters an error while trying to use the CITCA to execute the boot command, the boot microcode provides an error/status code to the console terminal. As shown in Example B–1 the error/status code is displayed on the last line of the console output.

Table B–1 describes the CITCA boot error/status codes.

Example B–1 Console Output of a Boot Failure

```
>>boot
[TCA Slot Number = 0, Mips Address = be000000]
[TCA Boot Code Version: V1]
[Boot: CI Number is 2]
[Boot: MSCP Unit Number is 0]
[Functional ucode version is : ad]
[TCA 1..[Functional ucode version is : ad]
[TCA 1..2..3..error/status code is: 9
```

Table B–1 Boot Error/Status Codes

Code	Description
1	User entered HELP command.
2	TCAST failed.
3	NOBOOT failed (syntax error).
4	Wrong device type.
5	Uninitialized state transition failed.
6	Enabled state transition failed.
7	Free queue empty.
8	Packet status error.
9	No CI path/no response from CI path.
A	Bad MSCP unit number.
B	Unsupported buffer size.
C	Disk unit number over maximum allowed (255).

(continued on next page)

Table B–1 (Cont.) Boot Error/Status Codes

Code	Description
D	Timeout waiting for a response from a CI device.
E	I/O request too large.
F	Wrong boot block.
10	Wrong CI node.
11	Wrong boot syntax.
12	Received out-of-sequence packets from CI device.
13	HSC controller failed.
14	Disk unit failed to be opened or disk unit not found.
15	MSCP read failed.
16	MSCP write failed.
17	MSCP unit not on-line.
18	Host memory system error.
19	Invalid destination port.
1A	Unable to set storage server characteristics.
1B	Console input error.
1C	Wrong or unsupported packet type.
1D	Virtual circuit not open or illegal state.
1E	Queue insertion failed.
1F	Port status error.
20	SCS connection failed.
21	Data buffer not found.
22	Queue removal failed.
23	Illegal SCS.
24	Error receiving RCV START.
25	Error receiving RCV STOP.
26	Error receiving unknown PPD.
27	Error receiving SETCHNL.[1]

[1]This error code includes a numerical-order listing of the nodes that the CITCA communicated with successfully. The failing node may be determined by identifying the next-higher node number that should follow the last node in the listing.

(continued on next page)

Table B–1 (Cont.) Boot Error/Status Codes

Code	Description
28	Error receiving SETCKT.
29	Error receiving illegal packets.
30	Bad datagram received.
31	Loopback both paths failed.
32	Bad message received.

C
Part Numbers

The following list includes the part numbers for the field replaceable units (FRUs) that are associated with the removal and replacement of the CITCA option.

Table C–1 CITCA Part Numbers

FRU	Part Number
CITCA module	54-20253-01
Cables, 10 meters (4 required)	17-2090x-10
Cables, 20 meters (4 required)	17-2090x-20
CI attenuator/loopback connector	12-19907
1/4-inch screws (12 required)	90-00038-19
Blank I/O panels (3)	74-41143-05

Index

V