Troubleshooting Procedures

HMM-156-A
CRAY T3E Series Systems
Last Modified: January 1998

Record of Revision

August 1997

Original printing.

Revision 0.1: October 1997

This revision corrects the cit command.

Revision A: January 1998

This revision adds a table that lists system serial numbers, describes how `t3e_hdw_boot` achieves the fast-band torus, how to boot the UNICOS/mk operating system in degraded mode, and how to use the sysmon interface to access errpt information. This revision also updates the screen snaps so that they are current with SWS-ION 3.7 system release.

Introduction

This document provides a troubleshooting strategy for the CRAY T3E™ system and references to detailed information that may help you troubleshoot the system. You can access most of the documents that this document refers to at the Service Publications and Training Web page: *http://servinfo.cray.com*.

Note: This document does not include the troubleshooting strategy for the SIO; for information about the SIO troubleshooting strategy, refer to the *SIO Troubleshooting Guide*, publication number HMM-204-0.

Table 1 lists the CRAY T3E serial numbers and whether the corresponding machine types are air cooled (AC) or liquid cooled (LC), their microprocessor speeds, and the marketing names.

Table 1. Description of the CRAY T3E Serial Numbers

Serial Numbers	Machine Type	Microprocessor Speed	Marketing Name
63*xx*	LC	300 MHz	CRAY T3E or CRAY T3E-600
65*xx*	AC	300 MHz	CRAY T3E or CRAY T3E-600
66*xx*	AC	450 MHz	CRAY T3E-900
67*xx*	LC	450 MHz	CRAY T3E-900
68*xx*	AC	600 MHz	CRAY T3E-1200
69*xx*	LC	600 MHz	CRAY T3E-1200

Note: Any cabinet can contain any microprocessor speed. For example, you can upgrade a 63*xx* liquid-cooled cabinet to contain modules with microprocessors that run at 600 MHz.

Offline Troubleshooting Strategy

Figure 1 and Figure 2 describe how to troubleshoot a CRAY T3E system failure in an offline environment. This strategy is based on the following assumptions:

- The system workstation (SWS) is functional
- The software is installed correctly

The subsections following the flowchart describe the troubleshooting strategy.

Figure 1. Troubleshooting Strategy in an Offline Environment (Part 1 of 2)

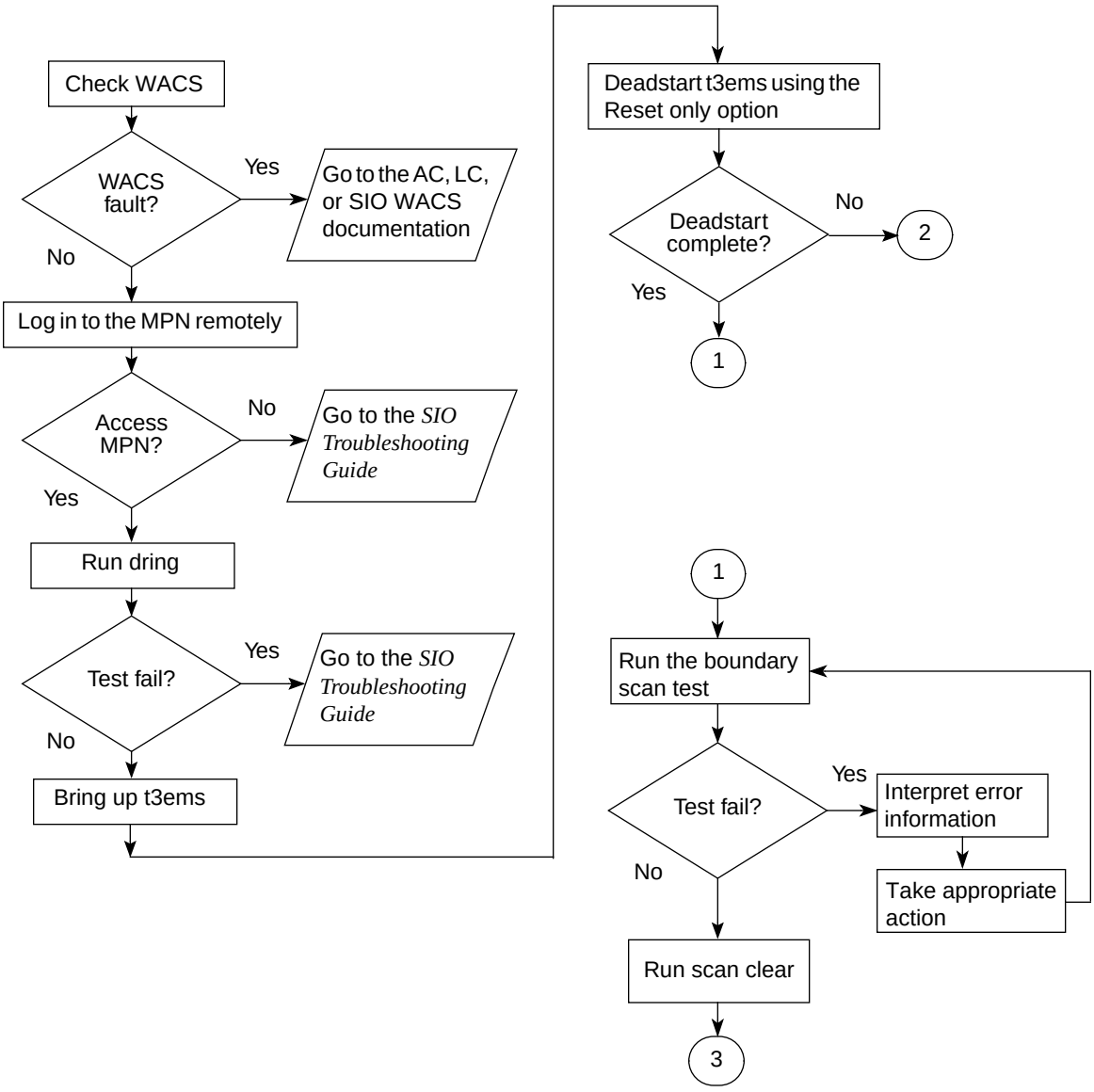
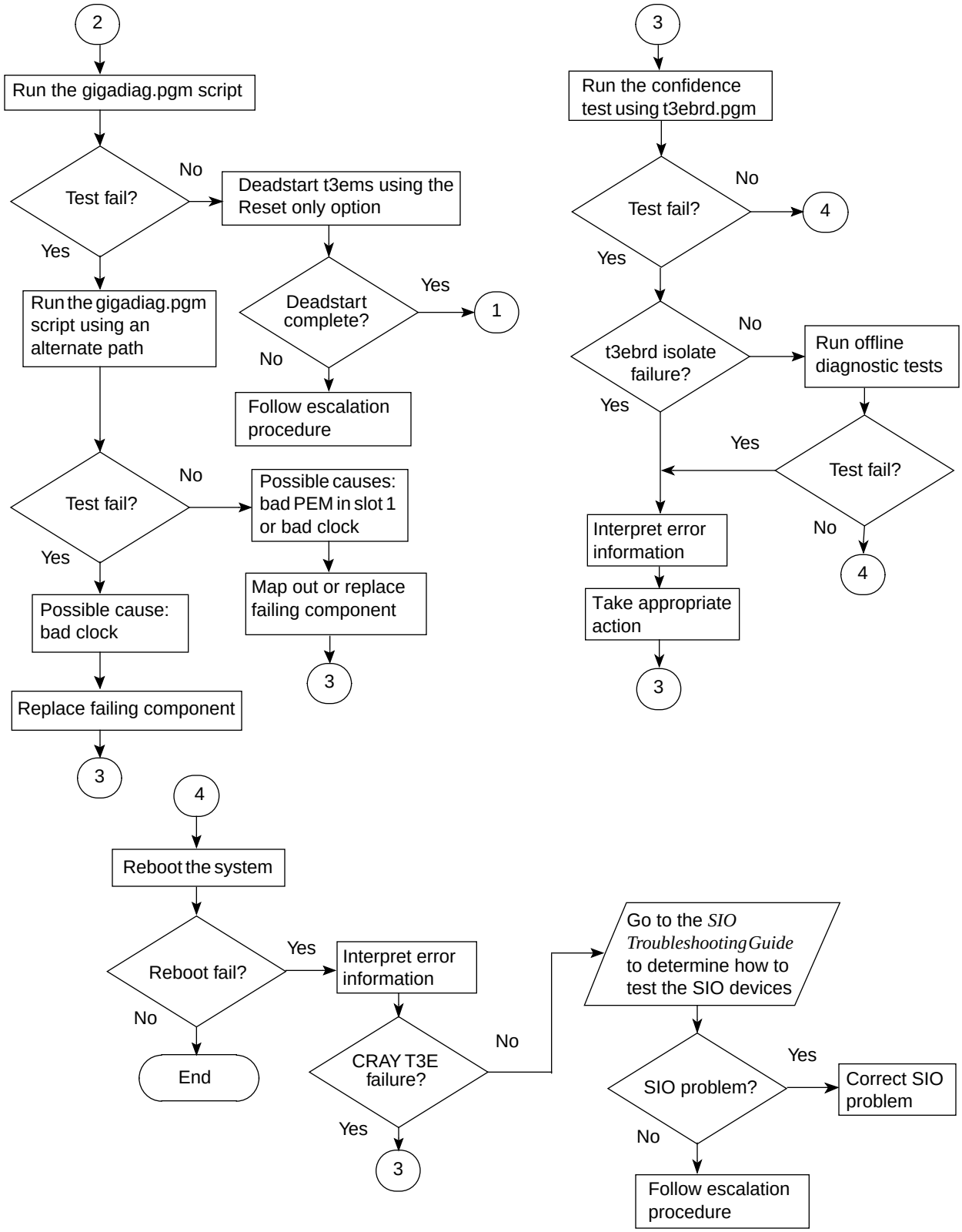


Figure 2. Troubleshooting Strategy in an Offline Environment (Part 2 of 2)



Check WACS

Refer to the *CRAY T3E AC Power, Control, and Electromechanical Troubleshooting* document, publication number HMM-169-A, or the *CRAY T3E LC Power, Control, and Electromechanical Troubleshooting* document, publication number HMM-176-0, for the WACS check procedure.

Log in to the MPN Remotely

To log in to the MPN remotely, type the following command line:

```
rlogin [mpn name]
```

The topology file (/opt/config) contains the device names.

When you cannot access the MPN, refer to the *SIO Troubleshooting Guide,* publication number HMM-204-0.

Run dring

To run dring, type the following command line in an SWS window:

```
dring -a initnodeclient
```

Note: The dring utility is located in the /opt/CYRIops/bin directory.

To clear a specific GigaRing™ channel, type the following command:

```
dring -a initnodeclient -r [ringname]
```

To initialize a specific node, type the following command:

```
dring -a initnodeclient -n [nodename] -r [ringname]
```

Note: When the initnodeclient action cannot complete, dring performs the isolate action. The isolate action includes the topology and continuity actions.

To isolate a fault on the GigaRing channel, type the following command:

```
dring -a isolate -r [ringname]
```

When the GigaRing channel fails, refer to the *SIO Troubleshooting Guide,* publication number HMM-204-0.

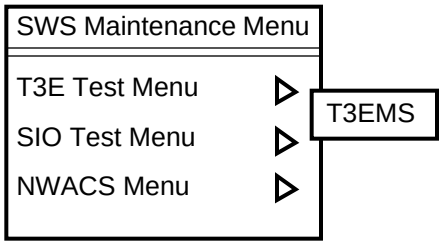
Bring up t3ems

To initiate the t3ems diagnostic interface in a common desktop environment (CDE), perform one of the following procedures:

1. Go to the root window and press the middle mouse button.

   Select T3EMS from the T3E Test Menu (refer to Figure 3).

   Figure 3. Selecting t3ems from a Menu

   or

2. Open a terminal window and type the following syntax:

   ```
   t3ems &
   ```

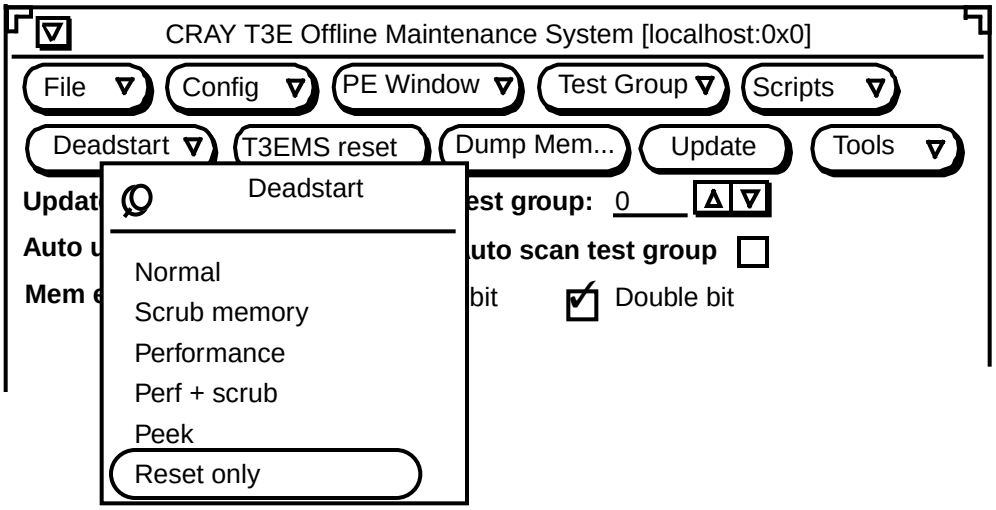
   Note: t3ems is located in the /opt/CYRIdiag/t3e/bin directory.

   For more information about how to initiate t3ems, refer to the *Offline Diagnostic Maintenance System* document, publication number HDM-135-B.

Deadstart t3ems Using the Reset Only Option

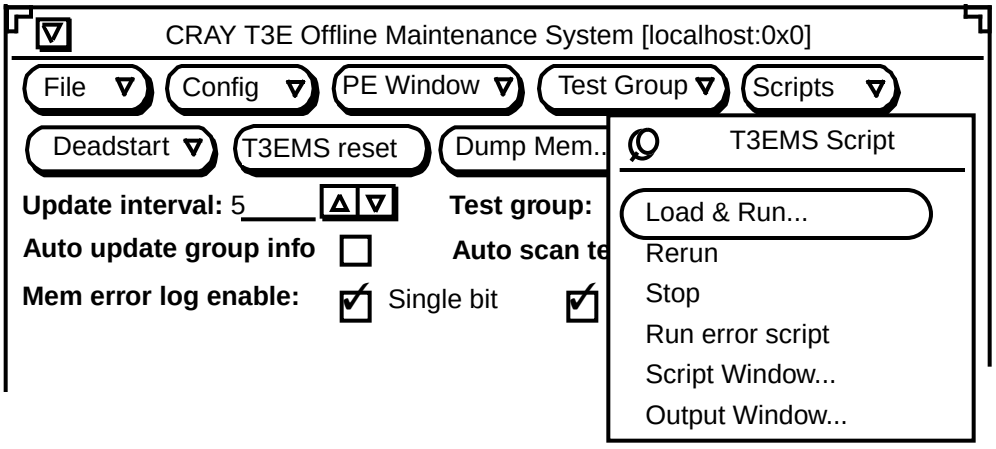
To deadstart t3ems using the Reset only option, select the Reset only option from the Deadstart menu (refer to Figure 4). This option initializes the connection to the boot PE without loading the PAL code.

Figure 4. Reset only Option



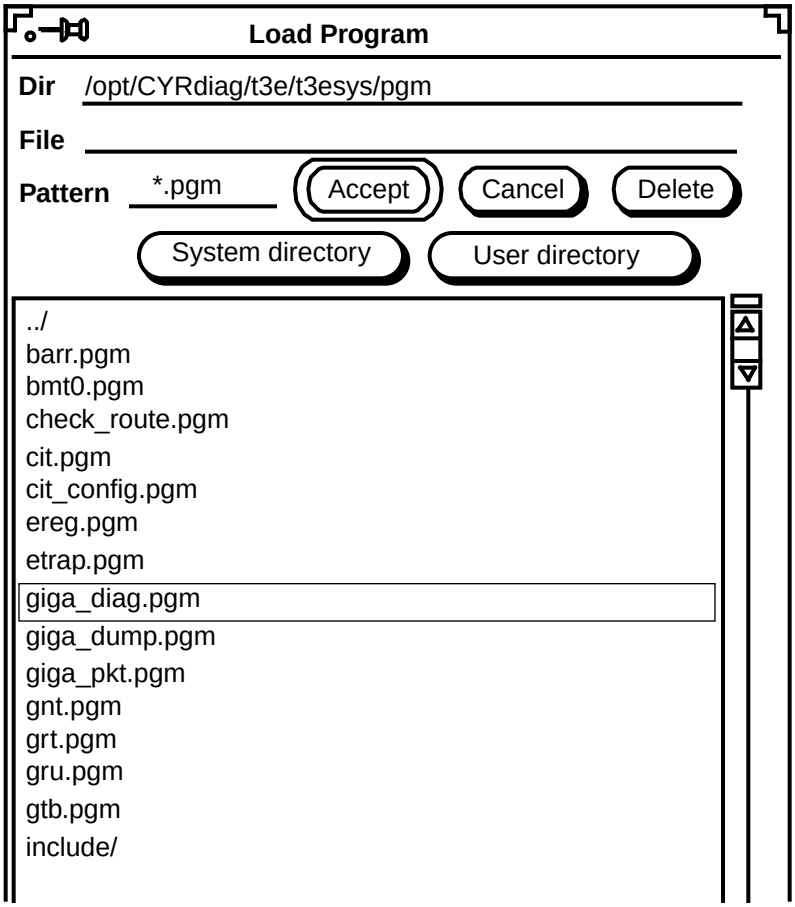
When you are not able to deadstart the CRAY T3E system with the Reset only option, run the gigadiag.pgm script. To initiate the gigadiag.pgm script, select the Load & Run option from the Scripts menu (refer to Figure 5).

Figure 5. Load & Run Option



Select the gigadiag.pgm script from the list and click on the Accept button (refer to Figure 6).

Figure 6. Script List

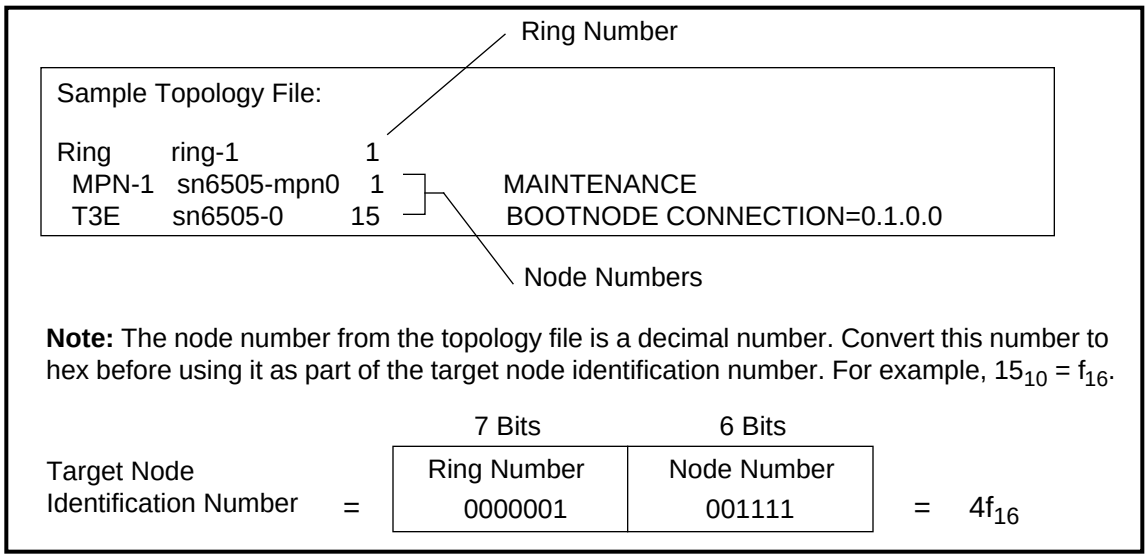


The giga_diag.pgm script prompts you to select sections to run. Once you make this selection, the script prompts you for a target node identification (ID) number. This number is the ring number and the node number from the topology file (refer to Figure 7). For systems that have more than one node on the GigaRing channel, the default number is the GigaRing node ID number of the boot node. This script also prompts you to select network routers to be tested and the number of passes to run.

Note: When you select the network router to test, you are also selecting the PE to test (the PE that is local to the selected network router). In addition, the network routers are on a printed circuit board (PCB) that connects to the GigaRing channel.

If the script does not fail, deadstart the CRAY T3E system using the Reset only option.

Figure 7. Determining the Target Node Identification Number



If the script fails and does not give adequate failure information, try the following procedures to help isolate the problem:

- Verify which network routers fail.

  To do this, restart the script with only one network router enabled. Repeat this step for all network routers.

- When only one network router fails, disable the network router.

- When all of the network routers fail and the system has other GigaRing channels, rerun the script on each GigaRing channel.

  To do this, change the boot host name, the boot node ID, and the slot number in the T3EMS System Configuration window to indicate the maintenance node that connects to a GigaRing channel and rerun the script.

  Note: Boundary scan runs only when the system uses the GigaRing channel that connects to cabinet 0, slot 1.

- When the script fails on all of the GigaRing channels, the clock may be failing.

- When the script passes on all of the GigaRing channels except for the first GigaRing channel, the module that contains the boot node (cabinet 0, slot 1) may be failing or the first GigaRing channel may be failing. Refer to the *SIO Troubleshooting Guide*, publication number HMM-204-0, to determine how to test the GigaRing channel.

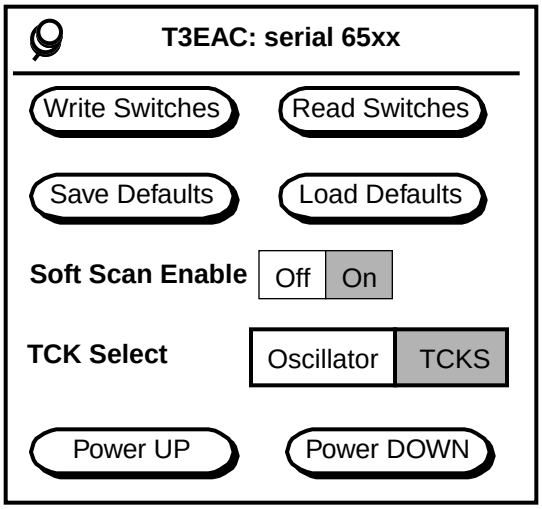
Run the Boundary Scan Test

Before you run the boundary scan test, enable the boundary scan hardware switch and the boundary scan software switch.

Note: For a liquid-cooled system, the boundary scan hardware switch is on the front side of chassis 0, above the WACS display. For an air-cooled system, the boundary scan hardware switch is on the rear side of chassis 0, above the clock module (slot 6).

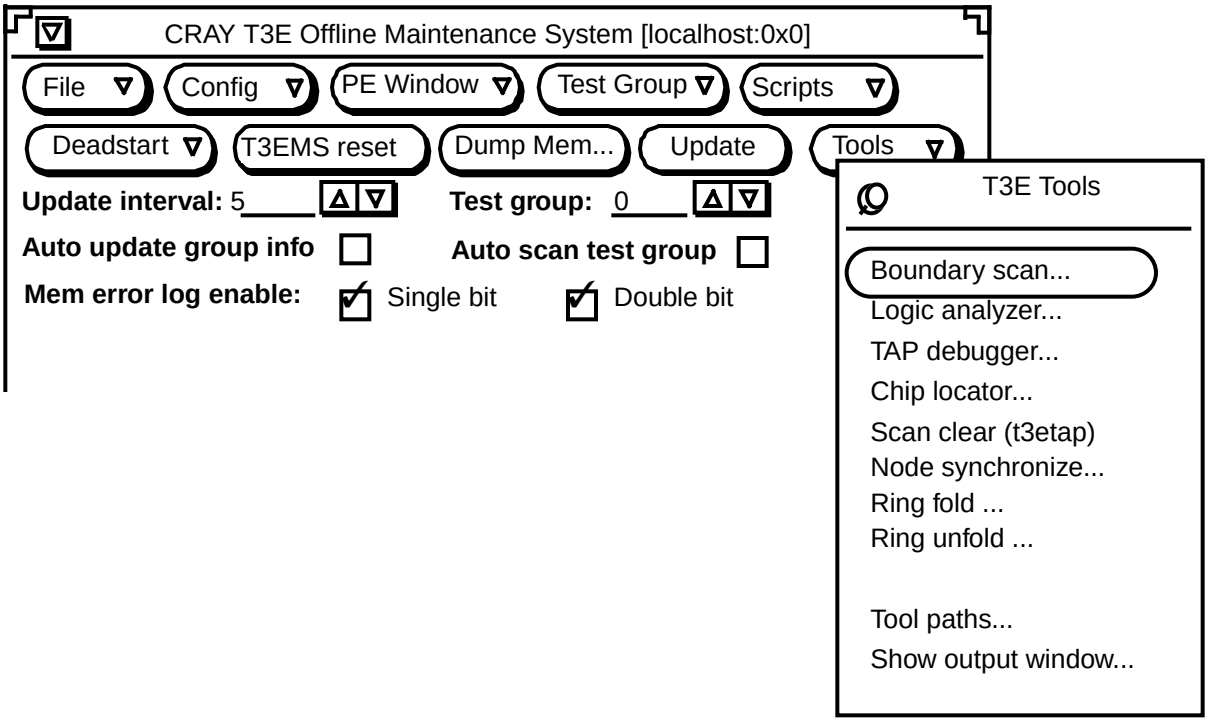
To enable the software switch, bring up the nwacs display, select On for the soft scan enable and click on the Write Switches button (refer to Figure 8).

Figure 8. Boundary Scan Software Switch (Soft Scan Enable)



To initiate the boundary scan test, select the Boundary scan option from the Tools menu (refer to Figure 9). For more information about how to initiate the boundary scan test, refer to the *Clock and Boundary Scan* document, publication number HMM-360-A.

Figure 9. Boundary scan Option



When the Boundary Scan Tool Options window appears on the screen, select an error notation option (logical, physical, or resource). When you want to map out the failing component, select the resource option. When you want to repair the system, select the physical option.

After you select the error notation option, click Start to start the boundary scan test. t3ems directs the status and error information to the log window.

TAP Test Failure

The boundary scan test displays the following error information when the boundary scan hardware switch or the boundary scan software switch is not enabled. This type of failure could also be caused by an open wire between the top module (liquid-cooled system) or leftmost module (air-cooled system) in the chassis and the scan master, failing scan master logic, or failing logic in the top or leftmost module in the cabinet.

```
=== Begin boundary scan test ===

T3E Boundary-Scan Test - t3ebst version 2.0

Hardware: A/C system with 8 PEs

Attempting connection to GRING node 0x0F via MPN sn6505-mpn0:0470 done.

Performing TAP test:

    Chain 00...failed
    All 1's coming out of instruction register chain.
    Verify power and communications to module(s) in system.

Unable to continue due to TAP test errors.

=== End boundary scan test ===
```

TAP Test Vector = IR_TEST TDO Failure

When the boundary scan test detects a boundary scan chain test error, it calculates the physical location of the component that detected the error (refer to the following error information). For this example, the component that detected the error is an I option (physical name is AI0) that resides on the upper PCB in slot 1 of chassis 0.

Note: Even though there is only one PCB on an air-cooled module, it is referred to as the upper PCB.

```
=== Begin boundary scan test ===

T3E Boundary-Scan Test - t3ebst version 2.0

Hardware: A/C system with 8 PEs

Attempting connection to GRING node 0x0F via MPN sn6505-mpn0:0470 done.

Performing TAP test:

    Chain 00...failed
        Invalid data shifted out of instruction register chain.
        Miscompare @ board 001U chip I (physical chip name AI0)
        Check component and JTAG signals to/from component.

VECTOR=ACTUAL IR TDO DATA
84090842 10842108 42108421 08421084 2108421F FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF

Unable to continue due to TAP test errors.

=== End boundary scan test ===
```

Interconnect Test Failure

The boundary scan test displays the following error information when it detects an interconnect failure.

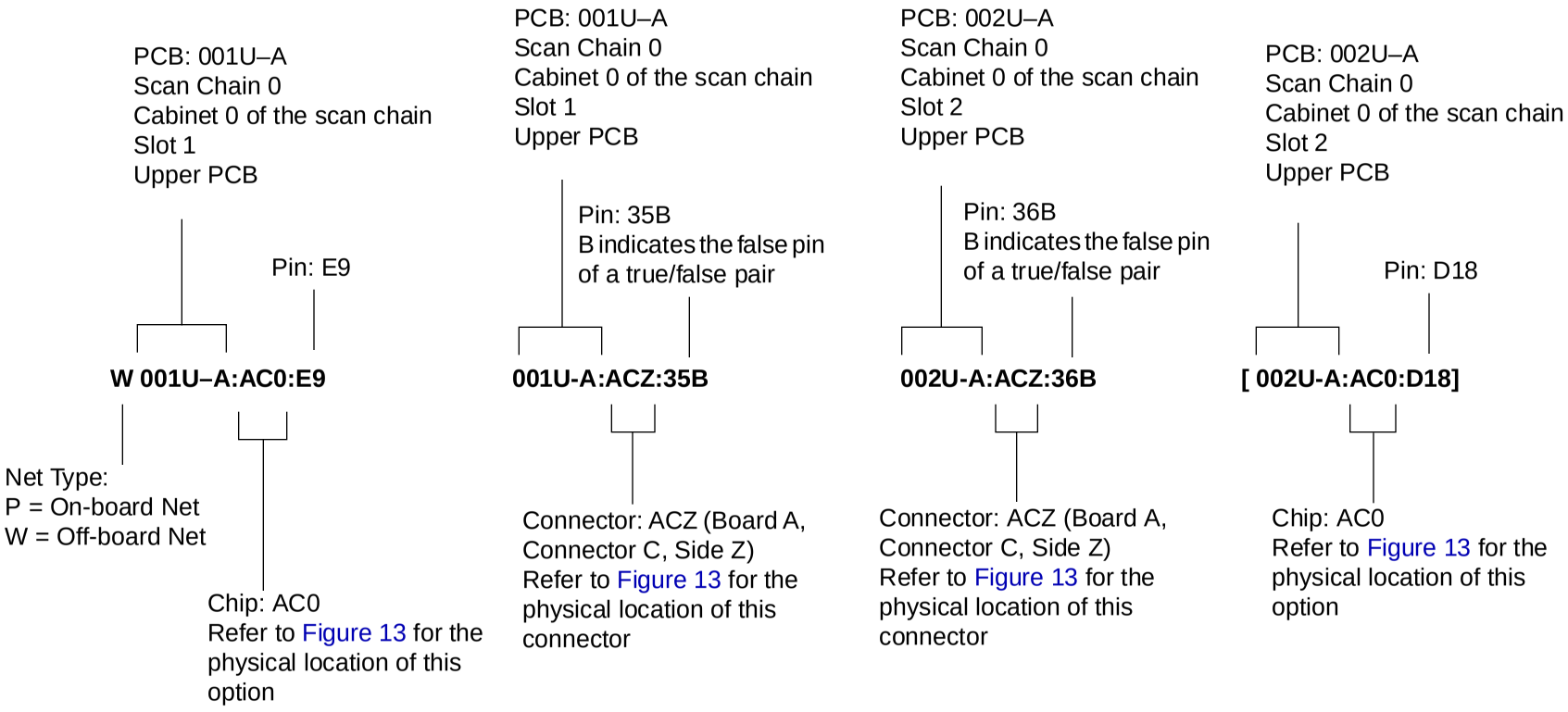
```
=== Begin boundary scan test ===

T3E Boundary-Scan Test - t3ebst version 2.0

Hardware: A/C system with 8 PEs

Attempting connection to GRING node 0x0F via MPN sn6505-mpn0:0470 done.

Performing TAP test:

        Chain 00...passed
                Detected from hardware: #chips=60 IR_length=310

Reading revisions from hardware and verifying configuration:
        Chain 00...done

        Boundary-scan revisions from hardware (by scan chain):
        <GRC> = '+' if GigaRing chip in scan chain; ' ' if chip in bypass
        <BRD> = board on 'A' or 'B' side of coldplate
        <REV> = bits 2^3 - 2^0 of BSTREV (in hexadecimal)

              G B R
              R R E
        00    C D V
        - - - -  - - -
        002U  + A 0
        001U  + A 0
  Testing boundary-scan registers:
        Chain 00...passed
```

```
Performing boundary scan interconnect test (pass 1 of 1):

(30 patterns will be applied)

Pattern 00...done
Pattern 01...done
Pattern 02...done
Pattern 03...done
Pattern 04...done
Pattern 05...done
Pattern 06...done
Pattern 07...done
Pattern 08...done
Pattern 09...done
Pattern 10...done
Pattern 11...done
Pattern 12...done
Pattern 13...done
Pattern 14...done
Pattern 15...done
Pattern 16...done
Pattern 17...done
Pattern 18...done
Pattern 19...done
Pattern 20...done
Pattern 21...done
Pattern 22...done
Pattern 23...done
Pattern 24...done
Pattern 25...done
Pattern 26...done
Pattern 27...done
Pattern 28...done
Pattern 29...done
```

Following the pattern information, the boundary scan test lists the failure information. The boundary scan test logs the failing nets using logical names, physical names (refer to Figure 10 and Figure 11), or resource information (refer to Figure 12). You select which format the boundary scan test uses to display the failing nets by clicking on the desired error notation in the T3EMS Scan Tool Options window.

Figure 10. Physical Boundary Scan Failure Information for an Air-cooled System (Off-board Net)

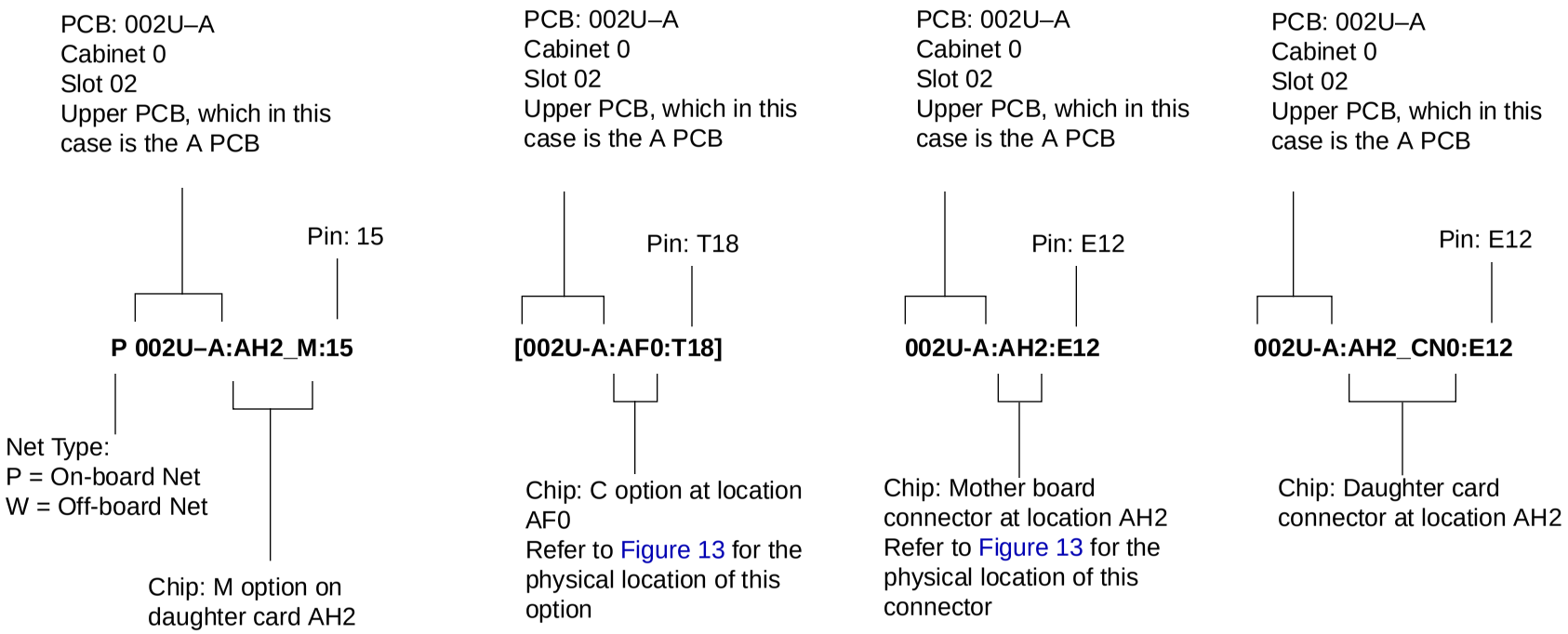
CHAIN POSITION	EXPECT DATA	ACTUAL DATA	NET INFORMATION (net type: P=on-board net, W=off-board net) Format of each node: <board>:<chip>:<pin>
00–11753	10101001100110100101100110 0110	11111111111111111111111111111111	W 001U–A:AC0:E9 001U-A:ACZ:35B 002U-A:ACZ:36B [002U-A:AC0:D18]
00–11754	10101001100110101010011001 0110	00000000000000000000000000000000	W 001U–A:AC0:F10 001U-A:ACZ:35 002U-A:ACZ:36 [002U-A:AC0:C19]
00–11755	10101001100110100101011010 1010	11111111111111111111111111111111	W 001U–A:AC0:D8 001U-A:ACZ:37B 002U-A:ACZ:38B [002U-A:AC0:A21]
00–11756	10101001100110101010010101 1010	00000000000000000000000000000000	W 001U–A:AC0:C7 001U-A:ACZ:37 002U-A:ACZ:38 [002U-A:AC0:B20]



Note: Brackets surround the component that detected the error; the component detects an error when it inputs the signal.

Figure 11. Physical Boundary Scan Failure Information for a Liquid-cooled System (On-board Net)

```
CHAIN     EXPECT                                ACTUAL                                NET INFORMATION (net type: P=on-board net, W=off-board net)
POSITION  DATA                                  DATA                                  Format of each node: <board>:<chip>:<pin>

00–006560 10010101100110010101100110101O  11111111111111111111111111111111  P 002U–A:AH2_M:15 [002U-A:AF0:T18] 002U-A:AH2:E12 002U-A:AH2_CN0:E12
                                                ^^ ^ ^  ^^  ^^ ^ ^  ^^  ^ ^ ^

t3ebst completed with 1 failures.

=== End boundary scan test ===
```

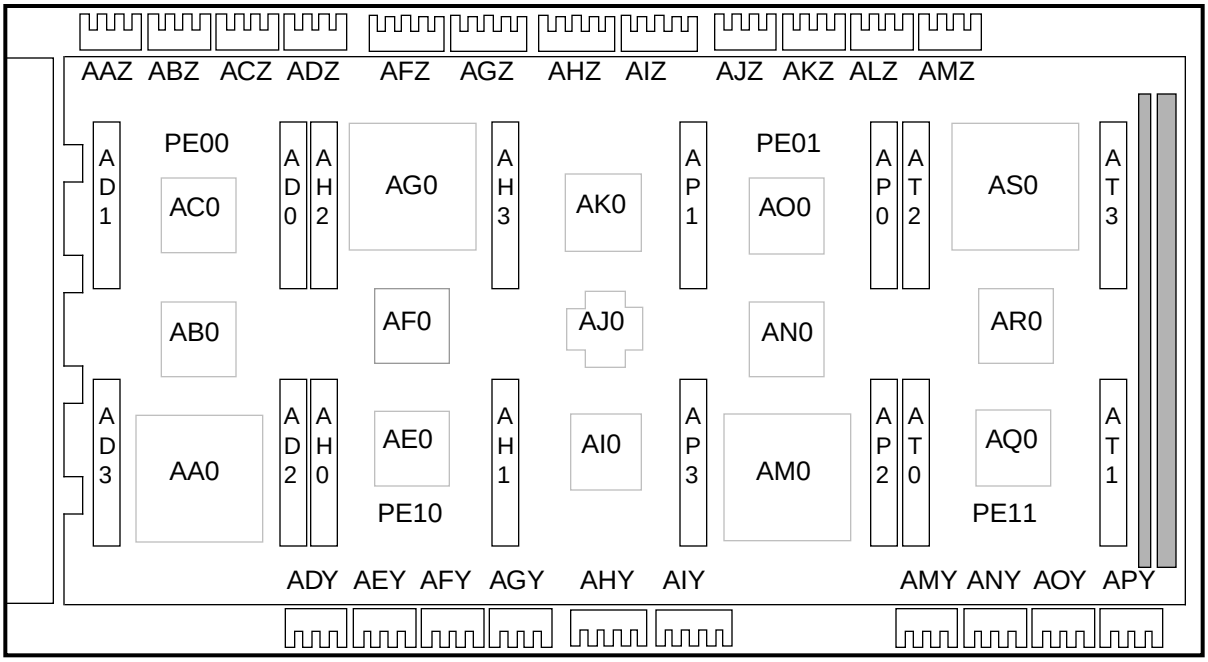


Note: Brackets surround the component that detected the error; the component detects an error when it inputs the signal.

Figure 12. Resource Boundary Scan Failure Information

```
Boundary scan errors detected!!!

Raw pin number          : 00-006560
Expected data           : 10010101100110010101100110101O
Actual data             : 11111111111111111111111111111111
FAILING BOARD           : CHASSIS = 0  SLOT = 2  BOARD = Upper(A)
FAILING NET NAME        : CHIP = N10M2_M      PIN = MOUT13
RESILIENCY REQUIREMENT  : Disable the PE port on PWHO 100

t3ebst completed with 1 failures.

=== End boundary scan test ===
```

Figure 13. Physical Option and Connector Locations

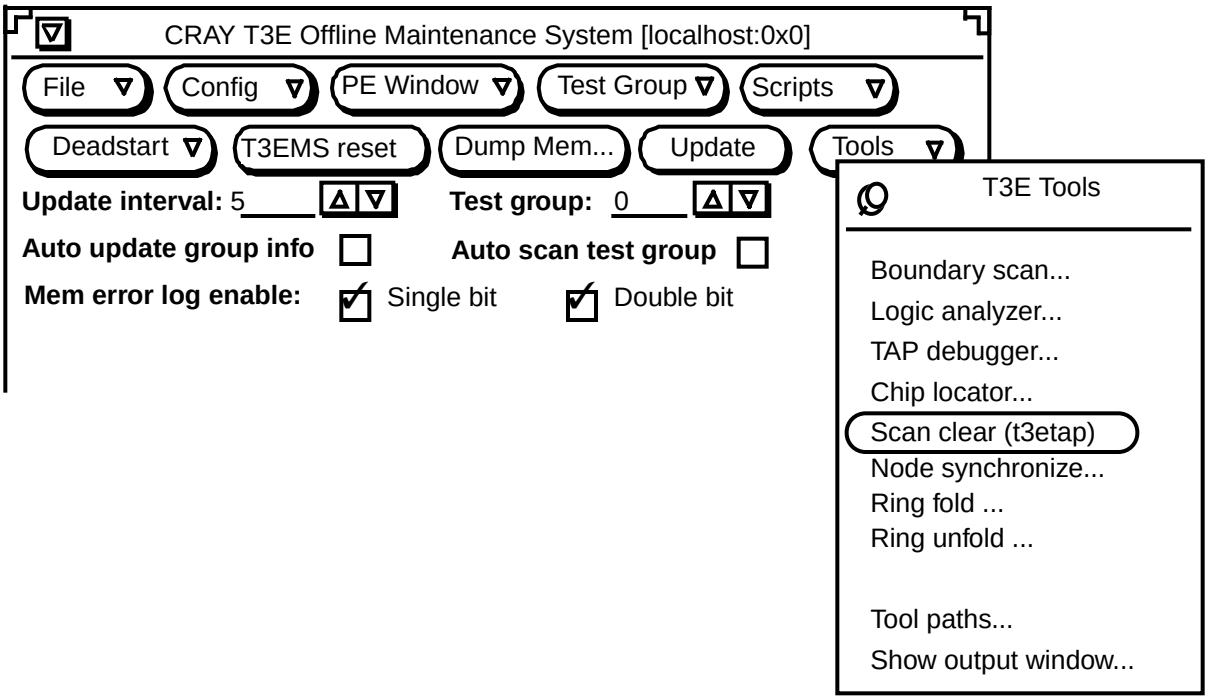


Note: This diagram represents the side of the PCB that is visible after it is mounted to the coldplate with the daughter cards removed. You cannot see the components with the gray, dotted outlines.

Run Scan Clear

To run the scan clear utility, select the Scan clear (t3etap) option from the Tools menu (refer to Figure 14).

Figure 14. Scan clear Option

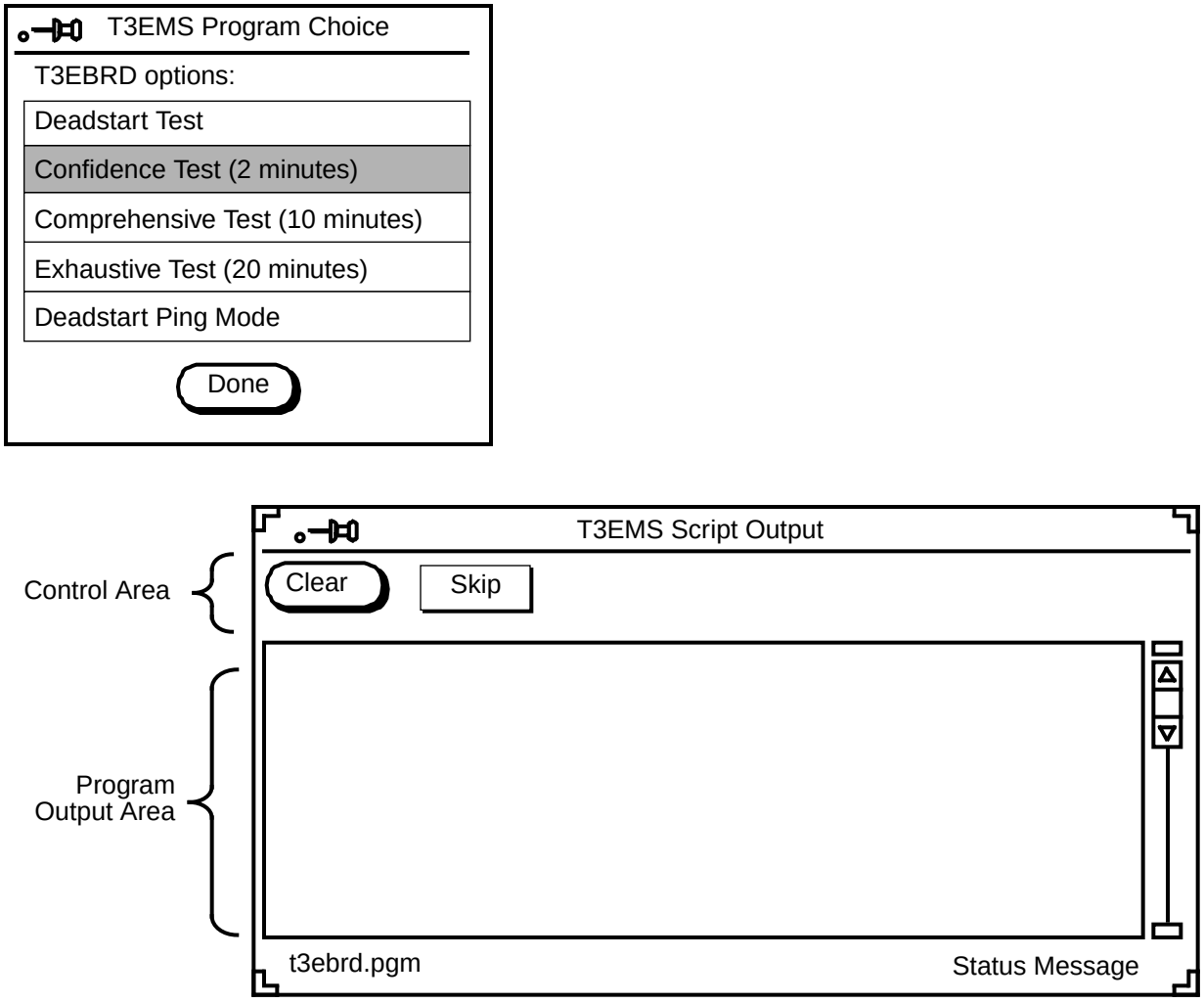


When the T3EMS Scan Tool Options window appears on the screen, click Start to start the Scan clear (t3etap) utility.

Run the t3ebrd.pgm Script

To run the boot resiliency diagnostic script, t3ebrd.pgm, select the Load & Run option from the Scripts menu. Select the t3ebrd.pgm script from the list and click on the Accept button. The windows in Figure 15 appear on the screen.

Figure 15. Windows that Appear after You Load t3ebrd.pgm



Click on the Confidence Test option from the T3EMS Program Choice window. The t3ebrd.pgm script runs the offline diagnostic tests listed in Table 2 and Table 3.

Table 2. Confidence Diagnostic Tests

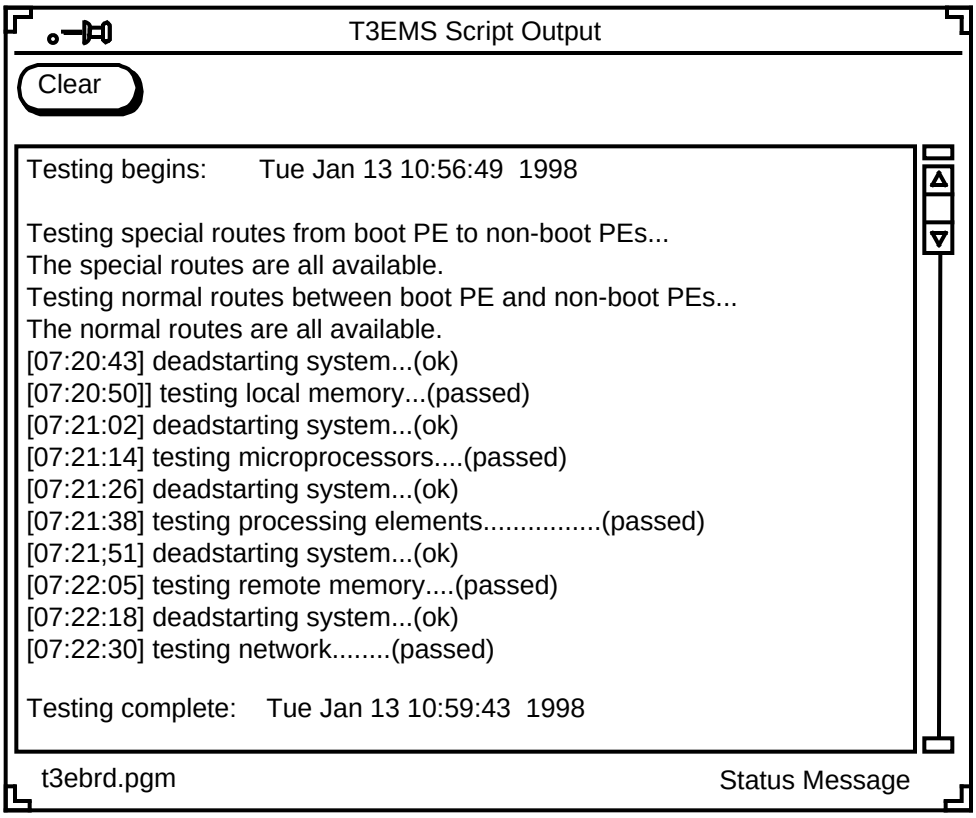
Area Tested	Diagnostic Test
Local memory	lmem
Microprocessor	peri
Processing element	rlo
Remote memory	rmem
Interconnect memory	rmem

Table 3. Exhaustive and Comprehensive Diagnostic Tests

Area Tested	Diagnostic Test
Local memory	lmem
Microprocessor	peri
Processing element	rlo
Remote memory	rmem
Interconnect memory	rmem
E registers	ereg

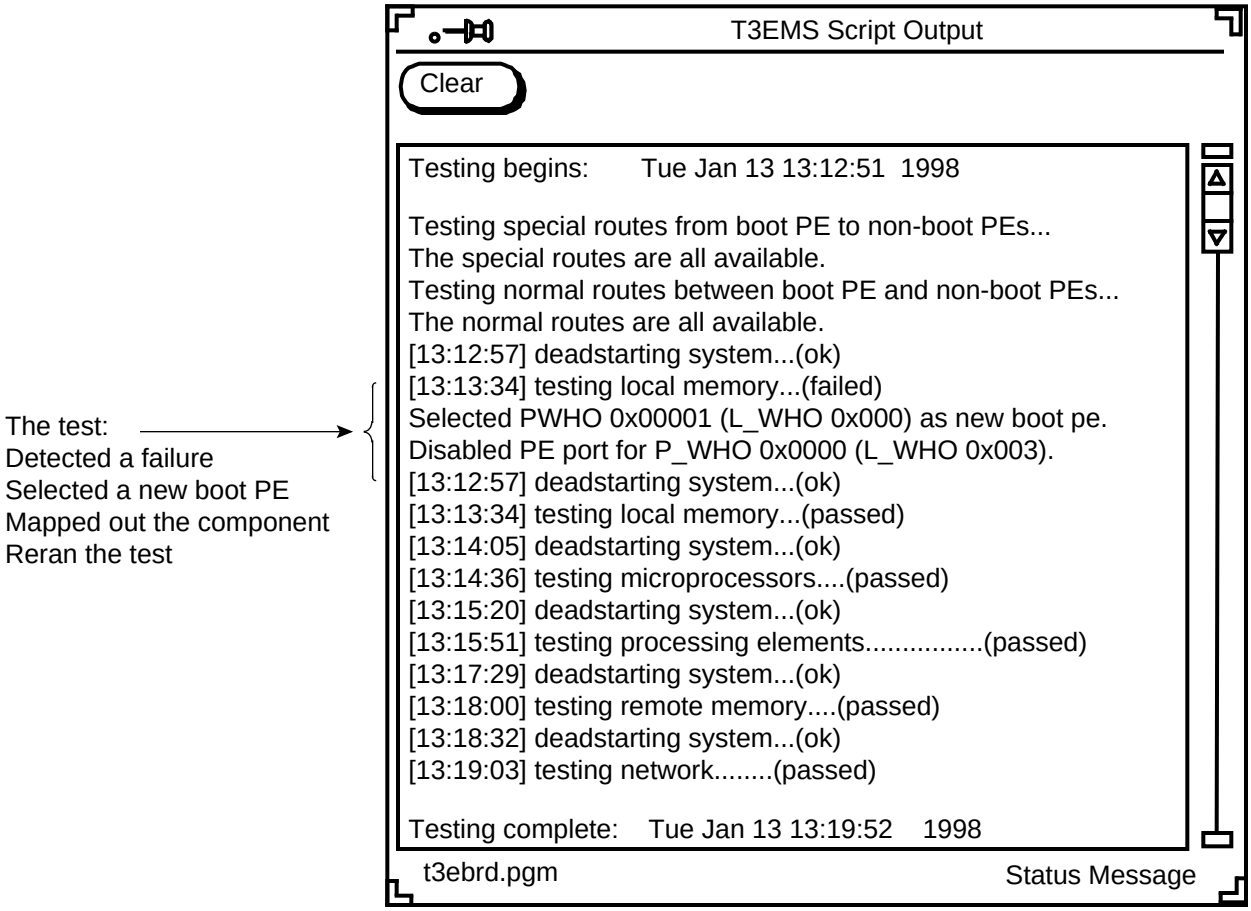
The t3ebrd.pgm script indicates the progress of the tests in the T3EMS Script Output window. Figure 16 shows the output of the confidence test when t3ebrd.pgm did not detect an error.

Figure 16. Output from t3ebrd.pgm When No Errors Occur



When the t3ebrd.pgm script detects an error, the script attempts to isolate the error and map out the failing component (refer to Figure 17).

Figure 17. t3ebrd.pgm Detects an Error

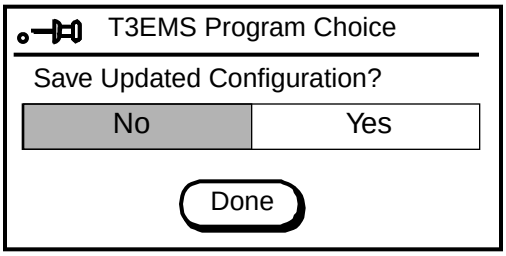


After the t3ebrd.pgm script maps out the failing component, t3ebrd.pgm reruns the failing diagnostic test to verify that the problem was corrected. When the diagnostic passes, t3ebrd.pgm continues to run the other diagnostic tests of the confidence test.

Note: t3ebrd.pgm creates a t3ebrd.log file in the directory /opt/CYRIdiag/t3e/dump, which contains the information that t3ebrd.pgm used to isolate the error. This file is an ASCII file; therefore, you can view it using the vi text editor.

When the confidence test completes and t3ebrd.pgm detects an error, the T3EMS Program Choice window appears on the screen (refer to Figure 18). This window prompts you to choose whether you want to save the updated configuration. When you save the updated configuration, t3ebrd.pgm writes the updated configuration to the t3e_config file.

Figure 18. T3EMS Program Choice Window for Reconfiguration



If t3ebrd.pgm does not isolate the problem, run the offline diagnostic test that failed during the confidence test (refer to the “Run Offline Diagnostic Tests” subsection).

Note: For additional error information, run the QUICKVERIFY mode of the mile.pgm script. This mode allows you to select which tests to run. The other modes, single and continuous, run all of the tests. For more information about the mile.pgm script, refer to the *Offline Diagnostics* document, publication number HDM-136-A.

Run Offline Diagnostic Tests

To load a diagnostic test, select the Load test - defaults option from the File menu. Select a test from the list and click on the Accept button. For more information on how to load and configure an offline diagnostic test, refer to the *Offline Maintenance* document, publication number HMM-158-B.

Table 4 lists the offline diagnostic tests that are available to troubleshoot the CRAY T3E system. Table 5 lists the utilities for the CRAY T3E system.

Table 4. Offline Diagnostic Tests

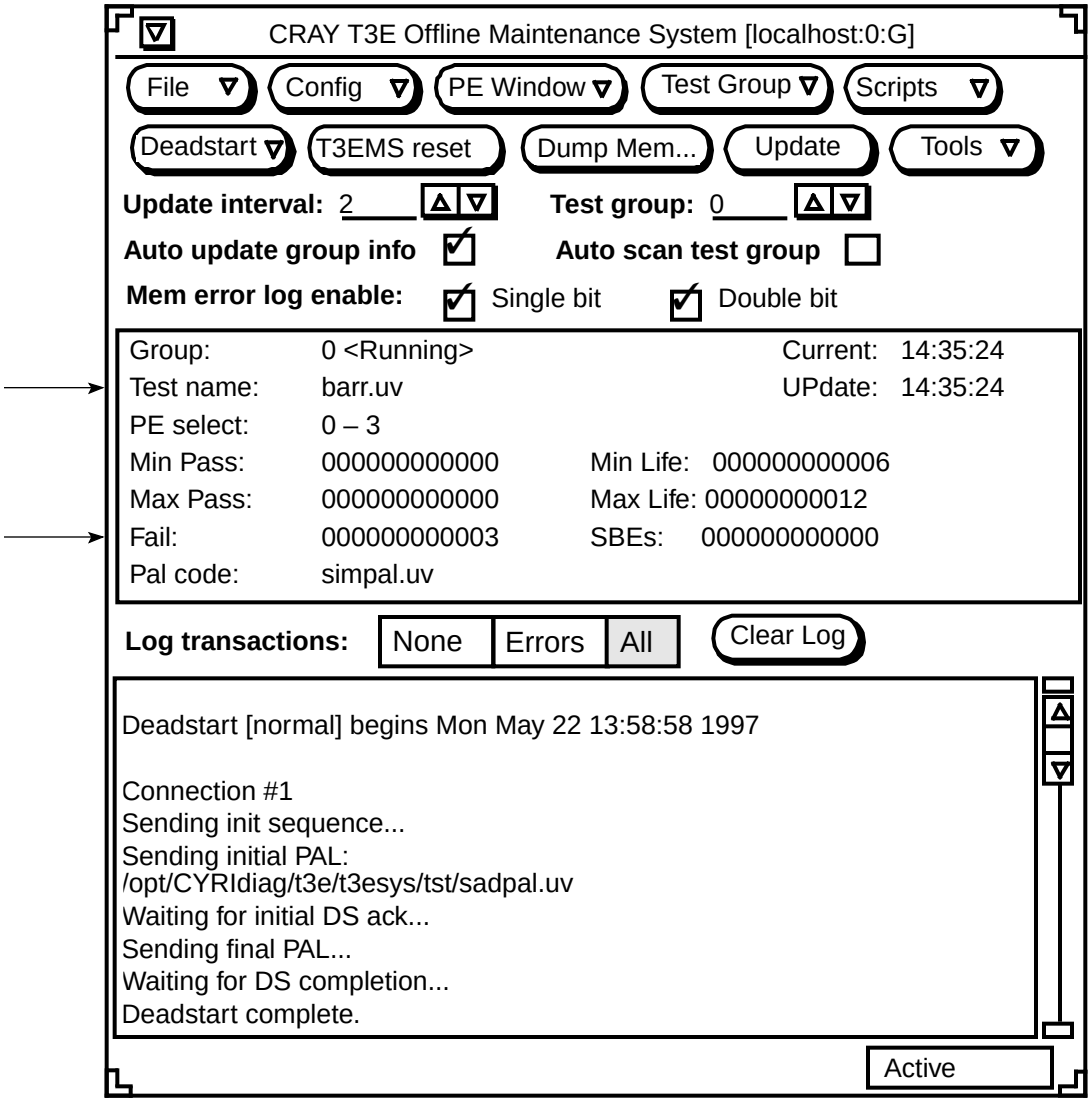
Test Name	Description
barr	Tests the barrier and eureka synchronization circuitry
cit	Tests the GigaRing client option (Use cit_cfg.pgm to execute this test)
ereg	Tests basic E-register operations
etrap	Tests the software-generated errors that are trapped by the C option
gnt	Tests the functionality of the GigaRing option
grt	Tests the functionality of the GigaRing channel and the client interfaces between the CRAY T3E system and the other Cray products that support the GigaRing channel
gtb	Tests the global address translation process
lmem	Tests local memory operations, SECDED, stream buffering, and cache coherence
luts	Tests the remote memory look-up tables
mqcw	Tests the message queue control word
peri	Tests the microprocessor's capability to execute random instruction sequences
pst	Tests the microprocessor stall timeout function and the PROD function
rchip	Tests the network router
rlo	Tests random local operations
rmem	Tests remote memory operations
segt	Tests the segment translation table
todt	Tests the time-of-day clock
xnet	Tests the interconnect network and the global E-register operations

Table 5. Offline Diagnostic Utilities

Utility Name	Description
bmt0.pgm	Tests the interconnects between the M option and the DRAM
check_route.pgm	Determines whether all routes are available and calculates which nodes need to be disabled to solve route problems
cit_config.pgm	Prompts the user for system parameters so that it can configure the cit test correctly
giga_diag.pgm	Tests the paths in and out of the deadstart GigaRing node
giga_dump.pgm	Dumps the contents of the memory-mapped registers that are located in the GigaRing option, the I options, and the R options
giga_pkt.pgm	Generates and executes a variety of packets
gru.pgm	Resets the I options and GigaRing options, dumps the memory-mapped registers of the I options and GigaRing options, and configures the memory-mapped registers of the GigaRing options
mfind	Finds a pattern in memory
mile.pgm	Executes a variety of CRAY T3E diagnostics (command buffer)
mpat	Patterns memory
query.pgm	Gathers and displays the following information about the PEs in a CRAY T3E system: option revisions, clock frequency, and daughter card configuration
r_err.pgm	Analyzes the R_ERR information that is imported from an ASCII file or read from the error information buffer after an offline diagnostic failure
router_init.pgm	Initializes the R options so that you can access all special registers, normal registers, and memory
sbe_dump.pgm	Formats and displays the single-bit error tables of the PEs that have logged single-bit and double-bit memory errors
t3echip	Identifies the physical location of a failing memory chip
t3ebrd.pgm	Tests the CRAY T3E hardware before booting the UNICOS/mk™ operating system

In the following example, the barr diagnostic test detects a failure (refer to Figure 19).

Figure 19. barr Diagnostic Test Logging a Fail Count



To determine the cause of the failure, run the barr.pgm error script. For information on how to load and run a script, refer to the *Offline Maintenance* document, publication number HMM-158-B.

The barr.pgm script brings up two windows (refer to Figure 20 and Figure 21). The T3EMS Program Choice window prompts you to select the type of output that the script displays in the Main Error Window for BARR. For example, the PE option displays the error information shown in Figure 22. The Summary option displays the error information shown in Figure 23.

Figure 20. T3EMS Program Choice for the barr Diagnostic Test

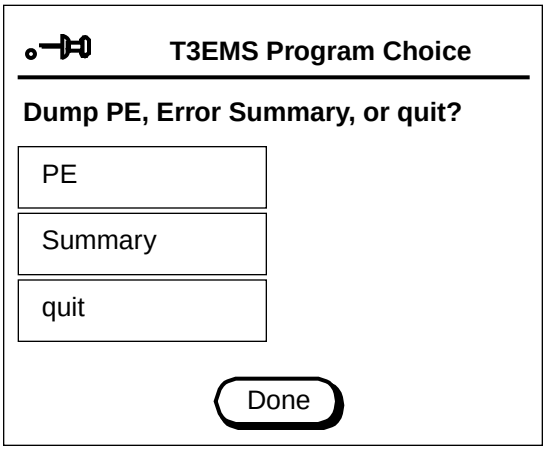


Figure 21. Output from the Error Script

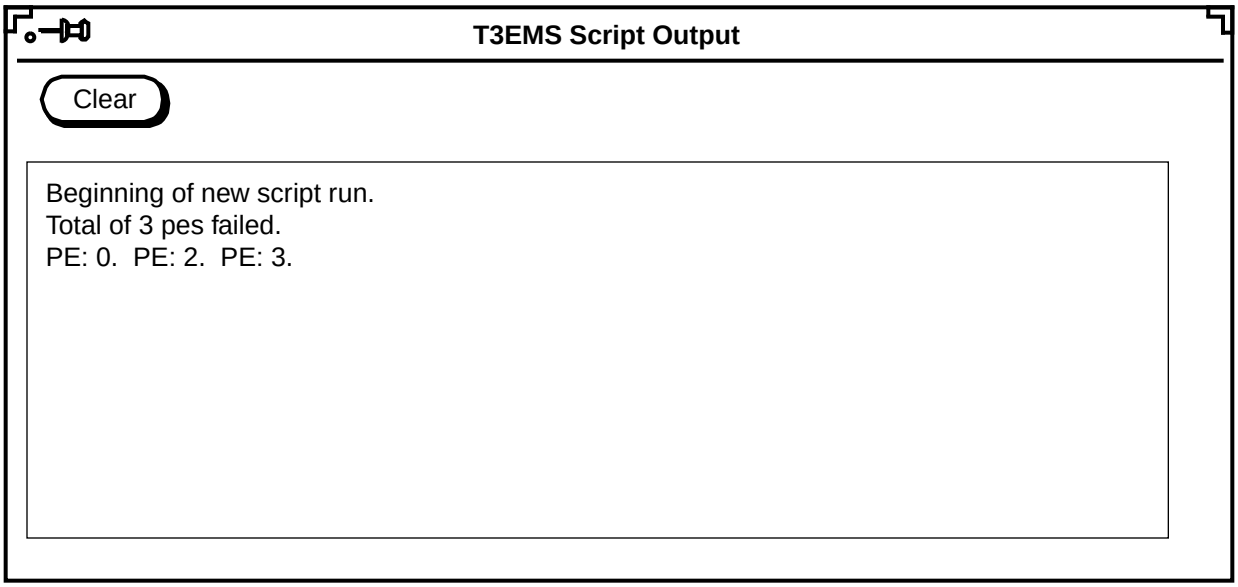


Figure 22. PE Output

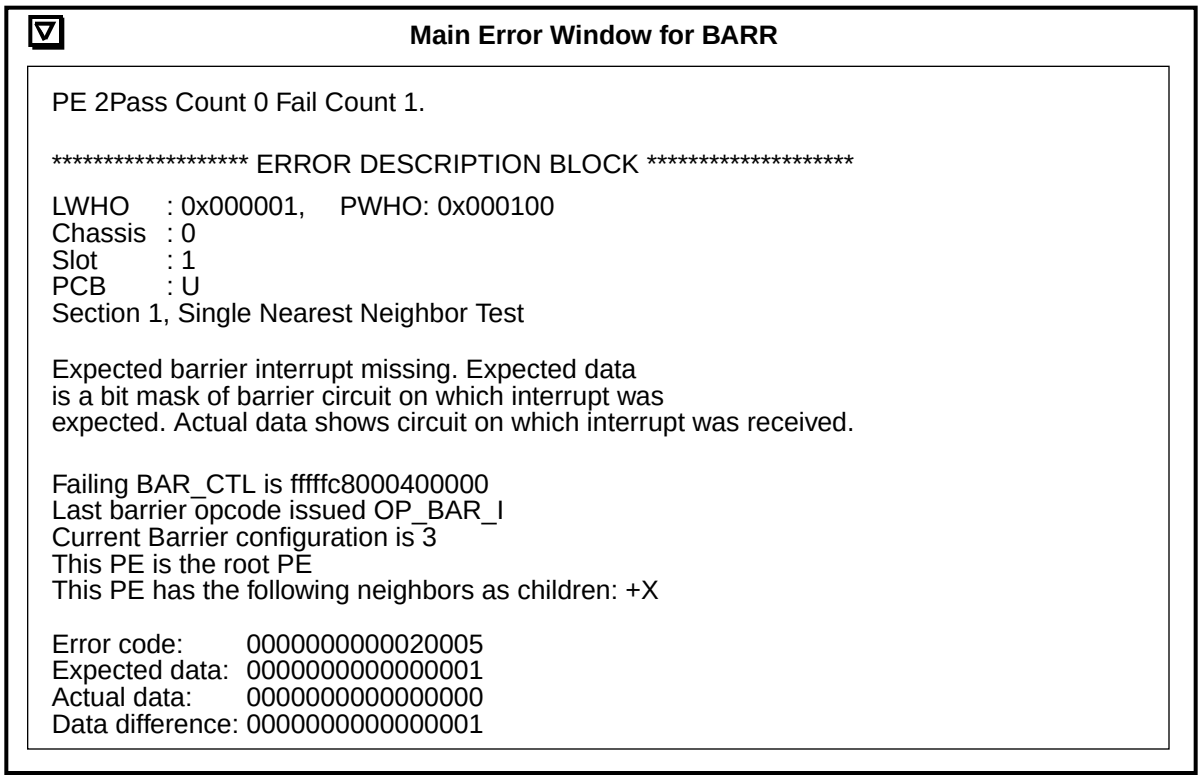
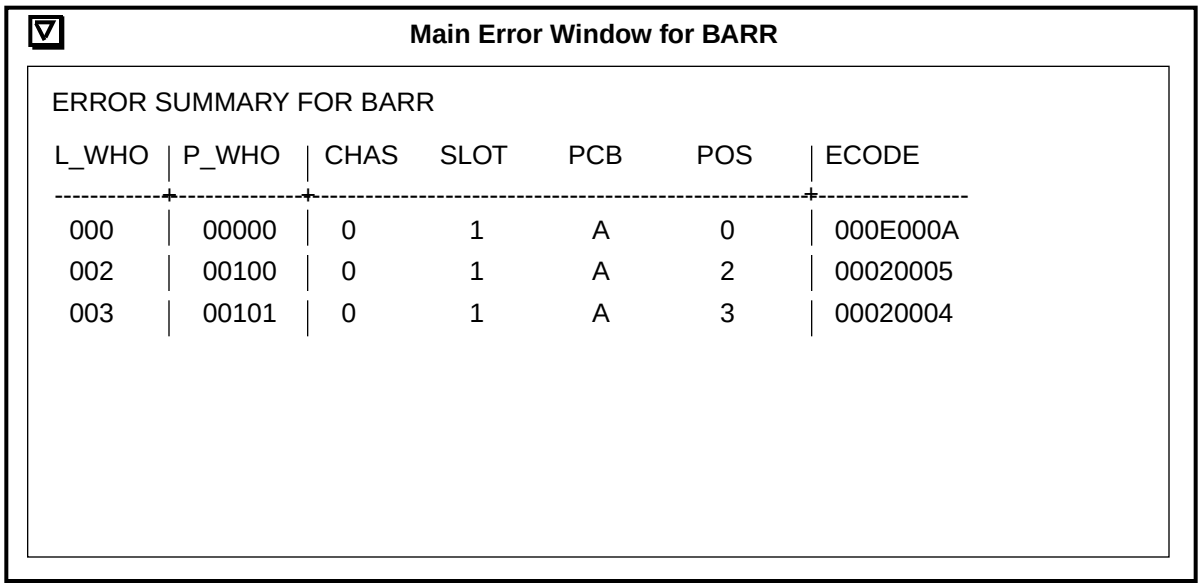


Figure 23. Summary Output



L_WHO	P_WHO	CHAS	SLOT	PCB	POS	ECODE
000	00000	0	1	A	0	000E000A
002	00100	0	1	A	2	00020005
003	00101	0	1	A	3	00020004

Figure 23 shows that PEs 0, 2, and 3 of chassis 0, slot 1, PCB A failed. The error codes indicate that PE 0 detected that another PE reported a failure. PE 1 detected that an expected barrier interrupt did not occur. PE 3 detected that while it was polling for a eureka event, a timeout occurred.

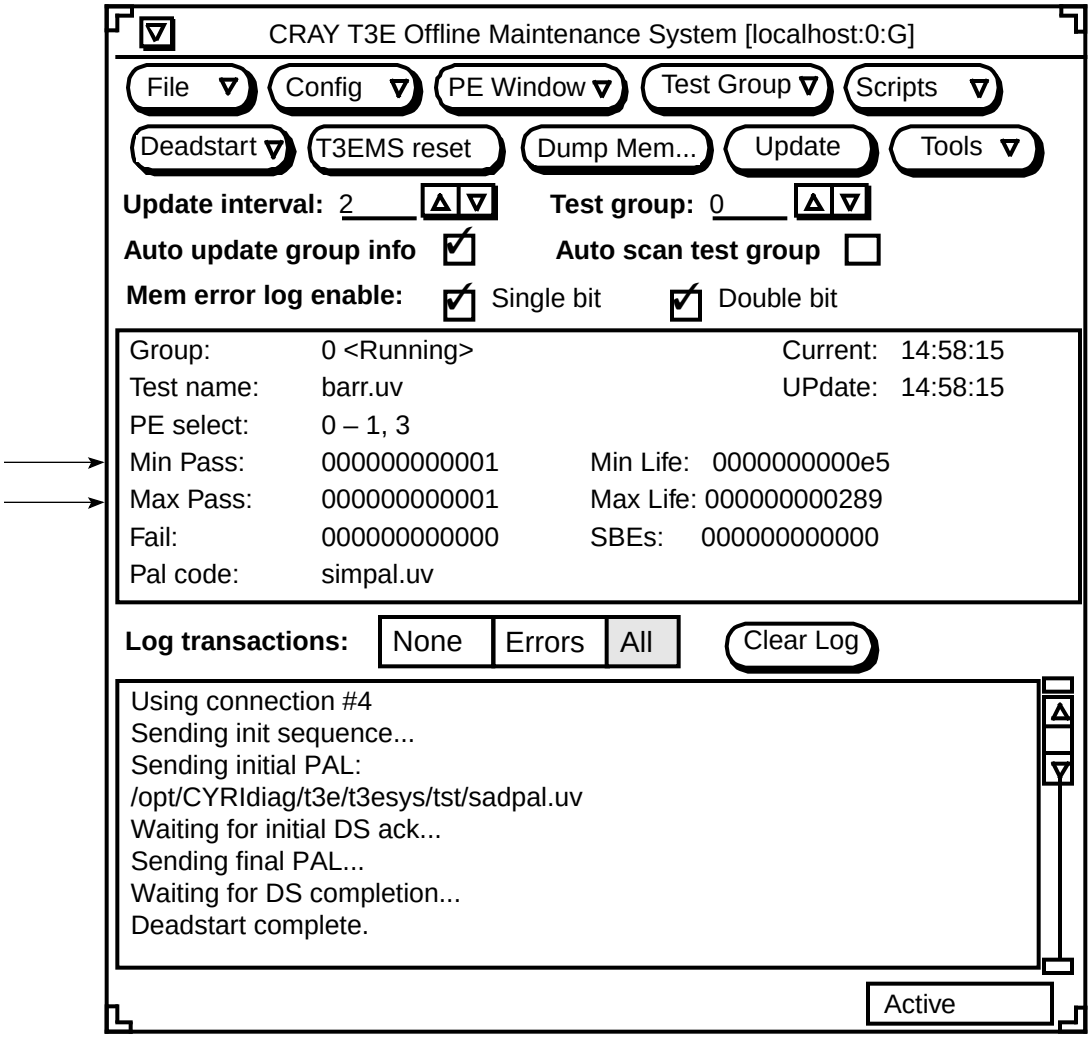
To verify which PE is failing, disable one of the failing PEs from running the test by performing the following steps:

Note: For this failure, disable PE 2 or PE 3 since PE 0 is reporting only that another PE detected an error.

1. Select PE Configuration from the Config menu.
2. When the T3EMS PE Configuration window appears, select PE 2 or PE 3 in the PE list box.
3. Click on the T for PE 2 or PE 3 in the PE list box.
4. Rerun the barr diagnostic test.

The barr diagnostic test passes with PE 2 disabled from test; therefore, PE 2 is causing the failure (refer to Figure 24).

Figure 24. barr Diagnostic Test Running without Errors



To correct the problem, you can replace the module that contains PE 2 or you can map out PE 2. For module replacement procedures, refer to *AC Repair and Adjustment*, publication number HMM-168-0 or *LC Repair and Adjustment*, publication number HMM-175-0. To map out PE 2, disable PE 2's communication link by performing the following steps:

1. Select PE Configuration from the Config menu.
2. When the T3EMS PE Configuration window appears, select PE 2 from the PE list box and click PE.
3. To save your change, select System Configuration from the Config menu. When the T3EMS System Configuration window appears, click Save.

Go to the SIO Troubleshooting Guide

The publication number of the *SIO Troubleshooting Guide* is HMM-204-0.

Follow the Escalation Procedure

For information about the escalation procedure, refer to:

http://www.csd.sgi.com/proc/escalation/

Reboot the System

To boot the system, perform the following steps:

1. Log into the SWS as `crayops`.
2. To boot the entire system, type the following command:

   `bootsys -c`

   This command reads the topology file to determine which components of the system should be booted; the `-c` option starts a console window.

   To boot the CRAY T3E system only, type the following command:
   `bootsys -c [mainframe name]`
3. When the CRAY T3E system is in single-user mode, the console window displays the following prompt: sn*xxxx*#

   Note: *xxxx* is the serial number of your CRAY T3E system.

   At the sn*xxxx*# prompt, type the following command:

   `init 2`

   This command puts the CRAY T3E system in multi-user mode.
4. When the command prompts you for a response, answer appropriately for your site.

   When the CRAY T3E system is in multi-user mode, the console window displays the following prompt: Console login:

t3e_hdw_boot Command

The `bootsys` command calls the `t3e_hdw_boot` command to initialize the CRAY T3E system during the operating system (OS) boot. To do this, `t3e_hdw_boot` establishes a connection between the SWS and the multipurpose node (MPN) host across the Ethernet connection. `t3e_hdw_boot` also retrieves configuration information from the SWS in order to configure the code that it downloads into the CRAY T3E system. This code is normally in the hdw_boot.uv file; however, you override the hdw_boot.uv file when you supply a filename argument on the command line.

`t3e_hdw_boot` sends commands to the CRAY T3E system via the MPN to initialize a node so that it can perform direct memory access (DMA) transfers. Using DMA transfers, the `t3e_hdw_boot` command downloads the hdw_boot.uv file into the node; the node executes the hdw_boot.uv file, which initializes the hardware (network router, routing tag look-up table, memory type, speed, etc.) so that the operating system can boot.

`t3e_hdw_boot` can verify the CRAY T3E hardware by running the boundary scan test and the CRAY T3E boot resiliency diagnostic script (t3ebrd.pgm). By default, the boundary scan test is disabled. To enable the boundary scan test, supply the `-S 1` argument on the command line; supply a `-D 1` argument to run the confidence test of the t3ebrd.pgm script.

The boundary scan test verifies the connections between cabinets, modules, and options in the CRAY T3E system. When `t3e_hdw_boot` detects a boundary scan error, it tries to isolate the faulty component and automatically maps it out of the system. The types of resources that can be mapped out are nodes, interconnect network links, and network router options. For more information about boundary scan for the CRAY T3E system, refer to the *Clock and Boundary Scan* document, publication number HMM-360-A.

The t3ebrd.pgm script executes CRAY T3E offline diagnostics to identify faulty components and automatically deselects the components from the system configuration; t3ebrd.pgm continues to deselect components until it corrects the problem. For more information about the t3ebrd.pgm script, refer to the “Run the t3ebrd.pgm Script” subsection in this document and to the *Offline Diagnostics* document, publication number HDM-136-A.

When `t3e_hdw_boot` completes and it has detected an error, it prompts you to review the changes, accept the changes, or discard the changes. When you review the changes, `t3e_hdw_boot` displays the failure information. When you accept the changes, `t3e_hdw_boot` saves the changes to the t3e_config file, continues with the hardware boot, and displays status information to indicate whether the hardware boot was successful. When you discard the

changes, `t3e_hdw_boot` does not save the changes, continues with the hardware boot, and displays status information to indicate whether the hardware boot was successful.

Command Line Options

The `t3e_hdw_boot` command accepts the following options:

`-c boot_conn`
Specifies the physical location of the boot node as cabinet.slot.board.position.

`-n bootnode [:scannode]`
Specifies the boot/scan node in hexadecimal.

`-h boothost [:scanhost]`
Specifies the Ethernet host names that connect to the GigaRing proxy. The default is local host (the SWS).

`-p host_port`
Specifies the Ethernet port name of the MPN. The default is 470.

`-f filename`
Overrides the normal CRAY T3E initialization program. The default program is /opt/CYRIdiag/t3e/t3esys/tst/hdw_boot.uv.

`-s`
Validates the CRAY T3E memory checkbits by scrubbing the CRAY T3E memory (read and rewrite). Use this option after a power cycle or whenever memory is corrupt.

`-C cfg_name`
Selects the configuration file entry that is designated by `cfg_name`.

`-D diag_level`
Selects the test level for t3ebrd.pgm. The following arguments are valid:

0 None
1 Confidence (default)
2 Comprehensive
3 Exhaustive

`-P perf_limit`
Specifies how many times `t3e_hdw_boot` scans and deadstarts the PEs of the CRAY T3E system. PEs require multiple scans and deadstarts to get them into their full bandwidth state (25 is the recommended number of scans and deadstarts; the default is 0). The PEs are in a fast or slow bandwidth state after a power cycle or after the boundary scan test runs. Multiple deadstarts can change the state of the PEs so that all of the PEs are in the fast state.

The following list contains important information about the `-P perf_limit` option:

- This option scrubs memory.
- This option requires that boundary scan and node synchronization be enabled.
- This option requires that the correct scan node ID and scan host are specified using the T3EMS System Configuration window or the `-n` option.
- This option requires the version of the boot PAL code that UNICOS/mk release 1.3 contains.

`-S scan_level`
Selects the boundary scan test. When the `scan_level` is a 1, the boundary scan test is enabled; when the `scan_level` is a 0, the boundary scan test is disabled. The default is 0.

`-T scan_tck`
Adjusts the scan TCK width. The default is 14.

`-V`
Prints the program and exits.

Initiating the Boundary Scan Test and the t3ebrd.pgm Script

The following command initiates the boundary scan test and the t3ebrd.pgm script:

```
t3e_hdw_boot -h sn6505-mpn0 -S 1 -D 1
```

Note: If you do not enter the `-h` option, the boundary scan test and the t3ebrd.pgm script fail. The boundary scan test indicates the error by using an exit code of 255. After running t3ebrd.pgm, an error message that reads “ERROR: Opening connection to proxy: Unable to connect to maintenance host ‘local host’ at port 470” appears on the screen.

Test Results

As `t3e_hdw_boot` executes the tests, it writes status information to the screen. `t3e_hdw_boot` writes the following status information when the boundary scan test and t3ebrd.pgm complete successfully:

```
sws6505$ t3e_hdw_boot -h sn6505-mpn0 -S 1 -D 1
Starting an AC8 system.
Performing boot testing...
Logfile is /opt/CYRIdiag/t3e/dump/bootlog03110756
Executing boundary scan test...
Boundary scan test completed: exit code = 0.
Executing scan clear...
Scan clear completed: exit code = 0.
Executing boot diagnostic test; starting t3ems...elapsed time 02:20
Boot diagnostic test completed normally: exit code = 0x40.
Boot testing finished.

Sending PAL:
Waiting for the boot to complete...
Successful boot complete
sws6505$
```

Note: At this point, `t3e_hdw_boot` opens a script window (refer to Figure 25).

When the boundary scan test detects an error, it tries to isolate the failing component. When it is able to isolate the failing component, `t3e_hdw_boot` writes the following status information:

```
sws6505$t3e_hdw_boot -h sn6505-mpn0 -S 1 -D 1
Starting an AC8 system.
Performing boot testing...
Logfile is /opt/CYRIdiag/t3e/dump/bootlog03110821
Executing boundary scan test...
Boundary scan test completed: exit code = 0.
Executing scan clear...
Scan clear completed: exit code = 0.
Executing boot diagnostic test; starting t3ems...elapsed time 02:20
Boot diagnostic test completed normally: exit code = 0x40.
Boot testing finished.
Press RETURN for a summary of resources disabled during boot test...
```

Note: At this point, `t3e_hdw_boot` opens a script window (refer to Figure 25).

Note: The last line “Press RETURN for a summary of resources disabled during boot test...” indicates that an error occurred. All other status information does not indicate an error because the boundary scan test was able to isolate and map out the failing component; the boundary scan test completed successfully.

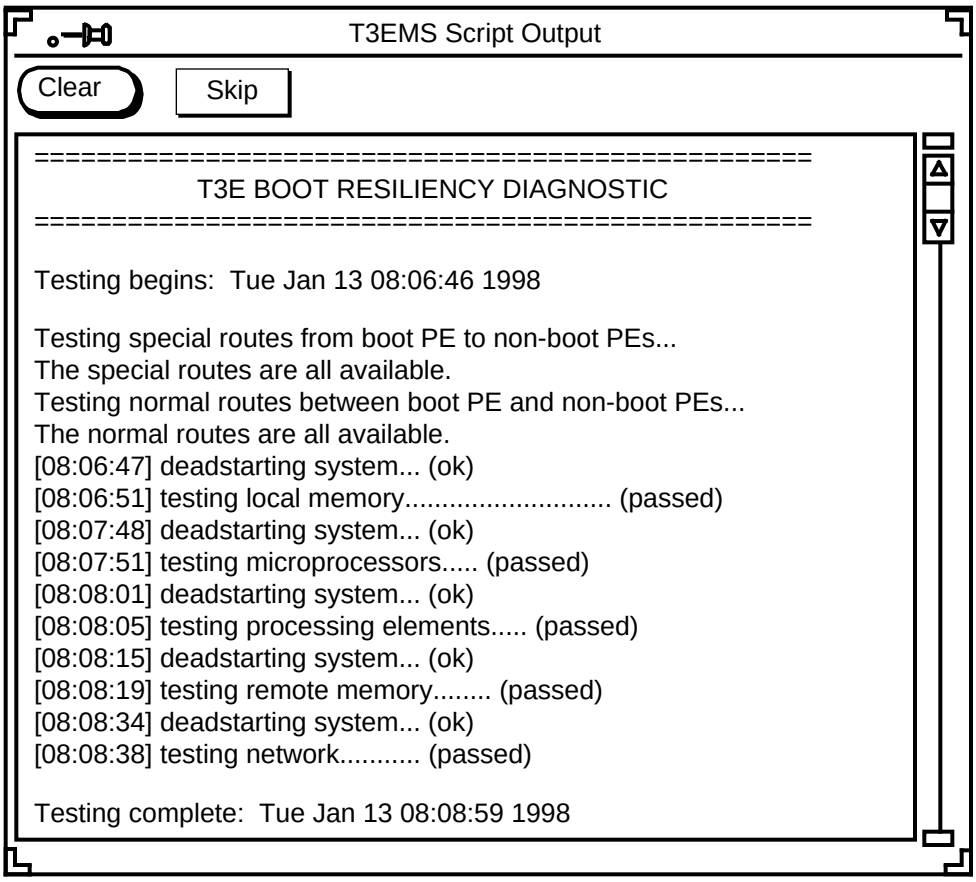
To determine which component(s) failed, press the Return key. For this example, `t3e_hdw_boot` displays the following information on the screen:

```
PE 10100:
PE 10101: -X disabled.
(R)eview changes, (A)ccept changes or (D)iscard changes:
```

`t3e_hdw_boot` prompts you to review the changes, accept the changes, or discard the changes. When you review the changes, `t3e_hdw_boot` repeats the failure information (PE 10100: PE 10101: -X disabled.). When you accept the changes, `t3e_hdw_boot` saves the changes to the t3e_config file, continues with the hardware boot, and displays status information to indicate whether the hardware boot was successful. When you discard the changes, `t3e_hdw_boot` does not save the changes, continues with the hardware boot, and displays status information to indicate whether the hardware boot was successful.

Note: In UNICOS/mk release 1.5.2 and greater, the information from the t3e_config file is passed to the operating system during the boot.

Figure 25. Script Window



`t3e_hdw_boot` also writes the output of the boundary scan test and the t3ebrd.pgm script to a bootlog file, which is located in the /opt/CYRIdiag/t3e/dump directory (refer to Figure 26). View this ASCII file when `t3e_hdw_boot` detects an error that you want more detailed information about.

Note: When the `t3e_hdw_boot` command calls the t3ebrd.pgm script, the `t3e_hdw_boot` command stores t3ebrd.pgm's output information in a bootlog file rather than the t3ebrd.log file that is noted on page 26.

Figure 26. Dump Directory

```
sws6505$ cd /opt/diag/t3e/dump
sws6505$ ls -l
total 98
-rw-rw-r--    1 craydiag craydiag    2184   Jan 13   07:41    bootlog03110738
-rw-r--r--    1 craydiag craydiag    1138   Jan 13   15:23    t3ebrd.log
```

Note: The CYRIdiag directory is linked to the diag directory; therefore, you can type `cd /opt/diag/t3e/dump` to get to the same directory as when you type `cd /opt/CYRIdiag/t3e/dump`.

Bootlog Examples

The bootlog examples provide bootlog files for the following scenarios:

- Example 1 - `t3e_hdw_boot` command runs successfully
- Example 2 - boundary scan test detects an open torus connection
- Example 3 - t3ebrd.pgm detects an open torus connection

t3e_hdw_boot Command Runs Successfully

The bootlog contains the following information when the `t3e_hdw_boot` command runs successfully:

```
sws6505$ more bootlog03110756
t3e_hdw_boot version 1.5b: begin system boot log on Tue Jan 13 10:34:38 1998

> Begin boundary scan test.
> /opt/CYRIdiag/t3e/bin/t3ebst -r res1 -d /opt/CYRIdiag/t3e/t3esys/scan -G sn65
05-mpn0:470 2>&1

T3E Boundary-Scan Test - t3ebst version 2.1

Hardware: A/C system with 16 PEs

Attempting connection to GRING node 0x0F via MPN sn6505-mpn0:0470 done.

Performing TAP test:
      Chain 00...passed
            Detected from hardware: #chips=118 IR_length=600

Reading revisions from hardware and verifying configuration:
      Chain 00...done
```

Note: The standard boundary scan output is omitted here; this output is shown in the *Clock and Boundary Scan* document, HMM-360-A.

```
t3ebst completed with no failures.

> End boundary scan test: exit code = 0.

> Begin scan clear.
```

Note: The standard scan clear output is omitted here; this output is shown in the *Clock and Boundary Scan* document, HMM-360-A.

```
> End scan clear: exit code = 0.
```

```
> Begin boot diagnostic test.
> /opt/CYRIdiag/t3e/bin/t3ems -T -R /opt/CYRIdiag/t3e/t3esys/pgm/t3ebrd.pgm

=======================================================
                T3E BOOT RESILIENCY DIAGNOSTIC
=======================================================
System Configuration Parameters:
- System Type          : Air-Cooled
-System Size (# PEs)   : 16
- LUT mode             : 0
- P_WHO of boot PE     : 0x010000
- GigaRing boot node   : 0xf
- GigaRing server host  : sn6505-mpn0
- Base directory       : /opt/CYRIdiag/t3e/t3esys
- PAL code location    : /opt/CYRIdiag/t3e/t3esys/tst/sadpal.uv
- Boot Diagnostic Level : 0x01

Testing begins:  Tue Jan 13 08:06:46 1998
Testing special routes from boot PE to non-boot PEs...
```

_L_WHO MASK	POPCOUNT: 1	"PE mask passed to rte_chk(5)"		
<255 : 196> 0000000000000000	<195 : 128> 0000000000000000	<127 : 64> 0000000000000000	<63 : 0> 0000000000008000	BASE 0

_L_WHO MASK	POPCOUNT: 0	"PE mask returned from rte_chk(5)"		
<255 : 196>	<195 : 128>	<127 : 64>	<63 : 0>	BASE

```
The special routes are all available.
Testing normal routes between the boot PE and non-boot PEs...
```

_L_WHO MASK	POPCOUNT: 1	"PE mask passed to rte_chk(3)"		
<255 : 196> 0000000000000000	<195 : 128> 0000000000000000	<127 : 64> 0000000000000000	<63 : 0> 0000000000008000	BASE 0

_L_WHO MASK	POPCOUNT: 0	"PE mask returned from rte_chk(3)"		
<255 : 196>	<195 : 128>	<127 : 64>	<63 : 0>	BASE

```
The normal routes are all available.
[08:06:47] deadstarting system... (ok)
[08:06:51] testing local memory..... (passed)
[08:07:48] deadstarting system... (ok)
[08:07:51] testing microprocessors..... (passed)
[08:08:01] deadstarting system... (ok)
[08:08:05] testing processing elements..... (passed)
[08:08:15] deadstarting system... (ok)
[08:08:19] testing remote memory..... (passed)
[08:08:34] deadstarting system... (ok)
[08:08:38] testing network..... (passed)
Testing complete:  Tue Jan 13 08:08:59 1998

> End boot diagnostic test: exit code = 0x40.

Boot testing finished.
```

Boundary Scan Test Detects an Open Torus Connection

Bootlog contains the following boundary scan failure information when the boundary scan test detects an open torus connection.

> **Note:** The boundary scan header information is omitted here; this header information is shown in the *Clock and Boundary Scan* document, HMM-360-A.

```
Boundary scan errors detected!!!
Raw pin number          : 00-14911
Expected data           : 10101001011001101001100110101O
Actual data             : 00000000000000000000000000000000
FAILING BOARD  (source) : CHASSIS = 0   SLOT =  1   BOARD = Upper(A)
FAILING BOARD  (dest)   : CHASSIS = 0   SLOT =  1   BOARD = Upper(A)
FAILING NET NAME        : CHIP = R10        PIN = X1OUT21
RESILIENCY REQUIREMENT  : Disable the -X port on PWHO 10101
```

•
•
•

```
Raw pin number          : 00-14915
Expected data           : 10101001011001101001011001101O
Actual data             : 00000000000000000000000000000000
FAILING BOARD (source) : CHASSIS = 0   SLOT = 1   BOARD = Upper(A)
FAILING BOARD (dest)   : CHASSIS = 0   SLOT = 1   BOARD = Upper(A)
FAILING NET NAME       : CHIP = R10        PIN = X1OUT23
RESILIENCY REQUIREMENT : Disable the -X port on PWHO 10101
```

> **Note:** This example shows only 2 of the 30 errors.

```
t3ebst completed with 30 failures.
> End boundary scan test: exit code = 0.

> Begin scan clear.
```

> **Note:** The standard scan clear output is omitted here; this output is shown in the *Clock and Boundary Scan* document, HMM-360-A.

```
> End scan clear: exit code = 0.
```

t3ebrd.pgm Detects an Open Torus Connection

The bootlog contains the following failure information from the t3ebrd.pgm script when it detects an open torus connection. t3ebrd.pgm detected this error when the `t3e_hdw_boot` command was initiated to run only t3ebrd.pgm (`t3e_hdw_boot -h sn`*`6505`*`-mpn`*`0`*` -D 1`).

```
sws6505$ more bootlog03110831
t3e_hdw_boot version 1.5b: begin system boot log on Tue Jan 13 08:31:23 1998

> Begin boot diagnostic test.
> /opt/CYRIdiag/t3e/bin/t3ems -T -R /opt/CYRIdiag/t3e/t3esys/pgm/t3ebrd.pgm

=======================================================
                T3E BOOT RESILIENCY DIAGNOSTIC
=======================================================

System Configuration Parameters:
- System Type          : Air-Cooled
- System Size (# PEs)   : 16
- LUT mode             : 0
- P_WHO of boot PE      : 0x010000
- GigaRing boot node    : 0xf
- GigaRing server host  : sn6505-mpn0
- Base directory        : /opt/CYRIdiag/t3e/t3esys
- PAL code location     : /opt/CYRIdiag/t3e/t3esys/tst/sadpal.uv
- Boot Diagnostic Level : 0x01

Testing begins:  Tue Jan 13 08:31:25 1998
Testing special routes from boot PE to non-boot PEs...
```

_L_WHO MASK	POPCOUNT: 1	"PE mask passed to rte_chk(5)"		
<255 : 196> 0000000000000000	<195 : 128> 0000000000000000	<127 : 64> 0000000000000000	<63 : 0> 0000000000008000	BASE 0

_L_WHO MASK	POPCOUNT: 0	"PE mask returned from rte_chk(5)"		
<255 : 196>	<195 : 128>	<127 : 64>	<63 : 0>	BASE

```
The special routes are all available.
Testing normal routes between the boot PE and non-boot PEs...
```

_L_WHO MASK	POPCOUNT: 1	"PE mask passed to rte_chk(5)"		
<255 : 196> 0000000000000000	<195 : 128> 0000000000000000	<127 : 64> 0000000000000000	<63 : 0> 0000000000008000	BASE 0

_L_WHO MASK	POPCOUNT: 0	"PE mask returned from rte_chk(5)"		
<255 : 196>	<195 : 128>	<127 : 64>	<63 : 0>	BASE

The normal routes are all available.
[08:31:25] deadstarting system... (ok)
[08:31:30] testing local memory... (failed)

ERROR SUMMARY FOR lmem

L_WHO (hex)	P_WHO (hex)	CHAS SLOT PCB POS (decimal)	ECODE	Description
006	10101	0 1 U 3	40030000	Unexpected CRAY Interrupt

_R_ERR INFORMATION FOR “AC 8" CONFIGURATION

L_WHO (hex)	P_WHO (hex)	CHAS SLOT PCB POS (decimal)	_R_ERR0 (hex)	_R_ERR0 & _R_ERR1 (hex)
006	10101	0 1 U 3	0000000001000000	0000000001000000

DECODE OF _R_ERR INFORMATION FOR EACH PE
(* indicates beginning of particle trace)

Destination L_WHO P_WHO PORT	Destination L_WHO P_WHO PORT	DECODE OF _R_ERR0 & _R_ERR1 (reported by destination node)
006 10101 –X, X0	005 10100 –X, X1	channel parity error (CPE)

Disabled X0 (-X) port for P_WHO 0x10101 (L_WHO 0x006).
[08:31:42] deadstarting system... (ok)
[08:31:49] testing local memory..... (passed)
[08:32:46] deadstarting system... (ok)
[08:32:54] testing microprocessors..... (passed)
[08:33:04] deadstarting system... (ok)
[08:33:08] testing processing elements..... (passed)
[08:33:18] deadstarting system... (ok)
[08:33:21] testing remote memory..... (passed)
[08:33:37] deadstarting system... (ok)
[08:33:41] testing network..... (passed)

Testing complete: Tue Jan 13 08:34:03 1998

> End boot diagnostic test: exit code = 0x41.

Config changes discarded by user.

Boot testing finished.
sws6505$

Achieving a Fast-band Torus

`t3e_hdw_boot` tries to achieve a fast-band torus when you initiate `t3e_hdw_boot` with the `-P perf_limit` option.

To achieve a fast-band torus, the clock phase between the C and R options must be the same (all even or all odd) and the bandwidth between the boot node and the other nodes in the system must be in the fast-band state.

To synchronize the clock phase between the C and R options, `t3e_hdw_boot` performs the following steps:

1. Performs normal hardware initialization and verifies the status of the system.

2. Selects an alternate boot node that checks the performance of the configured boot node and synchronizes the clock phase between the C and R option of the boot node.

   The boot node measures the bandwidth between itself and each node in the system. The boot node uses a slow_node bitmask to indicate the number of nodes that are in the slow-band state and the fast-band state.

3. Reads the slow_node bitmask to determine which band state contains the fewest nodes. The band state that contains the fewest nodes is also considered the minority phase group.

4. Reads the lpe_mask bitmask from the boot node and issues a RAMBIST function/command to the R option of each node in the minority phase group. The RAMBIST function is a boundary scan self-test that causes the system clock to restart. The restart of the system clock may toggle the clock phase between the C and R options.

5. Repeats Step 4 until all C-option-to-R-option clock phases are the same.

To change the bandwidth between the boot node and the other nodes in the system to the fast-band state, `t3e_hdw_boot` performs the following step:

- Determines whether the nodes are in the slow-band state or the fast-band state. When all of the nodes are in the fast-band state, `t3e_hdw_boot` completes the boot process. When all of the nodes are in the slow-band state, `t3e_hdw_boot` continues to reboot the system until all nodes are in the fast-band state. During the reboot, `t3e_hdw_boot` reasserts the reset on the R options.

Note: `t3e_hdw_boot` tests the torus performance until the retry limit is reached. The recommended number of retries is 25. After the retry limit is reached, if the torus is not in the fast-band state, `t3e_hdw_boot` completes the boot; however, the state of the torus is unknown. **Hint:** The efforts of `t3e_hdw_boot` are cumulative; therefore when `t3e_hdw_boot` times out, rerun `t3e_hdw_boot`. `t3e_hdw_boot` starts at the point where it left off when it timed out.

Note: `t3e_hdw_boot` may have to synchronize the clock phase between the C and R options after you power cycle the system and after you run the boundary scan test.

`t3e_hdw_boot` may have to initialize the R options so that they are in the fast-band state after you power cycle the system, after you run the boundary scan test, and after you reboot the system.

`t3e_hdw_boot` outputs the following information when you initiate it with the `-P perf_limit` option:

```
sws6505$ t3e_hdw_boot -h sn6505-mpn0:sn6505-mpn0 -n f:f -C Default -D0 -S0 -P25
Starting an AC16 T3E system.

Sending PAL...
Waiting for the boot to complete.....
Successful boot complete.

Beginning boot performance adjustment...
INTERRUPTS ARE BLOCKED DURING CRITICAL OPERATIONS.
Using alternate boot pwho 0x10001 to sync primary boot PE.

Sending PAL...
Waiting for the boot to complete.....

Hardware: A/C system with 16 PEs
Attempting connection to GRING node 0x0F via MPN sn6505-mpn0:0470 done.

Performing TAP test:
        Chain 00...passed
                Detected from hardware: #chips=118 IR_length=600

Reading revisions from hardware and verifying configuration:
         Chain 00...done

          Boundary-scan revisions from hardware (by scan chain):
          <GRC> = '+' if GigaRing chip in scan chain; ' ' if chip in bypass
          <BRD> = board on 'A' or 'B' side of coldplate
           <REV> = bits 2^3 - 2^0 of BSTREV (in hexadecimal)

                 G B R
                 R R E
            00   C D V
            ----  -  -  -
           004U    A 0
           003U    A 0
           002U + A 0
           001U + A 0
Scanning LPEs <pass 01>: 4-6, 9, e, f
Using alternate boot pwho 0x10001 to sync primary boot PE.

Sending PAL...
Waiting for the boot to complete.....
Scanning LPEs <pass 02>: 5, 6

Sending PAL...
Waiting for the boot to complete.....

Sending PAL...
Waiting for the boot to complete.....
Boot complete - all PEs measured at full speed.
sws6505$
```

Online Troubleshooting Procedures

This section provides information about the following online procedures:

- Monitor the system
- Monitor the warning and control system
- Submit and execute online diagnostics
- Dump the system
- Shut down the operating system
- Boot UNICOS/mk in degraded mode

Monitor the System

Table 6 lists the commands that you can use to monitor the CRAY T3E system. Table 6 also lists where you should execute the commands.

Table 6. CRAY T3E System Monitoring Commands

Command	Where to Execute
`ps`	CRAY T3E;UNICOS/mk
`grmview`	CRAY T3E;UNICOS/mk
`xmppview`	CRAY T3E;UNICOS/mk
`xsam`	CRAY T3E;UNICOS/mk
`jstat`	CRAY T3E;UNICOS/mk
`checkt3e`	SWS;Solaris
`errpt`	CRAY T3E;UNICOS/mk
`sysmon`	SWS;Solaris®

For more information on system monitoring tools and procedures, refer to the *UNICOS/mk General Administration* document, publication number SG-2601, and the *SWS-ION Administration and Operations Guide*, publication number SG-2204. For information on a specific command, refer to the man page.

The ps Command

The `ps` command reports process status. The following options generate additional information for UNICOS/mk systems:

`-M`
Displays only multiple-PE jobs, one job per line.

`-P`
Includes the number of PEs (NPES) for jobs. When you specify this option without the `-M` option, `ps` displays a line for each thread and shows the virtual PE number (VPE) that the thread is running on.

The grmview Command

The `grmview` command displays information from the global resource manager (GRM) about the current PE configuration (PE map), applications that are currently running on the PEs, and applications that are waiting to run on the PEs.

The `grmview` command accepts the following options:

`-a`
Displays the largest continuous span of application PEs.

`-d`
Displays PE numbers in decimal.

`-e`
Displays the PE range in running queue lines.

`-l`
Displays an extended PE map.

`-M`
Displays an uncompressed PE map.

`-m`
Displays a compressed PE map. A compressed PE map does not display map lines for consecutive PEs that have the same information; instead, the compressed PE map displays the following message: *nn* identical PEs skipped.

`-n`
Checks for nonoperational PEs.

`-q`
Displays the run queue and the wait queue.

`-r`
Displays the run queue.

`-t`
Suppresses the display of header lines.

`-w`
Displays the write queue.

The PE map contains the following fields:

- Type - Type of PE (OS, CMD, APP); a plus sign (+) indicates an operational PE, a negative sign (–) indicates a nonoperational PE, and an exclamation point (!) precedes an explanation for a nonoperational PE.
- PE - Logical PE number in hexadecimal.
- Ap. Size min. and max. - Minimum (min) and maximum (max) application size for the PE. The application size is the number of PEs that an application uses. For example, if PE 7 shows a minimum value of 4 and maximum value of 8, it means that it can run only applications that require 4 to 8 PEs.
- Number Aps running and limit - Number of applications currently running on the PE, and the application limit (the maximum number of applications that can run on the PE).
- Label - Label assigned to the PE by the administrator. A dash (-) indicates the administrator has not assigned a label to the PE.
- Lists - List of service (svc), users (uid), groups (gid), and accounts (acid) that are allowed to use the PE.

In addition to the fields listed above, the extended PE map includes the following fields:

- x - Physical X-coordinate of the PE
- y - Physical Y-coordinate of the PE
- z - Physical Z-coordinate of the PE
- Clock - Clock speed of the PE in MHz
- UsrMem - Available user memory in Mbytes
- FreMem - Memory that is currently free in Mbytes

The xmppview Command

The `xmppview` command shows a text or graphical display of the activity and configuration of the CRAY T3E system. The `xmppview` command updates the displays in real time.

To execute the `xmppview` command, perform the following steps to set the display variable so that `xmppview` displays the text or graphics to your workstation:

1. In a workstation window, type the following command:

   ```
   xhost + [mainframe name]
   ```

2. Log into the mainframe by typing the following command:

   ```
   rlogin [mainframe name]
   ```

3. Type the following command:

   ```
   export DISPLAY=[workstation name]:0.0
   ```

4. Type the following command:

   ```
   xmppview
   ```

The xsam Command

The `xsam` command monitors the activity of the CRAY T3E system in real time. It displays the information that it receives from the activity-monitoring daemon and the samdaemon.

To execute the `xsam` command, perform the following steps to set the display variable so that `xsam` displays the information to your workstation:

1. In a workstation window, type the following command:

   ```
   xhost + [mainframe name]
   ```

2. Log into the mainframe by typing the following command:

   ```
   rlogin [mainframe name]
   ```

3. Type the following command:

   ```
   export DISPLAY=[workstation name]:0.0
   ```

4. Type the following command:

   ```
   xsam
   ```

The jstat Command

The `jstat` command displays information that pertains to either one job or all active jobs. If the `-j` option is not specified, `jstat` displays all jobs in the following format:

```
              nproc   CMD memory CMD time   PEs
jid owner    use lim   use lim    use lim  use lim  command
--- -----    --- ---   --- ---    --- ---  --- ---  -------
```

The fields contain the following information:

`jid`	Job ID
`owner`	ASCII name of job owner
`nproc`	Number of processes that are in use and the limit
`CMD memory`	Number of clicks that are in use and the limit (A click is 512 Cray words.)
`CMD time`	Number of CPU seconds that the job uses and the limit
`PEs`	Number of PEs that are in use and the limit
`command`	Name of the command for the current job

When you specify the `-j` option, `jstat` displays all processes for the specified job using the following format:

```
pid   status  state   utime   stime  size  system call  command
---    ------  -----    -----   -----  ----  -----------  -------
```

The fields contain the following information:

`pid`	Process ID
`status`	Current status (run or sleep)
`state`	Current state
`utime`	User time in seconds
`stime`	System time in seconds
`size`	Size of process in clicks
`system call`	Last system call that was made
`command`	Process name

The checkt3e Command

The `checkt3e` command checks the status of the operating system that is running on the CRAY T3E system; when `checkt3e` determines that a panic occurred, it displays information about the panic.

When the operating system is running, the `checkt3e` command displays the following information:

```
checkt3e: INFO: Processing checkt3e ‘sn6505’
checkt3e: INFO: The kernel has been booted and is running
```

When the operating system is down, the `checkt3e` command displays the following information:

```
checkt3e: ERROR: sn6505: Check NOT performed.
checkt3e: ERROR:        State indicates mainframe is not running.
checkt3e: ERROR:       Use -F option to force check.
```

The errpt Command

The `errpt` command processes data that the error-logging mechanism (errdemon) collects and generates a report of that data. The default report is a summary of all errors in the files that you specify on the command line. The `errpt` command accepts the following options:

`-s date`
Ignores all error records that were posted earlier than `date`. `date` uses the mmddhhmmyy[ss] format and is consistent in meaning with the `date` command. The first mm is the month number; the second mm is the minute number.

`-s lnumber`
Specifies that an error report should start `lnumber` of days prior to the date and time specified by the `-e` options.

`-e date`
Ignores all error records that were posted later than `date`. `date` uses the mmddhhmmyy[ss] format and is consistent in meaning with the `date` command. The first mm is the month number; the second mm is the minute number.

`-e lnumber`
Specifies that an error report should end `lnumber` of days prior to the current date and time.

`-a`
Produces a detailed report that includes all error types.

`-b`
Displays system startup (boot) entries.

`-c`
Displays configuration entries for the CRAY T3E system.

`-d pdevlist`
Limits a detailed report to data about devices that you specify in `pdevlist`. `pdevlist` can be one of two forms: a list of device identifiers separated by commas, or a list of device identifiers enclosed in double quotation marks and also separated by commas. `errpt` recognizes the following device types: `comm`, `fddi`, `hippi`, `tape`, `lsp`, and `dsk`.

Note: `dsk` returns error information for all disk types. Use `xdd` or `pdd` to return error information for a specific disk driver.

The device type `ios` implies all devices connected to the I/O subsystem (all of the preceding identifiers). An additional identifier is `mem`, which includes a detailed report of memory errors.

The `-d pdevlist` option does not display the relative block number of a file system. The `bb` command must be used to find the relative block number.

`-f`
In a detailed report, limits the report of device errors to unrecovered errors.

`-h`
In a detailed report, limits the report of errors to actual hardware device errors.

`-o`
Produces a one-line summary for each error.

`-p pe [,pe ...]`
Limits the entries to entries that the specified PEs logged.

`-q`
Displays the summary-error-total information.

`-r`
Formats the error records in raw mode. This is useful if the error log has errors from devices or machine types that `errpt` is not prepared to format. You can also use the `-r` option to ensure that `errpt` formats new records properly.

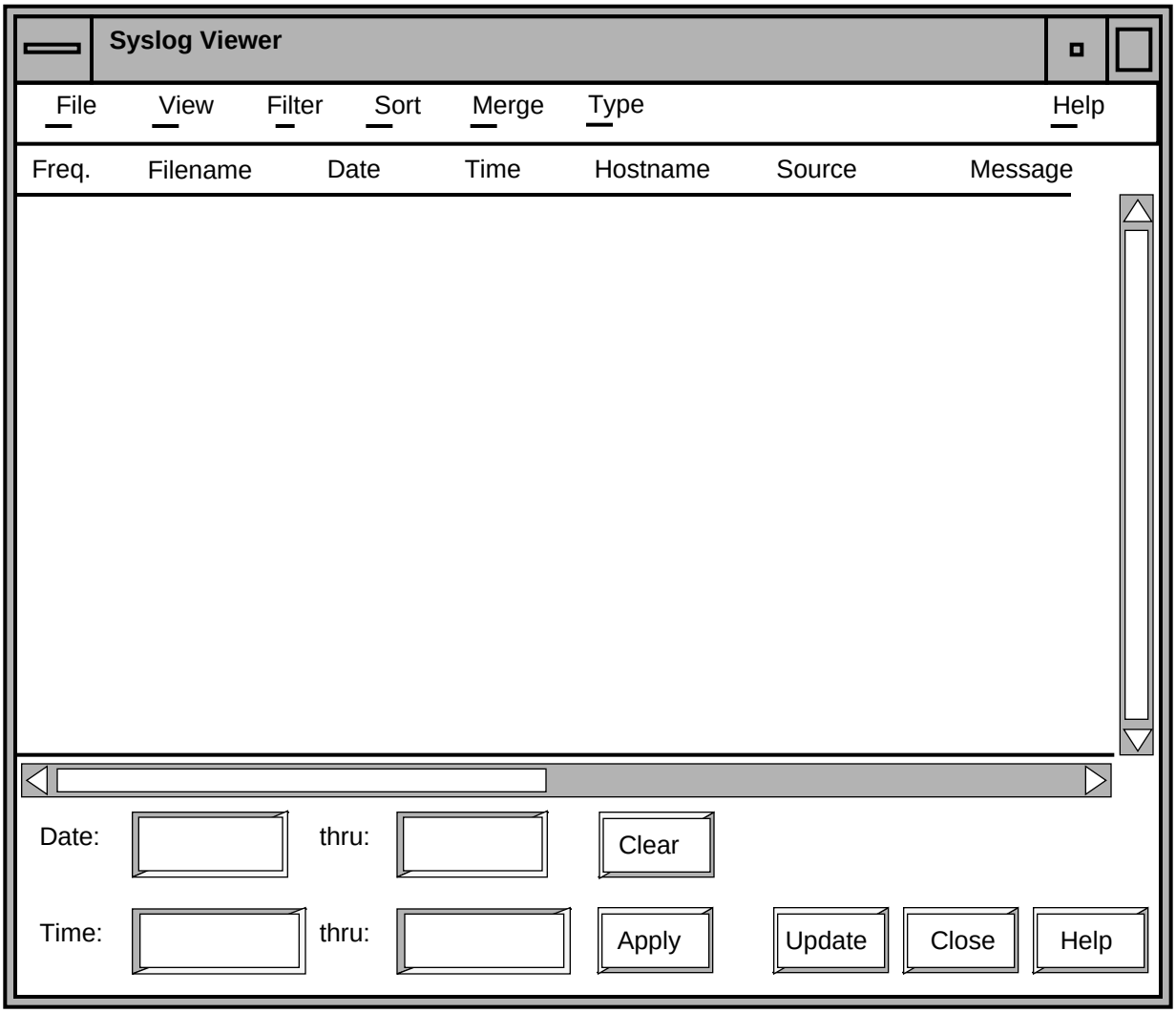
`-x`
When you use this option in conjunction with the `-r` option, displays the raw data in hexadecimal. The default display mode is octal.

`files`
Specifies the files that `errpt` will process. If you do not specify `files`, `errpt` tries to use /usr/adm/errfile.

The sysmon Command

The `sysmon` command provides a graphical view of all log files on the SWS and the mainframe. To bring up the Syslog Viewer window that Figure 27 shows, type the following command: **`sysmon`**

Figure 27. sysmon Interface



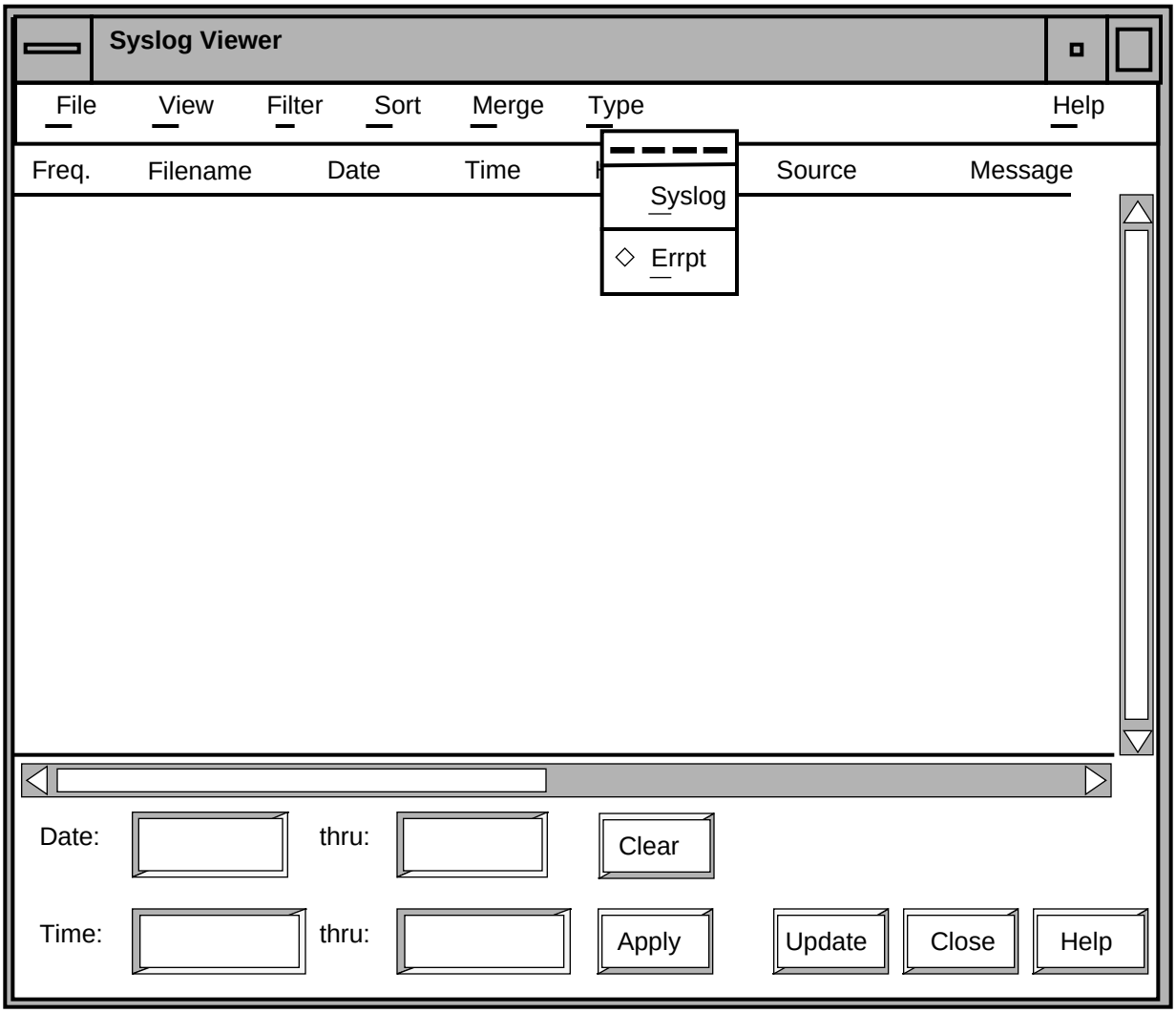
For more information about how to access and interpret information via the sysmon interface, refer to the *SWS-ION Administration and Operations Guide*, publication number SG-2204.

Use the sysmon Interface to Display errpt Information

You can use the sysmon interface to display information from the errpt file. To do this, perform the following steps:

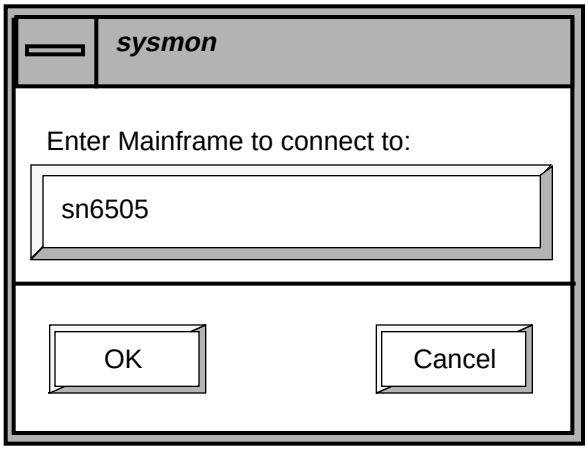
1. Start `sysmon` on the SWS.
2. Start `diagccmt` on the CRAY T3E system.
3. Select Errpt from the Type menu (refer to Figure 28).

Figure 28. Selecting Errpt from the Type Menu



4. When the popup window shown in Figure 29 appears, enter the CRAY T3E hostname and click the OK button.

Figure 29. Hostname Popup Window



5. When the popup window shown in Figure 30 appears, click the OK button.

   Note: In the command popup window, you can initiate the `errpt` command with any command line option. The `-u` option is the default.

Figure 30. Command Popup Window

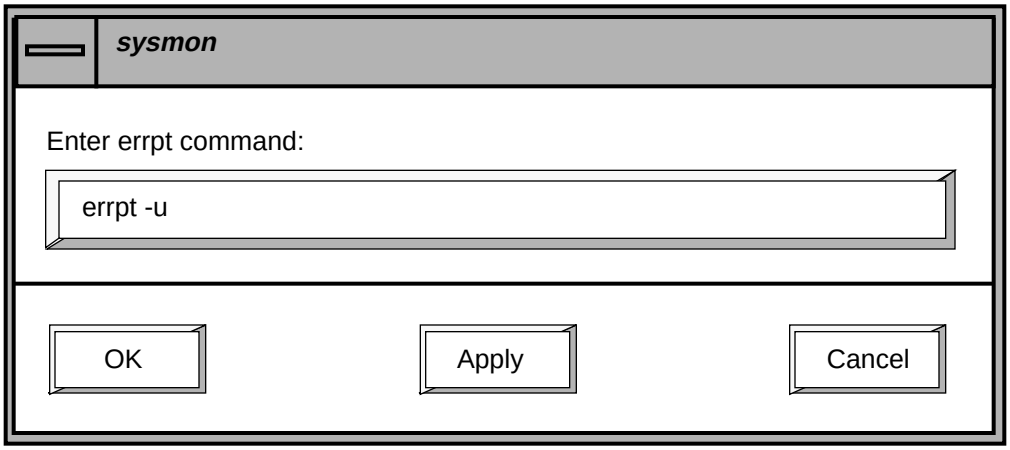
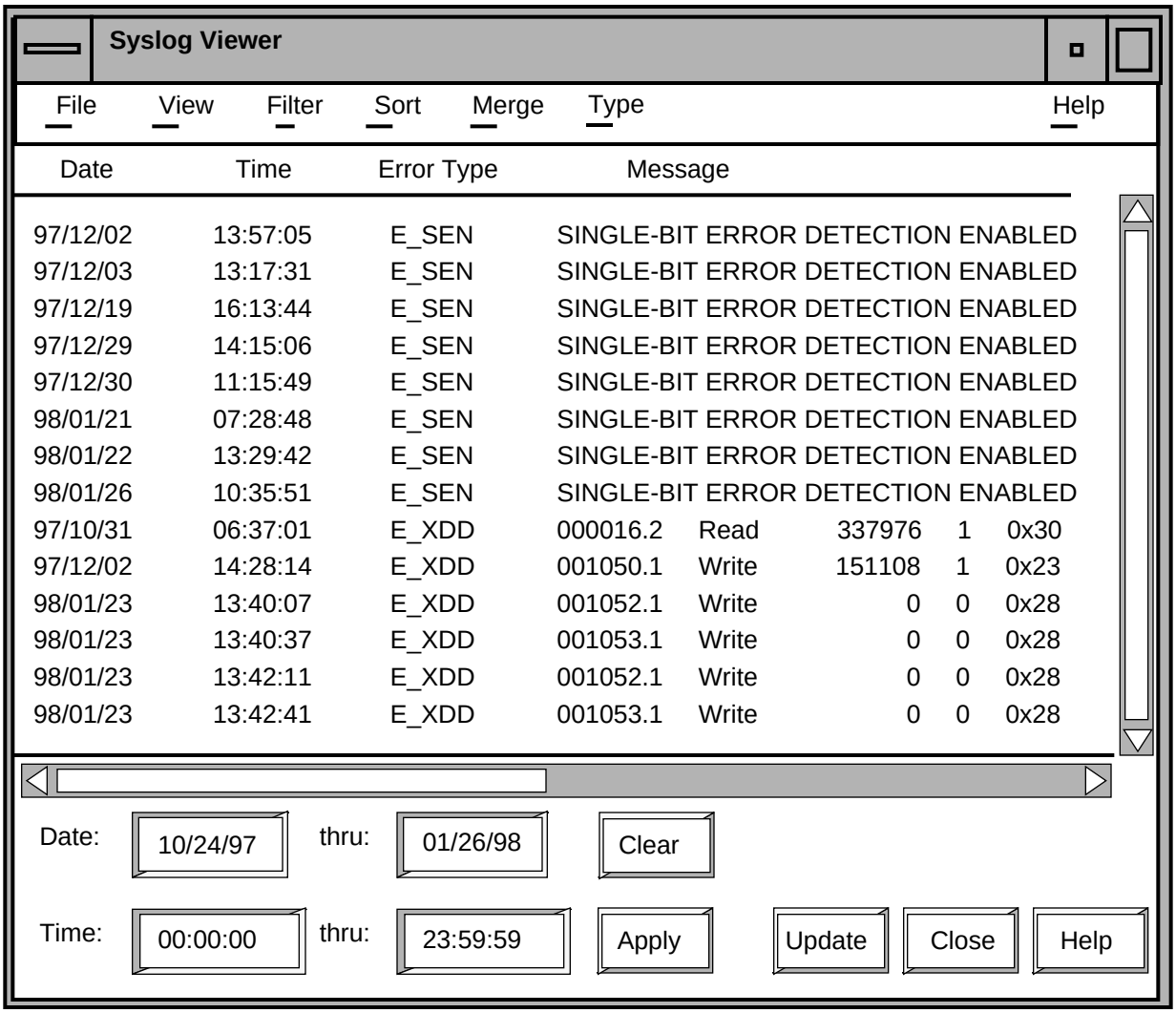


Figure 31 shows a sample errpt report.

Figure 31. errpt Report



Use a selection from the Filter menu to decrease the amount of information that the sysmon interface displays in the window (refer to Figure 32). For example, using the E_XDD filter option, sysmon displays the errpt information shown in Figure 33.

Note: E_GOxx, E_COPT, E_IOS(NW), E_XDD, E_PDD, and E_SEN are some of the error types that the sysmon interface displays for errpt.

Figure 32. Filter Menu

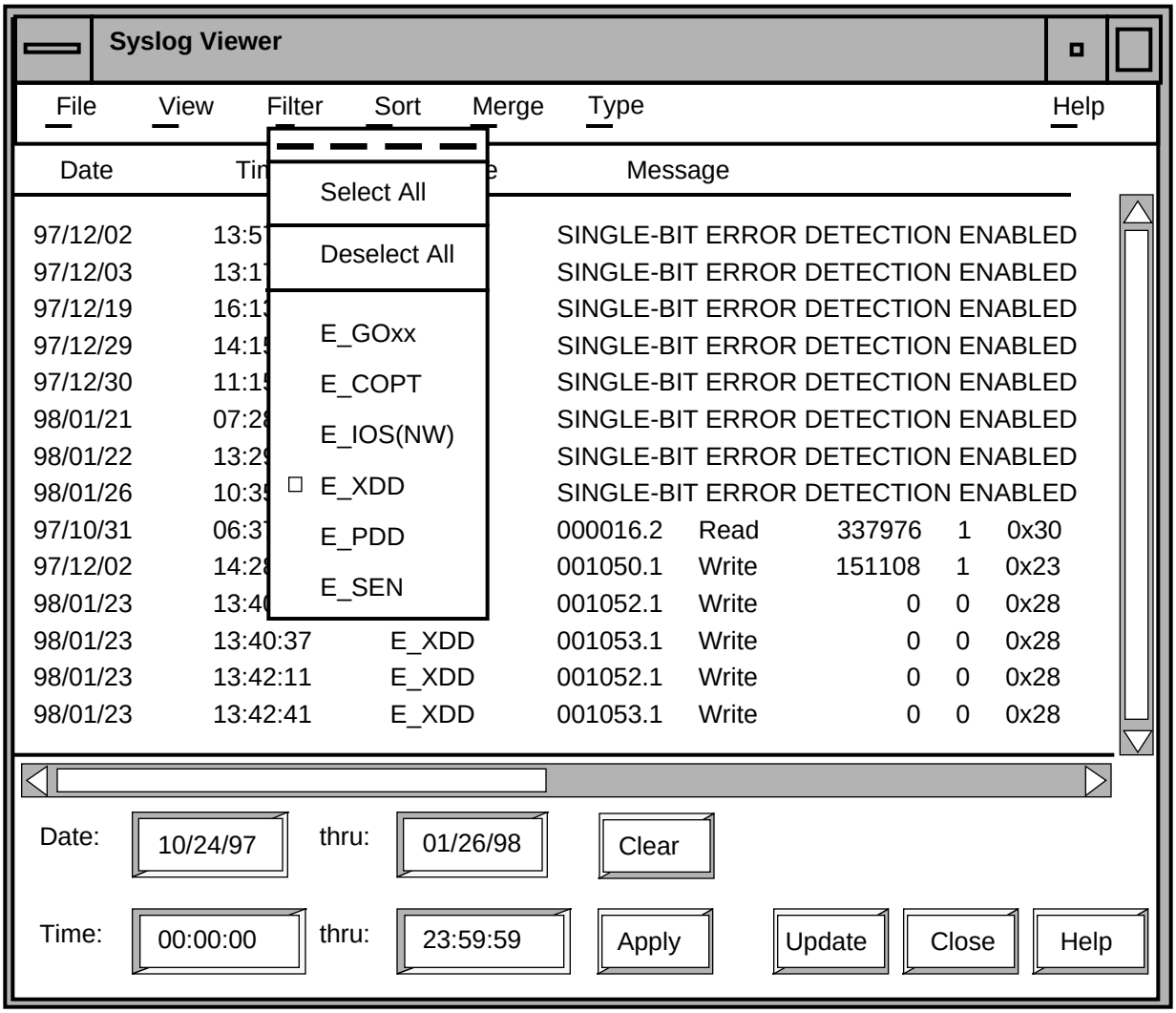
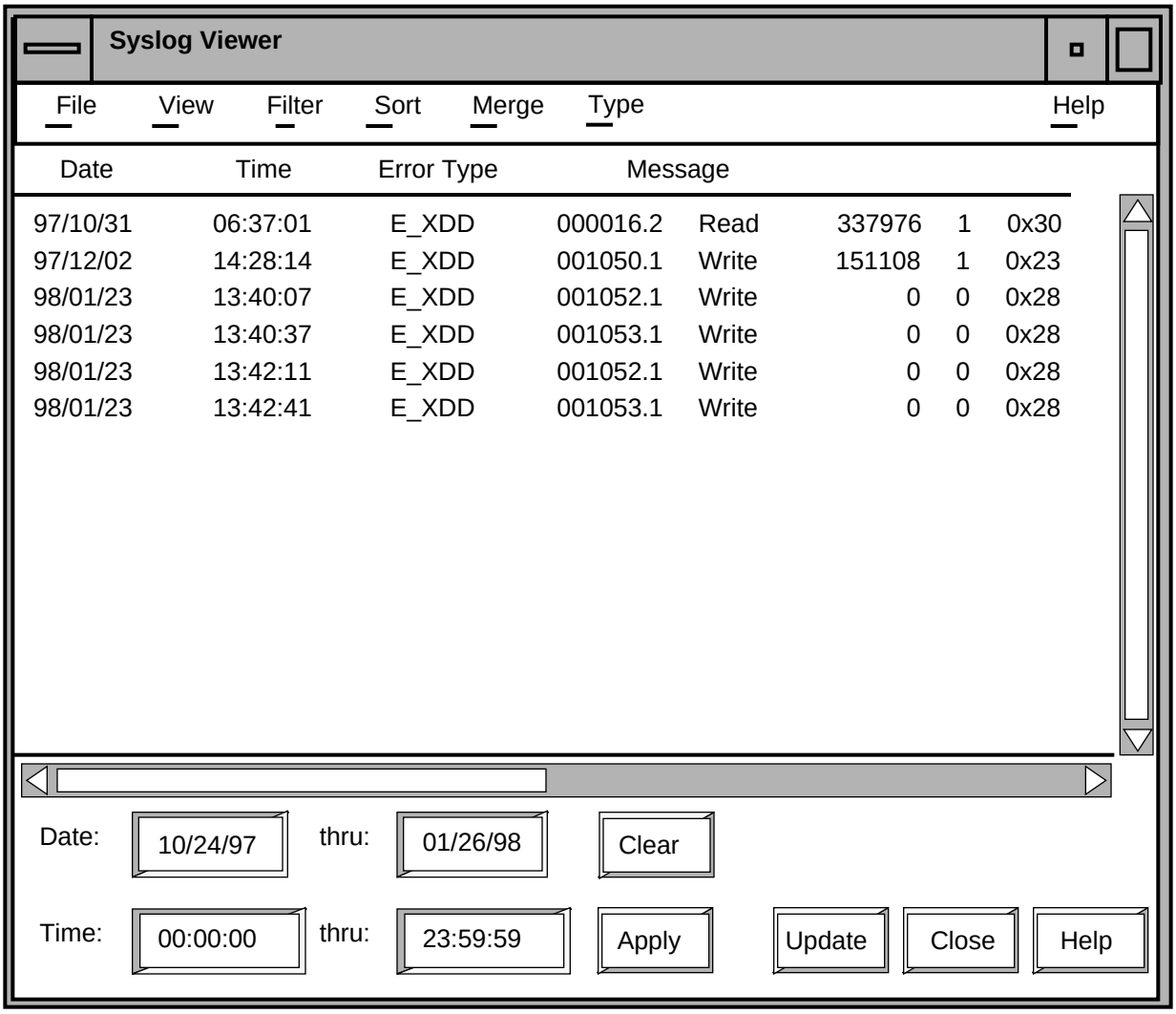
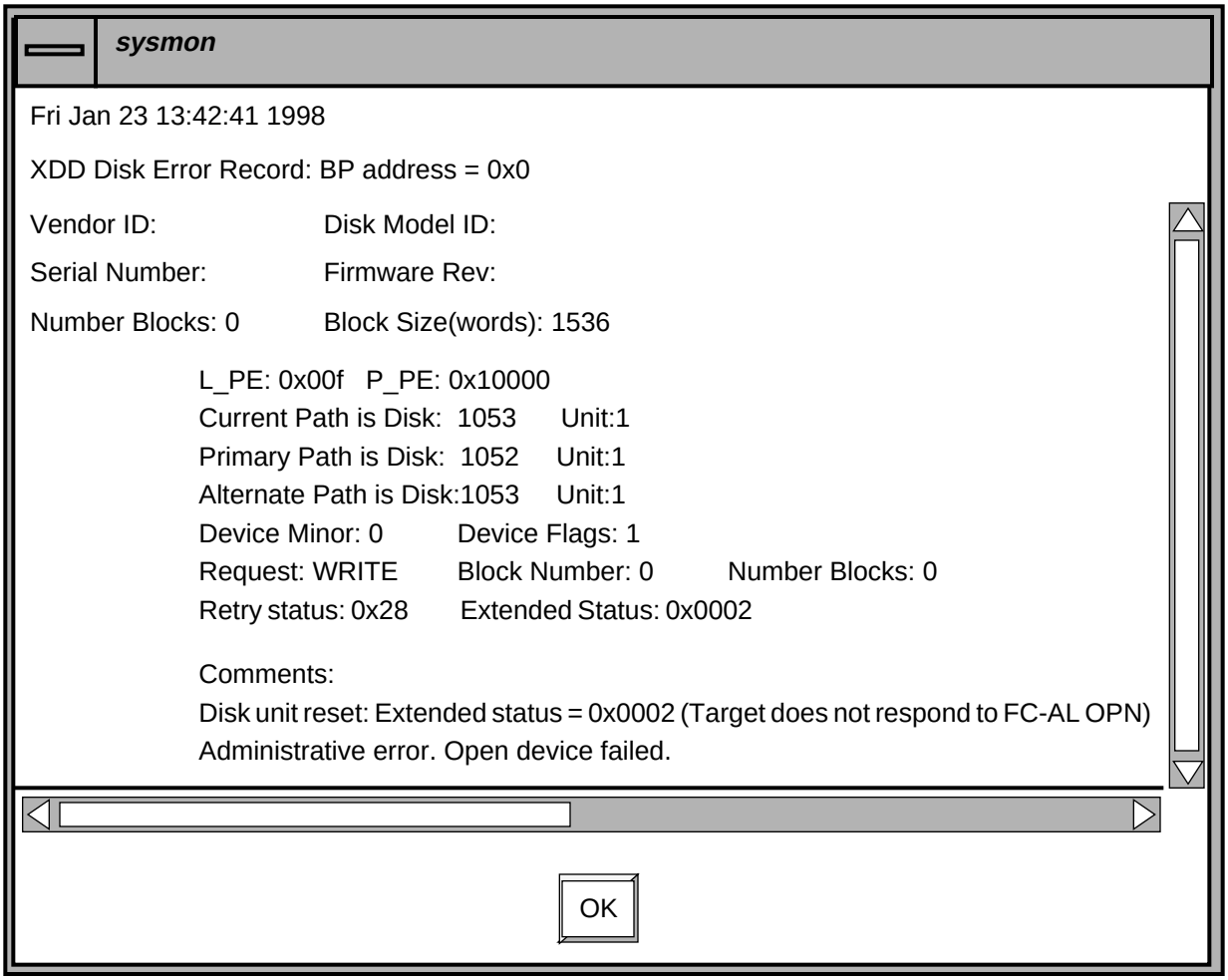


Figure 33. errpt Information Using the E_XDD Filter Option



To access more information about an entry, double click on the entry or select the entry and click the Help button (refer to Figure 34).

Figure 34. Detailed Information about an Entry



Monitor the Warning and Control System

The `nwacsuser` command allows you to monitor the warning and control system (WACS). The `nwacsuser` command has the following format:

```
nwacsuser [-option]
```

The `nwacsuser` command accepts the following options:

`-alarm on`
Activates the alarm. The alarm (bell) rings to notify you of faults or warnings. You can silence the alarm by clicking the SELECT mouse button on the Alarm On button; however, the alarm rings again when the next fault or warning occurs.

`-alarm off`
Disables the alarm.

`-nolog`
Disables the logging of faults and warnings to the /cri/mws/wacs/wacslog file.

`-log file_name`
Writes faults and warning log information to the file specified by `file_name`. Use the complete pathname.

`-title window_name`
Writes the name specified by `window_name` in the title bar of the nwacsuser window. The default title is WACS.

`-host mwsname`
Specifies which host machine is running the nwacs program. The default host is the same host that is running `nwacsuser`.

`-dump dump_interval [time_span]`
Automatically dumps data at the interval specified by `dump_interval`. The `time_span` parameter is optional; it defines the duration for which the dump interval will occur. For example, the command `nwacsuser -dump 5 600` dumps data every 5 minutes for a span of 10 hours (600 minutes). The `-dump` option enables you to dump data automatically at constant time intervals. If you do not define the time span, the dumps continue at the specified interval for as long as `nwacsuser` continues to run. The data dumps are appended to the /cri/mws/wacs/wacs.dump file.

`-maxlogsize nbytes`
Specifies the maximum size of the log file in decimal; the default is 100000.

`-s`
Runs `nwacsuser` in secure mode. This mode disables the Write Switches button from the View Switches menu. In this mode, you cannot change or write any switch values.

Submit and Execute Online Diagnostics

Three online diagnostic tests are available for the CRAY T3E system: olmeme, olperi, and olnxe. These tests are confidence tests. For more information about these online diagnostics, refer to the man pages.

To submit and execute an online diagnostic test, use the `mpprun` command. The `mpprun` command accepts the following options and arguments:

`-l lpe`
Executes the diagnostic on the logical PE that you specify with the `lpe` option. The PE that you specify becomes the base PE for the partition.

Note: In order to use the `lpe` option, the Diag Perm bit must be set in the user database (UDB).

`-n npes`
Executes the diagnostic with the number of PEs that you specify with the `npes` option. If you specify a value of one for `npes`, the diagnostic runs in a single command PE. If the value you specify for `npes` is two or greater, the diagnostic runs in the specified number of application PEs.

`diagnostic_name`
Name of the diagnostic to execute.

`arg...`
Arguments that are specific to the diagnostic that will execute.

For example, to execute olmeme in 4 PEs with logical PE 0 as the base PE of the partition (also referred to as the target PE), type the following command:

```
mpprun -l 0 -n 4 olmeme
```

You can submit the online diagnostics without using the `mpprun` command; however, when you do so, the online diagnostic runs in one command PE that the operating system selects. Table 7 lists the default partition sizes for the online diagnostics when you do not use the `mpprun` command to submit the online diagnostics. Table 7 also lists partition size information for the online diagnostics when you do use the `mpprun` command to submit the online diagnostics.

Table 7. Recommended Partition Sizes for the Online Diagnostics

Diagnostic	Default Partition Size When Not Submitted with the mpprun Command	Comments
olmeme	1 command PE	Runs effectively on any partition size
olnxe	1 command PE	Runs most effectively on multiple-PE partitions
olperi	None (needs to be submitted with mpprun)	Requires a partition that contains a minimum of 3 PEs

Dump the System Memory

Two procedures are available to dump the contents of the CRAY T3E memory:

- Mainframe resident procedure
- SWS resident procedure

The SWS resident procedure is the most commonly used procedure.

SWS Resident Procedure

For the SWS resident procedure, perform the following steps:

1. Configure the dump mode in the options file (by default, /opt/config/options).

   The `dumpsys` command supports six modes. By selecting the proper mode you can:

   - Choose either the SWS file system or the mainframe file system as the target file system for the dump procedure
   - Choose to dump the memory of all PEs or selected PEs

   The dump modes are:

   - `default` - Dumps the memory of all PEs to the mainframe file system
   - `pelist` - Prompts you for a list of PEs to be dumped to the mainframe file system
   - `manual` - Dumps the boot PE to the dump directory (by default /opt/CYRIdump/mf) on the SWS; reboots the CRAY T3E system in maintenance mode; and prompts you in the Cray console window for the dump configuration
   - `all-sws` - Dumps the memory of all PEs to the SWS dump directory
   - `pelist-sws` - Prompts you for a list of PEs to dump to the dump directory on the SWS

- `single-sws` - Prompts you for a specific PE to dump to the dump directory on the SWS

For more information about the dump modes, refer to the man page of the `dumpt3e` command.

Note: When the `dumpsys` command prompts you for a PE list, enter a list of logical PE numbers that are separated by commas or enter flags that indicate that all operating system PEs or command PEs should be dumped. The flags are `os` (operating system PEs) and `cmd` (command PEs).

To determine the dump mode, the `dumpsys` command reads the topology and options files. The `dumpsys` command retrieves the dump mode and passes it to a low-level command, `dumpt3e`.

The `dumpt3e` command dumps the appropriate files to a subdirectory within the dump directory. `dumpt3e` identifies this subdirectory by using the following naming convention: mmddhhmm. The first mm is the month; dd is the day of the month; hh is the hour (24-hour clock); and the second mm is the minute of the hour.

The following example shows an options file that has the dump mode set to `all-sws` and the boot/dump PE set to PE 7.

```
sn6505 boot -B 7
sn6505 dump -B 7 -M all-sws
sn6505 halt -B 7
sn6505-mpn0 boot
```

2. Dump the CRAY T3E system.

   The following command dumps a specific component (resource):

   ```
   dumpsys <sn6505>
   ```

   The following command inhibits boot resources after the dump:

   ```
   dumpsys -B
   ```

   The following command dumps all I/O nodes that are configured in the topology file and inhibits the dumping of mainframes:

   ```
   dumpsys -M
   ```

dumpsys Command

The `dumpsys` command performs the following tasks:

- Determines which memory segments to dump

  Because of the large amount of CRAY T3E global memory, the `dumpsys` command tries to minimize the amount of disk space that it needs to dump the contents of the CRAY T3E memory. It does so by comparing the code segments of all PEs to the code segments of a master PE. When the code segments match the master PE, the `dumpsys` command does not dump those segments; the dump analysis tools (`crashmk`) use the code segments of the master PE. The `dumpsys` command dumps the code segments that do not match the code segments of the master PE (for example OS PEs, command PEs, and application PEs that have corrupted memory).

- Halts the CRAY T3E system to preserve the memory and register states
- Ensures that the GigaRing channel is running
- Dumps the boot PE to the SWS using DMA transfers
- Boots the boot PE in maintenance mode

  Note: Booting the boot PE in maintenance mode corrupts memory; therefore, prior to this boot, the `dumpsys` command dumps the boot PE's memory to the SWS using DMA transfers.

- Dumps the nonboot PEs

Create a Dump Tape

Because of the different dump modes, you must prepare a dump tape before sending the dump for analysis or use the file transfer program (ftp) to transfer the appropriate files for analysis.

Whether you create a tape or use ftp, you must include the following files for analysis:

- Dump files that the `dumpsys` command created
- A current copy of the UNICOS/mk archive file that was used to boot the system when it crashed
- A current copy of the crashmk program file that comes with the UNICOS/mk archive

There are numerous ways to create a dump tape. The following text shows an example of one way to create a dump tape.

Note: In this example, the `dumpsys` command dumped the memory of all PEs to the dump directory on the SWS (/opt/CYRIdump/mf/03041200.sn6505).

1. Use the following commands to copy the UNICOS/mk archive file and the crashmk program file to the dump directory /opt/CYRIdump/mf/03041200.sn6505. The UNICOS/mk archive file and the crashmk program file are in the /opt/CYRIos/1.0/archive directory.

```
cp /opt/CYRIos/1.0/archive/crashmk-T3E
\ /opt/CYRIdump/mf/03041200.sn6505

cp /opt/CYRIos/1.0/archive/unicosmk.cray-t3e.unstrp
\ /opt/CYRIdump/mf/03041200.sn6505
```

2. Copy the dump directory to a tape.

   Note: For this example, the dump files are copied to the SWS digital audio tape (DAT).

   To copy the files to the SWS DAT, use the following steps:

   a. Insert a DAT cartridge into the DAT unit.

   b. Type the following command to copy all of the dump files to the DAT as a tape archive:

      ```
      tar cvf /dev/rmt/0
      ```

3. Write down the command that you used to create the tape and include it with the tape.

4. Send the tape to the appropriate service group.

Shut Down the Operating System

To shut down (halt) UNICOS/mk, perform the following steps:

1. In the Cray console window, login as `root`.
2. Start the shutdown sequence by typing the following command:

   `shutdown [grace period]`

3. At the UNICOS/mk prompt for single-user mode (sn*xxxx#*), type the following command:

   `sync`

4. In the Cray console window, type:

   `Control key then right bracket (])`

5. Type the following command:

   `haltsys sn`*`xxxx`
   This command halts the CRAY T3E system only.

   or

   `haltsys`

   When the halt is complete, a halt complete message appears in the window.

Boot UNICOS/mk in Degraded Mode

The system runs in degraded mode after you disable a communication link, PE, or node. To degrade a communication link, PE, or node, perform the following steps:

1. At the prompt, type **t3ems**.

   Note: t3ems is located in the /opt/CYRIdiag/t3e/bin directory.

2. Disable the appropriate link, PE, or node using the T3EMS PE Configuration window (refer to Figure 35).

3. Save the configuration by clicking on the Save button in the T3EMS System Configuration window. t3ems writes the configuration information to the t3e_config file.

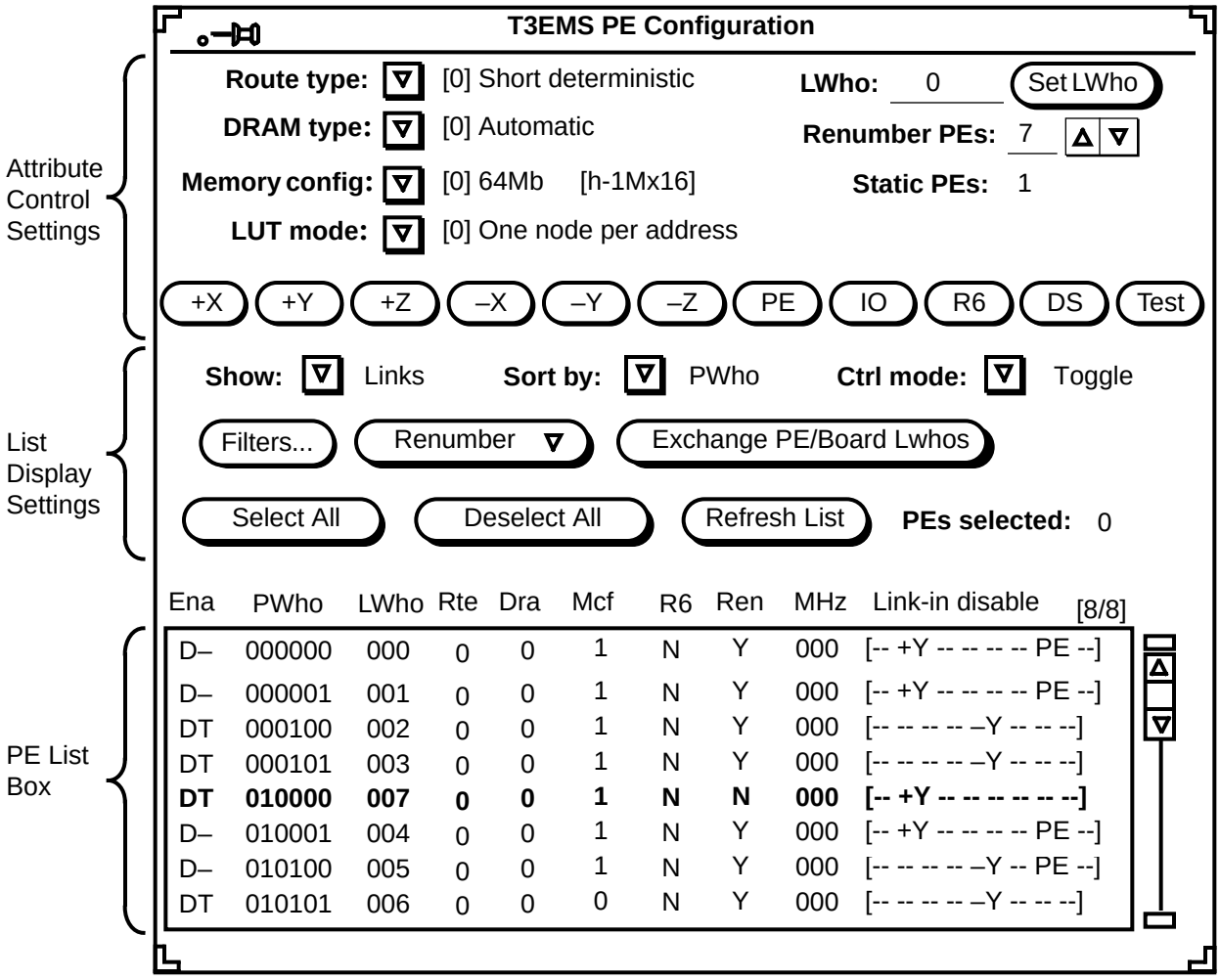
   Note: When you disable a PE or node, t3ems renumbers the PEs (reassigns logical PE numbers) after you save the configuration. This renumbering also occurs when you click on the Renumber button in the T3EMS PE Configuration window.

   For more information about how to disable a communication link, a PE, or a node, refer to the *Offline Maintenance* document, publication number HMM-158-B.

4. Exit t3ems and reboot the system using the `bootsys` command.

Note: Always use t3ems to make changes to the hardware configuration file before you initiate the `bootsys` command.

Figure 35. Disabled PEs Shown in the T3EMS PE Configuration Window



The `bootsys` command executes the `t3e_hdw_boot` command. The `t3e_hdw_boot` command uses the information from the t3e_config file to initialize the system. When a communication link is disabled, the `t3e_hdw_boot` command recalculates the routing tags so that the PEs route information around the disabled link.

The global resource manager (GRM) and an auto-edit feature also use the t3e_config file during the boot process. The GRM uses the t3e_config file to determine when a communication link is disabled. When a communication link is disabled, the GRM does not allow barrier partitions to use the disabled link.

When you disable a communication link, a PE, or a node, the auto-edit feature reads a copy of the software configuration file (/opt/os/1.0/config/param.conf) into memory and compares it to the hardware configuration file (t3e_config). When the two files differ, the auto-edit feature edits the copy of the software configuration file to indicate the disabled hardware; however, it does not write the changes to the original software configuration file.

The auto-edit feature is enabled by default. Figure 36 shows a sample output of the boot process when the auto-edit feature is enabled and four PEs are disabled. For more information about the auto-edit feature, refer to the *UNICOS/mk Configuration Reference* document, publication number SG-2603.

Figure 36. Sample Console Output When the Auto-edit Feature Is Enabled

```
(Site:0x007) CS: init_tables: Number of PEs configured in param file = 8
(Site:0x007) NOTICE: 20:30:51 (config) Begin auto-edit
(Site:0x007) NOTICE: 20:30:51 (config) t3e_config phys_npes=8
(Site:0x007) NOTICE: 20:30:51 (config) Setting PE_NOBOOT (PE link), LPE 0x6, PPE 0x0
(Site:0x007) NOTICE: 20:30:51 (config) Setting PE_NOBOOT (PE link), LPE 0x5, PPE 0x1
(Site:0x007) NOTICE: 20:30:51 (config) Setting PE_NOBOOT (PE link), LPE 0x4, PPE 0x10001
(Site:0x007) NOTICE: 20:30:51 (config) Setting PE_NOBOOT (PE link), LPE 0x3, PPE 0x10100
(Site:0x007) CS: Total PEs:        1 OS, 3 command, 4 application
(Site:0x007) CS: Disabled PEs: - 0 OS, 3 command, 1 application
(Site:0x007) CS:                   ---      ---              ---
(Site:0x007) CS: Enabled PEs: =  1 OS, 0 command, 3 application
(Site:0x007) NOTICE: 20:30:51 (config) Requested minimum number command PEs = 1
(Site:0x007) NOTICE: 20:30:51 (config) Requested minimum number of application PEs = 2
(Site:0x007) NOTICE: 20:30:51 (config) Command PE 0x6 is disabled, checking:
(Site:0x007) NOTICE: 20:30:51 (config) Swap LPEs 0x6 and 0x0
(Site:0x007) NOTICE: 20:30:52 (config) Command PE 0x5 is disabled, checking:
(Site:0x007) NOTICE: 20:30:52 (config) Reached minimum number of application PEs:
(Site:0x007) NOTICE: 20:30:52 (config) Command PE 0x5 will remain disabled
(Site:0x007) NOTICE: 20:30:52 (config) Command PE 0x4 is disabled, checking:
(Site:0x007) NOTICE: 20:30:52 (config) Reached minimum number of application PEs:
(Site:0x007) NOTICE: 20:30:52 (config) Command PE 0x4 will remain disabled
(Site:0x007) NOTICE: 20:30:52 (config) routes [0] .io_contr_ppe=0x10000
(Site:0x007) NOTICE: 20:30:52 (config) Using PPE 0x10000 for I/O controller, routes [0], ring 0
(Site:0x007) NOTICE: 20:30:52 (config) Setting routes [0] .io_contr_lpe=0x7
(Site:0x007) WARNING: 20:30:52 (config ) Ring 1: Both io_contr_ppe and io_contr_lpe are zero; using io_contr_ppe
(Site:0x007) NOTICE: 20:30:52 (config) routes [1] .io_contr_ppe=0x0
(Site:0x007) WARNING: 20:30:52 (config) Can't use PPE 0x0 for I/O (PE_NOBOOT)
(Site:0x007) WARNING: 20:30:52 (config) Can't use PPE 0x1 for I/O (PE_NOBOOT)
(Site:0x007) NOTICE: 20:30:52 (config) Using PPE 0x100 for I/O controller, routes [1], ring 1
(Site:0x007) NOTICE: 20:30:52 (config) Setting routes [1] .io_contr_lpe=0x0
(Site:0x007) CS: Total PEs:        1 OS, 3 command, 4 application
(Site:0x007) CS: Disabled PEs: - 0 OS, 2 command, 2 application
(Site:0x007) CS:                   ---      ---              ---
(Site:0x007) CS: Enabled PEs: =  1 OS, 1 command, 2 application
(Site:0x007) NOTICE: 20:30:52 (config) End auto-edit
(Site:0x007) NOTICE: 20:30:52 (config) Starting 8 PEs
(Site:0x007) CS: osboot: LPE 0x3 not booted (PE_NOBOOT)
(Site:0x007) CS: osboot: LPE 0x4 not booted (PE_NOBOOT)
(Site:0x007) CS: osboot: LPE 0x5 not booted (PE_NOBOOT)
(Site:0x007) CS: osboot: LPE 0x6 not booted (PE_NOBOOT)
(Site:0x007) NOTICE: 20:30:52 (config) Server initialization complete

 Lines cut....

 (Site:0x007) GRM: CS reports 8 physical PEs
(Site:0x007) NOTICE: 20:30:52 (grm) Boot logical PE 0x007
(Site:0x007) NOTICE: 20:30:52 (grm) LPE 0x003 not booted (PE_NOBOOT)
(Site:0x007) NOTICE: 20:30:52 (grm) LPE 0x004 not booted (PE_NOBOOT)
(Site:0x007) NOTICE: 20:30:52 (grm) LPE 0x005 not booted (PE_NOBOOT)
(Site:0x007) NOTICE: 20:30:52 (grm) LPE 0x006 not booted (PE_NOBOOT)
(Site:0x007) GRM: Setting BootSync hack for PEs 0x000 to 0x007
(Site:0x007) GRM: App. region starts at logical PE 0x001, length 2
(Site:0x007) GRM: 8 PEs configured.
(Site:0x007) WARNING: 20:30:52 (grm) Disabled 8 inter-PE links

(Site:0x007) GRM: PE allocator initialized
(Site:0x007) NOTICE: 20:30:52 (grm) Server initialization complete
```

Advanced Troubleshooting Techniques

The following information provides troubleshooting techniques that may help you isolate the cause of a failure.

Run t3ebrd.pgm to Analyze the Routes of a Degraded System

When you map out a PE or a node, you can cause route problems. To determine whether a route problem exits, run the t3ebrd.pgm script. The t3ebrd.pgm script analyzes a degraded system for route problems and automatically modifies the configuration (optimally, disabling the fewest number of nodes and PEs) to correct the route problems.

When the t3ebrd.pgm script detects a route problem, you may be able to correct the problem without having to disable healthy PEs/nodes by moving the processing element modules (PEMs) to different slots within the cabinet. Run the check_route.pgm script to determine configuration alternatives.

What to Do When One or More PEs Do Not Respond during a Deadstart

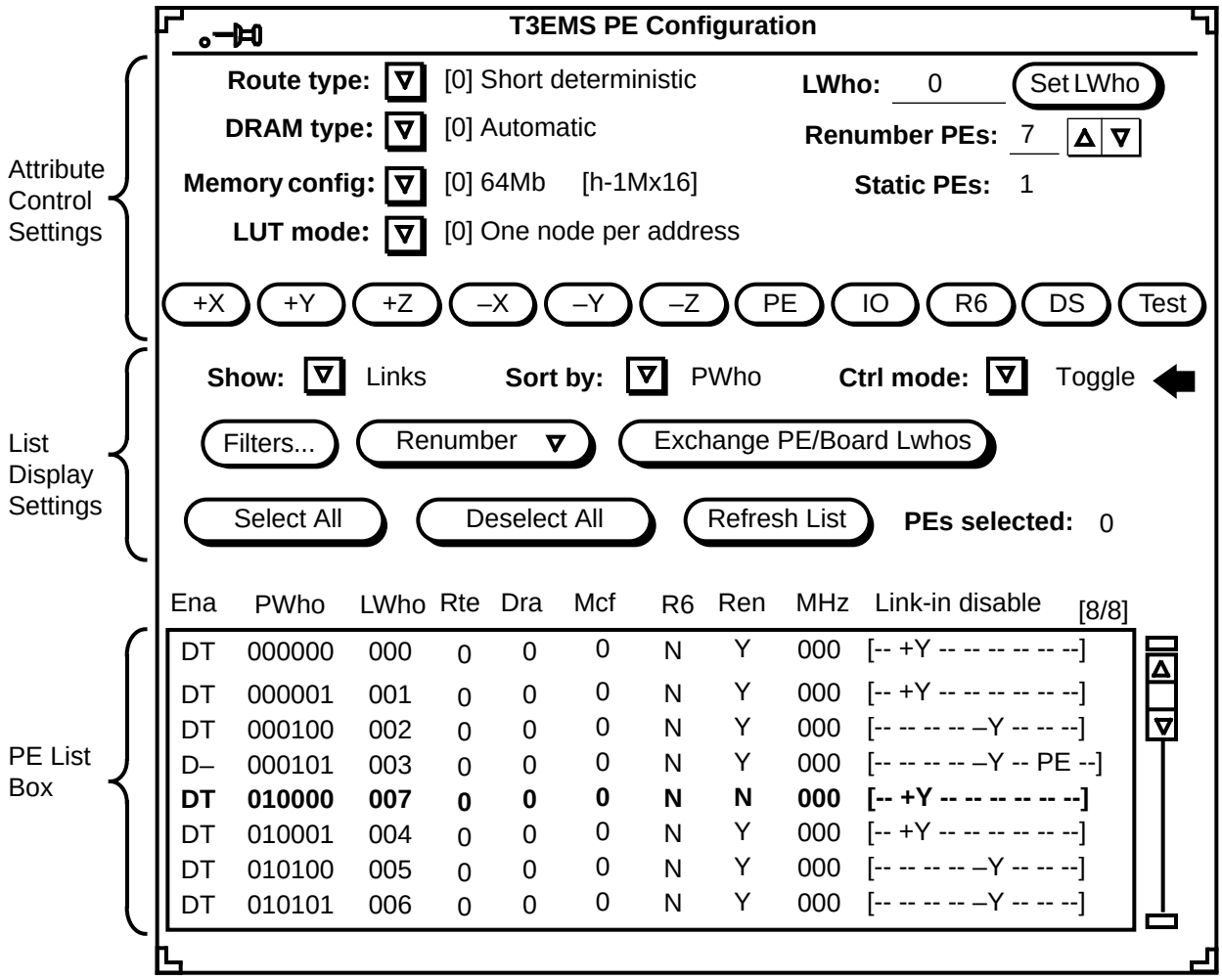
When one or more PEs do not respond during the deadstart sequence and t3ebrd.pgm did not isolate the failing component, map out the PE by disabling the PE communication link. To disable the PE communication link, perform the following steps:

1. Select PE Configuration from the Config menu.
2. When the T3EMS PE Configuration window appears, select Toggle for the Ctrl mode (refer to Figure 37).
3. Select the failing PE from the PE list box and click PE.

Note: When you want to save your change to the configuration file, select System Configuration from the Config menu. When the T3EMS System Configuration window appears, click Save.

4. Deadstart t3ems.

Figure 37. Disabling the Failing PE

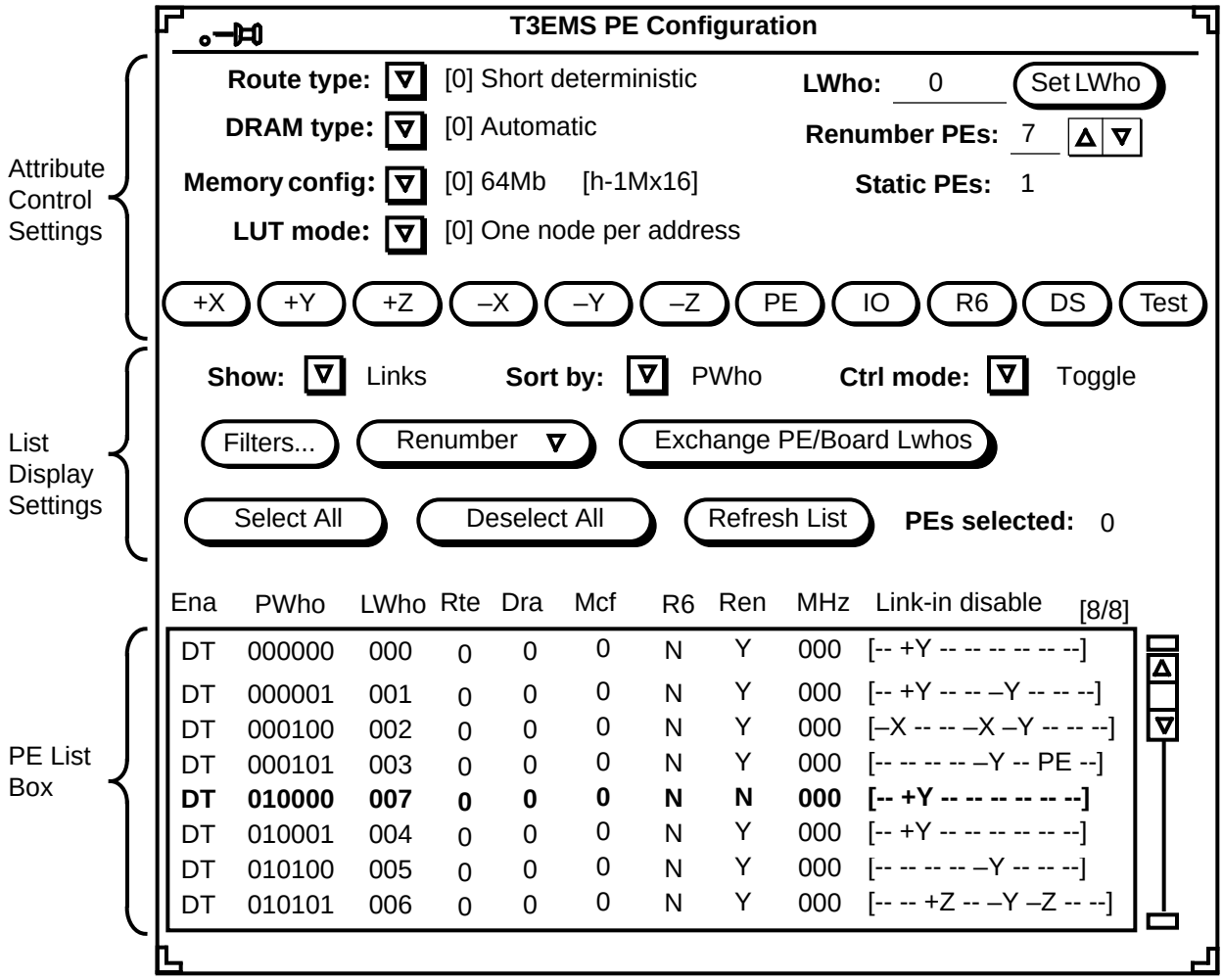


If the system still does not deadstart, disable the entire node.

Select the failing PE in the PE list box and click DS (refer to Figure 38).

To save this change, select System Configuration from the Config menu. When the T3EMS System Configuration window appears, click Save.

Figure 38. Disabling the Entire Node

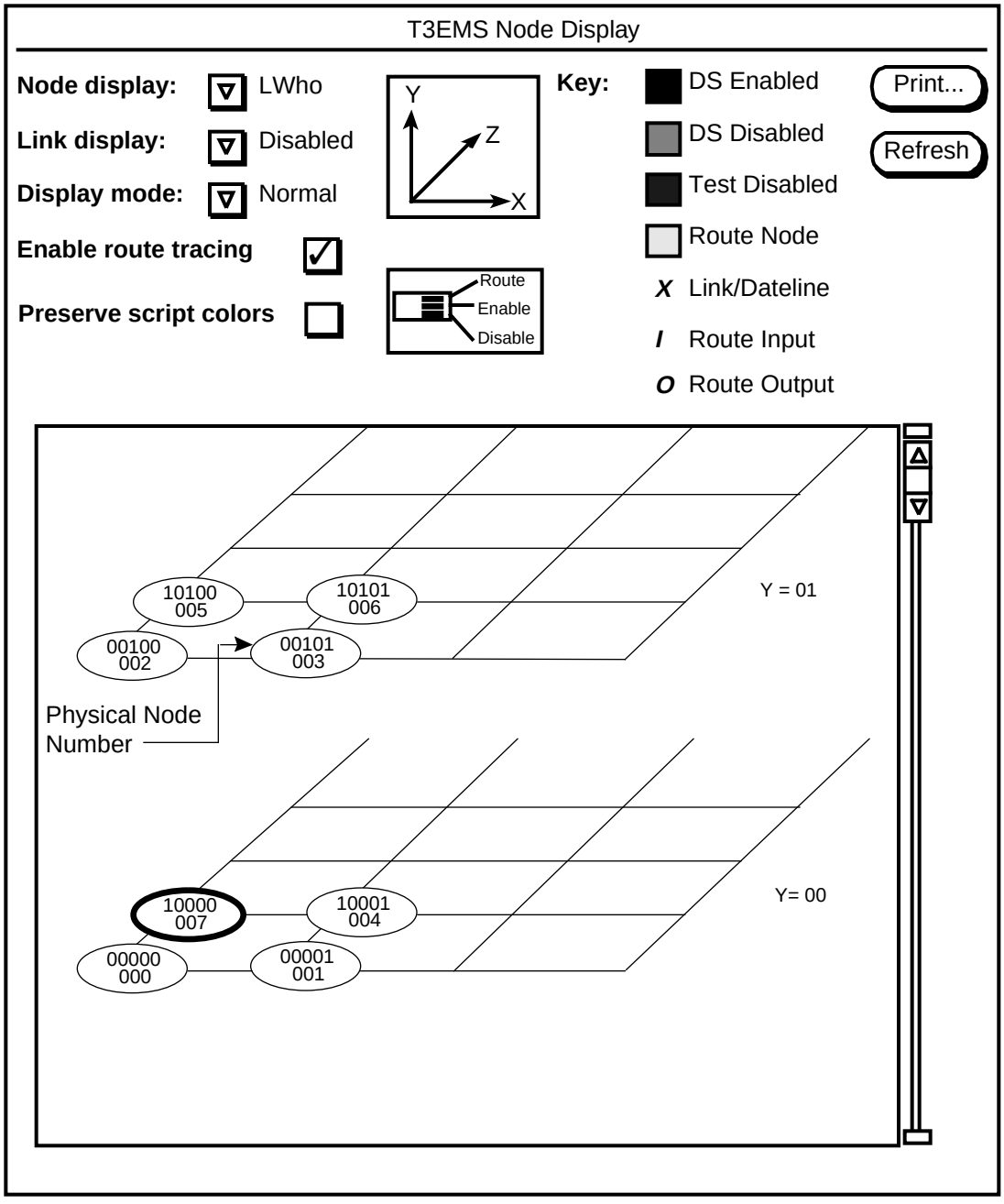


When disabling the node fixes the problem, the problem may be in the interconnect network. To determine why the interconnect network is failing, look for common routes between the failing PEs or between the failing PE and the boot PE. After you determine the common links, disable the common links, one at a time, until the problem no longer exists.

To determine which link or network router is bad, use the T3EMS Node Display window to identify the physical node number of the logical PE (refer to Figure 39). In this example, logical PE 3 did not respond; the physical node number of logical PE 3 is 00101.

Note: t3ems highlights (bolds) the boot node as shown in Figure 39. For this example, the boot node is 10000.

Figure 39. T3EMS Node Display Window



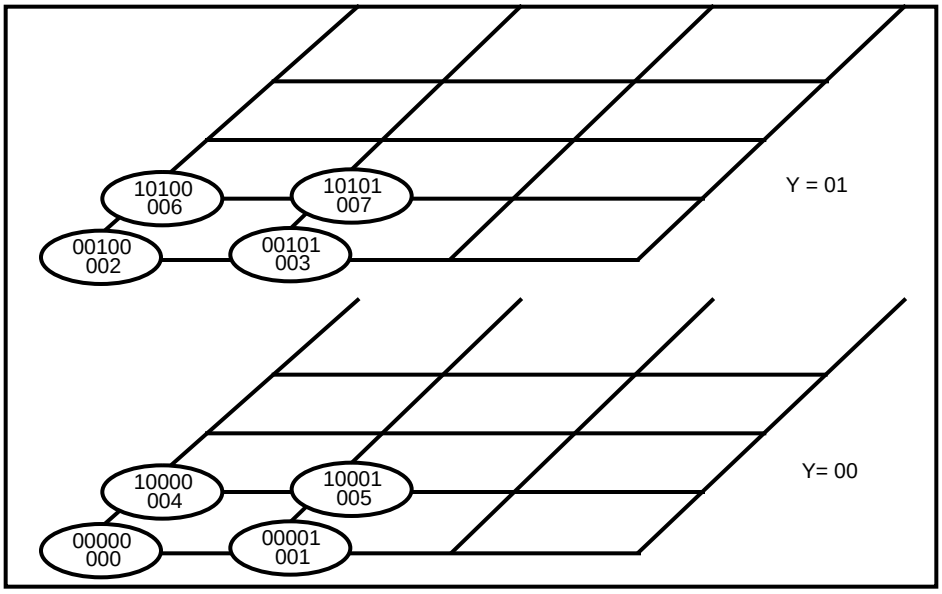
To trace the paths between the failing PE and the boot PE, use the T3EMS Node Display window (click on the failing PE and click on the boot PE) or use the T3EMS Route Tracing window. For this example, the packet travels from

the boot PE in the +X dimension, the +Y dimension, and the –Z dimension to the failing PE. From the failing PE to the boot node, the packet travels in the +Z dimension, the –X dimension, and the –Y dimension.

At this point, it is unknown whether the boot PE cannot communicate with the failing PE or the failing PE cannot communicate with the boot PE; therefore, examine some of the other routes to determine whether the links to/from the failing PE are used to communicate with other PEs.

For example, determine the paths between the boot node and PE 10101 (refer to Figure 40).

Figure 40. Paths between the Boot Node and PE 10101




```
Base: 10000   Target:  10101
Mode:0        Route:   0000000300810101
        -> PE   010000 -> +X
        -> +X   010001 -> +Y
        -> +Y   010101 -> PE

Base: 10101   Target:  10000
Mode: 0       Route:   0000000300818080
        -> PE   000101 -> –X
        -> –X   000100 -> –Y
        -> –Y   000000 -> PE
```

The paths between the boot node and PE 10101 are the same in the X and Y dimensions as the paths between the boot node and the failing PE; therefore, because the boot node was able to communicate with PE10101, you can conclude that the X and Y dimension communication links are functioning properly.

The two links that you cannot test are the +Z and –Z communication links. To determine which link is bad, disable the +Z or the –Z communication link by performing the following steps:

1. Select PE Configuration from the Config menu.
2. When the T3EMS PE Configuration window appears, select PE 00101 from the PE list box and click the +Z or –Z.

Note: When you want to save your change, select System Configuration from the Config menu. When the T3EMS System Configuration window appears, click Save.

3. Deadstart t3ems.

If PE 00101 does not respond, use the T3EMS PE Configuration window to reenable the disabled link, disable the other link, and deadstart t3ems.

Index

B

C

D

E

F

G

U

W

X