Sun™ StorEdge™ A7000 VME Dual Channel ESCON Diagnostics Reference Manual

THE NETWORK IS THE COMPUTER™

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

Part No. 805-4900-10
October 1998, Revision A

Send comments about this document to: `docfeedback@sun.com`

Please
Recycle

Contents

Tables

Preface

Sun StorEdge A7000 VME Dual Channel ESCON Diagnostics Reference Manual is specific to the Extended Diagnostic provided for testing the VME Dual Channel ESCON board. This manual contains the following information:

- Program initialization procedures
- Test descriptions
- Descriptions of messages produced by this program

How This Book Is Organized

Chapter 1 “Diagnostic Overview” describes the Extended Diagnostic used for testing the VME Dual Channel ESCON boards.

Chapter 2 “Program Initialization” describes the initialization options provided when running this diagnostic in interactive mode.

Chapter 3 “Program Tests” describes the individual diagnostic tests.

Chapter 4 “Program Messages” describes the messages produced by the diagnostic during program execution.

Typographic Conventions

TABLE P-1 Typographic Conventions

Typeface or Symbol	Meaning	Examples
AaBbCc123	The names of commands, files, and directories; on-screen computer output.	`Do you need help? Enter Y or N (CR)`
AaBbCc123	What you type, when contrasted with on-screen computer output.	`Enable External Loopback testing = N ;[cr,?,^,Y,N]?`**`Y`**
AaBbCc123	Book titles, new words or terms, words to be emphasized. Variable expressions; replaced with a real name or value.	Read Chapter 2 in the *Sun StorEdge A7000 ROM Monitor Reference Manual.* `slot` *`n`*`:Loading extended image escondiag`
[ ]	In system output examples, brackets indicate optional values. If several values are placed inside brackets, any or none of them can be displayed. Brackets are also used in system prompts to enclose the response choices.	In the following example, displaying the slot number is optional: `[Slot` *`n`*`:]`

Related Documentation

TABLE P-2 Related Documentation

Type	Title
User interface	*Sun StorEdge A7000 ROM Monitor Reference Manual*
Diagnostic reference	*Sun StorEdge A7000 Diagnostics Reference Manual*
Hardware reference	*Dual Channel VME to ESCON Interface Engineering Summary*

Note – The *Dual Channel VME to ESCON Interface Engineering Summary* provides additional information about the hardware, software, and diagnostics. Refer to this manual before running diagnostics to verify that the board jumpers and switch settings are correct.

Sun Documentation on the Web

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

`http://docs.sun.com`

Sun Welcomes Your Comments

We are interested in improving our documentation and welcome your comments and suggestions. You can email your comments to us at:

docfeedback@sun.com

Please include the part number of your document in the subject line of your email.

CHAPTER 1

Diagnostic Overview

Introduction

The VME Dual Channel ESCON Diagnostic is an Extended Diagnostic used to verify the functionality of the VME Dual Channel ESCON boards. This program verifies the:

- Ability of the board to execute a `Reset Channel` command.
- Operation of the VME interrupt lines.
- Ability to run onboard diagnostic tests on the board in internal loopback mode.
- Ability to run onboard diagnostic tests on the board in external loopback mode.
- Ability to download data and microcode into the board.
- VME DMA functions to Expansion Memory.
- Failover feature.
- Channel/Control Unit functionality.

This is a disk based diagnostic normally executed during the power up testing sequence. If configured, the program is automatically called from the ROM Monitor after the Built-in Self Tests (BISTs) are executed. You may also use individual ROM Monitor commands to execute this diagnostic. Refer to the *Sun StorEdge A7000 Diagnostics Reference Manual* for extended diagnostic commands and execution procedures.

Testing Considerations

- Use Control-C to terminate an operation in interactive mode. Any environment variable changes made under the control of this diagnostic are preserved when the Control-C sequence is initiated.
- Use the Escape key to abort an Extended Diagnostic and return control to the Interactive Mode Options Menu. Any environment variable changes made under the control of this diagnostic are preserved when the escape sequence is initiated.
- External Loopback testing requires the installation of an ESCON Optical Loopback Plug, part number 130-4012-01, for Tests 4 and 10.
- Channel / Control Unit testing requires the installation of an ESCON Duplex Fiber Optic cable, part number 530-2617-01, for Tests 11 through 37.

CHAPTER 2

Program Initialization

Initialization Options

When running in interactive mode, the VME Dual Channel ESCON Diagnostic provides initialization options for:

- Enabling and disabling External Loopback testing
- Enabling and disabling Channel/Control Unit testing
- Downloading firmware files across the network
- Enabling and disabling milestone messages
- Changing the default subtest selection

Select the `Initialize Diag.` option from the Interactive Mode Options Menu to initialize the diagnostic. The interactive mode options are described in the *Sun StorEdge A7000 Diagnostics Reference Manual*.

Initialization Procedure

Once you select `Initialize Diag.` from the Interactive Mode Options Menu, the program displays a series of initialization prompts. The following responses are standard for all initialization prompts:

TABLE 2-1 Standard Prompt Responses

Response	Description
`cr`	Press the return key to select the default response displayed in the prompt.
`?`	Display help information.
`^`	Return to the Interactive Mode Options Menu.

1. The diagnostic first displays the following help information prompt:

```
Do you need help? Enter Y or N (CR)
```

   Enter `Y` to display a help message. Enter `N` or press the Return key to bypass displaying help information. The default response is `N`.

2. The program then displays:

```
Enable External Loopback testing = N  ;[cr,?,^,Y,N]?
```

   Enter one of the following or a standard response:

Response	Description
`Y`	Enables running the External Loopback test.
`N`	Inhibits running the External Loopback test.

   If you entered `N`, go to Step 4. If you entered `Y`, the program displays:

```
Attach optical loopback plug. Press RETURN when ready
```

Note – External Loopback testing requires operator intervention to install a loopback cable.

3. The program displays:

```
Get firmware files via the network = N  ;[cr,?,^,Y,N]?
```

Enter one of the following or a standard response:

Response	Description
Y	Reads the firmware files across the network.
N	Uses the firmware code loaded with the program.

If you entered N, go to Step 6. If you entered Y, the program displays:

```
Enter firmware filename = escon.firm ;[cr,?,^,filename]?
```

Enter a firmware filename or a standard response.

The program then displays:

```
Enter micro filename = escon.micro ;[cr,?,^,filename]?
```

Enter a micro filename or a standard response and go to Step 6.

4. If you entered N in Step 2, the program displays:

```
Enable Channel/Control Unit testing = N  ;[cr,?,^,Y,N]?
```

Enter one of the following or a standard response:

Response	Description
Y	Enables running the Channel/Control Unit tests.
N	Inhibits running the Channel/Control Unit tests.

If you entered N, go to Step 5. If you entered Y, the program displays:

```
Attach fiber optics cable. Press RETURN when ready
```

The program then displays:

```
Select Channel address of Control Unit to test = 0 ;[cr,?,^,0,1]?
```

Enter the desired channel address or a standard response.

Note – Channel testing requires operator intervention to install a fiber optics cable.

---

5. The program displays:

```
Get firmware files via the network = N  ;[cr,?,^,Y,N]?
```

Enter one of the following or a standard response:

Response	Description
Y	Reads the firmware files across the network.
N	Uses the firmware code loaded with the program.

If you entered N, go to Step 6. If you entered Y, the program displays:

```
Enter Control Unit firmware filename = cu_firm ;[cr,?,^,filename]?
```

Enter a firmware filename or a standard response.

The program displays:

```
Enter Control Unit micro filename = cu_micro ;[cr,?,^,filename]?
```

Enter a micro filename or a standard response.

The program displays:

```
Enter Channel firmware filename = chn_firm ;[cr,?,^,filename]?
```

Enter a firmware filename or a standard response.

The program displays:

```
Enter Channel micro filename = chn_micro ;[cr,?,^,filename]?
```

Enter a micro filename or a standard response.

The program displays:

```
Enter source ip address = 000.000.000.000 ;[cr,?,^,#.#.#.#]?
```

Enter the IP address where the files are located or a standard response.

6. The program displays:

```
Enable milestone messages = N  ;[cr,?,^,Y,N]?
```

Enter one of the following or a standard response:

Response	Description
Y	Enables displaying all milestone messages.
N	Inhibits displaying milestone messages.

7. The program displays:

```
Change default subtest selection = N  ;[cr,?,^,Y,N]?
```

Enter one of the following or a standard response:

Response	Description
Y	Enables looping on a specific subtest.
N	Inhibits looping on a subtest.

If you entered N, initialization is complete. If you entered Y, the program displays:

```
Enter subtest selection(s) = a  ;[cr,?,^,(c c...)]?
```

Enter the characters associated with the desired subtests or enter one of the standard responses. Valid subtests range from a through z.

Note – To run tests in interactive mode, use the `a`, `e`, or `t` option from the Interactive Mode Options Menu to select the desired tests. Then use the `r` option to start test execution. The interactive mode options are described in the *Sun StorEdge A7000 Diagnostics Reference Manual*.

CHAPTER 3

Program Tests

Test Summary

The VME Dual Channel ESCON Diagnostic contains the following tests:

TABLE 3-1 Diagnostic Tests

Test	Test Name
1	Channel Reset
2	VME Interrupts
3	On-board Diagnostics
4	External Loopback
5	Download, Init, & Ring Testing
6	VME/DMA Data Transfer
7	VME/DMA Variable Transfer Size
8	VME/DMA Block Transfer
9	Concurrent Channel I/O
10	Failover
11	Download Control Unit (CU) & Channel (Chnl) Firmware
12	Link Initialization
13	Channel : Write - Control Unit : Read
14	Data Transfer (Chnl :Write - CU :Read)
15	Variable Transfer Size (Chnl :Write - CU :Read)

TABLE 3-1 Diagnostic Tests *(Continued)*

Test	Test Name
16	Channel : Read - Control Unit : Write
17	Data Transfer (Chnl :Read - CU :Write)
18	Variable Transfer Size (Chnl :Read - CU :Write)
19	Channel : Wr/Rd - Control Unit : Wr/Rd
20	Validate Initialization Command
21	Immediate Command
22	Command Chaining
23	Command Chaining with Variable Transfer Size
24	Command Chaining with Variable DMA Block Size
25	Concurrent Command Chaining
26	Data Chaining
27	Data Chain - Gather & Scatter
28	Non-Contiguous Data Chaining
29	Disconnect/Reconnect
30	Split Transfer
31	Even and Odd Boundaries (Chnl :Write - CU :Read)
32	Even and Odd Boundaries (Chnl :Read - CU :Write)
33	Sense Command
34	Command and Data Chaining
35	Command and Data Chaining with Variable Size
36	Device Status
37	Incorrect Length

Note – Only Tests 1 through 3 and 5 through 9 are run during the power up testing sequence or in automatic mode. All tests can be run in interactive mode.

Milestone messages may be displayed during test execution. Refer to Chapter 4 for descriptions of the milestone messages.

Test 1 Channel Reset

This test issues a `Reset Channel` command to the board and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues.

The test then sets the interrupt enable bit in the Interrupt register to enable the board interrupts (workstation interrupts). The test writes the `Reset Channel` command into the Reset register, one byte at a time. After the board has processed a byte, it generates an interrupt. If no interrupt is generated or a timeout occurs, an error is reported.

If an interrupt is generated, the test checks the Interrupt Status Vector. If it is correct, the test then checks the OK and ERR bits in the Status register. If the OK bit is set, the test writes the next byte of the `Reset Channel` command into the Reset register. If the ERR bit is set, the test reports an error and terminates execution.

After sending the last byte of the `Reset Channel` command, the test checks the DIAG_OK bit in the Status register. If the bit is set, the channel reset was successful. If the bit is not set, an error is reported.

The test then disables the interrupt enable bit in the Interrupt register and clears the previous status by writing to the Status register. The test sends a `Reset Channel` command and verifies the result in the Status register. If no response is received from the channel reset, an error is reported.

These operations are repeated for each channel.

Test 2 VME Interrupts

This test verifies six VME interrupt levels and different combinations of status ID. The test issues a `Reset Channel` command to the board and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues.

The test loads different combinations of Status ID and request level information for the channel. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the status vector is correct, the test then verifies the status. Any error is reported.

After checking all interrupt levels, the test disables the interrupt enable bit in the Interrupt register. The test enables the interrupt on the timer vector. The test then sends a `Reset Channel` command and verifies the result in the status register. If no response is received from the channel reset, an error is reported. After testing is completed, the test restores the default interrupt states and ID levels.

These operations are repeated for each channel.

Test 3 On-board Diagnostics

This test verifies the ability to run the following on-board diagnostics, and verifies that the diagnostics execute successfully.

- Uart Local Loopback Test
- VIC Write/Read Register Test
- SLC Write/Read Register Test
- Buffer Memory Write/Read Test
- Interrupt Test
- ESCON Micro Diagnostics

The test issues a `Reset Channel` command to the board and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues.

The test then performs the following operations for each channel:

- Enables interrupts.
- Sends a `Diag` command to the board, one byte at a time, to run the on-board diagnostics.

If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test verifies the status. If an error occurs, the test reports the error.

Test 4 External Loopback

This test verifies the ability to run the External Loopback on-board diagnostics, and verifies that the diagnostics execute successfully. The test contains the following subtests:

Subtest	Subtest Name
a	Testing Channel 0
b	Testing Channel 1

The test issues a `Reset Channel` command to the board and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues.

The test then performs the following operations for each channel:

- Enables interrupts.
- Sends a `Diag` command to the board, one byte at a time, to start the External Loopback test.

If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test verifies the status. If an error occurs, the test reports the error.

Note – This test is bypassed unless you enabled External Loopback testing during program initialization.

Test 5 Download, Init, & Ring Testing

This test verifies the ability to download microcode and initializes and tests the ring structures. The test issues a `Reset Channel` command to the board and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues.

The test then performs the following operations for each channel:

- Downloads the appropriate microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Enables channel interrupts.
- Sends an `Offline Ring` command and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then checks that all parameters of the response ring are correct.
- Enables channel interrupts.
- Sends a `Halt Ring` command and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then checks that all parameters of the response ring are correct.
- Enables channel interrupts.
- Sends a `Reset` command. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status.
- Disables workstation interrupts.

Test 6 VME/DMA Data Transfer

This test verifies the ability to run the `Write` and `Read Diagnostic Ring` commands using different data patterns. The test contains the following subtests:

Subtest	Data Pattern
a	All Ones
b	All Zeros
c	A's & 5's (0x`AA55AA55`)
d	Address in Address

The test issues a `Reset Channel` command to the board and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues.

The test then performs the following operations for each channel:

- Downloads the appropriate microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Fills the output buffer with a data pattern.
- Initializes the input buffer.
- Enables channel interrupts.
- Sends a `Write Diagnostic Ring` command and verifies that the operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then checks that all parameters of the response ring are correct.
- Enables channel interrupts.
- Sends a `Read Diagnostic Ring` command and verifies that the operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then checks that all parameters of the response ring are correct.
- Verifies that the data transferred correctly.
- Enables channel interrupts.

- Sends a `Halt Ring` command and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then checks that all parameters of the response ring are correct.
- Enables channel interrupts.
- Sends a `Reset` command. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status.
- Disables workstation interrupts.

Test 7 VME / DMA Variable Transfer Size

This test verifies the ability to run the `Write` and `Read Diagnostic Ring` commands using different DMA block sizes. The test contains the following subtests:

Subtest	Block Size
a	1K DMA Block Size
b	2K DMA Block Size
c	4K DMA Block Size
d	8K DMA Block Size
e	16K DMA Block Size
f	32K DMA Block Size

The test issues a `Reset Channel` command to the board and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues.

The test then performs the following operations for each channel:

- Downloads the appropriate microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Fills the output buffer with a data pattern.
- Initializes the input buffer.

- Enables channel interrupts.
- Sends a `Write Diagnostic Ring` command and verifies that the operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then checks that all parameters of the response ring are correct.
- Enables channel interrupts.
- Sends a `Read Diagnostic Ring` command and verifies that the operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then checks that all parameters of the response ring are correct.
- Verifies that the data transferred correctly.
- Enables channel interrupts.
- Sends a `Halt Ring` command and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then checks that all parameters of the response ring are correct.
- Enables channel interrupts.
- Sends a `Reset` command. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status.
- Disables workstation interrupts.

Test 8 VME / DMA Block Transfer

This test verifies the ability to run `Write` and `Read Diagnostic Ring` commands starting at different boundaries. The test contains the following subtests:

Subtest	Subtest Name
a	Single Block Transfer (Boundary tx: wd0 rx: byte)
b	Single Block Transfer (Boundary tx: wd4 rx: byte)
c	Single Block Transfer (Boundary tx: wd8 rx: byte)
d	Multiple Block Transfer (Boundary tx: wd0 rx: byte)
e	Multiple Block Transfer (Boundary tx: wd4 rx: byte)
f	Multiple Block Transfer (Boundary tx: wd8 rx: byte)

Subtest	Subtest Name
g	Single Block Transfer (Boundary tx: byte rx: wd0)
h	Single Block Transfer (Boundary tx: byte rx: wd4)
i	Single Block Transfer (Boundary tx: byte rx: wd8)
j	Multiple Block Transfer (Boundary tx: byte rx: wd0)
k	Multiple Block Transfer (Boundary tx: byte rx: wd4)
l	Multiple Block Transfer (Boundary tx: byte rx:wd8)

The test issues a `Reset Channel` command to the board and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues.

The test then performs the following operations for each channel using different boundaries:

- Downloads the appropriate microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Fills the output buffer with a data pattern.
- Initializes the input buffer.
- Enables channel interrupts.
- Sends a `Write Diagnostic Ring` command and verifies that the operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then checks that all parameters of the response ring are correct.
- Enables channel interrupts.
- Sends a `Read Diagnostic Ring` command and verifies that the operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then checks that all parameters of the response ring are correct.
- Verifies that the data transferred correctly.
- Enables channel interrupts.
- Sends a `Halt Ring` command and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then checks that all parameters of the response ring are correct.
- Enables channel interrupts.

- Sends a `Reset` command. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status.
- Disables workstation interrupts.

Test 9 Concurrent Channel I/O

This test verifies the ability to run concurrent channel I/O. The test contains the following subtests:

Subtest	Subtest Name
a	On-board Diagnostics
b	VME/DMA Data Transfer
c	On-board Diags & VME/DMA Xfer Command
d	On-board Diags & VME/DMA Xfer Single Channel
e	On-board Diags & VME/DMA Xfer Across Channels

The test issues a `Reset Channel` command to the board and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues. The `Reset Channel` is repeated for each channel.

Subtest a

This subtest performs the following operations:

- Sends a `Diag` command to channel 0, one byte at a time, to run the Buffer Memory Write/Read on-board diagnostics. The subtest continues without waiting for the last interrupt to complete.
- Sends a `Diag` command to channel 1, one byte at a time, to run the Buffer Memory Write/Read on-board diagnostics. The subtest waits for the last interrupt to complete.

  If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.

Subtest b

This subtest performs the following operations:

- Downloads the appropriate microcode and sets up the ring structures for both channels. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Fills the output buffer with a data pattern.
- Initializes the input buffer.
- Enables channel 0 interrupts.
- Sends a `Write Diagnostic Ring` command to channel 0. The subtest continues without waiting for the command to complete.
- Enables channel 1 interrupts.
- Sends a `Write Diagnostic Ring` command to channel 1 and verifies that the operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then checks that all parameters of the response ring for both channels (0 and 1) are correct.
- Enables channel 0 interrupts.
- Sends the following `Diagnostic Ring` commands to channel 0: `Read`, `Write`, `Read`.
- Enables channel 1 interrupts.
- Sends the following `Diagnostic Ring` commands to channel 1 and verifies that the operations are successful: `Read`, `Read`, `Write`. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then checks that all parameters of the response ring for both channels are correct.
- Verifies that the data transferred correctly.

Subtest c

This subtest performs the following operations for each channel:

- Fills the output buffer with a data pattern.
- Initializes the input buffer.
- Sends a `Diagnostic Ring` command (0x05) to execute both the Buffer Memory Write/Read and VME/DMA Write/Read operations and verifies that the operations were successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then checks that all parameters of the response ring for both channels are correct.
- Verifies that the data transferred correctly.

Subtest d

This subtest performs the following operations:

- Displays the following message:

```
NOTE: During this subtest the active/idle leds will flash
      on the distribution panel.
```

- Enables channel interrupts.
- Sends a `Diagnostic Ring` command (0x06) to start buffer memory transfers and verifies that the operations were successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then checks that all parameters of the response ring are correct. This operation is repeated for each channel.
- Fills the output buffer with a data pattern.
- Initializes the input buffer.
- Enables channel interrupts.
- Sends a `Write Diagnostic Ring` command and verifies that the operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then checks that all parameters of the response ring are correct. This operation is repeated for each channel.
- Enables channel interrupts.

- Sends a `Read Diagnostic Ring` command and verifies that the operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then checks that all parameters of the response ring are correct. This operation is repeated for each channel.
- Verifies that the data transferred correctly.
- Enables channel interrupts.
- Sends a `Diagnostic Ring` command (0x07) to stop buffer memory transfers. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error. This operation is repeated for each channel.

Subtest e

This subtest performs the following operations:

- Enables channel 0 interrupts.
- Sends a `Halt Ring` command to channel 0 and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then checks that all parameters of the response ring are correct.
- Fills the output buffer with a data pattern.
- Initializes the input buffer.
- Sends a `Diag` command to channel 0, one byte at a time, to run the Buffer Memory Transfers on-board diagnostics. The subtest continues without waiting for I/O to complete.
- Enables channel interrupts.
- Sends a `Write Diagnostic Ring` command to channel 1 and verifies that the operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then checks that all parameters of the response ring are correct.
- Enables channel interrupts.
- Sends a `Read Diagnostic Ring` command to channel 1 and verifies that the operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then checks that all parameters of the response ring are correct.
- Verifies that the data transferred correctly.

- Sends a `Halt Ring` command to both channels and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then checks that all parameters of the response ring are correct.
- Enables channel interrupts.
- Sends a `Reset` command to both channels. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status.
- Disables workstation interrupts.

Test 10 Failover

This test verifies the failover feature and verifies that the board failover executes successfully. The test contains the following subtests:

Subtest	Subtest Name
a	Testing Channel 0
b	Testing Channel 1

This test performs the following operations for each channel:

- Displays the following message:

```
NOTE: Test 10 requires operator intervention.
      Repeat this test for all channels on the ESCON under test.
      Repeat this test for all ESCON Controllers to be tested.
```

- Issues a `Reset Channel` command to the board and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues.

- Displays the following message:

```
This subtest requires the external loopback connector
to be connected to channel 0 of secondary node and
disconnected from channel 0 of primary node.

Press RETURN when ready.
```

- Enables interrupts.
- Sends a `Diag` command to the board, one byte at a time, to start the External Loopback on-board diagnostic test. If an interrupt is generated, the test reports an error.
- Downloads the appropriate microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Enables channel interrupts.
- Sends an `Online Ring` command with Failover switch enabled and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then checks that all parameters of the response ring are correct.
- Enables channel interrupts.
- Sends a `Halt Ring` command and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then checks that all parameters of the response ring are correct.
- Enables interrupts.
- Sends a `Diag` command to the board, one byte at a time, to start the External Loopback on-board diagnostic test. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error occurs, the test reports the error.
- Enables channel interrupts.
- Sends a `Reset` command to both channels. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status.

Note – This test is bypassed unless you enabled External Loopback testing during program initialization.

Test 11 Download Control Unit (CU) & Channel (Chnl) Firmware

This test verifies the ability to download control unit and channel firmware. The test performs the following operations. Each operation is repeated for the channel.

- Issues a `Reset Channel` command to the control unit and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues.
- Downloads the control unit microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.

The test then disables all interrupts.

Note – This test is bypassed unless you enabled Channel/Control Unit testing during program initialization.

Test 12 Link Initialization

This test verifies the ability to download control unit and channel firmware and initializes and tests the ring structures. The test performs the following operations:

- Issues a `Reset Channel` command to the control unit and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues. This operation is repeated for the channel.
- Downloads the control unit microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Open Ring` command to the control unit and verifies that the Open operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the control unit and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the control unit and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the channel and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that the logical paths were established. If an error is detected, the test reports the error.

- Sends an `Offline Ring` command to the control unit and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends an `Offline Ring` command to the channel and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Disables all interrupts.

Note – This test is bypassed unless you enabled Channel/Control Unit testing during program initialization.

Test 13 Channel: Write - Control Unit: Read

This test verifies the ability write to the channel and read from the control unit. The test performs the following subtests:

Subtest	Subtest Name
a	Single Device
b	Multiple Devices
c	Multiple Logical Control Units

Each subtest performs the following operations:

- Issues a `Reset Channel` command to the control unit and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues. This operation is repeated for the channel.
- Downloads the control unit microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error. This operation is repeated for the channel.
- Sends an `Open Ring` command to the control unit and verifies that the Open operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the control unit and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the control unit and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the channel and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that the logical paths were established. If an error is detected, the test reports the error.
- Initializes the input buffer.
- Initializes the output buffer.
- Issues a Start I/O `Write` of 0x100 bytes to the channel and verifies that the operation was successful.
- Sends a `Read` to the control unit and verifies that the operation was successful.

- Waits for the `Read` command to complete. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Waits for the `Write` command to complete. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly.
- Verifies the Residual Count.
- Verifies the CCW ending address.
- Sends an `Offline Ring` command to the control unit and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends an `Offline Ring` command to the channel and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Disables all interrupts.

Note – This test is bypassed unless you enabled Channel/Control Unit testing during program initialization.

Test 14 Data Transfer (Chnl: Write - CU: Read)

This test verifies the ability to write to the channel and read from the control unit using different data patterns. The test performs the following subtests:

Subtest	Data Pattern
a	All Ones
b	All Zeros
c	A's & 5's (0xAA55AA55)
d	Address in Address

The test performs the following operations:

- Issues a `Reset Channel` command to the control unit and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues. This operation is repeated for the channel.
- Downloads the control unit microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error. This operation is repeated for the channel.
- Sends an `Open Ring` command to the control unit and verifies that the Open operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the control unit and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the control unit and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.

- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the channel and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that the logical paths were established. If an error is detected, the test reports the error.
- Initializes the input buffer.
- Initializes the output buffer.
- Issues a Start I/O `Write` of 0x100 bytes to the channel and verifies that the operation was successful.
- Sends a `Read` of 0x100 bytes to the control unit and verifies that the operation was successful.
- Waits for the `Read` command to complete. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Waits for the `Write` command to complete. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly.
- Verifies the Residual Count.
- Verifies the CCW ending address.
- Sends an `Offline Ring` command to the control unit and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.

- Sends an `Offline Ring` command to the channel and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Disables all interrupts.

The data transfer operations are repeated for each subtest.

Note – This test is bypassed unless you enabled Channel/Control Unit testing during program initialization.

Test 15 Variable Transfer Size (Chnl: Write - CU: Read)

This test verifies the ability to write to the channel and read from the control unit using different block sizes. The test performs the following subtests:

Subtest	Subtest Name
a	Small Block Size
b	Medium Block Size
c	Large Block Size
d	Even and Odd Block Size

The subtests use the following block sizes and increments during testing:

Subtest	Min Block Size	Max Block Size	Incremented by:
a	0x4	0x100	0x50

Subtest	Min Block Size	Max Block Size	Incremented by:
b	0x250	0x500	0x50
c	0x600	0xfffc	0x500
d	0x1	0xffff	0x503

The test performs the following operations:

- Issues a `Reset Channel` command to the control unit and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues. This operation is repeated for the channel.
- Downloads the control unit microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error. This operation is repeated for the channel.
- Sends an `Open Ring` command to the control unit and verifies that the Open operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the control unit and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the control unit and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the channel and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that the logical paths were established. If an error is detected, the test reports the error.
- Initializes the input buffer.

- Initializes the output buffer.
- Issues a Start I/O `Write` to the channel with a binary progression data pattern and verifies that the operation was successful.
- Sends a `Read` to the control unit and verifies that the operation was successful.
- Waits for the `Read` command to complete. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Waits for the `Write` command to complete. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly.
- Verifies the Residual Count.
- Verifies the CCW ending address.
- Sends an `Offline Ring` command to the control unit and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends an `Offline Ring` command to the channel and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Disables all interrupts.

Note – This test is bypassed unless you enabled Channel/Control Unit testing during program initialization.

Test 16 Channel: Read - Control Unit: Write

This test verifies the ability to write to the control unit and read from the channel. The test performs the following subtests:

Subtest	Subtest Name
a	Single Device
b	Multiple Devices

The test performs the following operations:

- Issues a `Reset Channel` command to the control unit and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues. This operation is repeated for the channel.
- Downloads the control unit microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error. This operation is repeated for the channel.
- Sends an `Open Ring` command to the control unit and verifies that the Open operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the control unit and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the control unit and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.

- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the channel and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that the logical paths were established. If an error is detected, the test reports the error.
- Initializes the input buffer.
- Initializes the output buffer.
- Issues a Start I/O `Read` of 0x100 bytes with 0x01020304 pattern to the channel and verifies that the operation was successful.
- Sends a `Write` to the control unit and verifies that the operation was successful.
- Waits for the `Read` command to complete. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Waits for the `Write` command to complete. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly.
- Verifies the Residual Count.
- Verifies the CCW ending address.
- Sends an `Offline Ring` command to the control unit and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends an `Offline Ring` command to the channel and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.

- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Disables all interrupts.

The data transfer operations are repeated for each subtest.

Note – This test is bypassed unless you enabled Channel/Control Unit testing during program initialization.

Test 17 Data Transfer (Chnl: Read - CU: Write)

This test verifies the ability to write to the control unit and read from the channel using different data patterns. The test performs the following subtests:

Subtest	Data Pattern
a	All Ones
b	All Zeros
c	A's & 5's (0xAA55AA55)
d	Address in Address

The test performs the following operations:

- Issues a `Reset Channel` command to control unit and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues. This operation is repeated for the channel.
- Downloads the control unit microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error. This operation is repeated for the channel.

- Sends an `Open Ring` command to the control unit and verifies that the Open operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the control unit and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the control unit and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the channel and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that the logical paths were established. If an error is detected, the test reports the error.
- Initializes the input buffer.
- Initializes the output buffer.
- Sends a `Write` of 0x100 bytes to the control unit and verifies that the operation was successful.
- Issues a Start I/O `Read` to the channel and verifies that the operation was successful.
- Waits for the `Write` command to complete. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.

- Waits for the `Read` command to complete. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly.
- Verifies the Residual Count.
- Verifies the CCW ending address.
- Sends an `Offline Ring` command to the control unit and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends an `Offline Ring` command to the channel and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Disables all interrupts.

The data transfer operations are repeated for each subtest.

Note – This test is bypassed unless you enabled Channel/Control Unit testing during program initialization.

Test 18 Variable Transfer Size (Chnl: Read - CU: Write)

This test verifies the ability to write to the control unit and read from the channel using different block sizes. The test performs the following subtests:

Subtest	Subtest Name
a	Small Block Size
b	Medium Block Size
c	Large Block Size
d	Even and Odd Block Size

The subtests use the following block sizes and increments during testing:

Subtest	Min Block Size	Max Block Size	Incremented by:
a	0x4	0x104	0x50
b	0x250	0x500	0x50
c	0x600	0xfffc	0x500
d	0x1	0xffff	0x503

The test performs the following operations:

- Issues a `Reset Channel` command to the control unit and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues. This operation is repeated for the channel.
- Downloads the control unit microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error. This operation is repeated for the channel.
- Sends an `Open Ring` command to the control unit and verifies that the Open operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.

- Sends a `Define Ring` command to the control unit and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the control unit and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the channel and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that the logical paths were established. If an error is detected, the test reports the error.
- Initializes the input buffer.
- Initializes the output buffer.
- Sends a `Write` to the control unit and verifies that the operation was successful.
- Issues a Start I/O `Read` to the channel and verifies that the operation was successful.
- Waits for the `Write` command to complete. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Waits for the `Read` command to complete. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly.
- Verifies the Residual Count.
- Verifies the CCW ending address.
- Sends an `Offline Ring` command to the control unit and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.

- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends an `Offline Ring` command to the channel and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Disables all interrupts.

The data transfer operations are repeated for the each subtest.

Note – This test is bypassed unless you enabled Channel/Control Unit testing during program initialization.

Test 19 Channel: Wr/Rd - Control Unit: Wr/Rd

This test verifies the ability to write and read to and from the channel and the control unit. The test performs the following subtests:

Subtest	Subtest Name
a	Single Device
b	Multiple Devices

The test performs the following operations:

- Issues a `Reset Channel` command to the control unit and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues. This operation is repeated for the channel.
- Downloads the control unit microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error. This operation is repeated for the channel.
- Sends an `Open Ring` command to the control unit and verifies that the Open operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the control unit and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt segregated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the control unit and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the channel and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that the logical paths were established. If an error is detected, the test reports the error.
- Initializes the input buffer.
- Initializes the output buffer.
- Issues a Start I/O `Write` of 0x100 bytes to the channel and verifies that the operation was successful.
- Sends a `Read` to the control unit and verifies that the operation was successful.

- Waits for the `Read` command to complete. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Waits for the `Write` command to complete. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly.
- Verifies the Residual Count.
- Verifies the CCW ending address.
- Initializes the input buffer.
- Initializes the output buffer.
- Issues a Start I/O `Read` to the channel and verifies that the operation was successful.
- Sends a `Write` of 0x100 bytes to the control unit and verifies that the operation was successful.
- Waits for the `Read` command to complete. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Waits for the `Write` command to complete. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly.
- Verifies the Residual Count.
- Verifies the CCW ending address.

The preceding operations are repeated for the each subtest.

- Sends an `Offline Ring` command to the control unit and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.

- Sends an `Offline Ring` command to the channel and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Disables all interrupts.

Note – This test is bypassed unless you enabled Channel/Control Unit testing during program initialization.

Test 20 Validate Initialization Command

This test verifies the ability to execute the `Initialization` command using incorrect parameter values and verifies that the error condition is detected correctly. The test performs the following subtests:

Subtest	Subtest Name
a	Non Sequential Logical Control Unit & Device Range
b	Invalid Device
c	Invalid & Valid Device
d	Invalid Link Address
e	Invalid Logical Control Unit
f	No Device Configured

The test issues a `Reset Channel` command to the control unit and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues. This operation is repeated for the channel.

Subtest a

This subtest performs the following operations:

- Initializes the control unit application parameters with nonsequential logical control unit and devices.
- Downloads the control unit microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error. This operation is repeated for the channel.
- Sends an `Open Ring` command to the control unit and verifies that the Open operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.

- Sends a `Define Ring` command to the control unit and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the control unit and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the channel and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that the logical paths were established. If an error is detected, the test reports the error.
- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends an `Offline Ring` command to the channel and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.

Subtest b

This subtest performs the following operations:

- Initializes the channel application parameters with an invalid device.
- Downloads the channel microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the channel and verifies that the Define operation was not successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that invalid device status is reported.
- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.

Subtest c

This subtest performs the following operations:

- Initializes the control unit and channel application parameters with invalid devices and valid devices.
- Downloads the control unit microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error. This operation is repeated for the channel.
- Sends an `Open Ring` command to the control unit and verifies that the Open operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the control unit and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.

- Sends an `Online Ring` command to the control unit and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the channel and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that the logical paths were established. If an error is detected, the test reports the error.
- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends an `Offline Ring` command to the channel and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.

Subtest d

This subtest performs the following operations:

- Initializes the channel application parameters with an invalid link address.
- Downloads the channel microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.

- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that invalid link address status is reported. If not, an error is reported.
- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.

Subtest e

This subtest performs the following operations:

- Initializes the channel application parameters with an invalid logical control unit.
- Downloads the channel microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that invalid logical control unit status is reported.
- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.

Subtest f

This subtest performs the following operations:

- Initializes the channel application parameters with no device specified.
- Downloads the channel microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.

- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that the status indicates no device configured. If not, an error is reported.
- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Disables all interrupts.

Note – This test is bypassed unless you enabled Channel/Control Unit testing during program initialization.

Test 21 Immediate Command

This test verifies the ability to correctly run the `Immediate` command. The test performs the following subtests:

Subtest	Subtest Name
a	Single Device
b	Multiple Devices

The test performs the following operations:

- Issues a `Reset Channel` command to the control unit and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues. This operation is repeated for the channel.
- Downloads the control unit microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error. This operation is repeated for the channel.

- Sends an `Open Ring` command to the control unit and verifies that the Open operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the control unit and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the control unit and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the channel and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that the logical paths were established. If an error is detected, the test reports the error.
- Initializes the input buffer.
- Initializes the output buffer with an incrementing data pattern.
- Sends an `Immediate Write` to the channel and verifies that the operation was successful.
- Sends an `Immediate Read` to the control unit and verifies that the operation was successful.
- Waits for the `Read` command to complete. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Waits for the `Write` command to complete. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly.
- Verifies the Residual Count.

- Verifies the CCW ending address.
- Sends an `Offline Ring` command to the control unit and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends an `Offline Ring` command to the channel and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Disables all interrupts.

The preceding operations are repeated for each subtest.

Note – This test is bypassed unless you enabled Channel/Control Unit testing during program initialization.

Test 22 Command Chaining

This test verifies the ability to execute a list of command chained instructions. The test performs the following subtests:

Subtest	Subtest Name
a	Rd and Wr List (Single Dev)
b	Rd and Wr List (Multi Dev)
c	Wr, Rd, and Nop List (Multi Dev)
d	Rd List and Wr List (Multi Dev)

The test performs the following operations:

- Issues a `Reset Channel` command to the control unit and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues. This operation is repeated for the channel.
- Downloads the control unit microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error. This operation is repeated for the channel.
- Sends an `Open Ring` command to the control unit and verifies that the Open operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the control unit and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the control unit and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.

- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the channel and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that logical paths were established. If an error is detected, the test reports the error.
- Initializes the input buffer.
- Initializes the output buffer with an incrementing data pattern.
- Sends a command chained list to the channel and verifies that the operation was successful.
- Sends the corresponding command to the control unit and verifies that the operation was successful.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Waits for the channel to complete the command chained list. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly.
- Verifies the residual byte count, and the last ccw address.
- Sends an `Offline Ring` command to the control unit and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends an `Offline Ring` command to the channel and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.

- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Disables all interrupts.

Note – This test is bypassed unless you enabled Channel/Control Unit testing during program initialization.

The following CCW lists are used during testing:

Subtests a and b

CCW1, Cmd: Write, Flag: Command Chain, Size= 0x100

CCW2, Cmd: Write, Flag: Command Chain, Size= 0x100

CCW3, Cmd: Read, Flag: Command Chain, Size= 0x100

CCW4, Cmd: Read, Flag: none, Size= 0x100

Subtest c

CCW1, Cmd: Write, Flag: Command Chain, Size= 0x100

CCW2, Cmd: Nop, Flag: Command Chain, Size= 0

CCW3, Cmd: Write, Flag: Command Chain, Size= 0x100

CCW4, Cmd: Nop, Flag: Command Chain, Size= 0

CCW5, Cmd: Read, Flag: Command Chain, Size= 0x100

CCW6, Cmd: Nop, Flag: Command Chain, Size= 0

CCW7, Cmd: Read, Flag: Command Chain, Size= 0x100

CCW8, Cmd: Nop, Flag: None, Size= 0

Subtest d

CCW1, Cmd: Write, Flag: Command Chain, Size= 0x100

CCW2, Cmd: Write, Flag: Command Chain, Size= 0x200

CCW3, Cmd: Write, Flag: Command Chain, Size= 0x300

.

.

.

CCW10, Cmd: Write, Flag: None ...

Test 23 Command Chaining with Variable Transfer Size

This test verifies the ability to execute a list of command chained instructions with variable transfer counts. The test performs the following subtests:

Subtest	Subtest Name
a	Rd and Wr List (Single Dev)
b	Rd and Wr List (Multi Dev)
c	Rd,Wr, and Nop List (Multi Dev)
d	Rd List and Wr List (Multi Dev)
e	Wr List and Rd List (Multi Dev)
f	Rd List and Wr List / Even & Odd TC (Multi Dev)
g	Wr List and Rd List / Even & Odd TC (Multi Dev)

The test performs the following operations:

- Issues a `Reset Channel` command to the control unit and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues. This operation is repeated for the channel.
- Downloads the control unit microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error. This operation is repeated for the channel.
- Sends an `Open Ring` command to the control unit and verifies that the Open operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.

- Sends a `Define Ring` command to the control unit and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the control unit and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Initializes the input buffer.
- Initializes the output buffer with a binary progression data pattern.
- Sends a command chained list to the channel and verifies that the operation was successful.
- Sends the corresponding command to the control unit and verifies that the operation was successful.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Waits for the channel to complete the command chained list. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly.
- Verifies the residual byte count, and the last ccw address.
- Sends an `Offline Ring` command to the control unit and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends an `Offline Ring` command to the channel and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.

- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Disables all interrupts.

Note – This test is bypassed unless you enabled Channel/Control Unit testing during program initialization.

The following lists of CCWs are used during testing:

Subtests a and b

CCW1, Cmd: Write, Flag: Command Chain, Size= 0x100

CCW2, Cmd: Write, Flag: Command Chain, Size= 0xfffc

CCW3, Cmd: Read, Flag: Command Chain, Size= 0x100

CCW4, Cmd: Read, Flag: None, Size= 0xfffc

Subtest c

CCW1, Cmd: Write, Flag: Command Chain, Size= 0x100

CCW2, Cmd: Nop, Flag: Command Chain, Size= 0

CCW3, Cmd: Write, Flag: Command Chain, Size= 0xfffc

CCW4, Cmd: Nop, Flag: Command Chain, Size= 0

CCW5, Cmd: Read, Flag: Command Chain, Size= 0x100

CCW6, Cmd: Nop, Flag: Command Chain, Size= 0

CCW7, Cmd: Read, Flag: Command Chain, Size= 0xfffc

CCW8, Cmd: Nop, Flag: None, Size= 0

Subtest d

CCW1, Cmd: Write, Flag: Command Chain, Size= 0x500

CCW2, Cmd: Write, Flag: Command Chain, Size= 0x1000

CCW3, Cmd: Write, Flag: Command Chain, Size= 0x1500

.

.

.

CCW10, Cmd: Write, Flag: None, Size= 0xff00

Subtest e

CCW1, Cmd: Read, Flag: Command Chain, Size= 0x500

CCW2, Cmd: Read, Flag: Command Chain, Size= 0x1000

CCW3, Cmd: Read, Flag: Command Chain, Size= 0x1500

.

.

.

CCW10, Cmd: Read, Flag: None, Size= 0xff00

Subtest f

CCW1, Cmd: Write, Flag: Command Chain, Size= 0x503

CCW2, Cmd: Write, Flag: Command Chain, Size= 0x1006

CCW3, Cmd: Write, Flag: Command Chain, Size= 0x1509

.

.

.

CCW10, Cmd: Write, Flag: None, Size= 0xffff

Subtest g

CCW1, Cmd: Read, Flag: Command Chain, Size= 0x503

CCW2, Cmd: Read, Flag: Command Chain, Size= 0x1006

CCW3, Cmd: Read, Flag: Command Chain, Size= 0x1509

.

.

.

CCW10, Cmd: Read, Flag: None, Size= 0xffff

Test 24 Command Chaining with Variable DMA Block Size

This test verifies the ability to execute a list of command chained instructions with variable DMA block sizes. The test performs the following subtests:

Subtest	Subtest Name
a	1K DMA Block Size
b	2K DMA Block Size
c	4K DMA Block Size
d	8K DMA Block Size
e	16K DMA Block Size
f	32K DMA Block Size

The test performs the following operations:

- Issues a `Reset Channel` command to the control unit and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues. This operation is repeated for the channel.
- Downloads the control unit microcode and sets up the ring structures with a different DMA block size. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error. This operation is repeated for the channel.
- Sends an `Open Ring` command to the control unit and verifies that the Open operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the control unit and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.

- Sends an `Online Ring` command to the control unit and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the channel and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that the logical paths were established. If an error is detected, the test reports the error.
- Initializes the input buffer.
- Initializes the output buffer with an incrementing data pattern.
- Sends each of the command chained lists to the channel and verifies that the operation was successful.
- Sends the corresponding command to the control unit and verifies that the operation was successful.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Waits for the channel to complete the command chaining. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly.
- Verifies the residual byte count, and the last ccw address.
- Sends an `Offline Ring` command to the control unit and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.

- Sends an `Offline Ring` command to the channel and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Disables all interrupts.

Note – This test is bypassed unless you enabled Channel/Control Unit testing during program initialization.

The following lists of CCWs are used during testing:

List 1

CCW1, Cmd: Write, Flag: Command Chain, Size= 0x100

CCW2, Cmd: Write, Flag: Command Chain, Size= 0x4000

CCW3, Cmd: Read, Flag: Command Chain, Size= 0x100

CCW4, Cmd: Read, Flag: None, Size= 0x4000

List 2

CCW1, Cmd: Write, Flag: Command Chain, Size= 0x100

CCW2, Cmd: Nop, Flag: Command Chain, Size= 0

CCW3, Cmd: Write, Flag: Command Chain, Size= 0x4000

CCW4, Cmd: Nop, Flag: Command Chain, Size= 0

CCW5, Cmd: Read, Flag: Command Chain, Size= 0x100

CCW6, Cmd: Nop, Flag: Command Chain, Size= 0

CCW7, Cmd: Read, Flag: Command Chain, Size= 0x4000

CCW8, Cmd: Nop, Flag: None, Size= 0

List 3

CCW1, Cmd: Write, Flag: Command Chain, Size= 0x500

CCW2, Cmd: Write, Flag: Command Chain, Size= 0x1000

CCW3, Cmd: Write, Flag: Command Chain, Size= 0x1500

.

.

.

CCW10, Cmd: Write, Flag: None ...

List 4

CCW1, Cmd: Read, Flag: Command Chain, Size= 0x500

CCW2, Cmd: Read, Flag: Command Chain, Size= 0x1000

CCW3, Cmd: Read, Flag: Command Chain, Size= 0x1500

.

.

.

CCW10, Cmd: Read, Flag: None ...

Test 25 Concurrent Command Chaining

This test verifies the ability to concurrently execute lists of command chained instructions from different devices. The test performs the following subtests:

Subtest	Subtest Name
a	Rd and Wr List (Two Dev) / Variable Size
b	Rd and Wr List (Multi Dev) / Variable Size
c	Rd,Wr, Nop List (Multi Dev) / Variable Size
d	Rd List and Wr List (Multi Dev) / Variable Size
e	Wr List and Rd List (Multi Dev) / Variable Size

The test performs the following operations:

- Issues a `Reset Channel` command to the control unit and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues. This operation is repeated for the channel.
- Downloads the control unit microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error. This operation is repeated for the channel.
- Sends an `Open Ring` command to the control unit and verifies that the Open operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the control unit and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the control unit and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the channel and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that the logical paths were established. If an error is detected, the test reports the error.
- Initializes the input buffer.
- Initializes the output buffer with an incrementing data pattern.
- Sends the command chained list to each and every device on the channel without waiting for the operation to complete.
- Sends the corresponding command to each device on the control unit and verifies that the operation was successful.

- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Waits for the channel to complete the command chaining. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly for each device.
- Verifies the residual byte count, and the last ccw address.
- Sends an `Offline Ring` command to the control unit and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends an `Offline Ring` command to the channel and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Disables all interrupts.

Note – This test is bypassed unless you enabled Channel/Control Unit testing during program initialization.

The following lists of CCWs are used during testing:

Subtests a and b

CCW1, Cmd: Write, Flag: Command Chain, Size= 0x100

CCW2, Cmd: Write, Flag: Command Chain, Size= 0x4000

CCW3, Cmd: Read, Flag: Command Chain, Size= 0x100

CCW4, Cmd: Read, Flag: None, Size= 0x4000

Subtest c

CCW1, Cmd: Write, Flag: Command Chain, Size= 0x100

CCW2, Cmd: Nop, Flag: Command Chain, Size= 0

CCW3, Cmd: Write, Flag: Command Chain, Size= 0x4000

CCW4, Cmd: Nop, Flag: Command Chain, Size= 0

CCW5, Cmd: Read, Flag: Command Chain, Size= 0x100

CCW6, Cmd: Nop, Flag: Command Chain, Size= 0

CCW7, Cmd: Read, Flag: Command Chain, Size= 0x4000

CCW8, Cmd: Nop, Flag: None, Size= 0

Subtest d

CCW1, Cmd: Write, Flag: Command Chain, Size= 0x500

CCW2, Cmd: Write, Flag: Command Chain, Size= 0x1000

CCW3, Cmd: Write, Flag: Command Chain, Size= 0x1500

.

.

.

CCW10, Cmd: Write, ...

Subtest e

CCW1, Cmd: Read, Flag: Command Chain, Size= 0x500

CCW2, Cmd: Read, Flag: Command Chain, Size= 0x1000

CCW3, Cmd: Read, Flag: Command Chain, Size= 0x1500

.

.

.

CCW10, Cmd: Read, ...

Test 26 Data Chaining

This test verifies the ability to execute data chained commands. The test performs the following subtests:

Subtest	Subtest Name
a	Short Wr/Rd List (Single Dev)
b	Short Wr/Rd List (Multi Dev)
c	Long Rd List / Small Variable TC (Multi Dev)
d	Long Wr List / Small Variable TC (Multi Dev)
e	Long Rd List / Large Variable TC (Multi Dev)
f	Long Wr List / Large Variable TC (Multi Dev)
g	Long Rd List / Even & Odd TC (Multi Dev)
h	Long Wr List / Even & Odd TC (Multi Dev)

The test performs the following operations:

- Issues a `Reset Channel` command to the control unit and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues. This operation is repeated for the channel.
- Downloads the control unit microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error. This operation is repeated for the channel.
- Sends an `Open Ring` command to the control unit and verifies that the Open operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.

- Sends a `Define Ring` command to the control unit and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the control unit and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the channel and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that the logical paths were established. If an error is detected, the test reports the error.
- Initializes the input buffer.
- Initializes the output buffer with a binary progression data pattern.
- Sends a data chained list to the channel.
- Sends the corresponding command to the control unit and verifies that the operation was successful.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Waits for the channel to complete the data chaining. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly.
- Verifies the residual byte count, and the last ccw address.
- Sends an `Offline Ring` command to the control unit and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.

- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends an `Offline Ring` command to the channel and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Disables all interrupts.

Note – This test is bypassed unless you enabled Channel/Control Unit testing during program initialization.

The following lists of CCWs are used during testing:

Subtest a

CCW1, Cmd: Write, Flag: Data Chain, Size= 0x100

CCW2, Cmd: Write, Flag: None, Size= 0x100

Subtest b

CCW1, Cmd: Read, Flag: Data Chain, Size= 0x100

CCW2, Cmd: Read, Flag: None, Size= 0x100

Subtest c

CCW1, Cmd: Write, Flag: Data Chain, Size= 0x100

CCW2, Cmd: Write, Flag: Data Chain, Size= 0x200

CCW3, Cmd: Write, Flag: Data Chain, Size= 0x300

.

.

.

CCW10, Cmd: Write, Flag: None ...

Subtest d

CCW1, Cmd: Read, Flag: Data Chain, Size= 0x100

CCW2, Cmd: Read, Flag: Data Chain, Size= 0x200

CCW3, Cmd: Read, Flag: Data Chain, Size= 0x300

.

.

.

CCW10, Cmd: Read, Flag: None ...

Subtest e

CCW1, Cmd: Write, Flag: Data Chain, Size= 0x1000

CCW2, Cmd: Write, Flag: Data Chain, Size= 0x2000

CCW3, Cmd: Write, Flag: Data Chain, Size= 0x3000

.

.

.

CCW10, Cmd: Write, Flag: None ...

Subtest f

CCW1, Cmd: Read, Flag: Data Chain, Size= 0x1000

CCW2, Cmd: Read, Flag: Data Chain, Size= 0x2000

CCW3, Cmd: Read, Flag: Data Chain, Size= 0x3000

.

.

.

CCW10, Cmd: Read, Flag: None ...

Subtest g

CCW1, Cmd: Write, Flag: Data Chain, Size= 0x1003

CCW2, Cmd: Write, Flag: Data Chain, Size= 0x2006

CCW3, Cmd: Write, Flag: Data Chain, Size= 0x3009

.

.

.

CCW10, Cmd: Write, Flag: None ...

Subtest h

CCW1, Cmd: Read, Flag: Data Chain, Size= 0x1003

CCW2, Cmd: Read, Flag: Data Chain, Size= 0x2006

CCW3, Cmd: Read, Flag: Data Chain, Size= 0x3009

.

.

.

CCW10, Cmd: Read, Flag: None ...

Test 27 Data Chain - Gather / Scatter

This test verifies the ability to execute data chained commands in different modes. The test performs the following subtests:

Subtest	Subtest Name
a	Define (Multi Dev)
b	Vector (Multi Dev)
c	Gather / Scatter (Multi Dev)

The test performs the following operations:

- Issues a `Reset Channel` command to the control unit and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues. This operation is repeated for the channel.

- Downloads the control unit microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error. This operation is repeated for the channel.
- Sends an `Open Ring` command to the control unit and verifies that the Open operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the control unit and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the control unit and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the channel and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that the logical paths were established. If an error is detected, the test reports the error.
- Initializes the input buffer.
- Initializes the output buffer with an incrementing data pattern.
- Sends the data chained list to the channel.

Subtest a

- Sends the corresponding command to the control unit using the transfer address specified in the `Define` command and verifies that the operation was successful.

Subtest b

- Sends the corresponding command to the control unit using the transfer address specified in the vector list of one entry and verifies that the operation was successful.

Subtest c

- Sends the corresponding command to the control unit using the transfer addresses specified in the vector list of ten entries and verifies that the operation was successful.

The test then performs the following operations:

- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Waits for the channel to complete the data chaining. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly.
- Verifies the residual byte count, and the last ccw address.
- Sends an `Offline Ring` command to the control unit and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.

- Sends an `Offline Ring` command to the channel and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Disables all interrupts.

Note – This test is bypassed unless you enabled Channel/Control Unit testing during program initialization.

The following lists of CCWs are used during testing:

Subtest a

CCW1, Cmd: Write, Flag: Data Chain, Size= 0x100

CCW2, Cmd: Write, Flag: None, Size= 0x100

Subtest b

CCW1, Cmd: Write, Flag: Data Chain, Size= 0x100

CCW2, Cmd: Write, Flag: Data Chain, Size= 0x200

CCW3, Cmd: Write, Flag: Data Chain, Size= 0x300

.

.

.

CCW10, Cmd: Write, Flag: None ...

Subtest c

CCW1, Cmd: Read, Flag: Data Chain, Size= 0x100

CCW2, Cmd: Read, Flag: Data Chain, Size= 0x200

CCW3, Cmd: Read, Flag: Data Chain, Size= 0x300

.

.

.

CCW10, Cmd: Read, Flag: None ...

Test 28 Non-Contiguous Data Chaining

This test verifies the ability to execute data chained instructions in noncontiguous memory. The test performs the following subtests:

Subtest	Subtest Name
a	Short Wr/Rd List (Single Dev)
b	Short Wr/Rd List (Multi Dev)
c	Long Rd List / Variable Small TC (Multi Dev)
d	Long Wr List / Variable Small TC (Multi Dev)
e	Long Rd List / Variable Large TC (Multi Dev)
f	Long Wr List / Variable Large TC (Multi Dev)

The test performs the following operations:

- Issues a `Reset Channel` command to the control unit and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues. This operation is repeated for the channel.
- Downloads the control unit microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error. This operation is repeated for the channel.
- Sends an `Open Ring` command to the control unit and verifies that the Open operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the control unit and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.

- Sends an `Online Ring` command to the control unit and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the channel and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that the logical paths were established. If an error is detected, the test reports the error.
- Initializes the input buffer.
- Initializes the output buffer with a binary progression data pattern.
- Sends a data chained list with a noncontiguous transfer address space to the channel.
- Sends the corresponding command to the control unit and verifies that the operation was successful.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Waits for the channel to complete the data chained list. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly.
- Verifies the residual byte count, and the last ccw address.
- Sends an `Offline Ring` command to the control unit and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.

- Sends an `Offline Ring` command to the channel and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Disables all interrupts.

Note – This test is bypassed unless you enabled Channel/Control Unit testing during program initialization.

The following lists of CCWs are used during testing:

Subtest a

CCW1, Cmd: Write, Flag: Data Chain, Size= 0x100

CCW2, Cmd: Write, Flag: None, Size= 0x100

Subtest b

CCW1, Cmd: Read, Flag: Data Chain, Size= 0x100

CCW2, Cmd: Read, Flag: None, Size= 0x100

Subtest c

CCW1, Cmd: Write, Flag: Data Chain, Size= 0x100

CCW2, Cmd: Write, Flag: Data Chain, Size= 0x200

CCW3, Cmd: Write, Flag: Data Chain, Size= 0x300

.

.

.

CCW10, Cmd: Write , Flag: None...

Subtest d

CCW1, Cmd: Read, Flag: Data Chain, Size= 0x100

CCW2, Cmd: Read, Flag: Data Chain, Size= 0x200

CCW3, Cmd: Read, Flag: Data Chain, Size= 0x300

.

.

.

CCW10, Cmd: Read, Flag: None ...

Subtest e

CCW1, Cmd: Write, Flag: Data Chain, Size= 0x1000

CCW2, Cmd: Write, Flag: Data Chain, Size= 0x2000

CCW3, Cmd: Write, Flag: Data Chain, Size= 0x3000

.

.

.

CCW10, Cmd: Write, Flag: None ...

Subtest f

CCW1, Cmd: Read, Flag: Data Chain, Size= 0x1000

CCW2, Cmd: Read, Flag: Data Chain, Size= 0x2000

CCW3, Cmd: Read, Flag: Data Chain, Size= 0x3000

.

.

.

CCW10, Cmd: Read, Flag: None ...

Test 29 Disconnect / Reconnect

This test verifies the ability to disconnect and reconnect. The test performs the following subtests:

Subtest	Subtest Name
a	Disconnect at End of List (One Dev)
b	Disconnect at End of List (Two Dev)
c	Disconnect at End of List (Multi Dev)
d	Disconnect at Middle of Wr List (Multi Dev)
e	Disconnect at Middle of Rd List (Multi Dev)

The test performs the following operations:

- Issues a `Reset Channel` command to the control unit and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues. This operation is repeated for the channel.
- Downloads the control unit microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error. This operation is repeated for the channel.
- Sends an `Open Ring` command to the control unit and verifies that the Open operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the control unit and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the control unit and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.

- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the channel and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that the logical paths were established. If an error is detected, the test reports the error.
- Initializes the input buffer.
- Initializes the output buffer with a binary progression data pattern.

Subtests a, b and c

- Issues a Start I/O command chained list to every device on the channel without waiting for the operation to complete.
- Sends the corresponding command to the control unit and verifies that the operation was successful.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Sends a `Disconnect Ring` command at the end of list.
- Sends a `Reconnect Ring` command.

Subtests d and e

- Issues a Start I/O command chained list to every device on the channel without waiting for the operation to complete.
- Sends half of the commands to the control unit and verifies that the operation was successful.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Sends a `Disconnect Ring` command.
- Sends a `Reconnect Ring` command.
- Continues to send the corresponding commands to the control unit and verifies that the operation was successful.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.

- Waits for the channel to complete the command chained list. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly.
- Verifies the residual byte count, and the last ccw address.
- Sends an `Offline Ring` command to the control unit and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends an `Offline Ring` command to the channel and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Disables all interrupts.

Note – This test is bypassed unless you enabled Channel/Control Unit testing during program initialization.

The following lists of CCWs are used during testing:

Subtests a, b, and c

CCW1, Cmd: Write, Flag: Command Chain, Size= 0x100

CCW2, Cmd: Write, Flag: Command Chain, Size= 0x4000

CCW3, Cmd: Read, Flag: Command Chain, Size= 0x100

CCW4, Cmd: Read, Flag: None, Size= 0x4000

Subtest d

CCW1, Cmd: Write, Flag: Command Chain, Size= 0x100

CCW2, Cmd: Write, Flag: Command Chain, Size= 0x4000

CCW3, Cmd: Write, Flag: Command Chain, Size= 0x100

CCW4, Cmd: Write, Flag: None, Size= 0x4000

Subtest e

CCW1, Cmd: Read, Flag: Command Chain, Size= 0x100

CCW2, Cmd: Read, Flag: Command Chain, Size= 0x4000

CCW3, Cmd: Read, Flag: Command Chain, Size= 0x100

CCW4, Cmd: Read, Flag: None, Size= 0x4000

Test 30 Split Transfer

This test verifies the ability to perform a split transfer. The test performs the following subtests:

Subtest	Subtest Name
a	Ending Status without Data Transfer (Single Dev)
b	Ending Status without Data Transfer (Multi Dev)
c	Ending Status with Data Transfer (Multi Dev)

The test performs the following operations:

- Issues a `Reset Channel` command to the control unit and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues. This operation is repeated for the channel.
- Downloads the control unit microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error. This operation is repeated for the channel.

- Sends an `Open Ring` command to the control unit and verifies that the Open operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the control unit and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the control unit and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the channel and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that the logical paths were established. If an error is detected, the test reports the error.
- Initializes the input buffer.
- Initializes the output buffer with a binary progression data pattern.

Subtest a

- Issues a Start I/O `Write` to the channel.
- Sends the corresponding command to the control unit and verifies that the operation was successful in the following order:
  - Cmd: `Read`, Flag: initial status & vector list, Size= 0x20
  - Cmd: `Read`, Flag: vector list, Size= 0x180
  - Cmd: `Read`, Flag: end status, channel end and device end, Size=0

Subtest b

- Issues a Start I/O command chained list to the channel.
- Sends the corresponding command to the control unit and verifies that the operation was successful in the following order:
  - Cmd: `Read`, Flag: initial status & vector list, Size= 0x20
  - Cmd: `Read`, Flag: vector list, Size= 0xf000
  - Cmd: `Read`, Flag: vector list, Size= 0x100
  - Cmd: `Read`, Flag: end status, channel end and device end, Size=0
  - Cmd: `Read`, Flag: initial status & vector list, Size= 0x20
  - Cmd: `Read`, Flag: vector list, Size= 0xf000
  - Cmd: `Read`, Flag: vector list, Size= 0x100
  - Cmd: `Read`, Flag: end status, channel end and device end, Size=0
  - Cmd: `Write`, Flag: initial status, vector list, Size= 0x320
  - Cmd: `Write`, Flag: initial status, vector list & end status, channel end and device end, Size= 0x320

Subtest c

- Issues a Start I/O command chained list to the channel.
- Sends the corresponding command to the control unit and verifies that the operation was successful in the following order:
  - Cmd: `Read`, Flag: initial status & vector list, Size= 0x20
  - Cmd: `Read`, Flag: vector list, Size= 0xf000
  - Cmd: `Read`, Flag: end status & vector list, channel end and device end, Size= 0x100
  - Cmd: `Read`, Flag: initial status & vector list, Size= 0x20
  - Cmd: `Read`, Flag: vector list, Size= 0xf000
  - Cmd: `Read`, Flag: end status & vector list, channel end and device end, Size= 0x100
  - Cmd: `Write`, Flag: initial status, end status & vector list, channel end and device end, Size= 0x320
  - Cmd: `Write`, Flag: initial status, vector list & end status, channel end and device end, Size= 0x320
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.

- Waits for the channel to complete the command chained list. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly.
- Verifies the residual byte count, and the last ccw address.
- Sends an `Offline Ring` command to the control unit and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends an `Offline Ring` command to the channel and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Disables all interrupts.

Note – This test is bypassed unless you enabled Channel/Control Unit testing during program initialization.

The following lists of CCWs are used during testing:

Subtest a

CCW1, Cmd: Write, Flag: None, Size= 0x1a0

Subtests b and c

CCW1, Cmd: Write, Flag: Command Chain, Size= 0xf120

CCW2, Cmd: Write, Flag: Command Chain, Size= 0xf120

CCW3, Cmd: Read, Flag: Command Chain, Size= 0x320

CCW3, Cmd: Read, Flag: None, Size= 0x320

Test 31 Even and Odd Boundaries (Chnl: Write - CU: Read)

This test verifies the ability to run `Write` and `Read Ring` commands starting at different boundaries. The test contains the following subtests:

Subtest	Subtest Name
a	Single Block Transfer (Boundary tx:wd0 rx:byte)
b	Single Block Transfer (Boundary tx:wd4 rx:byte)
c	Single Block Transfer (Boundary tx:wd8 rx:byte)
d	Multiple Block Transfer (Boundary tx:wd0 rx:byte)
e	Multiple Block Transfer (Boundary tx:wd4 rx:byte)
f	Multiple Block Transfer (Boundary tx:wd8 rx:byte)
g	Single Block Transfer (Boundary tx:byte rx:wd0)
h	Single Block Transfer (Boundary tx:byte rx:wd4)
i	Single Block Transfer (Boundary tx:byte rx:wd8)
j	Multiple Block Transfer (Boundary tx:byte rx:wd0)
k	Multiple Block Transfer (Boundary tx:byte rx:wd4)
l	Multiple Block Transfer (Boundary tx:byte rx:wd8)

The test performs the following operations:

- Issues a `Reset Channel` command to the control unit and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues. This operation is repeated for the channel.
- Downloads the control unit microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error. This operation is repeated for the channel.

- Sends an `Open Ring` command to the control unit and verifies that the Open operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the control unit and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the control unit and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the channel and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that the logical paths were established. If an error is detected, the test reports the error.
- Initializes the input buffer.
- Initializes the output buffer with a binary progression data pattern.
- Issues a Start I/O `Write` command starting at different boundaries to the channel.
- Sends a `Read` command starting at different boundaries to the control unit and verifies that the operation was successful.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Waits for the channel to complete. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly.
- Verifies the residual byte count, and the last ccw address.

- Sends an `Offline Ring` command to the control unit and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends an `Offline Ring` command to the channel and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Disables all interrupts.

Note – This test is bypassed unless you enabled Channel/Control Unit testing during program initialization.

Test 32 Even and Odd Boundaries (Chnl: Read - CU: Write)

This test verifies the ability to run `Write` and `Read Ring` commands starting at different boundaries. The test contains the following subtests:

Subtest	Subtest Name
a	Single Block Transfer (Boundary tx:wd0 rx:byte)
b	Single Block Transfer (Boundary tx:wd4 rx:byte)
c	Single Block Transfer (Boundary tx:wd8 rx:byte)
d	Multiple Block Transfer (Boundary tx:wd0 rx:byte)
e	Multiple Block Transfer (Boundary tx:wd4 rx:byte)
f	Multiple Block Transfer (Boundary tx:wd8 rx:byte)
g	Single Block Transfer (Boundary tx:byte rx:wd0)
h	Single Block Transfer (Boundary tx:byte rx:wd4)
i	Single Block Transfer (Boundary tx:byte rx:wd8)
j	Multiple Block Transfer (Boundary tx:byte rx:wd0)
k	Multiple Block Transfer (Boundary tx:byte rx:wd4)
l	Multiple Block Transfer (Boundary tx:byte rx:wd8)

The test performs the following operations:

- Issues a `Reset Channel` command to the control unit and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues. This operation is repeated for the channel.
- Downloads the control unit microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error. This operation is repeated for the channel.

- Sends an `Open Ring` command to the control unit and verifies that the Open operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the control unit and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the control unit and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the channel and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that the logical paths were established. If an error is detected, the test reports the error.
- Initializes the input buffer.
- Initializes the output buffer with a binary progression data pattern.
- Issues a Start I/O `Read` command starting at different boundaries to the channel.
- Sends a `Write` command starting at different boundaries to the control unit and verifies that the operation was successful.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Waits for channel to complete. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly.
- Verifies the residual byte count, and the last ccw address.

- Sends an `Offline Ring` command to the control unit and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends an `Offline Ring` command to the channel and verifies that the Offline operation was successful. If no interrupt is generated, timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Disables all interrupts.

Note – This test is bypassed unless you enabled Channel/Control Unit testing during program initialization.

Test 33 Sense Command

This test verifies the ability to execute the `Sense` command successfully. The test performs the following operations:

- Issues a `Reset Channel` command to the control unit and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues. This operation is repeated for the channel.
- Downloads the control unit microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error. This operation is repeated for the channel.

- Sends an `Open Ring` command to the control unit and verifies that the Open operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the control unit and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the control unit and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the channel and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that the logical paths were established. If an error is detected, the test reports the error.
- Initializes the input buffer.
- Initializes the output buffer with a binary progression data pattern.
- Issues a Start I/O `Sense` command to the channel.
- Sends a `Write` command to the control unit and verifies that the operation was successful.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Waits for the channel to complete. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the sense data transferred correctly.
- Verifies the residual byte count, and the last ccw address.

- Sends an `Offline Ring` command to the control unit and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends an `Offline Ring` command to the channel and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Disables all interrupts.

Note – This test is bypassed unless you enabled Channel/Control Unit testing during program initialization.

Test 34 Command and Data Chaining

This test verifies the ability to execute command chained and data chained instructions. The test performs the following subtests:

Subtest	Subtest Name
a	Rd and Wr List (Single Dev)
b	Rd and Wr List (Multi Dev)
c	Rd, Wr, Nop List (Multi Dev)
d	Rd List and Wr List (Multi Dev)
e	Wr List and Rd List (Multi Dev)

The test performs the following operations:

- Issues a `Reset Channel` command to the control unit and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues. This operation is repeated for the channel.
- Downloads the control unit microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error. This operation is repeated for the channel.
- Sends an `Open Ring` command to the control unit and verifies that the Open operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the control unit and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the control unit and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the channel and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that the logical paths were established. If an error is detected, the test reports the error.
- Initializes the input buffer.
- Initializes the output buffer with a binary progression data pattern.
- Issues a Start I/O command and data chained list to the channel.
- Sends the corresponding command to the control unit and verifies that the operation was successful.

- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Waits for the channel to complete the list. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly.
- Verifies the residual byte count, and the last ccw address.
- Sends an `Offline Ring` command to the control unit and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends an `Offline Ring` command to the channel and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Disables all interrupts.

Note – This test is bypassed unless you enabled Channel/Control Unit testing during program initialization.

The following lists of CCWs are used during testing:

Subtests a and b

CCW1, Cmd: Write, Flag: Command Chain, Size= 0x100

CCW2, Cmd: Write, Flag: Command Chain, Size= 0x100

CCW3, Cmd: Write, Flag: Data Chain, Size= 0x100

CCW4, Cmd: Write, Flag: Command Chain, Size= 0x100

CCW5, Cmd: Read, Flag: Command Chain, Size= 0x100

CCW6, Cmd: Read, Flag: Command Chain, Size= 0x100

CCW7, Cmd: Read, Flag: Data Chain, Size= 0x100

CCW8, Cmd: Read, Flag: None, Size= 0x100

Subtest c

CCW1, Cmd: Write, Flag: Command Chain, Size= 0x100

CCW2, Cmd: Nop, Flag: Command Chain, Size= 0

CCW3, Cmd: Write, Flag: Command Chain, Size= 0x100

CCW4, Cmd: Nop, Flag: Command Chain, Size= 0

CCW5, Cmd: Write, Flag: Data Chain, Size= 0x100

CCW6, Cmd: Write, Flag: Data Chain, Size= 0x100

CCW7, Cmd: Read, Flag: Command Chain, Size= 0x100

CCW8, Cmd: Nop, Flag: Command Chain, Size= 0

CCW9, Cmd: Read, Flag: Command Chain, Size= 0x100

CCW10, Cmd: Nop, Flag: Command Chain, Size= 0

CCW11, Cmd: Read, Flag: Data Chain, Size= 0x100

CCW12, Cmd: Read, Flag: None, Size= 0x100

Subtest d

CCW1, Cmd: Write, Flag: Command Chain, Size= 0x100

CCW2, Cmd: Write, Flag: Data Chain, Size= 0x200

CCW3, Cmd: Write, Flag: Command Chain, Size= 0x300

CCW4, Cmd: Write, Flag: Data Chain, Size= 0x400

.

.

.

CCW10, Cmd: Write, Flag: None ...

Subtest e

CCW1, Cmd: Read, Flag: Command Chain, Size= 0x100

CCW2, Cmd: Read, Flag: Data Chain, Size= 0x200

CCW3, Cmd: Read, Flag: Command Chain, Size= 0x300

CCW4, Cmd: Read, Flag: Data Chain, Size= 0x400

.

.

.

CCW10, Cmd: Read, Flag: None ...

Test 35 Command and Data Chaining with Variable Size

This test verifies the ability to execute command chained and data chained instructions with variable transfer sizes. The test performs the following subtests:

Subtest	Subtest Name
a	Rd and Wr List (Single Dev)
b	Rd and Wr List (Multi Dev)

The test performs the following operations:

- Issues a `Reset Channel` command to the control unit and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues. This operation is repeated for the channel.
- Downloads the control unit microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error. This operation is repeated for the channel.

- Sends an `Open Ring` command to the control unit and verifies that the Open operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the control unit and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the control unit and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the channel and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that the logical paths were established. If an error is detected, the test reports the error.
- Initializes the input buffer.
- Initializes the output buffer with a binary progression data pattern.
- Issues a Start I/O command and data chained list with variable transfer counts to the channel.
- Sends the corresponding command to the control unit and verifies that the operation was successful.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Waits for the channel to complete the list. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly.
- Verifies the residual byte count, and the last ccw address.

- Sends an `Offline Ring` command to the control unit and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends an `Offline Ring` command to the channel and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt` Ring command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Disables all interrupts.

Note – This test is bypassed unless you enabled Channel/Control Unit testing during program initialization.

The following list of CCWs is used during testing:

Subtests a and b

CCW1, Cmd: Write, Flag: Command Chain, Size= 0x100

CCW2, Cmd: Write, Flag: Command Chain, Size= 0xfffc

CCW3, Cmd: Write, Flag: Data Chain, Size= 0x100

CCW4, Cmd: Write, Flag: Command Chain, Size= 0xfffc

CCW5, Cmd: Read, Flag: Command Chain, Size= 0x100

CCW6, Cmd: Read, Flag: Command Chain, Size= 0xfffc

CCW7, Cmd: Read, Flag: Data Chain, Size= 0x100

CCW8, Cmd: Read, Flag: None, Size= 0xfffc

Test 36 Device Status

This test verifies the ability to generate the correct device status. The test performs the following subtests:

Subtest	Subtest Name
a	Control Unit Busy Status (Mult Dev)
b	Unit Check Status (Mult Dev)
c	Retry Command - Immediate (Mult Dev)
d	Retry Command - Deferred (Mult Dev)
e	Device Busy Condition

The test performs the following operations:

- Issues a `Reset Channel` command to the control unit and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues. This operation is repeated for the channel.
- Downloads the control unit microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error. This operation is repeated for the channel.
- Sends an `Open Ring` command to the control unit and verifies that the Open operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the control unit and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the control unit and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.

- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the channel and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that the logical paths were established. If an error is detected, the test reports the error.
- Initializes the input buffer.
- Initializes the output buffer with a binary progression data pattern.

Subtest a

- Issues a Start I/O `Read` command to the channel and continues without waiting for the command to complete.
- Issues a second Start I/O `Read` command to the channel.
- Sends a `Write` command to the control unit with End status in the flag field, and Status Modifier and Channel Busy in the status field.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Sends a `Reconnect Ring` command with Control Unit End status.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Sends a `Write` command to the control unit.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Sends a second `Write` command to the control unit.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Waits for the channel to complete both `Read` commands. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly.
- Verifies the residual byte count, and the last ccw address.

These operations are repeated for four devices.

Subtest b

- Initializes the input buffer.
- Initializes the output buffer with a binary progression data pattern.
- Issues a Start I/O `Read` command to the channel.
- Sends a `Write` command to the control unit with End status in the flag field, and with Status Modifier and Channel Busy in the status field.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Sends a `Reconnect Ring` command with Control Unit End status.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Sends a `Write` command to the control unit with End status in the flag field and Unit Check status.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Waits for the channel to complete. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Issues a Start I/O `Read` command to the channel.
- Sends a `Write` command to the control unit.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Waits for the channel to complete. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly.
- Verifies the residual byte count, and the last ccw address.

These operations are repeated for four devices.

Subtest c

- Initializes the input buffer.
- Initializes the output buffer with a binary progression data pattern.
- Issues a Start I/O `Read` command to the channel.
- Sends a `Write` command to the control unit with End status in the flag field, and Status Modifier, Channel End, Device End, and Unit Check in status field.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Sends a `Write` command to the control unit with Initial status, Vector List and End status in the flag field and Channel End and Device End in the status field.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Waits for the channel to complete. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly.
- Verifies the residual byte count, and the last ccw address.

Subtest d

- Initializes the input buffer.
- Initializes the output buffer with a binary progression data pattern.
- Issues a Start I/O `Read` command to the channel.
- Sends a `Write` command to the control unit with End status in the flag field, and with Status Modifier, Device End and Unit Check in status field.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Sends a `Reconnect Ring` command to the control unit with Channel Exception in the flag field and Device End in the status field.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Sends a `Write` command to the control unit with Initial status, Vector List and End status in the flag field and Channel End and Device End in the status field.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.

- Waits for the channel to complete. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly.
- Verifies the residual byte count, and the last ccw address.

These operations are repeated for four devices.

Subtest e

- Initializes the input buffer.
- Initializes the output buffer with a binary progression data pattern.
- Issues a Start I/O `Read` command to the channel.
- Sends a `Write` command to the control unit with End status in the flag field, and Channel Busy in the status field.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Sends a `Reconnect Ring` command to the control unit with Device End in the status field.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Sends a `Write` command to the control unit with Initial status, Vector List and End status in the flag field and Channel End and Device End in the status field.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.
- Waits for the channel to complete. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly.
- Verifies the residual byte count, and the last ccw address.

These operations are repeated for four devices.

The test then performs the following operations:

- Sends an `Offline Ring` command to the control unit and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends an `Offline Ring` command to the channel and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Disables all interrupts.

Note – This test is bypassed unless you enabled Channel/Control Unit testing during program initialization.

Test 37 Incorrect Length

This test verifies the ability to successfully write and read with incorrect transfer lengths. The test performs the following subtests:

Subtest	Subtest Name
a	Short Block on Channel Input (Mult Dev)
b	Long Block on Channel Input (Mult Dev)
c	Short Block on Channel Output (Mult Dev)
d	Short Block on Channel Input with SLI (Mult Dev)
e	Long Block on Channel Input with SLI (Mult Dev)
f	Short Block on Channel Output with SLI (Mult Dev)

The test performs the following operations:

- Issues a `Reset Channel` command to the control unit and verifies that the board resets successfully. The test first verifies that the board is ready to receive commands. If not, resynchronization is attempted. If resynchronization fails, testing is terminated. If resynchronization is successful, testing continues. This operation is repeated for the channel.
- Downloads the control unit microcode and sets up the ring structures. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error. This operation is repeated for the channel.
- Sends an `Open Ring` command to the control unit and verifies that the Open operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the control unit and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.

- Sends an `Online` Ring command to the control unit and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends a `Define Ring` command to the channel and verifies that the Define operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Sends an `Online Ring` command to the channel and verifies that the Online operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that the logical paths were established. If an error is detected, the test reports the error.
- Initializes the input buffer.
- Initializes the output buffer with a binary progression data pattern.

Subtest a

- Issues a Start I/O `Read` command to the channel with a transfer size less than the output.
- Sends a Write command to the control unit with a transfer size greater than the input and verifies that the operation was successful.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.

Subtest b

- Issues a Start I/O `Read` command to the channel with a transfer size greater than the output.
- Sends a `Write` command to the control unit with a transfer size less than the input and verifies that the operation was successful.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.

Subtest c

- Issues a Start I/O `Write` command to the channel with a transfer size less than the output.
- Sends a `Read` command to the control unit with a transfer size greater than the input and verifies that the operation was successful.
- Waits for the control unit to respond. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error.

The operations in Subtests a, b, and c are repeated for Subtests d, e, and f, respectively, with the Suppress Length Indication flag set.

The test then performs the following operations:

- Waits for the channel to complete. The test then verifies that all parameters of the response ring are correct. If an error is detected, the test reports the error. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies the status. If an error is detected, the test reports the error.
- Verifies that the data transferred correctly.
- Verifies the residual byte count, and the last ccw address.
- Sends an `Offline Ring` command to the control unit and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the control unit and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends an `Offline Ring` command to the channel and verifies that the Offline operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Sends a `Halt Ring` command to the channel and verifies that the Halt operation was successful. If no interrupt is generated, a timeout occurs and an error is reported. If an interrupt is generated, the test checks the interrupt status vector. If the vector is correct, the test then verifies that all parameters of the response ring are correct.
- Disables all interrupts.

Note – This test is bypassed unless you enabled Channel/Control Unit testing during program initialization.

CHAPTER 4

Program Messages

The following types of messages are associated with the VME Dual Channel ESCON Diagnostic:

- Monitor
- Start
- Cleanup
- Milestone
- Pass and Error Count
- Error

Monitor Messages

If this diagnostic is configured during automatic testing, the ROM Monitor displays the following program information on the operator's console before calling the diagnostic:

```
Slot n: Loading extended image escondiag
```

Variable	Description
n	Specifies the slot number of the VME Dual Channel ESCON board.

Note – The ROM Monitor cannot identify the slot associated with an Extended Diagnostic running in interactive mode.

Start Messages

Once called by the monitor or loaded interactively, the VME Dual Channel ESCON Diagnostic displays its start message on the operator's console in the following format:

```
Extended ESCON Diagnostic
Diagnostic Revision: y.y
[Slot: s,] Board CSR Addr: 0xaaaaaaaa - 0xbbbbbbbb
[Copyright year Sun Microsystems, Inc.]
```

Variable	Description
y.y	Specifies the current revision of the Extended Diagnostic.
s	Specifies the slot number of the board under test. The slot number is displayed only when the diagnostic is running in automatic mode.
aaaaaaaa	Specifies the default starting address of the memory address range assigned to this board.
bbbbbbbb	Specifies the default ending address of the memory address range assigned to this board.
year	Specifies the year the program was copyrighted. The copyright message is displayed only when the diagnostic is running in interactive mode.

If the diagnostic is running in interactive mode, it then displays the Interactive Mode Options Menu.

If the board CSR address was changed during program initialization, the program then displays the selected value in the following format:

```
Board CSR Addr: 0xaaaaaaaa - 0xbbbbbbbb
```

Variable	Description
aaaaaaaa	Specifies the selected starting address of the memory address range assigned to this board.
bbbbbbbb	Specifies the selected ending address of the memory address range assigned to this board.

Before testing starts, the program displays the board information with the following message:

```
Board Id: id, Rev: rev, Date: date, Serial No: number
```

Variable	Description
id	Specifies the board identification mnemonic (ESCON).
rev	Specifies the board revision letter.
date	Specifies the board revision date.
number	Specifies the board serial number.

Note – The Online switches on the distribution panel must be in the ON position.

Cleanup Messages

Cleanup messages are displayed if you entered q or Control-C to exit the diagnostic in interactive mode and you enabled milestone messages during program initialization. The program first displays:

```
cleaning up -
```

If cleanup is required, the program appends one or more periods (.) to the message during the cleanup sequence. When the cleanup sequence is completed, the following is appended to the message:

```
- exiting
```

The program then displays:

```
Channel 0 status = 0xaa, Channel 1 status = 0xbb
```

Variable	Description
aa, *bb*	Specify the channel status from the board.

Milestone Messages

Milestone messages are displayed only when the diagnostic is running in interactive mode and you enabled milestone messages during program initialization. The milestone messages for this diagnostic are grouped into the following categories:

- Initialization
- Test Startup
- Resynchronization
- Test Specific

Initialization Milestone Messages

During its initialization sequence, the diagnostic obtains the Local Bus and VMEbus base addresses and displays them with the following milestone messages:

```
Local base address: 0xlbus
VME base address: 0xvbus
```

Variable	Description
lbus	Specifies the Local Bus base address.
vbus	Specifies the VMEbus base address.

Test Startup Milestone Messages

As each test starts execution, the diagnostic displays a test milestone message:

```
Running Test tt: testname
```

Variable	Description
tt	Specifies the test number.
testname	Specifies the name of the test.

As each subtest starts, the diagnostic displays a subtest milestone message:

```
Subtest s: subname
```

Variable	Description
s	Specifies the subtest letter.
subname	Specifies the name of the subtest.

Each test first verifies the board is ready to receive commands. If resynchronization is required, the diagnostic displays a series of Resynchronization milestone messages. After the board is ready to receive commands, the diagnostic displays:

```
ESCON Channel x
Status I.D. = xx
```

Variable	Description
x	Specifies the channel under test.
xx	Specifies the Status ID value configured on the board.

Resynchronization Milestone Messages

If the board under test is not ready to receive commands, resynchronization is attempted. During resynchronization, the test displays:

```
Attempting resynchronization -
```

The test appends one or more periods (.) to the message while waiting for resynchronization to complete. If resynchronization is unsuccessful, the following is appended to the message and testing is terminated:

```
- Resynchronization Failed!
```

If resynchronization is successful, the following is appended to the message and testing continues:

```
- Resynchronization Successful
```

Test Specific Milestone Messages

Some of the diagnostic tests have additional milestone messages that are test specific.

Test 1 Milestone Messages

The following message is displayed during Test 1 before executing the `Reset Channel` command:

```
reset channel x
```

Variable	Description
x	Specifies the ESCON channel under test.

Test 2 Milestone Messages

The following milestone messages are displayed during Test 2:

```
.   Testing interrupt status I.D.'s and levels.
    Level = lvl
```

Variable	Description
lvl	Specifies the interrupt level under test. During testing, one or more periods (.) are appended to the `Level` message.

Test 3 Milestone Messages

Test 3 milestone messages are associated with the on-board diagnostic tests. The following messages are displayed during Test 3:

```
Running on-board micro-code tests.
.Test 1 - Uart Local Loopback Test PASSED
.Test 2 - VIC Write/Read Register Test PASSED
.Test 3 - SLC Write/Read Register Test PASSED
.Test 4 - Buffer Memory Write/Read Test PASSED
.Test 5 - Interrupt Test PASSED
.Test 7 - ESCON load micro-code PASSED
```

Note – The test displays either `PASSED` or `FAILED` for each microcode test.

Test 5 Milestone Messages

The following milestone messages are displayed during Test 5:

- When the test begins downloading microcode, it displays:

```
.   Commanding board to download microcode and initiate rings.
```

- Each time a command is issued to the board, the test displays:

```
.   Sending cmd command.
```

Variable	Description
cmd	Specifies the command sent to the board.

Test 9 Milestone Messages

The following milestone messages are displayed during Test 9:

- When the on-board microcode test starts, the diagnostic displays one of the following messages:

```
.   Running on-board micro-code tests to channel n.

.   Running on-board micro-code test to channel n.
```

Variable	Description
n	Specifies the channel under test.

- When the test begins downloading microcode, it displays:

```
. Commanding board to download microcode and initiate rings to channel n.
  sent ring addr: 0xsentaddr, response ring addr: 0xrespaddr
```

Variable	Description
n	Specifies the channel under test.
sentaddr	Specifies the send ring address.
respaddr	Specifies the response ring address.

- The test identifies the buffer addresses with the following messages:

```
channel n output addr: 0xoutaddr, size:0xoutsize
channel n input addr: 0xinaddr, size:0xinsize
```

Variable	Description
n	Specifies the channel under test.
outaddr	Specifies the address of the output buffer.
outsize	Specifies the size of the output buffer.
inaddr	Specifies the address of the input buffer.
insize	Specifies the size of the input buffer.

- Each time a command is issued to the channel, the test displays a message in one of the following formats:

```
.   Sending cmd command.

.   Sending cmd command to channel n.

.   Sending cmd command (0xcode) to channel n.
```

Variable	Description
cmd	Specifies the command sent to the channel or board.
n	Specifies the channel under test.
code	Specifies the code associated with this command.

- When the test sends a command to stop buffer memory transfers, it displays:

```
diag ring command (0x07) end buf mem xfers to channel n.
```

Variable	Description
n	Specifies the channel under test.

Test 10 Milestone Messages

The following milestone messages are displayed during Test 10:

- When the test resets the board, it displays:

```
.   Reset board
```

- When the on-board microcode test starts, the diagnostic displays:

```
.   Running on-board micro-code test to channel n.
```

Variable	Description
n	Specifies the channel under test.

■ When the test begins downloading microcode, it displays:

```
. Commanding board to download microcode and initiate rings to channel n.
  sent ring addr: 0xsentaddr, response ring addr: 0xrespaddr
```

Variable	Description
n	Specifies the channel under test.
sentaddr	Specifies the send ring address.
respaddr	Specifies the response ring address.

■ When the test sends an `Online Ring` command with failover enabled, it displays:

```
.   Sending online ring command with failover to channel n
```

Variable	Description
n	Specifies the channel under test.

■ Each time a command is issued to the channel, the test displays:

```
.   Sending cmd command to channel n.
```

Variable	Description
cmd	Specifies the command sent to the channel.
n	Specifies the channel under test.

Pass and Error Count Messages

If the diagnostic is running during the automatic testing sequence, it displays PASS when the program executes successfully. If the program fails, the diagnostic displays FAIL.

If the diagnostic is running in interactive mode, the following message is displayed after each pass is completed:

```
Pass Count: pppppppp
```

If an error occurs, the program displays the number of errors at the end of each pass in the following format:

```
Pass Count: pppppppp, Error Count: eeeeeeee
```

Variable	Description
pppppppp	Specifies the number of program passes completed.
eeeeeeee	Specifies the number of errors encountered since the program started execution.

Error Messages

If an error is detected during interactive mode testing, the diagnostic displays an error header and message in the following format:

```
Extended ESCON Diagnostic Failed      Time:hh:mm:ss
Board CSR Addr: 0xaaaaaaaa - 0xbbbbbbbb
Error: Slot s: CPU n: ESCON: Test t:  test name
error message
```

Variable	Description
hh:mm:ss	Specifies the system elapsed time when the error occurred.
aaaaaaaa	Specifies the starting address of the memory address range assigned to this board.
bbbbbbbb	Specifies the ending address of the memory address range assigned to this board.
s	Specifies the slot number of the board under test. When the diagnostic is running in interactive mode, the board slot number is always reported as 0.
n	Specifies the number of the CPU running the diagnostic.
t	Specifies the number of the failing test. If an error occurs during the initialization sequence, the test number is reported as 0.
test name	Specifies the name of the failing test.
error message	Specifies one of the following types of error messages: Initialization Interrupt Command Execution Reset Channel On-board Diagnostic Cleanup Data Failover Download

Initialization Error Messages

If the Control/Status Register (CSR) address is not a multiple of 0x100 during the program initialization sequence for automatic testing, the diagnostic displays:

```
Invalid board CSR address
```

If the board is installed in slot 0, the diagnostic displays:

```
slot 0, Invalid slot number
```

If the ROM Monitor revision is not equal to the expected value, the diagnostic displays:

```
*** WARNING: Invalid ROM Monitor Version ***
```

If the program is unable to read the ROM Monitor revision, it displays:

```
MSC Read Monitor Revision Error
```

Interrupt Error Messages

If no interrupt is generated when a command byte is sent to the board, the diagnostic displays:

```
Interrupt timed out on VME level n
```

Variable	Description
n	Specifies the expected VME interrupt level from 1 through 7.

If no interrupt is generated when a specific command byte is sent to the board, the diagnostic displays:

```
Interrupt timed out for byte d of 16 on VME level n
```

Variable	Description
d	Specifies the failing command byte.
n	Specifies the expected VME interrupt level.

If the interrupt is generated but the Interrupt Status Vector contains the incorrect value, the diagnostic displays:

```
Interrupt Status Vector error, exp: ee  act: aa
```

Variable	Description
ee	Specifies the expected Interrupt Status ID Vector value.
aa	Specifies the actual Interrupt Status ID Vector value.

If the Interrupt Status ID contains the incorrect value, the diagnostic displays:

```
Interrupt Status I.D. error, exp: ee  act: aa
```

Variable	Description
ee	Specifies the expected Interrupt Status ID value.
aa	Specifies the actual Interrupt Status ID value.

If no interrupt is generated by a `Halt Ring` command, the diagnostic displays:

```
Interrupt timed out waiting for halt ring response on VME level n
```

Variable	Description
n	Specifies the expected VME interrupt level.

Command Execution Error Messages

If an error occurs when the board is receiving a command message byte, the diagnostic displays an error message in one of the following formats:

```
Error receiving byte, Error code: code

Error receiving byte d, Error code: code
```

Variable	Description
d	Specifies the command byte.
code	Specifies an error code associated with the board's error status. The error codes are described in the *Dual Channel VME to ESCON Interface Engineering Summary.*

Reset Channel Error Messages

If an error occurs during the execution of the `Reset Channel` command, the diagnostic displays:

```
Reset not accepted or error on reset, Error code: code
```

Variable	Description
code	Specifies an error code associated with the board's error status. The error codes are described in the *Dual Channel VME to ESCON Interface Engineering Summary.*

If no response is received to a `Reset Channel` command, the diagnostic displays:

```
No response from channel reset demand on channel x
```

Variable	Description
x	Specifies the channel under test.

If an error occurs after a `Reset Channel` command, the diagnostic displays:

```
VME Interface test error, Error code: code
```

Variable	Description
code	Specifies an error code associated with the board's error status. The error codes are described in the *Dual Channel VME to ESCON Interface Engineering Summary.*

On-board Diagnostic Error Messages

If the response ring parameters received from a `Halt` or `Offline Ring` command are incorrect, the diagnostic displays:

```
SENT:       command (0xcode)
RECEIVED:   Response ring command   = cmd
            Error code              = xx
```

Variable	Description
command	Specifies the name of the failing command.
code	Specifies the hexadecimal command code value.
cmd	Specifies the hexadecimal response ring command code value.
xx	Specifies an error code associated with the board's error status. The error codes are described in the *Dual Channel VME to ESCON Interface Engineering Summary.*

If the response ring parameters received from a `Write` or `Read Diagnostic Ring` command are incorrect, the diagnostic displays:

```
SENT:       command (0xcode)
RECEIVED:  Bad Command Response = cmd
cmd: xx, stat: stat, parm1: pp1, parm2: pp2
```

Variable	Description
command	Specifies the name of the failing command.
code	Specifies the hexadecimal command code value.
cmd	Specifies the hexadecimal response ring command code value.
xx	Specifies the ring command.
stat	Specifies the ring error status.
pp1	Specifies response ring parameter 1.
pp2	Specifies response ring parameter 2.

Cleanup Error Messages

If an error results from the Master Reset during the diagnostic cleanup sequence, the diagnostic displays:

```
Master Reset Failed on channel x, Status = status
```

Variable	Description
x	Specifies the channel (0 or 1).
status	Specifies the status received.

If an error occurs while reading the real-time clock during the diagnostic cleanup sequence, the diagnostic displays:

```
MSC Read Real Time Clock Error
```

Data Error Messages

If a data error is detected, the diagnostic displays:

```
Data Error(s)
Data exp: 0xexp, rcvd: 0xrcvd, at addr 0xaddr
```

Variable	Description
exp	Specifies the data value expected.
rcvd	Specifies the actual data received.
addr	Specifies the address of the data received.

Failover Error Messages

If the microcode tests fail during Test 10, the diagnostic displays:

```
On-board micro-code test failed.
Error receiving byte, Error code: x
```

Variable	Description
x	Specifies the status.

If the External Loopback test fails during Test 10, the diagnostic displays:

```
This subtest requires the external loopback connector
to be disconnected from channel n of primary node.
```

Variable	Description
n	Specifies the channel number.

Download Error Messages

If an error occurs while downloading microcode, the diagnostic displays one of the following:

```
Unsucessful download ...

Unsucessful download to channel n ...
```

Variable	Description
n	Specifies the channel number.