

Title:	DIL1 Test Scripts	File Number:	
Security Level:	Public	Release Status:	v 1.4
Owner Group:	RIOLAB	Revision Date:	31 Jan 2007

DIL1 TEST SCRIPTS

This document and all of its contents are proprietary to RIOLAB. No part of its contents may be used, copied, disclosed or conveyed to any party in any manner whatsoever without prior written permission from RIOLAB. Information is subject to change without notice. All trademarks are the property of their respective owners.

Copyright 2007. All rights reserved.

Title:	DIL1 Test Scripts	File Number:	
Security Level:	Public	Release Status:	v 1.4
Owner Group:	RIOLAB	Revision Date:	31 Jan 2007

1 REVISION HISTORY

Release Status	Rev. #	Date	Author/ Reviser	Group	Description of Changes
Draft	0.8	27 May 2006	FET	Engineering	Initial Draft.
First release	1.0	1 June 2006	FET	Engineering	First release
Update	1.1	2 June 2006	FET	Engineering	Update to scripts
Update	1.2	4 June 2006	FET	Engineering	Update to scripts
Update	1.3	8 June 2006	FET	Engineering	Update from feedback
Update	1.4	31 Jan 2007	FET	Engineering	Update

Title:	DIL1 Test Scripts	File Number:	
Security Level:	Public	Release Status:	v 1.4
Owner Group:	RIOLAB	Revision Date:	31 Jan 2007

TABLE OF CONTENTS

1 REVISION HISTORY2

2 INTRODUCTION5

 2.1 Purpose5

 2.2 Background5

 2.3 Related Documents5

3 TEST SCRIPTS6

 3.1 Overview6

 3.1.1 Comments6

 3.1.2 Commands6

 3.1.3 Command Parameters6

 3.2 Command Descriptions6

 3.2.1 User Prompt6

 3.2.2 Maintenance Read7

 3.2.3 Maintenance Write7

 3.2.4 Read Request8

 3.2.5 Write with Response8

 3.2.6 Write9

 3.2.7 Streaming Write9

 3.3 Checklist Test Script Examples10

 3.3.1 Test Scripts ID 25310

 3.3.2 Test Scripts ID 35711

 3.3.3 Test Scripts ID 36011

 3.3.4 Test Scripts ID 36111

 3.3.5 Test Scripts ID 36212

 3.3.6 Test Scripts ID 36312

 3.3.7 Test Scripts ID 36412

 3.3.8 Test Scripts ID 36513

 3.3.9 Test Scripts ID 41214

 3.3.10 Test Scripts ID 41714

 3.3.11 Test Scripts ID 57715

 3.3.12 Test Scripts ID 57815

 3.3.13 Test Scripts ID 57915

Title:	DIL1 Test Scripts	File Number:	
Security Level:	Public	Release Status:	v 1.4
Owner Group:	RIOLAB	Revision Date:	31 Jan 2007

3.3.14	Test Scripts ID 582	15
3.3.15	Test Scripts ID 583	15
3.3.16	Test Scripts ID 584	16
3.4	I/O Transaction Test Script Examples	16
3.4.1	Test Scripts ID Nwrite_R.....	16
3.4.2	Test Scripts ID Nwrite.....	20
3.4.3	Test Scripts ID Nread	24
3.4.4	Test Scripts ID sWrite.....	25
3.5	Miscellaneous Script Examples	28
3.5.1	Routing table Scripts.....	28

Title:	DIL1 Test Scripts	File Number:	
Security Level:	Public	Release Status:	v 1.4
Owner Group:	RIOLAB	Revision Date:	31 Jan 2007

2 INTRODUCTION

2.1 Purpose

This document describes the RapidIO test scripts associated with DIL1 testing. This is a living document that will be revised each time the DIL1 test scripts or syntax change.

2.2 Background

RapidFET is a network management and diagnostic tool used in RIOLAB to control interoperability testing and report results. The test scripts defined in this document are generated as an output from RapidFET.

These scripts provide detailed visibility of what RapidIO transactions were issued in any given checklist test. They also provide a detailed account of the results and if the transaction itself was completed successfully or not. This information can be of value in understanding why a particular checklist test passed or failed.

In lieu of using RapidFET to run interoperability tests from an endpoint, these scripts may also be used by a Script Command Interpreter as an input to repeat the same tests.

2.3 Related Documents

The following documents were used in the creation of, or are referenced in this document:

- DIL1_Test_Description.doc

Title:	DIL1 Test Scripts	File Number:	
Security Level:	Public	Release Status:	v 1.4
Owner Group:	RIOLAB	Revision Date:	31 Jan 2007

3 TEST SCRIPTS

3.1 Overview

The following sections describe the syntax associated with DIL1 test scripts.

3.1.1 Comments

Comment lines begin with a # character and exist for the purpose of providing a human readable description of the test or subsection of the test or to indicate when human interaction takes place.

Examples:

```
#Host Base Device ID CSR Host_base_deviceID field reset value is 0xFFFF.
```

3.1.2 Commands

Commands are representative of RapidIO transactions or user interaction prompts. Commands include parameters associated with destination, memory location, data, comments and status.

3.1.3 Command Parameters

For each command there is a list of associated input and output parameters. A script interpreter may choose to ignore output parameters. To facilitate this output parameters are preceded by a comment “#” character.

- All parameters are hexadecimal values (except for the “Prompt” command where the input parameter is a string.)
- All values and parameters are in big-endian [32:0] format.
- The parameters are not case sensitive.
- Commands begin on a new line, starting with the command name and are followed by the list of parameters separated by spaces and terminated with <CR><LF>.

3.2 Command Descriptions

3.2.1 User Prompt

Syntax:

Prompt:

Parameters:

Input:

Title:	DIL1 Test Scripts	File Number:	
Security Level:	Public	Release Status:	v 1.4
Owner Group:	RIOLAB	Revision Date:	31 Jan 2007

Text String

Output:

None

Example Script Line:

Prompt: Reset the board now

3.2.2 Maintenance Read

Syntax:

Maint_read

Parameters:

Input:

Destination ID (FFFFFFFF means local device)

Priority (2bits)

Hop Count (8bits)

Offset Address (24bits)

Num Bytes (8bits)

Output:

(Comment character)

Data Word #1 (32 bits)

Data Word #2 (32 bits)

.

.

Data Word #n (32 bits, n is [Num Bytes / 4])

Last parameter is a command Succeeded or Failed notation

Example Script Line:

Maint_read ffffffff 0 0 0 4 # 30 Succeeded

3.2.3 Maintenance Write

Syntax:

Maint_write

Parameters:

Input:

Destination ID (FFFFFFFF means local device)

Priority (2bits)

Hop Count (8bits)

Offset Address (24bits)

Num Bytes (8bits)

Title:	DIL1 Test Scripts	File Number:	
Security Level:	Public	Release Status:	v 1.4
Owner Group:	RIOLAB	Revision Date:	31 Jan 2007

Data Word #1 (32 bits)
 Data Word #2 (32 bits)
 .
 .
 Data Word #n (32 bits, n is [Num Bytes / 4])

Output:
 # (Comment character)
 Last parameter is a command Succeeded or Failed notation

Example Script Line:

```
Maint_write ff 0 0 11070 4 80000000 # Succeeded
```

3.2.4 Read Request

Syntax:

Nread

Parameters:

Input:

Destination ID
 Priority (2bits)
 Offset Address (66bits)
 Num Bytes (8bits)

Output:

(Comment character)
 Data Word #1 (32 bits)
 Data Word #2 (32 bits)
 .
 .
 Data Word #n (32 bits, n is [Num Bytes / 4])
 Last parameter is a command Succeeded or Failed notation

Example Script Line:

```
Nread 5 0 8012345678 c # aaaaaaaaa aaaaaaaaa ffffffff Succeeded
```

3.2.5 Write with Response

Syntax:

Nwrite_R

Parameters:

Input:

Destination ID
 Priority (2bits)

Title:	DIL1 Test Scripts	File Number:	
Security Level:	Public	Release Status:	v 1.4
Owner Group:	RIOLAB	Revision Date:	31 Jan 2007

Offset Address (66bits)
 Num Bytes (8bits)
 Data Word #1 (32 bits)
 Data Word #2 (32 bits)
 .
 .
 Data Word #n (32 bits, n is [Num Bytes / 4])

Output:
 # (Comment character)
 Last parameter is a command Succeeded or Failed notation

Example Script Line:

Nwrite_R 5 2 8012345678 8 aaaaaaaa aaaaaaaa # Succeeded

3.2.6 Write

Syntax:

Nwrite

Parameters:

Input:
 Destination ID
 Priority (2bits)
 Offset Address (66bits)
 Num Bytes (8bits)
 Data Word #1 (32 bits)
 Data Word #2 (32 bits)
 .
 .
 Data Word #n (32 bits, n is [Num Bytes / 4])

Output:
 # (Comment character)
 Last parameter is a command Succeeded or Failed notation

Example Script Line:

Nwrite 5 2 8012345678 8 55555555 55555555 # Succeeded

3.2.7 Streaming Write

Syntax:

Swrite

Parameters:

Input:
 Destination ID

Title:	DIL1 Test Scripts	File Number:	
Security Level:	Public	Release Status:	v 1.4
Owner Group:	RIOLAB	Revision Date:	31 Jan 2007

Priority (2bits)
 Offset Address (66bits)
 Num Bytes (8bits)
 Data Word #1 (32 bits)
 Data Word #2 (32 bits)
 .
 .
 Data Word #n (32 bits, n is [Num Bytes / 4])

Output:
 # (Comment character)
 Last parameter is a command Succeeded or Failed notation

Example Script Line:

```

Swrite 5 2 8012345678 200 ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
# Succeeded
  
```

3.3 Checklist Test Script Examples

The following test script examples use the following common parameters:

Destination ID = 5
 Priority (2bits) = 0

3.3.1 Test Scripts ID 253

#Test 253: A RapidIO device will accept packets of length up to 276 bytes. (Testing target device)

Prompt: Get Memory Address

#Write a packet with payload of 256 bytes of known content to a memory location

```

Nwrite 5 2 004000100 100 0 1 2 3 4 5 6 7 8 9 a b c d e f 10 11 12 13 14 15 16 17 18 19
1a 1b 1c 1d 1e 1f 20 21 22 23 24 25 26 27 28 29 2a 2b 2c 2d 2e 2f 30 31 32 33 34 35 36
37 38 39 3a 3b 3c 3d 3e 3f # Succeeded
  
```

#Read 256 bytes back from the same location previously written to confirm packet was correctly accepted

```

Nread 5 0 0 004000100 100 # 0 1 2 3 4 5 6 7 8 9 a b c d e f 10 11 12 13 14 15 16 17 18
19 1a 1b 1c 1d 1e 1f 20 21 22 23 24 25 26 27 28 29 2a 2b 2c 2d 2e 2f 30 31 32 33 34 35
36 37 38 39 3a 3b 3c 3d 3e 3f Succeeded
  
```

Title:	DIL1 Test Scripts	File Number:	
Security Level:	Public	Release Status:	v 1.4
Owner Group:	RIOLAB	Revision Date:	31 Jan 2007

#Checklist Test: Passed

3.3.2 Test Scripts ID 357

#Test 357: Component Tag CSR; reset value is 0x00000000.

#Issue a maintenance write to the Component Tag CSR

Maint_write 5 0 1 6c 4 12345678 # Succeeded

#Issue read it back to confirm it was successfully written

Maint_read 5 0 1 6c 4 # 12345678 Succeeded

Prompt: Reset the board now

#Read of the Component Tag CSR again. The Component tag register is compared to an expected value of 0x00000000.

Maint_read 5 0 1 6c 4 # 12345678 Succeeded

#Checklist Test: Failed

3.3.3 Test Scripts ID 360

#Test 360: Host Base Device ID CSR Host_base_deviceID field reset value is 0xFFFF.

#Issue a maintenance read to the Host Base Device ID CSR

Maint_read 5 0 1 68 4 # ffff Succeeded

#It is 0xFFFF, write a value of 0x1234 to this field

Maint_write 5 0 1 68 4 1234 # Succeeded

Prompt: Reset the board now

#Issue a maintenance read of the Host Base Device ID CSR

Maint_read 5 0 1 68 4 # 1234 Succeeded

#Checklist Test: Failed

3.3.4 Test Scripts ID 361

#Test 361: When Host Base Device ID CSR Host_base_deviceID field value is 0xFFFF, the field value can be changed to any value.

#Issue a maintenance read of the Host Base Device ID CSR

Maint_read 5 0 1 68 4 # 1234 Succeeded

Prompt: Reset the board now

#Confirm Host_base_deviceID field is now equal to 0xFFFF

Maint_read 5 0 1 68 4 # 1234 Succeeded

#Checklist Test: Failed

Title:	DIL1 Test Scripts	File Number:	
Security Level:	Public	Release Status:	v 1.4
Owner Group:	RIOLAB	Revision Date:	31 Jan 2007

3.3.5 Test Scripts ID 362

#Test 362: When Host Base Device ID CSR Host_base_deviceID field value is not 0xFFFF, the field value will change to 0xFFFF when a value equal to the current field value is written.

#Issue a maintenance read to verify that the Host Base Device ID CSR Host_base_deviceID field is not 0xFFFF

Maint_read 5 0 1 68 4 # 1234 Succeeded

#Verify that by writing the same value back to this register field that it currently holds that the Host_base_deviceID field will return to 0xFFFF

Maint_write 5 0 1 68 4 1234 # Succeeded

Maint_read 5 0 1 68 4 # ffff Succeeded

#Checklist Test: Passed

3.3.6 Test Scripts ID 363

#Test 363: When Host Base Device ID CSR Host_base_deviceID field value is not 0xFFFF, the field value does not change when a value not equal to the current field value is written.

#Issue a maintenance read to verify that the Host Base Device ID CSR Host_base_deviceID field is not 0xFFFF

Maint_read 5 0 1 68 4 # ffff Succeeded

#It is 0xFFFF, write any other value to this register using a maintenance write.

Maint_write 5 0 1 68 4 1234 # Succeeded

#Verify that by writing any other value than the one stored in the Host_base_deviceID field that the register field remains unchanged

Maint_write 5 0 1 68 4 1235 # Succeeded

Maint_read 5 0 1 68 4 # 1234 Succeeded

#Checklist Test: Passed

3.3.7 Test Scripts ID 364

#Test 364: When Host Base Device ID CSR Host_base_deviceID field value is 0xFFFF, and 0xFFFF is written to the field, subsequent writes of values not equal to 0xFFFF behave as per 4.2.2.B, 4.2.2.C, and 4.2.2.D.

#Issue a maintenance Read and determine what the content is in the Host Base Device ID CSR Host_base_deviceID field

Maint_read 5 0 1 68 4 # 1234 Succeeded

#not 0xFFFF, so write the contents of the register field back to the register field

Title:	DIL1 Test Scripts	File Number:	
Security Level:	Public	Release Status:	v 1.4
Owner Group:	RIOLAB	Revision Date:	31 Jan 2007

```

Maint_write 5 0 1 68 4 1234 # Succeeded
#verify that the value of the register field returns to 0xFFFF
Maint_read 5 0 1 68 4 # ffff Succeeded
#Write a value of 0xFFFF to the register field
Maint_write 5 0 1 68 4 ffff # Succeeded
#Write a value of other than 0xFFFF to the register field and read it back the value of the
register field to verify it changes
Maint_write 5 0 1 68 4 1234 # Succeeded
Maint_read 5 0 1 68 4 # ffff Succeeded
#Checklist Test: Failed
  
```

3.3.8 Test Scripts ID 365

```

#Test 365: The Component Tag CSR component_tag field can be written to any value.
#Issue a maintenance write to the Component Tag CSR component_tag field with any
value
Maint_write 5 0 1 6c 4 0 # Succeeded
#Confirm that this value has been accepted using a maintenance read
Maint_read 5 0 1 6c 4 # 0 Succeeded
#Issue a maintenance write to the Component Tag CSR component_tag field with any
value
Maint_write 5 0 1 6c 4 1c72 # Succeeded
#Confirm that this value has been accepted using a maintenance read
Maint_read 5 0 1 6c 4 # 1c72 Succeeded
#Issue a maintenance write to the Component Tag CSR component_tag field with any
value
Maint_write 5 0 1 6c 4 38e3 # Succeeded
#Confirm that this value has been accepted using a maintenance read
Maint_read 5 0 1 6c 4 # 38e3 Succeeded
#Issue a maintenance write to the Component Tag CSR component_tag field with any
value
Maint_write 5 0 1 6c 4 5555 # Succeeded
#Confirm that this value has been accepted using a maintenance read
Maint_read 5 0 1 6c 4 # 5555 Succeeded
#Issue a maintenance write to the Component Tag CSR component_tag field with any
value
Maint_write 5 0 1 6c 4 71c7 # Succeeded
#Confirm that this value has been accepted using a maintenance read
  
```

Title:	DIL1 Test Scripts	File Number:	
Security Level:	Public	Release Status:	v 1.4
Owner Group:	RIOLAB	Revision Date:	31 Jan 2007

```

Maint_read 5 0 1 6c 4 # 71c7 Succeeded
#Issue a maintenance write to the Component Tag CSR component_tag field with any
value
Maint_write 5 0 1 6c 4 8e38 # Succeeded
#Confirm that this value has been accepted using a maintenance read
Maint_read 5 0 1 6c 4 # 8e38 Succeeded
#Issue a maintenance write to the Component Tag CSR component_tag field with any
value
Maint_write 5 0 1 6c 4 aaaa # Succeeded
#Confirm that this value has been accepted using a maintenance read
Maint_read 5 0 1 6c 4 # aaaa Succeeded
#Issue a maintenance write to the Component Tag CSR component_tag field with any
value
Maint_write 5 0 1 6c 4 c71c # Succeeded
#Confirm that this value has been accepted using a maintenance read
Maint_read 5 0 1 6c 4 # c71c Succeeded
#Issue a maintenance write to the Component Tag CSR component_tag field with any
value
Maint_write 5 0 1 6c 4 ffff # Succeeded
#Confirm that this value has been accepted using a maintenance read
Maint_read 5 0 1 6c 4 # ffff Succeeded
#Checklist Test: Passed
  
```

3.3.9 Test Scripts ID 412

```

#Test 412: MAINTENANCE read request generates a MAINTENANCE read response.
#Issue a maintenance read of the same register and verify that the contents holds the
expected value
Maint_write 5 0 1 6c 4 1234 # Succeeded
#Read the data back and verify that the data read is the same value as the data written in
the previous step
Maint_read 5 0 1 6c 4 # 1234 Succeeded
#Checklist Test: Passed
  
```

3.3.10 Test Scripts ID 417

```

#Test 417: MAINTENANCE write request generates a MAINTENANCE write
responses.
#Issue a maintenance write to the Component TAG register
  
```

Title:	DIL1 Test Scripts	File Number:	
Security Level:	Public	Release Status:	v 1.4
Owner Group:	RIOLAB	Revision Date:	31 Jan 2007

```
Maint_write 5 0 1 6c 4 1212 # Succeeded
#Issue a maintenance read of the same register and verify that the contents holds the
expected value
Maint_read 5 0 1 6c 4 # 1212 Succeeded
#Checklist Test: Passed
```

3.3.11 Test Scripts ID 577

```
#Test 577: MAINTENANCE read transaction – TITLE HEADING ONLY
```

3.3.12 Test Scripts ID 578

```
#Test 578: MAINTENANCE read request size of 4 bytes must be supported.
#Issue a maintenance write to the Component TAG register
Maint_write 5 0 1 6c 4 12345678 # Succeeded
#Issue a maintenance read to the Component TAG register and confirm that the 4 byte
value is received same as written
Maint_read 5 0 1 6c 4 # 12345678 Succeeded
#Checklist Test: Passed
```

3.3.13 Test Scripts ID 579

```
#Test 579: MAINTENANCE read request generates a MAINTENANCE read response.
#Issue a maintenance write to the Component TAG register
Maint_write 5 0 1 6c 4 0 # Succeeded
#Issue a maintenance read of the same register and verify that the contents holds the
expected value
Maint_read 5 0 1 6c 4 # 0 Succeeded
#Checklist Test: Passed
```

3.3.14 Test Scripts ID 582

```
#Test 582: MAINTENANCE write transaction - TITLE HEADING ONLY.
```

3.3.15 Test Scripts ID 583

```
#Test 583: MAINTENANCE write request may be for 4 bytes.
#Issue a maintenance write to the Component TAG register
Maint_write 5 0 1 6c 4 12345678 # Succeeded
#Issue a maintenance read to the same register and confirm that the value read is the
same 4 bytes that was written
```

Title:	DIL1 Test Scripts	File Number:	
Security Level:	Public	Release Status:	v 1.4
Owner Group:	RIOLAB	Revision Date:	31 Jan 2007

```
Maint_read 5 0 1 6c 4 # 12345678 Succeeded
#Checklist Test: Passed
```

3.3.16 Test Scripts ID 584

```
#Test 584: MAINTENANCE write request generates a MAINTENANCE write response.
#Issue a maintenance write to the Component TAG register
Maint_write 5 0 1 6c 4 12345678 # Succeeded
#Issue a maintenance read of the same register and verify that the contents holds the
expected value
Maint_read 5 0 1 6c 4 # 12345678 Succeeded
#Checklist Test: Passed
```

3.4 I/O Transaction Test Script Examples

The following test script examples use the following common parameters:

```
Destination ID = 5
Priority (2bits) = 0
```

3.4.1 Test Scripts ID Nwrite_R

```
#Test Nwrite_R transaction: Demonstrate the Nwrite_R with a payload of 8 bytes
Prompt: Get Memory Address
#Write a payload of 0xFFFFFFFF word values write 512 bytes of data to a memory
location
Swrite 5 2 004000100 200 ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff #
Succeeded
#Write a data payload comprised of a word value, issue an nWrite with Response
transaction to an appropriate memory location
Nwrite_R 5 2 004000100 8 55555555 55555555 # Succeeded
#Issue an nRead transaction with a larger payload payload size to the same memory
location
```


Title:	DIL1 Test Scripts	File Number:	
Security Level:	Public	Release Status:	v 1.4
Owner Group:	RIOLAB	Revision Date:	31 Jan 2007

Nread 5 0 0 004000100 24 # 55555555 55555555 55555555 55555555 55555555
55555555 55555555 55555555 ffffffff Succeeded

#Write a payload of 0xFFFFFFFF word values write 512 bytes of data to a memory location

Swrite 5 2 004000100 200 ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff #
Succeeded

#Write a data payload comprised of a word value, issue an nWrite with Response transaction to an appropriate memory location

Nwrite_R 5 2 004000100 20 aaaaaaaaa aaaaaaaaa aaaaaaaaa aaaaaaaaa aaaaaaaaa aaaaaaaaa
aaaaaaaa aaaaaaaaa # Succeeded

#Issue an nRead transaction with a larger payload payload size to the same memory location

Nread 5 0 0 004000100 24 # aaaaaaaaa aaaaaaaaa aaaaaaaaa aaaaaaaaa aaaaaaaaa aaaaaaaaa
aaaaaaaa aaaaaaaaa ffffffff Succeeded

#Checklist Test: Passed

#Test Nwrite_R transaction: Demonstrate the Nwrite_R with a payload of 256 bytes

#Write a payload of 0xFFFFFFFF word values write 512 bytes of data to a memory location

Swrite 5 2 004000100 200 ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff
ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff ffffffff #
Succeeded

#Write a data payload comprised of a word value, issue an nWrite with Response transaction to an appropriate memory location

Title:	DIL1 Test Scripts	File Number:	
Security Level:	Public	Release Status:	v 1.4
Owner Group:	RIOLAB	Revision Date:	31 Jan 2007

aaaaaaaa aaaaaaaaa aaaaaaaaa aaaaaaaaa aaaaaaaaa aaaaaaaaa aaaaaaaaa aaaaaaaaa
 aaaaaaaaa aaaaaaaaa aaaaaaaaa aaaaaaaaa aaaaaaaaa aaaaaaaaa aaaaaaaaa aaaaaaaaa
 aaaaaaaaa aaaaaaaaa aaaaaaaaa aaaaaaaaa ffffffff Succeeded

#Checklist Test: Passed

3.4.3 Test Scripts ID Nread

#Test Nread transaction: Demonstrate the Nread with a payload of 8 bytes

Prompt: Get Memory Address

#Issue an nWrite with Response transaction to an appropriate memory location

Nwrite_R 5 2 004000100 8 55555555 55555555 # Succeeded

#Read the data back and compare it to what was written

Nread 5 0 0 004000100 8 # 55555555 55555555 Succeeded

#Issue an nWrite with Response transaction to an appropriate memory location

Nwrite_R 5 2 004000100 8 aaaaaaaaa aaaaaaaaa # Succeeded

#Read the data back and compare it to what was written

Nread 5 0 0 004000100 8 # aaaaaaaaa aaaaaaaaa Succeeded

#Checklist Test: Passed

#Test Nread transaction: Demonstrate the Nread with a payload of 32 bytes

#Issue an nWrite with Response transaction to an appropriate memory location

Nwrite_R 5 2 004000100 20 55555555 55555555 55555555 55555555 55555555
 55555555 55555555 55555555 # Succeeded

#Read the data back and compare it to what was written

Nread 5 0 0 004000100 20 # 55555555 55555555 55555555 55555555 55555555
 55555555 55555555 55555555 Succeeded

#Issue an nWrite with Response transaction to an appropriate memory location

Nwrite_R 5 2 004000100 20 aaaaaaaaa aaaaaaaaa aaaaaaaaa aaaaaaaaa aaaaaaaaa
 aaaaaaaaa aaaaaaaaa # Succeeded

#Read the data back and compare it to what was written

Nread 5 0 0 004000100 20 # aaaaaaaaa aaaaaaaaa aaaaaaaaa aaaaaaaaa aaaaaaaaa
 aaaaaaaaa aaaaaaaaa Succeeded

#Checklist Test: Passed

#Test Nread transaction: Demonstrate the Nread with a payload of 256 bytes

#Issue an nWrite with Response transaction to an appropriate memory location

Nwrite_R 5 2 004000100 100 55555555 55555555 55555555 55555555 55555555
 55555555 55555555 55555555 55555555 55555555 55555555 55555555 55555555
 55555555 55555555 55555555 55555555 55555555 55555555 55555555 55555555
 55555555 55555555 55555555 55555555 55555555 55555555 55555555 55555555

Title:	DIL1 Test Scripts	File Number:	
Security Level:	Public	Release Status:	v 1.4
Owner Group:	RIOLAB	Revision Date:	31 Jan 2007

```
ffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff
ffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff
ffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff
ffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff
ffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff
ffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff
ffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff
ffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff #
Succeeded
```

#Write a data payload comprised of a word value, issue an sWrite to an appropriate memory location

```
Swrite 5 2 004000100 20 55555555 55555555 55555555 55555555 55555555 55555555
55555555 55555555 # Succeeded
```

#Issue an nRead transaction with a larger payload payload size to the same memory location

```
Nread 5 0 0 004000100 24 # 55555555 55555555 55555555 55555555 55555555
55555555 55555555 55555555 ffffffff Succeeded
```

#Write a payload of 0xFFFFFFFF word values write 512 bytes of data to a memory location

```
Swrite 5 2 004000100 200 fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff
ffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff
ffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff
ffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff
ffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff
ffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff
ffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff
ffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff fffffff #
Succeeded
```

#Write a data payload comprised of a word value, issue an sWrite to an appropriate memory location

```
Swrite 5 2 004000100 20 aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
aaaaaaa aaaaaaaa # Succeeded
```

#Issue an nRead transaction with a larger payload payload size to the same memory location

```
Nread 5 0 0 004000100 24 # aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa aaaaaaaa
aaaaaaa aaaaaaaa ffffffff Succeeded
```

#Checklist Test: Passed

#Test sWrite transaction: Demonstrate the sWrite with a payload of 256 bytes

#Write a payload of 0xFFFFFFFF word values write 512 bytes of data to a memory location

Title:	DIL1 Test Scripts	File Number:	
Security Level:	Public	Release Status:	v 1.4
Owner Group:	RIOLAB	Revision Date:	31 Jan 2007

Maint_write ff 0 0 70 8 a ff # Succeeded
Maint_write ff 0 0 70 8 b ff # Succeeded
Maint_write ff 0 0 70 8 c 12 # Succeeded
Maint_write ff 0 0 70 8 d ff # Succeeded
Maint_write ff 0 0 70 8 e ff # Succeeded
Maint_write ff 0 0 70 8 f ff # Succeeded
Maint_write ff 0 0 70 8 10 ff # Succeeded
Maint_write ff 0 0 70 8 11 ff # Succeeded
Maint_write ff 0 0 70 8 12 ff # Succeeded

•

•

•

Maint_write ff 0 0 70 8 1ff ff # Succeeded
