# **ERADYNE 2017**

# **Di-Series Debugging Tools**

Tarra Marchetti | Teradyne | North Reading, MA| tarra.marchetti@teradyne.com

# Contents

Introduction	
Di-Series Alarm Viewer	
Log Files4	
Digital Test Editor Debugging Features5	
Debug Capture of a Test Program5	
Generating Test from Hardware7	
Loop and Sweep	
Conclusions	

## Introduction

Whether performing new TPS development, TPS rehost, or routine testing, there is always a chance of running into unexpected results that require debug. When those tests involve the Teradyne Di-Series digital test instrument, numerous tools are installed as part of the standard product. This paper will explain what these tools are, when they can be used, and how to use them.

## **Di-Series Alarm Viewer**

The Di-Series Alarm Viewer is installed with iStudio Digital Test Editor V3.2 and higher. This tool can be used to easily identify when a Di-Series instrument experiences an alarm condition from the test environment. These alarm conditions are: OverVoltage, OverCurrent, and OverTemperature.

The Di-Series Alarm Viewer is automatically started during the iStudio Digital Test Editor installation with software versions 4.3 and later. For earlier versions, the Alarm Viewer must be manually started. When an event occurs with the alarm viewer active, a notification bubble can be seen from the taskbar icon tray. The Alarm Viewer can be launched to view additional alarm information by selecting from the taskbar icon tray, or from the Start menu -> Teradyne -> Di-Series -> Alarm Viewer.



Figure 2 - Launching Di-Series Alarm Viewer from the Start menu

An OverVoltage alarm occurs when a Di-Series channel is subjected to an input voltage outside of the programmed voltage range, which could cause damage to the Di-Series instrument. When the event is detected the channel's relay is opened, which could cause unexpected TPS failures that require debugging. The Di-Series Alarm Viewer clearly identifies when an OverVoltage event is detected, which channel(s)

experienced the event, and what programmed voltage range the channel(s) was set to when the event occurred. A user can make use of this information to correct their test program or test setup.

An OverCurrent alarm occurs when current overage is detected on a Di-Series channel. When the event is detected the channel driver is tristated, which could cause unexpected TPS failures that require debugging. The Di-Series Alarm Viewer can also clearly identify when an OverCurrent event is detected and which channel(s) experienced the event so corrective actions can be taken in the test program or test setup.

An OverTemperature alarm occurs when a Di-Series instrument is subjected to temperatures that could potentially damage hardware. When this event is detected, the Di-Series pin electronics are shut down, which could cause TPS failures that require debug.

2017-04-14T13:34:10									Only	Range	Test Program
	Overtemperature	Channel	0,11	0	0	8	0	11		P6_N3	D:\Tarra\Wor
2017-04-14T13:34:10	Overtemperature	Channel	0,10	0	0	8	0	10		P6_N3	D:\Tarra\Wo
2017-04-14T13:34:26	Overcurrent	Channel	0,10	0	0	8	0	10		P3_3_N2_2	D:\Tarra\Wo
2017-04-14T13:34:38	Overvoltage	Channel	0, 4	0	0	8	0	4	Т	P6_N3	D:\Tarra\Wo
2017-04-14T13:34:38	Overvoltage	Channel	0,10	0	0	8	0	10	F	P6_N3	D:\Tarra\Wo

Figure 3 - Di-Series Alarm Viewer

This debug tool requires no user setup or maintenance and can be referenced whenever there is question around suspicious test results from the Di-Series instrument.

## **Log Files**

Test program failures may occur because of errors thrown by the Di-Series driver that are not handled properly in a test program or do not directly translate to a legacy instrument error when in simulation. Log files created by the driver are automatically updated when these errors occur, and they can be referenced to debug unexplained test failures.

Di-Series exceptions are logged in the terDi\_Exception.txt file located in the following location: "%LOCALAPPDATA%\Teradyne\DiDriver". This log file can be referenced during test program development if there is a chance exceptions are not being handled properly within a test program. Each exception is logged with the date and time of occurrence, exception type, and some additional information that can help identify at which Di-Series API the exception was thrown.

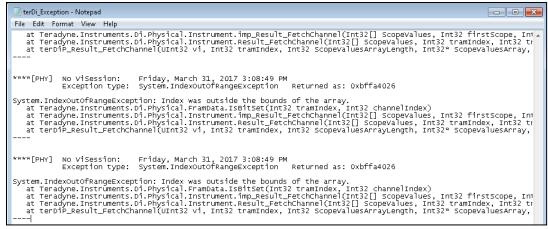


Figure 4 - Example Di-Series Exception Log

When emulating an M9-Series instrument with the Di-Series, a file is updated that logs when emulated instruments are created and closed, when unsupported function calls or features are used in the test program, and when errors occur. The default path for this M9-Series Interface log file is: C:\Documents and Settings\userName\Local Settings\Application Data\Teradyne\M9-Series Interface\ErrorLog.txt. This location can be changed to a preferred location through the M9-Series Interface Configuration Editor. The M9-Series Interface log file is especially useful if errors returned in an M9-Series emulated test program are not sufficient to debug the test program. Numerous Di-Series errors that cannot be mapped to an existing M9-Series error are reported as "TERM9\_ERROR\_INTERNAL", and this log file can provide additional information required to resolve the failure for these instances.

_	rrorLog - Notepad		
File	Edit Format View Help		
	31/2017 10:34 AM PSGUUT.exe 31/2017 10:34 AM PSGUUT.exe 31/2017 10:34 AM PSGUUT.exe 31/2017 10:34 AM PSGUUT.exe	<pre>TERM9::#0/TERDI::#0-2: Closing Di-Instrument. TERM9::#0/TERDI::#0-2: Creating new Di-Instrument using configuration file C:\Pro TERM9::#0/TERDI::#0-2: terM9_setLevelSetProbePRHI is only supported if the Di-Instru TERM9::#0/TERDI::#0-2: terM9_setLevelSetProbePRL0 is only supported if the Di-Instru TERM9::#0/TERDI::#0-2: terM9_setLevelSetProbePRL0 is only supported if the Di-Instru TERM9::#0/TERDI::#0-2: terM9_setPatternClutchEnable is only supported if the Di-Instru TERM9::#0/TERDI::#0-2: terM9_setPatternClutchEnable is only supported if the Di-Instru TERM9::#0/TERDI::#0-2: terM9_setPatternClutchEnable is only supported if the Di-Instru TERM9::#0/TERDI::#0-2: terM9_setSystemClutchEnable is only supported if the Di-Instru TERM9::#0/TERDI::#0-2: terM9_setSystemClutchLogicSelect is only supported if the Di-Inst TERM9::#0/TERDI::#0-2: terM9_setSystemClutchEnable is only supported if the Di-Inst TERM9::#0/TERDI::#0-2: terM9_setSystemClutchLogicSelect is only supported if the Di-Inst TERM9::#0/TERDI::#0-2: terM9_setSetAtternClutcLogicSelect is only supported if the Di-Inst TERM9::#0/TERDI::#0-2: terM9_setSetAtternClutChLogicSelect is only su</pre>	ogram Files (x86)\IVI Founda tent contains a utility modu ent contains a utility modu of instrument contains a utility wrent contains a utility mrent contains a utility mod rument contains a utility mod rices. -Instrument contains a utility mo strument contains a utility mo strument contains a utility mo strument contains a utility instrument contains a utility ment contains a utility ment contains a utility
•		III	Þ.

Figure 5 - Example M9-Series Interface log file

These log files are automatically generated and updated, and like the Alarm Viewer require no user setup or maintenance. These tools are a quick way to begin debugging unexpected test failures with little effort.

## **Digital Test Editor Debugging Features**

Various Di-Series debugging tools are available through the Digital Test Editor, which can be useful whether you are executing test programs though the M9-Series Interface or natively with Di-Series API's. When a test program fails, the Digital Test Editor may be used to clearly identify which channels are failing, if there are unexpected channel setup issues, and to easily loop a Di-Series test.

## **Debug Capture of a Test Program**

The Digital Test Editor has a Debug feature that can capture a test program as it is executed on Di-Series hardware into the graphical interface. This is especially useful when debugging complicated test programs

that do not provide sufficient Di-Series debug information or to graphically display failure patterns that are not obvious when reported in a test program. To utilize this feature of the Digital Test Editor create a new Digital Test from the iStudio program, and select "Debug -> Enable" from the top menu bar.

1	Digital Test* - iStu	dio Digital Test I	Editor					, • 🔀
-	ile Edit View		Debug Tools Help Enable					
_	Bignal Setup	· · ·	Remote Debugging					<b>-</b> ₽ ×
	Signal Name	UUT Pin	Generate Test from Hardware	Direction	Analog Setup	Digital Setup	Comments	
								-

Figure 6 - Enable Debug Capture in the Digital Test Editor

Note that the Xpress Databases do not need to be manually populated before using the Debug Capture capability of the Digital Test Editor. The configuration and pinmap information for the test will be captured by the tool along with set up information and patterns.

An example Debug Capture of a test program executed through the M9-Series Interface can be seen below. This test program failed, but was difficult to debug in the original test environment due to the complexity of the test executive and limited failure reporting. After a debug capture through the Digital Test Editor, it is clear that all failures occurred on a single channel. This could point to fixture or cabling issues since all channel settings are expected.

Si										_					
	gnal Name	UUT Pin	Conne	ct	Mod		Direction		nalog Setup		Digital Setuj	p	Comme	nts	
	ad_module	Debug/load_module	Digital		Single-Ended		Bi-Directional	Analogs			Setup_0				
	ock_module	Debug/clock_module	Digital		Single-Ended		Bi-Directional	AnalogS			Setup_0				
re	set_module	Debug/reset_module Debug/s0	Digital Digital		Single-Ended Single-Ended		Bi-Directional Bi-Directional	AnalogS AnalogS			Setup_0 Setup 1				
- 51		Debug/s0 Debug/s1	Digital		Single-Ended		Bi-Directional	Analog			Setup 1				
- 54		Debug/s1	Digital		Single-Ended		Bi-Directional	Analog			Setup_1				
- y		Debug/y0	Digital		Single-Ended		Bi-Directional	Analogs			Setup 1				
ri i	· · · · · · · · · · · · · · · · · · ·	nalog Setup 🙀 Digit	· · ·			į.			• • • •	1				1	
	ern Set			s0		52					1				<b>. . .</b>
9	Type Static Ti	load_mod_clock_i MH MH	no reset_mo MH	su	s1	52	y0	y1	y2	<b>y</b> 3	y4	y5	уб	y7	Pass/Fa
, 10	TimingSe			CL.	IL	IL	IOX	IOX	IOX	(OX	OL	OL	OL	OL	Pass/Fa
110	TimingSe			а. П.	TL.	IL I	OL	OH		DH	OH	OH	OH	OH	Pass/F
12	TimingSe		1	(H	IL	IL	он	OL	ОН	DH	OH	OH	OH	OH	Pass/F
13🔀	TimingSe		1	(L	IH	IL	ОН	OH	OL	DH	OH	OH	OH	OH	Pass/Fa
14🚫	TimingSe		1	(H	IH	IL	OH	OH	OH	DL	OH	OH	OH	OH	Pass/F
	TimingSe			IL.	IL	IH	он	OH		DH	OL	OH	OH	OH	Pass/F
	TimingSe			(H	IL	IH	OH	OH		DH	OH	OL	OH	OH	Pass/F
	TimingSe			(L	IH	IH	OH OH	OH		DH	OH	OH	OL	OH	Pass/F
	TimingSe TimingSe			(H (L	IH IL	IH IL	OL	oh oh		DH DH	OH OH	OH OH	OH OH	OL	Pass/F Pass/F
	rinningse			CH	IL	IL IL	OH	OL		DH	OH	OH	OH	OH	Pass/F
	TiminaSe											OH		OH	·····
20	TimingSe TimingSe				IH	IL	OH	OH	OL	UH	UH	UH	UH		Pass/F
20 21 <mark>8</mark> 0	TimingSe TimingSe TimingSe		1	01. 0H	IH IH	IL IL	OH OH	oh oh		DH DL	OH OH	OH	OH OH	OH	·····
20 21 🔏 22 🚫	TimingSe		1	[L					он						Pass/F Pass/F Pass/F
20 218 228 238	TimingSe TimingSe		1	0L 0H	IH	IL	OH	OH	OH OH	DL	он	OH	он	OH	Pass/F

Figure 7 - Debug Captured Digital Test

Debug captured tests and Xpress Databases can be saved and used for debug of the captured test program at a later time.

### **Generating Test from Hardware**

The Digital Test Editor also has the ability to generate a test from the current state of the Di-Series hardware. This is useful to capture instrument and channel settings at a specific point in a test program, particularly if your test program does not easily allow insight to this information or to view settings that were not available on legacy instruments that are being emulated. This should be done immediately after the static pattern or digital pattern set in question, before any instrument settings are reset. To generate a test from Di-Series hardware from within the Digital Test Editor, create a new Digital Test from an iStudio project, and select "Debug -> Generate Test from Hardware" from the top menu bar.

Digital Test* - iStudio D File Edit View Ins	ert Run	ditor Debug Tools Help Enable Remote Debugging			
Signal Name	UU	Generate Test from Hardware	Mode	Direction	
				1	

Figure 8 - Generate Test from Hardware from within the Digital Test Editor

The Generate Test from Hardware dialog box requires some inputs to get the desired information from the Di-Series hardware. First, the resource string of the target virtual instrument must be provided for the empty project. If an invalid string is provided, an error will be reported in the Output window. The user can select whether to only generate a Static Pattern or Dynamic Pattern Set, or both. If Dynamic Patterns Set is selected, there are options of gathering all patterns up to the PATC Halt, or to only generate a specific pattern range to isolate a failure. There is also the option to capture the test results from the hardware.

🌠 Generate Test from H	ardware	? <mark>×</mark>
└─Virtual Instrument Selectio	n	
C Virtual Instrument	Di_0	-
Resource String	TERDI0::0,9,0-2	
☑ Static Pattern		
🔽 Dynamic Pattern Set	Incremental	
Dynamic Pattern Selecti	on	
First PATC Halt		
C Pattern Range	to	
Results		
	ОК	Cancel

Figure 9 - Generate Test from Hardware Dialog Box

An example of a test generated from Di-Series hardware is shown below. The test program failed on multiple channels, and with the use of the Digital Test Editor it is clear the relays of the failing channels are open (Connect = None) when we expected the digital relays to be closed. This type of failure is not as easy to debug programmatically.

Debug/Pin_15 Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_2 Channel 15 [0,9,0,15]   Debug/Pin_16 None Single-Ended Bi-Directional AnalogSetup_0 DigitalSetup_0 Channel 15 [0,9,0,16]   Debug/Pin_17 None Single-Ended Bi-Directional AnalogSetup_0 DigitalSetup_0 Channel 17 [0,9,0,17]   Debug/Pin_18 None Single-Ended Bi-Directional AnalogSetup_0 DigitalSetup_0 Channel 18 [0,9,0,18]   Debug/Pin_19 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_0 Channel 18 [0,9,0,18]   Debug/Pin_19 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_1 Channel 18 [0,9,0,19]   Debug/Pin_19 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_1 Channel 19 [0,9,0,19]   Debug/Pin_10 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_1 Channel 19 [0,9,0,19]   None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_1 Channel 19 [0,9,0,	Pin_15 Debug/Pin_15 Home Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_2 Channel 15 [0,9,0,15]   Pin_16 Debug/Pin_16 None Single-Ended Bi-Directional AnalogSetup_0 DigitalSetup_0 Channel 15 [0,9,0,16]   Pin_17 Debug/Pin_18 None Single-Ended Bi-Directional AnalogSetup_0 DigitalSetup_0 Channel 17 [0,9,0,17]   Pin_18 Debug/Pin_19 None Single-Ended Bi-Directional AnalogSetup_0 DigitalSetup_0 Channel 18 [0,9,0,18]   Pin_19 Debug/Pin_20 Debug/Pin_20 Channel 18 [0,9,0,18] Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_0 Channel 19 [0,9,0,19]   Pin_19 Debug/Pin_20 Debug/Pin_20 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_1 Channel 19 [0,9,0,19]   gnal Setup Workanalog Setup Biolic Setue Biolic Setue DigitalSetue_1 Channel 19 [0,9,0,19]   gnal Setup Workanalog Setup Biolic Setue Biolic Setue DigitalSetue_1 Channel 19 [0,9,0,19]	Image: Second
Debug/Pin_16 None Single-Ended Bi-Directional AnalogSetup_0 DigitalSetup_0 Channel 16 [0,9,0,16]   Debug/Pin_17 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_0 Channel 17 [0,9,0,17]   Debug/Pin_18 None Single-Ended Bi-Directional AnalogSetup_0 DigitalSetup_0 Channel 18 [0,9,0,18]   Debug/Pin_19 None Single-Ended Bi-Directional AnalogSetup_0 DigitalSetup_0 Channel 18 [0,9,0,18]   Debug/Pin_19 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_0 Channel 19 [0,9,0,19]   Debug/Pin_20 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_0 Channel 19 [0,9,0,19]   Debug/Pin_20 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_0 Channel 19 [0,9,0,19]   Analog Setup ISingle-Ended Bi-Directional AnalogSetup_0 DigitalSetup_0 Channel 19 [0,9,0,19]   Analog Setup ISingle-Ended Bi-Directional AnalogSetup_0 DigitalSetup_0 Channel	Pin_16 Debug/Pin_16 None Single-Ended Bi-Directional AnalogSetup_0 DigitalSetup_0 Channel 16 [0,9,0,16]   Pin_17 Debug/Pin_17 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_0 Channel 17 [0,9,0,17]   Pin_18 Debug/Pin_19 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_0 Channel 17 [0,9,0,17]   Pin_19 Debug/Pin_19 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_0 Channel 18 [0,9,0,18]   Pin_19 Debug/Pin_19 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_0 Channel 19 [0,9,0,19]   Pin 20 Debug/Pin_20 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_1 Channel 20 [0,9,0,19]   gal Setup Wr Analog Setup @ None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_1 Channel 20 [0,9,0,20]   gal Setup @ Type Pin_14 Pin_15 Pin_17 Pin_18 Pin_20 <	tup_0 DigitalSetup_0 Channel 16 [0,9,0,16]   tup_1 DigitalSetup_0 Channel 17 [0,9,0,17]   tup_0 DigitalSetup_0 Channel 18 [0,9,0,18]   tup_1 DigitalSetup_1 Channel 18 [0,9,0,18]   tup_1 DigitalSetup_1 Channel 19 [0,9,0,19]   tup_0 DigitalSetup_1 Channel 20 [0,9,0,20]
Debug/Pin_17 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_1 Channel 17 [0,9,0,17]   Debug/Pin_18 None Single-Ended Bi-Directional AnalogSetup_0 DigitalSetup_0 Channel 18 [0,9,0,18]   Debug/Pin_19 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_0 Channel 18 [0,9,0,18]   Debug/Pin_19 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_1 Channel 19 [0,9,0,19]   None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_1 Channel 19 [0,9,0,19]   Pubul/Pin_20 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_1 Channel 19 [0,9,0,19]   AnalogSetup ISingle-Ended Bi-Directional AnalogSetup_1 DigitalSetup_1 Channel 19 [0,9,0,19]   ************************************	Pin_17 Debug/Pin_17 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_1 Channel 17 [0,9,0,17]   Pin_18 Debug/Pin_19 None Single-Ended Bi-Directional AnalogSetup_0 DigitalSetup_1 Channel 18 [0,9,0,18]   Pin_19 Debug/Pin_19 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_1 Channel 19 [0,9,0,19]   Pin_20 Debug/Pin_20 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_1 Channel 19 [0,9,0,19]   gnal Setup Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_1 Channel 19 [0,9,0,19]   gnal Setup Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_1 Channel 19 [0,9,0,19]   gnal Setup Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_1 Channel 19 [0,9,0,19]   gnal Setup Static_Timing_0 IOX	tup_1 DigitalSetup_1 Channel 17 [0,9,0,17]   tup_0 DigitalSetup_0 Channel 18 [0,9,0,18]   tup_1 DigitalSetup_1 Channel 19 [0,9,0,19]   tup_0 DigitalSetup_1 Channel 20 [0,9,0,19]   tup_1 DigitalSetup_0 Channel 20 [0,9,0,19]   tup_0 DigitalSetup_1 Channel 20 [0,9,0,19]   tup_0 DigitalSetup_0 Channel 20 [0,9,0,19]   tup_0 DigitalSetup_1 Channel 20 [0,9,0,20]
Debug/Pin_18 None Single-Ended Bi-Directional AnalogSetup_0 DigitalSetup_0 Channel 18 [0,9,0,18]   Debug/Pin_19 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_1 Channel 18 [0,9,0,18]   Debug/Pin_20 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_1 Channel 19 [0,9,0,19]   Debug/Pin_20 None Single-Ended Bi-Directional AnalogSetup 0 DigitalSetup 1 Channel 19 [0,9,0,19]   Analog Setup Image: Setup Image: Setup 1 DigitalSetup 0 Channel 20 [0,9,0,20]   Type Pin_14 Pin_15 Pin_17 Pin_18 Pin_19 Pin_20 Pin_21 Pin_22 Pin_23 Pin_24 Pin_25 Pin_26 Pin_27   Timing_0 IOX <	Pin_18 Debug/Pin_18 None Single-Ended Bi-Directional AnalogSetup_0 DigitalSetup_0 Channel 18 [0,9,0,18]   Pin_19 Debug/Pin_19 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_1 Channel 18 [0,9,0,18]   Pin_19 Debug/Pin_20 Debug/Pin_20 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_1 Channel 19 [0,9,0,19]   Pin 20 Debug/Pin_20 Bi-Directional AnalogSetup_0 DigitalSetup_1 Channel 19 [0,9,0,19]   Pattern Set Single-Ended Bi-Directional AnalogSetup_0 DigitalSetup_1 Channel 19 [0,9,0,20]   Pattern Set Single-Ended Bi-Directional AnalogSetup_0 DigitalSetup_1 Channel 20 [0,9,0,20]   Static_Timing_0 IOX IOX IOX IOX IOX IOX   1 X TimingSet_0_1 ILOH IOX MH IOX IOX IOX IOX   2 TimingSet_0_1 MIL IOX MH IOX IOX IOX <	tup_0 DigitalSetup_0 Channel 18 [0,9,0,18] tup_1 DigitalSetup_1 Channel 19 [0,9,0,19] tup_1 DigitalSetup_1 Channel 20 [0,9,0,19] tup 0 DigitalSetup 0 Channel 20 [0,9,0,20] in_21 Pin_22 Pin_23 Pin_24 Pin_25 Pin_26 Pin_27 IOX IOX IOX IOX IOX IOX tup 1 DigitalSetup 0 Channel 20 [0,9,0,18] tup 1 DigitalSetup 1 Channel 20 [0,9,0,18] tup 1 DigitalSetup 1 Channel 20 [0,9,0,18] tup 1 DigitalSetup 1 Channel 20 [0,9,0,18] IOX IOX IOX IOX IOX IOX IOX
Debug/Pin_19 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_1 Channel 19 [0,9,0,19]   Debug/Pin_20 None Single-Ended Bi-Directional AnalogSetup_0 DigitalSetup_1 Channel 19 [0,9,0,19]   AnalogSetup Image: DigitalSetup Single-Ended Bi-Directional AnalogSetup_0 DigitalSetup_0 Channel 20 [0,9,0,20]   Yppe Pin_14 Pin_15 Pin_16 Pin_17 Pin_18 Pin_19 Pin_20 Pin_21 Pin_22 Pin_23 Pin_24 Pin_25 Pin_26 Pin_2   Timing_0 IOX <td< td=""><td>Prin_19 Debug/Pin_19 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_1 Channel 19 [0,9,0,19]   Pin 20 Debug/Pin 20 None Single-Ended Bi-Directional AnalogSetup_0 DigitalSetup_1 Channel 19 [0,9,0,19]   gnal Setup Image: Single-Ended Bi-Directional AnalogSetup_0 DigitalSetup_10 Channel 19 [0,9,0,19]   Pattern Set Prin_14 Pin_15 Pin_16 Pin_17 Pin_18 Pin_19 Pin_20 Pin_21 Pin_22 Pin_23 Pin_24 Pin_25 Pin_25 Pin_20 Static_Timing_0 IOX IOX&lt;</td><td>tup_1 DigitalSetup_1 Channel 19 [0,9,0,19] tup 0 DigitalSetup 0 Channel 20 [0,9.0.20] in_21 Pin_22 Pin_23 Pin_24 Pin_25 Pin_26 Pin_27 10X 10X 10X 10X 10X 10X 10X</td></td<>	Prin_19 Debug/Pin_19 None Single-Ended Bi-Directional AnalogSetup_1 DigitalSetup_1 Channel 19 [0,9,0,19]   Pin 20 Debug/Pin 20 None Single-Ended Bi-Directional AnalogSetup_0 DigitalSetup_1 Channel 19 [0,9,0,19]   gnal Setup Image: Single-Ended Bi-Directional AnalogSetup_0 DigitalSetup_10 Channel 19 [0,9,0,19]   Pattern Set Prin_14 Pin_15 Pin_16 Pin_17 Pin_18 Pin_19 Pin_20 Pin_21 Pin_22 Pin_23 Pin_24 Pin_25 Pin_25 Pin_20 Static_Timing_0 IOX IOX<	tup_1 DigitalSetup_1 Channel 19 [0,9,0,19] tup 0 DigitalSetup 0 Channel 20 [0,9.0.20] in_21 Pin_22 Pin_23 Pin_24 Pin_25 Pin_26 Pin_27 10X 10X 10X 10X 10X 10X 10X
Debug/Pin 20 None Sindle-Ended Bi-Directional AnalogSetup DigitalSetup Channel 20 [0.9.0.20]   rAnalog Setup ** DigitalSetup ** DigitalSe	Yen Debud/Pin 20 None Single-Ended Bi-Directional AnalogSetup DiatalSetup 0 DiatalSetup Di	in_21 Pin_22 Pin_23 Pin_24 Pin_25 Pin_26 Pin_27 IOX IOX IOX IOX IOX IOX IOX
Type Pin_14 Pin_15 Pin_16 Pin_17 Pin_18 Pin_19 Pin_20 Pin_21 Pin_23 Pin_24 Pin_25 Pin_26 Pin_27   Timing_0 IOX <td>Image: Constraint of the state of</td> <td>in_21 Pin_22 Pin_23 Pin_24 Pin_25 Pin_26 Pin_27 TOX TOX TOX TOX TOX TOX TOX C</td>	Image: Constraint of the state of	in_21 Pin_22 Pin_23 Pin_24 Pin_25 Pin_26 Pin_27 TOX TOX TOX TOX TOX TOX TOX C
Type Pin_14 Pin_15 Pin_16 Pin_17 Pin_18 Pin_19 Pin_20 Pin_21 Pin_22 Pin_23 Pin_24 Pin_25 Pin_26 Pin_26 Pin_27   Timing_0 IOX	Statur Set Type Pin_14 Pin_15 Pin_16 Pin_17 Pin_18 Pin_19 Pin_20 Pin_21 Pin_22 Pin_23 Pin_24 Pin_25 Pin_25 Pin_23 Pin_24 Pin_25 Pin_25 Pin_26 Pin_26 Pin_26 Pin_26 Pin_26 Pin_27 Pin_26 Pin_24 Pin_25 Pin_25 Pin_26 Pin_2	
Get_0_1 ILOH IDX MH IDX MI IDX ID	ImingSet_0_1 ILOH IOX MH IOX MI IOX <t< th=""><th>Image: state state</th></t<>	Image: state
Set_0_1 ML IOX	2 (No. 1) ML IOX	
Set 0 1 IOX IOX IOX IOX IOX OX	4 TimingSet_0_1 10X 10X 10X 10X 10X 10X	
Set_0_1 IOX		
	tput	
	• •	

Figure 10 - Example test generated from hardware

With this new information, it can be investigated whether the relay state was programmed incorrectly, another program is opening the relay during the test execution, or an Alarm event caused the relay to open.

### Loop and Sweep

The Loop and Sweep features of the Digital Test Editor can be utilized for debugging tests originally created using the Digital Test Editor, or those that were Debug Captured or generated from hardware. Looping and Sweeping can be used to pinpoint causes for unexpected failures without the potential overhead of the original test executive.

The Digital Test Editor provides the ability to indefinitely loop a test program and to stop on a failure, without the use of a separate test executive such as TestStudio. Looping tests can help to reproduce an intermittent failure, which may be more difficult or time consuming if the original test program has significant overhead. To use this feature from within a digital test, select "Tools -> Loop" from the top menu bar.

	よ <b>10   10</b>   × % % % -	nsert Run Debug (ເດ ດາ ູ 🏾 🗝 🗤	Tools Help Import Export	• • • • • • • • • • • • • • • • • • •			
-~-Sig	nal Setup Signal Name	UUT Pin	Sweep	Mode	Direction	Analog Setup	
	load module	Debug/load module	Configuration	5ingle-Ended	Bi-Directional	AnalogSetup 0	Digita
	- clock module	Debug/clock module	Pinmap	5ingle-Ended	Bi-Directional	AnalogSetup 0	Digit
	reset_module	Debug/reset_module	Customize	Fingle-Ended	Bi-Directional	AnalogSetup_0	Digit
	s0	Debug/s0		5ingle-Ended	Bi-Directional	AnalogSetup_0	Digit
	s1	Debug/s1	Options	5ingle-Ended	Bi-Directional	AnalogSetup_0	Digit
		D-hl-o	Digital	Single-Ended	Bi-Directional	AnalogSetup_0	Digita
	s2	Debug/s2					

Figure 11 - Launching the Loop Dialog box from the Digital Test Editor

The Loop Dialog box allows the user to begin looping the test by selecting the Loop button and to stop at any time by selecting the Stop button. A tally of how many times the test passed and failed is displayed, as well as the number of total loops (including the one in progress), Not Tested results, and Error results. The option also exists to stop on a failure, allowing the ability to take any physical measurements on the test setup at the time of a failure. Select Cancel to exit the Loop dialog box.



Figure 12 - Loop Dialog box stopped on failure

The Digital Test Editor also provides the ability to execute the digital test by "sweeping" over a range of either threshold voltage values or timing set values for one specific channel. This allows the user to debug at what particular voltage threshold or time applied to a channel a test will begin to pass or fail, without repeatedly modifying and rebuilding a test program. Note that in order for the "Sweep" test results to provide valuable information, the test must first be passing or have only the one channel of concern reporting a failure. This is to ensure that the modified threshold voltage or timing values are causing the new pass or fail result. This debug tool may be required if physical access to pins to measure voltage or timing of a signal is not possible. To use this feature from within the Digital Test Editor, select "Tools -> Sweep" from the top menu bar.

		isent Run. Debug ທີ່ຕີ - ຈາ	Tools Help Import Export Loop	• • • • • • • • • • • • • • • • • • •	të ↓ë ke në •		
	Signal Name	UUT Pin	Sweep	Mode	Direction	Analog Set	
	- load module	Debug/load_module	Configuration	5ingle-Ended	Bi-Directional	AnalogSetup 0	
ſ	- clock_module	Debug/clock_module	Pinmap	5ingle-Ended	Bi-Directional	AnalogSetup_0	
	reset_module	Debug/reset_module	Customize	5ingle-Ended	Bi-Directional	AnalogSetup_0	
	s0	Debug/s0		5ingle-Ended	Bi-Directional	AnalogSetup_0	
	s1	Debug/s1	Options	5ingle-Ended	Bi-Directional	AnalogSetup_0	
	s2	Debug/s2	Digital	Single-Ended	Bi-Directional	AnalogSetup_0	
	y0	Debug/y0	Digital	Single-Ended	Bi-Directional	AnalogSetup_1	
_	gnal Setup 🛷 An Pattern Set 🔪	alog Setup 🔒 Digital	Setup 🔀 Pattern Ti	ming			
Inde	ex Type	load_mod_clock_mo	o reset_mo si	) 51	s2 y0	y1 y2	
	0 Static Ti	MH ML	MH		1		

Figure 13 - Launching the Sweep Dialog box from the Digital Test Editor

From the Sweep Dialog box, the user must select from a drop-down menu which signal from the Digital Test pinmap to perform the sweep function, and whether to perform a sweep of a threshold voltage or timing values for one of the timing sets available for the selected signal. When Voltage is selected, a varying binary threshold is set for the chosen signal. When Time is selected, the capture mode is set to Strobe, and the strobe is swept over the specified values. The example below has chosen to sweep across threshold voltages of 200mV up to 4V in 100mV increments for a specified signal.

🎇 Sweep				• <b>×</b>
Signal	y6	•	Voltage	
	Time			
Pattern Type		Ŧ		
VTH	2V		200 mV 500 mV 800 mV 1.1 V 1.4 V 1.7 V 2.1 V 2.3 V 2.6 V 2.9 V 3.2 V 3.5 V	3.8 V
Increment	100 mV		Time	
Start Value	200 mV			
End Value	4∨			
			VTH 2.9 V [PASS: 6 FAIL: 0] VTH 3V [PASS: 0 FAIL: 6] VTH 3V [PASS: 0 FAIL: 6] VTH 31 V [PASS: 0 FAIL: 6]	*
			Run Stop 0	Close

Figure 14 - Example Sweep Dialog box and results

The results show that when the threshold is set to less than 3.0V, the test passes. Therefore, we can conclude that the input to the signal is around 3.0V when it is testing for a signal high. Now, the user must determine if that is the expected input to the channel, and, if it is not, what could be causing the incorrect input. If the channel of concern were not accessible due to hardware cabling or fixture design, this tool is an effective way to verify what the Di-Series channel is sensing without modifying and recompiling a test program.

## **Conclusions**

The standard Di-Series software package includes numerous tools to aid in debugging unexpected test program results that involve the Di-Series instrument, whether executing native Di-Series tests or using the instrument in emulation for legacy programs. Some of these tools, including the Di-Series Alarm Viewer and the automatically generated log files, require no user setup or maintenance and are a good place to quickly start the debugging process. Others require some user setup, but they can save significant time when compared to debugging solely from an existing test executive.