

# DAISY Device Test Guide

Revision 1.1

2022. 11. 08

CRZ Technology

<http://www.crz-tech.com/>

# Document History

Revision	Date	Change note
1.0	2021.01.28	Initial Version
1.1	2022.11.08	Fixed 100G test setup

Index

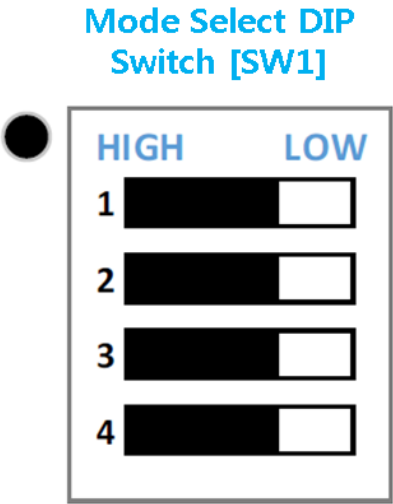
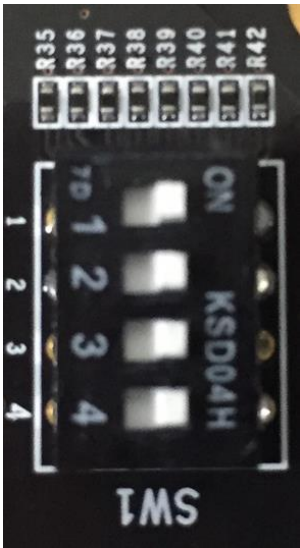
- 1. Overview ..... 4
- 2. Test Setup ..... 5
- 3. NVMe M.2 ..... 6
- 4. DDR4 DIMM ..... 10
- 5. PCIe x16 Endpoint..... 12
  - 5.1. PCIe Host BIOS Setup ..... 12
  - 5.2. Verifying the PCIe link..... 12
  - 5.3. XDMA test ..... 18
- 6. QSFP28 - 100GB Ethernet..... 20

## **1. Overview**

This document explains how to test the devices(NVMe M.2, DDR4 DIMM, PCIe x16 endpoint, QSFP28) mounted on DAISY.

## 2. Test Setup

Set mode select dip switch[SW1] to [JTAG / QSPI / SD] mode accordingly.



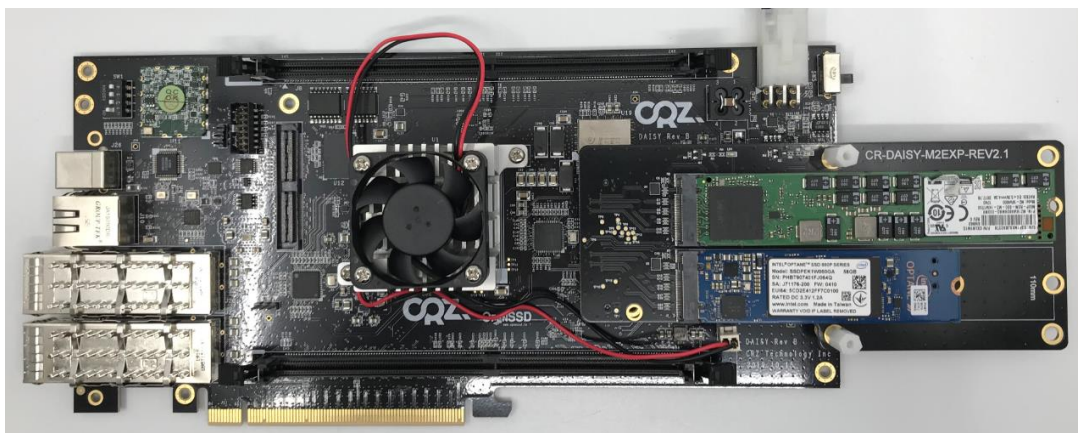
MODE	Switch			
	[4]	[3]	[2]	[1]
JTAG	LOW[ON]	LOW[ON]	LOW[ON]	LOW[ON]
QSPI 32	LOW[ON]	LOW[ON]	HIGH[OFF]	LOW[ON]
SD1	HIGH[OFF]	HIGH[OFF]	HIGH[OFF]	LOW[ON]

### 3. NVMe M.2

Samsung SM963 NVMe M.2 SSD 480GB MLC has been verified on DAISY.

Insert NVMe M.2 SSD to CR-DAISY-M2EXP1-REV2.1 board.

Connect CR-DAISY-M2EXP1-REV2.1 board to DAISY through J25.



Create bootable image for SD boot or QSPI boot by referring to [DAISY Petalinux Porting Guide](#) document.

Check if LED0 is turned on during boot. LED0 is on if PCIe link is correctly configured.

Verify PCIe link after logging in linux.

```
root@daisy:~# lspci
0000:00:00.0 PCI bridge: Xilinx Corporation Device 9134
0000:01:00.0 Non-Volatile memory controller: Samsung Electronics Co Ltd NVMe SSD
Controller SM961/PM961
0001:00:00.0 PCI bridge: Xilinx Corporation Device 9134
root@daisy:~#
```

Check if NVMe SSD is configured as block device.

```
root@daisy:~# lsblk
NAME                MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
mtdblock0           31:0    0    36M  0 disk
mtdblock1           31:1    0     1M  0 disk
mtdblock2           31:2    0    68M  0 disk
mtdblock3           31:3    0    11M  0 disk
mmcblk0             179:0    0   14.9G  0 disk
|-mmcblk0p1          179:1    0     1G  0 part /run/media/mmcblk0p1
`--mmcblk0p2          179:2    0    6.5G  0 part /run/media/mmcblk0p2
nvme0n1             259:0    0  447.1G  0 disk
root@daisy:~#
```

Create disk partition.

```
root@daisy:~# fdisk /dev/nvme0n1

Welcome to fdisk (util-linux 2.32.1).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0xcb728903.

Command (m for help): n
Partition type
   p   primary (0 primary, 0 extended, 4 free)
   e   extended (container for logical partitions)
Select (default p): p
Partition number (1-4, default 1):
First sector (2048-937703087, default 2048):
Last sector, +sectors or +size{K,M,G,T,P} (2048-937703087, default 937703087):

Created a new partition 1 of type 'Linux' and of size 447.1 GiB.

Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
[ 1883.709806] nvme0n1: p1
Syncing disks.
root@daisy:~#
```

## DAISY Device Test Guide

Check new partition name.

```
root@daisy:~# lsblk
NAME                MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
mtdblock0           31:0    0    36M  0 disk
mtdblock1           31:1    0     1M  0 disk
mtdblock2           31:2    0    68M  0 disk
mtdblock3           31:3    0    11M  0 disk
mmcblk0             179:0    0   14.9G  0 disk
|-mmcblk0p1         179:1    0     1G  0 part /run/media/mmcblk0p1
`--mmcblk0p2        179:2    0    6.5G  0 part /run/media/mmcblk0p2
nvme0n1             259:0    0  447.1G  0 disk
`--nvme0n1p1        259:1    0  447.1G  0 part
root@daisy:~#
```

Make file system on new partition.

```
root@daisy:~# mkfs -t ext2 /dev/nvme0n1p1
mke2fs 1.44.3 (10-July-2018)
Discarding device blocks: done
Creating filesystem with 117212630 4k blocks and 29310976 inodes
Filesystem UUID: 15a30903-8a05-4beb-9971-21cf586f1caf
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
    4096000, 7962624, 11239424, 20480000, 23887872, 71663616, 78675968,
    102400000

Allocating group tables: done
Writing inode tables: done
Writing superblocks and filesystem accounting information: done


root@daisy:~#
```

Create a directory and mount SSD to the directory.

```
root@daisy:~# mkdir /media/nvme
root@daisy:~# mount /dev/nvme0n1p1 /media/nvme
root@daisy:~# cd /media/nvme
root@daisy:/media/nvme# vi test.txt
```



Create a test file and check if the file is preserved during power cycling.



The screenshot shows a Tera Term VT window titled "COM14:115200baud - Tera Term VT". The menu bar includes "File", "Edit", "Setup", "Control", "Window", and "Help". The main window area is black with a vertical cursor on the left. At the bottom, a status bar displays "test.txt 1/1 100%".

```
root@daisy:/media/nvme# ls -al
total 24
drwxr-xr-x 3 root root 4096 Nov 28 09:23 .
drwxr-xr-x 3 root root 60 Nov 28 09:20 ..
drwx----- 2 root root 16384 Nov 28 09:18 lost+found
-rw-r--r-- 1 root root 21 Nov 28 09:23 test.txt
root@daisy:/media/nvme#
```

## 4. DDR4 DIMM

8GB x8 DDR4\_M393A1K43BB0-CRC / 16GB x4 DDR4\_M393A2K40BB1-CRC have been verified on DAISY.

Connect 8GB x8 DDR4\_M393A1K43BB0-CRC RDIMM to J8 and 16GB x4 DDR4\_M393A2K40BB1-CRC RDIMM to J9.

Set SW1 to JTAG mode.

Connect USB cable with host PC.

Connect 12V DC power adaptor.

Turn on board power by sliding power switch.

Open [dual\\_mig\\_x8.zip](#) Vivado project.

Vivado version 2019.1 should be used.

In the Address Editor tab, 8GB RDIMM is mapped to 0x1000000000 and 16GB RDIMM is mapped to 0x1800000000.

Cell	Slave Interface	Base Name	Offset Address	Range	High Address
zynq_ultra_ps_e_0					
Data (40 address bits : 0x00A0000000 [ 256M ], 0x0400000000 [ 4G ], 0x1000000000 [ 224G ])					
ddr4_0	C0_DDR4_S_AXI	C0_DDR4_ADDRESS_BLOCK	0x10_0000_0000	8G	0x11_FFFF_FFFF
ddr4_0	C0_DDR4_S_AXI_CTRL	C0_REG	0x00_A000_0000	4K	0x00_A000_0FFF
ddr4_1	C0_DDR4_S_AXI	C0_DDR4_ADDRESS_BLOCK	0x18_0000_0000	16G	0x1B_FFFF_FFFF
ddr4_1	C0_DDR4_S_AXI_CTRL	C0_REG	0x00_A000_1000	4K	0x00_A000_1FFF

Select "Open Hardware Manager" under "PROGRAM AND DEBUG" in "Flow Navigator".

Click "Program device".

Verify if LED0 and LED1 are turned on after FPGA is programmed.

Two LEDs must be turned on if MIG calibration is done correctly.

Run serial terminal to view debug log.

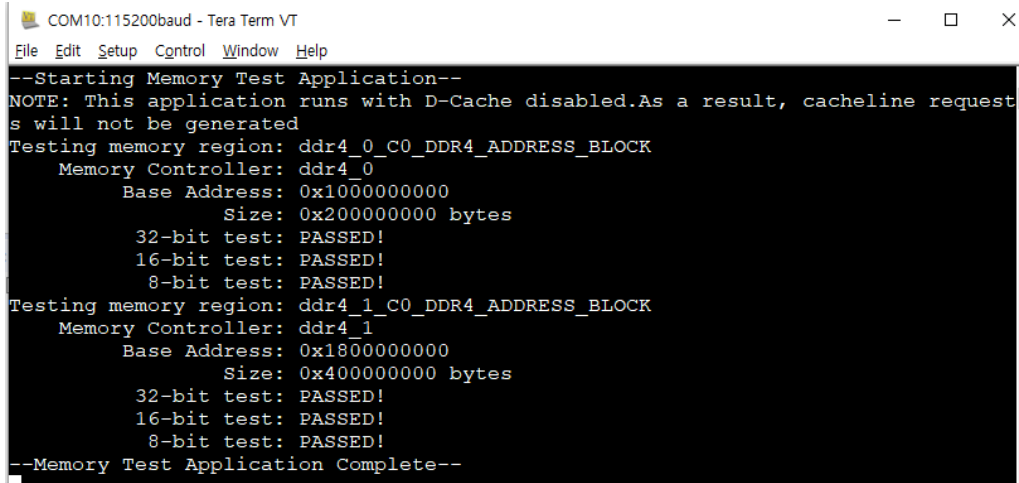
Launch SDK through "File" -> "Launch SDK" to check if RDIMM can be accessed.

Select "Run" -> "Debug History" -> "System Debugger on Local" on SDK menu.

Verify if two LEDs are turned on after FPGA is programmed.

Click Cortex-A53 #0 and press "F8" to execute the program.

Check if memory test passes.



```
COM10:115200baud - Tera Term VT
File Edit Setup Control Window Help
--Starting Memory Test Application--
NOTE: This application runs with D-Cache disabled.As a result, cacheline requests will not be generated
Testing memory region: ddr4_0_C0_DDR4_ADDRESS_BLOCK
  Memory Controller: ddr4_0
    Base Address: 0x100000000
    Size: 0x200000000 bytes
    32-bit test: PASSED!
    16-bit test: PASSED!
    8-bit test: PASSED!
Testing memory region: ddr4_1_C0_DDR4_ADDRESS_BLOCK
  Memory Controller: ddr4_1
    Base Address: 0x180000000
    Size: 0x400000000 bytes
    32-bit test: PASSED!
    16-bit test: PASSED!
    8-bit test: PASSED!
--Memory Test Application Complete--
```

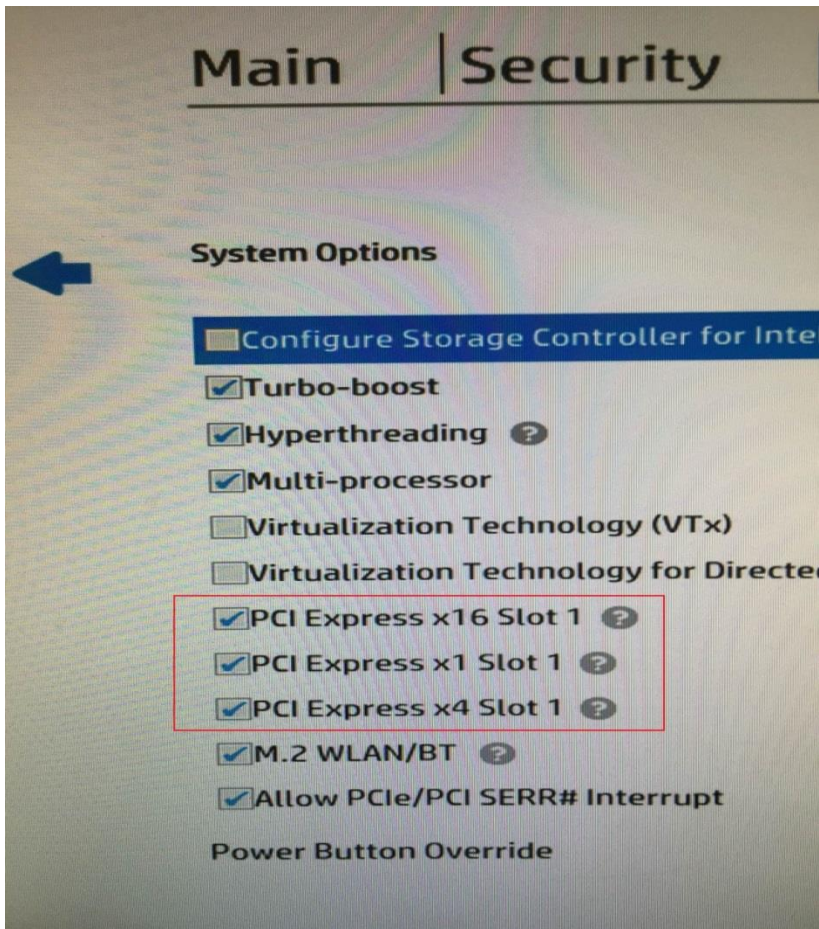
## 5. PCIe x16 Endpoint

During the test, set the Mode Select DIP Switch[SW1] to JTAG mode.

### 5.1. PCIe Host BIOS Setup

PCIe slot must be enabled in PCIe Host BIOS setup.

The host used for verification is HP PRODESK, and the PCIe slot must be activated in the BIOS as shown below.



### 5.2. Verifying the PCIe link

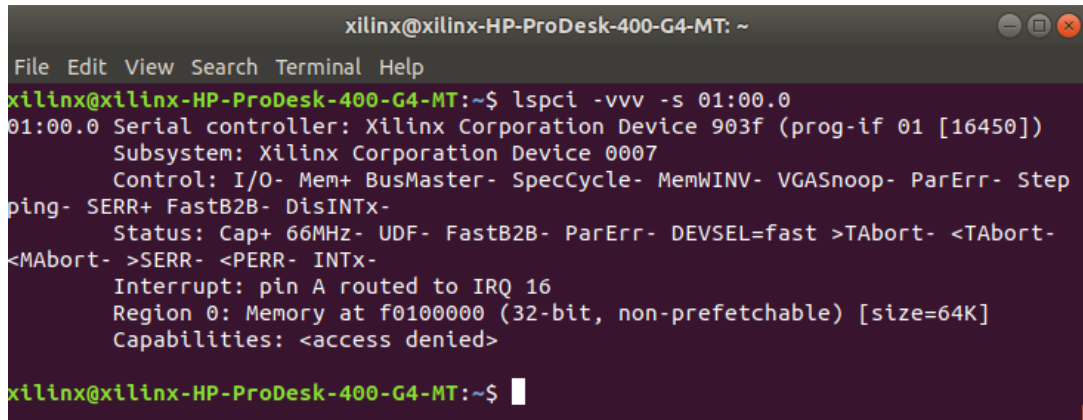
After attaching the PCIe extension cable to the Daisy board, plug it into the PCIe x16 slot of PRODESK (it is the top black slot among the three PCIe slots).

Apply 12V power to the board.

Open the provided Vivado project ([pcie\\_ep.zip](#)) and download the bitstream.  
Vivado 2019.1 version must be used.

Power on PRODESK and log in to Linux.  
At this time, check if LED0 of the board is on.

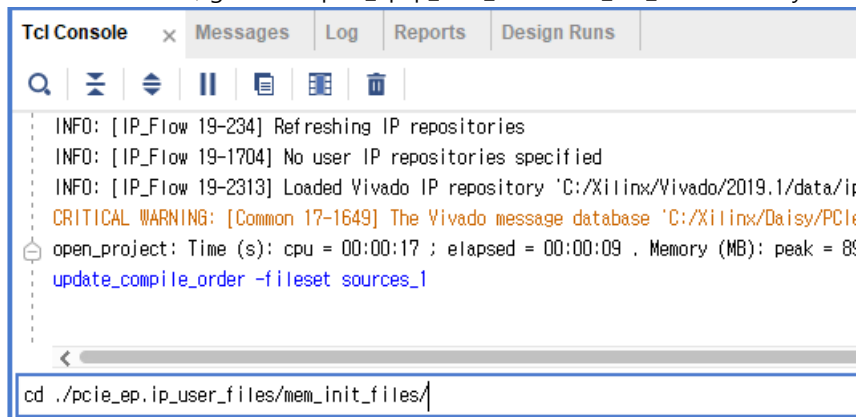
After logging in to Host Linux, run the command below from the command line to check if the link is established.



```
xilinx@xilinx-HP-ProDesk-400-G4-MT: ~
File Edit View Search Terminal Help
xilinx@xilinx-HP-ProDesk-400-G4-MT:~$ lspci -vvv -s 01:00.0
01:00.0 Serial controller: Xilinx Corporation Device 903f (prog-if 01 [16450])
    Subsystem: Xilinx Corporation Device 0007
    Control: I/O- Mem+ BusMaster- SpecCycle- MemWINV- VGASnoop- ParErr- Step
ping- SERR+ FastB2B- DisINTx-
    Status: Cap+ 66MHz- UDF- FastB2B- ParErr- DEVSEL=fast >TAbort- <TAbort-
<MAbort- >SERR- <PERR- INTx-
    Interrupt: pin A routed to IRQ 16
    Region 0: Memory at f0100000 (32-bit, non-prefetchable) [size=64K]
    Capabilities: <access denied>
xilinx@xilinx-HP-ProDesk-400-G4-MT:~$
```

The following is how to verify the PCIe link on the vivado tool. It must be executed with the board and JTAG connected.

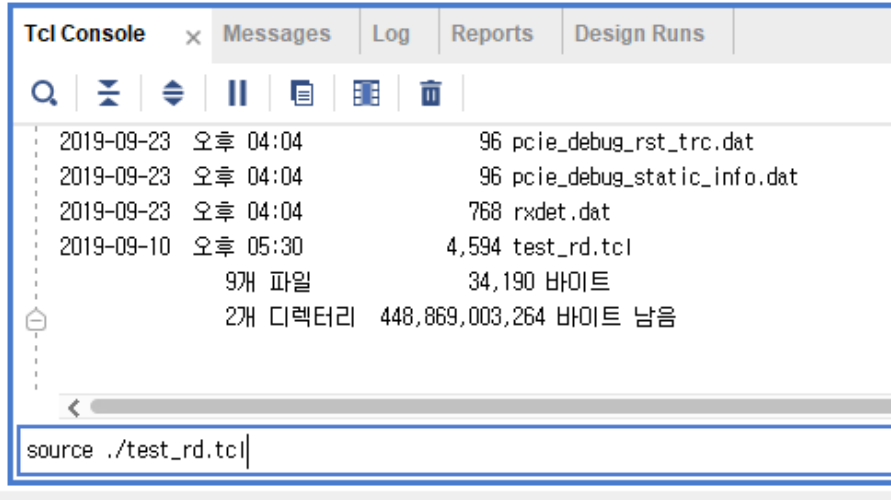
In the Tcl Console, go to the pcie\_ep.ip\_user\_files/mem\_init\_files directory.



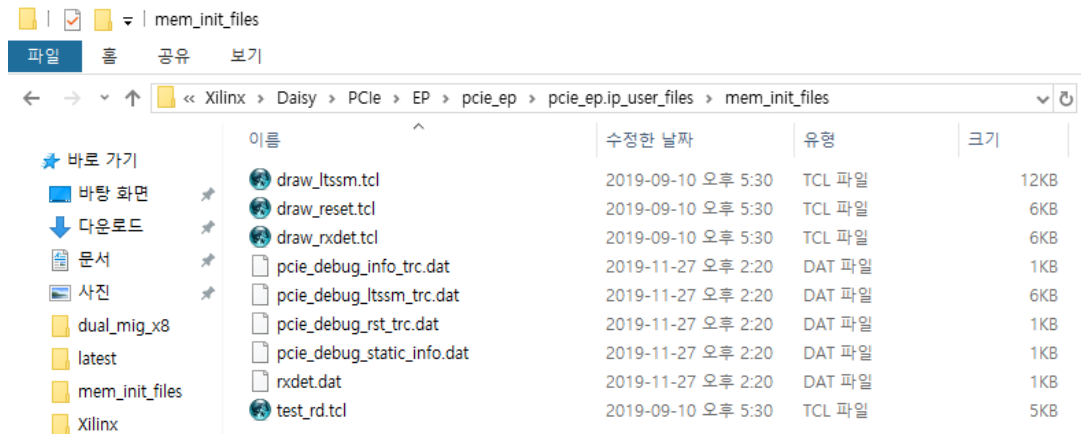
```
Tcl Console x Messages Log Reports Design Runs
[Icons: Search, Zoom, Run, Stop, Copy, Paste, Delete]
INFO: [IP_Flow 19-234] Refreshing IP repositories
INFO: [IP_Flow 19-1704] No user IP repositories specified
INFO: [IP_Flow 19-2313] Loaded Vivado IP repository 'C:/Xilinx/Vivado/2019.1/data/ip
CRITICAL WARNING: [Common 17-1649] The Vivado message database 'C:/Xilinx/Daisy/PCIE
open_project: Time (s): cpu = 00:00:17 ; elapsed = 00:00:09 . Memory (MB): peak = 89
update_compile_order -fileset sources_1
[Scrollbar]
cd ./pcie_ep.ip_user_files/mem_init_files/
```

## DAISY Device Test Guide

Run test\_rd.tcl.

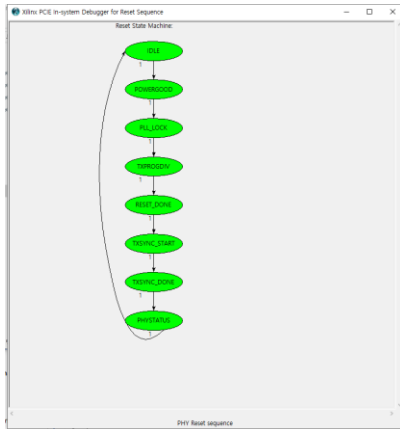


You can see that five dat files such as pcie\_debug\_info\_trc.dat, pcie\_debug\_ltssm\_trc.dat, pcie\_debug\_rst\_trc.dat, pcie\_debug\_static\_info.dat and rxdet.dat have been newly created.



Double-click `draw_reset.tcl`, `draw_rxdet.tcl` and `draw_ltssm.tcl` in Windows Explorer to check if it is normal.

draw\_reset.tcl – When the PHY is reset, it can be confirmed that it works normally.

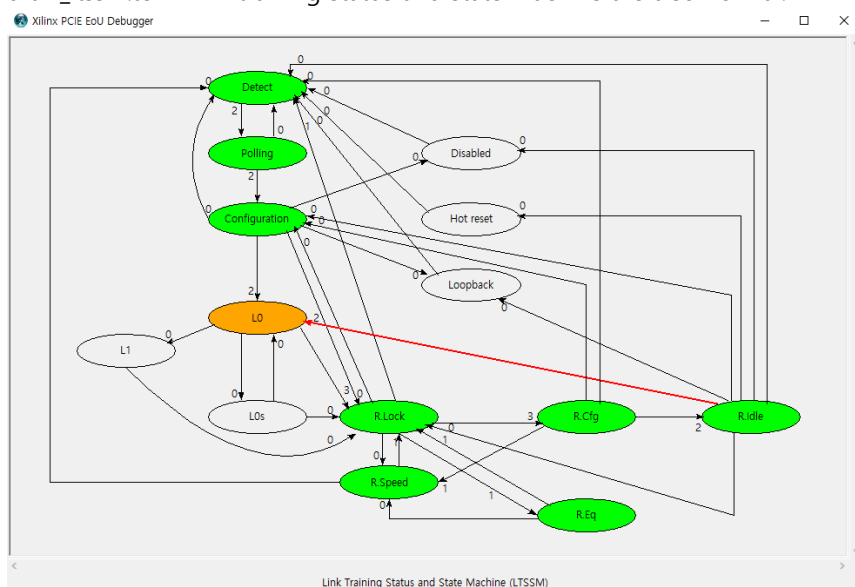


draw\_rxdet.tcl-You can see that all 16 lanes are fine.



## DAISY Device Test Guide

draw\_ltssm.tcl – Link training status and state machine are also normal.




In the Serial I/O Links tab, click Create Scan...


Name	TX	TX Pre-Cursor	TX Post-Cursor	TX Diff Swing	DPE Enabled
Ungrouped Link					
Link Group 0					
Link 0	MGT_X0Y0/RX	User Value	User Value	User Value	User Value
Link 1	MGT_X0Y1/RX	User Value	User Value	User Value	User Value
Link 10	MGT_X0Y10/TX MGT_X0Y10/RX	User Value	User Value	User Value	User Value
Link 11	MGT_X0Y11/TX MGT_X0Y11/RX	User Value	User Value	User Value	User Value
Link 12	MGT_X0Y12/TX MGT_X0Y12/RX	User Value	User Value	User Value	User Value
Link 13	MGT_X0Y13/TX MGT_X0Y13/RX	User Value	User Value	User Value	User Value
Link 14	MGT_X0Y14/TX MGT_X0Y14/RX	User Value	User Value	User Value	User Value
Link 15	MGT_X0Y15/TX MGT_X0Y15/RX	User Value	User Value	User Value	User Value




Add Link0 to Link15 and check if there is any abnormality in the eye diagram.



Create Scan



Set the description and other properties to create and optionally run a scan on the selected link.



Link: 

Link 0 (MGT\_X0Y0/TX, MGT\_X0Y0/RX)

Description: 

Scan 0

Scan Properties

Scan type: 

2D Full Eyescan

Horizontal increment: 

8

Horizontal range: 

-0.500 UI to 0.500 UI

Vertical increment: 

8

Vertical range: 

100%

Dwell


☒ BER: 

1e-5

☐ Time: 

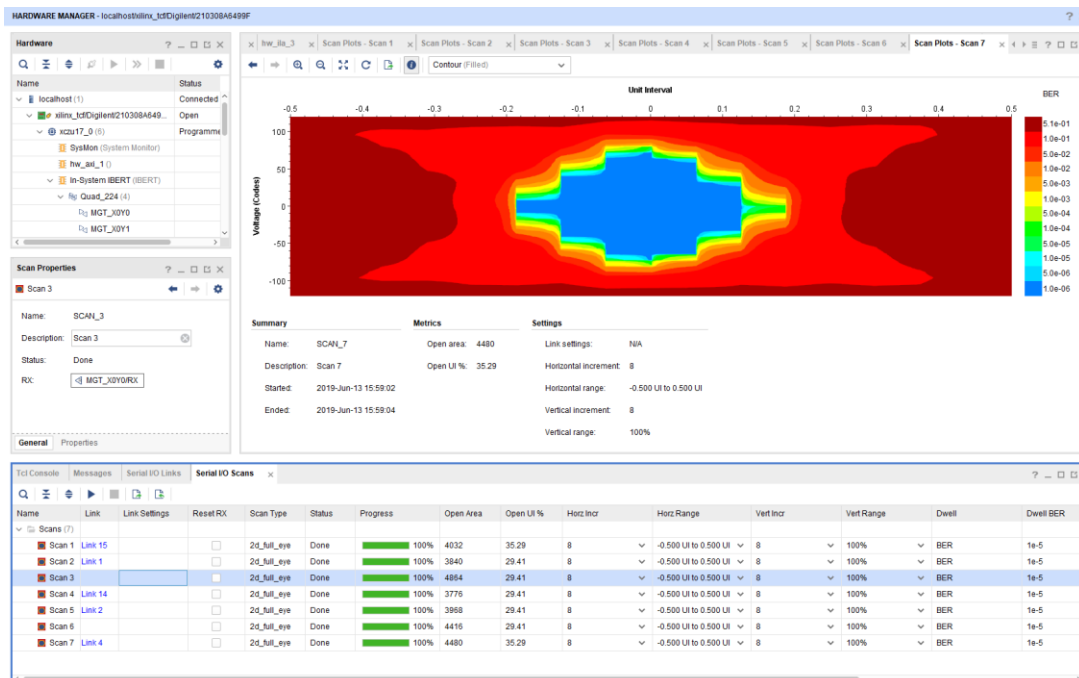
0

☒ Run scan



OK

Cancel



### 5.3. XDMA test

Access the following link and copy the necessary files to Ubuntu 16.04 Host.

[https://github.com/Xilinx/dma\\_ip\\_drivers](https://github.com/Xilinx/dma_ip_drivers)

Run the following command to build the XDMA driver kernel module and application program.

```
$ cd XDMA/linux-kernel
```

```
$ cd xdma
```

```
$ make install
```

```
$ cd tools
```

```
$ make
```

```
$ cd tests
```

Load the XDMA kernel module.

```
$ sudo ./load_driver.sh
```

Test whether XDMA is operating normally.

```
$ ./run_test.sh
```

```
xilinx@xilinx:~/Downloads/dma_ip_drivers-master/XDMA/linux-kernel/tests$ sudo ./run_test.sh
Info: Number of enabled h2c channels = 1
Info: Number of enabled c2h channels = 1
Info: The PCIe DMA core is memory mapped.
Info: Running PCIe DMA memory mapped write read test
      transfer size: 1024
      transfer count: 1
Info: Writing to h2c channel 0 at address offset 0.
Info: Wait for current transactions to complete.
** Average BW = 1024, 17.526144
Info: Writing to h2c channel 0 at address offset 1024.
Info: Wait for current transactions to complete.
** Average BW = 1024, 11.151161
Info: Writing to h2c channel 0 at address offset 2048.
Info: Wait for current transactions to complete.
** Average BW = 1024, 13.890962
Info: Writing to h2c channel 0 at address offset 3072.
Info: Wait for current transactions to complete.
** Average BW = 1024, 16.115833
Info: Reading from c2h channel 0 at address offset 0.
Info: Wait for the current transactions to complete.
** Average BW = 1024, 2.690623
Info: Reading from c2h channel 0 at address offset 1024.
Info: Wait for the current transactions to complete.
** Average BW = 1024, 4.612342
Info: Reading from c2h channel 0 at address offset 2048.
Info: Wait for the current transactions to complete.
** Average BW = 1024, 4.601089
Info: Reading from c2h channel 0 at address offset 3072.
Info: Wait for the current transactions to complete.
** Average BW = 1024, 4.605227
Info: Checking data integrity.
Info: Data check passed for address range 0 - 1024.
Info: Data check passed for address range 1024 - 2048.
Info: Data check passed for address range 2048 - 3072.
Info: Data check passed for address range 3072 - 4096.
Info: All PCIe DMA memory mapped tests passed.
Info: All tests in run_tests.sh passed.
xilinx@xilinx:~/Downloads/dma_ip_drivers-master/XDMA/linux-kernel/tests$
```

## 6. QSFP28 - 100GB Ethernet

Write [daisy\\_202001\\_100g\\_20220610\\_image.tgz](#) to the microSD card.

Boot two DAISY boards with microSD card and connect the QSFP28#1 channels to each other with 100GB 1m copper cable.

Set ip and mtu size in eth1 on board 1.

```
root@daisy:~# ifconfig eth1 down
root@daisy:~# ifconfig eth1 mtu 8192
root@daisy:~# ifconfig eth1 192.168.2.1 up
```

Set ip and mtu size in eth1 on board 2.

```
root@daisy:~# ifconfig eth1 down
root@daisy:~# ifconfig eth1 mtu 8192
root@daisy:~# ifconfig eth1 192.168.2.2 up
```

On boards 1 and 2, ping to see if the other party's ip is connected.

```
root@daisy:~# ping 192.168.2.1
PING 192.168.2.1 (192.168.2.1): 56 data bytes
64 bytes from 192.168.2.1: seq=0 ttl=64 time=0.167 ms
64 bytes from 192.168.2.1: seq=1 ttl=64 time=0.079 ms
64 bytes from 192.168.2.1: seq=2 ttl=64 time=0.071 ms
64 bytes from 192.168.2.1: seq=3 ttl=64 time=0.067 ms
64 bytes from 192.168.2.1: seq=4 ttl=64 time=0.073 ms
64 bytes from 192.168.2.1: seq=5 ttl=64 time=0.188 ms
^C
--- 192.168.2.1 ping statistics ---
6 packets transmitted, 6 packets received, 0% packet loss
round-trip min/avg/max = 0.067/0.107/0.188 ms
root@daisy:~#
```

```

root@daisy:~# ping 192.168.2.2
PING 192.168.2.2 (192.168.2.2): 56 data bytes
64 bytes from 192.168.2.2: seq=0 ttl=64 time=0.154 ms
64 bytes from 192.168.2.2: seq=1 ttl=64 time=0.205 ms
64 bytes from 192.168.2.2: seq=2 ttl=64 time=0.107 ms
64 bytes from 192.168.2.2: seq=3 ttl=64 time=0.148 ms
^C
--- 192.168.2.2 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 0.107/0.153/0.205 ms
root@daisy:~#

```

Set board 1 as an iperf client and board 2 as an iperf server to measure performance.

```

[ 3541.731054] CMAC going to reset
[ 3541.734289] CMAC RX alignment ...:d8
[ 3541.737863] CMAC leaving reset
Connecting to host 192.168.2.4, port 5201
[ 5] local 192.168.2.3 port 50502 connected to 192.168.2.4 port 5201
[ ID] Interval          Transfer      Bitrate      Retr  Cwnd
[ 5] 0.00-1.00 sec      521 MBytes  4.37 Gbits/sec    0   366 KBytes
[ 5] 1.00-2.00 sec      521 MBytes  4.37 Gbits/sec    0   390 KBytes
[ 5] 2.00-3.00 sec      521 MBytes  4.37 Gbits/sec    0   390 KBytes
[ 5] 3.00-4.00 sec      521 MBytes  4.37 Gbits/sec    0   390 KBytes
[ 5] 4.00-5.00 sec      521 MBytes  4.37 Gbits/sec    0   413 KBytes
[ 5] 5.00-6.00 sec      521 MBytes  4.37 Gbits/sec    0   723 KBytes
[ 5] 6.00-7.00 sec      521 MBytes  4.37 Gbits/sec    0   723 KBytes
[ 5] 7.00-8.00 sec      520 MBytes  4.37 Gbits/sec    0   723 KBytes
[ 5] 8.00-9.00 sec      521 MBytes  4.37 Gbits/sec    0   723 KBytes
[ 5] 9.00-10.00 sec     521 MBytes  4.37 Gbits/sec    0   723 KBytes
- - - - -
[ ID] Interval          Transfer      Bitrate      Retr
[ 5] 0.00-10.00 sec     5.09 GBytes  4.37 Gbits/sec    0
[ 5] 0.00-10.00 sec     5.09 GBytes  4.37 Gbits/sec
iperf Done.

```

```
[ 3564.067944] CMAC going to reset
[ 3564.071185] CMAC RX alignment ...:c0
[ 3564.074766] CMAC leaving reset
-----
Server listening on 5201
-----
Accepted connection from 192.168.2.3, port 50500
[ 5] local 192.168.2.4 port 5201 connected to 192.168.2.3 port 50502
[ ID] Interval      Transfer    Bitrate
[ 5]  0.00-1.00    sec    521 MBytes  4.37 Gbits/sec
[ 5]  1.00-2.00    sec    521 MBytes  4.37 Gbits/sec
[ 5]  2.00-3.00    sec    521 MBytes  4.37 Gbits/sec
[ 5]  3.00-4.00    sec    522 MBytes  4.37 Gbits/sec
[ 5]  4.00-5.00    sec    521 MBytes  4.37 Gbits/sec
[ 5]  5.00-6.00    sec    521 MBytes  4.37 Gbits/sec
[ 5]  6.00-7.00    sec    521 MBytes  4.37 Gbits/sec
[ 5]  7.00-8.00    sec    521 MBytes  4.37 Gbits/sec
[ 5]  8.00-9.00    sec    521 MBytes  4.37 Gbits/sec
[ 5]  9.00-10.00   sec    521 MBytes  4.37 Gbits/sec
[ 5] 10.00-10.00   sec    393 KBytes  4.22 Gbits/sec
-----
[ ID] Interval      Transfer    Bitrate
[ 5]  0.00-10.00   sec    5.09 GBytes  4.37 Gbits/sec
-----
receiver
```

When measured by iperf, the performance of about 4.37Gbps was confirmed.