

PCIE GEN3 x16

INSTALLATION GUIDE

DE10-Pro

for Intel® FPGA University Program



CONTENTS

Chapter 1	DE10-Pro Development Environment.....	2
1.1	About the Guide	2
1.2	Instruction	2
1.3	Block Diagram	2
1.4	Getting Help	3
Chapter 2	Run the PCIe Gen3 x16 Demo	4
2.1	Setup the DE10-Pro board to the host PC	4
2.2	Bring up the DE10-Pro.....	5

Chapter 1

DE10-Pro Development Environment

1.1 About the Guide

This guide introduces how to setup the DE10-Pro development kit onto the Host computer to support running the PCIe Gen3 x 16 demo. From this demonstration, it will show how the PC Linux and FPGA communicate with each other through the PCI Express interface. Avalon-MM Intel Stratix 10 Hard IP+ for PCI Express IP is used in this demonstration.

1.2 Instruction

The following items are required to set up PCIe Gen3 x 16 demo for DE10-Pro board:

- 64-bit Linux (Ubuntu 16.04 x long-term support (LTS) or CentOS 7.6) installed
- Intel Quartus Prime Pro Edition **19.1.0** installed, **license is required**

In this guide, we will use Ubuntu 16.04 long-term support (LTS), 64-bit version with kernel 4.8 for installation and introduction.

1.3 Block Diagram

Figure 1-1 shows the block diagram in the FPGA system. In the DMA PCIe design, the On-Chip memory and external DMA Controller are used for performing DMA testing.

The On-Chip memory, and External DMA Controller are connected to the PCI Express Hard IP controller through the Memory-Mapped Interface.

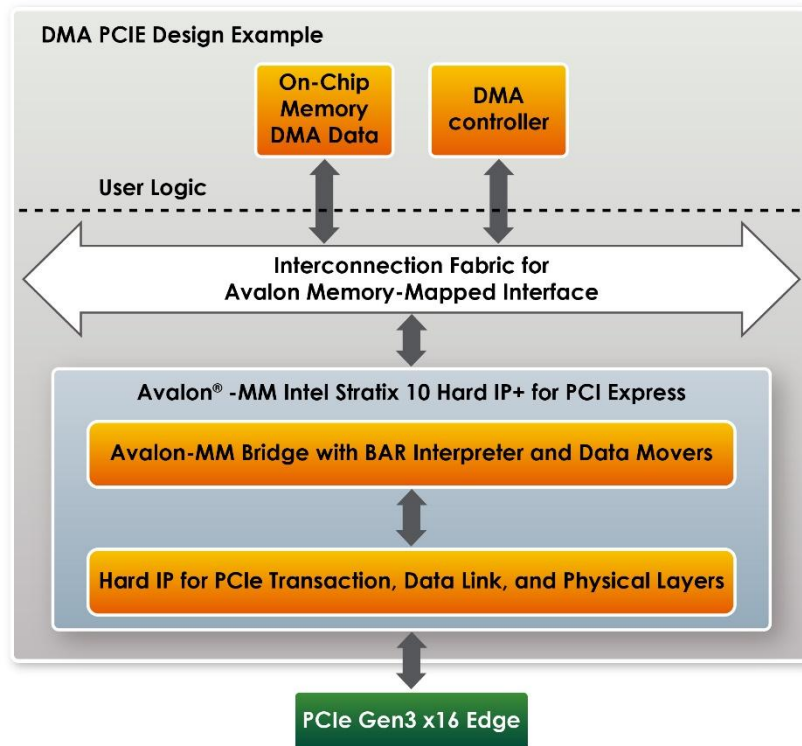


Figure 1-1 Hardware block diagram of the PCIe Gen3 x16 reference design

1.4 Getting Help

Here are the addresses where you can get help if you encounter any problem:

- Terasic Inc.
- 9F., No.176, Sec.2, Gongdao 5th Rd, East Dist, Hsinchu City, 30070. Taiwan
- Email: support@terasic.com
- Tel.: +886-3-5750-880
- Web: <http://DE10-Pro.terasic.com>

Run the PCIe Gen3 x16 Demo

This chapter introduces the installation of the DE10-Pro board for running the PCIe Gen3 x16 demo, following the below steps to set up the demonstration.

2.1 Setup the DE10-Pro board to the host PC

Follow the steps below to setup the DE10-Pro board as **Figure 2-1**:

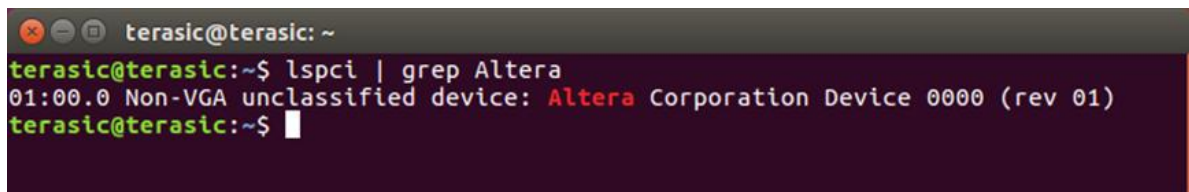
1. Make sure the host PC is powered off.
2. Insert the DE10-Pro board into the PCIe X16 socket of the host PC.
3. Connect PC's 12V PCI Express 8-pin power source to the DE10-Pro (recommended).
4. Connect PC's USB port to DE10-Pro micro UB II port(J8) using an USB cable.



Figure 2-1 DE10-Pro setup

2.2 Bring up the DE10-Pro

1. Power on the host PC and open the Terminal.
2. Copy and unzip the `pcie_gen3_x16_demo` compressed file to the host PC (unzip it to Desktop as an example).
3. Enter the command “`cd`” to switch to the path:
“`pcie_gen3_x16_demo/PCie_gen3x16_gh2e2/demo_batch`”.
4. Create an environment variable **QUARTUS_ROOTDIR** by pointing to the Quartus installation path. Typing the “`export`” commands in terminal.
Take the path: `/home/ubuntu/intelFPGA_pro/19.1/quartus` as an example:
“`export QUARTUS_ROOTDIR=/home/ubuntu/intelFPGA_pro/19.1/quartus`”
4. Execute “`sudo -E sh test.sh`” command to configure the FPGA.
5. Restart Linux.
6. Make sure the FPGA development kit can be detected by Linux. Using the command: “`lspci | grep Altera`” for the board detected as **Figure 2-2**.

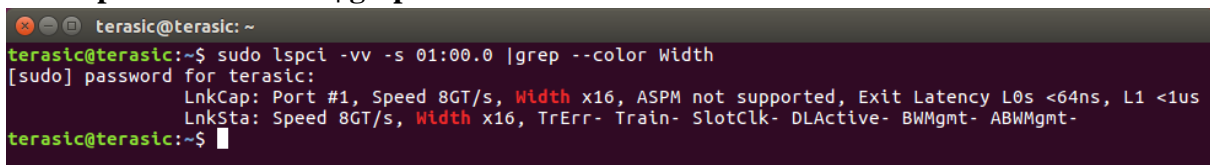


```
terasic@terasic: ~  
terasic@terasic:~$ lspci | grep Altera  
01:00.0 Non-VGA unclassified device: Altera Corporation Device 0000 (rev 01)  
terasic@terasic:~$
```

Figure 2-2 Board detected information

7. Verify the PCIe link information by typing command as **Figure 2-3**:

`sudo lspci -vv -s 01:00.0 | grep --color Width`



```
terasic@terasic: ~  
terasic@terasic:~$ sudo lspci -vv -s 01:00.0 | grep --color Width  
[sudo] password for terasic:  
LnkCap: Port #1, Speed 8GT/s, Width x16, ASPM not supported, Exit Latency L0s <64ns, L1 <1us  
LnkSta: Speed 8GT/s, Width x16, TrErr- Train- SlotClk- DLActive- BWMgmt- ABWMgmt-  
terasic@terasic:~$
```

Figure 2-3 PCIe link information

8. Open a terminal, use the command “`cd`” to switch to the path: “**`linux_software/kernel/linux`**”
9. Install the driver by entering the commands below as **Figure 2-4**:

`chmod 777 install load unload`

`sudo ./install`

```

terasic@terasic: ~/Desktop/pcie_gen3_x16_demo/linux_software/kernel/linux
terasic@terasic:~/Desktop/pcie_gen3_x16_demo/linux_software/kernel/linux$ chmod 777 install load unload
terasic@terasic:~/Desktop/pcie_gen3_x16_demo/linux_software/kernel/linux$ sudo ./install
[sudo] password for terasic:
make: Warning: File 'Makefile' has modification time 16733 s in the future
make -C /lib/modules/4.10.0-42-generic/build M=/home/terasic/Desktop/pcie_gen3_x16_demo/linux_software/kernel/linux clean
make[1]: Entering directory '/usr/src/linux-headers-4.10.0-42-generic'
make[2]: Warning: File '/home/terasic/Desktop/pcie_gen3_x16_demo/linux_software/kernel/linux/Makefile' has modification time 16731 s in the future
make[2]: warning: Clock skew detected. Your build may be incomplete.
make[1]: Leaving directory '/usr/src/linux-headers-4.10.0-42-generic'
make: warning: Clock skew detected. Your build may be incomplete.
make: Warning: File 'Makefile' has modification time 16731 s in the future
make -C /lib/modules/4.10.0-42-generic/build M=/home/terasic/Desktop/pcie_gen3_x16_demo/linux_software/kernel/linux modules
make[1]: Entering directory '/usr/src/linux-headers-4.10.0-42-generic'
make[2]: Warning: File '/home/terasic/Desktop/pcie_gen3_x16_demo/linux_software/kernel/linux/Makefile' has modification time 16730 s in the future
CC [M] /home/terasic/Desktop/pcie_gen3_x16_demo/linux_software/kernel/linux/intel_fpga_pcie_chr.o
CC [M] /home/terasic/Desktop/pcie_gen3_x16_demo/linux_software/kernel/linux/intel_fpga_pcie_dma.o
CC [M] /home/terasic/Desktop/pcie_gen3_x16_demo/linux_software/kernel/linux/intel_fpga_pcie_setup.o
CC [M] /home/terasic/Desktop/pcie_gen3_x16_demo/linux_software/kernel/linux/intel_fpga_pcie_ioctl.o
LD [M] /home/terasic/Desktop/pcie_gen3_x16_demo/linux_software/kernel/linux/intel_fpga_pcie_drv.o
make[2]: warning: Clock skew detected. Your build may be incomplete.
Building modules, stage 2.
make[2]: Warning: File '/home/terasic/Desktop/pcie_gen3_x16_demo/linux_software/kernel/linux/Makefile' has modification time 16726 s in the future
MODPOST 1 modules
CC /home/terasic/Desktop/pcie_gen3_x16_demo/linux_software/kernel/linux/intel_fpga_pcie_drv.mod.o
LD [M] /home/terasic/Desktop/pcie_gen3_x16_demo/linux_software/kernel/linux/intel_fpga_pcie_drv.ko
make[2]: warning: Clock skew detected. Your build may be incomplete.
make[1]: Leaving directory '/usr/src/linux-headers-4.10.0-42-generic'
make: warning: Clock skew detected. Your build may be incomplete.
terasic@terasic:~/Desktop/pcie_gen3_x16_demo/linux_software/kernel/linux$

```

Figure 2-4 Driver installation

10. Verify the driver installation:

lsmod | grep intel_fpga_pcie_drv

Expected result as [Figure 2-5](#):

intel_fpga_pcie_drv 24576 0

```

terasic@terasic: ~/Desktop/pcie_gen3_x16_demo/linux_software/kernel/linux
terasic@terasic:~/Desktop/pcie_gen3_x16_demo/linux_software/kernel/linux$ lsmod | grep intel_fpga_pcie_drv
intel_fpga_pcie_drv      24576  0
terasic@terasic:~/Desktop/pcie_gen3_x16_demo/linux_software/kernel/linux$

```

Figure 2-5 Driver information

11. Verify that Linux recognizes the PCIe* design example:

lspci -d 1172:000 -v | grep intel_fpga_pcie_drv

Note: If you have changed the Vendor ID, substitute the new Vendor ID for Intel® 's Vendor ID in this command.

Expected result as [Figure 2-6](#):

Kernel driver in use: intel_fpga_pcie_drv

```
terasic@terasic: ~/Desktop/pcie_gen3_x16_demo/linux_software/kernel/linux
terasic@terasic:~/Desktop/pcie_gen3_x16_demo/linux_software/kernel/linux$ lspci
-d 1172:0000 -v | grep intel_fpga_pcie_drv
      Kernel driver in use: intel_fpga_pcie_drv
terasic@terasic:~/Desktop/pcie_gen3_x16_demo/linux_software/kernel/linux$
```

Figure 2-6 Kernel Driver information

12. Use the command “cd” to switch the path to “linux_software/user/example”.

13. Compile the design example application as [Figure 2-7](#):

make

```
terasic@terasic: ~/Desktop/pcie_gen3_x16_demo/linux_software/user/example
terasic@terasic:~/Desktop/pcie_gen3_x16_demo/linux_software/user/example$ make
make: Warning: File 'Makefile' has modification time 16506 s in the future
g++ -std=c++0x -Wall -I ../api -I ../api/linux ../api/linux/intel_fpga_pcie_api_
linux.cpp intel_fpga_pcie_link_test.cpp -o intel_fpga_pcie_link_test
make: warning: Clock skew detected. Your build may be incomplete.
terasic@terasic:~/Desktop/pcie_gen3_x16_demo/linux_software/user/example$
```

Figure 2-7 Compile the example

14. You can run the Intel® FPGA IP PCIe* link test in manual or automatic mode.

- In automatic mode, the application automatically selects the device. The test selects the Intel® Stratix® 10 PCIe* device with the lowest BDF by matching the Vendor ID. The test also selects the lowest available BAR.
- In manual mode, the test queries you for the bus, device, and function number and BAR.

For the Intel® Stratix® 10-GX Development Kit, you can determine the BDF by typing the following command:

lspci -d 1172:0000

15. Run the test as [Figure 2-8](#):

sudo ./intel_fpga_pcie_link_test

```
terasic@terasic: ~/Desktop/pcie_gen3_x16_demo/linux_software/user/example
terasic@terasic:~/Desktop/pcie_gen3_x16_demo/linux_software/user/example$ sudo .
/intel_fpga_pcie_link_test
[sudo] password for terasic:

*****
Intel FPGA PCIe Link Test
Version 2.0
0: Automatically select a device
1: Manually select a device
*****
>
```

Figure 2-8 Test Result

16. Type “1” followed by the ENTER key to select **manual mode**, and Type the **bus(1)**, **device(0)**, **function (0)** and **BAR(2)** number, as [Figure 2-9](#).

```
terasic@terasic: ~/Desktop/pcie_gen3_x16_demo/linux_software/user/example
terasic@terasic:~/Desktop/pcie_gen3_x16_demo/linux_software/user/example$ sudo ./intel_fpga_pcie_link_test

*****
Intel FPGA PCIe Link Test
Version 2.0
0: Automatically select a device
1: Manually select a device
*****
> 1
Enter bus number, in hex:
> 1
Enter device number, in hex:
> 0
Enter function number, in hex:
> 0
BDF is 0x100
B:D.F, in hex, is 1:0.0
Enter BAR number (-1 for none):
> 2
Opened a handle to BAR 0x2 of a device with BDF 0x100

*****
0: Link test - 100 writes and reads
1: Write memory space
2: Read memory space
3: Write configuration space
4: Read configuration space
5: Change BAR
6: Change device
7: Enable SRIOV
8: Do a link test for every enabled virtual function
   belonging to the current device
9: Perform DMA
10: Quit program
*****
> 
```

Figure 2-9 PCIe Link Test

17. Type “9” followed by the ENTER key to select **Perform DMA** test
18. Type “0” to select **Run DMA** test. And enter “0” for infinite loop as [Figure 2-10](#):

```

terasasic@terasasic: ~/Desktop/pcie_gen3_x16_demo/linux_software/user/example
*****
0: Link test - 100 writes and reads
1: Write memory space
2: Read memory space
3: Write configuration space
4: Read configuration space
5: Change BAR
6: Change device
7: Enable SRIOV
8: Do a link test for every enabled virtual function
   belonging to the current device
9: Perform DMA
10: Quit program
*****
> 9

*****
Current DMA configurations
  Run Read  (card->system) ? 1
  Run Write (system->card) ? 1
  Run Simultaneous         ? 1
  Number of dwords/desc    : 2048
  Number of descriptors     : 128
  Total length of transfer  : 1024 KiB
*****
0: Run DMA
1: Toggle read DMA
2: Toggle write DMA
3: Toggle simultaneous DMA
4: Set the number of dwords per descriptor
5: Set the number of descriptors per DMA
6: Return to main menu
*****
> 0
Enter the number of DMA operations to initiate; enter 0 for infinite loop:
> 0

```

Figure 2-10 DMA Test

19. DMA Result as [Figure 2-11](#):

```

terasasic@terasasic: ~/Desktop/pcie_gen3_x16_demo/linux_software/user/example
*****
Current DMA configurations
  Run Read  (card->system) ? 1
  Run Write (system->card) ? 1
  Run Simultaneous         ? 1
  Number of dwords/desc    : 2048
  Number of descriptors     : 128
  Total length of transfer  : 1e+03 KiB

Current run #: 2737
Current time : Thu Jul 25 17:48:24 2019

DMA throughputs, in GB/s (10^9B/s)
  Current Read Throughput   : 11.52
  Average Read Throughput   : 11.39
  Current Write Throughput  : 6.90
  Average Write Throughput  : 6.85
  Current Simul Throughput  : 12.63
  Average Simul Throughput  : 13.20
*****

```

Figure 2-11 DMA Result

Appendix

Intel Avalon -MM Intel Stratix 10 Hard IP+ for PCI Express Solutions User Guide is used as a reference:

<https://www.intel.com/content/www/us/en/programmable/documentation/sox1520633403002.html>

Additional Information

Revision History

Date	Version	Change Log
2019.08	V1.0	Initial Version (Preliminary)

Copyright Statement

Copyright © 2019 Terasic Inc. All rights reserved.