ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE)
Vol 4, Issue 5, May 2017

Implementation and Testing of PCI Express IP Core using SpartaN 6 FPGA

[1] Megha B. Kumbar, [2] Sukesh Rao M.
[1][2]Department of E&C,
N.M.A.M. Institute of Technology, Nitte, Karkala, Udupi, Karnataka, India
[1]kumbarmegha1993@gmail.com, [2]sukesh@nitte.edu.in

***Abstract*- The motherboards nowadays come with PCI Express (PCIe) expansion slots in order to communicate with the external world. The data which is going to be received by the system is the radar data which will be in the form of Serial Front Panel Data Port (Serial FPDP) frame format. But the processors of the system can only understand the information in the form of PCI Express frame format. Thus the data has to get converted into PCI Express frame format. Therefore PCI Express IP core is used for the formation of PCI Express frame format. Once the data in the form Serial FPDP frame is received then data will be extracted and will be stored in buffer memory. Then the PCI Express IP core receives the data from memory and frames the data into PCI Express frame format and will be sent over PCI Express expansion bus to the processor. Thus in order to understand whether the PCI Express IP core works properly or not a loopback test needs to be done. Thus the PCI Express IP core is implemented on Spartan 6 FPGA along with the Bus Master DMA. In this paper the implementation and loopback testing of PCI Express IP core is mentioned.**

***Keywords:*---PCI Express frame format, PCI Express IP core, loopback test, Spartan 6 FPGA, Bus Master DMA**

I. INTRODUCTION

The PCI Express (PCIe) is the third generation high performance I/O bus used to interconnect peripheral devices in applications such as computing and communication platforms. The ISA, EISA, VESA, Micro Channel buses are the first generation buses and the PCI, AGP and PCI-X are the second generation buses. It has applications in mobile, desktop, workstation, server, embedded computing and communication platforms [1]. Today's processors come with PCI Express interface, which is a point-to-point connection, i.e., it connects only two devices, no other device can share this connection. The PCI Express protocol is a layered architecture where each layer adds its own information to the original data; frames it and sends it over PCI Express link to the destination device. This protocol is in the form of IP core in Xilinx. Thus this IP core needs to be tested in order to check whether the protocol works properly or not. This is done by doing a loopback test. The loopback test is the one where the transmitted information is looped back to itself in order to receive the same data as transmitted one.

II. BASIC CONCEPT

The data which is going to be received by the system is the radar data which will be in the form of Serial Front Panel Data Port (Serial FPDP) frame format. But the processors of the system can only understand the information in the form of PCI Express frame format. Thus the data has to get converted into PCI Express frame format. The conversion can be done by implementing both PCI Express IP core and Serial FPDP core on FPGA. As shown in the Fig. 1, the radar data in the form of Serial FPDP frame format will be received by Serial FPDP core. Then the conversion happens by extracting only the data from SFPDP frame structure and storing it in buffer memory and then data will be framed into PCI Express frame structure by using PCI Express IP core.

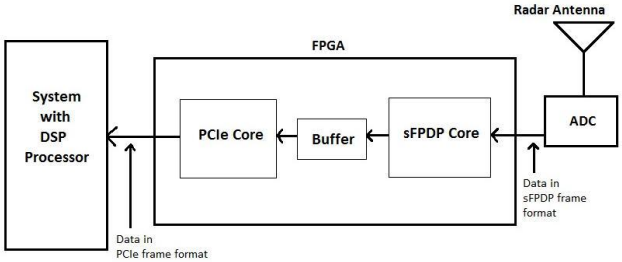

Fig. 1: Block diagram for Serial FPDP to PCIe conversion

Therefore the whole idea of conversion initially needs to test both PCI Express IP core and Serial FPDP core separately before implementing both on same FPGA for conversion. Therefore in this paper implementing and testing of only PCI Express IP core is shown.

III. PCI EXPRESS PROTOCOL

PCI Express is a high speed, low voltage, differential serial pathway for two devices to communicate with each other. It uses a protocol that allows devices to

ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE)
Vol 4, Issue 5, May 2017

communicate simultaneously by implementing dual unidirectional paths between two devices as shown in Fig.2.

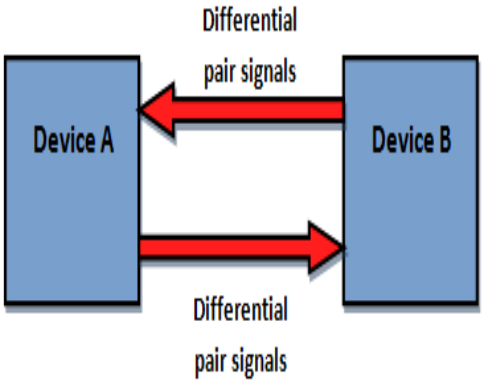

Fig. 2: Dual Unidirectional Path Concept

PCI Express is a serial, point-to-point interface i.e., devices no longer need to arbitrate for the right to be the bus driver prior to sending out a transaction. A PCI Express device is always the driver for its transmission pair(s) and is always the target for its receiver pair(s). The difference from parallel busses such as PCI; is the transmit pair of one device be the receiver pair for the other device. They must be point-to-point, one device to a second device [2]. TX of one is RX of the other and vice versa as shown in Fig. 3.

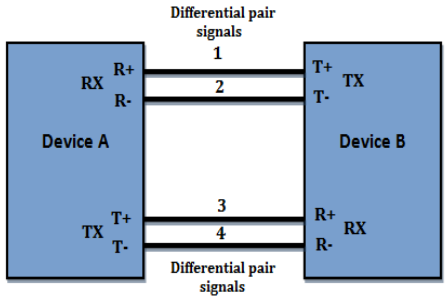

Fig. 3: Point-to-Point Connection between Two PCI Express Devices

3.1 Links, Lanes and Ports

A link is the connection between two PCI Express devices. A link consists of a number of lanes. Lane is a single set of differential transmits and receive pairs which can be seen in Fig. 3. A lane contains four signals, a differential pair for unidirectional transmission in both directions (dual unidirectional). The link shown in Fig. 4 is a four lane wide. The collection of transmission and receiver pairs that are associated with a link at the device is referred to as a port. Currently PCI Express defines the following configuration of serial links: x1, x2, x4, x8, x12, x16, and x32. This helps in scaling the bandwidth [2].

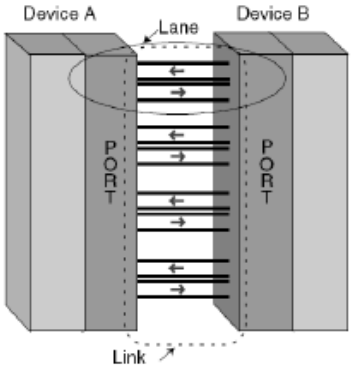

Fig. 4: Links, Lanes and Ports [3]

3.2 PCI Express Protocol Build Layers

The transformation of information between two PCI Express devices is based on the transaction between them. It uses a split transaction protocol. That is it has two transaction phases; request and completion. The transaction initiator is the requester and the intended target for request is the completer. The requester sends out the request packet. The completer sends back a completion packet to the requester. The transaction between two devices is built by three layers. They are the Transaction layer, the Data Link layer and the Physical layer as shown in Fig. 5.

The Transaction layer receives request or completion data from device core and turns it into PCI Express transactions. The Data Link layer is to ensure that the transactions are received properly across the link.

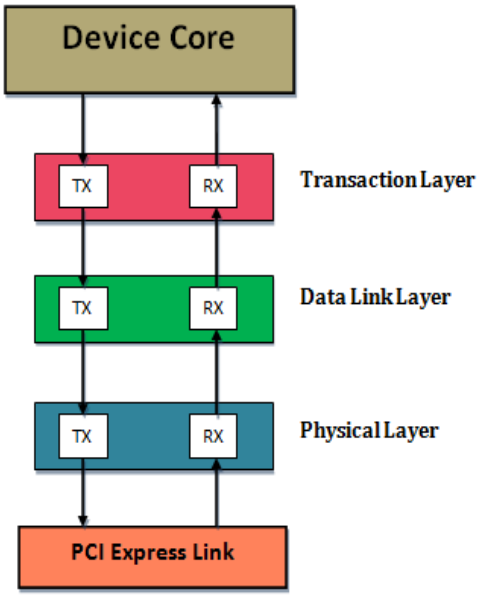

Fig. 5: The three architectural build layers

ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE)
Vol 4, Issue 5, May 2017

The Physical layer is for actual transmitting and receiving of data across the PCI Express link [2].

3.3 Packet formation:

The three architectural layers build the packets. As shown in Fig. 6, at the time of transmission the Transaction Layer adds a header and an optional ECRC (end-to-end CRC) to the data payload. The Data Link Layer adds the sequence number and LCRC (link CRC). The Physical Layer frames it for proper transmission to the other device.

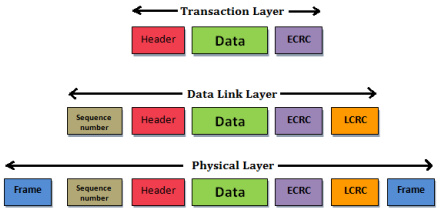

Fig. 6: Transaction buildup through architectural layers

At the time of reception the Physical Layer decodes the framing and passes along the sequence number, header, data, ECRC, and LCRC to its Data Link Layer. The Data Link Layer checks out the sequence number and LCRC, and then passes the header, data, and ECRC on to the Transaction Layer. The Transaction Layer decodes the header and passes the appropriate data on to its device core [3].

IV. LOOPBACK TESTING OF PCI EXPRESS IP CORE

The loopback testing means the data which is sent by the communication device is returned back to itself to determine whether the device under test is working properly or not. The PCI Express core is checked by doing a loop back test. This is done by making communication between the PCI Express core and the system. This communication takes place in PCI Express topological order as shown in Fig. 7. It includes a Root Complex, a PCI Express switch device, multiple Endpoint block for PCI Express and a PCI Express to PCI/PCI-X bridge.

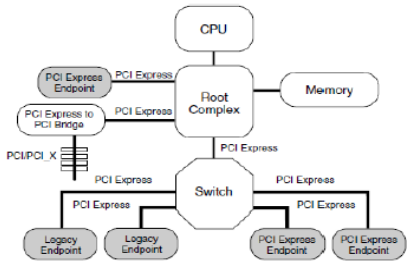

Fig. 7: PCI Express Fabric Topology [3]

A Root Complex (RC) is the root of an I/O hierarchy. It may have one or more PCI Express ports. It connects the CPU/memory subsystem to the I/O device. The Root complex may connect to Endpoint device directly or through a switch[4]. In this communication, the system is considered as Root Complex and the hardware connected to the PCI Express slot of the motherboard is considered as PCI Express Endpoint device. Here the PCI Express Endpoint device is the SP605 Evaluation kit where the implementation of PCI Express IP core is done on the Spartan 6 FPGA. This PCI Express IP core contains Bus Master DMA which helps in lower CPU utilization and large transfer of data. The communication between the system and the PCI Express core is checked by write and read operations.

During the write operation, the data is written by the Root Complex device to the Endpoint device and during read operation the data is read by the Root complex device from the Endpoint device i.e., the data is transferred from Endpoint device to the Root Complex device. This is a loop back test where the data read by the Root Complex device is same as the data which was written by it to the Endpoint device during the write operation.

The Table 1, shows the hardware and software tool required for the loopback testing.

Table 1: Hardware and the tool used

Hardware used:	Spartan 6 FPGA SP605 Evaluation kit (Fig. 8)
Tool used:	Xilinx ISE Design Suite version 14.4

A xapp1052.zip file is used which is provided by the Xilinx. This contains top-level ISE project directory named dma_performance_demo. This directory contains the original BMD design files [5].

In order to have the communication between the Root complex and the Endpoint devices certain steps need to be taken. These are as follows:

- Making the kit as a PCI Express Endpoint device.
- Connecting the kit to the system and driver installation for the same.
- Performing read and write operation between Root complex and PCI Express Endpoint device

These steps are explained in detail in the following sections.

ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE)
Vol 4, Issue 5, May 2017

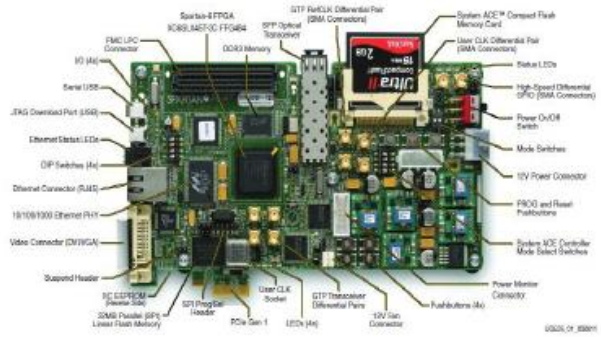

Fig. 8: SP605 Evaluation Board [6]

4.1 Step 1: PCI Express Endpoint IP core generation

In order to make the SP605 evaluation kit as a PCI Express Endpoint device first step is to generate a PCI Express Endpoint IP core. The steps for generating the IP Core Using ISE Design Suite:

- Start the CORE Generator tool and create a new project. Next selecting appropriate device xc6slx45t and family as Spartan6 as shown in Fig. 9.

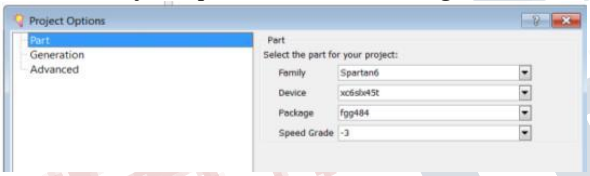

Fig. 9: Selecting proper FPGA device

- Customizing and generating the "Spartan6 Integrated Endpoint Block for PCI Express" for the targeted device from the taxonomy tree as shown in Fig.10.

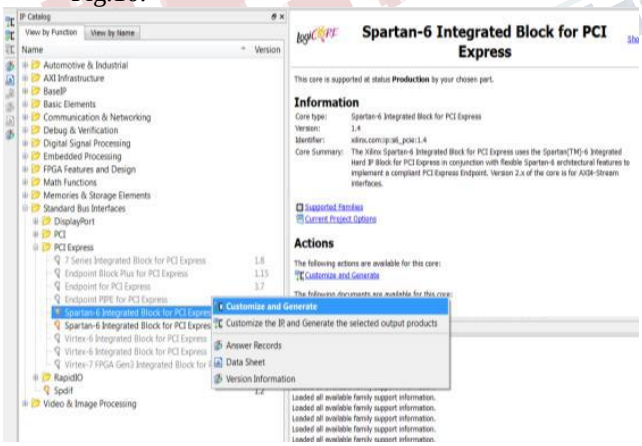

Fig. 10: Customizing the PCI Express IP core

- In the customization GUI giving the name for the core to be generated and appropriate vendor ID and host ID.
- Since it is a Gen1 x1 PCI Express, specifying the link width as 1 lane and maximum speed of data transmission is 2.5GT/s as shown in Fig. 11.

Fig. 11: PCI Express link width and speed

- The required target board is the SP605 board which needs to be selected the option and then generate the core.

4.2 Step 2: Implementing the Bus Master Design and generating bit file

Once the IP core is generated merge the BMD design files along with IP core files to generate the bit file. The steps are as follows.

- To implement Bus Master Design inside PCI Express Endpoint device, the BMD files which are located in the dma_performance_demo folder are merged with the generated IP core files.
- Then to build the design, implement_dma.pl file is used and is done by executing a command "xilperl implement_dma.pl" in the ISE Design Suite command prompt as shown in Fig. 12.

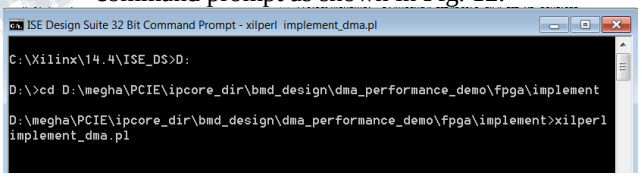

Fig. 12: Executing the command "xilperl implement_dma.pl"

- The perl script will present a series a prompts requesting user input as shown in Fig. 13. Based on the user input, the script will grab the necessary files to synthesize and build the design.
- In the first prompt, selecting the type board to which the design is targeted i.e., PCI Express development board→SP605 is a default development platform. In the next prompt select the core which is generated. In the next prompt it will ask for the board which is targeted.

ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE)
Vol 4, Issue 5, May 2017

- After synthesizing and building the design, the bit file will get generated. Information about the generated bit file is displayed in the ISE design Suite command prompt as shown in Fig. 14.

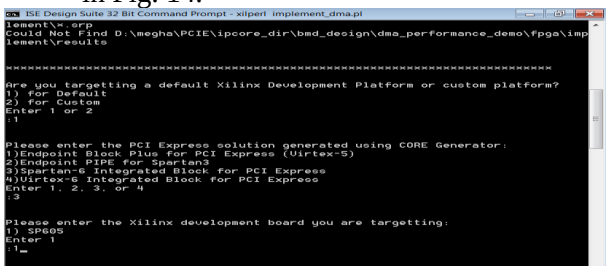

Fig. 13: Giving appropriate inputs

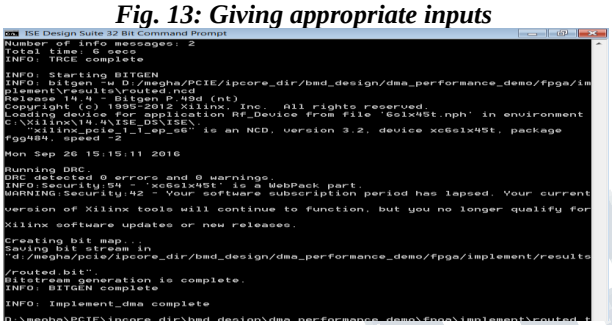

Fig. 14: Generation of bit file and completion of BMD implementation

4.3 Step 3: Programming the Spartan 6 FPGA

The Spartan 6 FPGA will be programmed through JTAG interface using the Xilinx ISE iMPACT tool. After dumping the bit file, the SP605 Evaluation kit becomes a PCI Express Endpoint device.

4.4 Step 4: Connecting SP605 Evaluation kit and PC

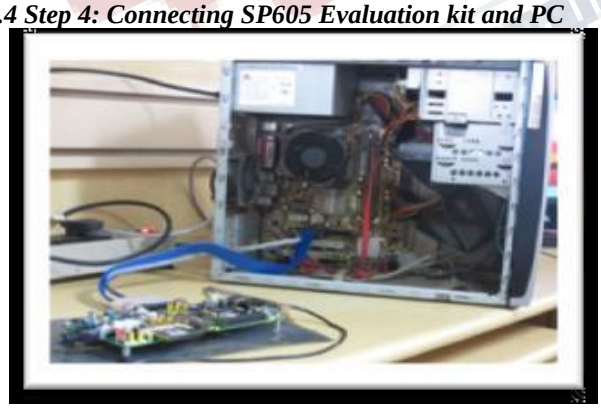

Fig. 15: SP605 Evaluation kit is connected to PCI Express slot of the motherboard

This Endpoint device is connected to the PCI Express slot of the Root Complex (PC) using a PCI Express cable as shown in the Fig. 15. Since it is new device connected to the system, the system will ask for the driver installation.

4.5 Step 5: Using DMA Driver for Windows XP

After installing the driver for PCI Express Endpoint the newly installed driver appears under device manger of the system and then selecting the newly installed driver for the PCI Express Endpoint device.

4.6 Step 6: Data transfer between PCI Express Endpoint device and PC

The GUI is launched by selecting "Performance Demo for PCIe" from the Windows start menu.

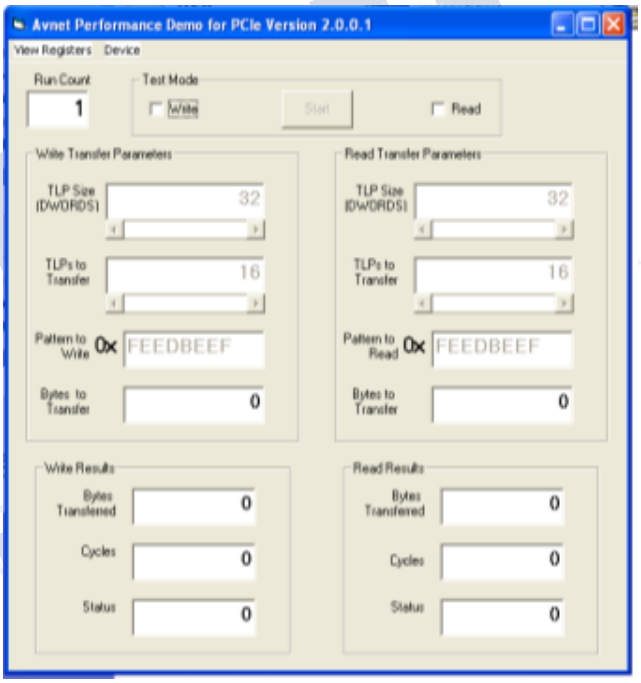

Fig. 16: Performance Demo Application GUI

If the software is installed correctly, the application GUI appears as shown in Fig. 16, and provides a description of the interface. This GUI helps in transferring data between Root Complex and Endpoint device. The write and read operation helps in understanding the functionality of the PCI Express core.

V. PCI EXPRESS IP CORE TESTING RESULT

In performing a DMA test first step is to specify the direction data transfer i.e., read or write. Next is to select the TLP size and number of TLPs to be transferred, data pattern to be transferred.

ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE)
Vol 4, Issue 5, May 2017

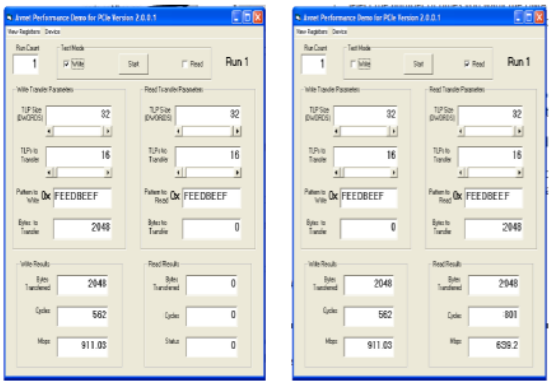

Fig (a): write operation Fig (b): Read operation
Fig. 17: BMD performance demonstrations by a loop back test i.e., write and read operation

During write operation, test mode in the application GUI is selected as write, and the data is written by the Root complex device to Endpoint device. The data is written by sending TLP of size 32 bit. The data is FEEDBEEF s shown in Fig. 17. Similarly during read operation, test mode is selected as read in the GUI and the data is read by the Root Complex device from the Endpoint device. i.e., the data is transferred from Endpoint device to the Root Complex device. The status box changes to Mbps and shows the calculated throughput if the test becomes successful. The test terminates and "FAIL" appears in the status field if the test is unsuccessful. It is observed that the same data is read by the Root Complex device which was sent by it to the Endpoint device during the write operation. Thus the loop back test is done.

VI. CONCLUSION

An application level loopback test is done for PCI Express protocol where the communication between the Root Complex and PCI Express Endpoint device is done. The Root Complex device is the PC and the Endpoint device is the SP605 Evaluation kit where the PCI Express IP core is implemented along with the Bus Master DMA (BMD) on Spartan 6 FPGA. The communication between the Root Complex device and the Endpoint device is done by performing write and read operation, where the data received by the Root Complex device from the Endpoint device during the read operation is same as the data which was sent by Root Complex device to the Endpoint device during the write operation. This completes the loopback testing.

Acknowledgment

We would like to thank our guide Ms. Suja Susan George, CPG group, BEL Bengaluru for their continuous support. And we would like to thank our college N.M.A.M. Institute of technology.

REFERENCES

[1] Ravi Budruk, Don Anderson, Tom Shanley "PCI Express System Architecture", PC System Architecture Series, pp. 9-39.

[2] Adam Wilen, Justin P. Schade, Ron Thornburg, "Introduction to PCI Express – A hardware and software guide", Intel Press 2002.

[3] Satish K. Dhawan, "Introduction to PCI Express –A New High Speed Serial Data Bus" USA, IEEE Nuclear Science Symposium Conference Record, 2005, N14-140.

[4] PCI Express Base Specification Revision 3.0 November 10, 2010

[5] Jason Lawley, "Bus Master Performance Demonstration Reference Design for the Xilinx Endpoint PCI Express Solutions", Xilinx document, 2015.

[6] Getting Started with the Xilinx Spartan-6 FPGA SP605 Evaluation Kit, Xilinx document, 2011.