DE4 NetFPGA Packet Generator Design User Guide

Revision History

Date	Comment	Author
01/30/2012	Initial draft	Harikrishnan

Contents

1. Introduction

DE4 NetFPGA packet generator is an open source port of the NetFPGA packet generator on Altera DE4 board. This tutorial describes the steps to download the DE4 NetFPGA packet generator design from the project website, compile the design using Altera Quartus II software, and to verify the packet generator.

2. System Requirements

To successfully install and test Packet Generator on DE4, you must satisfy the following system requirements.

Operating System Requirements

- Microsoft Windows XP

Software Requirements

- Altera Quartus II 11.0 or later and SOPC Builder
- Microsoft Visual Studio 2010
- Jungo Windriver 32bit

Hardware Requirements

- PC with PCIe x8 or x16 slot.
- Altera DE-4 board
- Gigabit Ethernet Cables for testing purpose.

IP License Requirements

- Altera Triple Speed Ethernet MAC (TSE MAC) IP Core

Table 1 provides additional information on obtaining the hardware and software necessary to compile DE4 NetFPGA packet generator.

Table 1: Software and Hardware Requirements

Requirement	Source
Altera Quartus II 11.0 and SOPC Builder	https://www.altera.com/download/software/quartus-ii-se
Microsoft Visual Studio 2010	http://www.microsoft.com/visualstudio/en-us/try
Jungo Windriver	http://www.jungo.com/st/windriver_usb_pci_driver_development_software.html
Altera DE-4 board	http://www.terasic.com.tw
Altera Triple Speed Ethernet MAC (TSE MAC) IP Core	http://www.altera.com/support/ip/interface-protocols/ips-inp-tse.html

Note: Altera provides a free time limited Open Core Plus core of Triple Speed Ethernet MAC (TSE MAC) with a time limit of 1 hour. This core is available as part of the Altera Quartus II software.

3. Installing DE4 NetFPGA Board in the Host PC

To operate Altera DE4 in PCI Express mode, perform the following two steps:

1. Select the 100MHz clock by setting SW7 switch to 00 (both ON) position as shown in Figure 1.

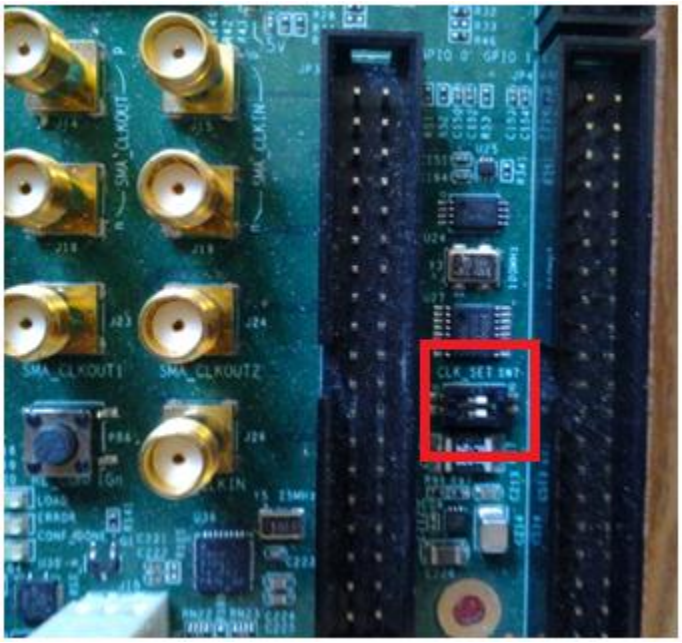


Figure 1: Select 100 MHz clock

2. Enable PCI mode by setting the SW0 switch in **OFF** position as shown in Figure 2.

3. Configure PCI Express for Gen1 operation. To enable Gen1, set the PCI Express Control DIP switch SW9.1 in **ON** position as shown in Figure 3.

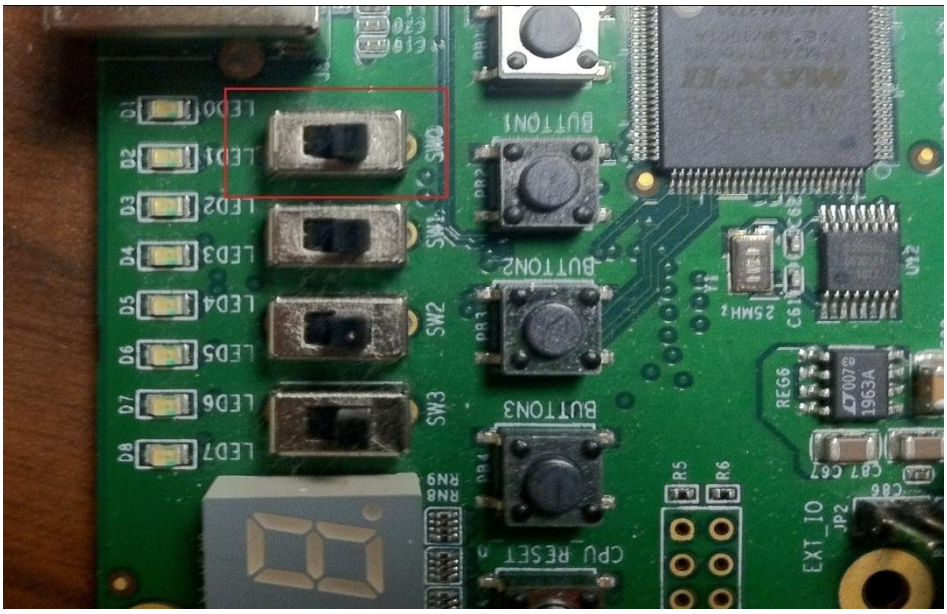


Figure 2: SW0 switch OFF position

Figure 3: PCI Express Control DIP Switch

The PCI Express mode does not require a standalone power supply. Instead, the Altera DE4 board can be powered up through the x8 or x16 PCI Express edge connector attached to a PC as shown Figure 4.

Figure 4: PCI Express Mode

4. Obtaining DE4 NetFPGA Packet Generator

The NetFPGA packet generator [3]. is a simple packet generator and capture tool for education and research purposes from Stanford University. The DE4 NetFPGA packet generator provides a port

of the NetFPGA packet generator on the Altera DE4 platform. Download the gateware and software for DE4 NetFPGA packet generator as bundled as two separate packages:

- Download **DE4 NetFPGA directories package (NF2_DE4_BASE.zip)** provided by University of Massachusetts, Amherst from [13]. A link to the repository is also available from the project website [1].
- Obtain the **DE4 Packet Generator package (DE4_Packet_Generator_v1.0.zip),** a modified version of Stanford University's NetFPGA packet generator design by sending an e-mail to Professor Russell Tessier (email:tessier@ecs.umass.edu).

5. Unzipping Files

- Unzip NF2_DE4_BASE.zip and DE4_Packet_Generator_v1.0.zip to the same directory. Unzipping NF2_DE4_BASE.zip will create the following directory structure:

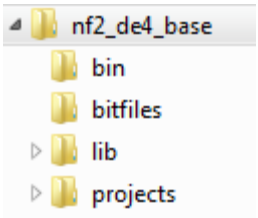


Figure 5: Base directory structure

- Unzipping **DE4_Packet_Generator_v1.0.zip** will merge DE4_Packet_Generator source files with the "lib" and "project" directories in the base package. These directories are shown in Figure 6 and Figure 7.

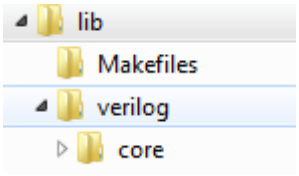


Figure 6: lib directory structure

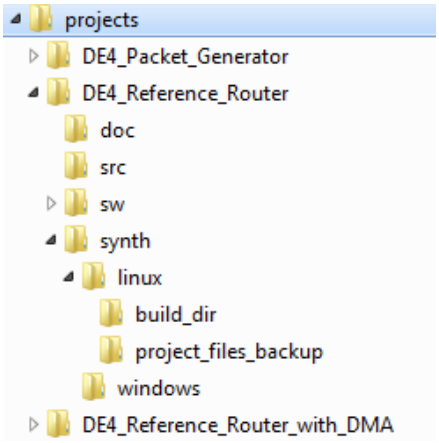


Figure 7: Packet Generator directory structure

6. Directory Contents

The nf2_de4_base folder contains the following subdirectories –

- **bin** - Scripts to build the project and download the bit-stream with Linux build.
- **bitfiles** - Bit-streams generated in Linux environments are copied after compilation.
- **Lib** - Contains source files and megafunctions that are common for all projects.
- **Projects** – Project specific source files and compilation environment.

The lib folder contains the following subdirectories –

- **Verilog/core** – Source files and megafunctions that are common across all projects
- **Makefiles** - Makefiles required for the Linux compilation.

The projects folder contains the following subdirectories –

- **src** – Project specific verilog files must be placed here. Source files placed in this folder override the files with the same name in **lib/verilog/core**. In Linux, these files will be automatically sourced from Makefiles. In Windows, if users need to create/replace a module in 'lib/verilog/core', they should place the files in 'src' and manually add them to the project (the corresponding 'lib/verilog/core' files have to be removed manually).
- **sw** - Contains two subdirectories - **jtag_config** and **pcie_config**. The 'jtag_config' folder contains TCL scripts for configuring the Gigabit Ethernet and reference router tables in JTAG mode of operation while 'pcie_config' contains the Visual Studio software application for the same in PCIE mode.
- **Synth** – Contains two subdirectories 'linux' and 'windows'. In 'windows' directory, the Altera project design files for compiling the project in Windows are included. Similarly in 'linux', 'build_dir' has the makefile to compile the project; 'project_files_backup' has the Altera project files that are copied into 'build_dir' when "***make clean***" command is used.

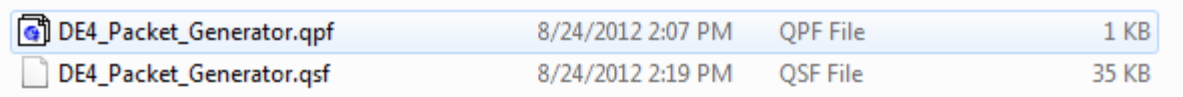
7. Compiling DE4 NetFPGA Packet Generator

Compiling the DE4 NetFPGA packet generator requires two separate packages as described in Section 4:

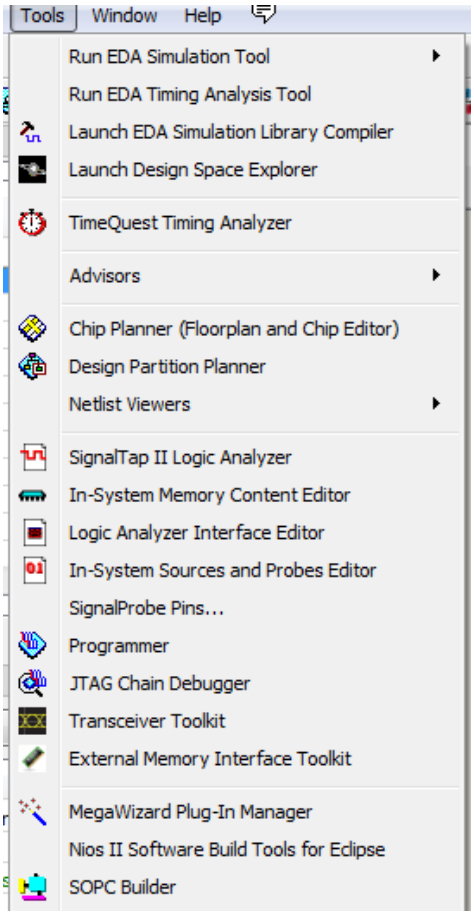
- DE4 NetFPGA directories package (**NF2_DE4_Base.zip**).
- A modified version of the NetFPGA Packet Generator RTL source files (**DE4_Packet_Generator_v1.0.zip**).

The detailed compilation steps are described below:

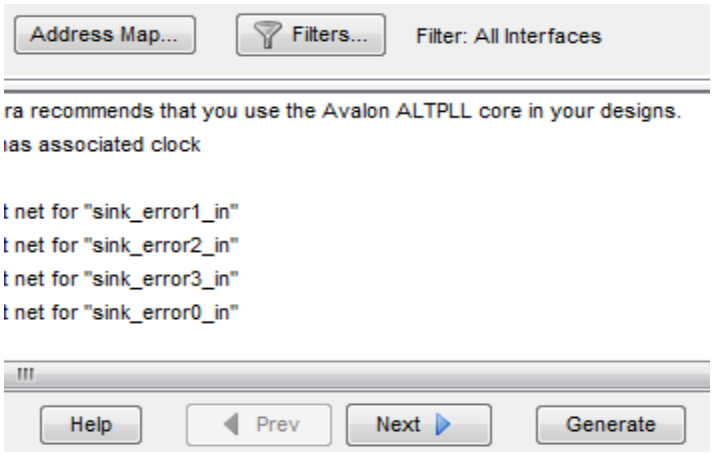
1. Open the DE4_Ethernet project in Quartus II by double clicking **NF2_DE4_BASE/projects/DE4_Packet_Generator/synth/windows/DE4_Packet_Generator.qpf**



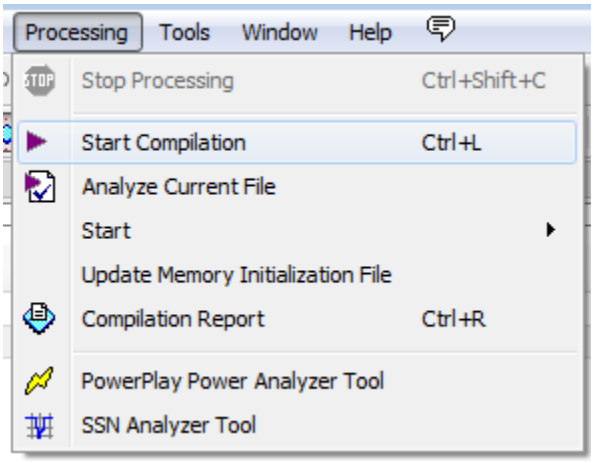
2. Open SOPC builder (Quartus → Tools → SOPC Builder)



3. Generate the SOPC System. Click Generate on the SOPC builder

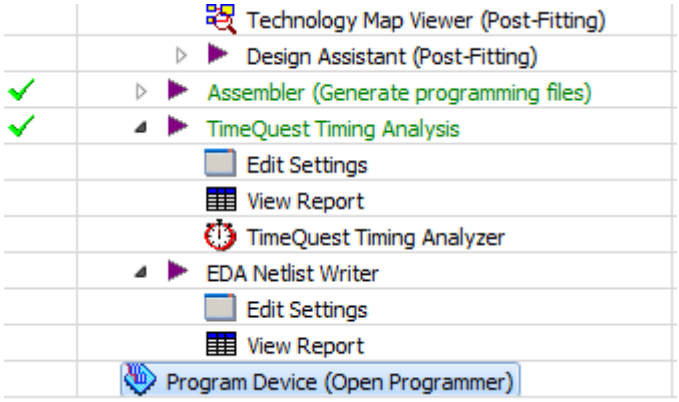


4. Once SOPC system generation is successful, compile the design from Quartus (In Quartus, Processing → Start Compilation).

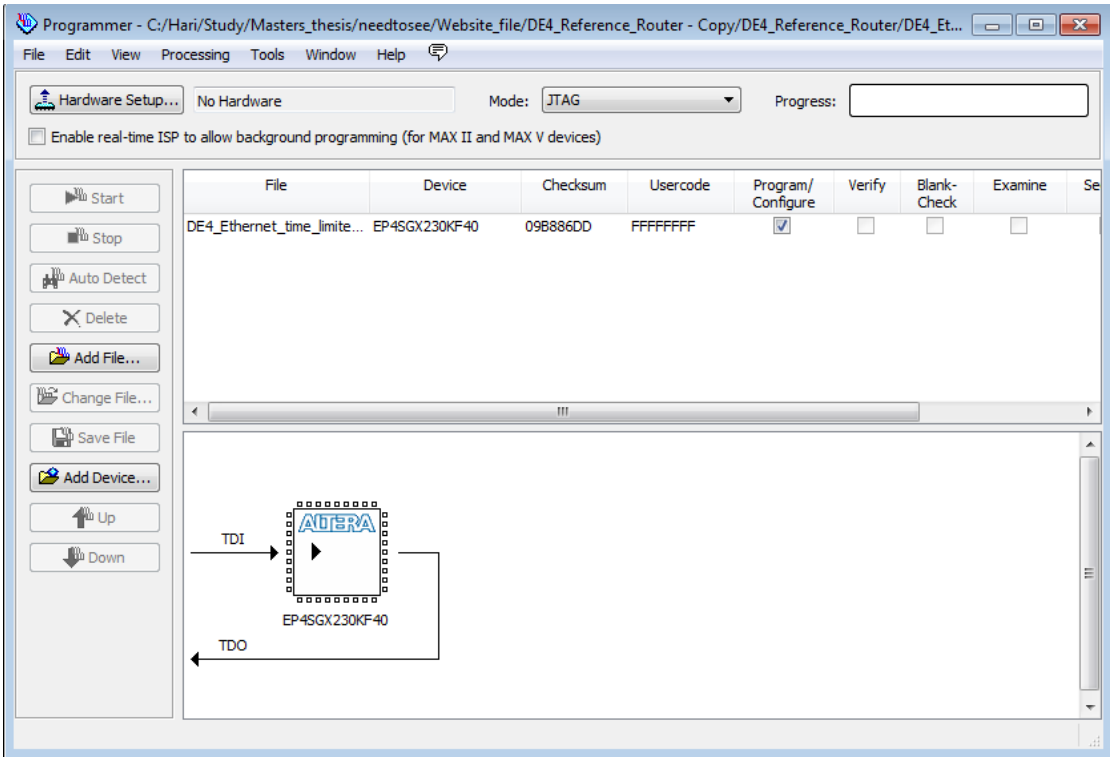


5. Install the DE4 board on the PCIe slot (Section 3). Make sure that the board is also connected to PC via the USB/JTAG cable.

6. Download the bit-file generated after the compilation to the target DE4 board by double clicking the "Program Device" in the Task Window.



7. Click Start in the Programmer window to begin download.



8. Reboot the host PC after downloading the bit-file.

8. Testing the DE4 NetFPGA Packet Generator

8.1. Configuring the DE4 NetFPGA Packet Generator

The DE4 NetFPGA packet generator can be used only in PCIe mode. After configuring the Altera DE4 board to be used in PCIe mode, the FPGA bit-stream must be downloaded through USB/JTAG cable. Now the packet generator functionality can be tested using the PCIe software once the Gigabit Ethernet registers are configured via the JTAG bus,

The Gigabit Ethernet registers can be configured as follows:

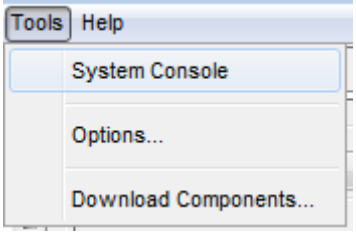
1. Bring up the Gigabit Ethernet Interfaces (hereafter referred to as **GigE Config**)
2. Set up the PCIe software

Bringing up Gigabit Ethernet Interfaces:

Configure the Broadcom PHY and Altera Triple Speed Ethernet MAC with the appropriate configuration values as follows:

1. Open SOPC Builder from Quartus II (In Quartus, select Tools → SOPC Builder)

2. In SOPC Builder, open system console (Tools → System Console).

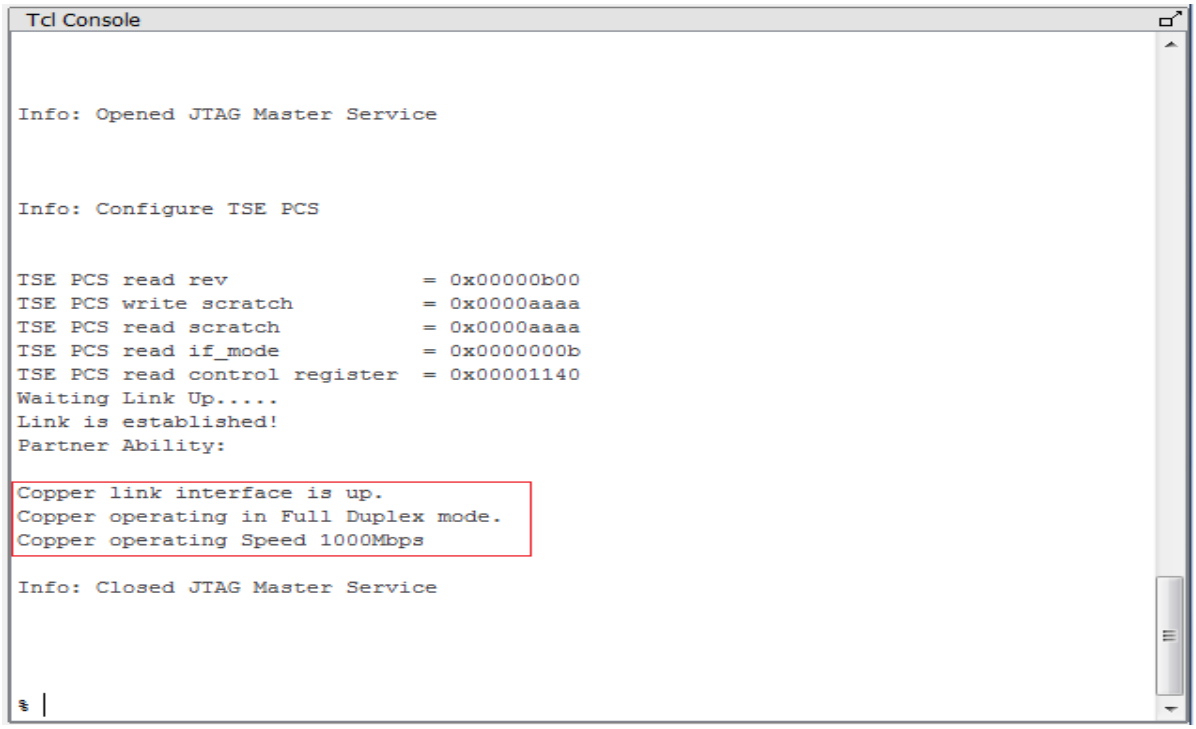


3. Change the current directory (synth/windows/) to **jtag_config**

   > **cd ../../sw/jtag_config**

4. Run the TCL script for configuring MAC and PHY for the four Gigabit Ethernet interfaces.

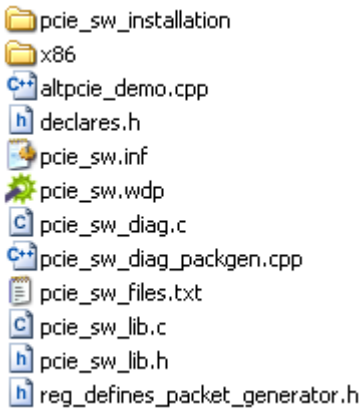
   > source config_gige.tcl



Setting up PCIe software:

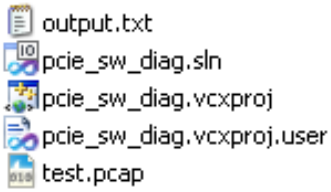
We use Jungo's Win Driver [7]. 32-bit to establish communication between the host PC running Windows XP and the Altera DE-4 board. Jungo's Win Driver exposes two APIs (OnRCSlaveWrite and OnRCSlaveRead) to applications to read and write using the PCI Express bus. We have developed a Visual Studio 2010 based application that uses these APIs to read and write the packet generator registers on the DE4 board. This application is available in **{project root}\sw\pcie_config** directory. **{project root}** refers to NF2_DE4_BASE/projects/DE4_Packet_Generator.

The file structure of the **pcie_config** folder is shown below:

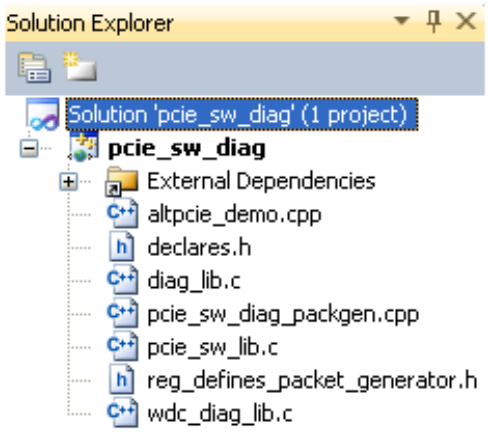


Perform the following steps to setup the PCIe software.

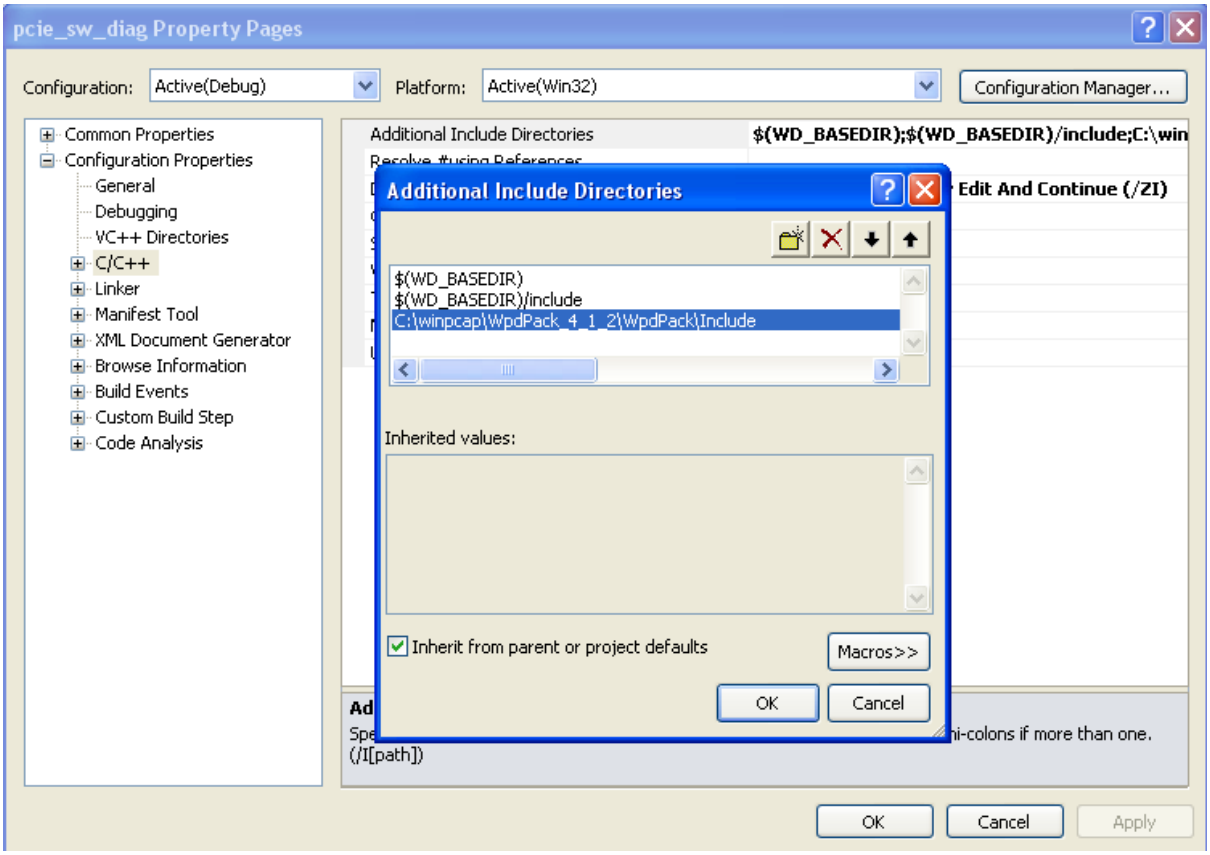
1. Navigate to **{project root}\sw\pcie_config\x86\msdev_2010**
2. Double click on pcie_sw_diag.sln



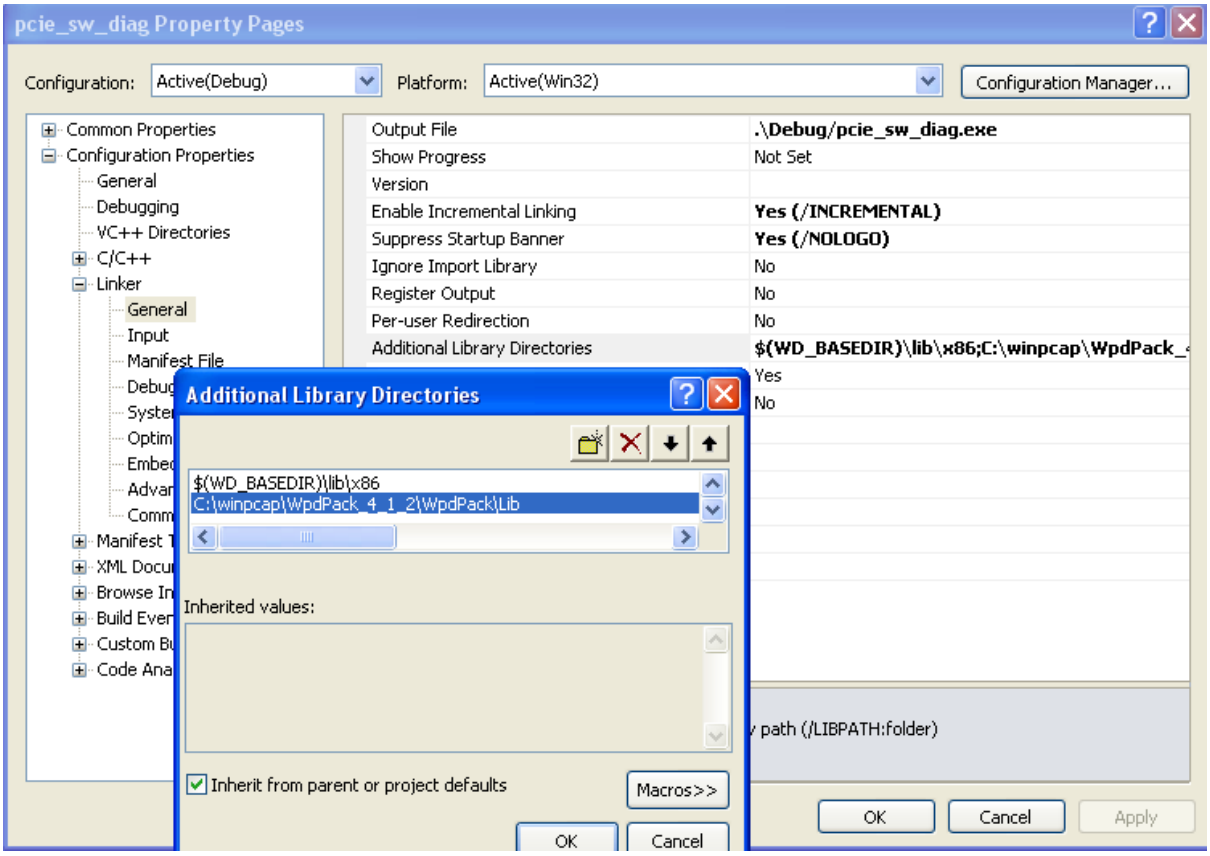
3. The Visual Studio 2010 window pops up with the following file structure.



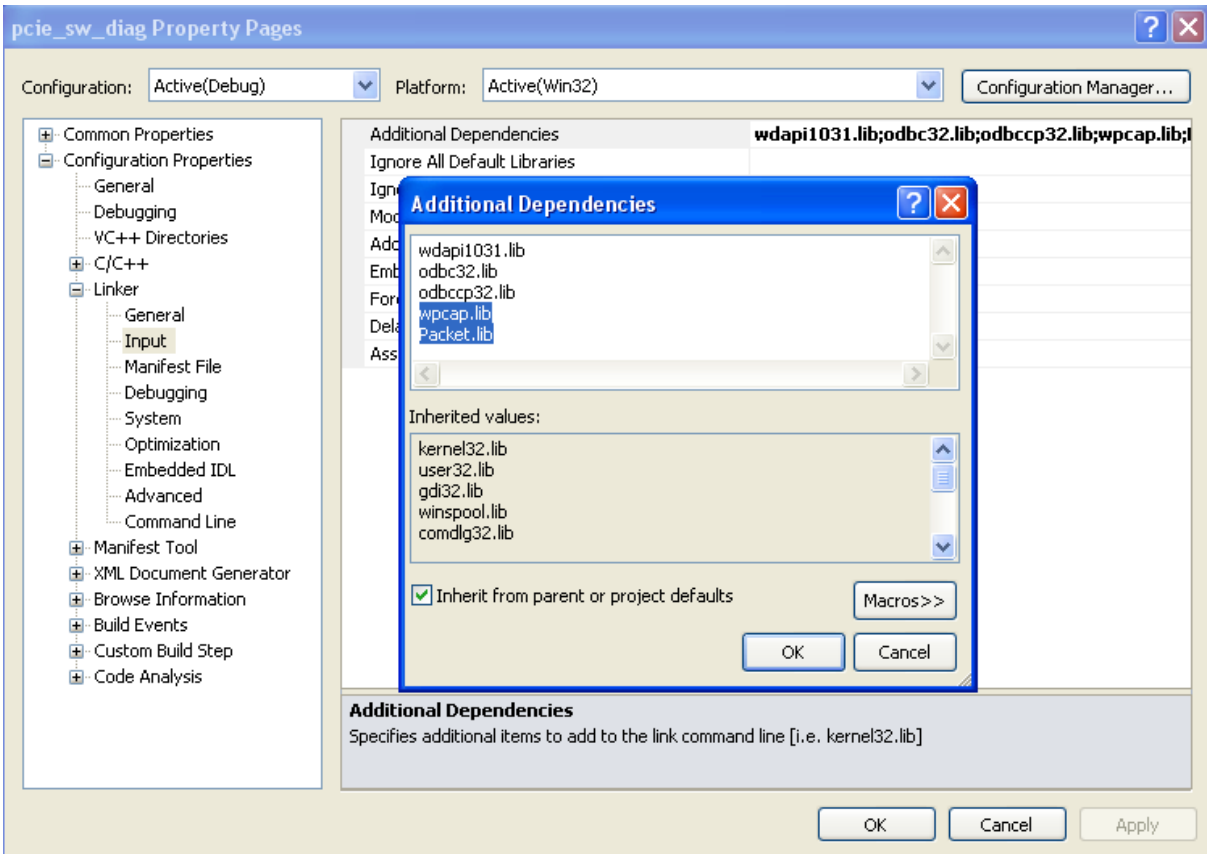
4. Install WinPcap driver [9]. and WinPcap developer's pack [10]. . The developer's pack is installed in the location "C:\WpdPack_4_0_2\WpdPack".
5. Link Visual Studio with the WinPcap libraries as mentioned below.
6. In Visual Studio, under the Configuration Properties → C/C++ → General tab, add the WinPcap include path to Additional Include Directories.



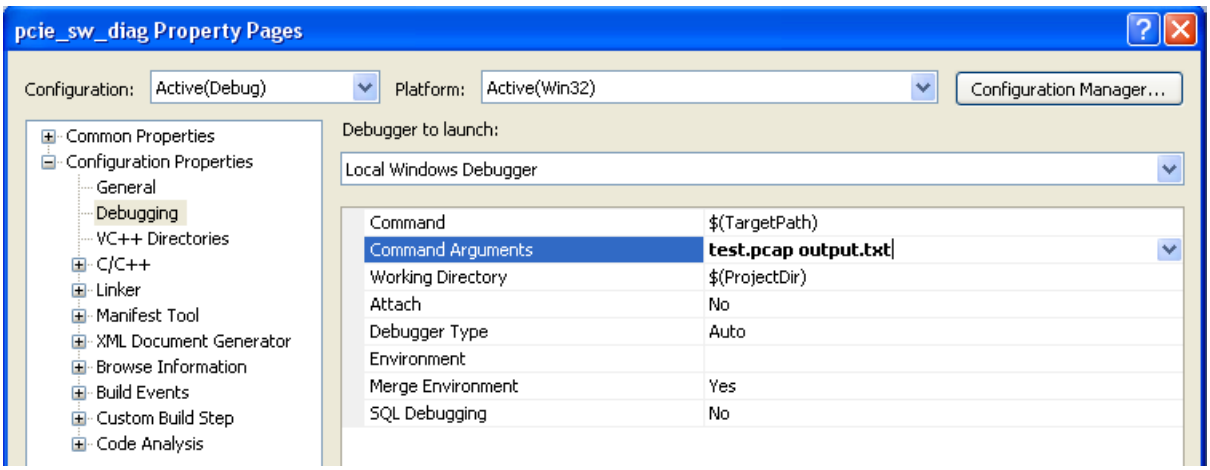
7. Under the Configuration Properties → Linker → General tab, add the WinPcap library path to Additional Library Directories.



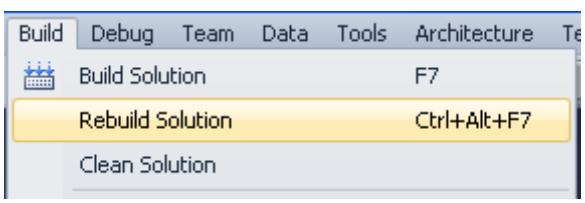
8. Under the Configuration Properties → Linker → Input tab, add the two main WinPcap libraries (wpcap.lib & Packet.lib) to Additional Dependencies.



9. Under the Configuration Properties → Debugging tab, add the files (test.pcap and output.txt) required for running the test.



10. Rebuild the design: In Visual Studio, click Build → Rebuild Solution.



The DE4 NetFPGA packet generator can be tested once the packet generator bit-file has been downloaded into the FPGA and the gigabit Ethernet registers are configured. The packet generator can be tested by sending sample packets and collecting the packets back in a loopback formation. The user can load a custom PCAP file into the DE4 NetFPGA packet generator (test.pcap) and subsequently transmit the packets in the PCAP file at user defined rates through DE4 NetFPGA ports.

8.2. Test Setup – Packet Generator

1. Install the DE4 board in the PCIe slot and connect the Ethernet cable in loopback formation as shown in Figure 8
2. Download the bit-file on the DE4 board through the JTAG cable and restart the PC
3. Setup the Gigabit Ethernet registers
4. Setup the PCIe software in Visual Studio.
5. Configure the required input parameters (rate, iterations, MAC queue) in the **{project root}\sw\pcie_config\declares.h** file.
6. Rebuild the design

8.3. Running the Test

A pcap file is provided {pcie_config/x86/msdev_2010/test.pcap} for testing. The packet generator is configured in loopback according to Figure 8. The experimental setup is shown in Figure 9.

.

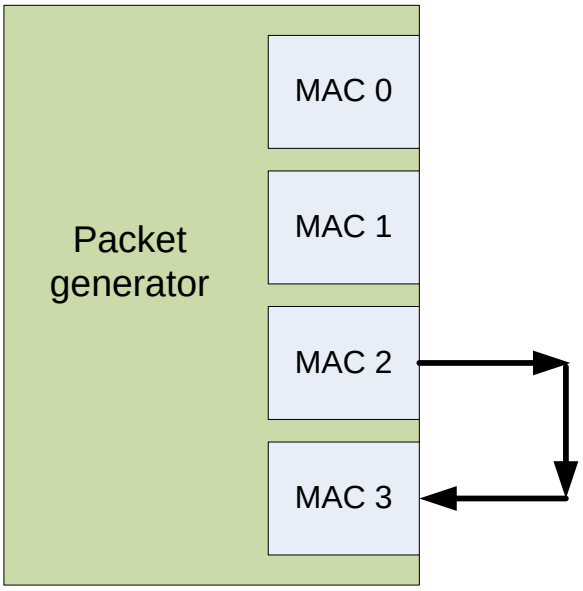


Figure 8: Packet Generator Test Setup

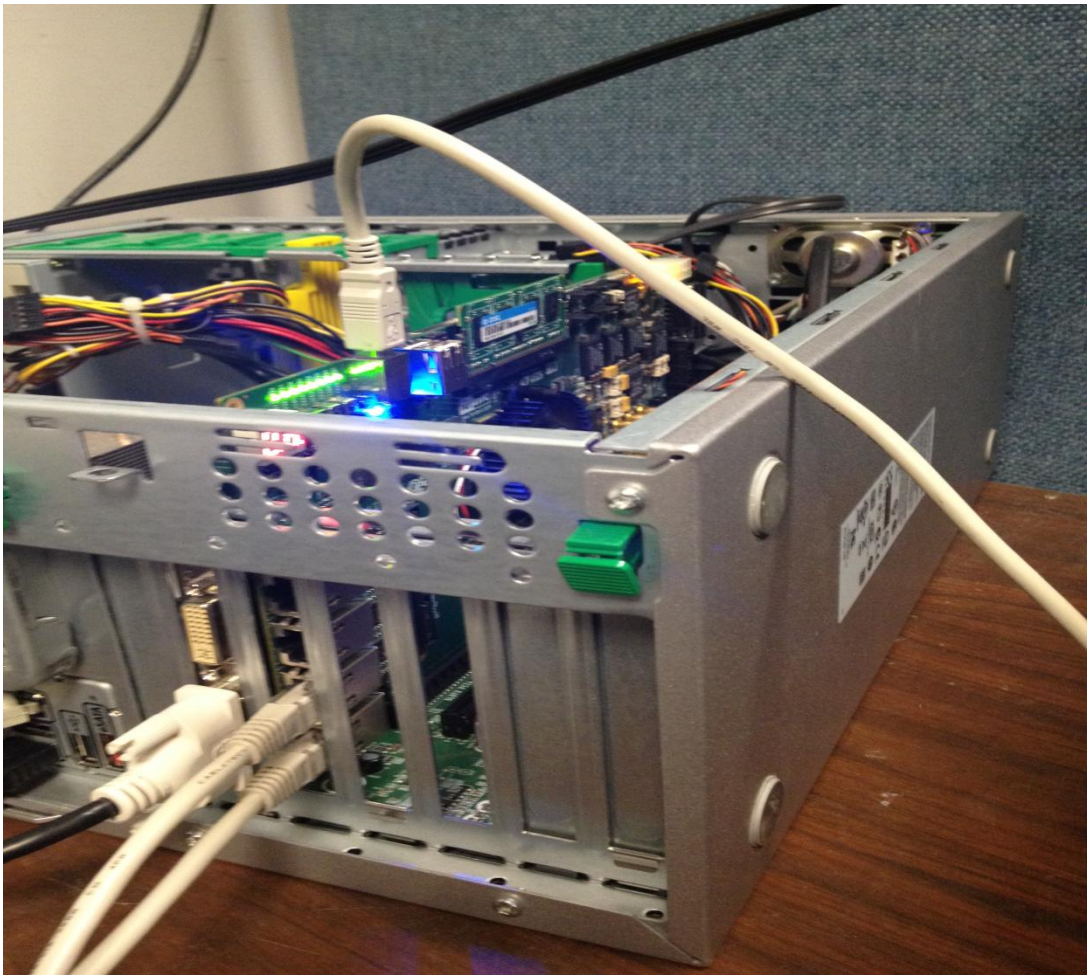


Figure 9: Packet Generator Experimental Setup

Run the packet generator by clicking on Start Debugging in Visual Studio as shown in Figure 10. This command sends packets through the specific Ethernet interface queue **(MAC_ENABLE)** at a user defined rate **(RATE_MAC)** with a user defined delay **(DELAY_MAC)** for a specific number of iterations **(ITER_MAC)**. All the parameters (**MAC_ENABLE, etc**) are defined in the declares.h file.

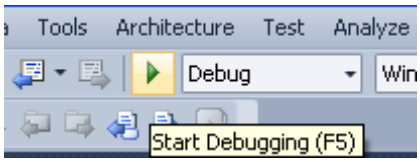


Figure 10: Start Debugging

Example:

To test packet transmission with 32 packets at line rate (1Gbps) through queue 2, use the following input parameters:

define RATE_MAC0 = 0;
define RATE_MAC1 = 0;
define RATE_MAC2 = 1000000;
define RATE_MAC3 = 0;
define ITER_MAC0 = 0;

```
# define ITER_MAC1 = 0;
# define ITER_MAC2 = 32;
# define ITER_MAC3 = 0;
# define MAC_ENABLE = 2;
```

When the topology as shown in Figure 8 is used, the forwarded packets are captured at queue 3 of the DE4 NetFPGA packet generator. The packet capture statistics is observed in **pcie_config/x86/msdev_2010/output.txt** as shown in Figure 11.

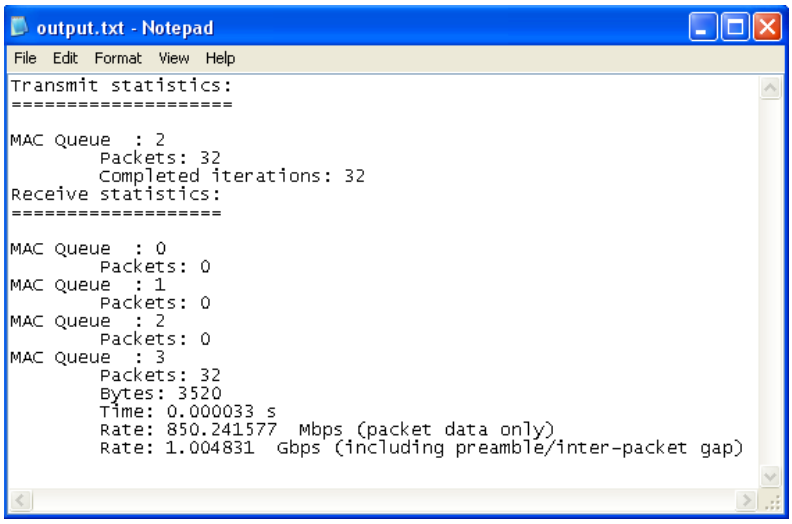


Figure 11: Sample packet capture at line rate (1Gbps)

9. Reference router-Packet generator test setup

The test setup for testing the DE4 NetFPGA reference router-Packet generator system is illustrated in Figure 12.

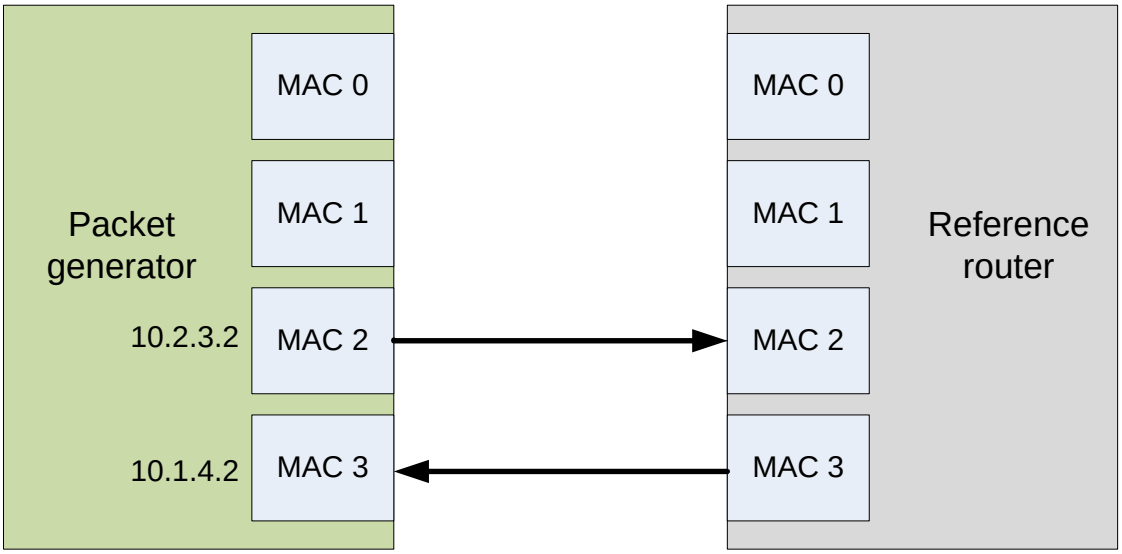


Figure 12: DE4 NetFPGA Reference router-Packet generator setup

The steps for testing the above test topology are explained below.

1. Connect Reference router and Packet generator as per Figure 12
2. Setup the reference router as per DE4 reference router user guide
3. Setup the packet generator
4. Run the PCIe software
5. Observe the output from the **output.txt** file

The experimental test setup is illustrated in Figure 13.

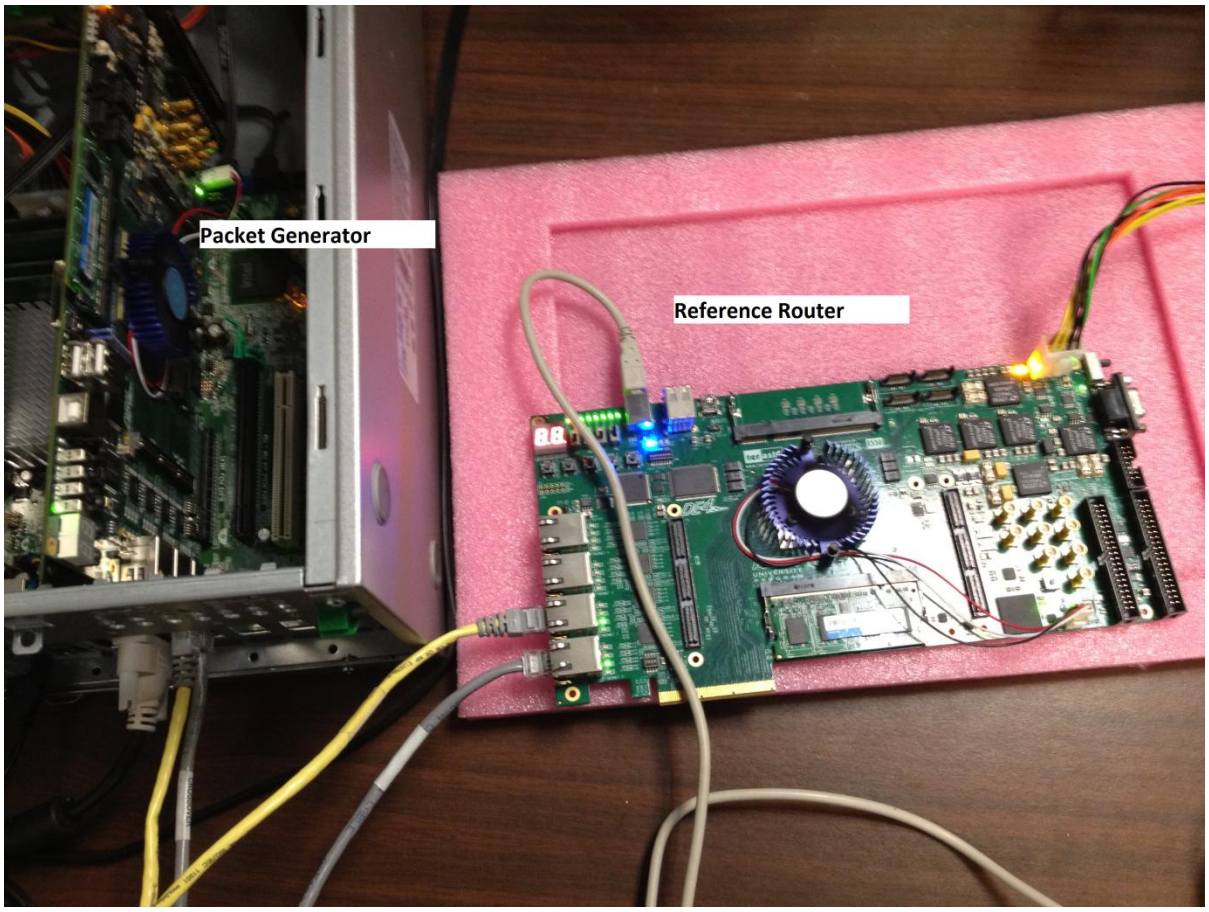


Figure 13: DE4 NetFPGA Reference router-Packet generator experimental setup

10.References

[1]. DE4 NetFPGA Project Web, http://keb302.ecs.umass.edu/de4web/DE4_NetFPGA/

[2]. NetFPGA Project, http://netfpga.org/

[3]. NetFPGA packet generator http://netfpga.org/foswiki/bin/view/NetFPGA/OneGig/Projects/PacketGenerator

[4]. Altera Triple Speed Ethernet MAC, http://www.altera.com/literature/ug/ug_ethernet.pdf.

[5]. Altera PCI Express, http://www.altera.com/literature/ug/ug_pci_express.pdf

[6]. Altera PCI Express Wiki, http://www.alterawiki.com/wiki/PCI_Express_in_Qsys_Example_Designs

[7]. Jungo USB Driver, http://www.jungo.com/st/windriver_windows.html

[8]. Terasic Technologies, http://www.terasic.com.tw/en/

[9]. WinPcap for Windows, http://www.winpcap.org/install/default.htm

[10]. WinPcap Developer Resources, http://www.winpcap.org/devel.htm

[11]. Altera Quartus II 11.0 and SOPC Builder, https://www.altera.com/download/software/quartus-ii-se

[12]. Microsoft Visual Studio 2010, http://www.microsoft.com/visualstudio/en-us/try

[13]. DE4 Base repository https://github.com/UmassRCG/nf2_de4_base